AIMS Mathematics, 9(2): 2781–2807.
DOI: 10.3934/math.2024138
Received: 08 October 2023
Revised: 24 December 2023
Accepted: 26 December 2023
Published: 29 December 2023

http://www.aimspress.com/journal/Math

Research article

A swarm intelligence-based ensemble learning model for optimizing customer churn prediction in the telecommunications sector

Bijan Moradi[1], Mehran Khalaj[1,*], Ali Taghizadeh Herat[1], Asghar Darigh[2] and Alireza Tamjid Yamcholo[3]

[1] Department of Industrial Engineering, Islamic Azad University, Parand and Robat Karim Branch, Tehran, Iran
[2] Department of Mathematics, Islamic Azad University, Parand and Robat Karim Branch, Tehran, Iran
[3] Department of Computer Science and Information Technology, Islamic Azad University, Parand and Robat Karim Branch, Tehran, Iran

* **Correspondence:** Email: m.khalaj@piau.ac.ir.

Abstract: In today's competitive market, predicting clients' behavior is crucial for businesses to meet their needs and prevent them from being attracted by competitors. This is especially important in industries like telecommunications, where the cost of acquiring new customers exceeds retaining existing ones. To achieve this, companies employ Customer Churn Prediction approaches to identify potential customer attrition and develop retention plans. Machine learning models are highly effective in identifying such customers; however, there is a need for more effective techniques to handle class imbalance in churn datasets and enhance prediction accuracy in complex churn prediction datasets. To address these challenges, we propose a novel two-level stacking-mode ensemble learning model that utilizes the Whale Optimization Algorithm for feature selection and hyper-parameter optimization. We also introduce a method combining *K*-member clustering and Whale Optimization to effectively handle class imbalance in churn datasets. Extensive experiments conducted on well-known datasets, along with comparisons to other machine learning models and existing churn prediction methods, demonstrate the superiority of the proposed approach.

Keywords: customer churn prediction; telecommunication industry; dataset balancing; feature selection; ensemble learning; Whale Optimization Algorithm

1. Introduction

Nowadays, companies across different sectors prioritize customer satisfaction through customer relationship management (CRM) strategies to boost retention rates [1]. Predictive models play a crucial role in identifying potential customer churn, enabling targeted incentive plans in various industries like banking [2–4], retail [5–7] and telecommunications [8–18]. Telecommunications is an industry that faces serious competition and customers consistently compare and switch providers for better quality services and cost savings. Therefore, effective customer churn prediction (CCP) approaches are vital for identifying attrition signs and reducing churn rates through satisfaction strategies, ensuring sustainability in the telecom market. However, addressing the CCP problem in telecom is challenging due to: 1) Large CCP dataset instances, 2) numerous customer attributes (demographics, service logs, bill info) and 3) imbalanced churn data with few churn cases compared to overall customers.

Machine Learning (ML) techniques are among the highly adopted and most potent approaches for tackling the CCP problem. This is mainly due to the capability of ML algorithms to automatically learn the complex relations among input data and their corresponding outputs and provide high-quality predictions. The ML techniques that are widely studied include Logistic Regression [19], Decision Trees [20], Naïve Bayes [21], Support Vector Machines [22] and Artificial Neural Networks [23], among others. In addition to the above-mentioned techniques, ensemble learning models such as Random Forest [24], Adaboost [25], Gradient boost [26], XGBoost [27], etc. have also been applied to address the CCP problem by integrating numerous ML models (called base-learners) in their structure. This is mainly because of the ensembles' ability to compensate for the shortcomings of every single base-learner by aggregating the strengths of multiple ML models.

Additionally, to mitigate the negative effects of a large number of attributes in customer churn datasets, feature selection techniques are vital in CCP approaches. These techniques aim to identify the most relevant and impactful subset of features, falling into two categories: filter and wrapper methods. Filter methods select convenient features based on statistical criteria, without directly considering their impact on the prediction model. In contrast, wrapper methods use feedback from ML prediction results to select input features through a search algorithm. Meta-heuristic algorithms, such as evolutionary optimization and swarm intelligence, excel in improving ML model performance by optimizing structure, parameters, instances and features [28,29]. These algorithms can effectively search for the subset of features maximizing the desired fitness function defined on the ML model outputs, achieving near-optimal results in fewer iterations.

Imbalance in a dataset refers to a significant difference in sample numbers between classes. In customer churn datasets, this translates to a scarcity of samples from customers who have left compared to those who remain. Consequently, the class distribution in churn datasets becomes heavily skewed, resulting in a pronounced imbalance. This imbalance adversely affects the training process of ML models. Two major techniques, random over-sampling and random under-sampling, are commonly used to address class imbalance in CCP datasets [30,31]. Random over-sampling adds minority-class instances (churn instances) to the dataset, aiming to alleviate class imbalance. However, it risks overfitting during model training, leading to reduced overall accuracy [32]. On the other hand, random under-sampling balances the dataset by randomly removing majority-class instances (non-churn instances). Nevertheless, this technique may discard useful data, compromising the efficiency of the final prediction model. To overcome these challenges, modified sampling techniques such as Synthetic Minority Over-Sampling Technique (SMOTE) [17], Mega-trend Diffusion Function (MTDF) [33],

Adaptive Synthetic Sampling Approach (ADASYN) [34], Minimal Redundancy Maximal Relevance (mRMR) [35], Weighted Minority Oversampling Technique (MWMOTE) [36] and PSO-based sampling technique [15] have been introduced to address the limitations of random sampling methods.

Despite significant advancements in customer churn prediction, there are important research gaps that need to be addressed. These gaps include the need for more effective techniques to handle class imbalance in CCP datasets and the requirement for a deeper understanding of intricate relationships within customer data to improve predictive accuracy and generalization of ML models. To tackle these challenges, this research proposes a customer churn prediction method called Swarm Intelligence-Based Ensemble Learning (SIBEL). SIBEL benefits from a combination of *K*-member clustering and Whale Optimization Algorithms (WOA) to effectively balance the churn dataset and address class imbalance issues. Furthermore, we suggest an ensemble learning model based on the two-level stacking technique, incorporating six prominent ML algorithms (RF, MLP, SVM-RBF, KNN, NB and LR) as base-learners in each level. The primary motivation behind employing a stacking ensemble lies in the abilities of meta-models (models that are used to combine the predictions of the base-learners) which can effectively capture and learn the complex and non-linear relationships present in CCP datasets. Therefore, the main hypothesis guiding this research is that the utilization of a hybrid clustering-WOA based technique to handle class imbalance and a two-level stacking model can enhance the prediction accuracy of the CCP problem.

In the proposed approach, we first preprocess the datasets by removing irrelevant attributes, handling missing data, transforming data and scaling features. Next, the customer churn datasets are split into a training and a test set for fitting and evaluating the model. The training set is then balanced by employing a novel hybrid under-sampling technique that combines *K*-member clustering and the WOA based instance selection. Subsequently, once again the WOA is utilized to optimize the prediction capability of the suggested stacking ensemble model by finding an appropriate subset of features for each base-learner and also tuning the model's hyper-parameters. The fitness function used in this phase includes three parameters by which the user can adjust the weights of the precision, recall and AUC criteria in accordance with the desired policies of the telecom company. Our major contributions can be listed as follows:

- A hybrid Clustering-WOA based approach is developed which utilizes *K*-member clustering and WOA to balance the churn dataset by under-sampling the instances of the majority class.
- We propose a novel customer churn prediction model called SIBEL, which benefits from a two-level stacking ensemble learning.
- Six remarkable ML algorithms (MLP, SVM-RBF, RF, NB, KNN and LR) are incorporated in each level of the proposed ensemble model which can precisely learn the customers' attrition behavior from a customer churn dataset and properly predict the customers who intend to leave a telecom company.
- The WOA optimizes the performance of the proposed two-level ensemble model by simultaneously searching for the best subset of features for base-learners and also adjusting the weights in the final stage voting of the SIBEL.
- SIBEL is an application-specific approach in the sense that the fitness function of the WOA is defined in a way that the user can set the weight of each evaluation criterion (precision, recall and AUC) based on the application specifications and the policies of the company.

The rest of the paper is organized as follows: initially, in Section 2, we review several important previous studies that employ ML algorithms to approach the CCP problem in the telecom market. In

Section 3, we provide the fundamentals of the WOA and in Section 4 we introduce the specifications of customer churn datasets employed in this study. In Section 5, we present our proposed methodology for addressing the CCP problem. Subsequently, the numerical results of applying the proposed approach to the churn datasets are reported in Section 6 and it is compared with other well-known models. Eventually, Section 7 concludes the paper and highlights several guidelines for future studies.

2. Literature review

In this section, we review several remarkable publications that utilized ML algorithms (especially ensemble learning models) to approach the CCP problem in the telecom industry. Additionally, we discuss studies that take advantage of optimization algorithms to enhance their introduced approach.

The research by Mozer et al. [8] explored the prediction of customer churn using ML models such as DT, ANN, LR and ensemble learning models based on boosting techniques. The study utilized a dataset of 47,000 local mobile phone subscribers, consisting of information on consumption history, billing, credit, usage and complaints. The experimental findings indicated that the ANN model outperformed the DT and LR models in predicting subscriber churn in the dataset.

Considering that a large number of features in the churn datasets reduces the quality of predictions, Hadden et al. [9] conducted a thorough analysis of feature selection techniques for solving the CCP problem. They examined popular ML models such as DT, regression analysis, ANN and Markov models. The study revealed three key advantages of feature selection: Improved classification performance by selecting informative features, reduced processing time and the suitability of lower-dimensional datasets, particularly for neural networks. The researchers also emphasized the significance of optimization techniques, such as genetic algorithms, in enhancing the performance of CCP. Additionally, they recommended exploring Naïve Bayes and fuzzy inference systems for future research in customer churn prediction.

In light of the capabilities of SVM models, Coussement et al. [10] conducted a comprehensive study on their use in the field of CCP. The research compared cross-validation and grid-search parameter selection methods to determine the more effective approach for tuning SVM parameters. Subsequently, the SVM models are compared with LR and RF methods in terms of prediction accuracy. The obtained results in this study illustrate that SVMs show high performance in the face of noisy data. The results also revealed that when optimal parameters were selected, SVM outperformed LR, while RF performed better than the SVM model.

To address the problems related to the class imbalance in CCP datasets, Burez and Van Den [11] focused on this issue. They specifically addressed this issue by analyzing and comparing the results obtained from different strategies for balancing the dataset, including random and advanced under-sampling techniques. The study implemented weighted RF and gradient boosting prediction models and evaluated the results using metrics such as AUC and Lift. The findings demonstrated that the utilization of advanced under-sampling techniques had positive effects over random techniques.

As one of the first studies on using meta-heuristic algorithms in addressing the CCP problem, Pendharkar [12] has proposed two ANN models utilizing the GA to optimize the churn prediction accuracy among telecom subscribers. The first proposed model utilizes a cross-entropy criterion for CCP, while the second model employs the GA to directly maximize the accuracy of churn prediction. The results of applying the models on real-world cellular service datasets and comparing them using the Z-Score show that the models utilizing GA achieve better results in terms of accuracy, lift decile

above 10%, and AUC criteria. In addition, the obtained results indicate that medium size neural networks perform best and the criterion based on cross-entropy exhibits greater resistance to overfitting during the training process.

In another study on using ensemble learning algorithms, Idris et al. [13] have shown that the use of ensemble models, especially boosting methods comprising DTs, can achieve high-quality results in solving the CCP problem. However, the large size of the churn datasets, their imbalanced nature and a large number of features, especially in the telecom datasets, confront the classification algorithms with challenges in predicting churns accurately. Therefore, to deal with the mentioned problems, researchers have presented a solution based on combining genetic programming and the Adaboost model to tackle the problem of predicting churns in the telecom industry. The prediction accuracy of the proposed method is compared with KNN and RF methods, which shows its superiority.

Vafeiadis et al. [14] presented a comprehensive comparative study on the ML methods used for the CCP problem in the telecom industry. In this research, common machine learning models, including DT, ANN, Naïve Bayes, SVM and LR have been applied and evaluated using the cross-validation method on a public dataset. Subsequently, the researchers studied the performance improvement achieved by applying ensemble learning based on boosting techniques. The results illustrate the superiority of the boosting technique.

To address the challenges related to predicting customer churn in telecom companies, Idris and Khan [15] have proposed an intelligent forecasting system called FW-ECP, in which a combination of wrapper and filter feature selection methods along with ensemble learning approaches using different base-learners has been investigated. In the first step of the proposed method, the particle swarm optimization (PSO) algorithm was utilized to balance the dataset, and then the minimum redundancy and maximum relevance (mRMR) technique was used for feature selection. Subsequently, in the next step, the GA has been used to remove irrelevant and redundant features. Finally, two ensemble learning based predictors are suggested using majority voting and clustering techniques, and the performance of the FW-ECP system is examined and compared on two public datasets. Since the proposed method has employed meta-heuristic algorithms to handle both the imbalanced nature and the large dimensions of the training sets, the obtained results show a superior prediction performance in comparison with other studied methods.

To fill the gap in comprehensive research that simultaneously evaluates a wide range of ML methods alongside various feature-reducing algorithms for solving the CCP problem, the research conducted by Imani [16] evaluated the performance of seven classifiers (RF, DT, FF-ANN, LR, KNN, SVM and LSTM) and seven target detectors (matched subspace detector, adaptive subspace detector, orthogonal subspace projection, spectral angle mapper, kernel spectral angle mapper, constrained energy minimization and sparsity-based target detector). Additionally, to reduce the number of features, four feature extraction algorithms (linear discriminant analysis, principal component analysis, median-mean feature line embedding and clustering-based feature extraction) and six feature selection algorithms (feature selection with adaptive structure learning, advanced binary ant colony optimization, Relief-F, least absolute shrinkage and selection operator, GA and sequential backward selection) have been investigated. The performance of these methods has been evaluated on three telecom datasets with six evaluation criteria (precision, accuracy, sensitivity, specificity, F1-score and AUC). The obtained results show that the RF and the FF- ANN along with the feature selection method based on the GA perform superior to other competitors.

Since few studies have been done to relate CCP approaches and customer segmentation, in the

research presented by Wu et al. [17], the aim is to provide an integrated framework for customer churn prediction and segmentation. The framework consisted of six stages: CCP data preprocessing, exploratory analysis, ML model development for churn prediction, churn factor analysis, customer segmentation based on churn attributes and analysis of customer behaviors within each segment. To analyze the framework, three public telecom datasets and six ML models (ANN, LR, DT, RF, Naïve Bayes and Adaboost) were evaluated based on accuracy and F1-score criteria. Also, to address the problems caused by the imbalance in the customer churn datasets, the synthetic minority oversampling technique (SMOTE) has been used in which new samples of minority class are constructed based on the Euclidean distance of each sample to all samples in the minority class and finding k-Nearest Neighbor samples. The results show that in the first dataset, the Adaboost model was superior and in the second dataset, the RF model performed best. Also, in the third dataset, although the RF model has the best accuracy, the ANN demonstrates the best performance in terms of the F1-score. After developing the churn prediction model, Bayesian LR was used to analyze churn factors and identify important features for customer segmentation. Finally, the k-means clustering algorithm was employed to divide customers into different groups.

Motivated by the strong performance of ensemble models, Beeharry and Fokone [18] introduced a two-layer soft voting model in their study to tackle the CCP problem in two public telecom datasets. After balancing the datasets through a random under-sampling technique, the authors analyzed the performance of the proposed model on both imbalanced and balanced datasets. The proposed ensemble model is composed of two parallel layers in which the first layer includes conventional ML algorithms comprising the RF, NB, KNN and LR models. In the second layer, the ensemble classifiers are utilized including extra trees, gradient boosting, XGboost and cat boost. Eventually, the final decision is derived by the weighted average technique, so-called soft-voting, applied to the results obtained from the two layers. The numerical experiments demonstrate that the proposed ensemble prediction model outperformed all of the base-learners and achieved a significant increase in the F1-score, particularly when dealing with balanced datasets.

3. Whale optimization algorithm

In this section, we present the basics of the WOA, a nature-inspired swarm intelligence optimization technique inspired by humpback whales' foraging strategies [37]. WOA is renowned for its effectiveness in solving complex optimization problems [38]. This algorithm incorporates three mechanisms to identify the optimal solution in a search space. These mechanisms are called encircling prey and bubble-net attack, which enable linear and non-linear exploitation, and a search for prey mechanism, which offers exploration capability.

The algorithm begins by initializing a population of candidate solutions, referred to as "whales". It then determines the best whale (W^*) by calculating the fitness values of the whales. Subsequently, the WOA enters the position updating phase, during which the position of each whale is updated based on the encircling prey, bubble-net attack and search for prey mechanisms. To select an update mechanism for a whale, first, a random number (p) in the range of [0,1] is selected. If $p<0.5$, then parameter A is calculated according to Eq (1).

$$A = 2a.r_1 - a, \tag{1}$$

where a linearly decreases from 2 to 0 with each iteration and r_1 is a random number in [0,1]. Now, if $|A|<1$, the encircling prey mechanism is employed, and if $|A|\geq 1$, the search for prey mechanism is used. Also, if $p\geq 0.5$, the bubble-net attacking mechanism is applied to update the whale's position. The fitness value of each whale at their updated positions is then calculated and the best whale (W^*) is updated by comparing the obtained fitness values. The algorithm continues iterating by returning to the position updating phase, gradually improving the quality of solutions until a termination criterion is met. Once the termination criterion is met, the algorithm returns W^* as the optimal solution.

The encircling prey mechanism is adapted from the encircling strategy of humpback whales when hunting their prey. However, since in WOA, the location of prey (the optimal solution) is not known in advance, the best solution found so far (W^*) is considered as the target prey in each iteration. Hence, other whales try to update their trajectory towards W^* in each iteration. The encircling mechanism can be expressed by the following equations.

$$W(it+1) = W^*(it) - A.D, \tag{2}$$

$$D = \left|C.W^*(it) - W(it)\right|, \tag{3}$$

$$C = 2r_2, \tag{4}$$

where r_2 is a random number in [0,1], it is the current iteration, $W(it)$ is the position of the whale (solution) in the current iteration, $W^*(it)$ is the best global solution found till the current iteration and $W(it+1)$ is the updated position of the whale.

Bubble-net attacking is another strategy used by humpback whales, which involves creating spiral-shaped bubbles around the prey and moving toward the surface. Therefore, a spiral position updating mechanism is modeled in WOA, which first measures the distance between a whale (W) and prey (W^*) and then uses a spiral equation to update the position of the whale. The bubble-net attacking mechanism can be formulated as Eqs (5) and (6).

$$W(it+1) = D'.e^{bl}\cos(2\pi l) + W^*(it), \tag{5}$$

$$D' = \left|W^*(it) - W(it)\right|, \tag{6}$$

where D' is the distance of the whale to the prey, b is a constant that determines the shape of the spiral function, and l is a random number in the range of $[-1,1]$.

In addition to the mentioned mechanisms, the WOA also incorporates the search for the prey mechanism, providing it with the exploration capability. Hence, whenever $|A|\geq 1$, the WOA moves the whale toward a randomly selected whale (W_{rand}) to explore the entire search space for the optimal solution. The search for prey mechanism can be modeled by Eqs (7) and (8).

$$W(it+1) = W_{rand}(it) - A.D'', \tag{7}$$

$$D'' = \left|C.W_{\text{rand}}(it) - W(it)\right|. \tag{8}$$

4. Customer churn datasets

In this section, we introduce two datasets used in this study to explore the applicability of the proposed SIBEL method to tackle the CCP problem in the telecom industry. In this regard, we explain

the properties of exploited customer churn datasets, namely the IBM Telco[1] and Duke Cell2Cell[2]. The main characteristics of these datasets are summarized in Table 1.

The IBM Telco dataset, released by the IBM Business Analytics Community, is widely recognized in the telecom industry for evaluating customer churn methods, used in several studies to evaluate the performance of the introduced CCP method [16–18,31,39–45]. The dataset is imbalanced, with only 26.54% (1,869 records) labeled as positive churn instances. It consists of 4 numerical and 17 categorical features. The second dataset used in this paper is Duke Cell2Cell, which is provided by the Centre for Customer Relationship Management at Duke University and has been utilized in numerous customer churn studies such as [13–15,17,18,46–49]. It comprises 71,047 records, but only 51,047 of them have a churn label and are applicable for CCP model performance analysis. The Duke Cell2Cell dataset is also imbalanced, with 14,711 records labeled as "Yes" in the churn field, accounting for 28.82% of the total records. It consists of 36 numerical and 22 categorical attributes.

Table 1. Specification of The Customer Churn Datasets in This Study.

Dataset	IBM Telco	Duke Cell2Cell
Total # of Records	7,043	71,047
# of Utilizable Records	7,043	51,047
# of Attributes	21	58
# of Churns	1,869 (26.54%)	14,711 (28.82%)
# of Non-Churns	5,174 (73.46%)	36,336 (71.18%)
# of Numerical Features	4	36
# of Categorical Features	17	22

5. Methodology

The proposed methodology for addressing the CCP problem consists of the following six key steps, as depicted in Figure 1. These steps are as follows:

Step 1: The initial step (data preprocessing) enhances the quality and usefulness of datasets for training ML models. It involves removing irrelevant attributes, handling missing data, transforming data and scaling features.

Step 2: Once preprocessing is complete, the datasets are divided into a training set for model training and a test set for evaluating model performance.

Step 3: To ensure that the training set is well-balanced, we employ a novel hybrid under-sampling technique that combines K-member clustering and the WOA. Specifically, initially, we apply a clustering approach to the majority class, grouping similar instances into clusters of size K. The WOA is then utilized to select one instance from each cluster, maximizing the average performance of individual ML models used in the CCP process.

Step 4: Following the balancing step, the proposed two-level stacking ensemble model is constructed by incorporating six prominent ML algorithms (RF, MLP, SVM-RBF, KNN, NB and LR) as base-learners in each level.

Step 5: Next, the WOA is used again for selecting features and tuning parameters to optimize the

[1] https://www.kaggle.com/datasets/yeanzc/telco-customer-churn-ibm-dataset

[2] https://www.kaggle.com/datasets/jpacse/datasets-for-churn-telecom

prediction capability of the stacking ensemble model. It identifies the most informative subset of features for each base-learner and fine-tunes hyper-parameters. This optimization uses a fitness function based on precision, recall and AUC criteria, with adjustable weights for each criterion. These weights allow the user to customize the significance of each criterion based on the specific policies and desired objectives of the telecom company.

Step 6: Eventually, the prediction performance of SIBEL is evaluated on the test set which includes data not seen by the model during the training process. It should be emphasized that the test set contains imbalanced data because we want to evaluate the SIBEL performance in a scenario that is as close as possible to the real situation in a telecom company.

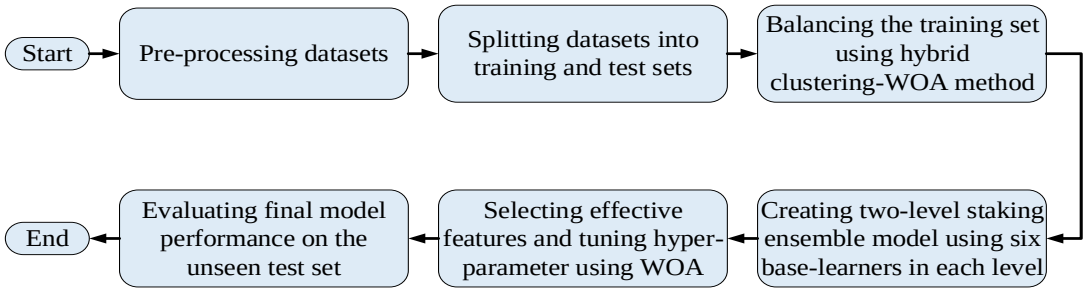


Figure 1. The major phases of the proposed approach.

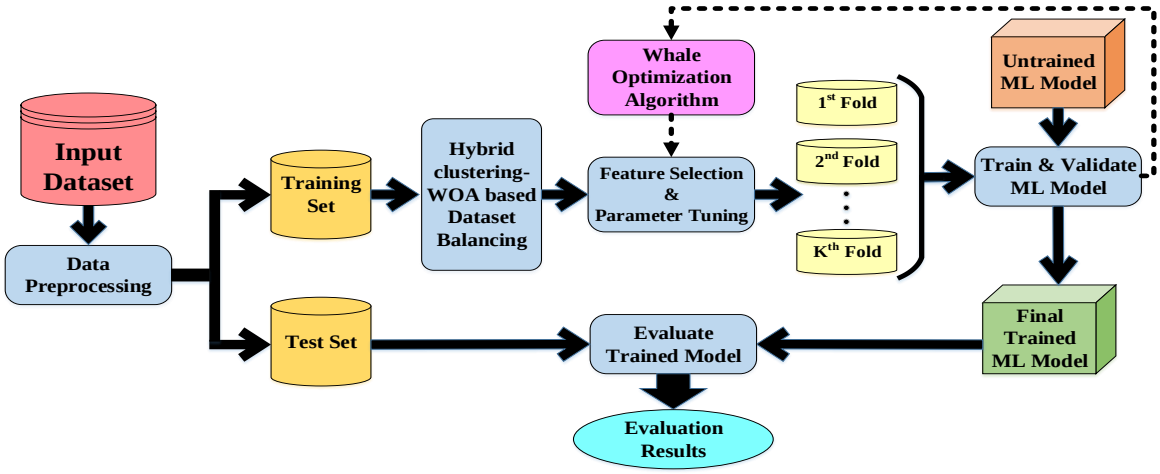


Figure 2. The procedure of the proposed approach in the CCP problem.

Figure 2 illustrates a more detailed view of the proposed CCP approach. The figure showcases a feedback loop between the "training & validation" and the "feature selection & parameter tuning" phases. This loop represents the iterative process employed in the WOA-based feature selection and hyper-parameter tuning technique, wherein the WOA repeatedly assesses the model's performance on the validation data and adjusts the selected subset of features and hyper-parameter values accordingly. This iterative refinement enables the model to converge towards an optimal solution that maximizes its predictive capability for customer churn prediction.

5.1. Data preprocessing

Data preprocessing, as its name implies, is a set of operations that are applied to the raw data to make it ready for the machine learning process. This process is of high significance because it has a direct impact on the performance of ML algorithms and a low-quality preprocessing stage will diminish the quality of predictions. The preprocessing stage consists of various operations, including removing irrelevant attributes, handling missing data, transforming categorical attributes to numeric ones and scaling the features into a consistent range. In the end, each preprocessed dataset is split into separate training and test sets using the hold-out technique. In this study, the ratio of instances in training and test sets is selected empirically which equals 80% to 20%. Table 2, summarizes the operations applied to the churn datasets during the preprocessing phase.

Table 2. Operations applied to the churn datasets during the preprocessing phase.

Operation	Description
Removing Irrelevant Attributes	Eliminating data of irrelevant attributes (e.g., customer ID and Service Area) that don't provide useful information for the ML algorithm's desired output.
Handling Missing Data	In the IBM Telco dataset, 11 instances of the "Total Charges" attribute with missing values (0.16% of total instances) were removed. In the Duke Cell2Cell dataset, there are 14 attributes with missing values. Instances were removed for 13 attributes with missing values less than 5% [50]. The "Handset Price" attribute has 28982 missing values (56.8% of total instances) imputed with mean (56.34).
Data Transformation	Categorical attributes with two values (Yes/No, Male/Female, etc.) were transformed using Label Encoding (0/1) in both datasets. Attributes with more than two categories converted to binary vectors (e.g., 'Multiple Lines', 'Internet Service', etc. in IBM Telco and 'Credit Rating', 'Occupation', etc. in Duke Cell2Cell) using One Hot Encoding.
Feature Scaling	Attributes normalized to a range of [0, 1] to equalize significance and simplify training.

5.2. Hybrid clustering-WOA based dataset balancing

As mentioned above, the low number of churn instances compared to the non-churn instances in both of "IBM Telco" and "Duke Cell2Cell" datasets, makes them imbalance datasets which can negatively affect the training phase of the CPP model and degrade the performance of the final model [51]. Therefore, in this phase, the *K*-member clustering and the WOA algorithms are utilized to intelligently under-sample the majority class (non-churners) in the training set so that the average performance of the desired ML models (MLP, SVM-RBF, RF, NB, KNN and LR) is maximized. The block diagram of the proposed balancing technique (the hybrid clustering-WOA based dataset

balancing block in Figure 2) is demonstrated in Figure 3.

As shown in Figure 3, we first randomly split the original imbalance training set into training′ and validation′ subsets with a ratio of 80% to 20%. Then, we separate the instances belonging to the majority and minority classes in the training′ subset and apply the *K*-member clustering algorithm to the majority-class instances. Unlike *K*-means clustering that the number of clusters is specified beforehand, in *K*-member clustering the number of members in each cluster is determined in advance. A greedy algorithm is introduced in [52] for *K*-member clustering. It starts by selecting a random instance as a cluster head and adding the *K*-1 nearest instances to that cluster to form the first cluster. Subsequently, the next cluster head is chosen as the furthest instance from the last instance added to the previous cluster, and again, the *K*-1 nearest instances join that cluster. This iterative process continues until the number of remaining instances is less than *K*. Finally, each of the remaining instances is added to the nearest cluster. In our proposed balancing method, we consider the *k* as the floor of the imbalance ratio of the training′ subset which is shown in Eq (9).

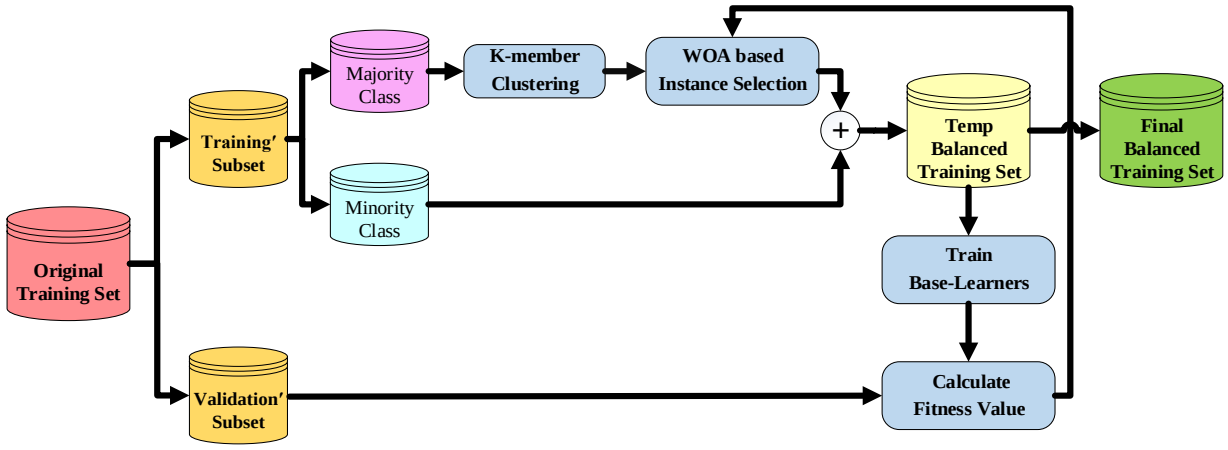


Figure 3. The proposed hybrid clustering-WOA based dataset balancing technique.

$$K = \left\lfloor \frac{N_{Maj}}{N_{Min}} \right\rfloor, \tag{9}$$

where N_{Maj} and N_{Min} are the number of the majority-class and minority-class instances in the training′ subset, respectively. The purpose of *K*-member clustering is to preserve the distribution of majority class instances in the balanced training set similar to the original training set. Afterward, the WOA is utilized to explore and choose an instance from each cluster that maximizes the average performance of the ML algorithms employed in the final prediction model (MLP, SVM-RBF, RF, NB, KNN and LR). Details of the solution representation and fitness function used for WOA based instance selection process are described in the following. As shown in Figure 4, a feasible solution (whale) for WOA based instance selection, *Wi*, can be represented by a vector whose size is equal to the number of *K*-member clusters (N_{cl}). The *j*th element in this vector (I_j) has a value between 1 to the number of instances in the *j*th cluster.

I_1	I_2	…	I_j	…	I_{Ncl}

Figure 4. Representation of a solution (*Wi*) for WOA instance selection.

In order to calculate the fitness value of a solution (whale), at first, the minority-class instances of the training′ subset are combined with the majority-class instances selected by that solution to create a temporary balanced training set. Then, we train each of the ML models (MLP, SVM-RBF, RF, NB, KNN and LR) on the temporary balanced training set. Finally, as shown in Eq (10), the fitness value of each whale is determined by calculating the average of the sum of precision, recall and AUC criteria for each individual ML model on the validation′ subset. In other words, the WOA tries to maximize the average predictive performance of the utilized ML models.

$$\text{Fitness} = \frac{1}{N_{ML}} \sum_{m=1}^{N_{ML}} Precision_m + Recall_m + AUC_m, \tag{10}$$

where N_{ML} is the number of considered ML models (i.e., 6), and Precision and Recall are defined as follows.

$$Precision = \frac{TP}{TP + FP}, \tag{11}$$

$$Recall = \frac{TP}{TP + FN}, \tag{12}$$

where *TP* and *TN* are the number of churn and non_churn instances correctly identified. Also, *FP* is the number of non-churns incorrectly identified as churns, and *FN* is the number of churns incorrectly identified as non-churns.

Moreover, *AUC* equals the area under the Receiver operating characteristic (ROC) curve which is a graphical diagram that can represent the performance of a binary classifier at different threshold levels [53]. The ROC curve plots the true positive rate (*TPR*) on the vertical against the false positive rate (*FPR*) on the horizontal axis, whose definitions are shown in Eq (13) and Eq (14). The higher the value of the area under the ROC diagram (i.e. the *AUC* value) of an ML model, the better the performance of that model in detecting both churn and non-churn classes.

$$TPR = \frac{TP}{TP + FN}, \tag{13}$$

$$FPR = \frac{FP}{FP + TN}. \tag{14}$$

5.3. SIBEL model

In this section, we describe the details of the proposed SIBEL model to tackle the CCP problem in the telecom industry. As mentioned above, considering the appropriate performance of ensemble learning algorithms in addressing the CCP problem, in this research, we propose a two-level ensemble learning model based on the stacking technique, which consists of six effective base-learner models including MLP, SVM-RBF, RF, NB, KNN and LR at each level. The structure of the proposed model is shown in Figure 5. This model evaluates the outputs of these base-learners (*clout*) in two levels, which are in the form of a continuous number in the range of [0, 1], and aggregates the results using the weighted-average voting technique using Eq (15). Finally, according to Eq (16), if the output of the weighted-average voting stage (*vo*) is greater than 0.5, the instance is labeled as churner (*ccpo*=1) and if it is less than 0.5, it is labeled as non-churner (*ccpo*=0).

$$vo = \frac{\sum_{i=1}^{N_{cl}} w_i * clout_i}{\sum_{i=1}^{N_{cl}} w_i}, \tag{15}$$

$$ccpo = \begin{cases} 0 & vo < 0.5 \\ 1 & vo \geq 0.5 \end{cases}. \tag{16}$$

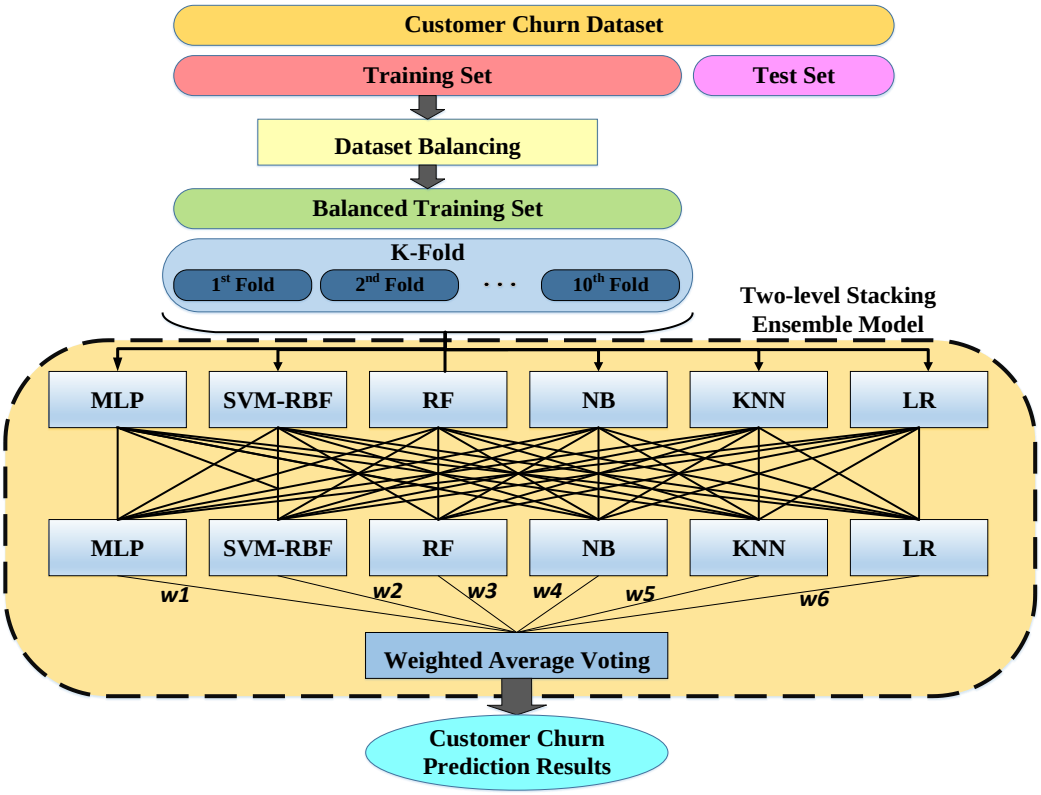


Figure 5. Two-level stacking-mode ensemble model of SIBEL method.

In the following subsections, we first explain the feature selection and parameter tuning procedure in SIBEL using the WOA. Then, we describe the training and evaluation phases of this method carried out on the train and test sets, respectively.

5.3.1. Feature selection and parameter tuning using WOA

In this study, we again take advantage of the WOA as a powerful swarm intelligence optimization algorithm, in order to select the most significant features for each base-learner model and also tune the weight parameters in the final weighted-average voting stage simultaneously. In this regard, to enhance the generalizability of the SIBEL, we first divide the balanced train dataset into five random folds to implement the *k*-fold cross-validation approach. Then we set the initial parameters of the WOA (*Max_Iter*, *Pop_Size* and *b*) and create the initial population of whales (solutions) according to the

solution representation presented in Section 5.3.2. Next, for each of the whales in the population, we decode the solution and extract the feature subset related to each of the base-learners as well as the weights used in the weighted-average voting stage of the proposed model.

Subsequently, we train the proposed two-level stacking-mode ensemble model considering the selected subset of features for each base-learner. It is noteworthy that since we use the *k*-fold cross-validation approach in the training process of the proposed model, the output of this stage includes the prediction results of the trained model for the entire instances available in the balanced training set. In the next step, we calculate the fitness value of each whale based on Eq (18) and save the best global solution found so far (W^*). Now, the termination criterion (reaching *Max_Iter*) of the WOA is checked and if this condition is not satisfied, it proceeds to the next iteration. In this case, after updating the random parameters of the WOA (a, A, C, l and p), we update the location of each whale according to the WOA population updating mechanisms (search for prey, encircling prey and bubble-net attack). The iterative process of decoding solutions, training the proposed model, calculating the fitness values and updating the population continues successively until the termination criterion is satisfied. Eventually, the final best global solution (W^*) found by the WOA is considered as the optimal solution determining the best subset of features for base-leaners and the optimum weights of the final weighted-average voting process. However, because of the random nature of the swarm intelligence algorithms, the selected subset of features may vary in different runs. Therefore, we run the WOA-based feature selection process 10 times and calculate the average selection frequency of each feature. Eventually, the features with a high average selection frequency are selected as the most effective subset of features for training the base-learners.

As described earlier, in the SIBEL method, the WOA is going to simultaneously select the best feature subset for each base-learner and also optimize the six weights of the final weighted-average voting (*w1* to *w6*). Therefore, as shown in Figure 6, a feasible solution representation for each whale can be in the form of a matrix of dimension $(N_{att}+1)\times 6$, where N_{att} equals the total number of attributes in the churn dataset after removing the irrelevant attributes in the preprocessing phase (i.e. N_{att}=20 for IBM Telco and N_{att}=56 for Duke Cell2Cell).

	MLP	*SVM*	*RF*	*NB*	*KNN*	*LR*
1	$att_{(1,1)}$	$att_{(1,2)}$	$att_{(1,3)}$	$att_{(1,4)}$	$att_{(1,5)}$	$att_{(1,6)}$
2	$att_{(2,1)}$	$attr_{(2,2)}$	$att_{(2,3)}$	$att_{(2,4)}$	$att_{(2,5)}$	$att_{(2,6)}$
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.
N_{att}	$att_{(N_{att},1)}$	$att_{(N_{attr},2)}$	$att_{(N_{attr},3)}$	$att_{(N_{attr},4)}$	$att_{(N_{attr},5)}$	$att_{(N_{attr},6)}$
weights	w_1	w_2	w_3	w_4	w_5	w_6

Figure 6. A representation for each whale to select optimized features and weights in the SIBEL.

It should be noted that in this matrix the first N_{att} rows are encoded in the form of binary numbers (0/1) determining whether a feature is selected or not, but the final row elements are continuous numbers in the range of [0, 1]. However, since the WOA was originally proposed for continuous

optimization problems, we apply a simple threshold function, shown in Eq (17), to the elements in the first N_{att} rows when decoding the solutions to obtain the selected feature subset for each of the base-learners.

$$Th(att(i,j)) = \begin{cases} 0 & att(i,j) < 0.5 \\ 1 & att(i,j) \geq 0.5 \end{cases} \quad \forall i \in \{1,2,...,N_{att}\},\ j \in \{1,2,...,6\}. \tag{17}$$

After decoding a whale and obtaining the selected subset of features for base-learners and also the weights of the final voting stage, we form the proposed two-level ensemble model. Next, the base-learners located on the first and second levels of this model are trained using their related subset of features and the outputs obtained from the first-level learners, respectively. In the end, the final prediction results of SIBEL are calculated by applying the weighted-average voting technique on the outputs of the second-level learners based on the weights extracted from the decoded whale. As stated before, in this process we utilize the k-fold cross-validation approach which means we first divide the input balanced training set into k=5 subsets and successively train the model on 4 subsets and evaluate it on the remaining subset. Repeating this process 5 times, we collect the prediction results of the proposed model on the entire input data. Now, in order to evaluate the fitness of each whale, we first measure the precision, recall and AUC criteria on the obtained prediction results and then calculate the fitness value using the function shown in Eq (18).

$$\text{Fitness}=W_{Pre} \times Precision + W_{Rec} \times Recall + W_{AUC} \times AUC, \tag{18}$$

where W_{Pre}, W_{Rec} and W_{AUC} are the weights of each evaluation criterion determined by the telecom company's CRM experts according to their customer retention policies. In other words, these parameters make SIBEL an application-specific approach that can be adjusted based on the desired goals of the company. For instance, if based on the customer retention policies of a telecom company, the primary objective is to identify all customers who are at risk of leaving, as opposed to precisely pinpointing the customers who are most likely to churn, then a higher emphasis should be placed on optimizing the recall criterion and the weight of the recall should be increased. Conversely, if the company's goal is to accurately predict the customers who are highly likely to churn while minimizing the occurrence of false positives, then prioritizing the precision criterion becomes crucial and the weight of the precision criterion must be raised.

5.4. Performance evaluation

In the process of testing SIBEL performance, in addition to evaluating each of the precision, recall and AUC criteria, as defined previously, we also report the values of accuracy and F1-score criteria, as defined in Eqs (19) and (20), respectively. To begin, we evaluate SIBEL by comparing it with both the base-learners and other well-known CCP models. Subsequently, we conduct a sensitivity analysis by changing the weights of the evaluation criteria (W_{Pre}, W_{Rec} and W_{AUC}) in the fitness function. This helps us gain insights into how changes in the weights impact the performance of the SIBEL model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}\ , \tag{19}$$

$$F1-score=\frac{2\times Precision\times Recall}{Precision+Recall}. \tag{20}$$

6. Experimental results

In this section, to evaluate the performance of the proposed churn prediction method (SIBEL) and compare it with other prominent ML models and existing approaches, we provide the results obtained from applying these methods on IBM Telco and Duke Cell2Cell datasets. All the analyzed methods in this study are implemented using the Scikit-Learn library in Python 3.10.8 programming language on a Windows machine with an Intel(R) Core(TM) i7-12700K CPU and 16GB of RAM.

6.1. Settings of SIBEL

Table 3 shows the values of user-controllable settings used in the implementation of the SIBEL. These adjustments are selected empirically based on the results obtained from numerous tests. All the other parameters of the models used in SIBEL are left to their default values.

Table 3. User-controllable settings of SIBEL.

Model	Parameter	Value	Model	Parameter	Value
WOA	Max Iterations (Max_Iter)	100	SVM	Kernel Fuction	*RBF*
	Initial Population Size (Pop_Size)	50		Kernel Coefficient (*gamma*)	10^{-2}
	Spiral Function Constant (b)	1		Regularization Parameter (C)	10^{1}
Fitness	W_{Pre}, W_{Rec} and W_{AUC}	0.33	RF	Split Criterion	*Gini*
k-fold	Number of Folds (k)	5		Number of Trees	10
MLP	Number of Hidden Layers	1	NB	Distribution	*Gaussian*
	Number of Hidden Layer Neurons	10 (IBM) 20 (Duke)	KNN	Number of Neighbors	10
				Distance Metric	*Euclidean*
	Activation function	*'relu'*	LR	Penalty Norm	*'l1'*
	Solver	*'adam'*		Solver	*'liblinear'*

6.2. Results of the SIBEL's training process

After adjusting the user-controllable settings of SIBEL, we first preprocess the datasets and then split them into training and test sets with a ratio of 80% to 20%. Next, we balance the training set using the suggested hybrid Clustering-WOA based dataset balancing approach. Table 4 demonstrates the number of instances in the balanced training set and test set for the IBM Telco and Duke Cell2Cell

datasets following the application of the dataset balancing algorithm.

Table 4. The number of instances in the balanced training set and test set.

Dataset	Original training set		Balanced training set		Test set	
	Non-Churn	Churn	Non-Churn	Churn	Non-Churn	Churn
IBM Telco	4106	1519	1642	1216	1027	380
Duke Cell2Cell	27874	11946	11149	9557	6967	2988

It is evident from this table that the number of non-churn and churn instances in the balanced training set have become much closer to each other compared to the original training set. However, they are not exactly identical because according to Eq (9), we floor the ratio of churn and non-churn instances in the training′ subset in order to calculate the number of non-churn instances in each *K*-member cluster. This will, in turn, result in a difference between the number of clusters (i.e. the number of non-churn instances in the balanced training set) and the number of churn instances in the training′ subset (i.e. the number of churn instances in the balanced training set).

After balancing the datasets, the training process of the SIBEL is started. In this regard, first, the balanced training set is divided into 5 folds. Then, the WOA searches for the optimized subset of input features and also the effective parameters of the final weighted average voting. As mentioned above, due to the random nature of swarm intelligence algorithms, we run the WOA-based feature selection phase 10 times and the features whose average selection frequency is higher than 1 are selected as effective features.

Table 5 summarizes the normalized weights of the final voting stage, optimized by the WOA for the IBM Telco and Duke Cell2Cell datasets. This table demonstrates that the RF and KNN models have the highest and lowest weights among other base learners in the IBM Telco dataset, respectively. However, for the Duke Cell2Cell dataset, MLP has the highest weight and again KNN is of the least weight in the list.

Table 5. Final weighted average voting parameters optimized by the WOA.

Parameter	IBM Telco	Duke Cell2Cell
Weight of MLP (*w1*)	0.19	**0.23**
Weight of SVM (*w2*)	0.16	0.16
Weight of RF (*w3*)	**0.25**	0.18
Weight of NB (*w4*)	0.12	0.15
Weight of KNN (*w5*)	0.10	0.11
Weight of LR (*w6*)	0.18	0.17

6.3. Results of the base-learners and SIBEL on the test set

This section presents the evaluation results of the base learners, one-level ensemble and the proposed two-level ensemble models. These results were achieved by running the mentioned models on the intact and imbalanced test set not seen by any of the models during the training process. The obtained results are reported in Tables 6 and 7 for IBM Telco and Duke Cell2Cell, respectively.

It is evident from Table 6 that the proposed SIBEL model demonstrates improvements of 1.9%,

1.1%, 3.4%, 2.7% and 1.1% over the best individual base-learner on the IBM Telco dataset in terms of accuracy, precision, recall, F1-score, and AUC, respectively. Additionally, it achieves improvements of 1.2%, 1%, 2.7%, 1.7% and 0.8% compared to the one-level ensemble model, across the analyzed criteria. Moreover, Table 7 shows an improvement of 1.7%, 5.2%, 2%, 6.2% and 2.1% over the best individual base-learner and an improvement of 1.2%, 3.2%, 1.1%, 2.6% and 1.6% over one-level ensemble model on the Duke Cell2Cell dataset, respectively, across accuracy, precision, recall, F1-score and AUC criteria. These tables illustrate the enhanced performance of SIBEL over the base-learners and one-level ensemble model in all the analyzed criteria, which proves the effectiveness of the proposed two-level stacking ensemble model in addressing the CCP problem. The suitable performance of SIBEL can be attributed to the aggregation of the base-learners' strengths and the ability of meta-models within the two-level stacking-mode ensemble model to discern intricate relationships in the data.

Table 6. Comparison of the base-learners with SIBEL on IBM Telco dataset.

Detection Method	Accuracy %	Precision %	Recall %	F1-Score %	AUC
MLP	77.6	56.3	75.9	64.6	82.4
SVM-RBF	76.5	54.3	75.3	63.1	82.8
RF	78.4	57.8	74.6	65.1	83.2
NB	75.8	54	71.1	61.4	82.3
KNN	74.4	51.9	69.9	59.6	82.9
LR	76.9	55.2	76.4	64.1	83.1
One-level ensemble	79.1	57.9	77.1	66.1	83.5
SIBEL (Proposed)	**80.3**	**58.9**	**79.8**	**67.8**	**84.3**

Table 7. Comparison of the base-learners with SIBEL on Duke Cell2Cell dataset.

Detection Method	Accuracy %	Precision %	Recall %	F1-Score %	AUC
MLP	56.9	37.1	62.3	46.5	59.7
SVM-RBF	57.1	36.6	58.7	45.1	58.6
RF	63.7	39.5	48.2	43.4	58.1
NB	58.8	36.8	51.9	43.1	57.8
KNN	55.9	33.7	46.9	39.2	56.4
LR	58.2	37.2	56.4	44.8	57.7
One-level ensemble	64.2	41.5	63.2	50.1	60.2
SIBEL (Proposed)	**65.4**	**44.7**	**64.3**	**52.7**	**61.8**

6.4. Sensitivity analysis of the proposed model

As mentioned, the proposed SIBEL method is an application-specific approach in which users can adjust the weights of the precision, recall and AUC criteria in the WOA fitness function according to the specifications of the application at hand. In this section, we conduct a sensitivity test that determines the performance of the SIBEL in different working conditions. To this end, three different scenarios are defined, each representing a situation in which the weight of one performance criterion is higher than that of the other weight criteria. Tables 8 and 9 summarize the given weight to each

evaluation criterion as well as the performance of SIBEL in each of the scenarios.

These tables clearly demonstrate that by changing the weight parameters related to each evaluation criterion in the fitness function of the WOA, its value increases in the results obtained from applying the corresponding trained model on the test set. The first and second scenarios in Tables 8 and 9 show that the precision and recall criteria have the most sensitivity to changes in the weights of the fitness function, which is due to the strong trade-off between them. For example, in the first row of Table 8 (W_{Pre} = 0.8) the precision and recall values are 64.2% and 55.3%, respectively. However, in the second row of this table (W_{Rec} = 0.8) the precision and recall values turn into 53.8% and 86.4%, which show a 10.4% decrease in precision and a 31.1% increase in recall. Also, the first and second scenarios in the Duke Cell2Cell dataset (Table 9) demonstrate 9.5% and 25.7% difference in precision and recall criteria, respectively.

Table 8. Sensitivity analysis of SIBEL on IBM Telco dataset.

Working scenarios	Precision %	Recall %	AUC
$\boldsymbol{W_{Pre}}$**=0.8**, W_{Rec}=0.1, W_{AUC}=0.1	**64.2**	55.3	82.7
W_{Pre}=0.1, $\boldsymbol{W_{Rec}}$**=0.8**, W_{AUC}=0.1	53.8	**86.4**	82.3
W_{Pre}=0.1, W_{Rec}=0.1, $\boldsymbol{W_{AUC}}$**=0.8**	57.4	77.3	**84.8**

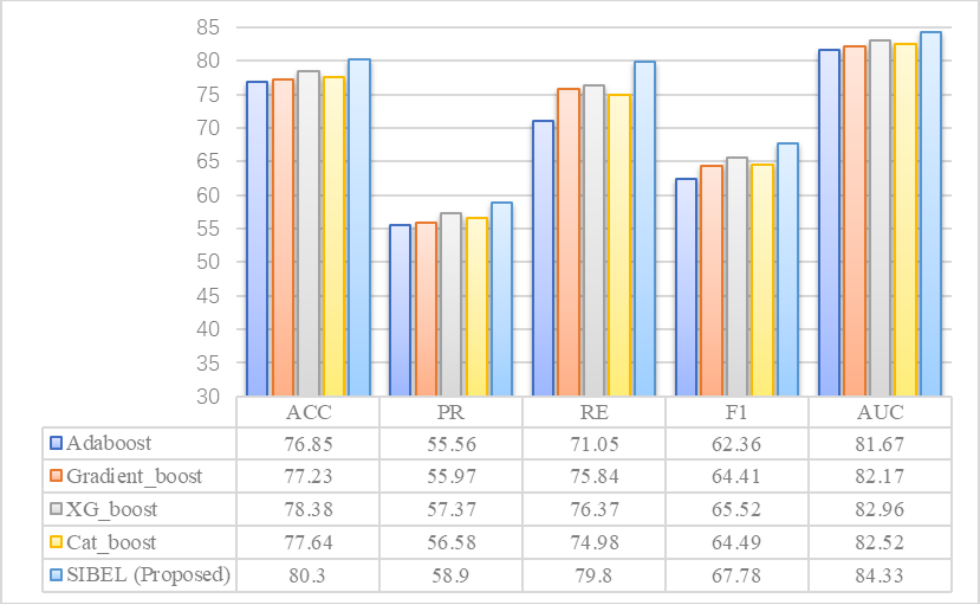
Table 9. Sensitivity analysis of SIBEL on Duke Cell2Cell dataset.

Working scenarios	Precision %	Recall %	AUC
$\boldsymbol{W_{Pre}}$**=0.8**, W_{Rec}=0.1, W_{AUC}=0.1	**51.8**	48.5	58.7
W_{Pre}=0.1, $\boldsymbol{W_{Rec}}$**=0.8**, W_{AUC}=0.1	42.3	**74.2**	59.4
W_{Pre}=0.1, W_{Rec}=0.1, $\boldsymbol{W_{AUC}}$**=0.8**	45.5	64.3	**62.1**

6.5. Comparison of SIBEL with prominent ensemble models

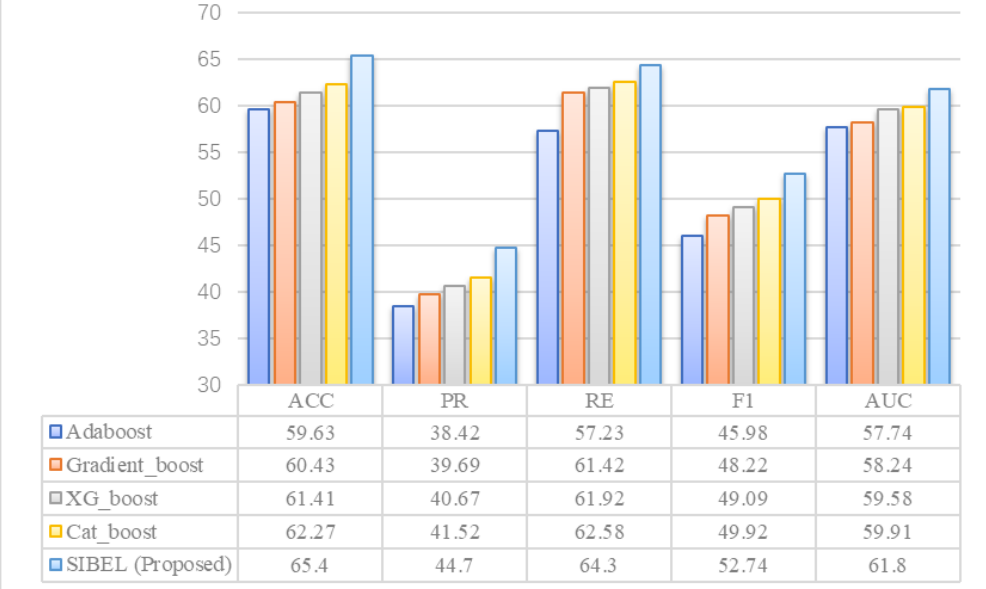
In order to conduct a fair evaluation regarding the performance of the proposed SIBEL model, in this section, we compare its results with other well-known ensemble methods, including Adaboost, Gradient boost, XG boost and Cat boost models. For this purpose, we first train the mentioned ensemble learning models on the balanced training set obtained from the application of the suggested hybrid Clustering-WOA dataset balancing approach to the IBM Telco and Duke Cell2Cell datasets. Subsequently, we calculate the prediction results of the models on the test set and evaluate their performance which is shown in the form of bar charts in Figures 7 and 8, respectively.

The presented figures demonstrate that the proposed SIBEL model exhibits remarkable performance superiority over the other analyzed ensemble models. The main reason for this is the use of a two-level stacking-mode ensemble model, which can effectively learn the complex relationships in the churn datasets. Furthermore, the results obtained in this section indicate that the incorporation of the WOA in the feature selection stage leads to the selection of an effective subset of features for the base-learners, resulting in a substantial improvement in the performance of the proposed customer churn prediction model.



	ACC	PR	RE	F1	AUC
Adaboost	76.85	55.56	71.05	62.36	81.67
Gradient_boost	77.23	55.97	75.84	64.41	82.17
XG_boost	78.38	57.37	76.37	65.52	82.96
Cat_boost	77.64	56.58	74.98	64.49	82.52
SIBEL (Proposed)	80.3	58.9	79.8	67.78	84.33

Figure 7. Comparison of the SIBEL with other ensemble models on the IBM Telco dataset.



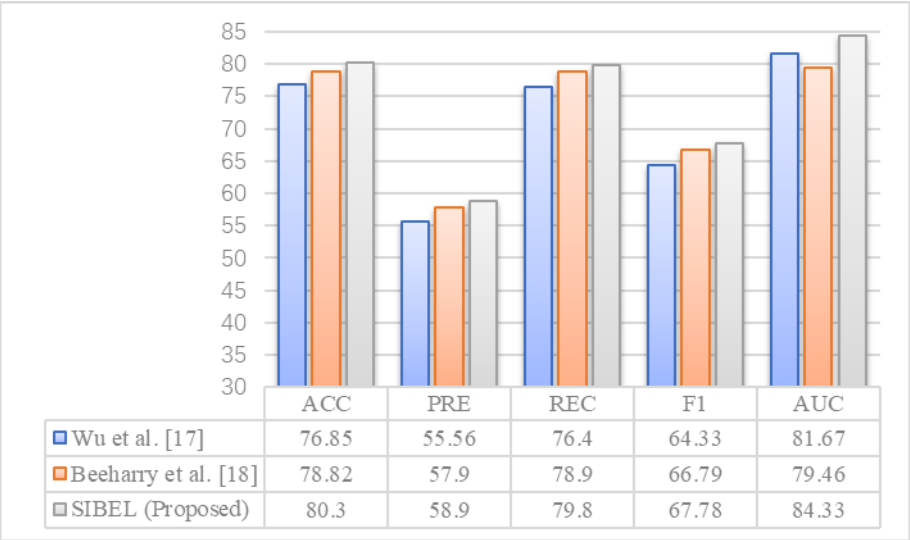
	ACC	PR	RE	F1	AUC
Adaboost	59.63	38.42	57.23	45.98	57.74
Gradient_boost	60.43	39.69	61.42	48.22	58.24
XG_boost	61.41	40.67	61.92	49.09	59.58
Cat_boost	62.27	41.52	62.58	49.92	59.91
SIBEL (Proposed)	65.4	44.7	64.3	52.74	61.8

Figure 8. Comparison of the SIBEL with other ensemble models on the Duke Cell2Cell dataset.

6.6. Comparison of SIBEL with similar existing approaches

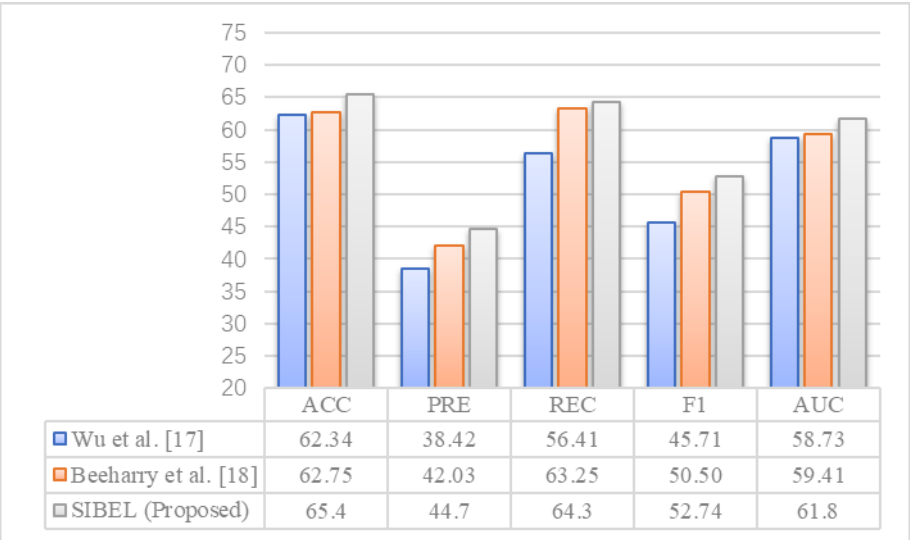
In this section, we compare the performance of SIBEL with the models presented in [17] and [18], which were discussed in the literature review section. We implement the models from these studies and evaluate their performance on the IBM Telco and Duke Cell2Cell datasets using accuracy, precision, recall, F1-score and AUC criteria. Figures 9 and 10 illustrate that SIBEL outperforms the model from [17] in all evaluated criteria for both datasets. It is important to note that [17] focuses on analyzing the performance of various ML models (LR, DT, RF, NB, Adaboost and MLP) on CCP

datasets, which is balanced using the SMOTE technique. We do not introduce an ensemble model for the CCP problem. Therefore, we consider the best performance achieved among the studied ML models in [17] for comparison. Moreover, SIBEL demonstrates superior performance across all evaluated criteria compared to the ensemble model introduced in [18].



	ACC	PRE	REC	F1	AUC
Wu et al. [17]	76.85	55.56	76.4	64.33	81.67
Beeharry et al. [18]	78.82	57.9	78.9	66.79	79.46
SIBEL (Proposed)	80.3	58.9	79.8	67.78	84.33

Figure 9. Comparing SIBEL with existing CCP models on the IBM Telco dataset.



	ACC	PRE	REC	F1	AUC
Wu et al. [17]	62.34	38.42	56.41	45.71	58.73
Beeharry et al. [18]	62.75	42.03	63.25	50.50	59.41
SIBEL (Proposed)	65.4	44.7	64.3	52.74	61.8

Figure 10. Comparing SIBEL with existing CCP models on the Duke Cell2Cell dataset.

The superior performance of the SIBEL can be attributed to the utilization of a two-level stacking ensemble model combined with the use of the WOA algorithm for dataset balancing, feature selection and parameter tuning stages, which play a crucial role in the performance improvement of the SIBEL. More specifically, the proposed hybrid clustering-WOA dataset balancing approach employed by SIBEL effectively preserves more information-rich instances during data balancing, surpassing the performance of the SMOTE and random under-sampling techniques used in [17] and [18]. Additionally, SIBEL leverages the predictive capabilities of multiple base-learners and meta-models within the two-

level stacking ensemble architecture, enabling it to more effectively capture complex and non-linear relationships inherent in CCP datasets. Eventually, unlike the [17] and [18] studies, SIBEL incorporates feature selection and parameter tuning strategies, where the features and weights selected by the WOA significantly contribute to its overall performance improvement.

6.7. Running time of the proposed model

In this section, we evaluate the running time of the proposed SIBEL model for the IBM Telco and Duke Cell2Cell datasets. Table 10 presents the running times for the offline processes (dataset balancing and training phases) and the online process (test phase) for each dataset. As indicated in this table, the balancing phase's running time, comprising the K-member clustering and WOA instance selection stages, is 4.6 hours for Telco and 9.9 hours for Cell2Cell datasets. Furthermore, the total running time for SIBEL's training phase, which includes feature selection and hyper-parameter tuning using the WOA algorithm, equals 20.8 hours for IBM Telco and 39.4 hours for Duke Cell2Cell datasets. Additionally, the test phase time of SIBEL, which measures the time to assess the result for a single input instance (customer), takes 25.1 ms for IBM Telco and 34.2 ms for Duke Cell2Cell datasets.

It is important to note that while the balancing and training phases require an almost large running time, they are conducted offline. The actual running time required during the online use of the model is equivalent to the testing time, which is acceptable for CCP applications. It's also important to mention that parallel processing techniques are employed in all phases to decrease the overall running time of SIBEL by evaluating the outputs of all base-learners at each level simultaneously.

Table 10. Running time analysis on datasets.

Dataset	Offline Processes		Online Process
	Balancing Phase	Training Phase	Test Phase (each instance)
IBM Telco	4.6 hour	20.8 hour	25.1 ms
Duke Cell2Cell	9.9 hour	39.4 hour	34.2 ms

7. Conclusions

In this study, we introduced a two-level stacking-mode ensemble model, called SIBEL, for predicting customer churn in telecom companies, using six well-known ML models (MLP, SVM-RBF, RF, NB, KNN and LR) as base-learners. Developed in Python, this model's performance was assessed using IBM Telco and Duke Cell2Cell datasets. Four sets of experiments were conducted to evaluate the performance of the proposed model, focusing on standard ML evaluation criteria. The initial set of experiments revealed that the proposed model outperforms individual base-learners and one-level ensemble model in churn prediction. Sensitivity analysis of the WOA fitness function parameters, forming the second experimental set, highlighted the SIBEL model's adaptability to telecom companies' specific needs. The third and fourth experimental sets compared our model with other renowned ensemble models and recent CCP approaches, respectively.

These experiments demonstrated the proposed model's improved performance across all evaluation criteria. The superiority of the proposed approach can be attributed to the effective utilization of the WOA in key stages of dataset balancing, feature selection and parameter tuning. Additionally, the results illustrated that the meta-models within the proposed stacking ensemble are

able to capture complex relationships in the data. Consequently, the primary research question—whether the SIBEL model can accurately predict customer churn better than existing methods—has been affirmatively answered through our comprehensive analysis. The accurate churn predictions of SIBEL can primarily benefit businesses by enabling targeted retention strategies and optimizing resource allocation. Such targeted strategies lead to improved customer satisfaction and loyalty, helping companies increase customer lifetime value and achieve sustainable business growth.

However, despite the merits of the proposed SIBEL method, there are limitations to consider. The first limitation is the high running time of the data balancing and training phases due to the use of the WOA algorithm. In fact, despite its innovative design and effective performance in many problems, WOA comes with some limitations such as slow convergence speed and a risk of falling into local minima, which may result in suboptimal solutions. Therefore, future research should analyze the performance of alternative heuristic and metaheuristic algorithms in customer churn prediction. Another limitation is the number of instances in the datasets used, which may not represent the larger customer base of telecom companies. Future studies should explore advanced ensemble structures and base-learners designed for handling larger datasets. Additionally, the application of deep learning and reinforcement learning models to address the customer churn prediction problem is an interesting topic for further investigation in the field.

Use of AI tools declaration

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

The authors agree with the contents of the manuscript, and there are no conflicts of interest among the authors.

References

1. J. Wu, A study on customer acquisition cost and customer retention cost: Review and outlook, *Proceedings of the 9th International Conference on Innovation & Management*, 2012, 799–803.
2. A. Bilal Zorić, Predicting customer churn in banking industry using neural networks, *INDECS*, **14** (2016), 116–124. https://doi.org/10.7906/indecs.14.2.1
3. K. G. M. Karvana, S. Yazid, A. Syalim, P. Mursanto, Customer churn analysis and prediction using data mining models in banking industry, *2019 International Workshop on Big Data and Information Security (IWBIS)*, 2019, 33–38. https://doi.org/10.1109/IWBIS.2019.8935884
4. A. Keramati, H. Ghaneei, S. M. Mirmohammadi, Developing a prediction model for customer churn from electronic banking services using data mining, *Financ. Innov.*, **2** (2016), 10. https://doi.org/10.1186/s40854-016-0029-6
5. J. Kaur, V. Arora, S. Bali, Influence of technological advances and change in marketing strategies using analytics in retail industry, *Int. J. Syst. Assur. Eng. Manag.*, **11** (2020), 953–961. https://doi.org/10.1007/s13198-020-01023-5

6. O. F. Seymen, O. Dogan, A. Hiziroglu, Customer churn prediction using deep learning, *Proceedings of the 12th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2020)*, 2020, 520–529. https://doi.org/10.1007/978-3-030-73689-7_50
7. A. Dingli, V. Marmara, N. S. Fournier, Comparison of deep learning algorithms to predict customer churn within a local retail industry, *Int. J. Mach. Learn. Comput.*, **7** (2017), 128–132. https://doi.org/10.18178/ijmlc.2017.7.5.634
8. M. C. Mozer, R. Wolniewicz, D. B. Grimes, E. Johnson, H. Kaushansky, Predicting subscriber dissatisfaction and improving retention in the wireless telecommunications industry, *IEEE T. Neural Networks*, **11** (2000), 690–696. https://doi.org/10.1109/72.846740
9. J. Hadden, A. Tiwari, R. Roy, D. Ruta, Computer assisted customer churn management: State-of-the-art and future trends, *Comput. Oper. Res.*, **34** (2007), 2902–2917. https://doi.org/10.1016/j.cor.2005.11.007
10. K. Coussement, D. Van den Poel, Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques, *Expert Syst. Appl.*, **34** (2008), 313–327. https://doi.org/10.1016/j.eswa.2006.09.038
11. J. Burez, D. Van den Poel, Handling class imbalance in customer churn prediction, *Expert Syst. Appl.*, **36** (2009), 4626–4636. https://doi.org/10.1016/j.eswa.2008.05.027
12. P. C. Pendharkar, Genetic algorithm based neural network approaches for predicting churn in cellular wireless network services, *Expert Syst. Appl.*, **36** (2009), 6714–6720. https://doi.org/10.1016/j.eswa.2008.08.050
13. A. Idris, A. Khan, Y. S. Lee, Genetic programming and adaboosting based churn prediction for telecom, *IEEE international conference on Systems, Man, and Cybernetics (SMC)*, 2012, 1328–1332. https://doi.org/10.1109/ICSMC.2012.6377917
14. T. Vafeiadis, K. I. Diamantaras, G. Sarigiannidis, K. C. Chatzisavvas, A comparison of machine learning techniques for customer churn prediction, *Simulation Modell. Prac. Theory*, **55** (2015), 1–9. https://doi.org/10.1016/j.simpat.2015.03.003
15. A. Idris, A. Khan, Churn prediction system for telecom using filter–wrapper and ensemble classification, *Comput. J.*, **60** (2017), 410–430. https://doi.org/10.1093/comjnl/bxv123
16. M. Imani, Customer Churn Prediction in Telecommunication Using Machine Learning: A Comparison Study, *AUT J. Model. Simulation*, **52** (2020), 229–250. https://doi.org/10.22060/miscj.2020.18038.5202
17. S. Wu, W.-C. Yau, T.-S. Ong, S.-C. Chong, Integrated churn prediction and customer segmentation framework for telco business, *IEEE Access*, **9** (2021), 62118–62136. https://doi.org/10.1109/ACCESS.2021.3073776
18. Y. Beeharry, R. Tsokizep Fokone, Hybrid approach using machine learning algorithms for customers' churn prediction in the telecommunications industry, *Concurrency Comput.: Prac. Exper.*, **34** (2022), e6627. https://doi.org/10.1002/cpe.6627
19. D. W. Hosmer Jr, S. Lemeshow, R. X. Sturdivant, *Applied logistic regression*, Hoboken: John Wiley & Sons, 2013. https://doi.org/10.1002/9781118548387
20. J. R. Quinlan, Induction of decision trees, *Mach. Learn.*, **1** (1986), 81–106. https://doi.org/10.1007/BF00116251
21. I. Rish, An empirical study of the naive Bayes classifier, *IJCAI 2001 workshop on empirical methods in artificial intelligence*, **3** (2001), 41–46.

22. C. Cortes, V. Vapnik, Support-vector networks, *Mach. Learn.*, **20** (1995), 273–297. https://doi.org/10.1007/BF00994018
23. M. H. Hassoun, *Fundamentals of artificial neural networks*, Cambridge: MIT press, 1995.
24. T. K. Ho, Random decision forests, *Proceedings of 3rd international conference on document analysis and recognition*, **1** (1995), 278–282. https://doi.org/10.1109/ICDAR.1995.598994
25. Y. Freund, R. E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, *J. Comput. Syst. Sci.*, **55** (1997), 119–139. https://doi.org/10.1006/jcss.1997.1504
26. J. H. Friedman, Greedy function approximation: a gradient boosting machine, *Ann. Statist.*, **29** (2001), 1189–1232. https://doi.org/10.1214/aos/1013203451
27. T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, 785–794. https://doi.org/10.1145/2939672.2939785
28. Z. Ghasemi Darehnaei, M. Shokouhifar, H. Yazdanjouei, S. M. J. Rastegar Fatemi, SI-EDTL: Swarm intelligence ensemble deep transfer learning for multiple vehicle detection in UAV images, *Concurrency Comput.: Prac. Exper.*, **34** (2022), e6726. https://doi.org/10.1002/cpe.6726
29. N. Behmanesh-Fard, H. Yazdanjouei, M. Shokouhifar, F. Werner, Mathematical Circuit Root Simplification Using an Ensemble Heuristic–Metaheuristic Algorithm, *Mathematics*, **11** (2023), 1498. https://doi.org/10.3390/math11061498
30. A. Amin, S. Anwar, A. Adnan, M. Nawaz, N. Howard, J. Qadir, et al., Comparing oversampling techniques to handle the class imbalance problem: A customer churn prediction case study, *IEEE Access*, **4** (2016), 7940–7957. https://doi.org/10.1109/ACCESS.2016.2619719
31. I. V. Pustokhina, D. A. Pustokhin, P. T. Nguyen, M. Elhoseny, K. Shankar, Multi-objective rain optimization algorithm with WELM model for customer churn prediction in telecommunication sector, *Complex Intell. Syst.*, **9** (2021), 3473–3485. https://doi.org/10.1007/s40747-021-00353-6
32. N. V. Chawla, Data mining for imbalanced datasets: An overview, In: *Data mining and knowledge discovery handbook*, Boston: Springer, 2009, 875–886. https://doi.org/10.1007/978-0-387-09823-4_45
33. D.-C. Li, C.-S. Wu, T.-I. Tsai, Y.-S. Lina, Using mega-trend-diffusion and artificial samples in small data set learning for early flexible manufacturing system scheduling knowledge, *Comput. Oper. Res.*, **34** (2007), 966–982. https://doi.org/10.1016/j.cor.2005.05.019
34. H. He, Y. Bai, E. A. Garcia, S. Li, ADASYN: Adaptive synthetic sampling approach for imbalanced learning, *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 2008, 1322–1328. https://doi.org/10.1109/IJCNN.2008.4633969
35. H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, *IEEE T. Pattern Anal. Mach. Intell.*, **27** (2005), 1226–1238. https://doi.org/10.1109/TPAMI.2005.159
36. S. Barua, M. M. Islam, X. Yao, K. Murase, MWMOTE--majority weighted minority oversampling technique for imbalanced data set learning, *IEEE T. Knowl. Data Eng.*, **26** (2012), 405–425. https://doi.org/10.1109/TKDE.2012.232
37. S. Mirjalili, A. Lewis, The whale optimization algorithm, *Adv. Eng. Software*, **95** (2016), 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008

38. A. Shokouhifar, M. Shokouhifar, M. Sabbaghian, H. Soltanian-Zadeh, Swarm intelligence empowered three-stage ensemble deep learning for arm volume measurement in patients with lymphedema, *Biomed. Signal Proc. Control*, **85** (2023), 105027. https://doi.org/10.1016/j.bspc.2023.105027
39. J. Pamina, B. Beschi Raja, S. Sathya Bama, M. Sruthi, S. Kiruthika, V. J. Aiswaryadevi, et al., An effective classifier for predicting churn in telecommunication, *J. Adv Res. Dyn. Control Syst.s*, **11** (2019), 221–229.
40. N. I. Mohammad, S. A. Ismail, M. N. Kama, O. M. Yusop, A. Azmi, Customer churn prediction in telecommunication industry using machine learning classifiers, *Proceedings of the 3rd international conference on vision, image and signal processing*, 2019, 34. https://doi.org/10.1145/3387168.3387219
41. S. Agrawal, A. Das, A. Gaikwad, S. Dhage, Customer churn prediction modelling based on behavioral patterns analysis using deep learning, *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*, 2018, 1–6. https://doi.org/10.1109/ICSCEE.2018.8538420
42. A. Amin, F. Al-Obeidat, B. Shah, A. Adnan, J. Loo, S. Anwar, Customer churn prediction in telecommunication industry using data certainty, *J. Bus. Res.*, **94** (2019), 290–301. https://doi.org/10.1016/j.jbusres.2018.03.003
43. S. Momin, T. Bohra, P. Raut, Prediction of customer churn using machine learning, *EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing*, 2020, 203–212. https://doi.org/10.1007/978-3-030-19562-5_20
44. E. Hanif, *Applications of data mining techniques for churn prediction and cross-selling in the telecommunications industry*, Master thesis, Dublin Business School, 2019.
45. S. Wael Fujo, S. Subramanian, M. Ahmad Khder, Customer Churn Prediction in Telecommunication Industry Using Deep Learning, *Inf. Sci. Lett.*, **11** (2022), 24–30. http://doi.org/10.18576/isl/110120
46. V. Umayaparvathi, K. Iyakutti, Automated feature selection and churn prediction using deep learning models, *Int. Res. J. Eng. Tech.*, **4** (2017), 1846–1854.
47. U. Ahmed, A. Khan, S. H. Khan, A. Basit, I. U. Haq, Y. S. Lee, Transfer learning and meta classification based deep churn prediction system for telecom industry, 2019. https://doi.org/10.48550/arXiv.1901.06091
48. A. De Caigny, K. Coussement, K. W. De Bock, A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees, *Eur. J. Oper. Res.*, **269** (2018), 760–772. https://doi.org/10.1016/j.ejor.2018.02.009
49. A. Idris, A. Khan, Y. S. Lee, Intelligent churn prediction in telecom: Employing mRMR feature selection and RotBoost based ensemble classification, *Appl. Intell*, **39** (2013), 659–672. https://doi.org/10.1007/s10489-013-0440-x
50. W. Verbeke, K. Dejaeger, D. Martens, J. Hur, B. Baesens, New insights into churn prediction in the telecommunication sector: A profit driven data mining approach, *Eur. J. Oper. Res.*, **218** (2012), 211–229. https://doi.org/10.1016/j.ejor.2011.09.031
51. Y. Xie, X. Li, E. Ngai, W. Ying, Customer churn prediction using improved balanced random forests, *Expert Syst. Appl.*, **36** (2009), 5445–5449. https://doi.org/10.1016/j.eswa.2008.06.121

52. J.-W. Byun, A. Kamra, E. Bertino, N. Li, Efficient k-anonymization using clustering techniques, *International Conference on Database Systems for Advanced Applications, DASFAA 2007*, 2007, 188–200. https://doi.org/10.1007/978-3-540-71703-4_18
53. A. P. Bradley, The use of the area under the ROC curve in the evaluation of machine learning algorithms, *Pattern Recog.*, **30** (1997), 1145–1159. https://doi.org/10.1016/S0031-3203(96)00142-2

AIMS Press