BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 03/15 pp294–314
DOI: 10.26599/BDMA.2023.9020041
Volume 7, Number 2, June 2024

Enhancing Telemarketing Success Using Ensemble-Based Online Machine Learning

Shahriar Kaisar*, Md Mamunur Rashid, Abdullahi Chowdhury, Sakib Shahriar Shafin, Joarder Kamruzzaman, and Abebe Diro

Abstract: Telemarketing is a well-established marketing approach to offering products and services to prospective customers. The effectiveness of such an approach, however, is highly dependent on the selection of the appropriate consumer base, as reaching uninterested customers will induce annoyance and consume costly enterprise resources in vain while missing interested ones. The introduction of business intelligence and machine learning models can positively influence the decision-making process by predicting the potential customer base, and the existing literature in this direction shows promising results. However, the selection of influential features and the construction of effective learning models for improved performance remain a challenge. Furthermore, from the modelling perspective, the class imbalance nature of the training data, where samples with unsuccessful outcomes highly outnumber successful ones, further compounds the problem by creating biased and inaccurate models. Additionally, customer preferences are likely to change over time due to various reasons, and/or a fresh group of customers may be targeted for a new product or service, necessitating model retraining which is not addressed at all in existing works. A major challenge in model retraining is maintaining a balance between stability (retaining older knowledge) and plasticity (being receptive to new information). To address the above issues, this paper proposes an ensemble machine learning model with feature selection and oversampling techniques to identify potential customers more accurately. A novel online learning method is proposed for model retraining when new samples are available over time. This newly introduced method equips the proposed approach to deal with dynamic data, leading to improved readiness of the proposed model for practical adoption, and is a highly useful addition to the literature. Extensive experiments with real-world data show that the proposed approach achieves excellent results in all cases (e.g., 98.6% accuracy in classifying customers) and outperforms recent competing models in the literature by a considerable margin of 3% on a widely used dataset.

Key words: telemarketing; machine learning; imbalanced dataset; oversampling; ensemble model; online learning

• Shahriar Kaisar is with the Department of Information Systems and Business Analytics, RMIT University, Melbourne 3000, Australia. E-mail: shahriar.kaisar@rmit.edu.au.
• Md Mamunur Rashid is with the School of Engineering and Technology, Central Queensland University, Rockhampton 4700, Australia. E-mail: m.rashid@cqu.edu.au.
• Abdullahi Chowdhury is with the Faculty of Engineering, Computer and Mathematical Science, University of Adelaide, Adelaide 5005, Australia. E-mail: abdul.chowdhury@adelaide.edu.au.
• Sakib Shahriar Shafin and Joarder Kamruzzaman are with the Centre for Smart Analytics, Federation University Australia, Ballarat 3350, Australia. E-mail: ss.shafin@federation.edu.au; joarder.kamruzzaman@federation.edu.au.
• Abebe Diro is with the School of Accounting, Information Systems and Supply Chain, RMIT University, Melbourne 3000, Australia. E-mail: abebe.diro3@rmit.edu.au.
* To whom correspondence should be addressed.
Manuscript received: 2023-05-08; revised: 2023-11-02; accepted: 2023-12-26

1 Introduction

An effective and widely used marketing strategy is direct marketing which establishes advertising as well as a response channel to communicate and interact with existing and potential customers to persuade them to buy certain products or services. With sophisticated tools, businesses nowadays can store the data collected through such campaigns which are later analysed to reveal interesting facts about customers' buying behaviour, brand choices, pricing, etc., and use this knowledge to plan for customised marketing to better target customers for possible revenue increase. Ubiquitous telecommunication and information technology have made telemarketing an attractive means of direct marketing, allowing businesses to reach a large customer base within a short span of time and analyse the collected data quickly using business intelligence (BI) tools. Enterprises most effectively using telemarketing include, among others, telecommunication service providers, financial institutions, banks, and insurance companies[1]. On the other hand, a study shows that 80% of consumers find telemarketing annoying and a potential source of privacy breaches[2]. Such findings are also supported by other studies in recent times[3, 4]. This underlines the need for a targeted and well-informed marketing strategy whereby, not everyone, but a niche group of customers can be targeted on the basis of their personal, social, and financial data which are specifically relevant to the product(s).

The success of a telemarketing campaign is largely dependent on the selection of an appropriate set of target customers who are most likely to buy the advertised products or services. This is, however, an extremely difficult problem to solve within a reasonable time frame to be useful for businesses, which is otherwise known as an NP-hard problem, in general[5]. Approaching the wrong set of customers on the contrary will waste money, time, and energy for the enterprise and perhaps hamper reputation as it would make some customers annoyed. To attract the right set of customers, it is important to analyse the attributes and profile of the customers to accurately assess how closely those match with the product/service attributes, indicating a likelihood of a sale. Data mining and machine learning (ML) techniques are good candidates for predicting such likelihood, and their efficacy has been demonstrated in many similar types of tasks, e.g., predicting churn customers in telecommunication services[6], food sale prediction in retail[7], and prediction of company failure in the hospitality sector[8]. Both the domestic and global markets nowadays have become highly competitive, and the COVID-19 pandemic has forced many businesses to move to online electronic commerce platforms, making the businesses even more competitive and resorting to digital marketing to reach a wider base of customers. This makes it more important than ever for business organisations to become more innovative in their marketing strategy to attract new customers while retaining the existing ones in a drive to sell the right product to the right customers at the right time to boost revenue and profit[9]. This undermines the need for machine learning based intelligent models to predict suitable customer target groups and assist in refining the market strategy. This intelligent model can be directly embedded into the customer relationship management system (CRM) to better design targeted interaction with customers for improved customer experience, satisfaction, retention, and service[10] and assist in making automated actionable marketing decisions[11] while boosting the revenue prospect. The focus of this paper is on building a machine learning aided decision support system that would be capable of accurately predicting whether a customer is likely to subscribe to a given advertised product.

Several works have been reported in Refs. [12–15] that proposed machine learning based models to predict the outcome of telemarketing campaigns. Building these models requires the pre-processing of data at the first stage and then an algorithm to train the model to learn the inherent knowledge hidden in the data. Feature selection techniques (e.g., Chi-square test[16], mutual information[17], and co-relation coefficient[18]) are used to find those features that are not only relevant but also highly influential for making predictions by the model. In model training, machine learning algorithms like Naïve Bayes (NB)[19], decision tree (DT)[20], classification and regression tree (CART)[21], neural network (NN)[22], support vector machine (SVM)[23], and K-nearest neighbour (KNN)[24] have been employed in different studies for predicting telemarketing outcomes and reported varied success of the training algorithms. While these works used only a single classifier for model building, another stream of research used an ensemble of classifiers with the aim of improving prediction accuracy[25–28].

The above-mentioned works mainly used the Portuguese bank datasets developed by Moro et al.[12, 13]] or a shorter version of these datasets. Both of these datasets are imbalanced in nature, whereby the number of unsuccessful instances (customers not accepting the offer) highly outweighs the number of successful instances. Most machine learning algorithms are designed to work the best when the dataset is nearly balanced in both classes. It was shown that which sampling technique to use may be influenced by the classifier selected; e.g., in Ref. [29], undersampling with random forest (RF) was found to be better, while oversampling was more suitable for NN and SVM for the same dataset used. However, further research is needed to clearly understand the impact of oversampling in telemarketing data with respect to a wide range of classifiers and find a better strategy to handle this challenge.

In addition to finding suitable model-building approaches and means to handle the class imbalance in the dataset, another issue is how to handle model retraining. Ideally, a bank or other business organisation would be able to create a baseline dataset first and later expand it as new data become available. In the context of telemarketing, it is highly relevant because a customer's attitude towards a product or offer may change over time due to many factors or a combination of them including the present economic situation, information availability, changed perception about products or offers, and availability/discontinuation of alternative offers. What was deemed unattractive six months ago may appear attractive to a customer now and vice versa. With the current advancement in business software tools, now it is far easier to collect data. That is why retraining a previously built model with the newly acquired data in a computationally efficient way is highly important so that new knowledge about customers can be incorporated without wasting previously acquired knowledge into the model. Another issue worth noting is that previous studies did not assess how their models' performance degraded for the group of customers without having any previous interaction history. No work in the literature has so far addressed these issues. To address the above issues, this paper has made the following contributions:

(1) Develop an ensemble-based machine model that utilizes feature selection and data balancing techniques for enhanced customer acceptance prediction for telemarketing campaigns. Our model outperforms other reported models in the literature.

(2) Demonstrate the robustness of the model in predicting a customer's acceptance, having no previous interaction data.

(3) The model is further extended for online training whereby new knowledge in freshly acquired data is incorporated without wasting the previously captured knowledge. No such model is available in the literature, and the proposed model improves on the base model.

2 Related Work

The success of telemarketing strategies predominantly relies on contacting the appropriate customer base rather than contacting everyone[30]. Different data mining and machine learning models[12, 13, 15, 26, 31] have been proposed in the existing literature to predict telemarketing success to target an appropriate customer base. Moro et al.[12] published one of the prominent works for telemarketing outcome prediction considering the test case of a Portuguese bank. They collected their dataset from 17 campaigns over a few years (May 2008–November 2010) and applied a cross industry standard process for data mining (CRISP DM) based methodology to determine successful outcomes. They used 16 features in their work and applied single classifier based machine learning models, such as NB, DT, and SVM. Their results indicated that the SVM classifier is more suitable for predicting successful outcomes. They further extended their initial work, collected a similar but different dataset in Ref. [13], and predicted telemarketing success. In this work, they analysed 150 features related to bank customers, their social and economic attributes, and the product under offer to predict successful outcomes. Their analysis suggested that the NN-based model with a semi-automated selection of 22 features produced the best result and achieved 91% accuracy.

The performance of the telemarketing outcome prediction model relies on the appropriate use of feature selection to identify relevant and influential features that play a key role in the decision-making process, the proper handling of imbalanced data to prevent the model from being biased and inaccurate, and the selection of suitable classification model to improve the overall performance. The following sections highlight how these key issues are addressed in the existing literature.

2.1 Feature selection

Feature selection methods are used for building better prediction models as a higher number of features may lead to lower classification accuracy[32]. Therefore, different feature selection methods for data pre-processing are used in the existing literature for telemarketing outcome prediction. To address feature selection issues, Parlar and Acaravci[14] used chi-squared and information gain methods to identify important features and assessed the performance of the NB model by using 5−15 features. They suggested that although both feature selection methods achieved similar results, they helped improve the classification performance. In contrast, Jiang[15] used a correlation coefficient based feature selection method and suggested that the logistic regression model achieved superior performance compared to NB, SVM, NN, and DT models. A similar correlation coefficient based feature selection was also used in Refs. [33, 34]. In this case, Tékouabou et al.[33] further proposed a class-membership based (CMB) customised classification model to improve performance. They considered the impact of different types of variables (i.e., features) in the classification problem, assigned relevant weights to different features, calculated their predictability, and finally used a voting model to classify samples. They reported that the proposed model achieved an accuracy of 97.3% while the area under the curve (AUC) and F1 scores reported in this work are 0.959 and 0.939, respectively. Recently, Ram et al.[35] proposed an interesting work where they used the genetic algorithm (GA) for identifying the relevant feature set and extracted 12 important features (out of 20) for the bank telemarketing dataset. Their results show that the logistic regression (LR) and AdaBoost algorithm achieves the highest accuracy. Table 1 highlights some of the existing works employing different feature selection techniques. Although feature selection helped in improving the prediction accuracy in the above-mentioned works, they also made the models highly dependent on the availability of historical information, i.e., information from previous campaigns, such as duration of the last call, number of days since the last contact, the outcome of the previous call, etc. However, such information may not be available for new customers or a completely new product or service being offered. Existing works did not address this issue.

A few works in the literature built their prediction model without using any feature selection methods. For example, Bahari and Elayidom[10] did not use any feature selection model and reported that the multi-layer perception neural network (MLPNN) classifier outperformed the NB classifier and achieved an accuracy of 88.63%. An interesting work by Lahmiri[1] considered the prediction task as a two-step process where the first step involved independent training of multiple backpropagation neural networks (BPNN) with different information (customer and campaign information) and the latter step used the prediction of previous models to produce an outcome. Their results indicated that the two-step system achieved better performance compared to individual classifiers. In contrast, artificial neural network (ANN) was employed in Ghatasheh et al.[36] to improve the prediction performance. A similar ANN-based model was also proposed by Selma[37]. However, such ANN-based models are time-consuming to build compared to

Table 1 Feature selection methods in existing works for telemarketing success prediction.

Reference	Number of features	Feature selection model	ML models used	Ensemble model used?	Imbalanced data handling technique used?	Model retraining used?
Moro et al.[12]	16	Manual	NB, DT, and SVM	No	No	No
Jiang[15]	20	Correlation coefficient based	LR, NB, SVM, NN, and DT	No	No	No
Parlar and Acaravci[14]	5−15	Chi-square and information gain	NB	No	No	No
Tékouabou et al.[33]	18	Correlation-based	CMB, SVM, ANN, NB, KNN, DT, and LR	No	No	No
Kaisar and Rashid[28]	16	Information gain	RF, ANN, SVM, and KNN	Yes	No	No
Ram et al.[35]	20	GA	LR, KNN, RF, gradient boosting (GB), AdaBoost, DT, and extra-tree	Yes	No	No

DT-based models and hence computationally expensive and particularly not suitable for online learning with stream data. Singh et al.[38] compared the performance of LR, SVM, RF, and DT models for the prediction task and suggested that the RF model achieved the best result. A similar study by Hou et al.[39] compared the performance of NB, DT, RF, SVM, and NN and concluded that RF achieved the highest accuracy while NN produced the highest sensitivity value, suggesting its robustness in the prediction task. However, they did not explicitly use any feature selection or class balancing technique.

Overall, although the above-mentioned works used a highly imbalanced dataset, this issue was not explicitly addressed in those articles.

2.2 Imbalanced data handling

In practical scenarios, data obtained from a telemarketing campaign are highly skewed, i.e., the number of customers subscribing to an offer is usually significantly lower (minority class) compared to the number of customers who do not (majority class). However, the identification of the minority class samples is more important for a successful campaign to ensure that enterprise resources are optimally used.

Considering the class imbalance problem, Abu-Srhan et al.[40] used different oversampling techniques including synthetic minority oversampling technique (SMOTE), adaptive synthetic (ADASYN), random oversampling (ROS), adjusting the direction of the synthetic minority class (ADOMS), selective preprocessing of imbalanced data (SPIDER), and agglomerative hierarchical clustering (AHC) to make the dataset more balanced and applied RF, SVM, NN, NB, and KNN models to measure the performance. For the Portuguese bank dataset[12], the combination of RF and SMOTE achieved the best result with an overall accuracy of 89.98%. In contrast, Miguéis et al.[29] compared the performance of SMOTE and easy ensemble oversampling and undersampling techniques, respectively, to assess their performance. Their results suggested that although easy ensemble produced better results when applied with the RF model, it was not a suitable method when classification models, such as LR, NN, and SVM, are used. By comparing the performance of NN, RF, LR, NB, and AdaBoost classification models on a balanced and unbalanced version of the Portuguese bank dataset, Vitorio and Marques[41] underlined the significance of data balancing. They balanced the dataset using random undersampling and showed that doing so improved the accuracy of identifying minority classes. The training phase, however, might suffer from information loss as a result of random undersampling. A recent work by Safarkhani and Moro[42] used a resampling technique to balance the dataset and improved the accuracy up to 94.39% for telemarketing outcome prediction. However, the above approaches did not use an ensemble-based model to combine the prediction performance of different learners, which can be helpful across different datasets to minimize the dependency of the model on the underlying dataset. A few existing works on imbalanced data handling for telemarketing outcome prediction are highlighted in Table 2.

Table 2 Existing works on imbalanced data handling for telemarketing success prediction using machine learning.

Reference	Number of features	Feature selection model	ML models used	Ensemble model used?	Imbalanced data handling technique used?	Model retraining used?
Safarkhani and Moro[42]	20	Dimension reduction	J48, NB, and LR	No	Resampling	No
Abu-Srhan et al.[40]	16	None	RF, SVM, NN, NB, and KNN	Yes	SMOTE, ADASYN, ROS, SPIDER, AHC, and ADOMS	No
Miguéis et al.[29]	16	None	RF, LR, NN, and SVM	Yes	Easy ensemble and SMOTE	No
Apampa[26]	16	None	LR, DT, NB, and RF	Yes	Random undersampling	No
Pan and Tang[25]	16	None	NN and LR	Yes	Ensemble learner	No
Lawi et al.[27]	20	None	Adaboost SVM and SVM	Yes	Random undersampling	No
Feng et al.[43]	20	None	RF, Adaboost, XGBoost, GBDT, and META-DES-AAP	Yes	Undersampling	No
Vitorio and Marques[41]	16	None	NN, RF, LR, NB, and AdaBoost	Yes	Random undersampling	No

2.3 Classfication model

The performance of the prediction model significantly relies on the use of appropriate classification models. In many previous works, the prediction task employed a single classifier based classification model. However, the performance of such single classifier based models may vary depending on the dataset, and they can be further improved by incorporating an ensemble model, where multiple weak learners can be fused to make the prediction. Such ensemble learning models were used in Refs. [25–27]. Pan and Tang[25] highlighted the data imbalance problem in telemarketing outcome prediction and suggested the use of ensemble learners to address this issue. Handling an imbalanced dataset is important, especially for the detection of the minority class, which in our case refers to the customers subscribing to the offer, and missing them reflects losing a potential business opportunity. They compared NN- and LR-based bagging methods and gradient boosting techniques as their ensemble learner and suggested that the bagged NN model produced the best result. In contrast, Apampa[26] and Lawi et al.[27] used random undersampling to handle the imbalanced dataset problem to reduce the number of entries in the majority class. However, such random undersampling may lead to the loss of important information from the dataset. Similar to our current work, Muppala et al.[44] used the logistic regression based feature selection model, SMOTE oversampling technique, and an ensemble classifier to improve the prediction performance. By using SMOTE and RF, they achieved an overall accuracy of 93% for the Portuguese bank dataset[12]. In contrast, Saeed et al.[45] used a correlation matrix based feature selection model and RF-based ensemble model for the classification task. To balance the input data, they also used random undersampling and oversampling methods. On the other hand, although Feng et al.[43] did not use any feature selection method, they employed random undersampling and ensemble machine learning models. Although these works show good results (accuracy of 93%–95%), their performance can be further improved through parameter tuning. A recent interesting work by Ghatasheh et al.[46] compared the performance of multiple ensemble models and suggested that XGBoost (XGB) produced the best result when GA-based optimization techniques were used for feature selection and a cost-sensitive analysis addressed the class balancing problem. However, they did not consider any model retraining approach that may often be required in businesses when a new service or product becomes available or to target a different customer group. Existing works on telemarketing success prediction using a single classifier and ensemble models are highlighted in Table 3. Please note that Table 3 does not include the references presented in Tables 1 and 2.

Although existing literature on telemarketing outcome prediction showed promising results, many of them did not incorporate suitable feature selection techniques, ensemble classifier based models, or explicitly address the class balancing issues in the training phase. Overall, the prediction performance of existing works can be further improved by incorporating an appropriate feature selection method, class balancing technique, and ensemble machine learning model. Furthermore, none of the previous works considered the unavailability of historical campaign information when a new campaign is launched (or no previous contact for the current campaign is made) or retraining a machine learning model when information about new potential customers becomes available or the condition of an existing customer changes. Customers' attributes and behaviours towards certain products or services may change dynamically in real-life marketing scenarios, necessitating additional attention to be incorporated into the model through retraining. The current work is a step forward in that direction.

3 Proposed Approach

This study presents a telemarketing outcome prediction system that can be incorporated into a CRM system. The proposed system consists of the following phases: data collection, pre-processing the data, selecting the important data features, and building an ensemble of machine learning models to forecast whether or not a client would subscribe to an offer. Each of these phases is described in the subsections below. Figure 1 shows a schematic diagram of the whole process.

3.1 Data collection

The proposed method assumes that the CRM system can automatically gather and extract personal and financial information, as well as contact history, for clients throughout prior and current marketing campaigns. In this research, the Portuguese bank dataset, a publicly available telemarketing dataset from

Table 3 Existing works on telemarketing success prediction using single classifier and ensemble models.

Technique used	Reference	Number of features	Feature selection model	ML models used	Ensemble model used?	Imbalanced data handling technique used?	Model retraining used?
Single-classifier, no feature selection, and no imbalanced data handling	Lahmiri[1]	20	None	BPNN-PSO	No	No	No
	Ghatasheh et al.[36]	16	None	ANN	No	No	No
	Hosseini[47]	14	None	Bayesian network, LR, DT, NN, and SVM	No	No	No
	Elsalamony[31]	16	None	MLPNN, tree-augmented Naïve Bayes (TAN), LR, and DT	No	No	No
	Bahari and Elayidom[10]	16	None	MLPNN and NB	No	No	No
Single classifier with feature selection and imbalanced data handling	Moro et al.[13]	150	Semi-automated multi-stage approach	LR, DT, SVM, and NN	No	Yes	No
Ensemble classifier, no feature selection, and no imbalanced data handling	Selma[37]	20	None	ANN	Yes	No	No
	Hou et al.[39]	20	None	NB, DT, RF, SVM, and NN	Yes	No	No
	Singh et al.[38]	16	None	LR, SVM, RF, and DT	Yes	No	No
Ensemble classifier with imbalanced data handling and feature selection	Muppala et al.[44]	20	Logistic regression	RF and LR	Yes	SMOTE	No
	Saeed et al.[45]	16	Correlation matrix based	RF, LR, DT, NB, SVM, and KNN	Yes	Random undersampling and oversampling	No
	Ghatasheh et al.[46]	20	GA-based optimization	XGB CatBoost and LightGBM	Yes	Cost-sensitive analysis	No

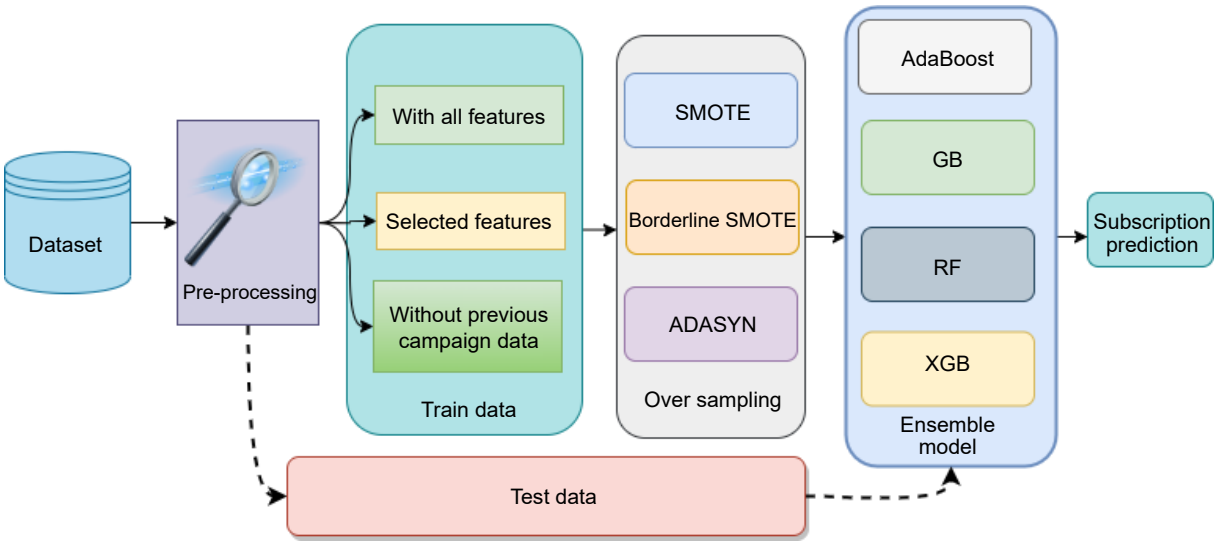

Fig. 1 Ensemble machine learning model based telemarketing outcome prediction.

University of California at Irvine Machine Learning Repository, was utilized. The dataset is available here and was introduced by Moro et al.[12] To the best of our knowledge, there are no other publicly available datasets on telemarketing campaign success, and this dataset and its minor variants have been extensively

used in the existing literature. Although there are other studies[48, 49] on telemarketing campaign success using different datasets, they are neither publicly available nor have not been tested as widely as the Portuguese bank dataset. That is why we used this dataset to compare our method to contemporary existing works. The dataset contains the results of multiple telemarketing campaigns undertaken by an undisclosed Portuguese bank during May 2008 and November 2010 to promote a long-term deposit plan. The dataset contains 45 211 records in which customers were contacted, with 5289 attempts to make the customer subscribe to the offered product being successful. This campaign had a success rate of roughly 11.69%, and it resulted in an unbalanced dataset with a significant number of negative outcomes (88.30%). For each of the records, 16 features and one class label are provided. The dataset contains no missing values. The details of the features along with their descriptions, types, and relations to the previous campaigns are presented in Table 4.

3.2 Data pre-processing

The data pre-processing phase consists of feature encoding and scaling based on the dataset's characteristics. The datasets employed in this study comprise both numeric and categorical feature values, with the bulk of features being numeric and only a few being categorical. To feed these data to the ML algorithm for training, all categorical features were turned into vectors. Various strategies, such as "label encoding" and "one hot encoding" are available to turn categorical data into vectors. The first strategy was used in this paper since the number of feature dimensions in the later technique is substantially higher. If α represents the categorical variable ($\alpha = \alpha_1, \alpha_2, ..., \alpha_n$, where α_n shows category n), the label encoding function L can be defined as

$$L(\alpha) = k, k \in 0, 1, 2, ..., n-1 \quad (1)$$

Additionally, because various features have varying values, feature scaling ensures that the range of features is standardized. In this way, no single feature, because of its large range, dominates the feature space

Table 4 Description of features for the telemarketing dataset.

Information type	Feature	Description	Type	Related to previous campaign
Customer information	Age	Customer's age at the time of contact	Number	No
	Job	Customer's current job title, e.g., admin, management, etc.	Category	No
	Marital	Customer's marital status	Category	No
	Education	Customer's highest level of education	Category	No
	Default	Does the customer have any credit default?	Binary	No
	Balance	Customer's average yearly balance (Euros)	Number	No
	Housing	Any housing loan taken by the customer?	Binary	No
	Loan	Any personal loan taken by the customer?	Binary	No
Current campaign	Contact	Which communication method was previously used? e.g., cellular or landline	Category	No
	Day	Day of the month in which the most recent contact was made	Number	No
	Month	Month of the year in which the most recent contact was made	Category	No
	Duration	Last contact's duration in seconds	Number	No
	Campaign	How many contacts are made for this customer during the current campaign?	Number	No
Contacts during previous campaign	Poutcome	Previous marketing campaign's outcome	Category	Yes
	Pdays	How many days have passed since the customer was last contacted during a previous campaign?	Number	Yes
	Previous	How many contacts were made for this customer before this campaign?	Number	Yes
Output variable (target class)	*y*	Have a term deposit subscription made by the customer?	Binary	No

and hence learning process. The minimum-maximum strategy[50] was employed in the experiments. In this case, the new feature value β_n can be derived as

$$\beta_n = \frac{\beta_o - \min_o}{\max_o - \min_o}(\max_n - \min_n) + \min_n \quad (2)$$

where β_o represents old feature value, $\min_o$ and $\max_o$ show previous maximum and minimum, respectively, and $\min_n$ and $\max_n$ show new minimum and maximum values, respectively, after scaling.

Although the dataset has some outliers, they were not removed in the pre-processing stage as they may represent important characteristics, attributes, or trends that the proposed model needs to learn, where the customers demonstrated specific behaviour in certain situations.

3.3 Determining the best feature set

The basic goal of feature selection is to identify the most important characteristics that will help provide the best recommendation. Feature selection approaches including information gain, chi-squared method, analysis of variance (ANOVA) correlation coefficient, and Kendall's rank coefficient method are used to do this. In this paper, the information-gain feature selection method has been used[20].

Mutual information (MI) is one of the most extensively used feature selection strategies, going back to the 1990s[51]. MI measures the mutual dependency between two random characteristics by determining how much information about one of the features can be derived from the other. Thus, it is related to the entropy of a random feature, which is determined by the quantity of information stored in the feature.

Choosing features is an important step in any data-driven knowledge discovery process. The current study examined the benefits and drawbacks of employing MI and data-based sensitivity analysis for feature selection in classification tasks by applying both to the previously described Portuguese bank telemarketing dataset. Figure 2 shows the MI scores of all 16 features of the dataset. Note that feature selection techniques have been extensively used in machine learning based solutions for a wide variety of business problems, such as predicting customer churn[52], detecting credit card fraud[53], predicting non-performing loans[54], credit scoring[55], and product recommendation[56]. As a standard practice, to improve performance, the top ten

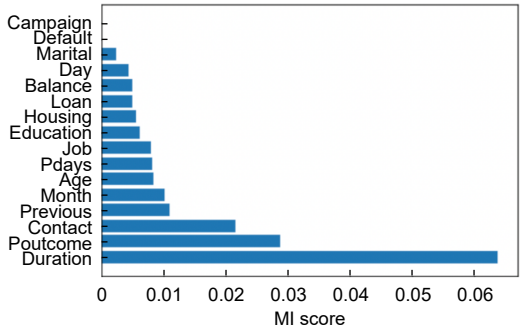

Fig. 2 MI scores of the features for the telemarketing dataset.

features were used in these studies for model building. Therefore, similar to these studies, the top ten features are also used in this paper to construct the prediction model. All of the top ten features had MI scores greater than 0.005 (MI score > 0.005).

3.4 Oversampling model

There are two approaches that are used to re-sample data in order to deal with data imbalance problems in machine learning: undersampling and oversampling. In undersampling, some samples from the majority class are removed, and therefore, some valuable information may be lost, which might otherwise be useful for capturing certain characteristics of the data. On the other hand, oversampling in the simplest form may simply duplicate minority class samples, thus random oversampling is likely to create redundant information. To overcome the redundant information issue, Chawla et al.[57] proposed the SMOTE method whereby new samples are created by interpolating between pairs of minority class samples. In this method, for each minority class sample, it takes several nearest KNN neighbours, and feature values are interpolated between that sample and one of its neighbours. For minority samples ($\boldsymbol{x}$), selecting a neighbour sample ($\boldsymbol{x}_{\text{ch}}$) from $k(=5)$ nearest neighbours and using the following equation give a new artificially created minority sample:

$$\boldsymbol{x}_{\text{new}} = \boldsymbol{x} + (\boldsymbol{x}_{\text{ch}} - \boldsymbol{x}) \times \text{random}(0,1) \quad (3)$$

where $\text{random}(0,1)$ generates a random number between 0 and 1. The value of k is normally set to 5, however, depending on the amount of oversampling required, the user can set other values. The above process continues until a balance between the two classes is reached or the desired ratio between the majority and minority classes is attained.

In literature, a few other variations of SMOTE were proposed, among them the following are notables. Borderline SMOTE proposed by Han et al.[58] put more emphasis on the borderline area in the input feature space separating the two classes and generated more minority class samples in this area. In this case, for each sample m in the minority class, its P nearest neighbours from the entire training set are identified. If the number of majority samples N_m within that P neighbours is higher (i.e., $N_m/2 \leqslant P < N_m$), it is considered a borderline sample, added to the candidate set, and used in the next step for synthetic sample generation. The synthetic sample generation follows a similar procedure as SMOTE.

Another variation (ADASYN) by He et al.[59] considered the level of difficulty in learning different minority samples and generated synthetic samples by applying weighted distribution for different minority class samples. In this case, the total number of synthetic samples to be generated for the minority class is determined as

$$\lambda = (N_m - N_l) \times \delta \tag{4}$$

where N_m and N_l represent the number of majority and minority samples, respectively, and $\delta \in [0,1]$ represents a tuning parameter to denote the balance level after synthetic sample generation. For each minority sample x_j, the number of synthetic samples to be generated is calculated as

$$\lambda_j = \psi \times \lambda \tag{5}$$

where ψ shows a density function to illustrate the relative importance of each sample in terms of the number of neighbours who belong to the majority class. A higher ψ value is assigned to a node (i.e., sample) that has a higher number of neighbours belonging to the majority class. Finally, the synthetic sample generation follows a similar approach as SMOTE.

3.5 Ensemble machine learning model

The proposed approach employed different ensemble machine learning models to assess their suitability for predicting telemarketing outcomes. Ensemble models employed in this paper include RF, AdaBoost (ADA), GB, and XGB. These methods are outlined briefly as follows.

RF: Random forest is a learning approach that constructs multiple decision trees during training and combines their outputs to enhance predictive accuracy while mitigating overfitting[60]. Basically, RF is an ensemble of decision trees. Each tree is trained on a subset of the training data, and the final prediction is determined by aggregating the predictions of individual trees. Mathematically, for a given dataset of N samples represented as $(x_1,y_1),(x_2,y_2),\ldots,(x_N,y_N)$, where x_i is the feature vector and y_i is the label, a random forest with T trees can be defined as follows:

- For each tree $t = 1$ to T:

(1) Randomly select a subset of N samples with replacement (bootstrapping): D_t.

(2) Randomly choose a subset of features (m) for node splits.

(3) Build a decision tree using D_t and the selected features.

- Aggregate predictions: For a classification task, the majority vote of tree predictions determines the final prediction.

As shown in the algorithm above, RF consists of tree construction iterations and aggregation phases. In tree construction iterations, during the first stage, for each tree iteration $t = 1$ to T, a carefully orchestrated process takes place. This process is designed to ensure diversity among the individual trees within the ensemble. To achieve this, a technique known as bootstrapping is employed. Bootstrapping involves randomly selecting a subset of N samples from the training data, and this selection is done with replacement. This creates a new dataset D_t for each tree, introducing variation and diversity in the data each tree learns from. Furthermore, to imbue the ensemble with different perspectives, a subset of features is randomly chosen for making node splits within each decision tree. This selection of features introduces another layer of variation, ensuring that each tree focuses on a different subset of features. In other words, the trees within the ensemble are not only learning from different subsets of data but also considering different subsets of features during their growth. This means that each tree in the ensemble captures distinct patterns and relationships present in the data, contributing its unique insights to the collective wisdom of the ensemble.

Once all the individual decision trees have been constructed, the second stage involves aggregating their predictions to arrive at the final prediction. In the context of classification tasks, the predictions of each tree are taken into account. Each tree "votes" for a

particular class, and the class that receives the majority of votes becomes the ensemble's final prediction. This majority voting mechanism ensures that the collective decision of the ensemble is robust and less prone to errors made by individual trees. This aggregation also has a smoothing effect on the predictions, reducing the impact of outliers and noise.

ADA: AdaBoost is a boosting technique that constructs an ensemble of weak learners sequentially. Each weak learner is trained to correct the mistakes of the previous ones by assigning higher weights to misclassified samples[61]. Consider the dataset $(x_1,y_1),(x_2,y_2),\ldots,(x_N,y_N)$, AdaBoost works as follows:

- Initialization: Set initial sample weights $w_i=\dfrac{1}{N}$ for $i=1$ to N.
- For each iteration $t=1$ to T:

(1) Train a weak classifier h_t using the weighted dataset.

(2) Calculate the classification error $\varepsilon_t=\dfrac{\sum_{i=1}^{N} w_i I(y_i\neq h_t(x_i))}{\sum_{i=1}^{N} w_i}$, where I is the indicator function.

(3) Compute the classifier weight $\alpha_t=0.5\ln\left(\dfrac{1-\varepsilon_t}{\varepsilon_t}\right)$.

(4) Update sample weights: $w_{i+1}=w_i\mathrm{e}^{-(\alpha_t y_i h_t(x_i))}/Z_t$, where Z_t is a normalization factor to ensure the weights sum to 1.

- Final prediction: Aggregate the weighted predictions of weak learners: $H(x)=\mathrm{sign}\left(\sum_{t=1}^{T}\alpha_t h_t(x)\right)$.

As shown above, at the outset, the algorithm assigns initial weight to each sample in the training dataset. These weights are uniformly distributed to ensure that every sample is equally influential in the early stages of learning. As the algorithm progresses, it delves into a series of iterations, each of which dedicated to the construction of a weak classifier. These classifiers are aptly named "weak" because they surpass random chance performance by only a marginal degree. For each iteration $t=1$ to T, the process unfolds in a carefully orchestrated manner.

(1) Train a weak classifier: A weak classifier h_t is trained on the weighted dataset, such as a decision stump, which is a decision tree with just a single level. The goal here is not to attain perfect accuracy but to perform slightly better than random guessing.

(2) Calculate classification error: The performance of the current weak classifier is assessed by calculating the classification error ε_t. This error is computed by summing the weights of the samples that the classifier misclassifies and then dividing it by the sum of all sample weights. This provides a weighted measure of how well the classifier handles the data.

(3) Compute classifier weight: The impact of each weak classifier is quantified by the classifier weight α_t. This weight is determined using a logarithmic function that takes into account the classification error. It is computed as $0.5\ln\left(\dfrac{1-\varepsilon_t}{\varepsilon_t}\right)$, reflecting the classifier's performance and influence on the final prediction.

(4) Update sample weights: The heart of AdaBoost lies in its ability to adapt to misclassified samples. The algorithm adjusts the weights of the samples based on their classification outcomes by employing a factor $\mathrm{e}^{-(\alpha_t y_i h_t(x_i))}$, where y_i is the true label and $h_t(x_i)$ is the weak classifier's prediction for the i-th sample. A normalization factor Z_t is applied to ensure that the updated weights sum up to 1.

AdaBoost brings together the individual weak classifiers to form the ensemble's final prediction. This aggregation involves summing the weighted predictions of all weak classifiers. Each weak classifier's prediction is multiplied by its corresponding weight α_t, and their signed sum determines the ensemble's prediction. If the sum is positive, the final prediction is towards positive class, otherwise towards negative class in binary classification. The algorithm's adaptability and collective strength have made it a cornerstone in the realm of ensemble learning.

GB: Gradient boosting builds an ensemble of weak learners in a stepwise manner, with each learner focusing on reducing the errors of the previous learners by fitting to the negative gradients (residuals) of the loss function[62]. Having the dataset (x_1,y_1), $(x_2,y_2),\ldots,(x_N,y_N)$, and a differentiable loss function $L(y,F(x))$, GB works as follows:

- Initialization: Set $F_0(x)$ to an initial constant (e.g., mean of target labels).
- For each iteration $t=1$ to T:

(1) Compute the negative gradient (residuals) of the loss: $r_t=-\left.\dfrac{\partial L(y,F(x))}{\partial F(x)}\right|_{F(x)=F_{t-1}(x)}$.

(2) Train a weak learner h_t to fit the residuals r_t.

(3) Update the model: $F_t(x)=F_{t-1}(x)+\eta h_t(x)$, where η is the learning rate.

- Final prediction: $F_T(x) = \sum_{t=1}^{T} \eta h_t(x)$.

The above algorithm starts by setting the initial prediction function $F_0(x)$ to a constant value, often the mean of the target labels. This provides a starting point for subsequent improvements. For each iteration $t = 1$ to T, the algorithm follows a stepwise process to enhance the model's predictive capabilities.

(1) Compute negative gradient (residuals): At the beginning of each iteration, the algorithm computes the negative gradient (residuals) of the loss function with respect to the current prediction. This gradient represents how much the current prediction deviates from the actual target value.

(2) Train a weak learner: A weak learner h_t is then trained to fit the residuals obtained in the previous step. The goal of the weak learner is to capture the patterns and relationships in the residuals, aiming to correct the errors made by the previous predictions.

(3) Update the model: The model is updated by incorporating the predictions of the current weak learner. The new prediction function $F_t(x)$ is formed by adding the weighted prediction of the weak learner to the previous prediction $F_{t-1}(x)$. The learning rate η determines the step size of this update.

As the iterations proceed, the algorithm keeps adding predictions from successive weak learners, each aimed at reducing the residual errors. The final prediction $F_T(x)$ is a refined prediction that benefits from the collective wisdom of multiple weak learners.

XGB: XGBoost, or extreme gradient boosting, is an advanced version of the gradient boosting algorithm that enhances the model's performance by incorporating additional regularization terms and efficient optimization strategies[63]. Like gradient boosting, XGB aims to minimize an objective function to iteratively refine predictions. Similar to gradient bing, XGB also employs an objective function J that encapsulates both the loss function $L(y_i, F_t(x_i))$ and various regularization terms. The objective function is defined as

$$J = \sum_{i=1}^{N} [L(y_i, F_t(x_i)) + \Omega(f_t)] + \gamma \Omega(F_t) + \eta \Phi_t,$$

where

- $L(y_i, F_t(x_i))$ represents the loss function for the t-th iteration. This term quantifies the discrepancy between predicted values and actual target values.
- $\Omega(f_t)$ is a regularization term associated with the t-th tree. It encourages simpler trees, preventing overfitting.
- γ is a regularization parameter for the overall ensemble. It controls the trade-off between fitting the data and maintaining model simplicity.
- $\Omega(F_t)$ is a regularization term for the ensemble itself. It serves as an additional control mechanism to prevent overfitting of the ensemble.
- η represents the learning rate (i.e., step size).
- Φ_t is a measure of tree complexity, ensuring that trees do not become too complex.

XGB optimizes this objective function using second-order Taylor expansions, allowing for efficient updates to the model while considering various regularization terms. By carefully controlling the complexity of individual trees and the ensemble as a whole, XGB creates a predictive model that is not only accurate but also resilient to overfitting, making it a favored choice in machine learning competitions and real-world applications.

These ensemble models offer distinct advantages. Random forest mitigates overfitting, AdaBoost excels in handling misclassified samples, gradient boosting focuses on residuals, and XGBoost optimizes through regularization.

3.6 Online learning

Online learning trains the learner incrementally by providing sequential data instances. These data instances can be provided individually or in small groups (also known as mini-batches). The online learning method allows the system to learn from the new data on the fly as it receives new mini-batches of (or individual) data in a predefined timeframe. This method is very efficient for the current e-commerce, banking, stock market, and other systems where the system receives data in a continuous flow or in batch mode. The online learner is fast and requires fewer resources than the traditional batch learners. This method is also good for handling concept drifts and can be used for out-of-core learning. In out-of-core learning, a huge dataset can be used to train the system when the full dataset is even larger than the main memory. In this case, the learning model can train itself with the portion of the full dataset and repeat the process with the other portions until the full training dataset is used. As the banking sector has a number of factors (e.g., interest rate, pandemic, and political situation) that can change the borrowing ability or

lending criteria, it is important to build a proper online learning algorithm so that the system can adopt the changes as they happen.

In the proposed model for online learning (referring to Fig. 3), a random forest classifier is first trained using the original training dataset, which is over sampled using oversampling techniques. The classifier is then improved by retraining as new batches of data arrive. As inputs, the model receives the original classifier and the new batch of data. The new batch of data are split into train and test sets, and these are augmented with the original train and test data, respectively. Afterward, the false negative rates (FNR) with the original test data and the new augmented test data are calculated using the original classifier. If the prediction falls below a predefined threshold δ, the augmented train data are over sampled, and the classifier is retrained using the same trees as in the original classifier. If the FNR does not improve after this step, new trees are added to the classifier. At first, the misclassified samples are identified, and their average misclassification probability (i.e., the probability of classifying a minority class sample into the majority class) is calculated. The calculation of the required number of additional trees (T_m) is shown as follows:

$$T_m = T_n \times P_a \tag{6}$$

where T_n is the number of trees in the old classifier and P_a is the average misclassification probability. The misclassification probability is calculated by taking the misclassified samples (n_{ji}) from class j into class i divided by the total number of misclassification (n_j) in the samples from class j, i.e., $P(i|j) = \frac{n_{ji}}{n_j}$. Thereafter, T_m trees are added to the original classifier, and only the newly added trees are retrained. The retraining strategy involves tagging each new tree with a unique identifier upon its addition and keeping a list or flag to identify them. Trees are represented as instances of a class with a method named train() in an object-oriented framework. New tree instances are placed in a distinct structure, and the train() method is invoked to train them using the augmented training data, which comprises both the initial data and new incoming data (e.g., batch-1 and batch-2 data). After this training phase, these newly trained trees are integrated back into the forest to construct the updated classifier C_r (shown in the upper right side of Fig. 3), which now embodies a blend of original and new knowledge.

3.7 Model evaluation metrics

Certain performance metrics are used to evaluate the

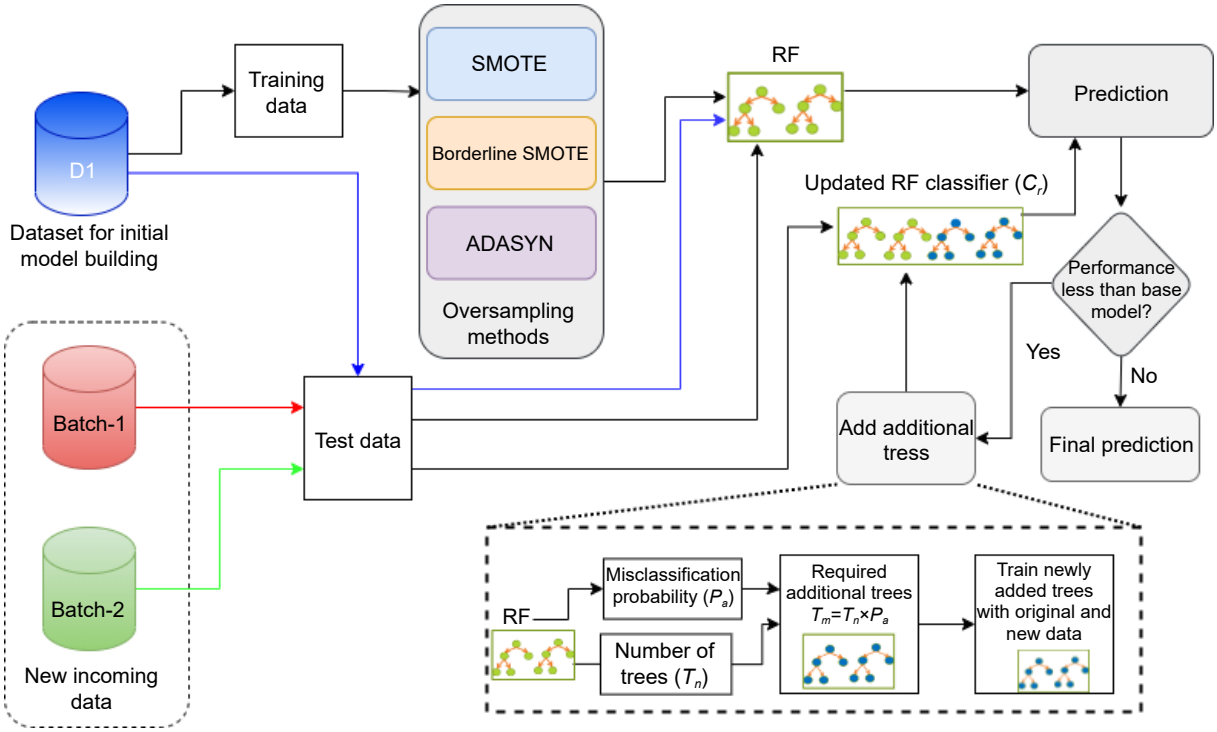

Fig. 3 Online learning method for our model. The newly added trees are shown as schematic diagram-coloured nodes.

algorithms' success in detecting potential subscriptions and determine the viability of the suggested model. The four elements of the confusion matrix used in this paper can be defined as

- True positive (TP) = Model successfully predicts a potential subscriber.
- True negative (TN) = Model correctly predicts a non-subscriber.
- False positive (FP) = Incorrect prediction when a non-subscriber instance is predicted as a subscriber.
- False negative (FN) = Incorrect prediction when a subscriber is incorrectly predicted as a non-subscriber.

Accuracy (ACC): The percentage of predictions for subscriber and non-subscriber classes with N being the total number of samples.

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{N} \times 100\% \tag{7}$$

Precision (PREC): It specifies the proportion of predictions that really subscribed out from that class.

$$\mathrm{PREC} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \times 100\% \tag{8}$$

Recall (REC): The proportion of actual subscribers that is correctly recorded.

$$\mathrm{REC} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \times 100\% \tag{9}$$

F1-score (F1): The evaluation of a model's general-class correctness.

$$\mathrm{F1} = \frac{2 \times \mathrm{Sensitivity} \times \mathrm{Precision}}{\mathrm{Sensitivity} + \mathrm{Precision}} \tag{10}$$

Receiver operating curve: Receiver operating curve (ROC) region maps TP versus FP to get performance analysis of specific class detection.

4 Performance Evaluation

The proposed model and its online learning variant have been rigorously tested with the dataset mentioned earlier. Ten-fold cross-validation is for model evaluation as this is the most accepted form to assess the true strength of a learning system. The models were built and evaluated under different scenarios, and the results are presented below.

4.1 Performance using all features

In the first experiment, all the features from the given dataset were considered, and the results for the ensemble methods using SMOTE, ADASYN, and borderline SMOTE (BSMOTE) are presented in Table 5. Table 5 shows that random forest achieved the highest accuracy (95%) with SMOTE oversampling technique, and GB achieved the lowest accuracy (89%) when BSMOTE oversampling method was applied. Similarly, RF with SMOTE achieved the best precision (74%) and recall (93%) while GB with BSMOTE attained the lowest precision (52%) and AdaBoost with ADASYN attained the lowest recall (78%) values. For an imbalanced dataset, the recall value carries more importance than the accuracy where a lower recall actually suggests that a high number of customers who actually subscribed to the term deposit were misclassified as non-subscribers during the prediction phase. This can lead to incorrect decision-making. From Table 5, it is evident that the precision value is relatively low for most of the learners, yet the RF model outperforms other models.

Table 5 Performance of ensemble models using SMOTE, ADASYN, and BSMOTE when all features are used.

Ensemble model	Oversampling method	ACC (%)	PREC (%)	REC (%)	F1	TP	TN	FP	FN
AdaBoost	ADASYN	90	54	78	0.64	1238	10 932	1045	349
	SMOTE	91	59	82	0.69	1295	11 084	893	292
	BSMOTE	90	54	80	0.64	1264	10 892	1085	323
GB	ADASYN	90	55	84	0.67	1337	10 885	1092	250
	SMOTE	90	56	83	0.67	1323	10 926	1051	264
	BSMOTE	89	52	82	0.64	1309	10 763	1214	278
XGB	ADASYN	91	58	86	0.70	1372	10 998	979	215
	SMOTE	92	60	90	0.72	1424	11 028	949	163
	BSMOTE	91	59	87	0.70	1381	11 016	961	206
RF	ADASYN	93	64	90	0.75	1436	11 156	821	151
	SMOTE	95	74	93	0.82	1478	11 458	519	109
	BSMOTE	94	69	92	0.79	1464	11 312	665	123

4.2 Performance using selected features

In the second experiment, mutual information gain feature selection techniques are used to select the best ten relevant features, and the performance of the models is shown in Table 6. Similar to the first experiment, random forest with SMOTE achieved the highest accuracy (99%), which means the application of feature selection increased the overall accuracy by about 4%. On the other hand, GB and AdaBoost with BSMOTE achieved the lowest results (92%). Similar improvements are also exhibited for recall and precision values where RF performed better than other learners.

Figure 4 shows the AUC value for the ROC curve for four different classifiers with the top ten selected features used in our model. From Fig. 4a, it is observed that the area under the ROC curve (AUROC) value is 0.9187, 0.9212, and 0.9008 for the ADASYN,

Table 6 Performance of ensemble models with top ten selected features.

Ensemble model	Oversampling method	ACC (%)	PREC (%)	REC (%)	F1	TP	TN	FP	FN
AdaBoost	ADASYN	93	68	81	0.74	1288	11 369	608	299
	SMOTE	94	70	84	0.77	1334	11 417	560	253
	BSMOTE	92	62	82	0.71	1299	11 195	782	288
GB	ADASYN	93	64	87	0.74	1376	11 202	775	211
	SMOTE	94	68	86	0.76	1371	11 319	658	216
	BSMOTE	92	63	86	0.73	1360	11 182	795	227
XGB	ADASYN	95	72	90	0.80	1424	11 412	565	163
	SMOTE	95	74	93	0.83	1481	11 467	510	106
	BSMOTE	94	69	90	0.78	1421	11 334	643	166
RF	ADASYN	97	82	95	0.88	1501	11 658	319	86
	SMOTE	99	93	98	0.95	1548	11 852	125	39
	BSMOTE	98	85	96	0.90	1517	11 718	259	70

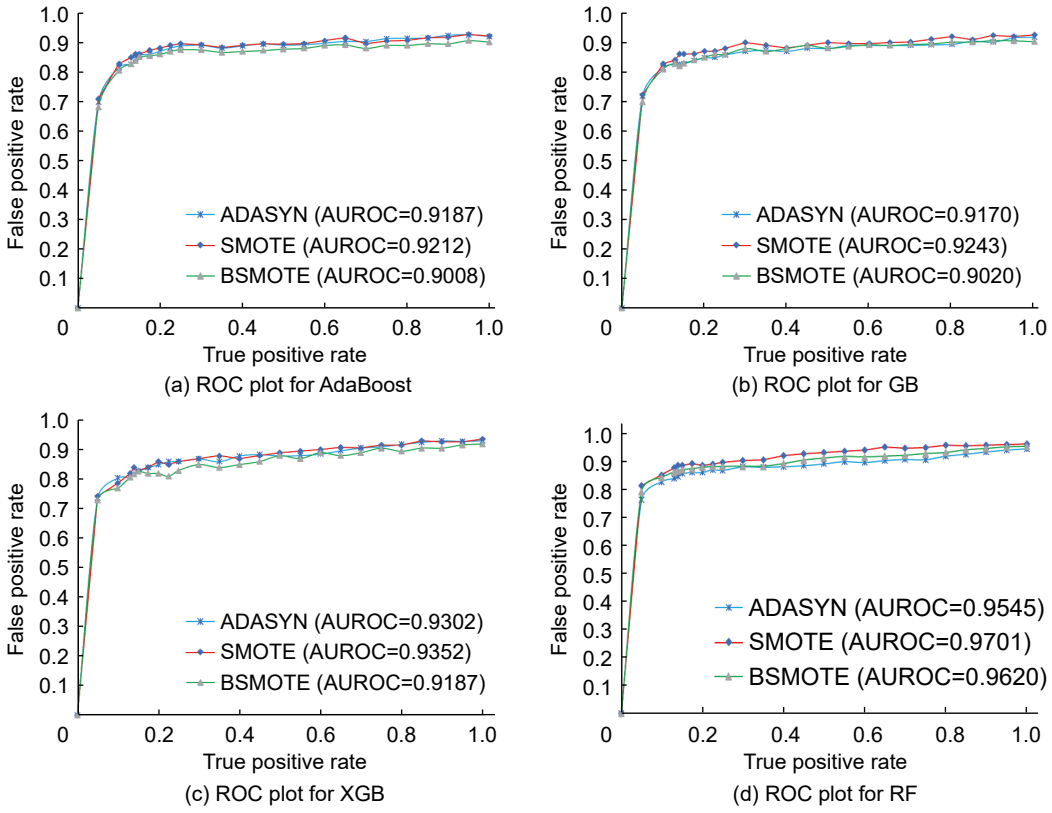

Fig. 4 ROC curves of ensemble models for different oversampling techniques with top ten selected features.

SMOTE, and BSMOTE oversampling methods, respectively. The value of AUC in the ROC curve shows better performance after using the SMOTE oversampling methods for all classifiers. Overall, RF achieved better results for all oversampling techniques, e.g., ADASYN (0.9545), SMOTE (0.9701), and BSMOTE (0.9620). The combination of SMOTE and RF attained the best AUC value of 0.9701 (refer to Fig. 4d).

4.3 Performance when previous campaign data are unavailable

In the third experiment, the performance of ensemble methods using SMOTE, ADASYN, and BSMOTE without prior campaign data is evaluated, and the results are shown in Table 7. In real-world circumstances, there may be customers for whom there is no previous campaign history. The model's performance for these customers will demonstrate how well it can handle such situations. To achieve this, features from the original dataset that are associated with past campaigns (previous, pdays, and poutcome) were discarded. Then, among the remaining 13 features, the MI score was used to identify the top ten features. For this part, an 80%−20% split was used for testing and training. Finally, the model was constructed by applying the oversampling techniques to the training samples. The generated models were then evaluated on the test data, and the results are presented in Table 7. Table 7 shows that random forest achieved the highest accuracy (92%) with SMOTE oversampling technique, and that AdaBoost with ADASYN, GB with SMOTE, and GB with BSMOTE achieved the lowest accuracy 81%. For recall, RF with SMOTE achieved the best results (90%) while AdaBoost with ADASYN achieved the lowest results (71%). Similar to the first and second experiments, the precision value decreased for all learners, though RF with SMOTE yielded the best results.

4.4 Comparison with other models

The result of our model is then compared with the results reported in Refs. [10, 26, 28, 33, 45]. The results achieved by RF with SMOTE are used as the basis of comparison as this model achieved the best results in our approach. From Table 8, among the existing works, Saeed et al.[45] obtained their best result (accuracy: 95.6%) using their proposed CMB model. Our model outperformed the existing models, exhibiting improved performance metrics, and achieved 98.6%, 93.4%, 98.1%, 0.975, and 0.954 in accuracy, precision, recall, AUC, and F1-score, respectively.

4.5 Performance of our model for online learning

The principle of online learning dictates that newly available data should be generated later than the dataset used for model building to assess whether the model can cope with new samples and produce satisfactory performance. The bank telemarketing dataset that we used is already ordered by date. Therefore, to evaluate the performance of the proposed online learning model, the first 70% of the original dataset was used to train the initial model. The next 10% was used to test the performance of the initial model. With the remaining

Table 7 Performance of ensemble methods using SMOTE, ADASYN, and borderline SMOTE when previous campaign data are unavailable.

Ensemble model	Oversampling method	ACC (%)	PREC (%)	REC (%)	F1	TP	TN	FP	FN
AdaBoost	ADASYN	81	35	71	0.47	1120	9893	2084	467
	SMOTE	84	40	75	0.52	1189	10 174	1803	398
	BSMOTE	82	36	73	0.49	1155	9957	2020	432
GB	ADASYN	85	42	79	0.55	1258	10 243	1734	329
	SMOTE	81	36	75	0.48	1191	9839	2138	396
	BSMOTE	81	35	75	0.48	1189	9779	2198	398
XGB	ADASYN	84	40	79	0.53	1258	10 081	1896	329
	SMOTE	84	41	82	0.54	1302	10 079	1898	285
	BSMOTE	86	44	82	0.57	1296	10 341	1636	291
RF	ADASYN	89	52	85	0.64	1347	10 724	1253	240
	SMOTE	92	61	90	0.72	1421	11 060	917	166
	BSMOTE	89	51	86	0.64	1368	10 686	1291	219

Table 8 Comparison with previous works.

Article	Accuracy (%)	Precision (%)	Recall (%)	AUC	F1-score
Apampa[26]	89.1	36.7	20.2	0.607	–
Bahari and Elayidom[10]	88.6	40.8	50.8	–	0.453
Elsalamony[31]	90.5	45.6	62.2	–	0.526
Kaisar and Rashid[28]	90.2	89.0	90.0	–	0.890
Saeed et al.[45]	95.6	94.0	97.5	–	0.957
Proposed method (RF with SMOTE)	98.6	93.4	98.1	0.975	0.954

20%, two mini batches were formed, each comprising 10% (ordered by date), which are viewed as new data from the model perspective and termed as batch-1 and batch-2, respectively. A major challenge in online learning is striking the right balance between stability (retaining older knowledge) and plasticity (being receptive to new information). The use of a two-batch method provides insights into whether the model leans excessively towards historical data or is overly enthusiastic about adapting, potentially neglecting established patterns. Segmenting data into batches grants analysts enhanced clarity. They can monitor the model's performance over varied timeframes, assessing improvements or potential declines in accuracy and other vital metrics. Moreover, as additional data batches are introduced over time, it serves as a robust test of the model's robustness and its capacity to manage increasing data volumes.

Table 9 shows the results of our experiments with online learning. The threshold value δ is set to 5% (0.05) for these experiments. As Table 9 shows, the initial model trained with the original data attained an FNR of 0.1021. Then the original model's performance is tested on the new batch-1 data which it had not seen before. This batch-1 data, comprising 10% of the total, was kept separate, and it was found that the model's FNR on this batch increased significantly to 0.2023. The augmented train dataset (i.e., initial 70% training data plus half of batch-1 data) was resampled as per the proposed approach and retrained the classifier, noting that the number of trees in the RF classifier is still the same. This improves the FNR to 0.1777 on the augmented test data (original test data and remaining half of batch-1). This newly retrained model was then tested with batch-2 data, and this model's performance on the test data (in this case batch-2 data, original test data, and half of batch-1) degraded to 0.2552 (again more than 0.05). However, resampling and retraining this model classifier again do not decrease FNR by the threshold level. Hence, the number of trees is increased in the classifier for further improvement as outlined in Section 3.6. The average misclassification probability of the misclassified positive class samples was calculated to be 21.8%, rounded up to 22%. Since the number of trees in the original classifier was 200, $200 \times 22\% = 44$ new trees were added to the classifier. After retraining the additional trees, the FNR dropped to 0.0473 which is even better than the original model's FNR of 0.1021. It is noteworthy that since only the additional trees were trained in this step, the training time decreased to 358 s. In contrast, retraining the entire classifier took 391 s in previous cases. The better performance and shorter training time illustrate the efficacy of our proposed online learning for identifying potential customers.

4.6 Major finding

The above subsections (i.e., Sections 4.1–4.5) discussed the performance of the proposed model, assessing its various aspects through extensive

Table 9 Analysis of the online learning approach.

Model at different stages	TP	TN	FP	FN	ACC (%)	PREC (%)	FNR	Oversampling time (s)	Training time (s)
Initial model	475	3673	319	54	91.75	59.82	0.1021	202	391
New data (batch-1)	422	3205	787	107	80.23	34.90	0.2023	192	385
After the first resampling	435	3442	550	94	85.76	44.16	0.1777	–	435
New data (batch-2)	394	3142	850	135	78.21	31.67	0.2552	291	381
After the second resampling	431	3182	810	98	79.92	34.73	0.1853	–	392
Retrained with new trees	504	3797	195	25	95.13	72.10	0.0473	198	358

evaluation. The major findings of this work are summarized as follows:

- The incorporation of appropriate feature selection, data balancing, and ensemble classification models produced the best prediction performance. The combination of all these aspects contributed towards building a highly accurate model (98.6%).
- Analysis of the result reveals that the availability of previous campaign data aids the model to capture customer intention more accurately. However, even in the absence of that, our model is capable of producing acceptable prediction accuracy (92%).
- The proposed online learning model showed promising results, as updating the model improved performance by 4%–5%. Thus, our model makes the prediction model more realistic and up-to-date by capturing emerging trends in customer behaviour and preferences. This will facilitate the quick adoption of automated decision-making models by financial institutions.
- For the telemarketing outcome prediction task, a model's "recall" value is the most important performance metric as it represents the number of actual subscribers that are correctly predicted. Our model outperformed existing works in terms of recall, accuracy, and AUC values while achieving a similar score for precision and F1-score (Table 8). This suggests its higher capability for identifying potential customers.

5 Conclusion

The effectiveness of telemarketing programs is dependent on selecting the right potential consumer base. The advancement of data analytics technologies and machine learning models has greatly enhanced the automated decision-making process for identifying potential customers and ensuring the success of a campaign. This paper considers the imbalanced nature of the dataset and uses a scheme that enhances the performance of the machine learning model to identify potential customers by using ensemble-based machine learning methods with appropriate feature selection and oversampling approaches. It further proposes an online learning model to solve model retraining and address scenarios when a fresh group of customers is targeted or they do not have past campaign history. Extensive simulation results demonstrate that the current work outperformed previous models and also achieved promising results when online training is needed. This indicates the proposed model's suitability for potential customer selection in telemarketing campaigns.

References

[1] S. Lahmiri, A two-step system for direct bank telemarketing outcome classification, *Int. J. Intell. Syst. Account. Finance Manag.*, vol. 24, no. 1, pp. 49–55, 2017.

[2] C. Page and L. Ye, Bank managers' direct marketing dilemmas–customers' attitudes and purchase intention, *Int. J. Bank Mark.*, vol. 21, no. 3, pp. 147–163, 2003.

[3] I. T. Javed, K. Toumi, F. Alharbi, T. Margaria, and N. Crespi, Detecting nuisance calls over internet telephony using caller reputation, *Electronics*, vol. 10, no. 3, p. 353, 2021.

[4] N. T. Martin, Stop telephonin' me: The problematically narrow conception of telemarketing abuse under the TCPA, *Wisconsin Law Review*, vol. 2022, no. 4, pp. 997–1026, 2022.

[5] F. T. Nobibon, R. Leus, and F. C. R. Spieksma, Optimization models for targeted offers in direct marketing: Exact and heuristic algorithms, *Eur. J. Oper. Res.*, vol. 210, no. 3, pp. 670–683, 2011.

[6] A. Chowdhury, S. Kaisar, M. M. Rashid, S. S. Shafin, and J. Kamruzzaman, Churn prediction in telecom industry using machine learning ensembles with class balancing, in *Proc. 2021 IEEE Asia-Pacific Conf. Computer Science and Data Engineering* (*CSDE*), Brisbane, Australia, 2021, pp. 1–6.

[7] G. Tsoumakas, A survey of machine learning techniques for food sales prediction, *Artif. Intell. Rev.*, vol. 52, no. 1, pp. 441–447, 2019.

[8] J. H. Brito, J. M. Pereira, A. F. D. Silva, M. J. Angélico, A. Abreu, and S. Teixeira, Machine learning for prediction of business company failure in hospitality sector, in *Advances in Tourism, Technology and Smart Systems*, Á. Rocha, A. Abreu, J. V. D. Carvalho, D. Liberato, E. A. González, and P. Liberato, eds. Singapore: Springer, 2019, pp. 307–317.

[9] Z. You, Y. W. Si, D. Zhang, X. Zeng, S. C. H. Leung, and T. Li, A decision-making framework for precision marketing, *Expert Syst. Appl. Int. J.*, vol. 42, no. 7, pp. 3357–3367, 2015.

[10] T. F. Bahari and M. S. Elayidom, An efficient CRM-data mining framework for the prediction of customer behaviour, *Procedia Comput. Sci.*, vol. 46, pp. 725–731, 2015.

[11] A. Intezari and S. Gressel, Information and reformation in KM systems: Big data and strategic decision-making, *J. Knowl. Manag.*, vol. 21, no. 1, pp. 71–91, 2017.

[12] S. Moro, R. M. S. Laureano, and P. Cortez, Using data mining for bank direct marketing: An application of the crisp-DM methodology, in *Proc. 2011 European Simulation and Modelling Conf.*, Guimarães, Portugal, 2011, pp. 117–121.

[13] S. Moro, P. Cortez, and P. Rita, A data-driven approach to predict the success of bank telemarketing, *Decis. Support. Syst.*, vol. 62, pp. 22–31, 2014.
[14] U. Parlar and S. K. Acaravci, Using data mining techniques for detecting the important features of the bank direct marketing data, *Int. J. Econ. Financ. News.*, vol. 7, no. 2, pp. 692–696, 2017.
[15] Y. Jiang, Using logistic regression model to predict the success of bank telemarketing, *Int. J. Data Science and Technology*, vol. 4, no. 1, pp. 35–41, 2018.
[16] M. L. McHugh, The Chi-square test of independence, *Biochem. Med.*, vol. 23, no. 2, pp. 143–149, 2013.
[17] R. Battiti, Using mutual information for selecting features in supervised neural net learning, *IEEE Trans. Neural Netw.*, vol. 5, no. 4, pp. 537–550, 1994.
[18] H. H. Hsu and C. W. Hsieh, Feature selection via correlation coefficient clustering, *J. Softw.*, vol. 5, no. 12, pp. 1371–1377, 2010.
[19] D. J. Hand and K. Yu, Idiot's bayes—Not so stupid after all? *Int. Stat. Rev.*, vol. 69, no. 3, pp. 385–398, 2001.
[20] J. R. Quinlan, Induction of decision trees, *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986.
[21] G. De'ath and K. E. Fabricius, Classification and regression trees: A powerful yet simple technique for ecological data analysis, *Ecology*, vol. 81, no. 11, pp. 3178–3192, 2000.
[22] J. Lawrence, *Introduction to Neural Networks*. Nevada City, CA, USA: California Scientific Software, 1993.
[23] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, Support vector machines, *IEEE Intell. Syst. Their Appl.*, vol. 13, no. 4, pp. 18–28, 1998.
[24] L. Peterson, K-nearest neighbor, *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.
[25] Y. Pan and Z. Tang, Ensemble methods in bank direct marketing, in *Proc. 2014 11th Int. Conf. Service Systems and Service Management* (*ICSSSM*), Beijing, China, 2014, pp. 1–5.
[26] O. Apampa, Evaluation of classification and ensemble algorithms for bank customer marketing response prediction, *J. Int. Technol. Inf. Manag.*, vol. 25, no. 4, p. 6, 2016.
[27] A. Lawi, A. A. Velayaty, and Z. Zainuddin, On identifying potential direct marketing consumers using adaptive boosted support vector machine, in *Proc. 2017 4th Int. Conf. Computer Applications and Information Processing Technology* (*CAIPT*), Kuta Bali, Indonesia, 2017, pp. 1–4.
[28] S. Kaisar and M. M. Rashid, Telemarketing outcome prediction using an ensemblebased machine learning technique, in *Proc. Australasian Conf. Information Systems* (*ACIS*), Wellington, New Zealand, 2020, p. 59.
[29] V. L. Miguéis, A. S. Camanho, and J. Borges, Predicting direct marketing response in banking: Comparison of class imbalance methods, *Serv. Bus.*, vol. 11, no. 4, pp. 831–849, 2017.
[30] J. Asare-Frempong and M. Jayabalan, Predicting customer response to bank direct telemarketing campaign, in *Proc. 2017 Int. Conf. Engineering Technology and Technopreneurship* (*ICE2T*), Kuala Lumpur, Malaysia, 2017, pp. 1–4.
[31] H. A. Elsalamony, Bank direct marketing analysis of data mining techniques, *Int. J. Comput. Appl.*, vol. 85, no. 7, pp. 12–22, 2014.
[32] A. G. K. Janecek, W. N. Gansterer, M. A. Demel, and G. F. Ecker, On the relationship between feature selection and classification accuracy, in *Proc. 2008 Int. Conf. New Challenges for Feature Selection in Data Mining and Knowledge Discovery*, Antwerp, Belgium, 2008, pp. 90–105.
[33] S. C. K. Tékouabou, Ş. C. Gherghina, H. Toulni, P. N. Mata, M. N. Mata, and J. M. Martins, A machine learning framework towards bank telemarketing prediction, *J. Risk Financ. Manag.*, vol. 15, no. 6, p. 269, 2022.
[34] S. C. T. Koumétio and H. Toulni, Improving KNN model for direct marketing prediction in smart cities, in *Machine Intelligence and Data Analytics for Sustainable Future Smart Cities*, U. Ghosh, Y. Maleh, M. Alazab, and A. S. K. Pathan, eds. Cham, Switzerland: Springer, 2021, pp. 107–118.
[35] B. A. Ram, D. J. S. Kumar, and A. Lakshmanarao, Improving efficiency of machine learning model for bank customer data using genetic algorithm approach, in *Proc. Int. Conf. Innovative Computing and Communications*, Delhi, India, 2021, pp. 649–657.
[36] N. Ghatasheh, H. Faris, I. AlTaharwa, Y. Harb, and A. Harb, Business analytics in telemarketing: Cost-sensitive analysis of bank campaigns using artificial neural networks, *Appl. Sci.*, vol. 10, no. 7, p. 2581, 2020.
[37] M. Selma, Predicting the success of bank telemarketing using artificial neural network, *Int. J. Economics and Management Engineering*, vol. 14, no. 1, pp. 1–4, 2020.
[38] M. Singh, N. Dhanda, U. K. Farooqui, K. K. Gupta, and R. Verma, Prediction of client term deposit subscription using machine learning, in *Proc. 4th Int. Conf. Communication, Devices and Computing*, Haldia, India, 2023, pp. 83–93.
[39] S. Hou, Z. Cai, J. Wu, H. Du, and P. Xie, Applying machine learning to the development of prediction models for bank deposit subscription, *Int. J. Bus. Anal.*, vol. 9, no. 1, pp. 1–14, 2022.
[40] A. Abu-Srhan, B. Alhammad, S. A. Zghoul, and R. Al-Sayyed, Visualization and analysis in bank direct marketing prediction, *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 7, pp. 651–657, 2019.
[41] A. Vitorio and G. Marques, Impact of imbalanced data on bank telemarketing calls outcome forecasting using machine learning, in *Proc. 2021 Int. Conf. Data Analytics for Business and Industry* (*ICDABI*), Sakheer, Bahrain, 2021, pp. 380–384.
[42] F. Safarkhani and S. Moro, Improving the accuracy of predicting bank depositor's behavior using a decision tree, *Appl. Sci.*, vol. 11, no. 19, p. 9016, 2021.
[43] Y. Feng, Y. Yin, D. Wang, and L. Dhamotharan, A

dynamic ensemble selection method for bank telemarketing sales prediction, *J. Bus. Res.*, vol. 139, pp. 368–382, 2022.

[44] C. Muppala, S. Dandu, and A. Potluri, Efficient predictions on asymmetrical financial data using ensemble random forests, in *Proc. Third Int. Conf. Computational Intelligence and Informatics*, Hyderabad, India, 2018, pp. 361–372.

[45] S. E. Saeed, M. Hammad, and A. Alqaddoumi, Predicting customer's subscription response to bank telemarketing campaign based on machine learning algorithms, in *Proc. 2022 Int. Conf. Decision Aid Sciences and Applications* (*DASA*), Chiangrai, Thailand, 2022, pp. 1474–1478.

[46] N. Ghatasheh, I. Altaharwa, and K. Aldebei, Modeling the telemarketing process using genetic algorithms and extreme boosting: Feature selection and cost-sensitive analytical approach, *IEEE Access*, vol. 11, pp. 67806–67824, 2023.

[47] S. Hosseini, A decision support system based on machined learned Bayesian network for predicting successful direct sales marketing, *J. Manag. Anal.*, vol. 8, no. 2, pp. 295–315, 2021.

[48] M. Mitik, O. Korkmaz, P. Karagoz, I. H. Toroslu, and F. Yucel, Data mining based product marketing technique for banking products, in *Proc. 2016 IEEE 16th Int. Conf. Data Mining Workshops* (*ICDMW*), Barcelona, Spain, 2016, pp. 552–559.

[49] D. W. Tan, S. Y. Liew, and W. Yeoh, Improving telemarketing intelligence through significant proportion of target instances, in *Proc. 2014 Pacific Asia Conf. Information Systems* (*PACIS*), Chengdu, China, 2014, p. 368.

[50] S. B. Kotsiantis, D. Kanellopoulos, and P. E. Pintelas, Data preprocessing for supervised leaning, *Int. J. Computer Science*, vol. 1, no. 1, pp. 111–117, 2006.

[51] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley-Interscience, 2006.

[52] M. M. Ulkhaq, A. T. Wibowo, M. R. Tribosnia, R. Putawara, and A. B. Firdauz, Predicting customer churn: A comparison of eight machine learning techniques: A case study in an Indonesian telecommunication company, in *Proc. 2021 Int. Conf. Data Analytics for Business and Industry* (*ICDABI*), Sakheer, Bahrain, 2021, pp. 42–46.

[53] M. M. Khaled and Z. A. Aghbari, ccfDetector: Utilizing GAN and deep learning for credit card fraud detection, in *Proc. 2023 Advances in Science and Engineering Technology Int. Conf.* (*ASET*), Dubai, United Arab Emirates, 2023, pp. 1–6.

[54] C. N. Nwafor and O. Z. Nwafor, Determinants of non-performing loans: An explainable ensemble and deep neural network approach, *Finance Res. Lett.*, vol. 56, p. 104084, 2023.

[55] S. Ruiz, P. Gomes, L. Rodrigues, and J. Gama, Assembled feature selection for credit scoring in microfinance with non-traditional features, in *Proc. 23rd Int. Conf. Discovery Science*, Thessaloniki, Greece, 2020, pp. 207–216.

[56] G. Kumar and N. Parimala, A weighted sum method MCDM approach for recommending product using sentiment analysis, *Int. J. Bus. Inf. Syst.*, vol. 35, no. 2, pp. 185–203, 2020.

[57] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, SMOTE: Synthetic minority over-sampling technique, *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002.

[58] H. Han, W. Y. Wang, and B. H. Mao, Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning, in *Proc. Int. Conf. Intelligent Computing*, Hefei, China, 2005, pp. 878–887.

[59] H. He, Y. Bai, E. A. Garcia, and S. Li, ADASYN: Adaptive synthetic sampling approach for imbalanced learning, in *Proc. 2008 IEEE Int. Joint Conf. Neural Networks* (*IEEE World Congress on Computational Intelligence*), Hong Kong, China, 2008, pp. 1322–1328.

[60] L. Breiman, Random forests, *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.

[61] Q. Wang and X. Wei, The detection of network intrusion based on improved adaboost algorithm, in *Proc. 2020 4th Int. Conf. Cryptography, Security and Privacy*, Nanjing, China, 2020, pp. 84–88.

[62] J. H. Friedman, Greedy function approximation: A gradient boosting machine, *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, 2001.

[63] D. Upadhyay, J. Manero, M. Zaman, and S. Sampalli, Gradient boosting feature selection with machine learning classifiers for intrusion detection on power grids, *IEEE Trans. Netw. Serv. Manag.*, vol. 18, no. 1, pp. 1104–1116, 2021.

Shahriar Kaisar received the master degree from University of Saskatchewan, Saskatoon, Canada in 2012, and the PhD degree from Monash University, Melbourne, Australia in 2018. He is a lecturer at the Department of Information Systems and Business Analytics, RMIT University, Melbourne, Australia. His research interests include business analytics, emerging technologies, cybersecurity, and health informatics.

Md Mamunur Rashid received the PhD degree in computer science from Monash University, Melbourne, Australia in 2016. Currently, he is working as a lecturer at the School of Engineering and Technology, Central Queensland University, Rockhampton, Australia. His research interest includes data mining and knowledge discovery from wireless sensor networks, big data analytics, network security, and distributed computing.

Abdullahi Chowdhury currently is a postdoctoral researcher at the Faculty of Engineering, Computer and Mathematical Science, University of Adelaide, Adelaide, Australia. From 2006 to 2007, he was a lecturer at Royal University of Dhaka, Dhaka, Bangladesh. From 2008 to 2020, he held various positions at Telstra, Melbourne, Australia; Australian Taxation Office, Canberra, Australia; and Australia Post, Melbourne, Australia. His research interests include game theories, natural language processing, cybersecurity, human-machine interaction, and intelligent transportation systems.

Joarder Kamruzzaman received the BSc and MSc degrees in electrical and electronic engineering from Bangladesh University of Engineering and Technology, Dhaka, Bangladesh in 1986 and 1988, respectively, and the PhD degree in information systems engineering from Muroran Institute of Technology, Hokkaido, Japan in 1993. He is a professor of information technology and the director of the Centre for Smart Analytics, Federation University Australia, Ballarat, Australia. His interests include Internet of Things, machine learning, and cybersecurity. He has published 300+ peer-reviewed articles which include over 90 journal and 180 conference papers. He is the recipient of the best paper award in four international conferences: ICICS' 15, Singapore; APCC'14, Thailand; IEEE WCNC'10, Australia; and IEEE-ICNNSP'03, China. He has served many conferences in leadership capacities including program co-chair, publicity chair, track chair, and session chairs, and since 2012 as an editor of the *Elsevier Journal of Network and Computer Applications*, and he had served as the lead guest editor of *Elsevier Journal Future Generation Computer Systems*.

Sakib Shahriar Shafin received the BSc degree in electrical and electronic engineering from Islamic University of Technology, Gazipur, Bangladesh in 2021. He currently is pursuing the PhD degree in information technology at the Centre for Smart Analytics, Federation University Australia, Ballarat, Australia. His research interests include cybersecurity, Internet of Things, and machine learning.

Abebe Diro is a lecturer at the School of Accounting, Information Systems and Supply Chain, RMIT University, Melbourne, Australia. He is a cyber security scientist with interests in machine learning based cyber security and cryptography. He has made outstanding contributions to these fields with publications in high quality journals. The research outputs include pioneering work on distributed machine learning for intrusion detection in Internet of Things. Further, he has proven his ability to establish relevant research collaborations with industry through various projects, which is supported by RMIT University's emphasis on aligning research with areas of national interest. He has also established collaborations with academics in Europe, Australia, Republic of Korea, Indonesia, and Türkiye, where he has co-published journal articles with researchers. His high-quality research in cyber security and his extensive research networks in cyber security and cryptography mark him as a leading researcher at RMIT University.