ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020

Audience of One: Machine Learning based Adaptive Graphical User Interface

Gungun Sharma [1], Pradip Mukhopadhyay[2], Ravi Lakshmanan[3]

P.G. Student, Department of ISM, Indira Gandhi Delhi technical University for Women, Delhi, India[1]

Senior Engineer, NetApp, Bangalore, India[2]

Member Technical staff, NetApp, Bangalore, India [3]

ABSTRACT: In past year, it was the difficult task to recommend data to a person based on his usage or behaviour. But now it is possible with help of machine learning. Machine learning algorithms learns from historical data and predicts the user's behaviour like novice, intermediate or advanced. In machine learning, various approaches have used like decision tree, random forest etc. In machine learning, the random forest is an ensemble learning method which is used for both classification and regression problem. Random forest generates the multiple decision tree and returns the label by using the 'majority vote' method. Random forest is used to avoid the overfitting as well as improves the accuracy of machine learning model. Machine learning algorithms deal only with the numerical data. So, we had to convert text data into numerical data to build the desired random forest which achieved the accuracy of 90%.

KEYWORDS: Random Forest, Decision tree, Feature Selection, Text dataset

I. INTRODUCTION

It is becoming difficult to personalize the user's requirements as consumers are growing. To determine the user's requirements and provide the accurate result is a difficult task. Consider an example- a user watches a web series on youtube. Every time user has to go to the search bar and type the series name and search for next episode he wants to see. Hence, this is a time-consuming process. But machine learning algorithms help to overcomes this type of problem. It learns from the user's history and provides the next episode as well as recommend the additional video of similar types [1].

Machine learning is a subset of artificial intelligence that predicts the outcome through self-learning. Machine learning does the work in the same way as human do. In traditional programming, programmer needs to write down all the rules that define the behaviour of system. As system continues to grow, more rule is added, thus increasing the system complexity and making it difficult to maintain. On the other hand, with machine learning, system itself make the rules based on its learning from given input and output.

One of the important features of machine learning is **feature engineering**[2]. Machine learning algorithms can read numerical variables only. Feature extraction is a way to read the categorical data. So, it is important to convert the categorical data into numerical data. Categorical variable can be encoded using functions like LabelEncoder, OneHotEncoding, DictVectorizer and pandas get_dummies. In python, LabelEncoder can be performed via Sklearn library. It encodes the label between zero to n-1, where n is the number of distinct labels and it assigns the same value for every repeated label.

Random forest is one of the important machine learning algorithms. Random forest is made up of multiple decision tree [3]. It is a bootstrap aggregating, also known as bagging, which generates the multiple model from single dataset and use the majority vote for prediction. Consider an example- suppose Monica wants to watch a movie. So, she asks her friends to recommend her a movie. To know about her taste of movies, each friend starts asking her some question

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020

like her favourite actor/actress, favourite director, last movie she watched and many more. Based on the answers from Monica, they start predicting the movie that she may like, and built the decision tree for the same. Now, Monica will go through all the suggestions she get. Each repeated suggestion will increase the likelihood of her watching that movie, leading to the conclusion for the movie that she should watch. In this scenario, all her friends are the tree that together makes the forest. Such a forest is called random forest.

To build the multiple decision tree, so many algorithms exist such as entropy, information gain, gain ratio, Gini index etc. There are many machine learning applications that use random forest algorithm for decision making. Use cases can vary from banking sector for identification of the good and bad customer, analysis of the patient's medical report in hospital to image and voice classification.

II. METHODOLOGY

Following are the step for making the random forest:
Step 1: Take the dataset

Step 2: Build a random forest/ decision tree: -

1. Randomly select the d feature from D. where D is an original dataset and d<<D.
2. Randomly select the value to categories each attribute
3. Among the d feature, select the one column as a root node by using the attribute selection method.
4. Split the dataset according to the value of root node and remove that column from the dataset.
5. Repeat the step 1 to 4 until all variables has been reached.
6. Repeat step 1 to 5 to make n number of decision tree to build the random forest.

Step 3: Predict the behaviour of the user.

III. EXPERIMENTAL RESULT WITH FIGURE/TABLE

Here, we have a large dataset which consists of numerical data as well as categorical data. Here, In the table 1, we have shown a snapshot of original dataset:

counterkey	Count
color.red	10
color.red	30
color.red.salmon	14
color.red.salmon	29
color.pink.ruby.ultra	45
color.pink.ruby.ultra	89
color.pink.ruby.ultra	15
color.blue.denim.sky.carolina	20
color.blue.denim.sky.carolina	18
color.blue.denim.sky.carolina	10
color.blue.denim.sky.carolina	8
color.green.pine.kelly	50
color.green.pine.kelly	30
color.green.pine.kelly	25
color.gray.mink.smoke	12
color.brown.umber	15
color.brown.chocolate.syrup	20
color.blue.navy.steel	35

Table 1. Description of dataset

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020

1. ***Combine the common row:*** Replace the dot with space and combine the all similar rows.

Counterkey	Counter_count
color red	40
color red salmon	43
color pink ruby ultra	149
color blue denim sky carolina	56
color green pine Kelly	105
color gray mink smoke	12
color brown umber	15
color brown chocolate syrup	20
color blue navy steel	35

2. ***Splitting the dataset:*** Split the Counterkey(c) attribute into *<c1, c2, c3,……,cn>*

0	1	2	3	4
color	red	None	None	None
color	red	salmon	None	None
color	pink	ruby	ultra	None
color	blue	denim	sky	Carolina
color	green	pine	Kelly	None
color	gray	mink	smoke	None
color	brown	umber	None	None
color	brown	chocolate	syrup	None
color	blue	navy	steel	None

3. ***Encoding of dataset:*** Use the LabelEncoder function to encode the categorical data into numerical data.

	0	1	2	3	4
0	0	12	N	N	N
1	0	12	8	N	N
2	0	15	5	19	N
3	0	29	16	28	50
4	0	40	20	18	N
5	0	10	31	13	N
6	0	11	32	N	N
7	0	11	30	45	N
8	0	29	67	65	N

4. ***Classification of dataset:*** Classify the dataset into three class labels: Novice, Intermediate, Advanced

	col_0	col_1	col_2	col_3	col_4	class_name
0	0	12	N	N	N	novice
1	0	12	8	N	N	novice
2	0	15	5	19	N	intermediate
3	0	29	16	28	50	advanced
4	0	40	20	18	N	advanced
5	0	10	32	13	N	novice
6	0	11	32	N	N	novice
7	0	11	30	45	N	novice
8	0	29	67	65	N	advanced

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020

Here, column 0 to 4 are predictors variable and class_name is target variable.

5. ***Plot the single decision tree:*** Divide the dataset into training set and test set in the ratio of 80:20. Next, apply the fit function which adjusts the weights according to data values in order to improve the accuracy. Here, we have achieved the accuracy of 90%. Following are the steps to plot single decision tree:

- **Step 5.1) Randomly select the value to categories each attribute:**

col_1	**col_2**	**col_3**	**col_4**
>=11	>=16	>=13	>=50
<11	<16	<13	<50

- **Step 5.2) Calculate the Gini index:**

Calculate the Gini index of each attribute and pick the attribute with smallest gini index [4].

- Feature 1 has 8 elements out of 9 for >=11 and 1 element out of 9 with <15 value.

For >=11 and class == novice :4/8
For >=11 and class == intermediate :1/8
For >=11 and class == advanced :3/8

- Gini (0,1,3) = 1- $(4/8)^2$ + $(1/8)^2$ + $(3/8)^2$ = 0.593

For <11 and class == novice :1/1
For >=11 and class == intermediate:0/1
For >=11 and class == advanced: 0/1

- Gini (5,0,0) = 1- $(1/1)^2$ + $(0/1)^2$ + $(0/1)^2$ = 0

Gini (target, 1) = 8/9 *0.593 + 1/9 * 0=0.527
Similarly, **Gini (target, 2)** = 0.3827
Gini (target, 3) = 0.2400
Gini (target, 4) = 0.4722

Therefore, attribute 3 is selected as a root node because attribute has a lowest gini index. Then decision tree is oks like:

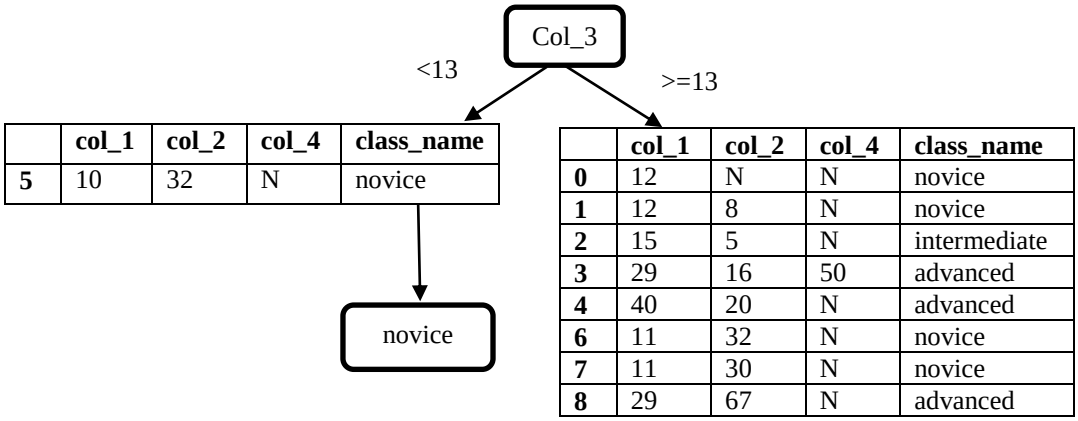
ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020



	col_1	col_2	col_4	class_name
5	10	32	N	novice

	col_1	col_2	col_4	class_name
0	12	N	N	novice
1	12	8	N	novice
2	15	5	N	intermediate
3	29	16	50	advanced
4	40	20	N	advanced
6	11	32	N	novice
7	11	30	N	novice
8	29	67	N	advanced

Fig 1: Decision Tree for the data specified

For col_3 <13 the target feature of remaining dataset is novice. Hence, select the novice as a leaf node.
For col_3>=13, an impurity of variables is existing. So,calculate the gini index for remaining dataset.

- **Step 5.3)** Calculate the Gini index for remain dataset:

  - Feature 1 has 8 elements out of 8 for >=11 and 0 elements out of 8 with <11 value.
    For >=11 and class == novice :4/8
    For >=11 and class == intermediate :1/8
    For >=11 and class == advanced :3/8

    - Gini (0,1,3) = 1- $(4/8)^2$ + $(1/8)^2$ + $(3/8)^2$ = 0.593

    For <11 and class == novice :0/0
    For >=11 and class == intermediate:0/0
    For >=11 and class == advanced: 0/0

    - Gini (5,0,0) = 1- $(1/1)^2$ + $(0/0)^2$ + $(0/0)^2$ = 0

  Gini (target, 1) = 8/8 *0.593 + 1/8 * 0=0.527
  Similarly, Gini (target, 2) = 1.047,
  Gini (target, 4) = 0.340

Therefore, attribute 4 is selected as an internal node because attribute has a lowest gini index.
For col_4 <50, the target feature of remaining dataset is advanced. Hence, select this as a leaf node.
For col_4>=50, an impurity of variables is existing. So, calculate the gini index for remaining dataset.

- **Step 5.4)** Repeat step 1 to 3 until all variables has been reached.
  Hence, the final decision tree is given below:

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020

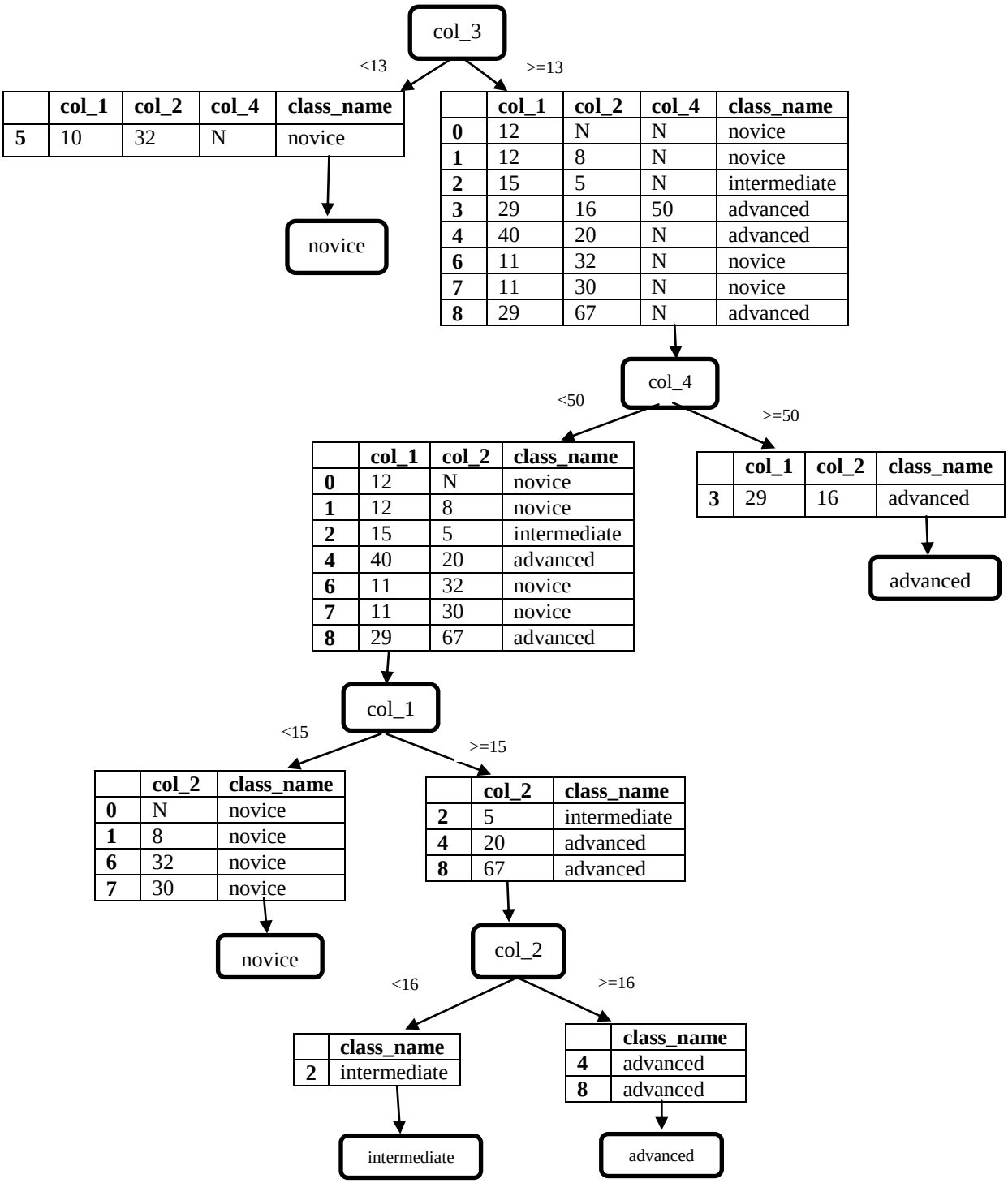


Fig 2: Single Decision Tree

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020

IV. CONCLUSION

The In this paper, we have classified the user into three categories and provide the accurate data. We have presented the random forest of categorical dataset. Here, Gini index is used as a best attribute selection method and label encoder to encode the categorical variable into numerical variable. Bootstrap aggregation technique is used to generates the multiple sampling dataset with the replacement that made the multiple decision tree. Majority method is used for predicting output resulting in 90% accuracy of random forest.

DISCLAIMER

The authors contributed to this article in their personal capacity. The views expressed are their own and do not necessarily represent the views of NetApp.

REFERENCES

1. Nixon, Lyndon, Krzysztof Ciesielski, and Basil Philipp. "AI for audience prediction and profiling to power innovative TV content recommendation services." *Proceedings of the 1st International Workshop on AI for Smart TV Content Production, Access and Delivery*. 2019.
2. Yao, Ying, et al. "Classification of fatigued and drunk driving based on decision tree methods: a simulator study." *International journal of environmental research and public health* 16.11 (2019): 1935.
3. Lee, Sunmin, et al. "Spatial prediction of flood susceptibility using random-forest and boosted-tree models in Seoul metropolitan city, Korea." *Geomatics, Natural Hazards and Risk* 8.2 (2017): 1185-1203.
4. Gupta, Bhumika, et al. "Analysis of various decision tree algorithms for classification in data mining." *Int. J. Comput. Appl* 163.8 (2017): 15-19.
5. Shaikhina, Torgyn, et al. "Decision tree and random forest models for outcome prediction in antibody incompatible kidney transplantation." *Biomedical Signal Processing and Control* 52 (2019): 456-462.
6. Amiri, Saeid, Bertrand S. Clarke, and Jennifer L. Clarke. "Clustering categorical data via ensembling dissimilarity matrices." *Journal of Computational and Graphical Statistics* 27.1 (2018): 195-208.
7. Xu, Baoxun, et al. "An Improved Random Forest Classifier for Text Categorization." *JCP* 7.12 (2012): 2913-2920.