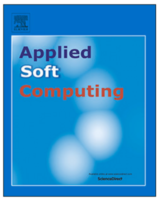
Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc



An adaptive learning approach for customer churn prediction in the telecommunication industry using evolutionary computation and Naïve Bayes

Adnan Amin [*], Awais Adnan, Sajid Anwar

Center for Excellence in Information Technology, Institute of Management Sciences, Peshawar, 25000, Pakistan

ARTICLE INFO

Article history:
Received 24 October 2022
Received in revised form 5 January 2023
Accepted 5 February 2023
Available online 21 February 2023

Keywords:
Evolutionary algorithm
Genetic Algorithm
Naïve Bayes
Adaptive learning
Churn prediction
Telecommunication

ABSTRACT

Customer churn is a complex challenge for burgeoning competitive organizations, especially in telecommunication. It refers to customers that swiftly leave a company for a competitor. Acquiring new customers has cost the telecommunication industry more than keeping existing customers. Traditionally, customer churn prediction (CCP) models are applied to aid in analyzing customer behavior and achieving prediction accuracy, which allows the telecommunication industry to target prior retention efforts toward them. However, only accurate CCP based on the available data or already trained supervised model is inadequate for efficient churn prediction, as existing approaches have not been shown or designed to learn with the skill of adaptation to respond quickly to changes in the customer behavior or a decision. Therefore, it is essential to design an approach that easily adapts to learn from new decision scenarios and provides instant insights. This study proposes an adaptive learning approach for this perplexing problem of CCP using the Naïve Bayes classifier with a Genetic Algorithm (subclass of an Evolutionary Algorithm) based feature weighting approach. Further, the performance of the proposed approach is evaluated on publicly available datasets (i.e., BigML Telco churn, IBM Telco, and Cell2Cell) which significantly enhances the prediction performance as compared to the baseline classifier (i.e., Naïve Bayes with default setting, Deep-BP-ANN, CNN, Neural Network, Linear Regression, XGBoost, KNN, Logit Boost, SVM, and PCALB methods) in terms of average precision of 0.97, 0.97, 0.98, a recall rate that stands at 0.84, 0.94, 0.97, and F1-score of 0.89, 0.96, 0.97, an MCC of 0.89, 0.96, 0.97, and accuracy 0.95, 0.97, 0.98 on subject datasets, respectively.

1. Introduction

In order to increase the telecommunication industry's revenue, each service provider is already motivated to offer more value-added services to their customers using customer relationship management (CRM) [1,2]. CRM is a business strategy aiming to ensure a higher degree of customer satisfaction. This satisfaction can be described as a customer's reaction or loyalty to a certain degree of expectation [3,4]. Loyal customers can help the telecommunication industry perform better by cutting the cost of publicity, negotiating, and attracting more new customers, while customer reaction results in customer churn [5]. The term "customer churn" is commonly used to describe those customers that cease doing business with the company or a service provider [6].

According to a survey [7], the telecommunication industry's annual churn rate ranges from 20% to 40%. The cost of retaining existing customers is 5–10 times cheaper than the associated cost of acquiring new consumers. The cost of anticipating customer attrition is 16 times less than the cost of acquiring new customers. The profit increases from 25% to 85% when the churn rate is reduced by 5%, which shows the importance of CCP in the telecommunication industry. The European Business Review [8] predicted that enterprises with a 2% turnover rate will lose $65 million per month, despite the fact that retaining current customers is less expensive than obtaining new ones. The cost of acquiring qualified customers for a network or service is more than the cost of retaining existing customers due to the significant cost, effort, and revenues [9].

Several studies [5,7,10] have shown that Machine Learning (ML) and Artificial Intelligence (AI)-driven analytics can dig massive data from CRM systems for the purpose of CCP and identify the root cause of churn. ML can assist decision-makers in making informed and targeted decisions to retain customer churns. In this context, CCP have become an instrumental tool in the telecommunication industry. Current CCP approaches are constrained by

[*] Corresponding author.
E-mail addresses: adnan.amin@imsciences.edu.pk (A. Amin), awais.adnan@imsciences.edu.pk (A. Adnan), sajid.anwar@imsciences.edu.pk (S. Anwar).

https://doi.org/10.1016/j.asoc.2023.110103
1568-4946/© 2023 Elsevier B.V. All rights reserved.

the lack of training data available. If the available data for the predictive model only partially covers the CCP problem, then the model's output will be unreliable and inaccurate, as the learning model will not provide a generalizing model reality [11]. This problem is always present because of feature reduction, resulting in partial data loss and outliers from the available data. It necessitates that the prediction model learns from new information, new instances, or the complex data of the target domain. Although state-of-the-art approaches (e.g., incremental learning) are incrementally learned when new data points are introduced and new knowledge is incorporated over time, it also requires considerable training time [12]. Additionally, the existing ML methods for churn prediction can perform well in offline mode or passive ML. However, the offline mode of ML faces challenges in such scenarios where the ML method is required to adapt changes in customer behavior or a decision. Hence, it is essential to design an approach that has the ability of adaptive learning regarding the environment and decision scenarios instead of a passive learning mode while the adaptive learning is time efficient, continuously enriched learning in real-time without loss of previous knowledge.[1]

This study proposes a novel approach based on adaptive ML for CCP in the telecommunication industry. Adaptive ML represents one of the significant concerns of ML and constitutes an open research domain that is the subject of several types of data mining methods. We describe a probabilistic algorithm, i.e., Naïve Bayes, in the context of adaptive ML at prediction time without retraining the CCP model again and again with minimal percent data lost. The significant contribution of this study can be summarized as follows: (i) employing self-learned optimum attribute weighting technique using an evolutionary algorithm (i.e., genetic algorithm) without losing data and keeping the attribute independence condition as well, and (ii) proposing a novel Adaptive Customer Churn Prediction (ACCP) model with the ability to learn continuously (i.e., brain-like improving knowledge boundary) in the telecommunication industry and maintaining good prediction accuracy.

The purpose of this research is, therefore, to provide a basic study for those in academia and industry practitioners who want to design, develop, and apply an adaptive learning-based CCP approach in competitive industries, particularly telecommunication. Listed below are some of this study's most significant contributions:

- To design a self-learned optimum feature weighing technique using a genetic algorithm (a sub-field of the Evolutionary algorithm), which keeps the attribute independence condition, and without losing any data or attributes from the original set.
- To develop an approach for designing an adaptive customer churn prediction model in the telecommunications industry based on customer behavior, which can learn new events with minimal supervision (without retraining a model from scratch) and the skill of adaptation.
- To highlight and summarize the potential research directions within the scope of our study for adaptive learning approaches for CCP in the telecommunication industry.

The rest of the paper is organized as: Section 2 explores the review of the CCP, Section 3 explains the proposed ACCP approach followed by a baseline classifier (i.e., Naive Bayes), Genetic algorithm, and evaluation measures. Section 4 describes the results and discussion. Section 5 concludes the paper.

2. Review of customer churn prediction

This section begins by defining the term "customer churn" followed by a systematic review of past studies.

2.1. Customer churn definition

Churn is defined as [13] "customers who switch from one brand to a competitor's brand". Customer churn has been interpreted in several ways across industries and domains. In general, churn can be defined simply as a prolonged period of inactivity [14]. To address the issue of customer churn in the competitive telecommunication industry, numerous practitioners and academicians have taken various approaches in a range of domains, which are discussed next in the following section.

2.2. Methods for predicting customer churn (CCP)

Nowadays, many service-oriented companies in general, and the telecommunication industry in particular, share the common goal of gaining more customers, increasing profits, and expanding the company's reputation in the target domain [15]. Due to increased globalization and competition in the telecommunication industry, customers may not always be loyal to service providers due to the availability of similar companies, offers, and service-related plans in this domain. As a result, decision-makers in the telecommunication industry attempt to retain existing customers rather than acquire new customers to avoid associated expenses. In a nutshell, customer churn has a negative impact on the company's reputation. Therefore, CCP is important not only for the company's reputation but also for retaining current customers at a lower cost [16].

Several studies [7,10,17] have explored the effects of CCP techniques, including machine learning (ML), data mining, and hybrid (ML and data mining) methods. These methods support the decision-makers in the telecommunication industry in classifying, predicting, and retaining customer churns. The most common technique in the literature for addressing this problem is "decision trees" a widely used ML method for CCP in the telecommunication industry [18]. However, Vijaya and Sivasankar [19] demonstrated the constraints of complex nonlinear connections between features in decision trees. Another study [20] investigated that pruning can improve the decision trees' performance. On the other hand, various studies [21,22] also observed some strengths of the decision trees algorithm in the context of CCP: (i) it can process both categorical and numerical data of customers, (ii) it can easily visualize and construct the classification model, and (iii) it can use a nonparametric method that does not need prior assumptions. But the main drawback of using a decision tree is that it does not have the classification's coverage rate [23]. Similarly, De Caigny et al. [5] discussed how decision tree and logistic regression ML methods had demonstrated accuracy and comprehensibility in CCP in the telecommunication industry. Despite its strengths, the decision tree frequently struggles with linear relationships between features; similarly, logistic regression frequently struggles with interaction effects between variables.

In another study [24], a neural network-based technique is applied to predict customer churn in the telecommunication industry. In connection with this study, Tsai and Lu [25] developed a hybrid approach for CCP by applying neural networks on a Self-Organized Map (SOM) and considered the American telecommunication dataset. Initially, they used neural networks to reduce data and remove superfluous samples from the training set. The previous step's output was then fed into SOM to develop a CCP model. Their findings demonstrated that a hybrid approach could

[1] https://insidebigdata.com/2019/12/15/advantages-of-adaptive-ai-over-traditional-machine-learning-models/ access at 08/09/2022 01:30 pm.

improve prediction performance more than a single neural network. However, they revealed the major limitation of their work in terms of data loss during the data reduction phase. Similarly, Olle et al. [26] explored a hybrid model to analyze the customer churn behavior in the Asian mobile operator telecommunication dataset using logistic regression and voted perception. Their findings indicated that the hybrid model is more accurate (e.g., obtained ROC value of 0.721) than the single ML method. Another study has also pursued similar work [27] in which a hybrid learning model was proposed for CCP in the telecommunication industry using weighted k-means clustering and rule induction methods. Authors have shown the improved performance of their proposed hybrid learning model in terms of accuracy, ROC and AUC evaluation measures.

The earlier work of [22] attempted to predict customer churn in the Syriatel telecom company by combining multiple ML techniques such as decision tree (DT), random forest, gradient boosted tree (GBT), and XGBoost. It was found that such an approach can predict the customers which are more likely to churn based on the obtained AUC value, which was approximately 93.3 percent. Additionally, another study [19] presented a comparative study on various boosting versions of ML methods (i.e., Ensemble learning) for CCP, where multiple learning algorithms (i.e., DT, ANN, SVM, naïve Bayes, and regression analysis) were employed to achieve high prediction performance. They applied all these targeted ML methods to the telecommunication dataset (obtained from the UCI ML repository at the University of California, Irvine). They claimed that the best performance was obtained by combining SVM-Poly with AdaBoost. However, one of the primary drawbacks of SVM is its black-box nature, which creates an illusion [28].

Several studies [7,29,30] have explored the effects of ensemble learning where CCP approaches suffer from incorrectly classifying customers using a single classifier. According to Xu et al. [7] combines multiple ML methods into one aggregated mode. A closer look at the review on ensemble learning revealed that it could achieve better classification performance than a single ML algorithm-based classifier. Koen et al. [23] combined the rotation forest and Rotboost algorithm and built an ensemble technique for CCP. The feature extraction is performed through the rotation forest ML method, while the Rotboost technique is used in combination with AdaBoost and rotation forest to predict the customer churn. They revealed that Rotboost outperformed the rotation forest in terms of accuracy while the forest achieved a higher area under the curve (AUC) and lift. However, this study's major limitation was the lack of interpretability and understandability of the customer churn factors. Therefore, De Bock et al. [31] suggested incorporating the Generalized Additive Models (GAM) technique into the ensemble method. In a nutshell, they have achieved comparatively best CCP performance than individually training the classifier with logistic regression and the GAM method.

Some other studies [24,32] emphasize the data, as the aim of CCP models is to train the classifier on the target dataset, hence enabling it to classify unknown samples correctly. If the samples of subject data are not balanced, there is a high probability that the classifier will show poor classification accuracy [33]. Therefore, the goal of the efficient predictive model is to locate the point where the model has a low bias and error rate in the data. In contrast, both studies [34,35] attempted to reduce the error rates in the data preprocessing phase for predictive modeling; thus, they usually exhibit a bias toward the majority class distribution [7]. On the other hand, Farquad et al. [32] found that traditional ML algorithms perform well in the imbalance class where the classifier tends to classify the majority class accurately because in the telecommunication industry, many customers belong to the majority class, and just a small number of samples represent the minority class. In contrast, sometimes, class imbalance may also cause conventional ML methods to go through difficulties during the learning phase, which eventually results in poor classification performance [36].

A review of prospective observational studies [37,38] have investigated six well-known oversampling methods in the telecommunication industry and analyzed the performance of these key methods, namely—Synthetic Minority Oversampling TEechnique (SMOTE), Mega trend diffusion function (MTDF), Adaptive Synthetic Sampling Approach (Adasyn), Couple top-N reverse k-nearest neighbor (CTRKNN), Majority Weighted Minority Oversampling TEchnique (MWMOTE), and Immune Centroids Oversampling Technique (ICOTE). They revealed that the overall CCP model performance using MTDF and genetic algorithm-based rules extraction for churn classification outperformed as compared to the rest of the five other oversampling methods and rule generation ML methods (learning from example module version 2 (LEM2), covering, and exhaustive). However, oversampling or under-sampling does not need to improve the performance. In contrast, Ling and Li [39] reported that oversampling of minority class samples and the under-sampling of majority class samples did not significantly improve the classification problem. Another study [25] investigated particle swarm optimization (PSO) under-sampling methods with data reduction using minimum redundancy and maximum relevance (mRMR), followed by ensemble ML methods using three algorithms, namely K-nearest neighbor (KNN), Random Forest, and Rotation Forest. They reported that handling the large dimensionality in the subject data during data preprocessing can improve the performance of the CCP in the telecommunication industry. In connection with this work, Benlan et al. [40] introduced the CCP model based on the SVM and random sampling technique, which changed the data distribution to balance the imbalanced class distribution. However, the CCP model did not improve the prediction performance by balancing the class distribution. Similarly, another study [41] explored utilizing the weighted random forest ML technique for the CCP problem; however, random forest is usually criticized for being difficult to understand and interpret.

Another interesting study [42] has introduced the Just-in-Time (JIT) concepts for CCP in the telecommunication industry. The term JIT has derived from the manufacturing philosophy that a "company produces only what is needed when needed and, in quantity, needed. The company produces only what the customer requests, to actual orders, and not for the forecasts" [43]. The telecommunication industry needs JIT specifically when a telecommunication company is newly established and wants to adopt a CCP model at a very early stage. However, in practice, previous data may not be available at the early stage in the newly introduced company, particularly for the training process of the CCP model. This type of limitation hinders the building of CCP [44]. Therefore, the CCP is hardly possible for these types of companies. Fortunately, a recent study reported that cross-company churn prediction (CCCP), is a concept of sharing data across the same industry, where matured company data is used as a training set and the data of such a company that is lacking historical data [44]. CCCP can handle the bottleneck in the single company churn prediction problem. Similarly, our previous study [42] followed the concepts of the Cross-company CCP model in the telecommunication industry using data transformation methods (i.e., log, z-score, rank, and box-cox). It was reported that the naïve Bayes method outperformed on transformed data (e.g., log, rank, and box-cox) compared to deep learner neural net, KNN and GBT.

Literature revealed that feature engineering also plays an important role in constructing the CCP model by formulating useful features from existing data. Samina Kanwal et al. [45] applied

PSO method followed by a subjective weight procedure using a domain expert to manually estimate the weights for each feature. As a result, the researchers have not only investigated the influence of features on various weights rather than feature reduction but also reported that subjective weights assignment is time-consuming and costly. Further, the authors have also claimed that a PSO-based feature selection using GBT outperformed and achieved 93% accuracy compared to 90% accuracy for the DT, 89% accuracy for KNN, and 89% accuracy with 72% precision for NB classifier. On the other hand, Umayaparvathi et al. [46] used a deep learning technique to avoid manual feature engineering and developed a hybrid deep neural network (DNN) architecture for CCP in the telecommunication industry using Crowd Analytix and Cell2Cell telecom datasets. The authors reported that manual feature selection is not necessary when using DNN, but it is necessary when using traditional models like SVM and random forest. Further, they reported the following major problems of feature engineering such as: (i) the existing feature engineering process is very time-consuming and often performed by a domain expert (manual feature ranking), which is very costly, and (ii) often it is tailored to specific data. Further, the authors proposed a deep learning algorithm that has the inherent capability of automatically feature selection for the input data and compared with traditional methods and reported that deep learning performed as well as traditional models. However, this model loses its generalization ability after training on a new observation. This means a new observation will likely override previously learnt weights, degrading model performance and increasing the cost of retraining the model from scratch. On the other hand, in order to overcome the shortcomings of manual feature ranking, Bin Luo et al. [47] investigated feature selection based on the K-means, SVD algorithms, and the feature filtering method before building the CCP model. The results indicate that the algorithm can significantly improve the accuracy while also reducing the errors associated with traditional classification. Table 1 summarizes some latest literature on CCP in the telecommunication industry.

Even though significant advancements in the research have been made in the area of incremental learning (i.e., incrementally adding new classes) and deep learning for the classification [11], It is generally considered undesirable in ML research if the data distribution changes between the model learning and prediction processes. This is referred to as domain shift, a component of the domain adaptation process. Most of the time, domain adaptation results in a decrease in the accuracy of the predictive model. As a result, a researcher in the field is concentrating on how to prevent the model's accuracy from deteriorating [52]. When the distribution of the test-set data differs from the distribution of the training set data, the field of domain adaptation is confronted with the problem of ML classification [53]. A study conducted by Francisco M. and colleagues [18] has addressed the problem of incremental learning to the training set by using deep learning techniques; however, deep learning is suffering from catastrophic forgetting, which significantly reduces the overall performance of the prediction model. On the other hand, incremental learning required a process for loading new data from discs in batches, where determining the batch size, load frequency, and hyperparameter for retraining on fresh data are difficult tasks. Similarly, ML algorithms frequently operate in a dynamically changing environment. Therefore, there is a need to close the aforementioned gap by implementing an approach adaptable to the evolution of hard data generation over time. The proposed study introduces a novel approach for ACCP in the telecommunications industry that is capable of accurately predicting difficult instances in data generation over time (dynamic environment).

3. The proposed ACCP approach

In this section, we present the overall design of the evaluation setup as well as the performance of the proposed ACCP. Fig. 1 depicts the Adaptive Customer Churn Prediction (ACCP) Approach.

3.1. Problem structure

In this section, the problem structure is described to understand the underlying problem before presenting the proposed adaptive ML approach for handling the customer churn issue in a saturated market of the telecommunication industry. As stated previously, this study describes the proposed adaptive learning approach using Naïve Bayes through an algorithmic prism.

This study aimed to predict a class-label (churn or non-churn) y={churn,non-churn} in a given set of input attributes $A \in \{a_1, a_2, a_3, \ldots, a_n\}$. An example is one pair of (X, y), for instance, X is a set of attributes that have values about the customer behavior such as *total_day_minutes*, *total_day_calls*, and total *total_day_charges*, etc., y is the target variable, and it can be either churn or non-churn. To predict the unseen labels of the target variable, the approach starts with an evaluation of each attribute's contribution during the prediction process. Features are assigned weights based on their contribution in the prediction process ($i.e., W = \{w_1, w_2, \ldots, w_n\}$), where $w_1(i.e.,\ a\ minimum\ weight\ value)$ is the weight assigned to the least significant attribute, and w_n (i.e., maximum weight value) is the weight assigned to the most significant attributes. For this purpose, a genetic algorithm is used to determine the importance of each attribute in the classification decision, such as $A = \{a_1w_1, a_1w_2, \ldots, a_1w_n\}$, and $A \in X$. Further, the prediction model based on the baseline classifier is constructed using the training set (X and Y), which includes all the attributes along with their assigned weights. The baseline classifier performs the prediction based on probabilistic method using prior ($i.e., p(y)$) and likelihood $p(X|y)$ of the target class ($i.e., y = \{churn, non-churn\}$). Moreover, the decision boundary of the baseline classifier is defined using the likelihood of the target variable values. This can be described through Eq. (1).

$$L(y|X) = \frac{P(y).P(X|y)}{P(X)} \times W \tag{1}$$

where W is the set of weights for each attribute and $p(X) = \sum_{y}^{c,nc} p(y).(p(X|y))$. Finally, the proposed approach learns instance by instance and accordingly update baseline classifiers knowledge boundary in continual learning manner.

3.2. Subject datasets

Several studies [48,54] have evaluated state-of-the-art ML methods for CCP on various private real-world telecommunication datasets. However, these private datasets prevent from reproducibility and extrapolation to further research novelty [54]. Therefore, in this study, we have used the following publicly available datasets (referred to as Dataset-1, Dataset-2, and Dataset -3 henceforth) and are considered as benchmark subject data of the telecommunication industry. Table 2 lists the characteristics of the subject datasets.[2,3,4,5]

[2] https://data.world/earino/churn.

[3] https://community.ibm.com/community/user/businessanalytics/blogs/steven-macko/2019/07/10/new-base-samples-for-ibm-cognos-analytics-1113.

[4] https://www.kdd.org/kdd-cup/view/kdd-cup-2009.

[5] https://bigml.com/user/bigml/gallery/datasets.

Table 1
Summary of some latest literature review on CCP in the telecommunication industry.

Author/Ref.	Methods used	Dataset	Evaluation metric	Outcomes
Samia Kanwal et al. [45]	Particle Swarm Optimization, Naïve Bayes, Decision Tree, KNN, Gradient Boosted Tree	Publicly available BigML Telco Dataset contains 20features and 3333 samples	Accuracy, Precision	Gradient Boosted tree achieved the highest accuracy of 89% and precision of 81% which outperformed as compared to decision tree, KNN, and Naïve Bayes.
Wu S et al. [1]	Oversampling (SMOTE), Logistic Regression, Decision Tree, Random Forest, AdaBoost, Multi-layer Perception	DS-1 (Features:19 Samples: 7032), DS-2 (Features:20 Samples:4031), DS-3 (Features:58 samples: 51 047)Publicly available	10-fold CV, Accuracy, F1-Score, Precision, Recall, AUC	AdaBoost, Random Forest and MLP Logistic regression performed best on DS-1, DS-2, and DS-3, respectively.
Xu T. et al. [7]	Stacking and soft voting for ensemble learning Regression, Decision Tree, Naive Bayes and Xgboost.	DS-1 (Features:20 samples 3333) Publicly available	Accuracy, Precision, Recall, F1-score	Using XGB in level 1, LR, DT, and NB in level 2, and soft voting in level 3, then they were able to achieve 98 percent accuracy.
Zhao M, et al. [5]	Logistic regression	Province-specific proprietary 11 255 samples was collected during the first three months of 2020.	Precision, Sensitivity, Specificity,	On the five factors, ARPU, DOU, current package value, Unit, and complaint, LR can outperform with 91 percent precision, 65 percent sensitivity, and 93 percent specificity.
Affan Ahmad T, [48] Usman M.	Cumulative Sum churn detector, Naive Bayes, DWM, MLP, and ALR have all been used.	Proprietary Dataset of 380 000 samples, covering a three-month period and containing 650 Call Detail Records from South Asian Telecom.	Accuracy, Time	In terms of performance, an improved two-sided churn detector outperformed CusumDM and ADWIN.
Ahmad A, et al. [22]	EGBoost, Decision Tree, Random Forest, Gradient Boosted Tree, and Gradient Boosted Tree were all used.	SyriaTel telecom's proprietary 9-month data of 70 TB was used to evaluate the model.	AUC	The best results were obtained when XGBoost was used.
Irfan Ullah et al. [49]	RT, J48, RF, Decision Stump, AdaboostM1, Bagging, NB, MP, LR, IBK, and LWL, Feature Selection (Info. Gain and Correlation ranking).	Utilized the publicly available dataset-1, which contained 29 distinct features 64 107 samples, Dataset-2, which originally contained 21 features and 3333 samples.	Precision, Recall, F-measure, TP Rate, FP rate, ROC area.	F-measure improved by 88% using Random Forest and J48.
Hemalatha Jain et al. [50]	CNN deep learning	Cell2Cell, Telco, and Orange datasets contains 51 048, 7048, 3333	Accuracy	Improve the accuracy of predicting customer churn by adding additional features.
Cenggoro, Tjeng Wawan et al. [18]	Deep learning	BigML Customer Churn Telco Dataset which contains 20 features and 3333 samples.	F1-Measure	Achieved predictive performance about 81.61%F1-measure.
Ibrahim Al-Shourbaji et al. [51]	Ant colony optimization	Seven Telecom Datasets with various list of features and various range of samples.	Accuracy, Fitness value matrix	Ant colony optimization with reptile search algorithm outperformed
Mishra and Reddy [13]	Deep learning by CNN	Telco dataset	Accuracy, Error rate, Precision, Recall, f-Score	An algorithm that uses deep learning by CNN performed well in terms of accuracy (86.85%), error rate (13.15%), precision (91.08%), recall (93.08%), and the f-score (92.06%).

Table 2
Characteristics of subject datasets.

Characteristics	Dataset-1	Dataset-2	Dataset-3
Description	[a]	[b]	[c]
Source URL	[1]	[2]	[3]
No. of instances	3333	7043	51 047
No. of attributes	20	21	58
Numerical Variables	15	3	35
Nominal Variables	5	18	23
Churn	17.70%	26.54%	28:82%
Non-churn	82.30%	73.46%	71.18%
Missing values	No	Yes	Yes
Class Labels	True, False	Yes, No	Yes, No

[a] BigML customer churn dataset contains samples of California telecom's customer data.
[b] IBM customer churn dataset contains instances of fictional telco company.
[c] Cell2Cell Teradata center CRM at Duke University.

3.3. Data preprocessing

Data preprocessing has a major influence on the performance of the ML models and improves the dataset's quality. Initially, the raw data was transformed into formats suitable for the baseline classifier of the proposed study. The following essential data preprocessing steps were carried out: (i) The nominal variables were transformed into corresponding numerical values (such as "yes" or "no", "true" or "false" etc. into 0s and 1s) using the ordinal encoder method [55]. This is accomplished with a recommendation of a study [56] by grouping into certain categories based on the number of samples in each group, (ii) ignoring all the unique identifiers (i.e., PhoneNumber in Dataset-1, CustomerID in Dataset-2, and 3) and descriptive attributes, (iii) Replaced numerical missing values with a statistical method (i.e., interpolate); however, it cannot handle those missing values which placed at last row of the attribute. So, the last row missing value is replaced with the means value of the attribute, and (iv) Treated

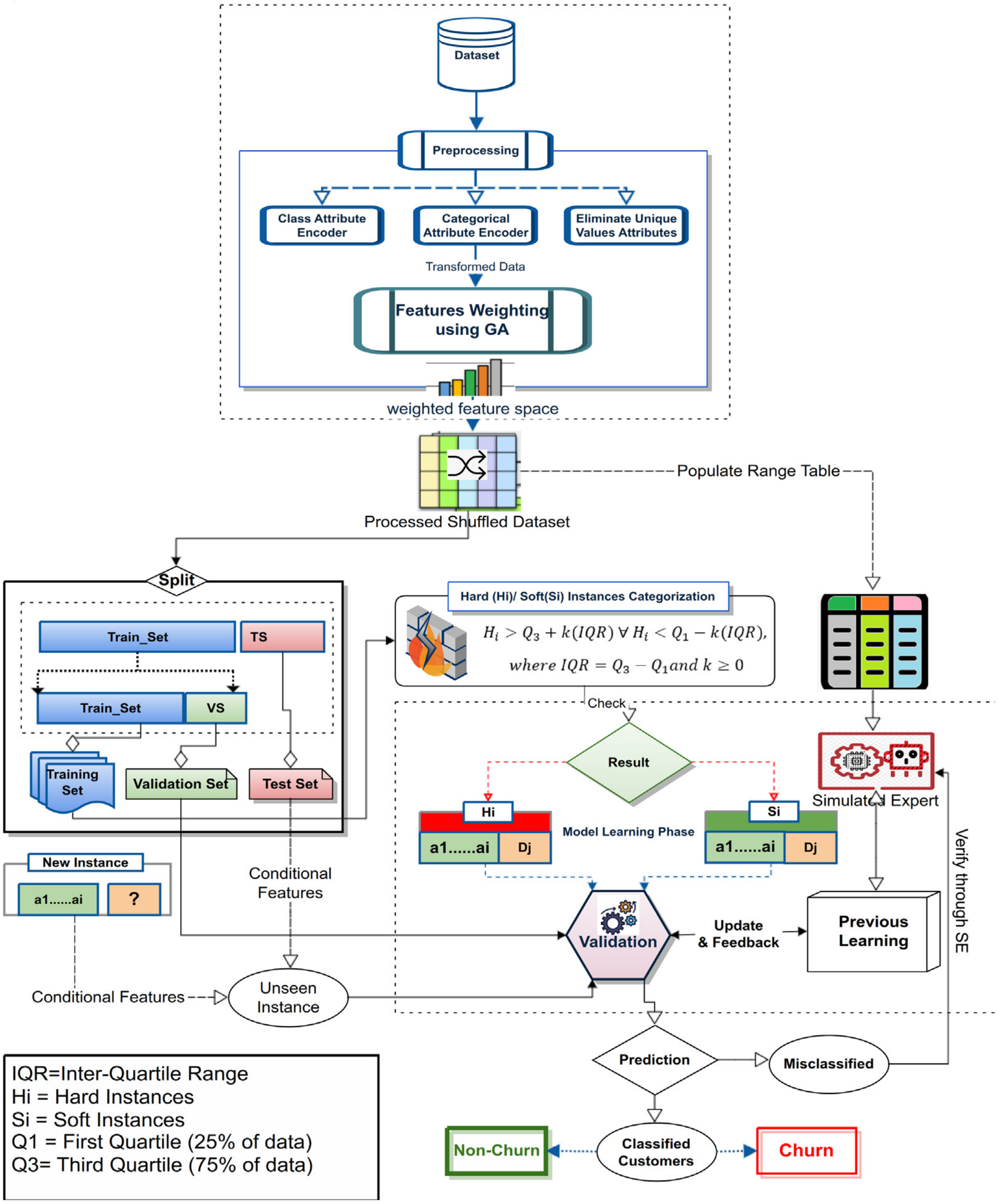


Fig. 1. Depicts the overall empirical setup of the proposed ACCP approach.

the missing values of the categorical attributes as a separate category.

3.4. Naïve Bayes as a baseline classification algorithm

Naïve Bayes (NB) is an “easy to implement” and “easy to interpret” ML method which was proposed by Reverend Thomas Bayes back in the 1760 [57]. NB is a powerful method for predictive modeling. It depends on two kinds of probabilities that can be directly calculated from the training set, such as (i) the prior probability of each class, and (ii) the conditional probability of each class label given each x value of the attribute [58]. After calculating these probabilities (to predict the class label of instance, estimate or compute the possible class label margin), the NB model can be used to predict the class label of unseen samples using the Bayes theorem [59]. For detail, follow the mathematical steps given in [60].

3.5. Genetic algorithm

In computer science, a Genetic Algorithm (GA) is a metaheuristic inspired by natural selection, a subclass of an Evolutionary Algorithm [61]. The GA algorithm provides high-quality solutions to optimize search problems and uses bio-inspired operators such as mutation, cross-over, and selection. In other words, we seek to achieve optimal or near-optimal solutions to complex problems by utilizing this algorithm which simulates natural selection. Steps for how the genetic algorithm works.

- **Initial Population** – Randomly initialize the population based on the available data.
- **Fitness function** – Find each chromosome’s fitness value (a chromosome is a set of parameters that define a proposed solution to the problem that the GA is trying to solve)
- **Selection** – The best-fitting chromosomes as parents to form a new population.

- **Cross-over** – Combine the parents' chromosomes to create a new population group.
- **Mutation** – Perform a mutation on a chromosome in the newly produced population set. The mutation increases the opportunity to select the population, and the following generation will use the population.

3.6. Feature weights

The subject datasets consist of comprehensive data of the customer behavior based on the call records, messages records, data usages, network services, and demographics records of the customers based on geographical information, resulting in multi-dimensionality of data. To efficiently address the curse of dimensionality, the researchers used different methods [62–66], known as feature extraction or feature selection process. Feature selection is a well-known approach that selects those features that contribute more to obtaining classification accuracy and reducing processing time; yet, decreasing the feature space increases the possibility of information loss. To avoid such loss of information through feature reduction, one solution is to involve the domain expert in assigning weights to the features, which is a subjective exercise and a costly process. Another approach is to improve the classifier's performance by assigning weights to features using ML methods (e.g., gradient descent, evolutionary algorithms) without the assistance of a domain expert or feature engineering [67]. The proposed study also attempts to solve the problem of data loss using GA for the self-assignment of weights to each attribute, which is important for accurate prediction of the class label (i.e., customer churn or non-churn). The advantage of using GA over gradient-based approaches (i.e., gradient descent, batch gradient descent, and mini-batch gradient descent) is that it can help achieve optimal solutions. This is due to the fact that GA is well-suited to multicriteria optimization and is self-adaptive in discovering optimal solutions to problems [68]. Moreover, due to its random walk, GA is insensitive to Pareto front shapes and tends to generate the optimal solution to tough situations [69], whereas gradient-based approaches are specialized in monocriteria optimization. Furthermore, the gradient-based approach needs to calculate the gradients for the entire dataset at each step of its execution. The efficiency of the approach drastically deteriorates particularly when the permutation are performed on large dataset with high dimensionality, leading to increase in the algorithm's time to convergence to produce an ideal output [70,71]. The algorithmic steps for assigning weights to features are described in algorithm 1 (FWAGA).

Algorithm 1 Features weight assignment using GA (FWAGA)

```
Input:   Set Dataset D,
         Set initial weight W,
         Set features set f,
Output: Return optimal weights of features
        which found within the given search space.
Function 1: Weight_Estimator (Dataset D)

        1. Extract features set
           $f := f_1, f_2, f_3 \ldots\ldots f_n$ from D
        2: Initially assigned the random weights
           $W := w_1, w_2, w_3 \ldots\ldots w_n$ where $w_i$
           is in range between 0 and 1.
        3: Select S samples in population p and
           chromosomes are the set of w for each
           $f_i$ in f-set. Such that
           $S_0 = w_{01}, w_{02}, w_{03} \ldots\ldots w_{0n}$
           $S_1 = w_{11}, w_{12}, w_{13} \ldots\ldots w_{1n}$
           $S_2 = w_{21}, w_{22}, w_{23} \ldots\ldots w_{2n}$
          ...................
           $S_n = w_{p1}, w_{p2}, w_{p3} \ldots\ldots w_{pn}$
        4: Repeat for each population k=1 ....... N
           Set $S_0 = w_{k1}f_1, w_{k2}f_2, w_{k3}f_3 \ldots\ldots\ldots\ldots w_{kn}f_n$
        5: Perform cross-over, and randomly
           select crossover sites. Then exchange the
           genes which produce an entirely new individual.
    $S_k = w_{k1}f_1, \ldots\ldots w_{kn/2}f_{(n/2)}, w_{\frac{(k+1)n}{2}+1}f_{\frac{n}{2}+1} \ldots\ldots w_{(k+1)n}f_n$
    $S_{(k+1)} = w_{(k+1)1}f_1, \ldots\ldots w_{(k+1)n/2}f_{n/2}, w_{\frac{kn}{2}+1)}f_{\frac{n}{2}+1} \ldots\ldots w_{(k)n}f_n$
        6: Perform mutation on random samples in
           offspring to keep diversity and avoid premature
           convergence.
           $w_{kn} = rand() * w_{k1}$
        7: Objective Function: Select the minimum
             least-square error
        8: Return ($f_i.w_i$)

End Function
```

In FWAGA, all the features f with random weights w have created N samples for each generation. The initial random weight w was assigned using the following steps:

- Select random weights 100 times, and then select the best set of initial random weights based on contribution in the prediction accuracy,
- Evaluate the performance of the baseline classifier on every selected set of samples ($e.g.,\ S_0, S_1, S_2, \ldots, S_{10}$) for each generation (e.g., No. of generations is set to 1000),
- The next step is to apply the cross-over and mutation operators of GA, which are core dependent operators of GA.

The mutation is used for multiplicity by allowing an offspring to modify so that it cannot be merely found by the inherited properties and by developing new genetic information. On the other hand, the crossover operator can generate a new solution from the population **S** samples from genetic detail (e.g., crossover S samples 10 + 10 and select best fit 10 samples). In every step of the iteration of generation, the value of the obtained **W** (e.g., weight) is assigned to the corresponding feature, and then the performance of the baseline classifier is evaluated with the contribution of feature weights.

3.7. Classification of hard and soft instances

It is observed from the literature that there is still more debate going on treating the outliers efficiently. It is usually considered an extreme (smaller or larger) or useless value because it can affect the performance of classification [72]. Therefore, the researchers do not consider such extreme cases during the model training phase and prefer to remove these values, samples, or even features from the dataset that can reduce the classification error. Schwertmanet et al. [73] recommended an approach for identifying anomalies in the dataset by a multiply inter-quartile range (IQR) to 1.5 (i.e., IQR *x* 1.5). The constant value 1.5 is used to discern the outlier. Hence, any target attribute value greater than 1.5 x IQR to the third quartile (Q3 of boxplot) is considered an anomaly or hard instance.

The presence of outliers in the customer behavior means there are some exceptional cases or hard instances which lies outside of the normal or soft pattern in the telecommunication industry [74]. Moreover, these hard instances can increase the complexity and decrease the performances of the prediction model.

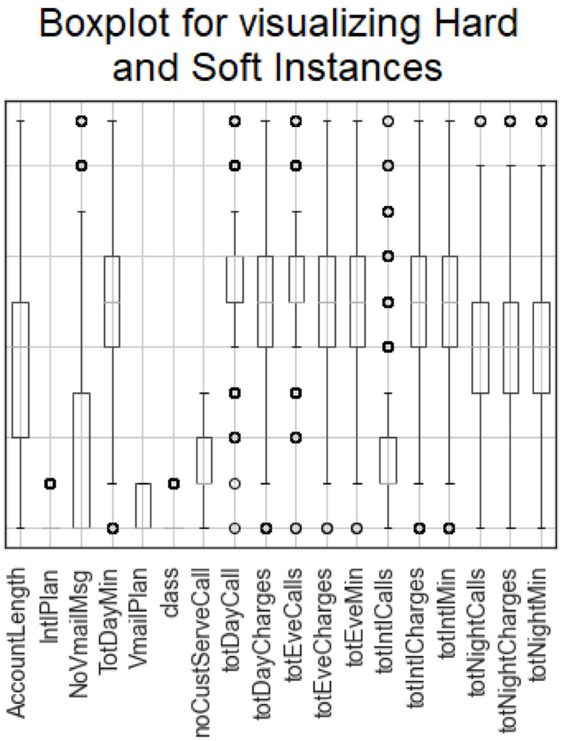


Fig. 2. Visualization of hard instances using Boxplot graph for Dataset-1.

Therefore, existing approaches [50,75] usually remove them from the training set. In addition, these hard instances (i.e., removed instances) also contain customer behavioral data which may not be useful for time being but later on such data may be required for model training. Therefore, it will be necessary to undergo retraining process of the model, which is both costly in term of required resources and time consuming in a dynamic environment of the telecommunication industry. In order to avoid the costly process of model retraining due to data lost; the proposed study focuses on handling the hard instances instead of removing them from the classification process. The algorithm for classification of hard instances is described as follows in Algorithm 2 (CHI).

Algorithm 2 Classification of Hard Instances (CHI)

```
Input: Extract feature set F from the training data sample
       $F = \{f_1, f_2, f_3 \ldots\ldots f_n\}$
       Set c := Class attribute, Q := Quantile
Output: Return Hard instances (Hi)
    1: Repeat $f_i$ to $f_n$ and $f_i \neq c$ where i=1,2,3....n
    2: Estimate inter-quartile-range
        $iqr = data(f_i).Q(0.75) - data(f_i).Q(0.25)$
    3: Estimate the lower (L) and upper (U) boundaries
        $L = data(f_i).Q(0.25) + (iqr \times 1.5)$
        $U = data(f_i).Q(0.75) - (iqr \times 1.5)$
    4: Classify Hard instances
        $instance = Not(data(f_i) < L)andNot(data(f_i) > U)$
        $Hi[] = append(instance)$
    5: Until (i=n)
    6: Return Hi
```

The CHI generates an array of Hard instance (H_i), and soft instance (S_i) which are visualized using boxplot graph, Fig. 2 depicting boxplots for H_i and S_i.

Table 3
Range table for dataset-1.

Features	Min_c	Max_c	Min_{nc}	Max_{nc}
Account_Length	1	238	1	243
Area_code	1	3	1	3
Intl_Plan	1	2	1	2
Vmail_Plan	1	2	1	2
No_Vmail_msg	0	51	0	52
Tot_day_Min	12.5	351.5	0	326.1
tot_day_call	42	158	0	163
tot_day_charges	2.13	59.76	0	55.44
tot_eve_min	75.3	349.4	0	361.8
tot_eve_calls	47	168	0	170
tot_eve_charges	6.4	29.7	0	30.75
tot_night_min	71.1	381.6	0	395
tot_night_calls	42	158	0	175
tot_night_charges	3.2	17.17	0	17.77
tot_intl_min	0	20	0	19.7
tot_intl_min	0	20	0	19
tot_intl_charges	0	5.4	0	5.32
tot_intl_charges	0	3	0	3

3.8. Construction of the range-table for simulated expert

Simulated expert (SE) is used to further classify the cases that the trained classifier's validator could not satisfy. A SE is designed to enhance the previous learning experience of the classifier by mapping the misclassified instances. The range table for simulated expert (RTSE) component of the proposed study contribute to formalizing the behavior of such instances which are not properly classified (e.g., Hard instances). The features of this type of instance are passed to RTSE component for predicting the class label based on the range table values. The range table structure is as follows (Attribute name, Minimum value of each feature against the class-label C and NC, Maximum value of each feature against the class-label C and NC). Once the required range table is maintained with the required values, then applied the SE. The following procedure will be followed as given in Algorithm 3 (RTSE). Table 3 described the range tables for Dataset-1, and similarly range tables are also generated for the rest of datasets.

3.9. Model evaluation process

The proposed study followed the recommended method train-validate-test cross-validation method of Stuart Russel and Peter Norvig [76], Max and Kjell [77], for model evaluation. For this purpose, initially, the subject datasets were divided using 80:20 ratio for training and test sets, respectively. The training set was further split into 60:20 ratio for training and validation sets, respectively. The model will learn from the training set, and simultaneously, the model evaluation is performed on the validation set. Additionally, the above process of the train-validation-test is repeated for Ten times and obtained an unbiased evaluation of the model on the training set. The proposed model is systematically presented step by step in Algorithm 4 (ACCP).

3.10. Evaluation measures for ML models

In order to evaluate the strengths and weaknesses of the classification model, the following state-of-the-art evaluation measures are used [78].

$$Accuracy = \frac{No.\ of\ correct\ prediction}{total\ no.\ of\ predicted\ samples} \tag{2}$$

In the case of CCP, the accuracy measure can be expressed as:

$$Accuracy = \frac{No.\ of\ correct\ churn\ or\ non-churn\ prediction}{total\ no.\ of\ customers\ prediction} \tag{3}$$

Algorithm 3 Range Table and Simulated Expert (RTSE)

```
Input: List of features F={$f_1, f_2, f_3, \ldots\ldots\ldots f_n$}
        List of instances S= {$s_1, s_2, s_3, \ldots\ldots\ldots s_n$}
        // Construction of Range Table
   Step 1: $DF_c$ will hold the feature value of churn sample
        Set $DF_c$ := values of $f_i$ for each $s_i$
        // $DF_{nc}$ will hold the feature value of non-churn sample
        Set $DF_{nc}$ := values of $f_i$ for each $s_i$
   Step 2: $Min_c$ the minimum value of churn sample features
        Get $Min_c$ := minimum value of $DF_c$ , $Max_c$:= maximum value of $DF_c$
        //$Min_{nc}$ will hold the minimum value of non-churn sample features
        Get $Min_{nc}$ := minimum value of $DF_{nc}$ , $Max_{nc}$ := maximum value of $DF_{nc}$
    Step 3: Create table (features names, $Min_c, Max_c, Min_{nc}, Max_{nc}$)
          // Values extraction from the Range Table i.e., RT
Function 2 $RT(f, p)$
          // f and p parameters will receive the feature and range table column sequence number, respectively.
    Step 4:    Match f in range table fields
    Step 5:    If $p == C_1$ then $val$ := $Min_c$   else   //$C_1$ represents the second column of the range table
         If $p == C_2$ then $val$ := $Max_c$   else   //$C_2$ represents the third column of the range table
         If $p == C_3$ then $val$ := $Min_{nc}$   else   //$C_3$ represents the fourth column of the range table
         If $p == C_4$ then $val$ := $Max_{nc}$          //$C_4$ represents the fifth column of the range table
    Step 6: Return val
End Function 2
          // simulated expert (SE)
Function 3 SE(Missed Classified Instances)
    Step 7: Set $Counter_C$:=0
         Set $Counter_{NC}$ := 0
         Set Status : =0
    Step 8: Repeat for i=0 to f
    Step 9: Get $valMin_C = callRT(f_i, C_1)$
    Step 10: Get $valMax_C = callRT(f_i, C_2)$
    Step 11: Get $valMin_{NC} = callRT(f_i, C_3)$
    Step 12: Get $valMax_{NC} = callRT(f_i, C_4)$
    Step 13: if  $f_i.val \geqslant valMin_C$  and  $f_i.val \leqslant valMax_C$ then $Counter_C := Counter_C + 1$
    Step 14: if $f_i.val \geqslant valMin_{NC} and f_i.val \leqslant valMax_{(NC)}$ then  $Counter_{NC} := Counter_{NC} + 1$
    Step 15: if $Counter_C > Counter_{NC}$ then status = C else status := NC
    Step 16: End for
    Step 17: Return status
End Function 3
```

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{4}$$

In the case of CCP, this evaluation measure is the fraction of relevant customers among the retrieved customers, and it can be expressed mathematically as:

$$Precision = \frac{No.\ of\ predicted\ churn}{Relevant\ customers\ among\ the\ predicted\ customers} \tag{5}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{6}$$

In the case of CCP, the recall metric can be expressed as:

$$Recall = \frac{No.\ of\ true\ churn}{total\ no\ of\ actual\ churn} \tag{7}$$

F1-Score shows, that the best-obtained result is reached at one and the worst result at zero and is mathematically expressed as [58]:

$$F - Measure = 2\ \times\ \frac{Precision\ x\ Recall}{Precision + Recall} = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{8}$$

Matthews Correlation Coefficient (MCC): it is a recognized suitable metric for the number of instances between churn and non-churn classes and is a relatively balanced evaluation metric for the CCP model evaluation. MCC is mathematically expressed as [79]:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{9}$$

The value range of MCC is between (−1 to 1), where 1 is the perfect prediction, 0 can be regarded as a random prediction, and −1 means that the prediction result is completely inconsistent with the actual situation.

4. Results and discussion

This section presents a detailed analysis and discussion of the results obtained at various stages of the proposed study. Initially, we summarize the analysis of the experiment conducted using FWAGA for obtaining optimum weights assignments to each feature of subject datasets. In order to find the optimum weight FWAGA is initialized with random weights to automatically find the optimum weight and then assign them to each feature vector of population of first generation. These random weights are then optimized through a series of iterations (i.e., around 1000) using

Algorithm 4 Adaptive Customer Churn Prediction (ACCP)

```
Input: Dataset D:= $\{s_1, s_2, \dots\dots\dots s_n\}$
Output: Classify whether the unseen customer is churn or non-churn
    1. Performed encoding of categorical attributes
        $A := \{a_1, a_2, \dots\dots\dots a_n\}$
    2. Assign weights to the attributes A by calling Function 1 of FWAGA
        $a_i.w_i := Weight\_Estimator\ (D)$
    3. Construct the range table for each attribute by call function 2 of RTSE
        $a_i\{Min_c, Max_c, Min_{nc}, Max_{nc}\}$
    4. Repeat for t=0 to 10 resampling set do
    5. Shuffle all the samples (0,1......n-1) in D
        repeat for i=0 to n-1 samples and decrease 1
            $l := random\ integer\ from\ 0 \leq l \leq i$
            $shuffle\ D[S_i]\ and\ D[S_l]$
            End for
    6. Perform train-validate-test cross-validation
            TrainSet := $80\%\ of\ S_{(n-20\%)} \in D$ and
            $TSet := S_{(n-TrainSet)}$
            $VSet := S_{(TrainSet-60\%)}$
    7. Identify the hard (Hi) and Soft instances (Si) from Training set by using CHI
    8. Fit the $Model_{si}$ through soft instances received from step 7 by calling function GNB(Si)
    9. Fit the $Model_{Hi}$ through hard instances received from step 7 by calling function GNB(Hi)
    10. Update the previous learning experience in knowledge base by step 8 and 9
    11. Predict the class label of unseen instances
    12. End for
    13. Calculate the average performance across all train-validate-test cross-validation prediction
    14. if predicted ClassLabel := Correct then
            pass classified sample s to Function 4 (GNB)
            Otherwise:
            Get Label := Pass misclassified sample s to Function 3 of RTSE
            Update := $Step$ 10
Function 4 GNB(s) {
    15. Calculate the mean, standard deviation stdev and count for each attribute A in D
    16. Calculate the class probabilities
    17. Calculate the Gaussian probability distribution function for each $s_i i = 1, 2, 3 \dots .n - 1$

        $gnb_{val} = \frac{1}{(sqrt(2 * pi) * stddev)} * exp(-\frac{(s_i - mean)}{2 * stdevs})$

    18. Return $S_{i\ gnbcal}$ for class in (c and nc)
END Function
```

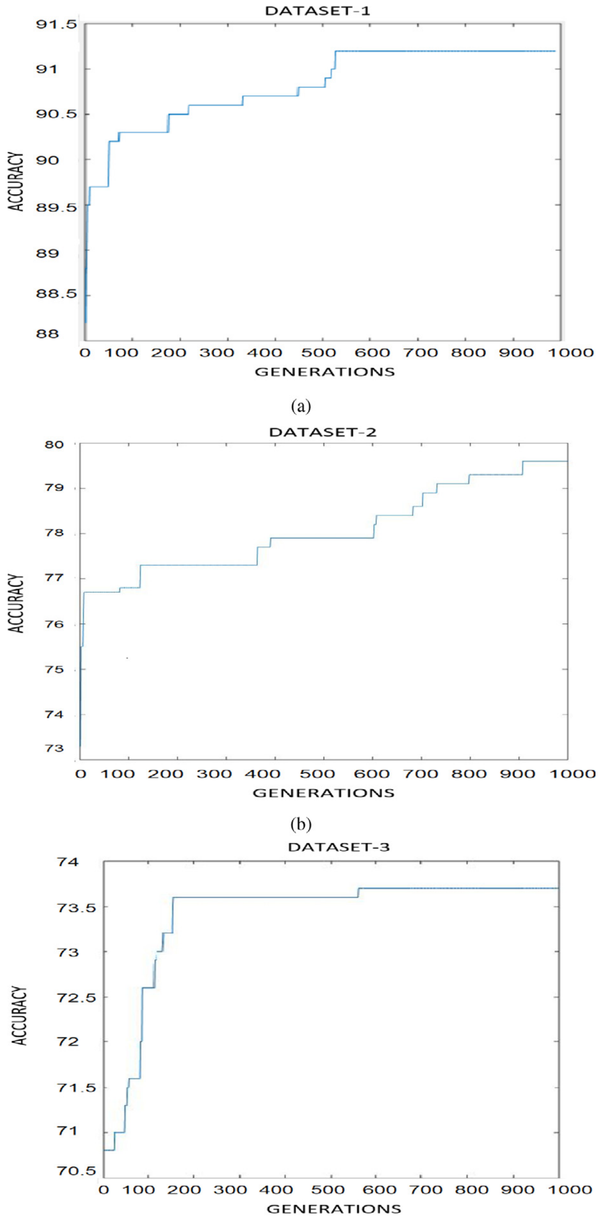
GA algorithm by generating 10 new members through crossover and mutation. Out of these twenty, best ten are selected to form the population of the next generation. Selection of the best weight assigned to different features is done by averaging of ten iterations for realistic result and to avoid any good or bad number obtained by chance. Initially, the FWAGA obtained 88% accuracy upon which the crossover function was carried out on population of samples and it regenerated the surviving features again as explained above. The process of population regeneration revealed that 1 to 530 generations of FWAGA produced varying results, as illustrated in Fig. 3. Similar process is carried out for the rest of the subject datasets. Fig. 3(a), (b) and (c) visualized the overall process of the FWAGA on dataset-1, 2 and 3, respectively.

Further, it is also investigated that when the FWAGA process reached to 531^{st} generation then the accuracy (i.e., 91.3%) remain the same till 1000^{th} generation. At this point, we have recorded weights for the features that can contribute more to the performance of the proposed approach, known as spot-point in dataset-1, as given in Table 4.

The main achievements of the proposed FWAGA, including contributions, may be summarized as follows: (i) Fig. 3 have shown the process of features weighting schematically, and its contribution to the overall performance of the classification, (ii) unhide the truth that which feature improve or reduce the accuracy of the model when the weight of the feature increase or decrease, (iii) achieved the optimum point (i.e., spot-point) of each feature were assigned weight value of the target features improved the model's performance. Finally, it is also concluded that the spot-point weights of all features are the optimum weight, proving the consistency in the improved model's performance. FWAGA self-learned and adopted optimum weights for each feature without losing any data from the original subject datasets just increase or decrease the weight value of feature and evaluated the performance of the model. In order to prove the best feature's weight identification i.e., the spot point, FWAGA is tested for different weights in the neighborhood of the peak/optimal values. The following two experimental results prove that the weights obtained in this way are best and cannot be further improved.

- Single variable experiment,
- Two best simultaneously selected variable experiment

Fig. 4(a) shows the accuracy around the neighborhood of the selected value (Spot Point) denoted through the red dot on the graph line obtained through a single variable. The spot point



(a)

(b)

(c)

Fig. 3. The overall process of FWAGA on Dataset-1, 2, and 3. The x-axis and y-axis represent the number of generations and obtained accuracy, respectively.

Table 4
Effects of the feature weights on baseline classifier's performance using dataset-1.

List of features	Optimum weights (Spot-point)
Account Length	0.0676
Area Code	0.6346
International_Plan	0.113
Voice Mail Plan	0.8494
No. Vmail Messages	0.9843
total_day_minutes	0.7551
total_day_calls	0.292
total_day_charge	0.1889
total_eve_minutes	0.2096
total_eve_calls	0.2087
total_eve_charges	0.4615
total_night_minutes	0.456
total_night_calls	0.903
total_night_charges	0.7431
total_intl_minutes	0.3145
total_intl_calls	0.5949
total_intl_charges	0.6685
number_cust_service_calls	0.8193

Table 5
Performance evaluation of the baseline classifier on default setting on all datasets.

	Class	Precision	Recall	F1-Score	MCC	Accuracy
Dataset-1	C	0.45	0.51	**0.48**	0.39	**0.85**
	NC	**0.92**	0.91	**0.92**		
	Overall	0.69	**0.71**	0.69		
Dataset-2	C	**0.74**	0.54	0.62	**0.47**	0.77
	NC	0.77	**0.89**	0.83		
	Overall	**0.76**	0.72	0.73		
Dataset-3	C	**0.54**	0.35	0.42	0.11	0.57
	NC	0.58	**0.75**	0.65		
	Overall	**0.56**	0.55	0.54		

value cannot be perfected further by adjusting the feature weight value up or down as it will compromise the performance of the classifier. If done, the same dot would appear in the other portion (less or greater than the spot point) of the graph showing its imperfection. Experimental results show that no other value of these features gives better accuracy than the selected value obtained from the GA algorithm. In the second experiment of validating the best feature selection using the two best simultaneously selected variable experiments, we choose two significant features based on their optimum weight, which contributes significantly to the improvement of the classifier. Fig. 4(b) shows the accuracy around the neighborhood of the selected value (Spot Point) denoted through the red dot on the graph line obtained through the two most significant variables. The spot point value cannot be perfected further by adjusting the feature weight value up or down as it will compromise the performance of the classifier. Again, experimental results show that the best accuracy is achieved at the values obtained by the GA algorithm. Fig. 4. (a) Single variable combined effects of all variables, (b) represent two best simultaneously selected variable experiments.

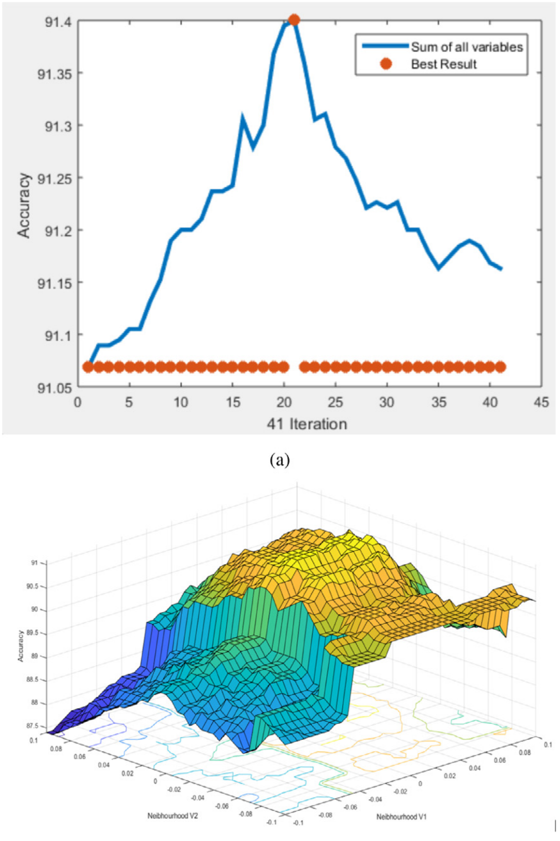
4.1. Comparison between proposed ACCP approach and naive Bayes (with default setting)

To show the performance of ACCP, we have compared the performance of the ACCP approach and the baseline line classifiers (i.e., Naïve Bayes) with the default setting. Tables 5 and 6 represents the obtained performance in term of the precision, recall, accuracy, f1-score, and MCC during the evaluation of the proposed model (i.e., ACCP) on subject datasets.

Table 5 represents the performance of the baseline classifier on default setting on preprocessed subject datasets (i.e., Dataset-1, Dataset-2, and Dataset-3).

In another experiment, the performance of the proposed approach on the subject datasets (i.e., Dataset 1, 2, and 3) was evaluated using the target evaluation measures (i.e., accuracy, precision, recall, F1-score, and MCC). Table 6 reflects the overall performance of the proposed ACCP techniques on the subject's datasets.

It can be observed from Tables 5 and 6 that the proposed ACCP approach achieved the average precision values of 0.97, 0.973, and 0.98 during the model evaluation on subject datasets, respectively. Similarly, the baseline classifier with default setting achieved the average precision values of 0.69, 0.76, and 0.56 on subject dataset-1, 2 and 3, respectively. This finding reveals that overall performance of the proposed approach is better by 30% in terms of average precision as compared to the baseline classifier with default setting. On the other hand, it is also investigated from Table 5 that the proposed ACCP approach achieved the



(a)

(b)

Fig. 4. (a) Single variable combined effects of all variables, (b) represents two best simultaneously selected variable experiment.

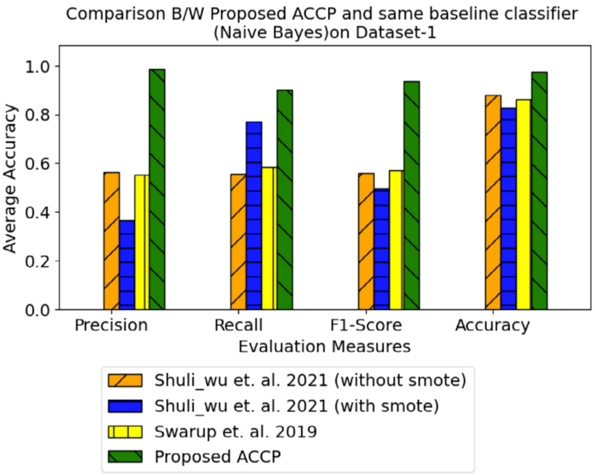


Fig. 5. Comparison between the performance of the proposed approach and state-of-the-art studies which used the same baseliner classifier and Dataset-1.

Table 6
Performance evaluation of the proposed technique on subject datasets.

	Class	Precision	Recall	F1-Score	MCC	Accuracy
Dataset-1	C	0.94	**0.99**	0.97	0.80	0.955
	NC	**0.99**	0.68	0.81		
	Overall	**0.97**	0.84	0.89		
Dataset-2	C	0.95	**0.99**	0.97	0.92	0.975
	NC	**0.99**	0.89	0.94		
	Overall	**0.97**	0.94	0.96		
Dataset-3	C	0.95	**0.99**	0.98	**0.96**	**0.985**
	NC	**0.99**	0.94	0.97		
	Overall	**0.98**	0.97	0.97		

average recall values of 0.84, 0.94, and 0.97 during the model evaluation on subject datasets, respectively. Similarly, the baseline classifier with default setting achieved the average recall values of 0.71, 0.72, and 0.55 on subject datasets, respectively. This finding reveals that overall performance of the proposed approach is better by 25.6% in terms of average recall as compared to the baseline classifier with default setting. Further, we have analyzed that the proposed ACCP approach achieved the average F1-Score values of 0.89 0.96, and 0.97 during the model evaluation on subject datasets, respectively. Similarly, the baseline classifier with default setting achieved the average F1-Score values of 0.69, 0.73, and 0.54 on subject datasets, respectively. This finding reveals that overall performance of the proposed approach is better by 40.6% in terms of average F1-Score as compared to the baseline classifier with default setting.

Moreover, the proposed ACCP approach achieved the average MCC values of 0.80, 0.92, and 0.96 during the model evaluation on subject datasets, respectively. Similarly, the baseline classifier with default setting achieved the average MCC values of 0.39, 0.47, and 0.11 on subject datasets, respectively. This finding reveals that overall performance of the proposed approach is better by 36.6% in terms of MCC as compared to the baseline classifier with default setting. Finally, it is also analyzed from Table 6 that the proposed approach obtained 0.95, 0.97, and 0.98 accuracies on Dataset-1, 2, and 3, respectively. The proposed ACCP approach achieved the average accuracies values of 0.95, 0.97, and 0.98 during the model evaluation on subject datasets, respectively. Similarly, the baseline classifier with default setting achieved the average accuracy values of 0.85, 0.77, and 0.57 on subject datasets, respectively. Moreover, the overall average accuracy of the proposed approach i.e., 0.97, and the baseline classifier with a default setting is 0.73. Therefore, the approach is that about 23% outperforms the baseline classifier with a default setting in terms of the average accuracy. Hence, the overall performance in terms of Precision, Recall, F1-score, MCC, and Accuracy of the proposed ACCP is improved than the performance of the baseline classifier with default settings. Furthermore, the baseline classifier with a default setting is always required to train the model and involve the human expert or domain expert, which is costly processes in terms of time and cost.

4.2. Comparison between proposed ACCP and state-of-the-art methods using same baseline classifiers and subject datasets

In this section, an extensive comparison is performed between the proposed ACCP approach and other state-of-the-art studies using the same baseline classifier (i.e., Naïve Bayes) and same subject datasets (i.e., Dataset-1, 2, and 3). For this purpose, the results of the proposed ACCP approach are analyzed and compared with some most recent studies [1,56,80,81]. It can be observed from the Figs. 5, 6, and 7 that the proposed approach outperformed all other studies in terms of average precision, average recall, average f1-score, and average accuracy, which are 0.971 (precision), 0.842 (recall), 0.893 (f1-score), and 0.955 (accuracy).

4.3. Comparison between proposed ACCP and state-of-the-art methods using different baseline classifiers on the same subject datasets

This section compares the proposed ACCP approach and state-of-the-art techniques using the different baseline classifiers (i.e., Logistic Regression, Decision Tree, Random Forest, AdaBoost, Layer

Table 7
Results of proposed ACCP approach and other state-of-the-art ML techniques on dataset-1.

Source studies	Techniques		Precision	Recall	F1-Score	Accuracy
Wu et al. [1]	Logistic Regression	Without smote	0.638	0.193	0.292	0.874
		With smote	0.344	0.753	0.472	0.771
	Decision Tree	Without smote	0.886	0.685	0.770	0.945
		With smote	0.706	0.790	0.742	0.925
	Random Forest	Without smote	0.917	0.719	0.804	**0.953**
		With smote	0.746	0.807	0.772	0.936
	AdaBoost	Without smote	0.710	0.218	0.331	0.882
		With smote	0.495	0.665	0.556	0.862
	Layer Perception	Without smote	0.898	0.731	0.805	0.952
		With smote	0.610	0.799	0.691	0.903
Hemlata Jain et al. [80]	Random Forest		0.952	**0.840**	**0.880**	0.945
	Linear Regression		0.729	0.584	0.605	0.858
	Logit Boost		0.793	0.691	0.725	0.883
	CNN		0.789	0.577	0.599	0.873
	SVM		0.775	0.546	0.545	0.846
	XGBoost		0.782	0.551	0.558	0.860
	PCALB		0.650	0.530	0.527	0.850
Proposed approach	ACCP		**0.971**	**0.842**	**0.893**	**0.960**

Table 8
Results of proposed ACCP approach and other state-of-the-art ML techniques on dataset-2.

Source studies	Techniques		Precision	Recall	F1-Score	Accuracy
Wu et al. [1]	Logistic Regression	Without smote	0.651	0.545	0.595	0.801
		With smote	0.517	0.787	0.624	0.748
	Decision Tree	Without smote	0.615	0.507	0.552	0.783
		With smote	0.549	0.720	0.622	0.767
	Random Forest	Without smote	0.661	0.475	0.552	0.795
		With smote	0.551	0.732	0.628	0.769
	AdaBoost	Without smote	0.653	0.532	0.586	0.800
		With smote	0.554	0.733	0.631	0.771
	Layer Perception	Without smote	0.654	0.534	0.587	0.801
		With smote	0.528	0.763	0.624	0.756
Samah et al. [56]	Deep-BP-ANN		0.847	**0.930**	0.886	0.881
	Linear Regression		0.745	0.789	0.766	0.759
	XG-Boost		0.818	0.920	0.866	0.857
	KNN		0.731	0.847	0.785	0.768
Proposed approach	ACCP		**0.971**	**0.939**	**0.961**	**0.975**

Table 9
Results of proposed ACCP approach and other state-of-the-art ML techniques on dataset-3.

Source studies	Techniques		Precision	Recall	F1-Score	Accuracy
Wu et al. [1]	Logistic Regression	Without smote	0.467	0.201	0.375	0.710
		With smote	0.351	0.536	0.410	0.576
	Decision Tree	Without smote	0.501	0.559	0.875	0.683
		With smote	0.346	0.408	0.359	0.593
	Random Forest	Without smote	0.489	0.496	0.801	0.688
		With smote	0.346	0.270	0.293	0.630
	AdaBoost	Without smote	0.574	0.336	0.569	0.700
		With smote	0.365	0.493	0.405	0.586
	Layer Perception	Without smote	0.437	0.479	0.780	0.695
		With smote	0.346	0.611	0.428	0.534
Samah et al. [56]	Deep-BP-ANN		0.745	0.893	0.812	0.793
	Linear Regression		0.593	0.583	0.584	0.591
	XG-Boost		0.707	0.750	0.728	0.720
	KNN		0.619	0.720	0.666	0.638
Proposed approach	ACCP		**0.981**	**0.972**	**0.971**	**0.985**

Perception, Logit Boost, CNN, SVM, XGBoost, PCALB, Deep-BP-ANN, and KNN) but all these techniques were applied on the same subject datasets (i.e., Dataset-1, Dataset-2, and Dataset-3). For this purpose, the analysis and investigation expanded to compare the performance of the proposed ACCP approach with some of the most recent studies such as [1,56,80], reflected in Tables 7, 8, and 9 with same subject dataset (i.e., Dataset-1, 2, and 3, respectively).

It is observed from Tables 7–9 that the proposed approach applied on the Dataset-1, 2, and 3 so it has outperformed all other classifiers in terms of precision, recall and F1-score, which

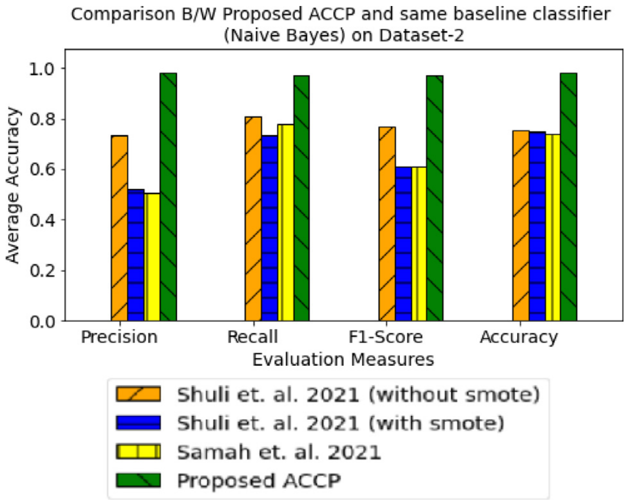


Fig. 6. Comparison between the performance of the proposed approach and state-of-the-art studies which used same baseliner classifier and Dataset-2.

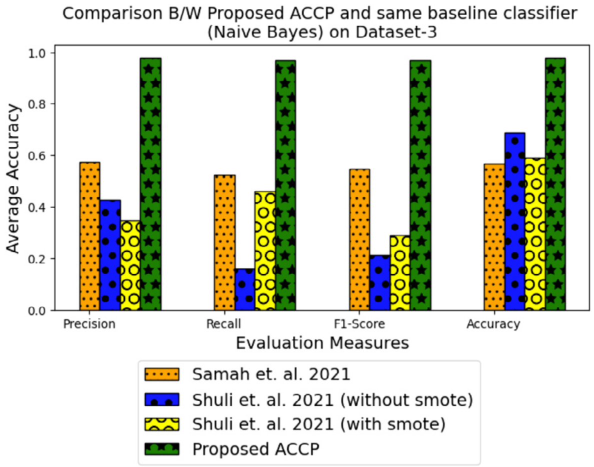


Fig. 7. Comparison between the proposed approach's performance and state-of-the-art studies that used the same baseline classifier and Dataset-3.

are 0.971, 0.842, 0.893, and 0.951 (for Dataset-1), 0.971, 0.939, 0.961, and 0.975 (for Dataset-2), and 0.981, 0.972, 0.971, and 0.985 (for Dataset-3). The next best technique which performs comparatively better than other techniques was the Random Forest method in term of accuracy.

It is found that the overall performance of the proposed model in Comparison to the same baseline classifier and other state-of-the-art machine learning techniques is higher in terms of F1 score, precision, and recall. Further, it is also investigated that the proposed approach has shown better results on large datasets because Tables 5, 6, 7, 8, and 9 clearly shows that the proposed approach achieved the maximum performance on large subject data (e.g., Dataset-3 has 51 047 samples) as compared to the rest of the datasets. Similarly, the medium size of the Dataset-2 (i.e., contains 7043 samples) also performs better than the small size of Dataset-1 (i.e., holds 3333 samples).

5. Conclusion

Since customers in the telecommunication industry always tend to be saturated, it is more beneficial for decision makers to predict the customers who are about to churn and retain them for better relationship management. Although extensive work is carried out to address problem of CCP in the telecommunication industry using various ML and data mining models; however, there is no standard model which continuously learn the customer behavior and accurately predict the customer churn. Keeping all such limitations into consideration; therefore, this study conducted and focused on adaptive learning approach for CCP (see ACCP algorithm). All the experiments performed on Three well-known publicly available datasets including (i) BigML Telecom dataset comprising 3333 customers data, (ii) IBM Telco dataset comprising 7043 customers records, and (iii) Cell2Cell dataset which has 51 047 telecom customers' data. The performance of the proposed study was evaluated in term of average precision of 0.97, 0.972, 0.981, a recall rate that stands at 0.84, 0.94, 0.97, and f1-score of 0.89, 0.96, 0.97, MCC values 0.89, 0.96, 0.97, accuracy 0.95, 0.97, 0.985 on abovementioned datasets, respectively. The proposed study outperformed when compared to the baseline classifier– Naïve Bayes with default setting (results are reflects in Tables 5 and 6), and other state-of-the-art ML techniques i.e., Deep-BP-ANN, CNN, NN, Linear regression, XG-Boost, KNN, Logit Boost, SVM, and PCALB (results are reflected in Tables 7–9 and Figs. 5–7). Overall, this study has provided a novel adaptive learning approach for CCP in the telecommunication industry. The proposed work could help apply adaptive learning, which can improve the performance of the baseline classifier compared to state-of-the-art techniques, and make an informed business and technical decision.

Future work could further enhance this research by including an adaptive learning approach for CCP in BigData environment. Another aspect of the proposed approach would be to analyze its robustness when applied to various competitive industries such as online gaming, human resources, health care, social networks, financial sector, banking sector, subscriber services, e-commerce, insurance services, and online question and answer (Q&A) forums.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Used publicly available datasets.

References

[1] S. Wu, W.-C. Yau, T.-S. Ong, S.-C. Chong, Integrated churn prediction and customer segmentation framework for telco business, IEEE Access 9 (2021) 62118–62136.

[2] M. Zhao, Q. Zeng, M. Chang, Q. Tong, J. Su, A prediction model of customer churn considering customer value: An empirical research of telecom industry in China, Discrete Dyn. Nat. Soc. 2021 (2021) 1–12, http://dx.doi.org/10.1155/2021/7160527.

[3] M.-K. Kim, M.-C. Park, D.-H. Jeong, The effects of customer satisfaction and switching barrier on customer loyalty in Korean mobile telecommunication services, Telecommun. Policy 28 (2) (2004) 145–159.

[4] N. Mustafa, L.S. Ling, S.F.A. Razak, Customer churn prediction for telecommunication industry: A Malaysian case study, Natl. Libr. Med. Open Res. J. F1000Res. 10 (2021) 1274, http://dx.doi.org/10.12688/f1000research.73597.1.

[5] M. Zhao, Q. Zeng, M. Chang, Q. Tong, J. Su, A prediction model of customer churn considering customer value: an empirical research of telecom industry in China, Discrete Dyn. Nat. Soc. 2021 (7160527) (2021) 12, http://dx.doi.org/10.1155/2021/7160527.

[6] E. Zdravevski, P. Lameski, C. Apanowicz, D. Ślęzak, From big data to business analytics: The case study of churn prediction, Appl. Soft Comput. 90 (2020) 106164.

[7] T. Xu, Y. Ma, K. Kim, Telecom churn prediction system based on ensemble learning using feature grouping, Appl. Sci. 11 (11) (2021) 4742.

[8] How costly is customer churn in the telecom industry, the European business review, 2020, [Online] Available: https://www.europeanbusinessreview.com/how-costly-is-customer-churn-in-the-telecom-industry/, accessed February 25, 2021.

[9] A. Amin, B. Shah, A. Abbas, S. Anwar, O. Alfandi, F. Moreira, Features weight estimation using a genetic algorithm for customer churn prediction in the telecom sector, in: World Conference on Information Systems and Technologies, Springer, 2019, pp. 483–491.

[10] J. Handajani, D. Kusumajati, H. Fathiyah, H. Susilowati, R.T. Tandelilin, Peer review: 1 approved, Universitas 10 (1274) (2021) 1–20, http://dx.doi.org/10.12688/f1000research.73597.1.

[11] A. Chefrour, Incremental supervised learning: algorithms and applications in pattern recognition, Evol. Intell. 12 (2) (2019) 97–112.

[12] J. He, R. Mao, Z. Shao, F. Zhu, Incremental learning in online scenario, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13926–13935.

[13] A. Mishra, U.S. Reddy, A novel approach for churn prediction using deep learning, in: 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), IEEE, 2017, pp. 1–4.

[14] Á. Periáñez, A. Saas, A. Guitart, C. Magne, Churn prediction in mobile social games: Towards a complete assessment using survival ensembles, in: 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), IEEE, 2016, pp. 564–573.

[15] I. Ulku, F.U. Yuksektepe, O. Yilmaz, M.U. Aktas, N. Akbalik, Churn prediction in a pay-TV company via data classification, Int. J. Artif. Intell. Mach. Learn. (IJAIML) 11 (1) (2021) 39–53.

[16] A. Amin, S. Anwar, A. Adnan, M. Nawaz, N. Howard, J. Qadir, A. Hawalah, A. Hussain, Comparing oversampling techniques to handle the class imbalance problem: A customer churn prediction case study, IEEE Access 4 (2016) 7940–7957.

[17] R. Vadakattu, B. Panda, S. Narayan, H. Godhia, Enterprise subscription churn prediction, in: 2015 IEEE International Conference on Big Data (Big Data), IEEE, 2015, pp. 1317–1321.

[18] T.W. Cenggoro, R.A. Wirastari, E. Rudianto, M.I. Mohadi, D. Ratj, B. Pardamean, Deep learning as a vector embedding model for customer churn, Procedia Comput. Sci. 179 (2021) 624–631.

[19] T. Vafeiadis, K.I. Diamantaras, G. Sarigiannidis, K.C. Chatzisavvas, A comparison of machine learning techniques for customer churn prediction, Simul. Model. Pract. Theory 55 (2015) 1–9.

[20] A. Idris, K. Asifullah, Ensemble based efficent churn prediction model for telecom, in: 12th International Conference on Frontiers of Information Technology (FIT), 2014, pp. 238–244, http://dx.doi.org/10.1109/FIT.2014.52.

[21] S. Nisha, Churn prediction in telecommunication industry using decision tree, Streamed Info-Ocean 1 (2016) 1–9.

[22] A. Ahmad, A. Jafar, K. Aljoumaa, Customer churn prediction in telecom using machine learning in big data platform, J. Big Data 6 (1) (2019) 1–24.

[23] K.W. De Bock, D. Van den Poel, Reconciling performance and interpretability in customer churn prediction using ensemble learning based on generalized additive models, Expert Syst. Appl. 39 (8) (2012) 6816–6826.

[24] A. Sharma, P.P. Kumar, A neural network based approach for predicting customer churn in cellular network services, Int. J. Comput. Appl. 27 (2011) 26–31.

[25] C.-F. Tsai, Y.-H. Lu, Customer churn prediction by hybrid neural networks, Expert Syst. Appl. 36 (2009) 12547–12553, http://dx.doi.org/10.1016/j.eswa.2009.05.032, URL http://linkinghub.elsevier.com/retrieve/pii/S0957417409004758.

[26] G.O. Olle, S. Cai, A hybrid churn prediction model in mobile telecommunication industry, Int. J. E-Educ. E-Bus. E-Manage. E-Learn. 4 (1) (2014) 1–8.

[27] Y. Huang, T. Kechadi, An effective hybrid learning system for telecommunication churn prediction, Expert Syst. Appl. 40 (14) (2013) 5635–5647, http://dx.doi.org/10.1016/j.eswa.2013.04.020, URL https://www.sciencedirect.com/science/article/pii/S0957417413002571.

[28] H. Lee, S. Kim, Black-box classifier interpretation using decision tree and fuzzy logic-based classifier implementation, Int. J. Fuzzy Logic Intell. Syst. 16 (1) (2016) 27–35.

[29] J.N. van Rijn, G. Holmes, B. Pfahringer, J. Vanschoren, Having a blast: Meta-learning and heterogeneous ensembles for data streams, in: 2015 Ieee International Conference on Data Mining, IEEE, 2015, pp. 1003–1008.

[30] A. Idris, A. Khan, Ensemble based efficient churn prediction model for telecom, in: 2014 12th International Conference on Frontiers of Information Technology, IEEE, 2014, pp. 238–244, http://dx.doi.org/10.1109/FIT.2014.52.

[31] K.W. De Bock, D. Van den Poel, An empirical evaluation of rotation-based ensemble classifiers for customer churn prediction, Expert Syst. Appl. 38 (10) (2011) 12293–12301.

[32] M. Farquad, V. Ravi, S.B. Raju, Churn prediction using comprehensible support vector machine: An analytical CRM application, Appl. Soft Comput. 19 (2014) 31–40.

[33] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, F. Herrera, A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches, IEEE Trans. Syst. Man Cybern. C (Appl. Rev.) 42 (4) (2011) 463–484.

[34] V.S. Babar, R. Ade, A review on imbalanced learning methods, Int. J. Comput. Appl. 975 (2015) 23–27.

[35] B. Kim, Y. Ko, J. Seo, Novel regularization method for the class imbalance problem, Expert Syst. Appl. 188 (2022) 115974.

[36] W. Verbeke, K. Dejaeger, D. Martens, J. Hur, B. Baesens, New insights into churn prediction in the telecommunication sector: A profit driven data mining approach, European J. Oper. Res. 218 (2012) 211–229.

[37] T.Y. Fei, L.H. Shuan, L.J. Yan, G. Xiaoning, S.W. King, Prediction on customer churn in the telecommunications sector using discretization and Naïve Bayes classifier, Int. J. Adv. Soft Comput. Appl. 9 (3) (2017) 23–34.

[38] Y. Kamei, T. Fukushima, S. McIntosh, K. Yamashita, N. Ubayashi, A.E. Hassan, Studying just-in-time defect prediction using cross-project models, Empir. Softw. Eng. 21 (2016) 2072–2106, http://dx.doi.org/10.1007/s10664-015-9400-x, URL http://link.springer.com/10.1007/s10664-015-9400-x.

[39] Ling, Li, Data mining for direct marketing problems and solutions, in: Proceedings of the Fourth International Conference on Knowl- Edge Discovery and Data Mining (KDD-98), AAAI Press, New York, NY, 1998.

[40] B. He, Y. Shi, Q. Wan, X. Zhao, Prediction of customer attrition of commercial banks based on SVM model, 31, Elsevier Masson SAS, 2014, pp. 423–430, http://dx.doi.org/10.1016/j.procs.2014.05.286,

[41] J. Burez, D.V. den Poel, Handling class imbalance in customer churn prediction, Expert Syst. Appl. 36 (2009) 4626–4636.

[42] A. Amin, B. Shah, A.M. Khattak, T. Baker, S. Anwar, et al., Just-in-time customer churn prediction: With and without data transformation, in: 2018 IEEE Congress on Evolutionary Computation (CEC), IEEE, 2018, pp. 1–6.

[43] M. Radisic, Just-in-time concept, 2009, Retrieved May 20, 2009.

[44] W.N. Poon, K.E. Bennin, J. Huang, P. Phannachitta, J.W. Keung, Cross-project defect prediction using a credibility theory based naive bayes classifier, in: 2017 IEEE International Conference on Software Quality, Reliability and Security (QRS), IEEE, 2017, pp. 434–441.

[45] S. Kanwal, J. Rashid, J. Kim, M.W. Nisar, A. Hussain, S. Batool, R. Kanwal, An attribute weight estimation using particle swarm optimization and machine learning approaches for customer churn prediction, in: 2021 International Conference on Innovative Computing (ICIC), IEEE, 2021, pp. 1–6.

[46] V. Umayaparvathi, K. Iyakutti, Automated feature selection and churn prediction using deep learning models, Int. Res. J. Eng. Technol. (IRJET) 4 (3) (2017) 1846–1854.

[47] B. Luo, S. Lin, Research on the anonymous customer segmentation model of telecom, in: 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), IEEE, 2020, pp. 1026–1031.

[48] A.A. Toor, M. Usman, Adaptive telecom churn prediction for concept-sensitive imbalance data streams, J. Supercomput. 78 (3) (2022) 3746–3774.

[49] I. Ullah, B. Raza, A.K. Malik, M. Imran, S.U. Islam, S.W. Kim, A churn prediction model using random forest: analysis of machine learning techniques for churn prediction and factor identification in telecom sector, IEEE Access 7 (2019) 60134–60149.

[50] M. Hemalatha, S. Mahalakshmi, Predicting chum customer in telecom using peer-grading regression learning technique, Volume 9, 2020, 1025–1036.

[51] I. Al-Shourbaji, N. Helian, Y. Sun, S. Alshathri, M. Abd Elaziz, Boosting ant colony optimization with reptile search algorithm for churn prediction, Mathematics 10 (7) (2022) 1031.

[52] A. Royer, C.H. Lampert, Classifier adaptation at prediction time, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1401–1409.

[53] H. Jain, A. Khunteta, S. Srivastava, Telecom churn prediction using CNN with variational autoencoder, in: A. Somani, A. Mundra, R. Doss, S. Bhattacharya (Eds.), Smart Systems: Innovations in Computing. Smart Innovation, Systems and Technologies, Vol. 235, Springer, Singapore, 2021, pp. 583–600, http://dx.doi.org/10.1007/978-981-16-2877-1_55.

[54] D. Chicco, G. Jurman, The advantages of the matthews correlation 609 coefficient (MCC) over F1 score and accuracy in binary classification 610 evaluation, BMC Genom. 21 (6).

[55] L. Geiler, S. Affeldt, M. Nadif, A survey on machine learning methods for churn prediction, Int. J. Data Sci. Anal. (2022) 1–26.

[56] S. Wael Fujo, S. Subramanian, M. Ahmad Khder, Customer churn prediction in telecommunication industry using deep learning, Inf. Sci. Lett. 11 (1) (2021) 24.

[57] J.D. Holland, The reverend Thomas Bayes, F. R. S. (1702-61), J. R. Statist. Soc. Ser. A (Gen.) 125, 451, http://dx.doi.org/10.2307/2982413.

[58] G. Shobha, S. Rangaswamy, Chapter 8-machine learning handbook of statistics, 2018.

[59] R.L. Kissell, J. Poserina, Optimal Sports Math, Statistics, and Fantasy, Academic Press, 2017.

[60] J.M. Bernardo, Modern Bayesian inference: Foundations and objective methods, in: Philosophy of Statistics, Elsevier, 2011, pp. 263–306.

[61] X. Han, Genetic algorithms, in: Computational Inverse Techniques in Nondestructive Evaluation, CRC Press, 2003, pp. 131–198.

[62] D. Sorokina, Application of additive groves ensemble with multiple counts feature evaluation to KDD cup'09 small data set, in: KDD-Cup 2009 Competition, PMLR, 2009, pp. 101–109.

[63] T.M. Khan, S. Xu, Z.G. Khan, et al., Implementing multilabeling, ADASYN, and relieff techniques for classification of breast cancer diagnostic through machine learning: Efficient computer-aided diagnostic system, J. Healthc. Eng. PMID: 33859807 (PMCID: PMC8009715) (March. 2021) 1–15, http://dx.doi.org/10.1155/2021/5577636.

[64] H. Motoda, H. Liu, Feature selection, extraction and construction, Commun. IICM (Inst. Inf. Comput. Mach., Taiwan) 5 (67–72) (2002) 2.

[65] I. Guyon, S. Gunn, M. Nikravesh, L.A. Zadeh, Feature Extraction: Foundations and Applications, Vol. 207, Springer, 2008.

[66] L. Ladha, T. Deepa, Feature selection methods and algorithms, Int. J. Comput. Sci. Eng. 3 (5) (2011) 1787–1797.

[67] A. Tamaddoni, S. Stakhovych, M. Ewing, Comparing churn prediction techniques and assessing their performance: A contingent perspective, J. Serv. Res. 19 (2) (2016) 123–141.

[68] H. Zhang, Y. Deng, F.T. Chan, X. Zhang, A modified multi-criterion optimization genetic algorithm for order distribution in collaborative supply chain, Appl. Math. Model. 37 (14) (2013) 7855–7864, http://dx.doi.org/10.1016/j.apm.2013.05.021, URL https://www.sciencedirect.com/science/article/pii/S0307904X13003429.

[69] E. Guerrero-Peña, A.F.R. Araújo, Multi-objective evolutionary algorithm with prediction in the objective space, Inform. Sci. 501 (2019) 293–316, http://dx.doi.org/10.1016/j.ins.2019.05.091, URL https://www.sciencedirect.com/science/article/pii/S0020025519305249.

[70] A. Eiben, J. Smith, Introduction to Evolutionary Computing, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015, http://dx.doi.org/10.1007/978-3-662-44874-8.

[71] S. Ruder, An overview of gradient descent optimization algorithms, 2016, arXiv preprint arXiv:1609.04747.

[72] A. Amin, S. Anwar, A. Adnan, M. Nawaz, K. Alawfi, A. Hussain, K. Huang, Customer churn prediction in the telecommunication sector using a rough set approach, Neurocomputing 237 (2017) 242–254.

[73] J. Osborne, O. A., The power of outliers (and why researchers should always check for them), Pract. Assess. Res. Eval. 9 (6) (2004) 1–12, http://dx.doi.org/10.7275/qf69-7k43.

[74] N.C. Schwertman, M.A. Owens, R. Adnan, A simple more general boxplot method for identifying outliers, Comput. Statist. Data Anal. 47 (1) (2004) 165–174.

[75] M. Daibo, Toroidal vector-potential transformer, in: 2017 Eleventh International Conference on Sensing Technology (ICST), IEEE, 2017, pp. 1–4.

[76] H.J. Steinhauer, T. Helldin, G. Mathiason, A. Karlsson, Topic modeling for anomaly detection in telecommunication networks, J. Ambient Intell. Humaniz. Comput. (2019) 1–12.

[77] S.J. Russell, Artificial Intelligence a Modern Approach, Pearson Education, Inc., 2010.

[78] D.J. Hand, P. Christen, N. Kirielle, F*: an interpretable transformation of the F-measure, Mach. Learn. 110 (3) (2021) 451–456.

[79] D. Chicco, G. Jurman, The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation, BMC Genom. 21 (1) (2020) 1–13.

[80] H. Jain, A. Khunteta, S.P. Shrivastav, Telecom churn prediction using seven machine learning experiments integrating features engineering and normalization, 1, 2021, 1–25.

[81] K.D. Swarup, K. Soumen, M. Subrata, S. Chandrika, B. Manju, Adaptive customer profiling for telecom churn prediction using computation intelligence, 1031, Springer Verlag, 2019, pp. 70–83, http://dx.doi.org/10.1007/978-981-13-8581-0_6,