

OTA ( Online Travel Agency )

Submitted by:

Sourov Sahoo

**ACKNOWLEDGMENT**

**Guidance personals:-**

1. Fliprobo [**Tushar Saraswat**](https://www.flipnwork.com/index.php/team_members/view/16).

**Literature study:-**

1. Previous Classification problem statements.

**INTRODUCTION**

* Business Problem Framing

The dataset is from an online travel agency that tracks their users’ browsing history from search to bookings and payments.

This dataset has 3 possible outcome variables: clicked, booking\_value and booked. Select one of those variables to model and train a machine learning model of your choice.

**Solution is**:-

1-By using sklearn (XGBClassifier)we have build a machine learning model that will predict whether the customer booked or notbooked.

2-Wether the customer booked(1) or notbooked(0) can be predicted using the model above.

Conceptual Background of the Domain Problem

Data is imbalanced...

Large dataset

* Motivation for the Problem Undertaken

“Revenue generation”

Because 'booked' or 'notbooked' is basically the revenue part.

And that is what i am interested in, and that is why i also choose 'booked' as my dependent(target)feature.

**Analytical Problem Framing**

* Mathematical/ Analytical Modelling of the Problem

Importing required libraries------------------------------------------------------------------------

a. Load the dataset into Python and identify the type of the dataset features and report them

# General data checking(checking shape and missing value)

b. Perform summary statistics and explain what issues these statistics reveal.

#df.describe

#Manual EDA

#AUTOMATED EDA by sweetviz library

#AUTOMATED EDA by dtale library

c. Perform exploratory analysis to identify any collinearities and explain which issues collinearity causes.

PREPROCESSING-------------------------------------------------------------------------------------------------------------

1.Missing value treatment(With logic)

2.Feature engineering--->>feature('timestamp')-->8 new features generated.

# Dropping feature('timestamp') as we have extracted our features

3.VIF technique for multi collinearity implementation

# Dropping features as we have extracted and made our new VIF feature.

#df.corr()--->collinerity check.

#We can manually delete the collinear feature.

#But I have implemented the VIF technique to remove multicollinearity.

-------------------------------------------------------------------------------------------------------------

4. -ve value treatment technique.

5. Adding 21000 to all numerical feature to cancel out the max -ve value in the dataset (), then i will take log().This will not affect the distribution of the dataset.

----------------------------------------------------------------------------------------------------------------------------

6. Segregating target('booked') and independent features

-----------------------------------------------------------------------------------------------------

7) Over sampling (RandomOverSampler) as it is faster than SMOTETomek.

8) Removing skewness by power transformation technique.

9) Applying StandardScaler.

---------------------------------------------------------------------------------------------------

#Saving the cleaned data for future use

-------------------------------------------------------------------------------------------------------------------------------

\*\*PCA(NOT applied as the scree plot did not go constat at any point of time WRT all the feature.)

---------------------------------------------------MOdel BUilding------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Best performing (XGBoostClassifier):-

10) Applying XGBoostClassifier here, And checking scores.

11) GridsearchCV on XGBoostClassifier(masked here as this is just a review model,and training takes lot of time)#Trained for12hours but no result.

12) Cross validating the model.(masked here as this is just a review model,and training takes lot of time)

13) Saving the final model in joblib-->'OTA\_review2.obj'

14) Loading the model-->'OTA\_review2.obj'

15) testing the model on test data.

16) Saving final predictions in 'final\_predictions\_OTA\_review2\_xgb .csv' format

17) Conclusion

* Data Sources and their formats

The dataset is from an online travel agency that tracks their users’ browsing history from search to bookings and payments. The details for the data fields are given in “Data Dictionary.xlsx”. And the given dataset is “train\_small.csv” …

The dataset is quite varying, “notbooked(0)”class have few values. (imbalanced dataset)

The dataset train\_small.csv folder contains ROWS= 2380557

COLUMNS=54

**Preprocess technique**

1.Missing value treatment(With logic)

2.Feature engineering--->>feature('timestamp')-->8 new features generated.

# Dropping feature('timestamp') as we have extracted our features

3.VIF technique for multi collinearity implementation

# Dropping features as we have extracted and made our new VIF feature.

#df.corr()--->collinerity check.

#We can manually delete the collinear feature.

#But I have implemented the VIF technique to remove multicollinearity.

-------------------------------------------------------------------------------------------------------------

4. -ve value treatment technique.

5. Adding 21000 to all numerical feature to cancel out the max -ve value in the dataset (), then i will take log().This will not affect the distribution of the dataset.

----------------------------------------------------------------------------------------------------------------------------

6. Segregating target('booked') and independent features

-----------------------------------------------------------------------------------------------------

7) Over sampling (RandomOverSampler) as it is faster than SMOTETomek.

8) Removing skewness by power transformation technique.

9) Applying StandardScaler.

---------------------------------------------------------------------------------------------------

#Saving the cleaned data for future use

-------------------------------------------------------------------------------------------------------------------------------

\*\*PCA(NOT applied as the scree plot did not go constant at any point of time WRT all the feature.)

* Data Inputs- Logic- Output Relationships

The independent data input(x) is shape(4628338, 58 ).

The dependent data input(y) is booked feature which contains the data is a hotel is booked or notbooked.

The model is 100% recall and ROC for test data .

* State the set of assumptions (if any) related to the problem under consideration

No Assumptions made .

Hardware and Software Requirements and Tools Used

**Hardware**-64bit, 12GB RAM, 240GB SSD.

**Software-**Excel, Anaconda,jupyter notebook,python 3.6

Libraries used:-

1. numpy

2. pandas

3. matplotlib

4. sklearn

5. xgboost

6.joblib

**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)

Data balancing technique may be used for better results.

* Key Metrics for success in solving problem under consideration.

This is a binary class classification problem,we could use the following 2 popular metrics:

1:Recall

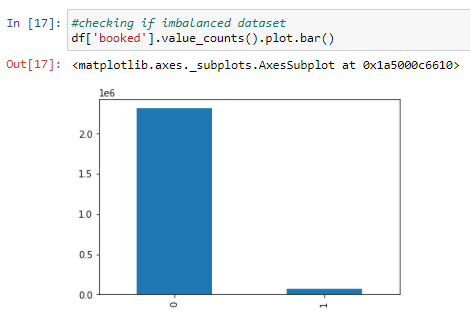
2:AUC-ROC(Area under the reciver operating charecteristic curve)

The ROC\_AUC is the area under the curve when plotting the(nomalized)true positive rate(x-axis) and the false positive rate(y-axis)

Our main metric here would be Recall value,while AUC-ROC score would take care of how well predicted probabilities are able to differantiate between the two classes.

**Visualizations**

**Training data Accuracy:-**



**Pre-processing:-**

1.Missing value treatment(With logic)

2.Feature engineering--->>feature('timestamp')-->8 new features generated.

# Dropping feature('timestamp') as we have extracted our features

----------------------------------------------------------------------------------------------------------

3.VIF technique for multi collinearity implementation

# Dropping features as we have extracted and made our new VIF feature.

#df.corr()--->collinerity check.

#We can manually delete the collinear feature.

#But I have implemented the VIF technique to remove multicollinearity.

-------------------------------------------------------------------------------------------------------------

4. -ve value treatment technique.

5. Adding 21000 to all numerical feature to cancel out the max -ve value in the dataset (), then i will take log().This will not affect the distribution of the dataset.

----------------------------------------------------------------------------------------------------------------------------

6. Segregating target('booked') and independent features

-----------------------------------------------------------------------------------------------------

7) Over sampling (RandomOverSampler) as it is faster than SMOTETomek.

8) Removing skewness by power transformation technique.

9) Applying StandardScaler.

---------------------------------------------------------------------------------------------------

#Saving the cleaned data for future use

-------------------------------------------------------------------------------------------------------------------------------

\*\*PCA(NOT applied as the scree plot did not go constat at any point of time WRT all the feature.)

---------------------------------------------------MOdel BUilding------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Best performing (XGBoostClassifier):-

10) Applying XGBoostClassifier here, And checking scores.

11) GridsearchCV on XGBoostClassifier(masked here as this is just a review model,and training takes lot of time)#Trained for12hours but no result.

12) Cross validating the model.(masked here as this is just a review model,and training takes lot of time)

13) Saving the final model in joblib-->'OTA\_review2.obj'

14) Loading the model-->'OTA\_review2.obj'

15) testing the model on test data.

16) Saving final predictions in 'final\_predictions\_OTA\_review2\_xgb .csv' format

17) Conclusion

**CONCLUSION**

**1-Wether the customer booked(1) or notbooked(0) can be predicted using the model above.**

**2-By using sklearn(XGBClassifier)we have build a machine learning model that will predict wether the customer booked or notbooked.**

* Key Findings and Conclusions of the Study

Imbalanced dataset

From analysis:------->

The "listing\_stars"==3 has the max. clicks a listing received. For listing\_stars==3 has the max. conversion rates on a listing received.

* Learning Outcomes of the Study in respect of Data Science

Good marketing strategy can be implemented with the features given and revenue can be generated with minimum manual effort.

* **Limitations of this work. And Scope for Future Work**

Computational complexity:

GridsearchCV ,cross\_val\_score

Cross\_val\_score takes too much time.

Hardware problem:-

Need more powerful system.

My maximum time went in GridsearchCV and model building which is just opposite for a data scientist working hours.

With a upgraded system next time I will be concentrating and spending more time in EDA and data analysis.

**Problems I faced during project**

Unbalanced data set.

GridsearchCV is taking too long.

I could not play around the values while hypertuning due to low computational power.

Maximum time went in computing rather than analysis of the data.

**Future works**

Try implementing gridsearchcv and and cross\_val\_score.

Try different combinations for GridsearchCV hypertuning.