Customer Churn Prediction: Decision Tree Classifier Approach

Adylan Roaffa Ilmy 13516016
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
adylanrff@gmail.com

***Abstract*—Companies have been trying to find a way to increase the profit of their company. One of them is by analyzing the customer churn. This is why customer churn prediction is crucial. One of the customer churn model is Decision Tree Classifier that implements the tree structure. The implementation of the Decision Tree Classifier can be easily done by using Python, Jupyter Notebook, and the libraries. The accuracy of the customer churn model is really high, but it can be improved. The accuracy is also prone to modelling error, overfitting.**

***Keywords*—tree, decision tree classifier, customer churn, prediction**

I. Introduction

Companies have been trying to find a way to increase their profit and maximize their use of capitals. There are several aspects that is considered crucial to the longetivity of a company's life. One of them is the user or the customer aspect.

Customer or the user of the service is the key player to the business that a company run. By having a lot of customers, the profit will increase, and the money generated will be much larger. This is why customer is very important in a company's business. If there is no customer, there is no means at producing things that no one uses/buys.

Getting more new customers may look so profitable. The more the customer the more the money. But several studies has shown that maintaining a customer relationship is much more profitable than gaining new customers. This may happen because of the marketing cost is so big to gain new customers. The loss of old customers is proven to cause a lot of damage to the company.

With that being said, many companies have started to look for a way to know what the major cause of the customer churn is. A lot of research had being done by the companies to know and to predict the cause of the customer churn.

Recently, by the emergence of the machine learning algorithms that could make a prediction model based on data that runs efficiently, companies has been developing prediction models with the machine learning approach.

One of the popular and frequently used algorithm is the Decision Tree Classifier algorithm. This algorithm is proven to be efficient and accurate to predict and make a model to predict the cause of the customer churn.

By implementing the model, companies are expected to focus more on how to maintain their relationship with the customer by using the model that predicts the causes. This will also prevent the great loss caused by the customer stop subscribing to the certain products of the company.

II. THEORY

A. Tree

Tree is a connected-undirected graph that contains no simple circuit [1].

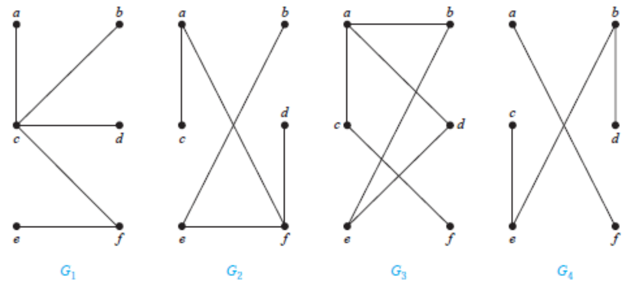

Figure 1 Example of Trees and Non-Trees [1]

From the figure above, we can conclude that G1 and G2 are trees, because G1 and G2 are connected and has no simple circuits, whereas G3 is not a tree because node *a, b, e, d,* construct a simple circuit. G4 is also not a tree it is not connected. This definition covers the meaning of a *free tree* that will differ in terms with *rooted tree*, which will be explained later in the following chapter.

B. Tree Properties

Suppose that $G = (V, E)$ is a simple undirected-graph and the number of its nodes is n. Then, the following terms are all equivalent [2]:

1. G is a tree.
2. Every pair of nodes in G is connected with a single edge.
3. G is connected and has $m = n-1$ number of edges.
4. G has no simple circuit and has $m = n-1$ number of edges.
5. G has no simple circuit and the addition of an edge to the graph will result to the formation of only one circuit.

6. G is connected and all of its edges are *bridges* (a *bridge* is an edge that if it is deleted from the graph, will separated the graph into two components).

All of the statement above can also be defined as another definition of a tree [1].

Various type in trees will have the properties that are stated in these theorems:

1. A tree with n vertices has n-1 edges.
2. A full m-ary tree with i internal vertices contains $n=mi + 1$ vertices.
3. A full m-ary tree with
   - n vertices has $i = (n - 1)/m$ internal vertices and $l = [(m - 1)\ n + 1]/\ m$ leaves,
   - i internal vertices has $n = mi + 1$ vertices and $l = (m - 1)\ i + 1$ leaves,
   - l leaves has $n = (ml - 1)/(m - 1)$ vertices and $i = (l - 1)/(m - 1)$ internal vertices.

C. Rooted Tree

A rooted tree is a tree which one vertex has been designated as the root, and every edge is directed away from the root[1].

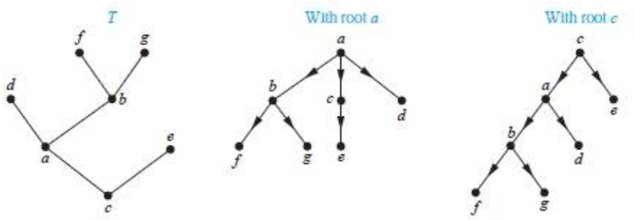

Figure 2 Example of Rooted Trees

From the figure above, the leftmost figure is a root tree where node *c* acts as the root. The figure in the center is a rooted tree where node a acts as the root. The rightmost figure is also a root tree, where node c act as the root.

D. Classification (machine learning)

Classification in machine learning is the ability of a system to predict categorical class labels, whether it is a discrete value or a nominal value. This process classifies data (construct a model) based on the training set and the values (class labels) in a classifying attribute and uses it in classifying new data[3].

Application of classification in the real world includes but not limited to:

- Credit approval
- Target marketing
- Medical diagnosis
- Fraud detection
- Customer churn rate

Types of classifiers that are commonly used in the real-world are:

1. Bayes
2. Nearest-neighbors.
3. Neural network.
4. Decision tree

In this paper we will only cover the definition and the application of decision tree classifier.

E. Decision Tree (Classifier)

A decision tree is a classifier expressed as a recursive partition of the instance space. The decisiont tree will form a *rooted tree,* which has a node that act as a *root* that has no incoming edges. All the other nodes have exactly one incoming edge [4].

A node with outgoing edges is called an internal or *test* node. The leaves of the nodes are called the *terminal* or decision nodes [4].

In a decision tree, each internal node splits the instance space into two or more sub-spaces according to a certain discrete function of the input attributes values. Each path from the root of a decision tree to one of its leaves can be transformed into a rule simply by conjoining the tests along the path to form the antecedent part, and taking the leaf's class prediction as the class value [4].

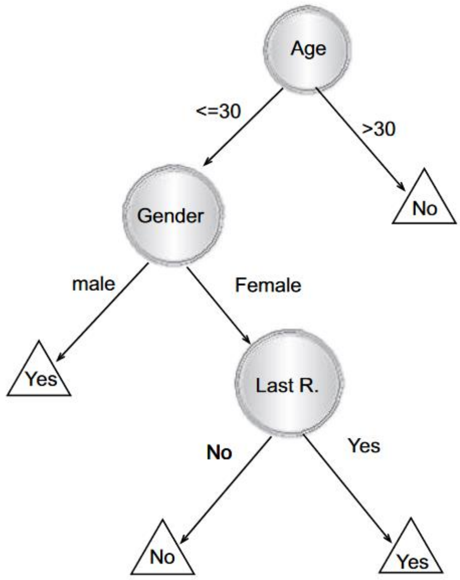

Figure 3 Example of Decision Tree Presenting Response to Direct Mailing

From the figure above, the leftmost leaf can be transformed to the rule: "if the customer age is less than or equal to 30, and if the gender of the customer is male, then the customer will respond to the mail". Another example is based on the decision tree on the Figure 3, if the customer age is more than 30, than the customer will not respond to the mail.

F. Decision Tree Impurity And Entropy

Impurity is when we have traces of one division of a class into the other. This can arise due to several factors [5]:

1. We run out of available features to divide the class upon.
2. We tolerate some percentage of impurity

Entropy is a degree of randomness of elements or in the other words, it is *measure of impurity*. Mathematically, the value of entropy can be calculated with the formula below [5]:

$$H = -\Sigma\, p(x) \log p(x)$$

G. Customer Churn

Customer churn is a condition where a customer (player, subscriber, user, etc.) ceases his or her relationship with a company. Online bussiness typically treat a customer as churned once a particular amount of time has elapsed since the customer's last interaction with the site or service. The full ost of customer churn includes both lost revenue and the marketing costs involved with replacing those customers with new ones.

The ability to predict that a particular customer is at a high risk of churning, represents a huge additional potential revenue source for every online business [6].

Besides the direct loss of revenue that results from a customer abandoning the business, the costs of initially acquiring that customer may not have already been covered by the customer's spending to date. (In other words, acquiring that customer may have actually been a losing investment.) Furthermore, it is always more difficult and expensive to acquire a new customer than it is to retain a current paying customer.

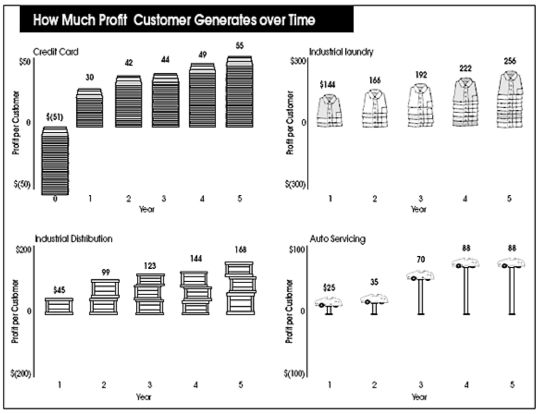

Figure 4 How Much Profit Customer Generates over Time [7]

Research done by Fred Reichheld [8] shows that a 5% increase in costumer retention produces more than 25% increase in profit. This happens because return customers tend to buy more from a company over time. As the customer do, the company's operating costs to serve them decline.

Len Markidan wrote four most common causes of customer churn [9]. They are:

1. Bad Customer Service
2. Bad Onboarding
3. Lack of Ongoing Customer Success
4. Natural Causes

III. IMPLEMENTATION

A. Introduction

This paper is going to show how to implement the Decision Tree Classifier to do a customer churn prediction of a company. The dataset is taken from *iainpardoe.com* [. The dataset contains 3334 customer data from Telco Company, a telecommunication company focusing in phone and mail services.

The dataset contains 21 features: account length, videomail Message, day minutes, evening minutes, night mins, international minutes, customer services calls, churn, international plan, videomail plan, day calls, day charge, evening calls, eve charge, night calls, night charge, international calls, international charge, state, area code, phone number.

The 'minutes' feature means how many minutes a customer use the service in the related time period. For example, the 'day minutes' feature means that it shows how many minutes in the daytime, the 'evening minutes' feature shows how many minutes in the eveningtime, and so on.

The goal of the implementation of the Decision Tree Classifier is to make a model that can predict the customer churn causes of the Telco Company.

B. Implementation

This paper is going to use Python and Jupyter Notebook to implement the Decision Tree Classifier. The implementation is separated into six steps: importing the libraries, getting the dataset, modify the dataset, make training and test sets, use the decision tree classifier, and build the tree visualization.

1. Import libraries

The program will use Numpy, Pandas, and Sci-kit Learn to implement the classifier.

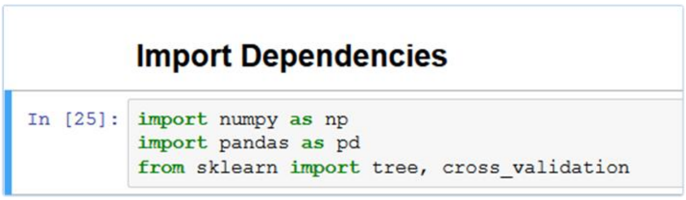

Figure 5 Importing Dependencies
Author's Document

Numpy provides an easy array modification and built-in array calculation that is ready to be used. This is important for the manipulation of the datasets.

Pandas provides a high-performance, easy to use data structures and data analysis tool. This library will be used to fetch the dataset and convert it into a data frame. Data frame is a two-dimensional data structure that is aligned as rows and columns.

Sci-kit Learn (sklearn) is the main tool in this program. The library contains many kinds of classifiers, including the Decision Tree Classifier. The library also provide a tool to cross validate the data that will be explained in later section.

2. Getting the dataset

Using the Pandas built-in function *read_csv*, we are able to get the data from a *CSV* file and convert it into the Pandas Data Frame.

4. Make training and test sets

The program will use cross validation to evaluate the model. In this program, X_train and y_train are the part of the data that will be used to train the model, whereas X_test and y_test will be used to test the trained model.

Training and test set

```
X = np.array(df.drop(['Churn'],1))
y = np.array(df['Churn'])

X_train, X_test, y_train, y_test = cross_validation.train_test_split(X,y,test_size=0.25)
```

Figure 8 Making the training and test set
Author's Document

Get dataset

```
In [26]: df = pd.read_csv('F:\Documents\Adylan\'s\ITB\KULIAH\TINGKAT 2\Matematika Diskrit\Churn-csv.csv')
         df.head(n=5)
Out[26]:
```

	Account Length	VMail Message	Day Mins	Eve Mins	Night Mins	Intl Mins	CustServ Calls	Churn	Int'l Plan	VMail Plan	...	Day Charge	Eve Calls	Eve Charge	Night Calls	Night Charge	Intl Calls	Intl Charge	State	Area Code
0	128	25	265.1	197.4	244.7	10.0	1	0	0	1	...	45.07	99	16.78	91	11.01	3	2.70	KS	415
1	107	26	161.6	195.5	254.4	13.7	1	0	0	1	...	27.47	103	16.62	103	11.45	3	3.70	OH	415
2	137	0	243.4	121.2	162.6	12.2	0	0	0	0	...	41.38	110	10.30	104	7.32	5	3.29	NJ	415
3	84	0	299.4	61.9	196.9	6.6	2	0	1	0	...	50.90	88	5.26	89	8.86	7	1.78	OH	408
4	75	0	166.7	148.3	186.9	10.1	3	0	1	0	...	28.34	122	12.61	121	8.41	3	2.73	OK	415

Figure 6 Getting the dataset
Author's Document

The *head* function is used to show the *n*-th elements of the data from the top. This is important to make sure that the data that we are getting is the correct one or not.

3. Modify the dataset

To improve the accuracy of the classifier, the irrelevant features that does not support the models should be removed from the data.

Modify the dataset

```
# Drop the irrelevant columns for prediction
df.drop('Account Length',1,inplace=True)
df.drop('State',1,inplace=True)
df.drop('Area Code',1,inplace=True)
df.drop('Phone',1,inplace=True)

df.head(n=5)
```

Figure 7 Removing irrelevant features
Author's Document

As we can see from *Figure 6*, there are several features that will not be good as a parameter to our decision tree. They are the account length, state, area code, and phone numbers. This should be removed from the data as they act only as an identifier, not a feature that will affect the model.

In this program, 25% of the data will be used as the test data, and the rest are going to be the training data.

5. Use the decision tree classifier

This snippet of the program shows how to use the Decision Tree Classifier using the Sci-kit Learn library.

Do prediction using Decision Tree Classifier

```
# Use the decision tree Classifier
clf = tree.DecisionTreeClassifier()

accuracy_sum = 0
accuracies = []
for i in range(100):
    clf.fit(X_train,y_train)
    accuracy = clf.score(X_test,y_test)
    accuracies.append(accuracy)
    accuracy_sum+=accuracy

min_accuracy = min(accuracies)
max_accuracy = max(accuracies)
avg_accuracy = accuracy_sum/len(accuracies)

print "Average accuracy : " , avg_accuracy
print "Minimum accuracy : " , min_accuracy
print "Maximum accuracy : " , max_accuracy
```

```
Average accuracy :  0.923896882494
Minimum accuracy :  0.914868105516
Maximum accuracy :  0.931654676259
```

Figure 9 Prediction score
Author's Document

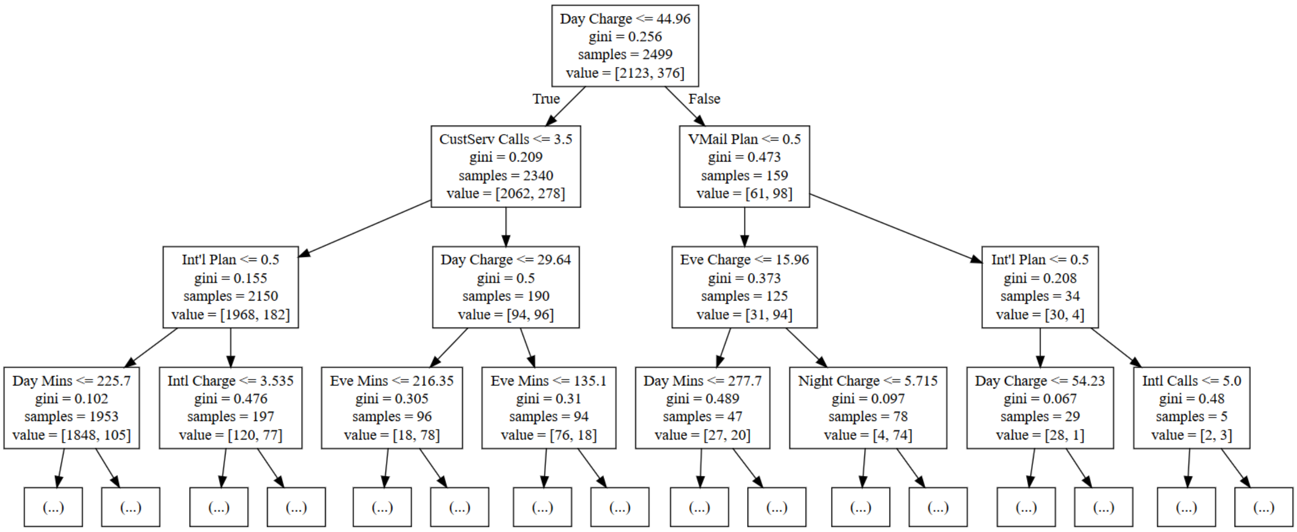

Figure 11 Decision Tree Visualization
Author's Document

By using the Decision Tree Classifier, we are able to get the average of 92% of accuracy, a minimum of 91% accuracy, and the maximum of 93% accuracy. This result is collected from 100 training session. The accuracy represents how well the model guess the output (y_test) from the input (X_test) from the current model denoted by *clf*. The sum of the accuracies then will be divided by number of trainings to get the average accuracy of the current set of training sessions.

6. Building the tree visualization

By using the *export_graphviz* function, the program can create a visualization of a tree that the classifier is using. The visualization is shown by the *Figure 11*.

Make tree visualization

```
header = np.delete(df.columns.values,6)
tree.export_graphviz(clf,out_file='churn-dtc.dot',max_depth=3, feature_names= header)
```

Figure 10 Making tree visualization using graph_viz

Author's Document

Note that this is the simplified version of the decision tree, the visualizer is set to only showing the first three levels of the tree. The original tree is about 20 levels high.

C. Analysis

What the program do basically is splitting the data by a certain features to get the least possible *gini* index. *Gini* index is the degree of randomness of a data. The maximum value of the *gini* index is 0.5. It means the data has an equal number of members of each of the labels. This process is repeated until the best condition is fulfilled. That is when the gini index reaches zero. In the real application of the classifier, it is often too hard to split the data until the gini index is zero. So what the program do is to make the least possible *gini* index, and stops until a certain *treshold.*

From the *Figure 9*, the prediction accuracy from the model is very high, reaching a minimum of 91% of accuracy. This shows that Decision Tree Classifier is showing a good performance on predicting the customer churn given by the datasets from the Telco Company.

Although 90% looks good enough, the company is expecting to go for a much better accuracy, 97% at the minimum. By using a more accurate model, they can maximize the profit and maximize the use of their money towards something else, rather than spending their time thinking about how to prevent the churn of customers.

To increase the accuracy of the model, we should look deeper at the data, and conclude some informations out of it. In this program, we are examining raw data, without any deeper look at the data itself.

After looking at the data and extracting informations, we can make some custom features to be added to the model. This feature is expected to increase the accuracy of the model.

Examining a bigger portion of data will also increase the accuracy of the data, and make a more accurate generalization of the problem. The bigger the data, the more accurate the model is.

Note that the very high initial accuracy shown at the *Figure 9* could be the result of a modelling error called *overfitting. Overfitting* is the state where the model is too closely fit to a limited set of data, meaning that it will only be accurate to the current data, namely the training data that is used. This should be avoided to make a better generalization towards the problem.

IV. CONCLUSION

Tree can be used to help companies to make customer churn predictions. The model that is used is the Decision Tree Classifier. Python and Jupyter Notebook can be used to make an easy implementation of the Decision Tree Classifier. To make customer churn predictions, we need data. The data is used to make a predicting model by splitting the data into training and test data. The accuracy of the data is quiet high, but it can be improved by increasing the number of data used, and looking deeper into the data. The model should be aware of modelling error such as *overfitting*.

V. ACKNOWLEDGEMENT

First of all, I would like to express my deepest gratitude to our Almighty God, Allah SWT. Because with His mercy and graces, I am able to finish this paper without any difficulty whatsoever.

Second, I would like to thank my parents for the supports, and all the uncountable things that they have given to me.

I also thank Mr. Rinaldi for the Silver Queen you gave me at that time (sorry I ate it really fast I love chocolate) and the guidance for the Discrete Mathematics at the class and how you motivates us to pass the final *projects quizzes* , and final tests through time.

I would also like to thank my friends that has supported me and being the reason why I worked so hard.

REFERENCES

[1] K. H. Rosen, Discrete Mathematics and Its Application (Seventh Edition).New York: McGraw-Hill, 2012, ch. 11

[2] Munir, Rinaldi. 2003. Matematika Diskrit Edisi Keempat. Bandung: Penerbit Informatika.

[3] Lexis Nexis

[4] Rokach, Lior. *Decision Trees* [PDF Document] Retrieved October 28, 2017 from http://www.ise.bgu.ac.il/faculty/liorr/hbchap9.pdf

[5] Patel, Savan (2017). *Decision Tree Classifier – Theory* Retrieved October 26, 2017 *from https://medium.com/machine-learning-101/chapter-3-decision-trees-theory-e7398adac567*

[6] *Customer Churn (*n.d). Retrieved November 1, 2017 from https://www.optimove.com/learning-center/customer-churn-prediction-and-prevention

[7] Reichheld, Frederick (1990). *Zero Defections: Quality Comes to Services*. Retrieved November 1, 2017 from https://hbr.org/1990/09/zero-defections-quality-comes-to-services

[8] Reichheld, Frederick (2001). *Prescription for cutting costs* Retrieved November 1, 2017 from http://www.bain.com/Images/BB_Prescription_cutting_costs.pdf

[9] Markidan, Len (2015). *Why Customer Churn Happens, And What You Can Do About It.* Retrieved November 1, 2017 from https://www.groovehq.com/support/reduce-customer-churn

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 3 Desember 2017

Adylan Roaffa Ilmy, 13516016