Journal on Artificial Intelligence
DOI:10.32604/jai.2021.026851
Article

Tech Science Press

Churn Prediction Model of Telecom Users Based on XGBoost

Hao Chen*, Qian Tang, Yifei Wei and Mei Song

Beijing University of Posts and Telecommunications, Beijing, 100876, China
*Corresponding Author: Hao Chen. Email: chaohao@bupt.edu.cn
Received: 05 January 2022; Accepted: 10 January 2022

Abstract: As the cost of accessing a telecom operator's network continues to decrease, user churn after arrears occurred repeatedly, which has brought huge economic losses to operators and reminded them that it is significant to identify users who are likely to churn in advance. Machine learning can form a series of judgment rules by summarizing a large amount of data, and telecom user data naturally has the advantage of user scale, which can provide data support for learning algorithms. XGBoost is an improved gradient boosting algorithm, and in this paper, we explore how to use the algorithm to train an efficient model and use this model one month in advance to predict whether users will churn. Our work is mainly divided into two aspects: (1) By completing data exploration, feature engineering and data preprocessing, we obtained a data set that can be used to train a prediction model and features that can effectively predict user churn. And using these features and data sets, two prediction models were trained based on Random Forest and XGBoost. (2) According to the business needs of telecom operators, we continuously evaluated and optimized these models. And by comparing the test results of the two models, we proved that the XGBoost model performs better for the precision and recall of user churn.

Keywords: Telecom users; churn prediction; XGBoost

1 Introduction

The rapid development of the Internet has led to intensified competition among telecom operators, and applications developed by Internet companies have gradually replaced the previous communication methods dominated by operators [1]. In order to have more user resources and ensure the company's profits, operators continue to lower their own network access standards. The subsequent phenomenon of "user churn after arrears" has brought huge economic losses to the company. Operators have also found that the cost of inviting a new user into the network is much higher than the cost of retaining existing users [2], which makes them more eager to find ways to prevent user churn.

Telecom operators have divided customers into different stars based on business experience and some traditional methods in the past, which has certain reference significance for judging whether users will churn after arrears. However, the results of artificial division are always subjective. These experiences and traditional methods also have many shortcomings, and compared with the number of staff, the scale of telecommunication users is too large. According to the "2020 China Radio Management Annual Report" issued by the Radio Administration, by the end of 2020, the total number of mobile phone users in China has exceeded 1.6 billion, and the information contained in these user data is huge [3]. Research on how to effectively use this information to drive industry progress is imminent.

Nowadays, research on user churn in the telecommunications industry generally focuses on two aspects. An important direction is to study how to use machine learning methods to improve the ability to judge user churn [4]. For example, Lu et al. [5] began to use Boosting to predict customer churn in the telecommunications industry a few years ago. Ullah et al. [6] proposed a method to reduce customer churn

by establishing a Random Forest model, which achieved an efficient prediction. Alboukaey et al. [7] used LSTM-based and CNN-based models to predict user churn on the basis of Random Forest. The experimental results affirm the performance of these three models in that reference [7], proposed that when predicting customer churn, it is important to consider the daily behavior of customers and prove that the performance of these models is significantly better than the statistical-based models. Another research direction focuses on the impact of specific feature on user churn. For example, Mahajan et al. [8] analyzed and listed a number of factors related to user churn in the telecommunications industry. Grzybowski et al. [9] analyzed the impact of whether fixed terminals and mobile terminals are bundled on user churn.

In this paper, to response the problems above, we aim to combine operators' business experience and feature importance analysis to obtain a set of high-influential features, and take advantage of XGBoost to accurately predict user churn. Through experiments, we compare the performance of two different models on the test set, and concludes that the optimized Random Forest model has better precision, while the optimized XGBoost model has better recall. In the business field, operators usually trade off precision and recall, and the XGBoost model performs better in this regard, and we consider it is the optimal model to predict user churn.

2 Prediction Model

This section mainly shows the data analysis, processing and the use of XGBoost in the process of model building.

2.1 Data Exploration and Feature Engineering

In this paper, the telecom data is provided by a certain operator, which is all users in a city, with about one million people. The purpose of data exploration is to find effective features to train the model.

For label selection, we define whether users have churn as labels, divide users off-net as negative samples, and users who receive services normally as positive samples. In one month, among the approximately one million users, 97.9% of users received services normally, and 2.1% of users who had cancelled services.

The selection of user features is required to effectively determine whether the user has a tendency to churn. The specific feature selection steps are as follows:

1) Based on business experience and statistics, we first perform basic screening of features. Based on business judgment, some features can be easily selected. For example, it is not possible to predict whether users will churn based on the difference of cities, so we delete the feature of which region the user belongs to; Users who are always in arrears are more likely to churn, so we will use the feature of monthly arrears; And Some features are selected based on statistical methods. For example, the payment channels used by normal users and churn users are basically similar in statistics. It is difficult to determine whether users have a tendency to churn based on this feature, so it is deleted.

2) At the end of the paper, we delete features of the trained model and observe the impact of deleting a feature on the model. If deleting the features has little effect on the performance of the model, then we think that the feature can be deleted. The model evaluation index at the end of this paper is the sum of the precision and the recall, and we limit that the deletion of features will not cause the model to drop by 1% on this index.

Feature engineering mainly combines original features provided by operators to generate some new features. For example, the date when the user starts the service will be processed as how long the service has been received; the payment record will be calculated as the number of payments based on how many times it has been recorded; And encode string-type data such as the main card flag into integer-type 0 and 1.When the original features are processed, their interpretation from a business perspective is clearer, and their data types are more suitable for training the model, thereby further improving the performance of the model.

Table 1 shows all the features used to train the model.

Table 1: Features of telecom users

Feature name	Description
innet_time	The time when the user accepted the service
total_amount	The total amount spent this month
pack_amount	The amount of package for this month
payment amount	The amount of payment for this month
payment times	The number of payments this month
flag_main	The flag indicating that this card is the main card
arrears_fee	The amount of arrears for this month
last_fee	The amount of phone bill remaining last month

2.2 Data Pre-Processing

The data we extract includes the user's ID, the label of the current month, and the features data of the previous three consecutive months. For the data to be able to train the model more effectively, we preprocessed the data as shown in Fig. 1.

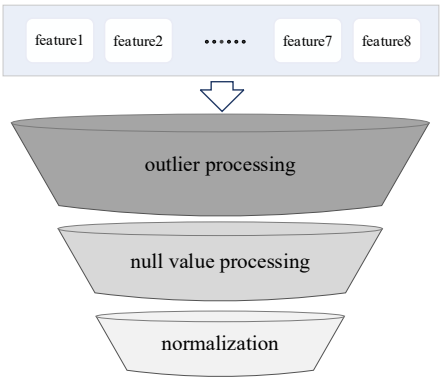

Figure 1: Steps of data preprocessing

In the past, operators left part of the test data during internal tests, and some errors often occur when data was entered into the database, which would cause some feature values to be unreasonable. For example: some users' payment is negative, or reaches several hundred thousand yuan. In order to make the trained model effective, we deleted this kind of outliers.

In the process of matching various features of a user, there will be a situation where the feature value is null, which is caused by incomplete data entry. For example, if there is a user's basic information, but his bills in recent months have not been entered, then the user's bill will be null after all features are matched. In order to maximize the use of the information contained in the data, we subdivide the data with null values into long-null data and short-null data according to whether there is billing data.

Long-null data will lose most of the information due to the lack of billing information. Adding this to the training set may affect the performance of the model, so it will be deleted. Short-null data have billing information. Although there are some null values, such user data still carries a large amount of information, which can be used to train models. We set the null value in these users to zero for model training.

For numerical features, they may affect each other due to different measurement units. For example, the amount of billing data is often very large, which may affect features like the number of payments, which will reduce the impact of this feature on the model. In order to eliminate the influence between features, we map all numerical features to the range of 0–1.

2.3 XGBoost

XGBoost is an algorithm based on tree learner. Its integration strategy is to serially generate trees to predict errors, during which the predicted value gradually approaches the real errors, and finally add up the predicted errors of each tree, and the cumulative predicted value is the result [10].

Suppose there are data of n users, and each user data contains m features, then $\mathrm{D} = \{(x_i, y_i)\}$, $i = 1, 2, \ldots, n$, where $x_i \epsilon R^m$, $y_i \epsilon R$. When training the t-th tree, Eq. (1) is the objective function that needs to be continuously minimized.

$$\mathrm{Obj}^{(\mathrm{t})} = \sum_{i=1}^{n} l(y_i, \hat{y_i}^{(t-1)} + f_t(x_i)) + \Omega(f_t) + C \tag{1}$$

where $\hat{y_i} = \sum_{k=1}^{K} f_k(x_i)$, $f_k \epsilon F$, and it stands for decision tree integration. K is the number of activation functions, F is a collection of all CART trees, f_k is a specific CART tree. l is the loss function, $\Omega(f_t)$ is regular term and C is a constant. When Taylor expansion is used to approximate it, the new objective function can be obtained as Eq. (2).

$$\mathrm{Obj}^{(\mathrm{t})} \approx \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) + C \tag{2}$$

where $g_i = \partial_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right)$, $h_i = \partial^2_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$, and they represent the transfer of residuals.

Compared with other boosting tree algorithms, the cost function of XGBoost uses the first and second derivatives and adds a regular term to control the complexity of the model [11].

3 Model Testing and Analysis

3.1 Model Evaluation Index

There are many indicators for evaluating the efficiency of a model. Starting from business needs, we select accuracy, precision and recall as the evaluation indicators of the model [12].

Accuracy is the ratio of correctly classified users to the total number of users. Then we introduce the concept of confusion matrix, which can show the specific division of the test set. After model prediction, there are generally four types of data, which are true positive examples, false positive examples, true negative examples, and false negative examples. We express these four types of data by TP, FP, TN, and FN. Table 2 is the confusion matrix of the classification results of telecom users.

Table 2: Confusion matrix of classification results

True situation	Prediction	
	On-net users	Off-net users
On-net users	TP	FN
Off-net users	FP	TN

In this experiment, the meaning of the precision is: When a list of off-network users is given, the proportion of users who will actually be off-network next month. The meaning of the recall is: the proportion of the detected off-network users to the total off-network users in the next month. The precision and recall are defined as formulas Eq. (3) and Eq. (4).

$$P = \frac{TP}{TP + FP} \tag{3}$$

$$R = \frac{TP}{TP + FN} \tag{4}$$

Usually due to the conflict of physical meaning, the recall and precision can only be optimized by one to the best. When the recall is increased, the precision will decrease. Similarly, when the precision is increased, the recall will also decrease.

3.2 Model Test

In this experiment, we train two models, the Random Forest model and the XGBoost model. The total data set is divided into two parts, the first part is about 700,000 users' data for model training, and the other about 300,000 users' data is used for model testing. The evaluation index is selected as accuracy, and the test results are shown in Table 3.

Table 3: Accuracy of prediction model

Model	Random forest	XGBoost
Accuracy	93.28%	91.27%

Judging from the accuracy shown by the test results, the Random Forest model performs better.

The accuracy of the model represents the efficiency of the model to a certain extent, and can be used as one of the important indicators for evaluating the performance of the model. However, a single evaluation index is not enough to objectively evaluate the real performance of the model, nor can it support the business needs of operators. Next, we use confusion matrix and precision and recall to evaluate the model. Table 4 and Table 5 are the confusion matrices of the two models, which show the specific classification of on-net and off-net users by the model.

Table 4: Confusion matrix of random forest model

True situation	Prediction	
	On-net users	Off-net users
On-net users	287521	19449
Off-net users	1626	4902

Table 5: Confusion matrix of XGBoost model

True situation	Prediction	
	On-net users	Off-net users
On-net users	280710	26260
Off-net users	1117	5411

Using the confusion matrix, we calculate the precision and recall of each model.

In terms of recall, the Random Forest model has a lower recall of 75.1% for off-net users, while the XGBoost model has a higher recall of 82.89% for off-net users.

In terms of precision, the Random Forest model has a higher precision of 20.13% for off-net users, while the XGBoost model has a lower precision of 17.09% for off-net users.

After introducing two new evaluation indicators, the performance of the two models on the test set is as the Table 6.

Table 6: Evaluation of prediction model

	Random forest	XGBoost
Recall of off-net	75.1%	82.89%
Precision of off-net	20.13%	17.09%

3.3 Model Optimization

The specific business needs of telecom operators are to reduce the maximum arrears of users with the trend of off-net according to the list of off-net users output by the model, or to make some actions in advance to keep them on the network. It requires as many off-net users as possible in the output list, and can cover as many off-net users as possible, so it requires the model to have as high a recall and precision as possible for off-net users. Considering the business requirements, in this paper, we will comprehensively consider the two evaluation indicators of recall and precision to optimize the model.

In order to comprehensively improve the performance of recall and precision, we modify the weight of training samples. When training the model, if the weight of off-net users is greater, the model will have a stronger ability to classify off-net users. It should be noted that while the recall of off-net users continues to increase, the model's precision of off-net users may become weaker. The explanation is that although the recall of off-net users will continue to increase, the ability to judge on-net users will become weaker, which will cause more on-net users to be misjudged as off-net users, and the precision of off-net users will decrease. Weighing these two evaluation indicators, we take the maximum sum of the two indicators as the objective function to optimize the model. We adjust the Random Forest model parameter class_weights and XGBoost model parameter scale_pos_weight, which all adjust the weight of the class through repeated sampling to optimize the effect of the model. The test results are shown in Table 7.

Table 7: Evaluation of the model under the optimal parameters

	Random forest	XGBoost
Recall of off-net	71%	69.98%
Precision of off-net	29.6%	32.2%

Comparing the results, the optimized XGBoost model has better performance in the classification of off-net users. It is a trade-off between the precision and recall of off-net users, which can meet the above-mentioned business needs.

3.4 Experimental Results Analysis of the Optimized Model

After preliminary testing and further optimization based on business needs, we have the following results.

In the prediction of off-net users, the XGBoost model performs better. The specific performance is that when the model is used to test a user set, the correct users account for 96.31% of all users; the model can detect 69.98% of all off-net users; in the output list of off-net users, there are 32.2% is indeed an off-net user. The test results are shown in Table 8.

Table 8: Test results of the optimized XGBoost model

Accuracy	Recall	Precision
96.31%	69.98%	32.2%

After testing a model, we delete individual features or combined features, and observe the changes in model performance in the process. If the feature changes have little effect on the final performance of the model, we will delete them to simplify the model [13].

In this paper, we only show the features that have a certain degree of influence on the model. Many features have been deleted according to the judgment criteria in Part 2 of Section 2.1 when performing feature engineering.

4 Conclusion

In order to reduce the negative impact caused by the lowering of network access standards, and solve the problem of telecom user churn. This paper proposes to establish an optimized XGBoost model, and uses a telecom user test set to verify it. The model results show that the XGBoost-based telecom user churn prediction model performs well in the evaluation indicators: the accuracy of churn users is over 96%, the

recall is close to 70%, and the precision is over 32%. When providing services to telecom users, the application of this model helps to reduce the number of off-net user, increase the profitability of operators, and meet the company's requirements for improving service quality.

Funding Statement: This work was supported by the National Natural Science Foundation of China (61871046).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] H. Chen, L. Li and Y. Chen, "Sustainable growth research–A study on the telecom operators in China," *Journal of Management Analytics*, pp. 1–15, 2021.

[2] S. A. Qureshi, A. S. Rehman, A. M. Qamar, A. Kamal and A. Rehman, "Telecommunication subscribers' churn prediction model using machine learning," in *Eighth Int. Conf. on Digital Information Management*, Islamabad, Pakistan, pp. 131–136, 2013.

[3] Y. Hong, Z. Li and J. Wang, "Business value of telecom operators' big data," *Journal of Physics: Conference Series*, vol. 1437, no. 1, pp. 12–67, 2020.

[4] V. Umayaparvathi and K. Iyakutti, "A survey on customer churn prediction in telecom industry: Datasets, methods and metrics," *International Research Journal of Engineering and Technology*, vol. 3, no.4, pp. 1065–1070, 2016.

[5] N. Lu, H. Lin, J. Lu and G. Q. Zhang, "A customer churn prediction model in telecom industry using boosting," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp.1659–1665, 2012.

[6] I. Ullah, B. Raza, A. K. Malik, M. Imran, S. U. Islam *et al.,* "A churn prediction model using random forest: analysis of machine learning techniques for churn prediction and factor identification in telecom sector," in *IEEE Access*, vol. 7, pp. 60134–60149, 2019.

[7] N. Alboukaey, J. Ammar and G. Nada, "Dynamic behavior based churn prediction in mobile telecom," *Expert Systems with Applications*, vol. 162, 113779, 2020.

[8] V. Mahajan, R. Misra and R. Mahajan, "Review on factors affecting customer churn in telecom sector," *International Journal of Data Analysis Techniques and Strategies*, vol. 9, no. 2, pp.122–144, 2017.

[9] L. Grzybowski, J. Liang and C. Zulehner, "Bundling and consumer churn in telecommunications markets," *Review of Network Economics*, vol. 20, no. 1, pp. 35–54, 2021.

[10] T. Q. Chen, T. He and M. Benesty, "XGboost: extreme gradient boosting," *R Package*, vol. 1, no. 4, pp. 1–4, 2015.

[11] T. Chen and C. Guestrin, "XGboost: A scalable tree boosting system," in *Proc. of the 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, San Francisco, USA, pp. 785–794, 2016.

[12] M. Sokolova, N. Japkowicz and S. Szpakowicz, "Beyond accuracy, *F*-score and ROC: A family of discriminant measures for performance evaluation," in *Australasian Joint Conf. on Artificial Intelligence*, Hobart, TAS, Australia, pp. 1015–1021, 2006.

[13] X. P. Shi, Y. D. Wong, M. Z. Li, C. Palanisamy and C. Chai, "A feature learning approach based on XGBoost for driving assessment and risk prediction," *Accident Analysis & Prevention*, vol. 129, pp. 170–179, 2019.