Ouf *et al. Journal of Big Data* (2024) 11:70
https://doi.org/10.1186/s40537-024-00922-9

Journal of Big Data

RESEARCH Open Access

A proposed hybrid framework to improve the accuracy of customer churn prediction in telecom industry

Shimaa Ouf[1*], Kholoud T. Mahmoud[1] and Manal A. Abdel-Fattah[2]

*Correspondence: shimaaouf@commerce.helwan.edu.eg

[1] Department of Information Systems, Faculty of Commerce and Business Administration, Helwan University, Cairo, Egypt
[2] Department of Information Systems, Faculty of Computers and Artificial Intelligence, Helwan University, Cairo, Egypt

Abstract

In the telecom sector, predicting customer churn has increased in importance in recent years. Developing a robust and accurate churn prediction model takes time, but it is crucial. Early churn prediction avoids revenue loss and improves customer retention. Telecom companies must identify these customers before they leave to solve this issue. Researchers have used a variety of applied machine-learning approaches to reveal the hidden relationships between different features. A key aspect of churn prediction is the accuracy level that affects the learning model's performance. This study aims to clarify several aspects of customer churn prediction accuracy and investigate state-of-the-art techniques' performance. However, no previous research has investigated performance using a hybrid framework combining the advantages of selecting suitable data preprocessing, ensemble learning, and resampling techniques. The study introduces a proposed hybrid framework that improves the accuracy of customer churn prediction in the telecom industry. The framework is built by integrating the XGBOOST classifier with the hybrid resampling method SMOTE-ENN, which concerns applying effective techniques for data preprocessing. The proposed framework is used for two experiments with three datasets in the telecom industry. This study determines which features are most crucial and influence customer churn, introduces the impact of data balancing, compares the classifiers' pre- and post-data balancing performances, and examines a speed-accuracy trade-off in hybrid classifiers. Many metrics, including accuracy, precision, recall, F1-score, and ROC curve, are used to analyze the results. All evaluation criteria are used to identify the most effective experiment. The results of the accuracy of the hybrid framework that respects balanced data outperformed applying the classifier only to imbalanced data. In addition, the results of the proposed hybrid framework are compared to previous studies on the same datasets, and the result of this comparison is offered. Compared with the review of the latest works, our proposed hybrid framework with the three datasets outperformed these works.

Keywords: Telecom industry, Churn prediction, Machine learning, XGBOOST classifier, Hybrid framework, Data preprocessing

Springer Open

© The Author(s) 2024. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

Introduction

Customers are considered one of the most important assets for businesses in numerous dynamic and competitive companies within the marketplace [1].

Competitive market companies, in which customers have numerous choices of service providers, can easily switch a service or even the provider. Such customers are referred to as churned customers [1].

Acquiring a new customer not only costs 5–6 times more than retaining a customer but also requires time to develop customer loyalty to the service provider, subject to the satisfaction of demanded services. While retaining a customer does not involve any additional marketing or other expenses, attention to the resolution of customer concerns is sufficient in most cases.

Furthermore, long-term customers relationship generates more profit because other competitors do not easily attract them, and they can refer to new customers and eventually become less expensive to attend. Thus, a fractional improvement in customer retention can impact the growth and sustainability of telecom businesses [2].

The issue of customer churn or attrition is not unique to the telecom industry. At some point in time, every company in every industry faces the problem of losing customers to competitors. This is usually a source of large financial losses for companies, as it is considered easier and much cheaper to keep an existing customer than to attract new ones. Several studies and surveys support this finding. Van den Poel and Larivière [3] confirmed this in their study of the importance of the economic value of customer retention in the context of a European financial services company [4].

There are different types of artificial intelligence. Machine learning is an application of artificial intelligence, which is used to predict churners and non-churners; Kaličanin et al. in [5] demonstrated that decision makers need to implement some AI applications. Machine learning (ML) represents the main pillar of artificial intelligence and consists of diverse mathematical models like statistics, probabilistic, and neural networks. These models are applied to large datasets to identify data patterns, learn, or predict output values.

Luckert et al. in [6] defined ML algorithms as supervised, unsupervised, semi-supervised, and reinforcement learning. Supervised learning: In this, some data is already tagged with the correct answer, and the machine is trained using this labelled data. It generates a function that predicts the output based on the input observations. Unsupervised learning: This uses information that is neither classified nor labelled and allows the algorithm to act on that information without guidance. Here, the machine is forced to train from an unlabeled dataset and then differentiate it based on certain characters. In [7], semi-supervised learning was defined as training on a combination of labelled and unlabeled data. This learning method combines a small amount of labelled data with a large amount of unlabeled data during training. Reinforcement learning: learning happens in the environment, and this learning employs rewarding desired behaviors and punishing undesirable ones.

Machine learning (ML) has shown outstanding results in a variety of applications, including speech recognition [8], computer vision [9], medical diagnostics [10], and telecom [11]. Among the many approaches developed in the literature for predicting customer churn, supervised machine learning (ML) techniques are the most widely

investigated. Supervised ML involves the development of models that can be learned from labelled data. ML includes a wide range of algorithms, such as decision trees, k-nearest neighbors, linear regression, naive bayes, neural networks, support vector machines (SVM), and genetic programming [12].

Ensemble learning is one of the most popular methods that can be used to improve classification accuracy. These methods are a class of highly successful machine-learning algorithms that combine several models to obtain an ensemble that is supposedly more accurate than its individual members [13, 14]. Widely used ensemble classification methods include bagging [15], AdaBoost [16], random forest [17], random subspace [18], and gradient boosting [19, 20]. Because ensemble classification methods have advantages in terms of accuracy, stability, and generalization, they are widely adopted to solve various problems such as multiple-label learning and imbalance learning.

As far as preprocessing is concerned, in our study, the data preprocessing can often have a significant impact on generalization performance of a supervised ML algorithm. The elimination of noisy instances is one of the most difficult problems in inductive ML [21, 22].

In [23], data preprocessing methods that include data cleaning, integration, transformation, and reduction were proposed. In [24], the authors survey the most popular preprocessing steps required for environmental data analysis and present a proposal to systematize them. As in [24], the authors represented Fig. 1, which shows that it yields significant advantages when optimizing the competitive algorithms and tools used to enhance data quality. The aim of this study is to expand on the results of previous studies in which, in the preprocessing layer as well as the classification layer, which still don't satisfy telecommunication providers as a result, it spotted light from both layers. To predict churn and improve accuracy, a hybrid framework based on the chosen ensemble learning classifier is used to predict churn. Preprocessing is essential for obtaining good-quality results and useful new knowledge from them.

The purpose of this study is to minimize the misclassification of churners and non-churners. A hybrid framework based on the integration of ensemble machine learning and hybrid resampling techniques with the selection of the best techniques for improving data quality is developed to predict customer churn. Previous research on data preparation and preprocessing within a machine learning cycle is argued. Also, it focuses on the limitations of accuracy and performance trade-offs to appropriately target churners, facilitating decision-makers to offer suitable retention plans. Ensemble learning is one of the most popular methods that can be used to reduce the variance in models and improve classification accuracy. The success of XGBoost is primarily due to its scalability in all applications and its speedy learning due to distribution and parallel computation, which allow for quicker model exploration. The SMOTE-ENN method is one of the

Fig. 1 Enhancing decision quality

well-known methods that combines SMOTE as an over-sampling technique and ENN as an under-sampling technique to improve the results. The oversampling led to overfitting the Churner class, resulting in incredibly low precision. Undersampling increased the recall but lowered the precision compared to the baseline score. They suggested opening a research gate for combining techniques. In this study, combining SMOTE and ENN can help determine the benefits and drawbacks of both techniques. Standardization, feature selection, feature reduction, missing value imputation, SMOT-ENN, and XGBoost ensemble learning classifiers are applied in this study. These techniques are applied to improve the accuracy of customer churn prediction.

This study compares pre- and post-data balancing performances for the classifiers, identifies the most important variables that affect customer turnover, and focuses on the speed-accuracy trade-off in hybrid classifiers. It reports an analysis of three datasets with two different experiments using ROC and AUC curves. It has been shown that the result of experiment (a) with the third dataset achieved the highest accuracy of 99.92%, the highest f-score, and the best AUC of 99%, in which hybrid resampling using the SMOTE-ENN technique and STD scaling were performed. While the lowest accuracy level was in experiment (b) in which the resampling is ignored.

The rest of this study is structured as follows: Section "Related Work" presents the previous studies to predict Churn. Section "A proposed hybrid framework for customer churn prediction" presents the main stages of a developing system to predict Churn based on the Proposed Framework. Section "Experiment Results and Discussion" presents experimental results and discussion. Finally, the conclusions are presented in Section "Conclusion".

Related work

In this section, the data preprocessing, key concepts, and criteria of its applications will be discussed. It's worth mentioning the impact of data preprocessing methods, as well as the classification layer, on model performance and the ability to derive best practices within churn management. In addition to that, the study of data mining and machine learning applications of classification within the related domains of the telecom sector will be discussed.

In [25], a study presented an approach to the classification problem with non-discrimination constraints. This classification is based on preprocessing the training dataset. The proposed solution is massaging, reweighing, and sampling. However, they first theoretically studied the trade-off between discrimination and accuracy in a general setting. Similarly, they pointed out the importance of preprocessing and its impact on accuracy.

A recent study in [26] developed a methodological guideline for studies focused on using machine learning. It included a detailed checklist of the eight items and demonstrated detailed data preparation as a list of tasks. These guidelines would improve the quality of research methods.

The study in [7] investigated the influence of data preprocessing and the effects of over- and under-processing. It analyzed a comparison of the popular data preprocessing techniques and their effects on different data classification algorithms. It has been observed that when the dataset was preprocessed using standardization, the best accuracy of

98.24% was obtained. The paper has shown strong over- and under-processing effects when a is overly preprocessed, and the performance of the algorithm decreases.

Regarding the key concepts of preprocessing, as in the prior study [27], the authors suggested a research framework for customer churn prediction. They performed experiments on independent feature selection methods and independent prediction algorithms. They experimentally compared them to construct a two-phase optimization algorithm, although they selected the best match between the feature selection method and the prediction algorithms. This study examined the significance of feature selection in increasing accuracy levels.

In [28], the aim was to predict customer churn using big data. Statistical analyses and J48 decision trees were constructed for the three different datasets. In the preprocessing stage, to provide discriminating features to the classifier, Fisher's ratio and F-score feature extraction methods were used. This study showed that data preprocessing and normalization are indispensable for better comparability of usage trends between months. They focus on the quality of data, which leads to the quality of decisions.

In a recent study by Zhou and Ooka [29], the paper addressed the impact of preprocessing in terms of time concerns. They found that general preprocessing methods, such as standardization, normalization, and other methods such as a number, proportion, and nondimensionalization for input and output, can result in similar performance on a constructed NN model. This study strongly recommends preprocessing for the output, particularly for situations with variables having different orders of magnitude in the output.

The return to study in [14] has been identified as a major contributing factor for prediction accuracy. They used the AdaBoost algorithm, which is a boosting ensemble method. In this study, the hybrid approach consisted of a support vector machine (SVM) classifier with feature selection. The experimental results showed that the proposed hybrid approach achieved better performance than the base classifier SVM alone. An accuracy of 99.24% was obtained using the hybrid approach. However, the study might have been comprehensive if the authors had considered data resampling techniques to improve accuracy within their model without overfitting.

Similarly, Lalwani in [30] argued that data preprocessing is an important phase. They highlighted the role of feature selection to improve classification accuracy and applied a gravitational search algorithm to perform feature selection. This study proved that the XGBOOST and CatBoost classifiers have significant results in terms of accuracy rates.

Considering the findings reported by Rocha et al. [31], they proposed a model based on a rough set theory using four different rule generation algorithms. This study investigated the performance of two oversampling techniques, SMOTE and MTDF. In addition, they used mRMR feature extraction to improve the classification performance and reduce the computational cost and complexity by eliminating unnecessary features from the dataset. The experiments showed that the genetic algorithm yields the best performance using MTDF for oversampling to manage the imbalanced class problem. Overall, these studies highlight the need to combine resampling techniques to achieve improved results.

For instance, returning to the study in [24], the authors proposed a general preprocessing methodology. A systematic study of preprocessing tasks was reported by the authors and believed to be linked to this review for systematic dealing with preprocessing and

data preparation phases, which might be an interesting concern in applying research. As they pointed out, while trying to balance data by oversampling the minority classes, under sampling the majority class, or a combination of both, and using ensemble methods, policies are also used in some cases to gain robustness.

In [32], resampling techniques were utilized to solve class imbalance issues. It has been reported that oversampling led to overfitting the Churner class, resulting in incredibly low precision. Under sampling increased the recall but lowered the precision compared to the baseline score. They suggested opening a research gate for combining techniques.

Earlier research [33] demonstrated that when they investigated the capabilities of preprocessing techniques in an anomaly detection environment. The authors focused primarily on the issue of missing data insertion and data normalization, and the conclusion of the results is quite surprising. The preprocessing techniques for the replacement of missing values are insignificant. Moreover, scaling the dataset can have a considerable impact on the detection capabilities of the anomaly detection system. The paper is limited to their case study or might be the dataset, as these preprocessing techniques enhanced the model performance in other cases.

Another study [34] investigated the impact of combining the resampling techniques of oversampling and under sampling to solve an imbalanced dataset and improve prediction accuracy. It presented a hybrid approach to enhance prediction accuracy. The proposed study was conducted in two phases: (1). Novel hybrid approach: SMOTE oversampling and borderline under sampling (SOS-BUS) (2). Classifier ensemble formation. Experiments performed with different resampling methods were applied to the churn dataset and showed that the hybrid model outperformed the other reference techniques in terms of the area under the ROC curve AUC.

Another study [35] introduced a hybrid model that improved the results of predictions based on an SVM. They used a preprocessing method that combines oversampling and under sampling to achieve more accurate results.

In [36], the authors proposed the accuracy of churn prediction using a deep learning model with machine-learning algorithms. The random forest model has given better accuracy, which is approximately 85%. The study demonstrated the impact of hyperparameter optimization on performance.

The authors in this study apply a three-stage technique for customer churn prediction. In the first stage, the feature selection technique has been used to choose the most relevant features by removing the redundancy and maximizing the significance. In the second stage, the ripple-down rule has been used to build a knowledge-based system to gather knowledge about the customer's behavior. In the final stage, the knowledge acquisition is evaluated by the simulated expert technique. The proposed approach is applied to the churn dataset. It outperformed well for churn prediction in the telecommunications industry, with an accuracy of 95.169% [37].

This study addressed the problem of cross-company churn prediction (CCCP), where the company that called a target doesn't have enough data and should depend on data from another company that called a source to successfully predict customer churn. Until now, there has been no effective method to transfer data in CCCP. It introduced data transformation methods for CCCP using z-score, log, rank, and box-cox. It measured the effect of these methods on CCCP using different classifiers like k-nearest neighbor,

naive bayes, single rule induction, gradient boosted tree, and deep learner neural networks for predicting customer churn in telecommunication companies. The experiments were applied to customer churn datasets in telecommunication companies. The results confirmed that the proposed data transformation methods based on log, rank, and box-cox improved the performance of CCCP and achieved better results using naive bayes [38–40].

The authors confirmed the importance of applying the proposed approach to evaluate the classifier's reliance on the used data and estimate its accuracy. The certainty estimation of the classifier is done by calculating the correlation of all the dataset's features using the normal distribution's six-sigma rules. The approach is applied to extract the classifier's certainty level of decision. It classified the consumers into different groups according to the lower and upper zones. Then the evaluation of the certainty of the classifier is measured before classifying the consumer churn and non-churn [41].

The authors confirmed a lack of effective, rule-based to predict customer churn in telecom companies. They proposed a technique-based intelligent rule to make accurate decisions to classify churn from non-churn customers and identify the customers who are likely to churn or may churn soon [42].

The authors highlighted the important impact of efficient manipulation of a high-dimensional dataset. They said that scientific papers applied feature reduction to reduce processing time and enhance accuracy. However, the authors confirmed that feature reduction causes the loss of important information. They confirmed that assigning weights to the features and avoiding information loss by domain experts is effective, but it is expensive and requires human expertise in the domain. They proposed a novel automatic technique to assign weights without needing domain experts. The results have shown that the proposed technique achieved an accuracy of 89.1% and outperformed feature reduction [43].

As mentioned above and in Table 1, while numerous factors could impact the accuracy level of those churners during the prediction model, most fields of study primarily concentrate on classification algorithms that increase churn prediction accuracy levels. Previous studies often utilized algorithms like decision trees, ANN, logistic regression, Naïve Bayes, and K-nearest neighbor. Limited studies are focused on data preparation, which accounts for more than half of the learning models in the total data discovery

Table 1 Review of classification models used with preprocessing techniques

Article	ML model	Data preprocessing				
	XGBoost	Standard ization	Feature selection	Feature reduction	Data sampling	Missing values imputation
[44]	✓	✓	✓	✓	✖	✖
[45]	✓	✓	✖	✖	✖	✓
[46]	✖	✓	✓	✖	✖	✓
[47]	✖	✓	✓	✖	✓	✓
[48]	✓	✓	✖	✖	✖	✖
[49]	✖	✓	✓	✓	✖	✖
[50]	✓	✖	✖	✖	✓	✖
Present work	✓	✓	✓	✓	✓	✓

process and requires more time and effort. Many studies have emphasized methodical data preparation. In practice, applying the preprocessing processes sequentially is unnecessary, which leads to their implementation as a pipeline of tasks. This is based on the nature of the dataset used. The analysis and integration of appropriate preprocessing techniques and capabilities received insufficient attention from the research community. Advanced ensemble learning models like XGBoost provide excellent predictions on classification issues. Only a limited number of studies have used XGBoost to predict customer churn. The datasets deployed to estimate customer churn are frequently unbalanced, containing many non-churn instances and very few churn instances. To balance the data, previous work has mostly used the Synthetic Minority Oversampling Technique (SMOTE). Hybrid resampling techniques like SMOTE-ENN have been suggested as unique and efficient techniques. Few studies have used this hybrid approach to predict customer churn. The achievement of applying a hybrid framework that combines the benefits of selecting the proper techniques for improving data quality, resampling methods, and ensemble learning hasn't been studied before. As a result, this study intends to provide a novel contribution to research in customer churn by constructing a prediction model combining suitable techniques for data preprocessing with hybrid resampling methods and ensemble learning algorithms. It compares the model's prediction performance with earlier studies. The publicly accessible telecom dataset is subjected to the XGBOOST classifier. To produce more accurate results, efficient techniques for improving data quality and a hybrid preprocessing technique that combines oversampling and undersampling techniques have been employed. The model used a stratified k-fold to create a balance between training and test data and improve classification accuracy. Also, it covers a variety of input data by verifying the model's performance on several folds.

In the next section, an overview of the proposed framework has been presented for accurate prediction. In addition, the results of the proposed hybrid framework are compared with those of previous studies using the same datasets, and the results of this comparison are presented.

A proposed hybrid framework for customer churn prediction

The hybrid framework for customer churn prediction addresses the key concepts for the accuracy of customer churn prediction. The proposed framework introduces a solution over three layers: the data, algorithm, and evaluation layers, as shown in Fig. 2.

Phase one: data layer

It represents the data preprocessing phase. It performs two main tasks: data preparation and feature engineering. For the first data preparation phase, IBM telco customer churn, orange telecom, and Iranian churn experimental datasets were chosen for related experiments, which were acquired online because they are not publicly available due to customers' privacy. Before any data manipulation, an exploratory data analysis was performed to discover the dataset's characteristics. After the analysis, cleaning and transforming some of the data was necessary because of their inferior quality, including

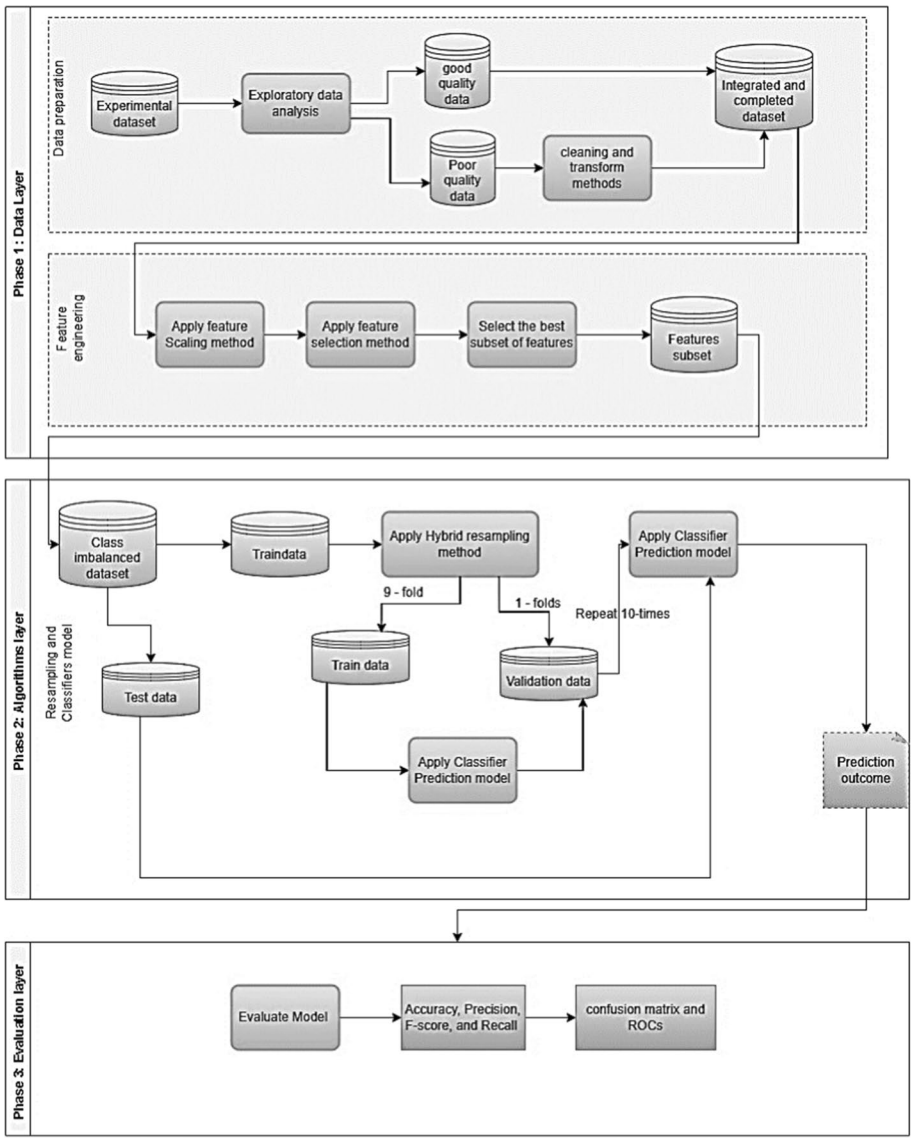

Fig. 2 A proposed hybrid framework for customer churn prediction

different data types and missing and redundant values. Once the datasets have been cleaned and transformed, three datasets become ready for the next task.

The second phase was the feature engineering, a pipeline of preprocessing steps that transform raw data into features used in machine learning algorithms such as predictive models [35]. Predictive models consist of an outcome variable and predictor variables. The most helpful predictor variables are created and selected for the predictive model during the feature engineering process. The datasets needed to be scaled using the STD scaler and then applied to the feature selection step by investigating datasets. The first dataset contains 21 variables, but the variables, including customer ID, do not contain relevant information that can be used for prediction. The second dataset contains 18 variables. The third dataset includes 16 variables. Further information about the datasets can be found in the data availability section.

The feature selection step was performed using univariate feature selection, which works by selecting the best features based on univariate statistical tests (chi-square tests) [14, 44]. This can be considered a preprocessing step for an estimator. Each feature was compared with each feature of the target variable to determine whether there was a statistically significant relationship. This is also called an analysis of variance (ANOVA). When the relationship was analyzed, the relationship between one feature and the target variable was ignored. That is why it is called the univariate. Each feature has its test score. Finally, all test scores were compared, and features with the highest scores were selected.

Comparatively, feature reduction (PCA) was used instead of selection, according to the study in [7, 51], which is a method to reduce the number of variables in a dataset. This is achieved by combining highly correlated variables. PCA allows for the representation of data along one axis, which is called the principal component. This simplifies the higher dimensions into lower ones.

Phase two: algorithms layer

It represents the resampling and classification phase. Because exploratory data analysis was performed, it can be observed that the datasets were unbalanced because it is a special case of the classification problem, where the distribution of a class is not usually homogeneous with other classes. The dataset is unbalanced if one of its categories is 10% or less compared to the other [52]. However, the churn class is usually noticeably smaller than the non-churn class. To solve this problem, a hybrid resampling method was applied after splitting the datasets into training and testing sets. To avoid the overfitting problem, the cross-validation method was adopted, with a stratified k-fold when $k = 10$.

The most popular method of cross-validation is K-fold cross-validation; in this method, the data is divided equally into k groups or folds that are equal to or identical. To test the model on the other k-1 folds at each iteration, we retain a single fold for testing those divided folds in each k iteration. The model's accuracy is then determined by measuring the accuracy achieved in each iteration. To avoid overfitting problems, the stratified cross-validation method was adopted, with k-fold when $k = 10$, as shown in Fig. 3. The dataset is randomly split into ten disjoint subsets, each containing (approximately) 10% of the data.

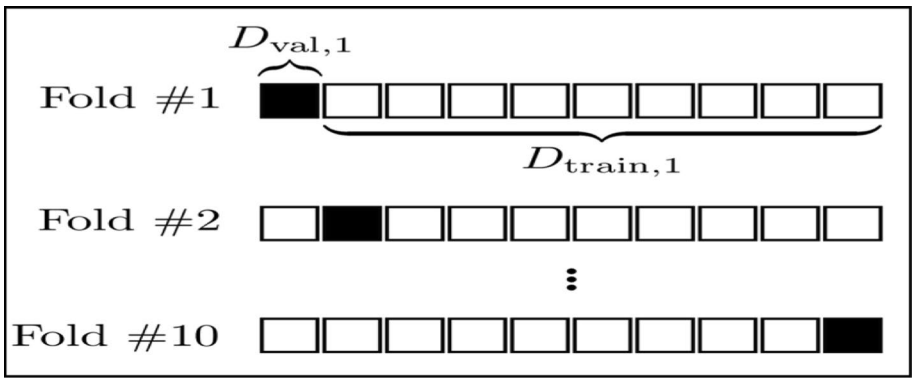

Fig.3 tenfold- cross validation

The model is trained on the training set and then applied to the validation set. Then tenfold cross-validation is performed using the sampled data. This means that in each fold, 90% of the data is used for training and 10% is left for testing the model. In each fold, the training and test data are different, and the stratified cross-validation is the same, meaning that the ratio of churners is the same in every fold.

Hybrid Resampling was adopted according to the procedure used by Rustam, Utami et al. in [35]. They presented a recent method for resampling using SMOTE-ENN, which applies rules to data cleansing by deleting several data samples from both classes.

In [44, 53], the XGBOOST classifier was adopted in the classification phase based on a study by Sarker IH. In [44, 54], the authors identified several advantages of the XGBOOST. Owing to its good cache optimization, the XGBOOST classifier yields satisfactory results in the prediction model, but it requires more training time for the iteration process. The XGBOOST classifier manages a sparse dataset in which missing values are managed properly. The XGBOOST algorithm addresses one of the key problems in tree learning: finding the best split. As shown in algorithm 1. Explain How XGBOOST works in pseudo code.

Algorithm 1 XGBOOST algorithm

Data: dataset and hyperparameters
Initialize $f_0(x)$;
for κ = 1, 2, ..., M do
Calculate $\mathcal{G}\kappa = \frac{\partial \mathcal{L}(y,f)}{\partial f}$
Calculate $\hbar\kappa = \frac{\partial^2 \mathcal{L}(y,f)}{\partial f^2}$
Determine the structure by choosing splits with maximized gain.
A = $\frac{1}{2}\left[\frac{G_L^2}{H_L} + \frac{G_R^2}{H_R} - \frac{G^2}{H}\right]$;
Determine the leaf weights $\omega^* = -\frac{G}{H}$;
Determine the base learner $\hat{b}(x) = \sum_{j=1}^{T} \omega \, \mathrm{I}$;
Add trees $f_\kappa(x) = f_{\kappa-1}(x) + \hat{b}(x)$;
End
Result: $f(x) = \sum_{\kappa=1}^{M} f_\kappa(x)$

Following the benefits of using gradient boosting, after the boosted trees are constructed. It is straightforward to retrieve the importance scores for each attribute. Importance provides a score that indicates the usefulness or value of each feature in the construction of the boosted decision trees within the model. The more an attribute is used to make key decisions with decision trees, the higher its relative importance. This importance was calculated explicitly for each attribute in the dataset, allowing attributes to be ranked and compared [55].

The XGBoost algorithm is an advanced version of gradient boosting, which gives better performance and less computational time. L(Ø) represents the objective function [56, 57]. It consists of two factors, training loss and the regularization term, that can be described as the following.

$$\mathrm{L}(\emptyset) = \sum \mathrm{Bloss}(\mathrm{Y'Yi}) + \sum \mathrm{C}\Omega(\mathrm{fC}) \tag{1}$$

$$\Omega(\mathrm{f}) = \gamma \mathrm{T} + \mathrm{BD}\lambda\|\omega\|\mathrm{D} \tag{2}$$

Table 2 XGBoost standard hyperparameters

Hyperparameters	Description
max_depth (*Optional[int])*	Maximum tree depth for base learners
n_estimators *(int)*	Number of boosting rounds
n_jobs (*Optional[int])*	Number of parallel threads used to run xgboost. When used with other Scikit-Learn algorithms like grid search, you may choose which algorithm to parallelize and balance the threads Creating thread contention will significantly slow down both algorithms
random_state (Optional, *int*)	Random number seed
gamma (*Optional[float])*	(min_split_loss) Minimum loss reduction required to make a further partition on a leaf node of the tree

where.

T: number of leaf node of the tree.

*f*K: independent tree structure q ω: leaf weights.

loss (): the loss function that measures the difference between the actual *y*i and its prediction ŷi.

Ω: used to penalize the complexity of the model for avoiding overfitting.

γ: the leaf weight penalty parameter [57]

λ: the tree size penalty parameter [57]

Before running XGBoost, we must set three types of parameters: general parameters, booster parameters and task parameters.

- General parameters relate to which booster we are using to do boosting, commonly tree or linear model.
- Booster parameters depend on which booster you have chosen.
- Learning task parameters decide on the learning scenario. For example, regression tasks may use different parameters with ranking tasks.

Table 2 introduces the standard hyperparameters for XGboost ensemble classifier.

The hyperparameters of the algorithms were evaluated and validated using K-fold cross-validation. The value of k was 10. Once the predicted outcome is calculated, it is evaluated in the next layer.

Phase three: evaluation layer

When handling imbalanced data, accuracy measures can be decisive. This phase represents the evaluation of the model's performance according to the classification measurements. The comparison between experiments was made using the confusion matrix, precision, recall, F-score, accuracy, and AUC-ROC.

Experiment results and discussion

Experiment setup

Data management and analysis were performed using Anaconda version 3 in the Python 3.8 environment. Anaconda is a distribution of Python and R programming languages

for scientific computing that aims to simplify package management and deployment. The investigations were conducted using the Jupiter Notebook, written in Anaconda 3. The data were analyzed using data exploration and analysis techniques using NumPy, Pandas, Scikit-learn, and imbalance-combine libraries. These are open-source machine-learning libraries in Python.

In this study, the XGboost parameters are settled to the following: Max depth to 5 levels, the gamma (for splitting threshold) to 1, as presented in Table 3.

Dataset description

This study was conducted using online data. The first dataset, the telecom dataset, is not publicly available because of the customer's privacy. It was obtained from the IBM Watson dataset released in 2015. The dataset contains 7043 instances and 21 attributes. The last attributes denote churn or not, of which 5174 are not churners and 1869 are churners. The percentage of churners was 26.53%, and that of non-churners was 73.46%, which indicates unbalanced data. The second dataset, Orange Telecom's churn dataset, was acquired from the Orange Company in America. It contains 3333 instances and 20 attributes. The last attributes denote churn or not, of which 2850 are not churners and 483 are churners. The percentage of churners was 14.49%, and that of non-churners was 85.51%, which indicates unbalanced data. The third dataset, the Iranian churn dataset collected from the Iranian telecom company's dataset, was released. It contains 3150 instances and 16 attributes. The last attributes denote churn or not, of which 2655 are not churners and 495 are churners. The percentage of churners was 15.71%, and that of non-churners was 84.29%, which indicates unbalanced data. These datasets help determine customer predictions and build retention possibilities. Each row in the three datasets represents a customer, and each column contains the customer's attributes described in the column metadata. The results are presented in Table 4.

Performance measures

Several standard performance metrics have been proposed to compare the effectiveness of different classifiers for churn prediction. These metrics are suitable for analyzing the performance model built using both balanced and unbalanced datasets. The metrics are described as follows:

1) Confusion matrix

Table 3 XGBoost hyperparameters applied values

Hyperparameters	Values
max_depth (Optional[int])	5
n_estimators (int)	100
n_jobs (Optional[int])	56
random_state (Optional, int)	0
gamma (Optional[float])	1

Table 4 Datasets metadata and description

	Attributes categories	Description
Dataset 1	Customers	Who left within the last month—the column is called churn
	Services	Each customer has signed up for—phone, multiple lines, internet, online security, online backup, device protection, tech support, and streaming TV and movies
	Customer account information	how long they have been a customer, contract, payment method, paperless billing, monthly charges, and total charges
	Demographic	info about customers—gender, age range, and if they have partners and dependents
Dataset 2	Customers	determining whether a customer canceled the subscription– the column is called churn
	Customer account information	Info about Total day calls—Total day minutes—Total charge—Account length
	Demographic	Info about States—and area code
Dataset 3	Customers	The state of the customers at the end of 12 months
	Customer account information	Info about type of service, the charge amount, number of complaints, call failures, frequency of SMS, subscription length, number of distinct calls, age group, status, and frequency of use
	Demographic	Info about age

It is a table with two rows and two columns that reports the number of false positives. (FP), false negatives (FN), true positives (TP), and true negatives (TN). It provides the information required for analyzing the churn prediction accuracy in terms of false. According to Lalwani [30], there are four terms for understanding the evaluation criteria:

- True positive (TP): The number of customers in the churner category and the predictive model predicted them correctly.
- True negative (TN): The number of customers in the non-churner category and the predictive model predicted them correctly.
- False positive (FP): the number of customers who are non-churners, but the predictive algorithm has labelled or identified them as churners.
- False negative (FN): The number of customers who are churners, but the predictive model has labelled or identified them as non-churners.

2) Performance indicators

Accuracy: This can be described as the ratio of correctly predicted observations to the total number of observations [43]. The formula used is as follows:

$$(TP + TN)/(TP + FP + TN + FN) \tag{3}$$

Precision: This can be described as the ratio of correctly predicted positive observations to total predicted positive observations. The formula used is as follows:

$$TP/(TP + FP) \tag{4}$$

where TP is true positive, and FP is false negative.

Recall: This can be described as the ratio of correctly predicted positive observations to all observations in an actual class. The formula used is as follows:

$$TP/(TP + FN) \tag{5}$$

F1-Score: It can be described as the weighted average of precision and recall. Therefore, this score considers both false positives and negatives.

The formula is as follows:

$$2 \times (P \times R)/(P + R) \tag{6}$$

where P is precision and R is recall.

One of the prominent methods for assessing how well classifiers operate on unbalanced datasets is receiver operating characteristic (ROC) curve analysis. The signal detection theory is where ROC analysis primarily evolved as a technique for choosing a threshold or operating point for the receiver to detect the presence or absence of a signal. The true positive rates and false positive rates are plotted on the ROC curve. Based on the tradeoffs between true positives and false positives, classifiers can be chosen. AUC curve: In contrast to other metrics, AUC is not influenced by any threshold value, as it considers all thresholds for the predicted probabilities [58]. The higher the AUC, the better the model is at predicting the 0 class as 0 and the 1 class as 1. The ROC curve, in our case, is the relationship between the true churn rate on the y-axis and the false churn rate on the x-axis. So, it concerned true and false predictions. It is used with binary classification, and it's highly considered in imbalanced datasets. In addition, the ROC and AUC summarize the performance of a classifier over all thresholds. The ROC curve is mostly used to measure a test's ability as a criterion. The area under the curve was calculated for the two experiments before and after data balancing, and the results are represented in ratio values as represented.

The ROC-based AUC is the highest in experiment (a) which respects the data balancing. It achieved an accuracy of 0.99 with dataset 3. The points of each curve in Fig. 7 are close to the upper left corner, which indicates that the probability of correct prediction for each class is relatively high.

Implementation results

1) Results of exploratory data analysis (EDA)

EDA is used in the data layer to discover dataset characteristics and prepare data for machine learning. The datasets were explored using the library Seaborn to visualize the churn rate, as shown in Fig. 4. In dataset 1, customer churn rate plots show that 26.60% of customers left the company, and 73.40% of customers stayed with the company. In

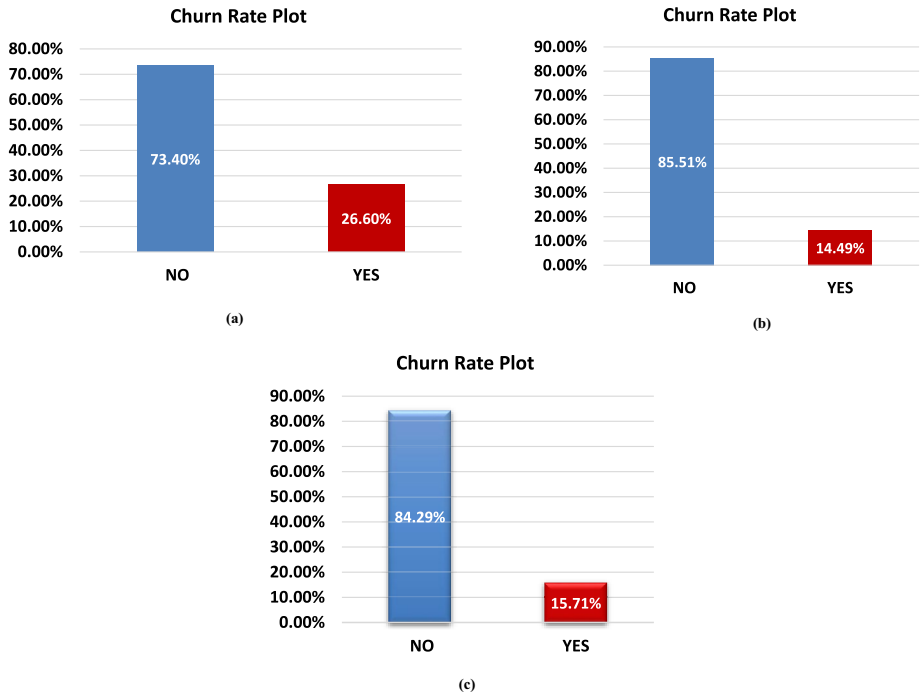

Fig. 4 **a** Churn rate plot of dataset 1. **b** Churn rate plot of dataset 2. **c** Churn rate plot of dataset 3

dataset 2, customer churn rate plots show that 14.49% of customers left the company, and 85.51% of customers stayed with the company. In dataset 3, customer churn rate plots show that 15.71% of customers left the company, and 84.29% of customers stayed with the company.

According to the theory in [48, 52] of Pearson correlation, this is another method for understanding a feature's relation to the target variable and can be used for feature selection. This method was also used to determine the association between features in datasets. The resulting value is $[-1, 1]$, where -1 indicates a perfect negative correlation, $+1$ indicates a perfect positive correlation, and 0 indicates that the two variables do not have a linear correlation. The correlation among the features was assessed and sorted in descending order. As mentioned in Fig. 5, it provides the interpretability of the correlation in terms of score.

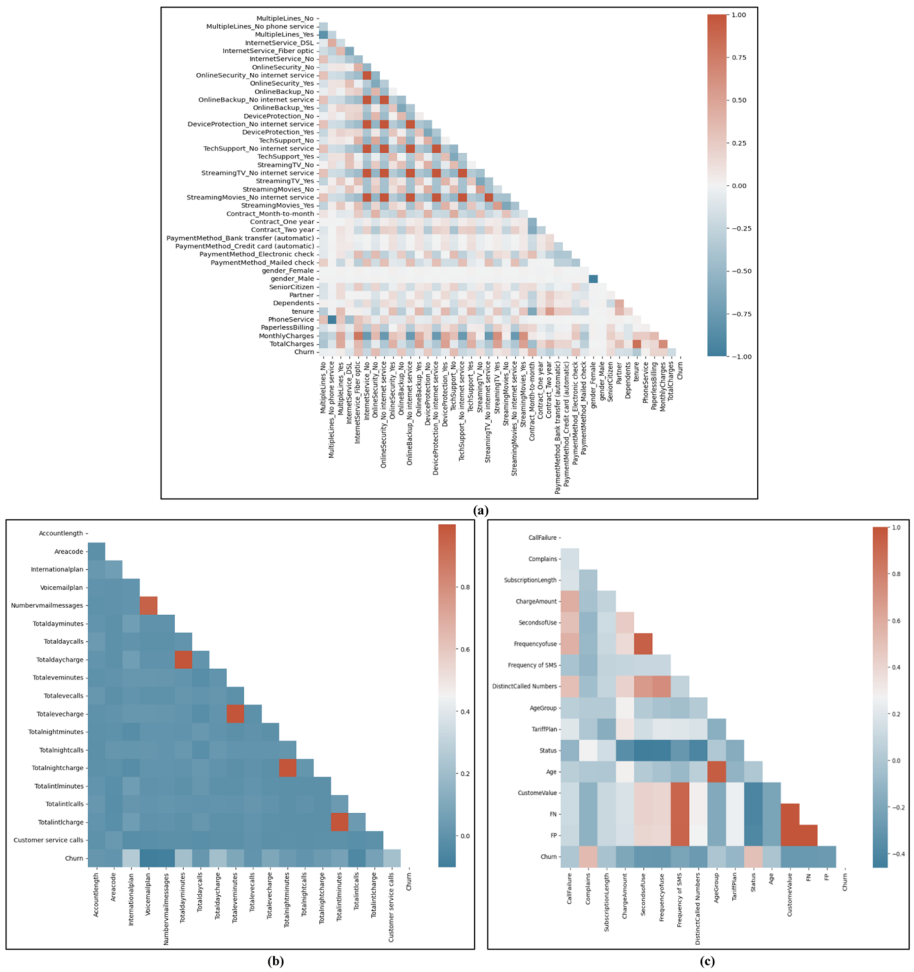

Fig.5 **a** Feature correlation heatmap triangle masked of dataset 1. **b** Feature correlation heatmap triangle masked of dataset 2. **c** Feature correlation heatmap triangle masked of dataset 3

2) Results of data preprocessing

This subheading of the experiment concerned the application of different preprocessing tasks manipulated in a comparative manner. The experiment was grouped by preprocessing categories as follows: data balancing, data scaling, and data reduction.

Feature Standardization Method It was applied as a fundamental preprocessing step. Python's Sci-kit has various scalers: STD Scaler, Min–Max Scaler, Max-Abs Scaler, and Robust Scaler are the four types of scalers. In this paper, we standardize the features by using the Stander Scaler function, and all columns in the three datasets were rescaled.

Univariate Feature Selection Method Univariate assigns scores for each feature, the important and best features based on high scores were selected. Table 5 (a) shows the scores of all features after applying the univariate method to dataset 1. We can see that the TotalCharges registered the first highest score with 476139.219272. Tenure registered the second *highest* score with 12349.737573. OnlineBackup_No registered the lowest score at 222.579066 for all features.

Table 5 (b) shows the scores of all features after applying the univariate method to dataset 2. We can see that the Total_day_minutesregistered the first highest score with

Table 5 Best 10 features selected by univariate feature selection

a

Features	Score in numeric values
TotalCharges	476139.219272
Tenure	12349.737573
MonthlyCharges	2795.912582
Contract_Month-to-month	409.776537
Contract_Two year	386.138647
PaymentMethod_Electronic check	328.970764
OnlineSecurity_No	326.284059
TechSupport_No	313.220029
InternetService_Fiber optic	286.338882
OnlineBackup_No	222.579066

b

Features	Score in numeric values
Total_day_minutes	1672.481918
Number_vmail_messages	534.664151
Total_eve_minutes	354.514460
Total_day_charge	284.314857
International_plan	174.881205
Customer_service_calls	132.535103
Total_eve_charge	30.131524
Voice_mail_plan	20.966810
Total_night_minutes	14.543284
Total_intl_calls	11.667127

c

Features	Score in numeric values
Seconds_of_Use	905672.543557
Custome_Value	121432.369097
FN	109289.132188
Frequency_of_SMS	21236.222217
Frequency_of_use	11188.620137
FP	5217.723645
Distinct_Called_Numbers	2584.474298
Complains	687.022355
Charge_Amount	257.547347
Status	96.397384

1672.481918. The Number_vmail_messages registered the second *highest* score with 534.664151. The Total_intl_calls registered the lowest score at 11.667127 for all features.

Table 5 (c) shows the scores of all features after applying the univariate method to dataset 3. We can see that the Seconds_of_Use registeredthe first highest score with 905672.543557. The Custome_Value registered the second *highest* score with 121432.369097. The Status has registered the lowest score at 96.397384for all features.

Class balancing and Standardization (STD) method: SMOTE-ENN was imported from imblearn-combine to implement class balancing; hence, we benefit from the SMOTE method, which is good at handling this minority class problem. However, because most of the class samples to be deleted are the result of deleting randomly selected data, this method can sometimes delete important data samples from the training dataset. To

avoid this, the hybrid SMOTE-ENN is used. ENN works by removing the sample data whose class label value is different from most of the k values of its closest neighbor.

3) Splitting dataset

Splitting each of the three datasets into a training set and a testing set. The stratified cross validation used to train the models using the training set, and the result of the cross validation is registered. The models that are used to the testing set are evaluated, and the results of the testing set are registered.

4) Model training ensemble classifier

Following that, apply the ensemble-boosting algorithm XGBOOST. The boosting algorithm has a method to obtain feature importance, and when it was used, as in

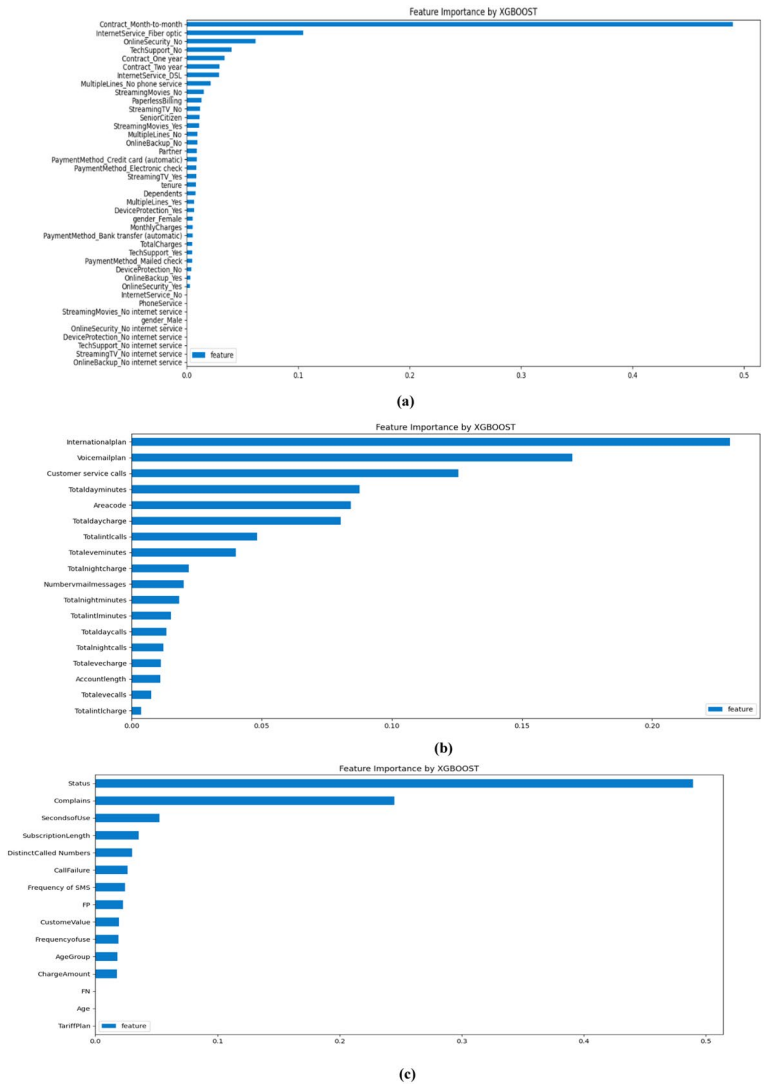

Fig. 6 **a** Importance of XGBOOST of the dataset 1. **b** Importance of XGBOOST of the dataset 2. **c** Importance of XGBOOST of the dataset 3

Table 6 Classification report with different preprocessing tasks

Experiment	Dataset	Technique	Accuracy (A)	Precision (P)	Recall (R)	F1-Score (f)
(a)	(1) Dataset—IBM	SMOTE-ENN + STD Scaler	98%	97%	98%	98%
	(2) Dataset—Orange		98.25%	98%	99%	99%
	(3) Dataset—iran		99.92%	99.87%	100%	99.93%
(b)	(1) Dataset—IBM	STD Scaler	82%	71%	52%	59%
	(2) Dataset—Orange		84%	74%	59%	66%
	(3) Dataset—iran		84%	75%	62%	68%

Figs. 6a, b, and c. It was observed that feature correlation and feature importance by XGBOOST had the same scores for features.

As mentioned in Table 6, the best accuracy was 99.92% obtained in experiment (a) in dataset 3. This result was extracted from the XGBOOST classification method using data that was resampled with SMOTE-ENN and only scaled with the STD Scaler. It can be observed that in experiment (a), datasets 1 and 2 roughly achieved the same accuracy ratios, but there was a slight difference in their f-scores. Meanwhile, the lowest level of accuracy resulted from the XGBOOST classifiers using STD Scaler, which was without the use of a resampling technique and was equal to 82% in experiment (b) dataset 1.

5) Evalutaing classification model

In the cross-validation result, the SMOTE-ENN with STD scalar in experiment (a) in dataset 3 has registered the highest performance (A = 99.92%, P = 99.87%, R = 100%, and F = 99.93%), while the lowest performance (A = 98%, P = 97%, R = 98%, and F = 98%) has been recorded in dataset 1. Experiment (b) STD scalar in dataset 3 has recorded the highest performance (A = 84%, P = 75%, R = 62%, and F = 68%), while the lowest performance (A = 82%, P = 71%, R = 52%, and F = 59%) has been recorded in dataset 1.

Discussion and analysis

This section reports an analysis of two different experiments using three datasets using ROC and AUC curves. Compared with a review of the latest works, it empowered the proposed framework by listing the advantages and disadvantages of applied techniques. The ROC and AUC summarize the performance of a classifier over all thresholds [59]. The ROC curve is mostly used to measure a test's ability as a criterion. As shown in Fig. 7, the area under the curve was calculated for the two experiments, and the results are represented in ratio values as represented in the following Table 7.

Table 7 ROC and AUC for experiments

Experiment	Dataset	Technique	ROC and AUC
(a)	(1) Dataset—IBM	SMOTE-ENN + STD Scaler	84%
	(2) Dataset—Orange		92%
	(3) Dataset—iran		99%
(b)	(1) Dataset—IBM	STD Scaler	83%
	(2) Dataset—Orange		90%
	(3) Dataset—iran		92%

Table 8 Elapsed time results of machine learning algorithms

Experiment	Dataset	Technique	No of records	Elapsed Time
(a)	(1) Dataset—IBM	SMOTE-ENN + STD Scaler	7044	10.28
	(2) Dataset—Orange		3334	8.73
	(3) Dataset—iran		3151	7.09
(b)	(1) Dataset—IBM	STD Scaler	7044	9.10
	(2) Dataset—Orange		3334	7.86
	(3) Dataset—iran		3151	6.27

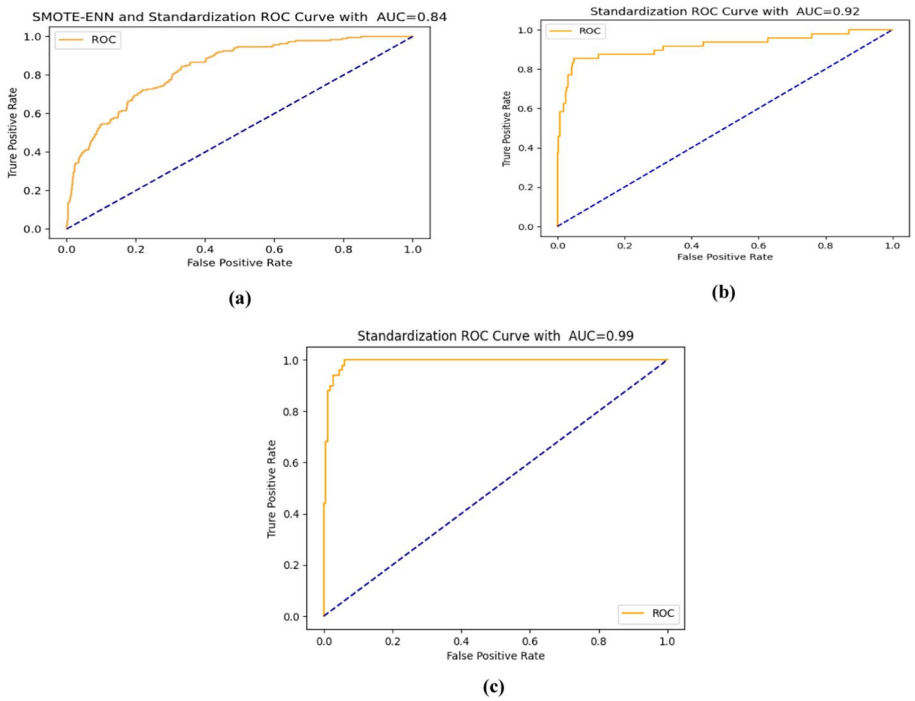

Fig. 7 **a** ROC and AUC for experiments of dataset 1. **b** ROC and AUC for experiments of dataset 2. **c** ROC and AUC for experiments of dataset 3

As shown in Tables 6 and 7, experiment (a) in dataset 3 achieved the highest accuracy of 99.92% and the highest AUC of 99%. In comparison, dataset 2 achieved the second-highest accuracy of 98.25% and the highest AUC of 92%. AUC of applying the hybrid framework respects the balanced data outperformed the second experiment which ignored the resampling. In experiment (b), dataset 1 achieved the lowest accuracy (82%) and AUCs of 0.83. This result agrees with the results presented in [54, 60] that XGBOOST achieved the highest accuracy.

By tracking the execution time that the model uses, the performance of the model is determined. This will help us determine how long the run will take. In this study, the time library is used to measure performance. Table 8 and Fig. 8 contain the computing time for every experiment for every dataset.

Table 9 Review of models, data preparations, and accuracy using the same datasets by customer churn prediction

Author	Dataset	Algorithms	Data preparation	Accuracy %
[45]	IBM Watson	SVM XGBOOST	-MiniMax Scaler -Removing missing values -Remove customer IDs -Convert predictor variable to a binary numeric -convert categorical variables into dummy	82% 83%
[61]	IBM Watson	XGBOOST	-STD Scaler -ignore resampling -Feature engineering	79% 80%
[44]	IBM Watson	KNN RF XGB	-did not manage missing values -convert data types into binary and numeric variables -Univariate Feature selection	75% 77% 79%
[62]	IBM Watson	Ensemble Random Forest	-Label Encoder -randomizedSearchCV -Did not do resampling -focus on Feature Reduction to achieve the same accuracy	79%
[63]	IBM Watson	DecisionTreeClassifier RandomForestClassifier + GridSearchCV KNeighborsClassifier logistic Regression	-remove CustID -deal with categorial data -use STD scaler -use SMOTE	81% 85% 77% 84%
[64]	IBM Watson	DecisionTreeClassifier	-numeric transformer -categorical transformer -Simple Imputer by median -STD Scaler, -OneHotEncoder	79%
[65]	IBM Watson	Hybrid Model (SOM + ANN)	-feature selection, -data filtering, -data cleaning	79.5 3%
[66]	IBM Watson	Deep-BP-ANN	missing values Label Encoding Normalization Resampling the datasets Feature selection Exploratory Data Analysis (EDA)	88.12%
[67]	IBM Watson	XGBoost	Explorative Data Analysis (EDA) p	82.20%
[68]	Orange Telecom Churn	Decision Tree Random Forest k-nearest neighbor Gaussian Naïve Bayes Multinomial Naïve Bayes Bernoulli Naïve Bayes XGBoost Artificial Neural Network	Exploratory Data Analysis (EDA)	93.40% 95.20% 88.30% 88.15% 63.71% 86.20% 95.65% 86.80%
[43]	Orange Telecom Churn	CCP *with* Genetic Algorithm	Feature selection	89.10%
[69]	Orange Telecom Churn	-Handling Missing Values -Imputation -Outlier Detection and Treatment	Stacking model	97.65%

Table 9 (continued)

Author	Dataset	Algorithms	Data preparation	Accuracy %
[70]	Iranian churn	-Artificial neural networks based on error-driven and self-organizing learning approaches	Feature selection	98.30%
The study framework	-IBM Watson -Orange Telecom Churn -Iranian churn	-XGBOOST	Managed missing value Deal with categorial data STD scaler Used SMOTE-ENN	98% 98.25% 99.92%

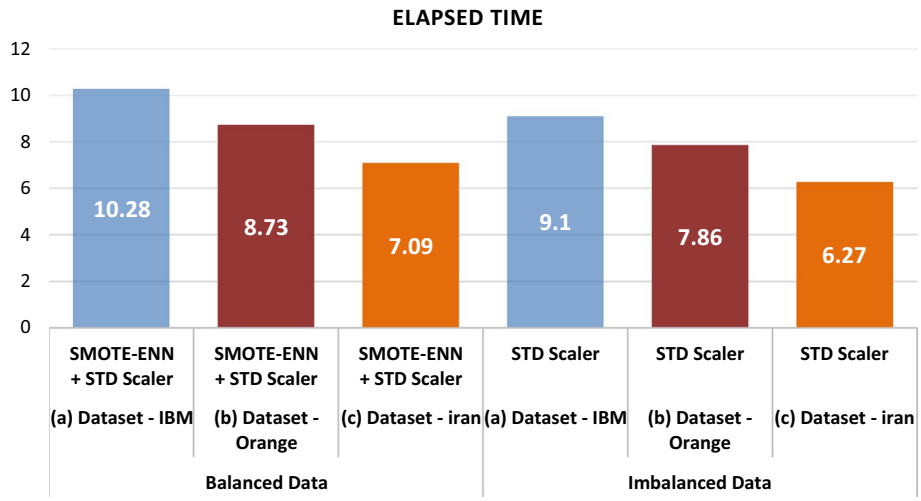

Fig. 8 Computational time of classification methods

The hybrid framework and vital layers of this study, as compared to similar research works mentioned in Table 9, in which its power encompasses the integration of modelling techniques. A review of the latest works mentioned in this table reveals the in-depth reasons for the performance improvement achieved by the proposed method. It must be pointed out that our results in datasets 1, 2, and 3 achieved the highest accuracy. Compared with the review of the latest works, our proposed hybrid framework outperformed these works.

Similarly, to the study in [71], Through our study and from the results obtained, we can notice some advantages to the XGBOOST model and hybrid resampling [72]. They summarized them according to the following:

XGBOOST classifier advantages.

- Higher accuracy
- It is extremely fast because of its parallel

computation capability.

- highly efficient in balanced and imbalanced datasets.
- It is versatile because it can be a regression, classification, or ranking.
- Missing values are imputations, so there is no need for feature engineering.

Hybrid resampling advantages:

- adopt SMOTE to generate synthetic minority class examples and then use data cleaning techniques like the edited nearest neighbor rule (ENN) to detect and delete noisy examples.

The main objective of this study is to improve the classification performance of machine learning algorithms for the prediction of customer churn. A comparative study offers the utilization of two experiments before and after data balancing. A hybrid framework increases the accuracy by up to 99.92%. It improves the performance of the machine learning classifier and solves the class imbalance issue.

Conclusion

This study proposes a hybrid framework for customer churn prediction that addresses the key concepts for churn prediction accuracy. It presents a comparative analysis of two experiments with three datasets. The proposed framework introduces a solution over three layers: data, algorithm, and evaluation. In phase one, the data layer represents the data preprocessing phase. It conducted two main tasks: data preparation and feature engineering. In phase two, the algorithm layer integrates two resampling techniques—SMOTE-ENN, a preprocessing step to reduce the negative effects caused by a class imbalance—and applied the XGBOOST classifier as an ensemble learning algorithm. In phase three, the evaluation layer applies four classification evaluation methods: accuracy score, precision, recall, and F-score. They are used to validate the results, and the results of cross-validation and the testing data are registered. This study offeres results that outperformed the findings of previous works in this field with the same datasets mentioned above in Table 9. Hence, our proposed hybrid framework shows the highest performance and accuracy.

Acknowledgements
This research did not receive any specific grants from funding agencies in the public, commercial, or non-profit sectors.

Author contributions
Manal A. Abdel-Fattah and Shimaa Ouf supervised the paper; Kholoud T. Mahmoud, Manal A. Abdel-Fattah, and Shimaa Ouf designed the proposed framework; Shimaa Ouf and Kholoud T. Mahmoud provided the datasets; Kholoud T. Mahmoud, Shimaa Ouf, and Manal A. Abdel-Fattah presented methodology; Shimaa Ouf and Kholoud T. Mahmoud made implementation; Shimaa Ouf, Manal A. Abdel-Fattah, and Kholoud T. Mahmoud provided a discussion; Kholoud T. Mahmoud writing—original draft preparation; Shimaa Ouf writing—review and editing.

Funding
Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB).

Data availability
IBM Telco Customer Churn Dataset, available at: https://www.kaggle.com/blastchar/telco-customer-churn

Declarations

Competing interests
The authors declare no competing interests.

Received: 3 July 2023 Accepted: 16 April 2024
Published online: 09 May 2024

References

1. Coussement K, Lessmann S, Verstraeten G. A comparative analysis of data preparation algorithms for customer churn prediction: a case study in the telecommunication industry. Decis Support Syst. 2017;95:27–36. https://doi.org/10.1016/j.dss.2016.11.007.
2. Óskarsdóttir M, Bravo C, Verbeke W, Sarraute C, Baesens B, Vanthienen J. Social network analytics for churn prediction in telco: model building, evaluation, and network architecture. Expert Syst Appl. 2017;85:204–20. https://doi.org/10.1016/j.eswa.2017.05.028.
3. Huang Y, Kechadi T. An effective hybrid learning system for telecommunication churn prediction. Expert Syst Appl. 2013;40:5635–47.
4. den Poel V, et al. Customer attrition analysis for financial services using proportional hazard models. Eur J Oper Res. 2004;157(1):196–217.
5. Kaličanin K, Čolović M, Njeguš A, Mitić V. Benefits of Artificial Intelligence and Machine Learning in Marketing. Beograd: Singidunum University; 2019. p. 472–7. https://doi.org/10.15308/sinteza-2019-472-477.
6. Luckert M, Schaffer-Kehnert M. Using Machine Learning Methods for Evaluating the Quality of Technical Documents. MS thesis, Dept Comput Sci Linnaeus Unv. 2015:102.
7. Rao S, Poojary P, Somaiya J, Mahajan P. a Comparative study between various preprocessing techniques for machine learning. Int J Eng Appl Sci Technol. 2020;5(3):431–8. https://doi.org/10.33564/ijeast.2020.v05i03.069.
8. Deng L, Li X. Machine learning paradigms for speech recognition: an overview. IEEE Trans Audio Speech Lang Process. 2013;21(5):1060–89.
9. Huang MQ, Nini´c J, Zhang QB. Bim, machine learning and computer vision techniques in underground construction: current status and future perspectives". Tunn Undergr Space Technol. 2021;108:103677.
10. P. Oza, P. Sharma, and S. Patel. Machine learning applications for computer-aided medical diagnostics. In: *Proceedings of the Second International Conference on Computing, Communications, and Cyber-Security*Springer, New York, NY, USA, 2021.
11. Ullah I, Raza B, Malik AK, Imran M, Islam SU, Kim SW. A churn prediction model using random forest: analysis of machine learning techniques for churn prediction and factor identification in telecom sector. IEEE Access. 2019;7:60134–49. https://doi.org/10.1109/ACCESS.2019.2914999.
12. Adwan O, Faris H, Jaradat K, Harfoushi O, Ghatasheh N. Predicting customer churn in telecom industry using MLP neural networks: modeling and analysis. Life Sci J. 2014;11(3):1097–8135. https://doi.org/10.7537/marslsj110314.11.
13. Sri Bala M, Rajya GV. Efficient ensemble classifiers for prediction of breast cancer. Int J Adv Res Comput Sci Softw Eng. 2016;6(3):5–9.
14. Buslim N, Zulfiandri Z, and KyungOh L. Ensemble learning techniques to improve the accuracy of predictive model performance in the scholarship selection process. J Appl Data Sci.2023;(3): 264–75.
15. Breiman L. Bagging predictors. Mach Learn. 1996;24(2):123–40.
16. Hastie T, Rosset S, Zhu J, Zou H. Multi-class adaboost. Stat Interfac. 2009;2(3):349–60.
17. Breiman L. Random forests. Mach Learn. 2001;45(1):5–32.
18. Ho T K. Random decision forests. In: *Proceedings of the 3rd international conference on document analysis and recognition*. 1995, 278–282.
19. Friedman JH. Stochastic gradient boosting. Comput Stat Data Anal. 2002;38(4):367–78.
20. Dong X, Yu Z, Cao W, Shi Y, Ma Q. A survey on ensemble learning. Front Comput Sci. 2020;14(2):241–58. https://doi.org/10.1007/s11704-019-8208-z.
21. Kotsiantis SB, Kanellopoulos D, Pintelas PE. Data preprocessing for supervised leaning. IJCS, 2006, 1(2), 111–17.
22. C. M. Teng. Correcting noisy data. *In: Proc. 16th international conf. on machine learning. pp* 239–248. San Francisco, 1999.
23. Sivakumar A, Gunasundari R. A survey on data preprocessing techniques for bioinformatics and web usage mining. Int J Pure Appl Math. 2017;117(20):785–94.
24. Gibert K, Sànchez-Marrè M, Izquierdo J. A survey on pre-processing techniques: Relevant issues in the context of environmental data mining. AI Commun. 2016;29(6):627–63. https://doi.org/10.3233/AIC-160710.
25. Kamiran F, Calders T. Data preprocessing techniques for classification without discrimination. Knowl Inf Syst. 2012. https://doi.org/10.1007/s10115-011-0463-8.
26. Haneef R, Tijhuis M, Thiébaut R, Májek O, Pristaš I, Tolenan H, Gallay A. Methodological guidelines to estimate population-based health indicators using linked data and/or machine learning techniques. Arch Publ Health. 2022;80(1):1–12. https://doi.org/10.1186/s13690-021-00770-6.
27. Lang TG, Yiannis. Report Information from ProQuest Содержание. Prod Manag. 2016;15(May):2016–9.
28. Naga N, Prithvi, P. Customer churn prediction using big DataAnalytics. 2016. http://www.diva-portal.org/smash/record.jsf?pid=diva2:1049992.
29. Zhou Q, Ooka R. Influence of data preprocessing on neural network performance for reproducing CFD simulations of non-isothermal indoor airflow distribution. Energy Build. 2021;230:110525. https://doi.org/10.1016/j.enbuild.2020.110525.
30. Lalwani P, Sethi P, Kumar M, Jasroop M, Chadha S. Customer churn prediction system: a machine learning approach. Computing. 2022;104(2):271–94. https://doi.org/10.1007/s00607-021-00908-y.
31. Rocha Á, Correia AM, Costanzo S, Reis LP. New contributions in information systems and technologies. Adv Intell Syst Comput. 2015;353:III–IV. https://doi.org/10.1007/978-3-319-16486-1.
32. Avon V. Machine learning techniques for customer churn prediction in banking environments. 2016. http://tesi.cab.unipd.it/53212/.
33. Vavra J, Hromada M. Evaluation of data preprocessing techniques for anomaly detection systems in industrial control system. Ann DAAAM Proc Int DAAAM Symp. 2019;30(1):738–45.
34. Salunkhe UR, Mali SN. A hybrid approach for class imbalance problem in customer churn prediction: a novel extension to undersampling. Int J Intell Syst Appl. 2018;10(5):71–81. https://doi.org/10.5815/ijisa.2018.05.08.

35. Rustam Z, Utami DA, Hidayat R, Pandelaki J, Nugroho WA. Hybrid preprocessing method for support vector machine for classification of imbalanced cerebral infarction datasets. Int J Adv Sci, Eng Inf Technol. 2019;9(2):685–91. https://doi.org/10.18517/ijaseit.9.2.8615.
36. Bristy BN. Customer Churn Analysis and Prediction. 2022. http://dspace.uiu.ac.bd/handle/52243/2325.
37. Amin A, Rahim F, Ramzan M, Anwar S. A prudent based approach for customer churn prediction. In: Beyond Databases, *Architectures and Structures: 11th International Conference, BDAS 2015, Ustroń, Poland*, Proceedings 11 (pp. 320–332). Springer International Publishing] 2015.
38. Amin A, Shah B, Khattak AM, Moreira FJL, Ali G, Rocha A, Anwar S. Cross-company customer churn prediction in telecommunication: a comparison of data transformation methods. Int J Inf Manage. 2019;46:304–19.
39. Amin, A., Shah, B., Khattak, A. M., Baker, T., & Anwar, S. (2018, July). Just-in-time customer churn prediction: with and without data transformation. In: *2018 IEEE congress on evolutionary computation (CEC)*. pp. 1–6. IEEE]
40. Amin A, Al-Obeidat F, Shah B, Tae MA, Khan C, Durrani HUR, Anwar S. Just-in-time customer churn prediction in the telecommunication sector. J Supercomput. 2020;76:3924–48.
41. Amin A, Al-Obeidat F, Shah B, Adnan A, Loo J, Anwar S. Customer churn prediction in telecommunication industry using data certainty. J Bus Res. 2019;94:290–301.
42. Amin A, Anwar S, Adnan A, Nawaz M, Alawfi K, Hussain A, Huang K. Customer churn prediction in the telecommunication sector using a rough set approach. Neurocomputing. 2017;237:242–54.
43. Amin A, Shah B, Abbas A, Anwar S, Alfandi O, Moreira F (2019). Features weight estimation using a genetic algorithm for customer churn prediction in the telecom sector. In: *New Knowledge in Information Systems and Technologies: Volume 2* (pp. 483–491). Springer International Publishing]
44. Beschi Raja J, Chenthur PS. An optimal ensemble classification for predicting Churn in telecommunication. J Eng Sci Technol Rev. 2020;13(2):44–9. https://doi.org/10.25103/jestr.132.07.
45. bandi, atindra. "Telecom Churn Prediction | Kaggle." Kaggle: Your Machine Learning and Data Science Community, Kaggle, 19, Jan. 2019, https://www.kaggle.com/bandiatindra/telecom-churn-prediction.
46. Kumar S, Kumar M. Predicting customer churn using artificial neural network, vol. 1000. New York City: Springer International Publishing; 2019.
47. Agrawal S, Das A, Gaikwad A, Dhage S. Customer churn prediction modelling based on behavioral patterns analysis using deep learning. 2018, https://doi.org/10.1109/ICSCEE.2018.8538420.
48. Hu X, Yang Y, Chen L, Zhu S. Research on a Customer Churn Combination Prediction Model Based on Decision Tree and Neural Network. In: *2020 IEEE 5th Int. Conf. Cloud Comput. Big Data Anal. ICCCBDA* 2020, 2020, https://doi.org/10.1109/ICCCBDA49378.2020.9095611.
49. Brandusoiu IB, Toderean G. Churn prediction in the telecommunications sector using neural networks. Acta Tech Napocensis. 2016;57(1):27.
50. Kimura T. Customer churn prediction with hybrid resampling and ensemble learning. JMIDS, 2022; 25(1):1–23.
51. Azeem M, Usman M. A fuzzy based churn prediction and retention model for prepaid customers in telecom industry. Int J Comput Intell Syst,2018, 11(1), 66–78]
52. Chawla N. Data mining for imbalanced datasets: an overview. In: Maimon O, Rokach L, editors. Data mining and knowledge discovery handbook. Berlin: Springer; 2005. p. 853–67.
53. Guillaume L, et al. Imbalanced-learn: a python toolbox to tackle the curse of imbalanced datasets in machine learning. J Mach Learn Res. 2017;18:1.
54. Sarker IH. Machine learning: algorithms, real-world applications and research directions. SN Comput Sci. 2021;2(3):1–21. https://doi.org/10.1007/s42979-021-00592-x.
55. Brownlee Jason. Feature Importance and Feature Selection With XGBoost in Python. *Machinelearningmastery*. 2016, https://machinelearningmastery.com/feature-importance-and-feature-selection-with-xgboost-in-python/. Accessed 22 Feb 2022.
56. Tianqi Chen and Carlos Guestrin 2016 ACM. XGBoost: A Scalable Tree Boosting System. In: Proc. of the 22nd ACM SIGKDD *Int. Conf. on Knowledge Discovery and Data Mining*, pp 785– 94.
57. Liu J, Zhang X, Li Y, Wang J, Kim H-J. Deep learning-based reasoning with multi-ontology for IoT applications *IEEE*. Access. 2017;7:124688–701.
58. Emmanuel T, Maupong T, Mpoeleng D, Semong T, Mphago B, Tabona O. A survey on missing data in machine learning, vol. 8. New York City: Springer International Publishing; 2021. https://doi.org/10.1186/s40537-021-00516-9.
59. Sarker IH, Alqahtani H, Alsolami F, Khan A, Abushark YB, Siddiqui MK. Context pre-modeling: an empirical analysis for classifcation based user-centric context-aware predictive modeling. J Big Data. 2020;7(1):1–23.
60. Fauzan MA, Murfi H. The accuracy of XGBoost for insurance claim prediction. Int J Adv Soft Comput its Appl. 2018;10(2):159–71.
61. Bonat, Michelle. "Case Study: Predict Customer Churn Using Machine Learning." Git Hub, 2019. https://colab.research.google.com/github/michellebonat/Predict_Customer_Churn_ML/blob/master/Predict_Customer_Churn_Case_Study.ipynb. Accessed 3 Jan 2022.
62. Burleigh, Tyler. "Predicting Customer Churn - Tyler Burleigh." Tyler Burleigh. 2020, https://tylerburleigh.com/blog/predicting%20customer-churn-telco-customer-churn/. Accessed 12 Sep 2021.
63. Envex. "Analysis and Prediction of Telecom User Churn." Programmer Group—a Programming Skills Sharing Group, 2020, https://programmer.group/analysis-and-prediction-of-telecom-user-churn.html. Accessed 23 May 2023.
64. Khandelwal Ashutosh. "Machine Learning | Customer Churn Analysis Prediction—CodeSpeedy. 2020. https://www.codespeedy.com/machine-learning-customer-churn-analysis-predict. Accessed 17 Apr 2022.
65. Shen TJ, Shibghatullah ASB. Customer churn prediction model for telecommunication industry. J Adv Artif Life Robot. 2022;3(2):85–91.
66. Thorat AS, Sonawane VR. Customer churn prediction in the telecommunication industry using deep learning. J Data Acquis Process. 2023;38(3):1417–25.
67. Hota L, Dash PK. Prediction of customer churn in telecom industry: a machine learning perspective. Comput Intell Mach Learn. 2021;2(2):1–9.

68. Gowd S, Mohite A, Chakravarty D, Nalbalwar S. Customer churn analysis and prediction in telecommunication sector implementing different machine learning techniques. In: *First international conference on advances in computer vision and artificial intelligence technologies (ACVAIT 2022)*. Atlantis Press. 2023. pp. 686–700
69. Wahul RM, Kale AP, Kota PN. An ensemble learning approach to enhance customer churn prediction in telecom industry. Int J Intell Syst Appl Eng. 2023;11(9s):258–66.
70. Jafari-Marandi R, Denton J, Idris A, Smith BK, Keramati A. Optimum profit-driven churn decision making: innovative artificial neural networks in telecom industry. Neural Comput Appl. 2020;32:14929–62.
71. Abdu-Aljabar RD, Awad OA. A comparative analysis study of lung cancer detection and relapse prediction using XGBoost classifier. IOP Conf Ser Mater Sci Eng. 2021;1076(1):012048. https://doi.org/10.1088/1757-899x/1076/1/012048.
72. Guan H, Zhang Y, Xian M, Cheng HD, Tang X. SMOTE-WENN: solving class imbalance and small sample problems by oversampling and distance scaling. Appl Intell. 2021;51(3):1394–409. https://doi.org/10.1007/s10489-020-01852-8.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Shimaa Ouf I am currently an associate professor in the Information Systems Department, Faculty of Commerce and Business Administration, Helwan University. I was born in Cairo, Egypt. I received a Ph.D. in Information Systems from the Faculty of Computers and Information, Helwan University, Egypt, in 2017. Dissertation Topic: A Proposed Paradigm for a Smart E-Learning Ecosystem Based on the Semantic Web. I have published many articles in international journals and conferences on e-learning ecosystems, Web 2.0 technologies, cloud computing, smart learning environments, Personalized e-learning ecosystems using ontology and semantic web rule language, business intelligence, big data, and blockchain. I am an active reviewer for numerous international journals.

Kholoud T. Mahmoud received the B.Sc. degree in Information Systems from the Business Information Systems Department, Faculty of Commerce and Business Administration, Helwan University, Egypt, in 2017. She is currently pursuing a M.Sc. degree in machine learning. Her research interests include machine learning, data mining, deep learning, business intelligence, and big data.

Manal A. Abdel-Fattah received the Ph.D. degree in information systems from the Faculty of Computers and Information, Cairo University. She worked as a business development consultant at the Management National Institute and as a project manager at the Ministry of State and Administrative Development. She is currently a professor in the Faculty of Computers and Artificial Intelligence at Helwan University. She has supervised many master's and Ph.D. theses. Her research interests include big-data analytics, data mining, and evaluation methodologies. She is a reviewer of many information systems journals.