E-Commerce Pricing Model

1B.Uma Rani, 2B.Vinni, 3M.Aishwarya, 4Hemanth Krishna, 5Reddi Prasadu
1Student, 2Student, 3Student, 4Student, 5Asst.Professor
Anill Neerukonda Institute of Technology & Sciences

Abstract - **Our study and project aim to develop an E-Commerce Pricing Model for helping customers to buy their products at low cost. Nowadays everything is becoming online many people started buying goods online on websites like amazon, flip kart ...etc. Many of them don't have any idea about products cost and they get confused with this fall and rise of products cost. So, our project helps those people to buy their products without any second thoughts. Our proposed system provides users with price history charts of products, predict the price drop, and send alert messages to the customers. So that customers can save their money on every online purchase. In this paper, we present an overview of our project E-Commerce Pricing Model UI-user interface, Architectural Design, Introduction, Motivation of our project, the methodologies we use, technologies we use, our research outcomes, societal use, snips of the working application, a piece of code etc. According to the success of other systems which are developed under different project goals, we hope our project helps people who were using.**

I. Introduction

Nowadays everything is becoming online many of us started buying goods online on websites like amazon, flip kart ...etc. Many of them don't have any idea about products cost and they get confused with this fall and rise in products cost. There are some scenarios where the value of the product decreases within days or weeks after user buy their product which ends up in customer disappointment and confusion about when to shop for the merchandise with all these price fall and rise. We also faced this kind of situation over and over. These experiences motivate us to return to this project idea. and therefore, the other motivation that drives us to try to do this project is to avoid wasting the shopper's money on every online purchase. On completion of the project with higher accuracy on the dataset can help users to shop for their products at an inexpensive price with no second thoughts. This software project concentrates on predicting the selling price. when the user logins to our system with a username and password we will analyze the data set for the product Id given by the user and show the predicted price for that product by employing machine algorithms.

II. Research objectives

To design the system to provide price prediction for the product. The developed system will assist all the online buyers to get know the real cost of the product and will save their money on their every online purchase. The proposed solution will let people buy their product without any second thoughts. In comparison to buying products without knowledge about the price history and actual cost of the product the customer may get cheated with false costs that every online shopping app show, this system will cut time wasted by users for online research for product costs. Users can log in through their account to the model.

The following are the objectives of the project:

- To provide E-Commerce Pricing Model with a good user interface.
- To provide the predicted price of the product using the previous cost of the products so that the user can know the real cost of the product.
- It also helps new online shopping users to buy their products without any second thoughts.
- Users can also ask for help if got stuck at any point by keeping mail to our mails.

III. Project Scope and Direction

The main intention of our project is to make customers buy their products at reasonable products. Many systems which are were developed earlier do not provide an option for price prediction.We are developing this system so that users can find a reasonable cost for every product they want to buy online.

The following are our project scopes

- The target groups of this system are customers who buy their goods online using online apps.
- The application is made as a Windows application for anyone with a PC with windows in it can access.

IV. Impact, Significance, and contribution

Nowadays shopping online can be a lot more convenient than heading out. Instead, we can buy through online shopping sites.So, our project helps people to analyze the prices of products.

- Prices for most of the Products on Shopping Sites change every day. We at our model track them and store them on our servers.
- We make that data available for you at your fingertips for free. We provide buyers with more buying power than ever before.
- Our pricing model tells you the best time and best price to buy a product.

V. HISTORICAL DEVELOPMENT PRIOR TO THE PROJECT

Earlier models don't have Price Prediction they just display graphs and bar charts. In earlier projects, the datasets they used had old data, so the prediction was not that accurate. The dataset that they used was 2-3 years old and the sales that it was predicting now we're based on that data. Nowadays data is being generated data such a large rate and there would be so many changes in new data in comparison with the stored data There is a certain correlation between the parameters which affect the sales. The attributes which have a better correlation amongst them must be considered for more accurate prediction.

VI.SYSTEM ENVIRONMENT:

- Pandas:

pandas are very useful python package which is used to work with relational data easily. It aims to be the basic high-level building block for doing practical, real-world data analysis in Python. Additionally, it's the broader goal of becoming the foremost powerful and versatile open-source data analysis/manipulation tool available in any language. it's already well on its way toward this goal.

- Tkinter:

  We have developed the Front-End using Tkinter, it is a python module widely used for creating GUI supported in all operating systems like windows, Linux, etc. Python has many other modules for GUI among all Tkinter is widely used and easy to develop applications using it.

- Matplotlib:

  Matplotlib is an excellent visualization library in Python for 2D arrays. Matplotlib is a multi-platform data visualization library built on the NumPy array and designed to work with a wide range of SciPy stacks. We chose this library because we had an idea to add an accuracy rate in quey writing for the users, for which we use python dictionaries. Using it is easy to plot a graph. Matplotlib contains many platforms such as lines, bars, scatters, histograms, etc. Our system likely uses bars for plotting.

VII.SYSTEM DEVELOPMENT

To describe this application in terms of development, we used dataset to train our model.

Train Dataset:

Our Train dataset is of length 2453 and it consists of 8 columns like ProductID, Brand, category, subcategory, variant, rating, date, price. We use this Train dataset to train our model for predicting the selling price of a product.

product	brand	category	subcategory	variant	rating	date	price
P-2610	B-659	bags wallets belts	bags	hand bags	4.3	2/3/2017	291
P-2453	B-3078	clothing	women s clothing	western wear	3.1	7/1/2015	897
P-6802	B-1810	home decor festive needs	showpieces	ethnic	3.5	1/12/2019	792
P-4452	B-3078	beauty and personal care	eye care	h2o plus eye care	4	12/12/2014	837
P-8454	B-3078	clothing	men s clothing	t shirts	4.3	12/12/2013	470
P-5597	B-1487	home decor festive needs	table decor handicrafts	showpieces	5	9/4/2020	746
P-8398	B-3078	footwear	women s footwear	casual shoes	4.1	4/12/2017	1798
P-10744	B-2830	kitchen dining	cookware	pots pans	3.1	1/12/2013	955

Test Dataset:

Our Test dataset is of length 1052 and it consists of 7 columns like ProductID, Brand, category, subcategory, variant, rating, Date. We use this Test dataset to test the model we created using train dataset.

product	brand	category	subcategory	variant	rating	date
P-11284	B-2984	computers	network components	routers	4.3	1/12/2018
P-6580	B-1732	jewellery	bangles bracelets armlets	bracelets	3	20-12-2012
P-5843	B-3078	clothing	women s clothing	western wear	1.5	1/12/2014
P-5334	B-1421	jewellery	necklaces chains	necklaces	3.9	1/12/2019
P-5586	B-3078	clothing	women s clothing	western wear	1.4	1/12/2017
P-11206	B-3078	clothing	women s clothing	western wear	2.1	3/12/2018
P-9966	B-2609	mobiles accessories	mobile accessories	speakers	2.8	20-05-2015
P-5367	B-1431	jewellery	rings	unknown	4.9	7/1/2016
P-9075	B-2326	jewellery	rings	unknown	4.9	1/1/2015

To seek out the accurate selling price of the Product we used four different machine learning algorithms.They are Lightgbm regression, XGboost regression, Catboost regression, and Random forest regression.
First, we take each algorithm and predicting the selling price of products and store them in an excel sheet. We will get four excel sheets after going through four algorithms using those four excel sheets we find the ensembled predicted values of the product.

```
def rmse(y_true, y_pred):
    return mean_squared_error(y_true, y_pred) ** 0.5
```

1.LightGBM Regression:

LightGBM may be a gradient boosting framework supported by decision trees to extend the efficiency of the model and reduce memory usage.

It uses two novel techniques: Gradient-based One Side Sampling and Exclusive Feature Bundling (EFB) which fulfills the constraints of the histogram-based algorithm that's primarily employed in all GBDT (Gradient Boosting Decision Tree) frameworks. the 2 techniques of GOSS and EFB described below form the characteristics of the LightGBM Algorithm. They
comprise together to create the model work efficiently and supply it innovative over other GBDT frameworks

Gradient-based One Side Sampling Technique for LightGBM:

Different data instances have varied roles within the computation of data gain. The instances with larger gradients (i.e., under-trained instances) will contribute more to the knowledge gain. GOSS keeps those instances with large gradients (e.g., larger than a predefined threshold, or among the highest percentiles), and only randomly drops those instances with small gradients to retain the accuracy of knowledge gain estimation. This treatment can result in a more accurate gain estimation than uniformly sampling, with the identical target rate, especially when the worth of knowledge gain contains a large range.

```
lgbm = LGBMRegressor (n_estimators=1000, num_leaves=127, max_depth=-1,min_child_samples=4, learning_rate=0.02, colsample_bytree=0.4, reg_alpha=0.5, reg_lambda=2)
_ = lgbm.fit(X_trn, np.log(y_trn), eval_set = [(X_val, np.log(y_val))], verbose=100, early_stopping_rounds=100, eval_metric='rmse')
```

Above code is the important part of the algorithm we are sending no.of leaves, depth of the tree…etc as parameters for the predefined LGBMRegressor in Python language.

Here, the OOF val score is the accuracy rate of the Lightgbm algorithm and it is 0.6466.

OOF val score: 0.6466212461498502
Mean rmsle: 0.6448 and std Dev. is 0.05

After getting Predicted price for each product in the dataset we are saving those values in an excel sheet called mean_encoding_lgbm2.xlsx.

```
sub.to_excel('mean_encoding_lgbm2.xlsx', index=False)
```

2.XGboost Regression:

The objective function contains a loss function and a regularization term. It tells about the difference between actual values
and predicted values, i.e how far the model results are from the 000 values. the foremost common loss function in XGBoost
for regression problems is reg: linear, which for binary classification is reg: logistics.
Ensemble learning involves training and mixing individual models (known as base learners) to urge one prediction, and
XGBoost is one in all the ensemble learning methods. XGBoost expects to own the bottom learners which are uniformly bad
at the rest so when all the predictions are combined, bad predictions eliminate and better one sums up to create final good predictions.

```
xgb = XGBRegressor(n_estimators=1000, max_depth=12, learning_rate=0.05, colsample_bytree=0.45) _ = xgb.fit(X_trn, np.log(y_trn), eval_set = [(X_val, np.log(y_val))], verbose=100, early_stopping_rounds=100, eval_metric='rmse')
```

Above code is the important part of the algorithm we are sending learning rate,depth of the tree…etc as parameters for the predefined XGBRegressor in Python language.

Here, the OOF val score is the accuracy rate of the Xgboost algorithm and it is 0.6468.

OOF val score: 0.6468229090430355
Mean rmsle: 0.6447 and std Dev. is 0.05

After getting Predicted price for each product in the dataset we are saving those values in an excel sheet called mean_encoding_xgboost2.xlsx.

```
sub.to_excel('mean_encoding_xgboost2.xlsx', index=False)
```

3.Catboost Regression:

The main idea of boosting is to sequentially combine many weak models (a model performing slightly better than random chance) and thus through greedy search create a powerful competitive predictive model.

```
cat = CatBoostRegressor(n_estimators=2000, learning_rate=0.05, max_depth=9, rsm=0.5)_ = cat.fit(X_trn, np.log(y_trn),
    eval_set = [(X_val, np.log(y_val))], verbose=100, early_stopping_rounds=100)
```

Above code is the important part of the algorithm we are sending learning rate,depth of the tree…etc as parameters for the Predefined CatBoostRegressor in Python language.

Here, the OOF val score is the accuracy rate of the Catboost algorithm and it is 0.6515.

OOF val score: 0.6515039864490813
Mean rmsle: 0.6505 and std Dev. is 0.04

After getting Predicted price for each product in the dataset we are saving those values in an excel sheet called mean_encoding_catboost2.xlsx.

```
sub.to_excel('mean_encoding_catboost2.xlsx', index=False)
```

4.Random Forest regressor:

Random Forest could be a popular machine learning algorithm that belongs to the supervised learning technique. It will be used for both Classification and Regression problems in ML. it's supported the concept of ensemble learning, which could be a process of mixing multiple classifiers to unravel a posh problem and improve the performance of the model. because the name suggests, "Random Forest may be a classifier that contains variety of decision trees on various subsets of the given dataset and takes the typical to enhance the predictive accuracy of that dataset." rather than wishing on one decision tree, the random forest takes the prediction from each tree, and supported the bulk votes of predictions, it predicts the ultimate output. The greater number of trees within the forest results in higher accuracy and prevents the matter of overfitting.

```
rfg = RandomForestRegressor(n_estimators=1000,random_state=1234,max_depth=8)_ = rfg.fit(X_trn, np.log(y_trn))
```

Above code is the important part of the algorithm we are sending random states,depth of the tree…etc as parameters for the Predefined RandomForestRegressor in Python language.

Here, the OOF val score is the accuracy rate of the Radom Forest algorithm and it is 0.6600.

OOF val score: 0.6600398177336928
Mean rmsle: 0.6588 and std Dev. is 0.04

After getting Predicted price for each product in the dataset we are saving those values in an excel sheet called mean_encoding_rfg2.xlsx.

```
sub.to_excel('mean_encoding_rfg2.xlsx', index=False)
```

When we compare the prediction values of 4 algorithms with one another we notice that random forest contains a more accurate value than the other algorithms.

5.Ensembling:

After obtaining individual prediction values for four algorithms, we find the mean of these prediction values so the expected value are going to be very accurate.

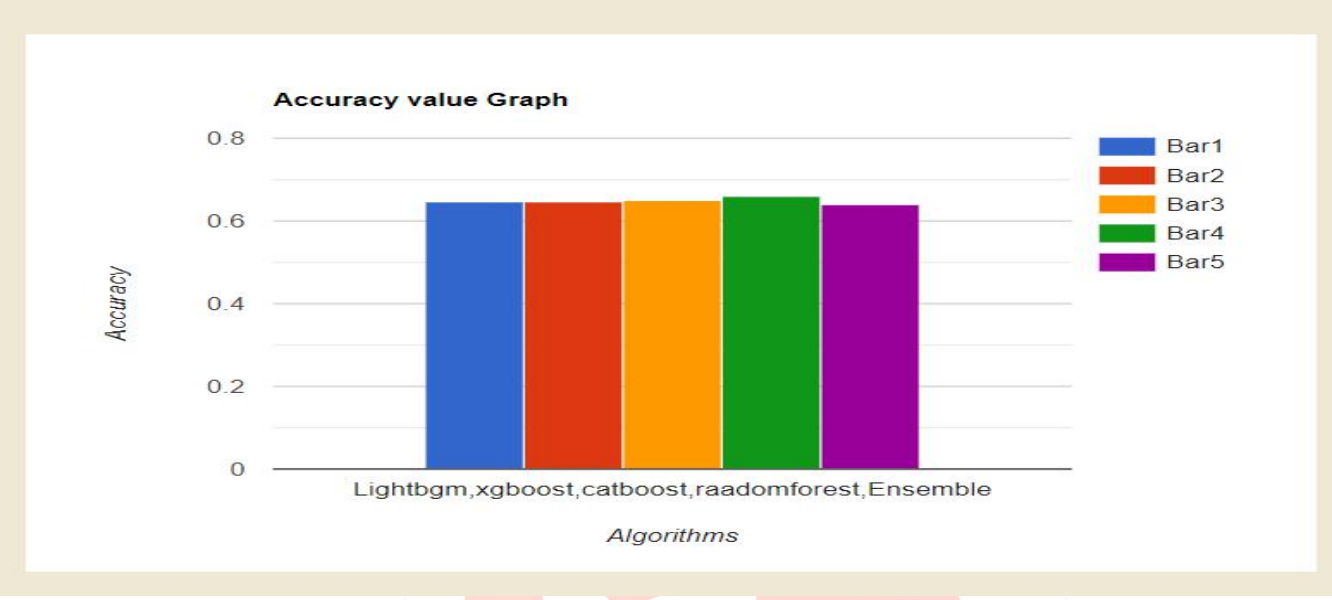
```
test_preds_ensemble = tp0*0.2 + tp4*0.4 + tp2*0.4
sub = pd.DataFrame({'Selling_Price': test_preds_ensemble})
sub['Selling_Price'] = np.clip(sub['Selling_Price'], y.min(), y.max())
sub['Selling_Price'].describe()
sub.head(10)
```

Above code is the important part of the algorithm.Here, the OOF val score is the accuracy rate of the Radom Forest algorithm and it is 0.6452.

OOF val score: 0.6452050224015999

After getting Ensembled Predicted prices for each product in the dataset we are saving those values in an excel sheet called mean_encoding_ensemble_lgbcatrfg2.xlsx.

```
sub.to_excel('mean_encoding_ensemble_lgbcatrfg2.xlsx', index=False)
```



VIII. Advantages of the System:

- Consumers in the modern world are becoming more sophisticated in their purchase decisions.So,the E-Commerce pricing model will help the users to take advantage of the predicted values so that they can make their purchasing decisions.
- The previous prices of products will help the customer to find the patterns and can expect the least possible cost.
- The prediction will be able to provide a crucial impact towards the business decision making process.
- Also, it can help customers to pre plan their purchasing decision.

IX. Acknowledgment

My team wish to acknowledge Reddi Prasadu (Ph.D), Assistant Professor, Department of Information Technology, ANITS,

Visakhapatnam for assisting me in various aspects which impact the successful completion of the project. The authors for the

below mentioned journals in the reference section are also greatly acknowledged.

X. Conclusion

In existing model predictions finished unstructured or semi-structured knowledge. connected works explicit results with one explicit machine learning algorithmic program. A comparative study on varied algorithms start off with best model supported accuracy. Accuracy of the algorithmic program varies depend upon client knowledge. This project projected churn prediction in e- retail with structured knowledge, imperative options, machine learning techniques. On cross validation higher algorithms square measure picked and nonappointive supported vote. Ensemble algorithmic program is that the closing result and counsel acceptable preventive measures on churners

In future we'll enhance the purchasers World Health Organization square measure foretold as churn square measure once

more analyzed supported their reason of churn. Offers and edges square measure given solely to voluntary churners, that results in target specific selling. The churn prediction are going to be enforced in mobile application. the applying can mechanically send notification to churned customers concerning the offers and edges. it'll inform the client to try and do searching and inform them concerning updates. it'll serve the marketers as user friendly and price economical to predict their churn with the information and to proactively take retention actions on their customers.

XI. References

[1] Zhao, K. and Wang, C. (2017) 'Sales Forecast in E-commerce using Convolutional Neural Network', (August 2017). Available at: http://arxiv.org/abs/1708.07946.

[2] Bandara, K. et al. (2019) 'Sales Demand Forecast in E-commerce using a Long Short-Term Memory Neural Network Methodology'.

[3] Li, M., Ji, S. and Liu, G. (2018) 'Forecasting of Chinese E-Commerce Sales: An Empirical Comparison of ARIMA, Nonlinear Autoregressive Neural Network, and a Combined ARIMA-NARNN Model', Mathematical Problems in Engineering, 2018, pp. 1–12. doi: 10.1155/2018/6924960.

[4] Kuo-Kun Tseng & Regina Fang-Ying Lin & Hongfu Zhou & Kevin Jati Kurniajaya & Qianyu Li, 2018. "Price prediction of e-commerce products through Internet sentiment analysis,"

[5]https://www.researchgate.net/publication/330011161_PRICE_TRACKING_BEHAVIOUR_IN_ELECTRONIC_COMMERCE_AND_THE_MODERATING_ROLE_OF_FAIR_PRICE_PERCEPTION

[6] https://www.ijert.org/predicting-online-product-sales-using-machine-learning