

Customer Churn Prediction in Telecom Based on Machine Learning

Shengchen Wu *

Information Management and Information System, Shanghai University of Finance and Economics,
Shanghai, China

* Corresponding Author Email: 2021110087@stu.sufe.edu.cn

Abstract. Customer churn, a critical metric in the telecommunications industry, reflects the propensity of customers to discontinue service usage, thereby affecting the revenue and growth of a company. Customer churn is greatly affected by individuals and products, and there is no obvious rule, which greatly increases the difficulty of customer loss prediction. With the development of machine learning, the prediction in telecom churn using machine learning has attracted wide attention from researchers. However, there are great differences between different algorithms, which increases the difficulty of application. Therefore, this paper studies the issue of customer churn prediction from the aspect of algorithm. This paper introduces the common data sets and evaluation indexes of customer churn prediction, and then introduces logistic regression, decision tree, K-nearest neighbor, random forest, Gaussian naive Bayes, XGBoost, gradient enhancement and neural network algorithms. Then, through the experimental analysis, the differences between these models are compared. Finally, the paper summarizes the full text and discusses the future research hotspots.

Keywords: Customer Churn Prediction, Machine Learning, Neural Network.

1. Introduction

Reducing customer churn is of great significance for businesses in telecommunications competition. In telecommunications, customer churn represents a significant challenge with profound implications for business sustainability and growth. Customer churn, defined as the rate at which customers discontinue their service over a given period, is a critical metric that directly affects a company's revenue and profitability [1]. The ability to predict churn with high accuracy allows companies to take preemptive actions to retain customers, which is often more cost-effective than acquiring new ones. Churn predicting is to forecast the future consumption changes of customers through the analysis of customers' historical behavior and data. The difficulty of customer churn prediction lies in the significant differences in personal information, behavior habits and consumption records of customers, and it isn't easy to construct the input and output data of the model directly. In addition, due to the privacy issues involved, the data information between customers and merchants is challenging to obtain, increasing the difficulty of modeling.

Fortunately, machine learning has achieved excellent performance in image, text, speech, and other related tasks. With distributed representation, a variety of deep models became the method to solve problems on NLP, which brings inspiration to research on issues related to churner prediction [2]. Machine learning models can be trained on extensive data sets to complete complex nonlinear mappings to achieve accurate churn probability predictions [3]. However, there are more researches on customer churn algorithms based on machine learning, and less analysis and comparison between different methods, which significantly increases the difficulty of application of these algorithms.

The focus of this paper is to analyze the various ML models that have been applied to churn prediction, highlighting their strengths, limitations, and suitability for different scenarios. The paper will be organized as follows. In Section 2, three performance evaluation metrics and relevant data used in the research are presented. In Section 3, we explain the methods used in previous research. In Section 4, the performance of models will be introduced. In Section 5, conclusions and future challenges of churn prediction research are drawn.

2. Dataset

Data sets are the basis of model research, and the customer churn prediction model used in this paper is derived from Kaggle [4]. The publishers wanted to predict which customers may churn from the telecom company, by using the accumulated customers' usage, billing information, and their demographic information. The dataset consists of 7043 customers whose information is described by 20 attributes and a class label (whether they have churned, Yes/No). The training set includes 5634 records and the test set has 1409 records. Among this, 5174 customers have churned from the telecom company and there were 1869 non-churners in training and test set. Table.1 shows the information and the missing of the data.

Table 1. Information about data.

features	count	isna	%pop
CID	7043	0	1.000000
Gender	7043	0	1.000000
Senior	7043	0	1.000000
Partn	7043	0	1.000000
Depend	7043	0	1.000000
Ten	7043	0	1.000000
PhonServ	7043	0	1.000000
MultiLine	7043	0	1.000000
IntServ	7043	0	1.000000
OnSec	7043	0	1.000000
OnBack	7043	0	1.000000
DevProtect	7043	0	1.000000
TechSup	7043	0	1.000000
StreamTV	7043	0	1.000000
StreamMovi	7043	0	1.000000
Contract	7043	0	1.000000
NoPaperBill	7043	0	1.000000
PayMeth	7043	0	1.000000
MonChar	7043	0	1.000000
TotChar	7032	11	0.998438
Churn	7043	0	1.000000

In Table.1, partial TotalCharges' missing can be observed. To address the missing data, the missing ones are imputed based on the mean total charges for combination of gender, seniorcitizen and partner.

In evaluating the model effect, Accuracy, AUC, and KS statistics are generally used to calculate. Table.2 represent the result of predicted and actual values.

Table 2. Confusion Matrix.

	Be classified as Yes	Be classified as No
Actually Yes	TP	FP
Actually No	FN	TN

Accuracy: The percentage of accurate predictions the model produced out of all forecasts is represented by this key parameter. It is a simple measure of the overall efficacy of a model.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (1)$$

AUC: The area under the ROC curve is represented by the AUC (Area under the Curve) value. True Positive Rate (TPR) and False Positive Rate (FPR) are used as the horizontal and vertical coordinates in ROC to characterize the classifier's performance.

$$TPR = \frac{TP}{TP + FN} \quad (2)$$

$$FPR = \frac{FP}{FP + TN} \quad (3)$$

The AUC is a critical measure for models operating within imbalanced datasets, a common scenario in churn prediction. It assesses the model's ability to distinguish between churners and non-churners across all classification thresholds.

KS Statistic: The KS statistic is particularly valuable in churn prediction for its ability to measure the degree of separation between the distributions of churners and non-churners.

$$KS = \max(TPR - FPR) \quad (4)$$

3. Methods

In this section, the classical methods along with their outcomes, including Logical Regression, Decision Tree, K-Nearest Neighbors, Random Forest, Gaussian Naive Bayes, XGBoost, Gradient Boosting and Neural Network.

3.1. Logistic Regression (LR)

LR is a method of statistical analysis for regression analysis with two categories, which is fit for churn prediction because one customer only has two situations whether he/she is a churner or not. LR is a nonlinear model, and its parameters are generally estimated through maximum likelihood [5]. However, in the dataset, the number of churners is far less than the number of non-churners, resulting in an imbalance in the data that may affect the accuracy of the LR model.

3.2. Decision Tree (DT)

DT is a non-parametric supervised learning technique used for regression and classification. Through the learning of basic decision rules deduced from the input attributes, the model forecasts the value of a target variable [6]. DT is a dependable method for exploratory analysis because it is simple to comprehend and interpret. This approach is capable of modeling intricate, nonlinear relationships and handling both numerical and classified data.

But DTs may overfit, particularly if the tree gets overly complicated. They may produce whole distinct trees in response to slight changes in the data, which makes them unstable as well. The maximum tree depth in this paper was set at 10.

3.3. K-Nearest Neighbors (KNN)

KNN is a lazy learning method that finds the k items in the training set that are most similar to the object in the test or new set of data [7]. The Euclidean distance, whose formula is as follows, is frequently used to define the distance between two training objects and testing [8].

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (5)$$

KNN is a non-parametric technique for classification and regression. In churn prediction, it operates on the principle that similar customers exhibit similar behavior, including the likelihood of churning KNN is intuitive and straightforward, making it easy to implement [9].

KNN adapts quickly to changes in the input data and can be very accurate if the dataset is representative of the population. There is no need for assumptions about the distribution of the data, which makes it versatile across various applications. However, KNN can be computationally

expensive because it requires storing the entire dataset and calculating the distance for each query, which can be slow for large datasets. It also suffers from the curse of dimensionality; its performance degrades as the number of features increases unless dimensionality reduction is performed. Furthermore, KNN can be sensitive to the scale of the data and irrelevant features, which can skew the distance measurements. In the experiment of this paper, considering the calculation amount and effect, K is set to 20.

3.4. Random Forest

RF is a robust classifier consisting of multiple decision trees, where the decision trees are trained and the final result synthesizes the results of the decision trees [10]. This method reduces the risk of overfitting and classification errors.

In this paper, 80 percent of the data set is trained using 20 decision trees, which means building 20 trees and taking the average of the predicted results, and the maximum depth of the tree is set to 10. When the subtree partition sample is less than 60, the subtree partition stops, and the minimum sample number of leaf nodes is 30. In this method, the final results was decided by a vote of 20 decision trees according to majority voting rule.

3.5. Gaussian Naive Bayes

Naive Bayes approach is predicated on probability. When evaluating large-scale datasets, this technique can produce accurate prediction results with high efficiency [11]. Naïve Bayes is based on the Naive Bayes equation as follows:

$$P(C|x) = \frac{P(C)P(x|C)}{P(x)} \quad (6)$$

C is a class influenced by feature x, and when x is continuous, it will have a Gaussian distribution with average (μ) and standard deviation (σ) [12]. The equation will be as follows:

$$P(x|C) = \frac{1}{\sqrt{2\pi}\sigma_c} e^{-\frac{(x-\mu)^2}{2(\sigma_c)^2}} \quad (7)$$

3.6. XGBoost

Gradient descent and boosting are combined to create XGBoost (eXtreme Gradient Boosting). The approach is an enhancement of the Gradient Boosting Decision Tree (GBDT) boosting algorithm. [13]. In this algorithm, CART was used as the primary classifier and the first and second derivatives, regularization parameters and complexity functions are needed in the process of solving the objective function to improve the accuracy. This method find the best segment through accurate greedy algorithm. Because of its good cache optimization, this method performs well when training on the training set, but it will take more time on iteration [14]. In this paper, the number of weak classifiers was set as 50, and the maximum depth of the trees was 3.

3.7. Gradient Boosting

Gradient Boosting is a method combining regression and classification. It uses multiple weak prediction models (decision tree) to produce a strong prediction model. Every tree will learn from the mistake made by the former one. In this method, each model may make mistakes when predicting the residuals and then the algorithm will update the prediction using gradient descent after fitting the model through residuals. This process will be repeated until the value of prediction is accurate [15]. In order to ensure good results, the number of weak prediction models is set to 50, and the maximum depth is set to 3.

3.8. Neural Network

The form and operation of a biological neural network serve as the foundation for the neural network computer system [16]. The input, output, and hidden layers make up a fully linked neural network. The back propagation technique is used to optimize the network parameters during model training. The experiment's number of hidden layers was set to 10, and the activation function was set to sigmoid to prevent overfitting.

4. Experiment

This section mainly introduces the performance of algorithms used to predict telecom customer churn. The cross-validation method was used to train each model on the training set. KS, AUC and accuracy rate were shown in Table 3.

It can be seen that there is little difference between different algorithms, among which the method based on Gradient Boosting (such as XGBoost) gets better results. The introduction of regular terms by XGBoost can effectively reduce overfitting, which a certain improvement is compared with Gradient Boosting. The algorithm based on NN is second only to Gradient Boosting, which also gets good results because NN can learn more parameters. In contrast, logistic regression and KNN algorithms are less effective, but they have faster training speeds. In practice, the performance and effect of the model should be considered comprehensively.

Table 3. The Performance of each model.

methods	KS	AUC	Accuracy
Logistic Regression	0.534202	0.8349473	0.7389066
Decision Tree	0.520024	0.8250469	0.7389054
K Nearest Neighbors	0.492450	0.8124972	0.7433389
Random Forest	0.501064	0.8209147	0.7435194
Gaussian Naive Bayes	0.519478	0.8264908	0.6844163
XGBoost	0.537811	0.8392524	0.7548827
Gradient Boosting	0.549929	0.8407939	0.7550595
Neural Network	0.539288	0.8384138	0.7399703

5. Conclusion

The final evaluation on the hold-out set demonstrated that the models' performance was consistent with the cross-validated scores obtained during training. This consistency is indicative of the models' generalizability and their capacity to maintain performance on new, unseen data. Despite the close performance of the models, the Logistic Regression model was recommended for its computational efficiency and ease of interpretation. This recommendation underscores the importance of not only model performance but also operational considerations such as computational resources and the ability to convey insights to stakeholders.

As we look to the future, several challenges and research directions emerge from the current landscape of churn prediction:

(1) The trade-off between model complexity and interpretability remains a significant challenge. Future research must strive to develop models that not only provide high predictive accuracy but also offer transparent insights into their decision-making processes.

(2) The ever-changing patterns of customer behavior necessitate models that can adapt and learn in real-time. Research into online learning algorithms and dynamic modeling approaches that can continuously update with new data will be critical.

In conclusion, the field of customer churn prediction is expected to make significant progress, driven by both technological advances and a deeper understanding of customer behavior. Integrate machine learning models into business analytics frameworks to provide data reference for customer retention strategies in the telecom industry and beyond.

References

- [1] Yan L, Miller D J, Mozer M C, et al. Improving prediction of customer behavior in nonstationary environments. IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222). IEEE, 2001, 3: 2258-2263.
- [2] Young T, Hazarika D, Poria S, et al. Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine*, 2018, 13(3): 55-75.
- [3] Khodabandehlou S, Zivari Rahman M. Comparison of supervised machine learning techniques for customer churn prediction based on analysis of customer behavior. *Journal of Systems and Information Technology*, 2017, 19(1/2): 65-93.
- [4] <https://www.kaggle.com/becksddef/churn-in-telecoms-dataset/data>
- [5] Li P, Li S, Bi T, et al. Telecom customer churn prediction method based on cluster stratified sampling logistic regression. 2014.
- [6] Akbar T A R, Apriono C. Machine Learning Predictive Models Analysis on Telecommunications Service Churn Rate. *Green Intelligent Systems and Applications*, 2023, 3(1): 22-34.
- [7] Agrawal R. K-nearest neighbor for uncertain data. *International Journal of Computer Applications*, 2014, 105(11): 13-16.
- [8] Thakur A S, Sahayam N. Speech recognition using euclidean distance. *International Journal of Emerging Technology and Advanced Engineering*, 2013, 3(3): 587-590.
- [9] Xia Y, Cui B, Duan Y. Analysis and Prediction of Telecom Customer Churn based on Machine Learning. *Highlights in Science, Engineering and Technology*, 2022, 16: 131-145.
- [10] Breiman L, Cutler R A. Random forests machine learning. *Journal of clinical microbiology*, 2001, 2: 199-228.
- [11] Yulianti Y, Saifudin A. Sequential feature selection in customer churn prediction based on Naive Bayes. *IOP Conference Series: Materials Science and Engineering*. IOP Publishing, 2020, 879(1): 012090.
- [12] Jain M, Richariya V. An improved techniques based on naive Bayesian for attack detection. *International Journal of Emerging Technology and Advanced Engineering*, 2012, 2(1): 324-331.
- [13] Wang H, Liu S, Sun C, et al. Short term prediction of cell uplink and downlink traffic based on Prophet Algorithm. *Proceedings of the 7th International Conference on Cyber Security and Information Engineering*. 2022: 419-423.
- [14] Hanif I. Implementing extreme gradient boosting (xgboost) classifier to improve customer churn prediction. 2020.
- [15] Friedman J H. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 2001: 1189-1232.
- [16] Pagare R, Khare A. Churn prediction by finding most influential nodes in social network. 2016 *International Conference on Computing, Analytics and Security Trends (CAST)*. IEEE, 2016: 68-71.