Customer Churn Prediction in Telecommunication

Kajanan Selvanesan
Department of Computer Science and Engineering,
University of Moratuwa,
Colombo, Sri Lanka.
kajananselvanesan@gmail.com

***Abstract*—Customer churn is the loss of customers by a service provider. Since acquiring new customers costs more than retaining existing ones, it is a big concern for telecom service providers. Therefore, it's very important for telecommunication industries to analyze the behaviors of the various customer to predict which customers are about to leave the subscription from a telecom company. The sample dataset we used for this experiment has been created by Chatterbox Telco Pvt Ltd. In this paper, we focus on various machine learning techniques for predicting customer churn through which we can build the classification models such as Logistic Regression, SVM, Linear SVM, K-Nearest Neighbor, Gaussian Naive Bayes, and Decision Tree and also compare the performance of these models using cross-validation. Eventually, we try a ensemble technique to improve the accuracy more. We end up with the Random Forest as the best model with a 0.993067 f1 score.**

***Index Terms*—Customer churn, Telecommunication, Classification, Decision Tree, Random Forest.**

I. Introduction

Customer churn is the loss of customers by a business for different reasons such as poor service and better prices somewhere else. It is one of the most critical and challenging problems for telecommunication companies, credit card companies, cable service providers, etc. Since acquiring new customers costs more than retaining existing ones, analyzing customer churn and finding ways to reduce it are vital for businesses.

The most difficult problem faced by the telecom industry is customer churn. Customer churn models aim to detect customers with a high probability to jump or leave the service provider. A database of customers who might churn allows the company to target those customers and start retention strategies that reduce the percentage of a customer churning. Retention of old customers is always the preferable option for the company. Attracting new customers costs almost five to six times more than retaining the old customers. Attracting a new customer includes recruits of manpower, cost of publicity, and discounts. A loyal customer, who has been with a business for quite a long time, tends to generate higher revenues and is less sensitive to competitor prices. Such customers also cost less to keep and in addition, provide valuable word-of-mouth marketing to the business by referring to their relatives, friends, and other acquaintances. A small step towards retaining an existing customer can lead to a significant increase in revenues and profits. The requirement of retaining customers craves for accurate customer churn prediction models that are both accurate and comprehensible. The Models have to identify customers who are about to churn and their reason for churn to avoid the losses to the telecom industry, a model should be developed to identify the reasons to churn and the improvements required to retain customers.

II. Methodology

A. Dataset

The sample dataset we used for this experiment has been created by Chatterbox Telco Pvt Ltd. It consists of 19 predictor variables and one target variable.

TABLE I
Each variable with its type

NO	Variable Name	Data Type	Attribute Type
01	customer_id	Integer	Categorical Nominal
02	account_length	Integer	Metric Discrete
03	location_code	Integer	Categorical Nominal
04	international_plan	String	Categorical Nominal
05	voice_mail_plan	String	Categorical Nominal
06	number_vm_messages	Integer	Metric Discrete
07	total_day_min	Float	Metric Continuous
08	total_day_calls	Integer	Metric Discrete
09	total_day_charge	Float	Metric Continuous
10	total_eve_min	Float	Metric Continuous
11	total_eve_calls	Integer	Metric Discrete
12	total_eve_charge	Float	Metric Continuous
13	total_night_minutes	Float	Metric Continuous
14	total_night_calls	Integer	Metric Discrete
15	total_night_charge	Float	Metric Continuous
16	total_intl_minutes	Float	Metric Continuous
17	total_intl_calls	Integer	Metric Discrete
18	total_intl_charge	Float	Metric Continuous
19	customer_service_calls	Integer	Metric Discrete
20	Churn (Target variable)	String	Categorical Nominal

The dataset contains the variables mentioned-above with one extra column, with some quality issues. Therefore it's essential to do the data pre-processing to this dataset. Within the variables discussed above, 4 are categorical columns:

- location_code - 445 or 452 or 547
- international_plan - yes or no
- voice_mail_plan - yes or no
- Churn - Yes or No

You can see the dataset description in the below image.

	count	mean	std	min	25%	50%	75%	max
customer_id	2321.0	2161.000000	670.159309	1001.00	1581.000	2161.00	2741.0000	3321.00
account_length	2319.0	101.400172	40.044985	1.00	74.000	101.00	127.0000	232.00
location_code	2321.0	473.470918	42.011853	445.00	445.000	452.00	452.0000	547.00
number_vm_messages	2318.0	7.557377	14.250001	-202.00	0.000	0.00	14.0000	51.00
total_day_min	2320.0	182.718103	73.332822	-179.90	144.000	180.35	221.0000	2283.90
total_day_calls	2318.0	105.324418	221.100535	-1.00	87.000	102.00	115.0000	10700.00
total_day_charge	2316.0	30.961524	9.830271	-25.60	24.480	30.60	37.5900	60.96
total_eve_min	2318.0	203.511734	115.552100	-103.30	165.925	202.40	236.4000	5186.40
total_eve_calls	2317.0	100.125162	20.536224	-80.00	87.000	101.00	114.0000	170.00
total_eve_charge	2313.0	17.123130	4.327327	0.00	14.180	17.21	20.0900	30.83
total_night_minutes	2319.0	209.543467	408.066120	23.20	167.350	201.10	235.0500	19700.00
total_night_calls	2316.0	87.641192	12.737232	33.00	79.000	90.00	98.0000	105.00
total_night_charge	2316.0	9.436710	18.656075	1.04	7.530	9.05	10.5825	900.15
total_intl_minutes	2319.0	10.247736	2.795472	-9.30	8.600	10.30	12.0000	18.30
total_intl_calls	2318.0	4.439172	2.461172	0.00	3.000	4.00	6.0000	20.00
total_intl_charge	2316.0	2.773364	0.733526	0.00	2.320	2.78	3.2400	4.94
customer_service_calls	2320.0	1.651724	1.429166	0.00	1.000	1.00	2.0000	9.00
Unnamed: 20	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN

Fig. 1. Description of the dataset

B. Data Preprocessing

1) Data Cleaning: There are some quality issues in the dataset. Therefore it is essential to do data cleaning.

- Detect Remove Duplicates: Without considering the customer_id, if you see the dataset, there are four duplicate rows in the indexes 772, 807, 1704, and 2295. Those rows have been removed.

- Handling Missing Values, Out of Range Values Outliers: In the dataset, there are negative values. But no attribute gets negative values here. Such values have been replaced by using their absolute values. We can take this decision with the help of observations on the dataset.
  Most of the columns contain missing values in the dataset. It's shown in the below image.

```
customer_id                0
account_length             2
location_code              0
intertiol_plan             3
voice_mail_plan            6
number_vm_messages         3
total_day_min              1
total_day_calls            3
total_day_charge           5
total_eve_min              3
total_eve_calls            4
total_eve_charge           8
total_night_minutes        2
total_night_calls          5
total_night_charge         5
total_intl_minutes         2
total_intl_calls           3
total_intl_charge          5
customer_service_calls     1
Churn                      5
Unnamed: 20             2317
dtype: int64
```

Fig. 2. Columns with missing values count

Unwanted column (Unnamed: 20 that contain full of null values) has been removed from the dataset.
Boxplot and scatterplot have been used to identify some outliers and out-of-range values. Initially, those values had been replaced with null. After that, outliers and out-of-range values have been dealt with the missing values. If any column in the total_day_min, total_day_call, or total_day_charge has a value of zero for a row, those values have been substituted with regard to the values in the other two columns. If any of the two values for the columns listed above is zero in a row, then the third column in that row is also zero. This technique is applied to evening, night, and international calls also.

TABLE II
MISSING VALUES HANDLING FOR EACH VARIABLE

NO	Variable Name	Handling Missing Values
01	customer_id	There are no missing values.
02	account_length	Filled using the median.
03	location_code	Filled using the last valid observation.
04	international_plan	Filled using the most frequent value. {no}
05	voice_mail_plan	Filled using the most frequent value. {no}
06	number_vm_messages	If voice_mail_plan is no then filled using 0 else filled using the median of where voice_mail_plan is yes.
07	total_day_min(/ total_eve_min/ total_night_minutes/ total_intl_minutes)	Sorted the dataset by total_day_charge(/ total_eve_charge/ total_night_charge/ total_intl_charge) then filled missing values using the linear interpolation of the variable before and after a timestamp for a missing value.
08	total_day_calls(/ total_eve_calls/ total_night_calls/ total_intl_calls)	Sorted the dataset by total_day_min(/ total_eve_min/ total_night_minutes/ total_intl_minutes) then filled missing values using the last valid observation.
09	total_day_charge(/ total_eve_charge/ total_night_charge/ total_intl_charge)	Sorted the dataset by total_day_min(/ total_eve_min/ total_night_minutes/ total_intl_minutes) then filled missing values using the linear interpolation of the variable before and after a timestamp for a missing value.
10	customer_service_calls	Filled using the most frequent value. {1}
11	Churn	Removed the specific rows.

2) Data Transformation: There are some non-numerical categorical columns. We cannot feed them into models. We need to transform such features.

- Data Encoding: There are four categorical columns altogether that need to be encoded. I used a dummy encoding technique for the location code, and label encoding technique for the international_plan, voice_mail_plan and Churn. When using label encoding, I used 1 for yes/Yes and 0 for no/No.

C. Handling Imbalanced Dataset

Imbalanced classifications pose a challenge for predictive modeling as most of the machine learning algorithms used for classification were designed around the assumption of an equal number of examples for each class. This results in models that have poor predictive performance, specifically for the minority class. This is a problem because typically, the minority class is more important, and therefore the problem is more sensitive to classification errors for the minority class(Yes) than the

majority class(No). Imbalance of the dataset is shown in the below image.

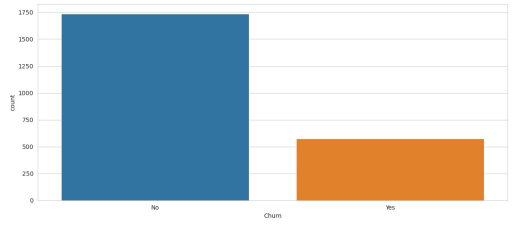

Fig. 3. Imbalance of the dataset

The dataset has been oversampled (duplicate examples in the minority class) to overcome this issue. After the oversampling, the dataset has been shuffled. You can see the picture of the balanced dataset below.

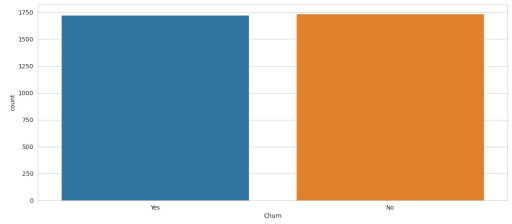

Fig. 4. Balanced dataset

D. Feature Engineering

1) Feature Selection: Unwanted features have been dropped using the filtering feature selection technique. Here, customer_id has been dropped as an unwanted feature.

2) Feature Creation: Some extra high level features have been created. You can see them below one by one.

- total_min: It has been created by adding total_day_min, total_eve_min, total_night_minutes and total_intl_minutes.

- expected_total_day_charge, expected_total_eve_charge, expected_total_nyt_charge: There are possibilities for a customer to churn when Chatterbox charges an unusual charge per minute from the customer during the day or evening, or night. The below picture shows the relationship between the total minutes and the total charge for any specific period. Here, you can notice that the most probably churned customers are not in the plotted straight line. But that's not the case for unchurned customers.
  First, we trained a linear regression model for each time period by considering total minutes as a predictor variable and total charge as a target variable. Then, expected_total_charge values of each time period has been predicted for the correct training examples of this model using a correct trained model.

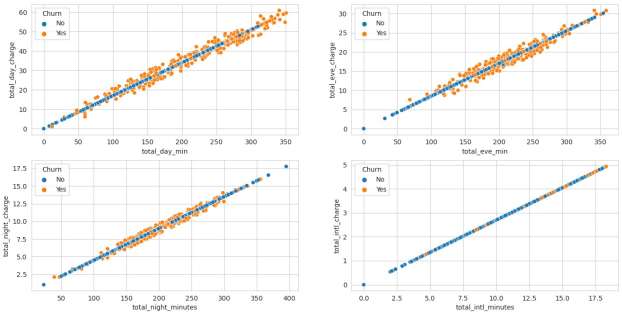

Fig. 5. Linear relationships between minutes and charge

- error_total_day_charge, error_total_eve_charge, error_total_nyt_charge: They have been created by taking absolute values for different between expected_total_charge and total_charge of each time period.

You can notice that mutual information score of newly created features with the target variable relatively higher than that of given features.

```
error_total_nyt_charge       0.269731
error_total_day_charge       0.266601
total_night_charge           0.258302
error_total_eve_charge       0.228885
total_day_charge             0.192910
total_eve_charge             0.180220
expected_total_day_charge    0.154610
total_day_min                0.152482
total_min                    0.150354
total_eve_min                0.111154
expected_total_eve_charge    0.106364
total_intl_minutes           0.087079
account_length               0.086676
total_night_minutes          0.083157
expected_total_nyt_charge    0.081439
customer_service_calls       0.073327
total_intl_charge            0.066472
intertiol_plan               0.044320
number_vm_messages           0.037434
voice_mail_plan              0.014623
total_eve_calls              0.009636
total_intl_calls             0.007411
total_day_calls              0.006355
location_code_452            0.005739
location_code_547            0.002007
total_night_calls            0.001359
location_code_445            0.000000
Name: MI Scores, dtype: float64
```

Fig. 6. Mutual information score of each feature with the target variable

E. Models

Customer churn is a binary classification problem. Therefore initially, we started with six different classification models (Logistic Regression, Support Vector Machine(SVM), Linear SVM, K-Nearest Neighbor, Gaussian Naive Bayes, and Decision Tree) and trained them.

F. Evaluation Metrics

To select the best model from the above list, we did the 5-fold cross-validation (We used f1 scoring within the cross-validation) to our dataset. Every data point gets to be tested

exactly once and is used in training k-1 (in this case 4) times when we use cross-validation. And the f1 score is balancing precision and recall on the positive class while accuracy score looks at correctly classified observations both positive and negative.

G. Model Selection

Here, Decision Tree works well and has performed better than the rest.

TABLE III
MEAN OF F1 SCORE AND STANDARD DEVIATION OF F1 SCORE

NO	Model Name	F1 Score Mean	F1 Score Standard Deviation
01	Decision Tree	0.970073	0.004975
02	K-Nearest Neighbor	0.837872	0.007510
03	Logistic Regression	0.830532	0.005424
04	Gaussian Naive Bayes	0.712258	0.022474
05	Linear SVM	0.694023	0.030608
06	Support Vector Machine	0.592401	0.024580

H. Improve The Selected Model

We tried the ensemble technique with the above-selected model. Random Forest has been chosen as the ensemble model for the Decision Tree. We trained both and evaluated them. As a result Random Forest have given the better result over Decision Tree.

TABLE IV
MEAN OF F1 SCORE AND STANDARD DEVIATION OF F1 SCORE

NO	Model Name	F1 Score Mean	F1 Score Standard Deviation
01	Random Forest	0.991926	0.004876
02	Decision Tree	0.970073	0.004975

Finally, we did the hyper parameter tuning for the Random Forest by using RandomizedSearchCV and GridSearchCV. RandomizedSearchCV had been used to find the ranges of the best hyper parameters. GridSearchCV had been used to find the best hyper parameters within the ranges that have been found using RandomizedSearchCV. We had tried to tune the hyper parameter of Random Forest such as bootstrap, max_depth, max_features, min_samples_leaf, min_samples_split, and n_estimators. As a result tuned Random Forest have given the better result over untuned Random Forest.

TABLE V
MEAN OF F1 SCORE AND STANDARD DEVIATION OF F1 SCORE

NO	Model Name	F1 Score Mean	F1 Score Standard Deviation
01	Tuned Random Forest	0.993067	0.003585
02	Untuned Random Forest	0.991926	0.004876

III. RESULT

We obtained mean of f1 score as 0.993067 using the Random Forest model with below mentioned hyper parameters on cross-validation with 0.003585 standard deviation.

Results of tuned hyper parameters,

- bootstrap: False
- max_depth: 55
- max_features: auto
- min_samples_leaf: 1
- min_samples_split: 2
- n_estimators: 1200

IV. CONCLUSION

To retain existing customers, Telecom providers need to know the reasons for churn, which can be realized through the knowledge extracted from Telecom data. In this paper, to predict whether or not a customer churn for unseen customers, we train six machine learning models which are Logistic Regression, Support Vector Machine(SVM), Linear SVM, K-Nearest Neighbor, Gaussian Naive Bayes, and Decision Tree. We can say that the Decision Tree is best in among six models. When trying ensemble model, Random Forest have given the better result over Decision Tree.