Understanding Online Shoppers' Purchase Intentions using Data Analytics

[1]Dr.V.Anantha Krishna, , [2]Dr.MD.Nazmoddin,[3]P.Avinash, [4]Dr. A.Nagarjuna Reddy,

[1,2,3,4] Professors, Department of Computer Science and Engineering
[1,2,3,4] Sridevi Women's Engineering College Telangana, Hyderabad India
[4]krishnaananthav@gmail.com, [2]najmuddinmohd4u@gmail.com,

[3]avinashjntuh@gmail.com, [4]anr304@gmail.com,

Abstract— With the advancement in technology, e-commerce or online shopping has gained popularity in comparison to traditional shopping, which has made it difficult to understand customer intentions. In our research, we plan to construct a real-time prediction machine learning system for the online shopping environment to predict the purchase intentions of prospective buyers through various analytical models. We have classified the users based on their revenue-generating propensity, and have applied alternative models including logistic regression, Support Vector Machine, Ada boost, Voting Classifier to predict their intention to purchase. The class imbalance was one of the major issues with the data, we tried Random oversampling, SMOTE, and SMOTEEN techniques to counter this. Minority class prediction being the objective, the models that overfit were removed. The best model is selected based on the F1 score, cross-validation accuracy, and cross-validation ROC AUC.

From the results of this research, the most important features came out to be Page Values, Product related page duration, exit rates, traffic type, among others. The online retail companies will do better to convert their customers if they monitor the insights closely.

Keywords— Overfitting, voting classifier, minority class prediction

I. Introduction

E-commerce, the activity of buying and selling products online, is one of the many fields revolutionized by data science. More than 10 years ago, e-commerce was at 5.1% of total retail purchases. E-commerce now accounts for 16% of the total retail sale in the U.S. Consumers spent $601.75 billion online in 2019, a 14.9% jump compared to 2018, companies like Amazon, Flipkart and Myntra have created retail empires off e-commerce websites. With e-commerce becoming more and more prevalent in today's economy businesses within this sector need to understand what factors influence a visitor to transform into a purchaser. To increase conversion, e-commerce companies and researchers have devoted considerable efforts in analyzing the behavior of site visitors.

For this project, we used the Online Shoppers Purchasing Intention Dataset, obtained from the UCI Machine Learning repository [1]. The goal is to build a predictive machine learning model that could categorize users into revenue-generating and non-revenue generating, based on their behavior while navigating a website. The intention is to develop a model using selected variables for predicting the purchasing intention of users. The model can help e-commerce businesses identify customers who are more likely to complete transactions and adjust marketing strategies accordingly.

II. Literature Review

There are multiple studies to understand the intention of the visitors of ecommerce sites. In a study [2] the prediction was done using the classification algorithms and ensemble methods on the data. After comparison using various evaluation metrics Random Forest was concluded to be the best method based on the highest accuracy in test data. In [3] Moe tried to categorize the customers based on their behavior on the online shop website. The categories were "Direct buying", "Knowledge building", "Search", "Shallow" where direct buying was defined as the users who directly visit a page and purchase an item. On the other extreme, users who leave the website after visiting two pages are categorized as shallow users. It had the clickstream data of the users and their intention was classified based on the belief that the user activities on the shopping website depend on their intention. A set of features of the user activity was collected and fed to the K means clustering to categorize the users. In another study [4] the prediction of purchase of any product at the end of the session is based on product popularity and temporal data. In this, it had been found that the visitors with a profile on the website, are easy to predict as their previous history is available. But it is hard to determine the intention of any new customer. The temporal visiting data is used along with the products' popularity to predict their intention. In another incitation [5], the discussion was on dealing with the imbalance dataset using the Synthetic Minority Over-sampling Technique for Nominal and Continuous features (SMOTE-NC) technique.

III. Methodology

We have designed our study to predict the purchasing intentions of the visitors, by dividing them in revenue and non-revenue generating customers and eventually put forward a reasonable marketing strategy for the company to increase the number of successful online purchases. The algorithm is represented in Fig 1.

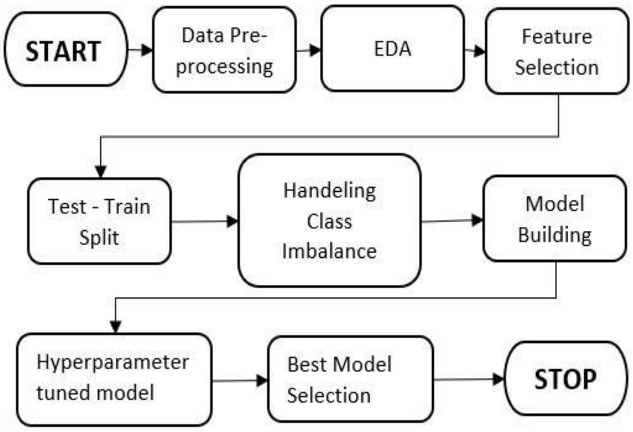


Fig.1. Study Framework

IV. Dataset

A. Dataset Description & Processing

The dataset [1] is obtained from the open UCI Machine Learning repository. It consists of 12330 records, each containing metrics of web visits of a user within a one-year timeframe. 85.4% (10422) of the customers did not complete the transaction. Completed transactions take up only 15.5% (1908) of the dataset.

In the dataset, Administrative, Administrative Duration, Informational, Informational Duration, Product Related, and Product Related Duration represents different types of pages visited by the visitor in a session and total time spent in each of these page categories is also available. The Bounce Rate, Exit Rate, and Page Value features represent the metrics which has been measured by Google Analytics. The bounce rate feature indicates the percentage of visitors who entered the site from a particular pages at the site and left the website without any activity. The exit rate of a page is the percentage of users who have their last session on that page. The page value indicates the average value of a page that visitors have visited before purchasing any product. The special day feature indicates approach of any special occasion when the users are more likely to purchase a product. The range of value is between 0 and 1. There are 8 categorical variables namely operating system, browser, region, traffic type, visitor type, weekend, month, revenue. The revenue value indicates whether a visitor purchased an item or not. This is the response variable.

As part of data preprocessing for our dataset, we did relevant imputations for NULL (NA) values and treat outliers as well as negative values, we performed one-hot encoding of categorical attributes and label encoding of response variable, Revenue.

V. Modelling Approach

A. Feature Selection

After all the preprocessing and feature engineering we were left with around 50 features. Using the correlation matrix, we eliminated the highly correlated variables from the dataset. In addition to it, we employed various techniques, selected features which turn out important among all the techniques. The techniques used for feature selection has been listed below.

- Information Gain (Mutual information & SelectKBest)
- Fisher Score (Categorical Variables)
- Univariate ROC_AUC
- Step Forward, Step Backward and Exhaustive Feature Selection
- Random forest feature importance
- Random forest recursive feature elimination
- Feature shuffling
- Hybrid recursive feature elimination(XGBoost)
- Hybrid recursive feature addition(XGBoost)
- Gradient boosting importance

We performed the Principal Component Analysis (PCA) to get an idea of how many variables should we take. As per the results in Table 1, number of features between 20-25 are enough to explain the variation in the data while reducing the feature space.

Number of Features	Total Explained Variance
10	79.07%
15	89.68%
20	94.67%
25	97.15%
30	98.76%
40	99.94%
50	100.00%

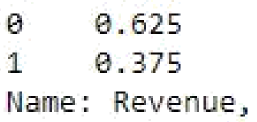
Table 1. PCA check

B. Handling class imbalance

After we split the dataset in 70:30 ratio into training and test data, we worked on the class imbalance. The minority class was only around 15% in the dataset. Therefore, we tried three different techniques to adjust for imbalance

1. Random oversampling: Random oversampling involves randomly selecting examples from the minority class (Revenue = 0), with replacement, setting the number of samples to match that of the majority class (Revenue = 1) and adding them to the training dataset. The ratio of two classes becomes 1:1.

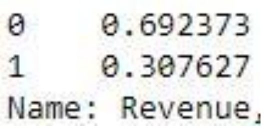
2. Synthetic Minority Oversampling Technique (SMOTE)
SMOTE works by selecting examples from the minority class to synthesize new examples. These new synthetic examples are created by slightly perturbing feature values. Table 2 includes the result after SMOTE is applied to the data.



```
0    0.625
1    0.375
Name: Revenue,
```

Table 2. SMOTE oversampling

3. <u>Oversampling and Undersampling SMOTE and Edited Nearest Neighbors (SMOTEEN)</u>
This method is a combination of first oversampling of minority class using SMOTE and then under-sampling the majority class using edited nearest neighbor (ENN) to reduce the number of overall examples. Like SMOTE, the sampling strategy can be adjusted to fix the ratio of majority and minority class. Below Table 3 is includes the result after SMOTEEN applied to the data.



```
0    0.692373
1    0.307627
Name: Revenue,
```

Table 3. SMOTEEN oversampling and under-sampling

C. Modelling Approach

Several ML algorithms were applied to the data after feature engineering and class adjustment, as listed below.

Logistic Regression
Naive Bayes
K Nearest Neighbor (KNN)
Support Vector Machine (SVM)
AdaBoost
Gradient Boosting
Bagging Tree
Decision Tree
Random Forest
XGBoost
Voting Classifier
Stacking

VI. Result of Analysis

Models to be evaluated using Accuracy, Precision, Recall, F1 Score and AUC ROC with K-Fold Cross Validation. Model with the best performance, will be used for classifying revenue generating user sessions from non-revenue generating ones.

1) True Positive (TP): The test cases which have shown positive in the prediction of "Revenue" attribute and are actually positive in the dataset.

2) True Negative (TN): The test cases which have shown negative in the prediction of "Revenue" attribute and are actually negative in the dataset.

3) False Positive (FP): The test cases which have shown positive in the prediction of "Revenue" attribute but are actually negative in the dataset.

4) False Negative (FN): The test cases which have shown negative in the prediction of "Revenue" attribute but are actually positive in the dataset

$$\textit{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \qquad (1)$$

$$\textit{Precision} = \frac{TP}{TP + FP} \qquad (2)$$

$$\textit{Recall} = \frac{TP}{TP + FN} \qquad (3)$$

$$\textit{F1 score} = \frac{2 \times (\textit{Recall} \times \textit{Precision})}{\textit{Recall} + \textit{Precision}} \qquad (4)$$

B. Model comparison

1. <u>Models on Base Data:</u> Below is a comparison of models built on categorical encoded and scaled data. This is without applying any oversampling technique.

MODEL	Train Accuracy	Test Accuracy	True Negative (Revenue)	CV F1 Score	Remarks
Decision Tree	100	86	315	55	Overfit
Naïve Bayes	36	36	547	31	Worst
AdaBoost	90	89	316	62	
XGBoost	92	90	347	68	Best
Gradient Boosting	92	89	338	66	

Table 4: Comparison of Models (Base data)

These models were run with all features. They were trained on original data i.e. no under or over sampling has been done. Decision Tree model overfitted as training accuracy is too high than the testing accuracy. Naïve Bayes model performed the worst as it has least testing accuracy as well as lowest F1 Score. Boosting models fared better than single classifiers which is to be expected. XGBoost comes out to be the best model with highest F1 Score, good training and testing accuracy and cross validation accuracy of 91. It is also among the Top 2 to be able to predict largest number of Revenue generating samples (Revenue=1) – 347

2. <u>Models with feature selection, upsampling, and hyperparameter tuning:</u> Below is a comparison of models built on categorical encoded and scaled data. Selected features are considered, random upsampling of minority data is done and models were hyperparameter tuned to produce better results. Hyperparameter tuning is the process of searching the ideal model architecture, i.e. selecting the

optimal hyperparameters (parameters which define the model architecture)

MODEL	Train Accuracy	Test Accuracy	True Negative (Revenue)	CV F1 Score	CV ROC AUC
Decision Tree	32	33	543	73	70
Random Forest	52	54	548	96	95
Voting Classifier	81	80	549	96	96
XGBoost	39	41	501	93	93

Table 5: Comparison of Models (Feature selected, random upsampled and hyperparameter tuned)

These models were run with selected 23 features. All the models were hyperparameter tuned to get the best set of hyperparameters corresponding to the respective classifier. Models were trained on upsampled data i.e. minority class were randomly upsampled. Decision Tree model performed the worst as it has least testing accuracy as well as lowest F1 Score. Whereas, Voting Classifier comes out to be the best model with highest F1 Score, good training and testing accuracy and cross validation accuracy of 96. It is also among the Top 3 to be able to predict largest number of Revenue generating samples (Revenue=1) - 549.

3. Models with feature selection, SMOTE, and hyperparameter tuning: Below is a comparison of models built on categorical encoded and scaled data. Feature selection is done. Oversampling of minority data is done using SMOTE. Built models were hyperparameter tuned to produce better results.

MODEL	Train Accuracy	Test Accuracy	True Negative (Revenue)	CV F1 Score	CV ROC AUC
KNN	100	81	358	87	91
SVM	99	85	186	92	94
Random Forest	100	89	376	90	92
Voting Classifier	97	88	321	87	89
XGBoost	91	88	402	87	90

Table 6: Comparison of Models (Feature selected, SMOTE and hyperparameter tuned

These models were run with selected 23 features. All the models were hyperparameter tuned. Models were trained on SMOTE oversampled data. KNN and SVM models overfitted as training accuracy is too high than the testing accuracy. XGBoost comes out to be the best model with F1 Score of 87, good training and testing accuracy and cross validation accuracy of 91. It is also among the Top 3 to be able to predict largest number of Revenue generating samples (Revenue=1) - 402.

4. Models with feature selection, SMOTEEN, and hyperparameter tuning: Below is a comparison of models built on categorical encoded and scaled data. Feature selection is done. Oversampling of minority data is done using SMOTE and under-sampling of majority data using ENN. Built models were hyperparameter tuned to produce better results.

MODEL	Train Accuracy	Test Accuracy	True Negative (Revenue)	CV F1 Score	CV ROC AUC
KNN	100	80	393	92	94
SVM	100	83	318	93	95
Bagging Tree	91	88	413	84	87
Voting Classifier	99	85	437	92	94
XGBoost	95	88	428	89	92

Table 7: Comparison of Models (Feature selected, SMOTEEN and hyperparameter tuned)

These models were run with selected 23 features. All the models were hyperparameter tuned. Models were trained on SMOTEEN oversampled and undersampled data. KNN and SVM models overfitted as training accuracy is too high than the testing accuracy. XGBoost comes out to be the best model with F1 Score of 89, pretty good training/testing accuracy and cross validation accuracy of 94. It is also among the Top 3 to be able to predict largest number of Revenue generating samples (Revenue=1) - 428.

C. *Model Stacking*

Since many of the models demonstrate a fair performance, we will try ‘stacking’ the models to see if we can tease out a higher F1 Score and more importantly increase the number of correctly predicted True Negatives i.e. Revenue generating sessions.

Here we have Random Forest, Decision Tree, and XGBoost as Base Learners. These are hyperparameter tuned learners. Logistic Regression will be our Meta Learner. Base learners are trained on normal data. Using the predictions of Base Learners as inputs, the correct responses from the output, we train the Meta Learner.

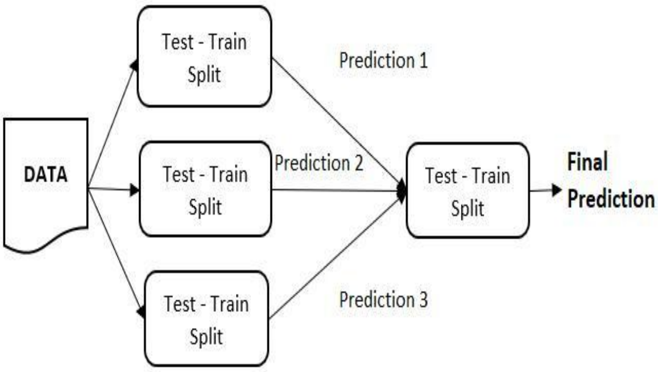


Fig.2. Model stacking framework

In all these cases, we observed the cross-validation accuracy and F1 score are close to 100% without any significant standard deviation. There is also improvement in the prediction of True Negatives. But somehow this doesn't look like generalized predictions. We can see every metric is touching 100% which may be a case of high overfitting. Hence stacking is not the best model.

VII. Conclusion

The aim of our study is to find out the users who are giving Revenue. Hence XGBoost appears to be the best model for our requirement, it generalizes well on the unseen data (high testing accuracy) and have higher F1 score and higher cross validation ROC AUC score among other models in most of the scenarios above. F1 Score is considered best indicator as it is a general rule to look for a higher F1 Score if the aim is to predict the minority class which here is "Revenue".

VIII. recommendations

Following management level insights and recommendations that we derived from our study:

- The number of administrative and information web pages on the website should be as low as possible, as users' interest is on Product related pages and users don't visit other pages.
- Product-related web pages can be expanded as users are willing to spend more time on these irrespective of other conditions.
- Returning website visitors are contributing more to revenue generation. Therefore, various promotion strategies should focus more on these users.
- Most users visit the website in May and November. Also, November accounts for maximum number of purchases. This might be due to Thanksgiving and Christmas shopping. Businesses should also look to the maximize the conversion of online visits to actual purchases in May.
- Target audiences from regions 1 and 3 directly as these regions account for maximum revenue generation.
- For Operating System type 2 and Browser type, 1-2 revenue is more. This might be due to the ease of access and user-friendliness of these browsers/OS. Therefore, we can put up something on the website to ask users to access the website from these OS/browsers something like – "For best results use OS x and browser y" etc.
- New visitors take up a larger percentage in those who complete the purchase. So, it will be a good idea if business form marketing plans to attract new users every day.
- Administrative pages like login, logout, password recovery, profile, email wish list, etc. need to be fixed. Users are spending way too much time which is not good. Both the web page and the back end needs to be made more efficient and speedier.
- Automated client-side scripts should be embedded into the web pages so that dormant/idle sessions get logged out after some time of inactivity. Right now, this is absent which is causing wrong data being collected for analysis.

References

[1] Online Shoppers PurchasingIntention Dataset Data Set: https://archive.ics.uci.edu/ml/datasets/Online+Shoppers+Purchasing+Intention+Dataset

[2] Md Rayhan Kabir, Faisal Bin Ashraf and Rasif Ajwad " Analysis of Different Predicting Model for Online Shoppers' Purchase Intention from Empirical Data" ICCIT, 2019

[3] Moe, Wendy W. "Buying, searching, or browsing: Differentiating between online shoppers using in-store navigational clickstream." Journal of consumer psychology 13.1-2 (2003): 29-39.

[4] Mokryn, Osnat, Veronika Bogina, and Tsvi Kuflik. "Will this session end with a purchase? Inferring current purchase intent of anonymous visitors." Electronic Commerce Research and Applications 34 (2019): 100836. purchase intent of anonymous visitors,Osnat Mokryna,b, Veronika Boginac.

[5] Fernando Aguilar. (2019, October 9). SMOTE-NC in ML Categorization Models for Imbalanced Datasets: https://medium.com/analytics-vidhya/smote-nc-in-ml-categorization-models-fo-imbalanced-datasets-8adbdcf08c25

[6] Hoi Piew Tan, Choon Ling Kwek & Teck-Chai Lau. (2010, August). Investigating the Shopping Orientations on Online Purchase Intention in the e-Commerce Environment: A Malaysian Study: https://www.researchgate.net/publication/288582068_Investigating_the_Shopping_Orientations_on_Online_Purchase_Intention_in_the_e-Commerce_Environment_A_Malaysian_Study

[7] Yuqing Zhang. (2019, September 23). Predicting Online Shoppers Purchasing Intention with H2O: https://zhangyuqing.github.io/2019/09/predicting-online-shoppers-purchasing-intention-with-h2o/

[8] Chen Ling, Tao Zhang & Yuan Chen. (2019, August 27). Customer Purchase Intent Prediction Under Online Multi-Channel Promotion: A Feature-Combined Deep Learning Framework: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8795449

[9] Humphrey Sheil, Omer Rana & Ronan Reilly. (2018, July 21). Predicting purchasing intent: Automatic Feature Learning using Recurrent Neural Networks: https://arxiv.org/pdf/1807.08207.pdf

[10] C. Okan Sakar, S. Olcay Polat, Mete Katircioglu & Yomi Kastro. (2018, May 9). Real-time prediction of online shoppers' purchasing intention using multilayer perceptron and LSTM recurrent neural networks: https://link.springer.com/article/10.1007/s00521-018-3523-0

[11] Jason Brownlee. (2020, January 6). ROC Curves and Precision-Recall Curves for Imbalanced Classification: https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-imbalanced-classification/

[12] Jason Brownlee. (2020, January 22). Combine Oversampling and Undersampling for Imbalanced Classification: https://machinelearningmastery.com/combine-oversampling-and-undersampling-for-imbalanced-classification/