Capstone Final Report - Predicting Item Purchases in eCommerce

Introduction

The eCommerce industry is experiencing significant growth, with more consumers turning to online platforms for their shopping needs. Understanding user behavior in eCommerce is essential for businesses to enhance user experiences and drive sales. In this project, I explore a dataset from a multi-category eCommerce store to predict item purchases. By analyzing this data, I aim to uncover insights that can help businesses optimize their strategies and improve customer engagement.

Data Overview

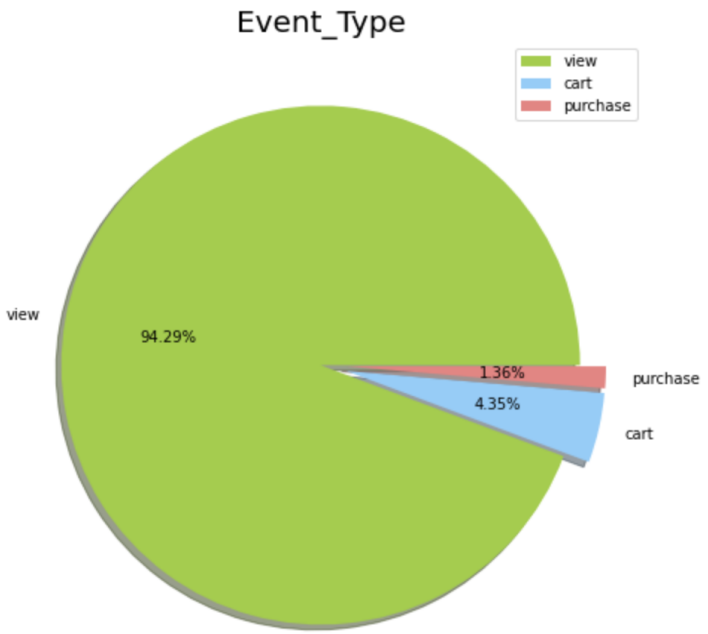
The dataset used for this project is sourced from eCommerce behavior data collected in November 2019. It contains valuable information about user interactions on the platform, including event timestamps, product details, and user attributes. Below is an overview of the dataset:

- **Shape**: The dataset consists of 67501979 rows and 9 columns.
- **Data Types**: These include timestamps, numeric values, and categorical variables.
- **Missing Values**: Some columns have missing values, such as category_code, and brand user_session. These missing values will be addressed during data preprocessing.

Exploratory Data Analysis (EDA)

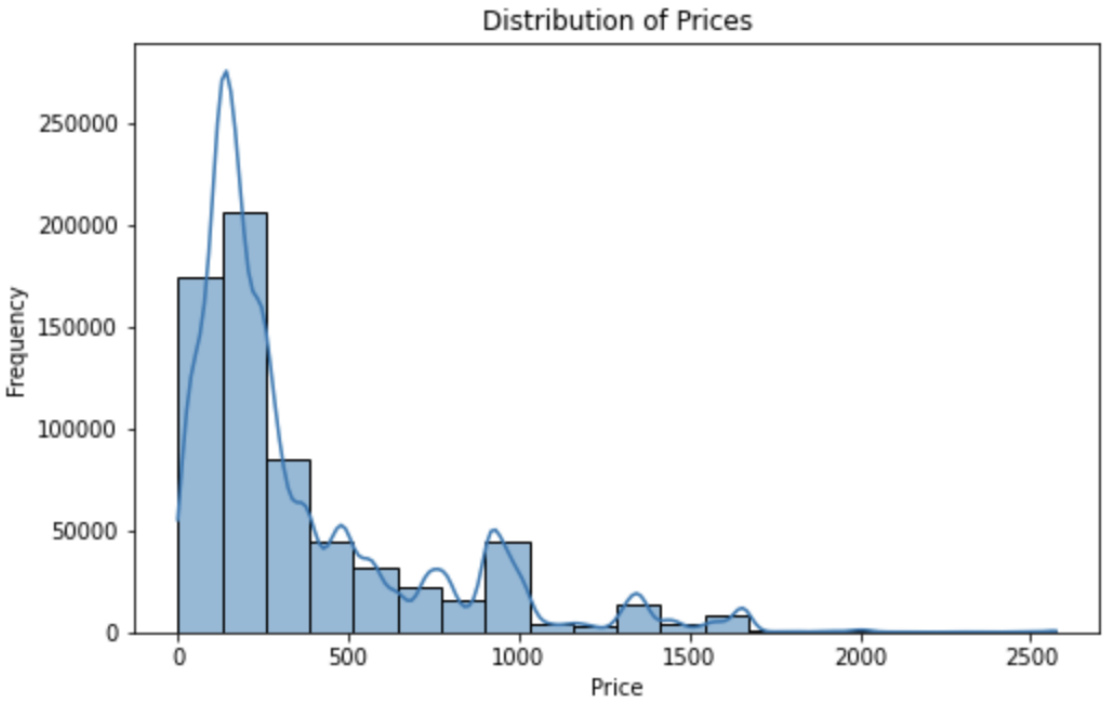
Exploratory Data Analysis is a crucial step in understanding the dataset and extracting meaningful insights. Let's explore some key aspects of the data:

- **Data Cleaning**: Duplicate records were identified and removed from the dataset. Timestamps were converted to datetime objects for time-based analysis.
- **Event Types**: The dataset contains event types such as 'view,' 'cart,' and 'purchase.' The distribution of event types was visualized, revealing that most events are ‘view’, and not all items which were added to the cart were purchased.



- **Brands**: The most popular brands among sellers were identified, providing insights into purchasing preferences.

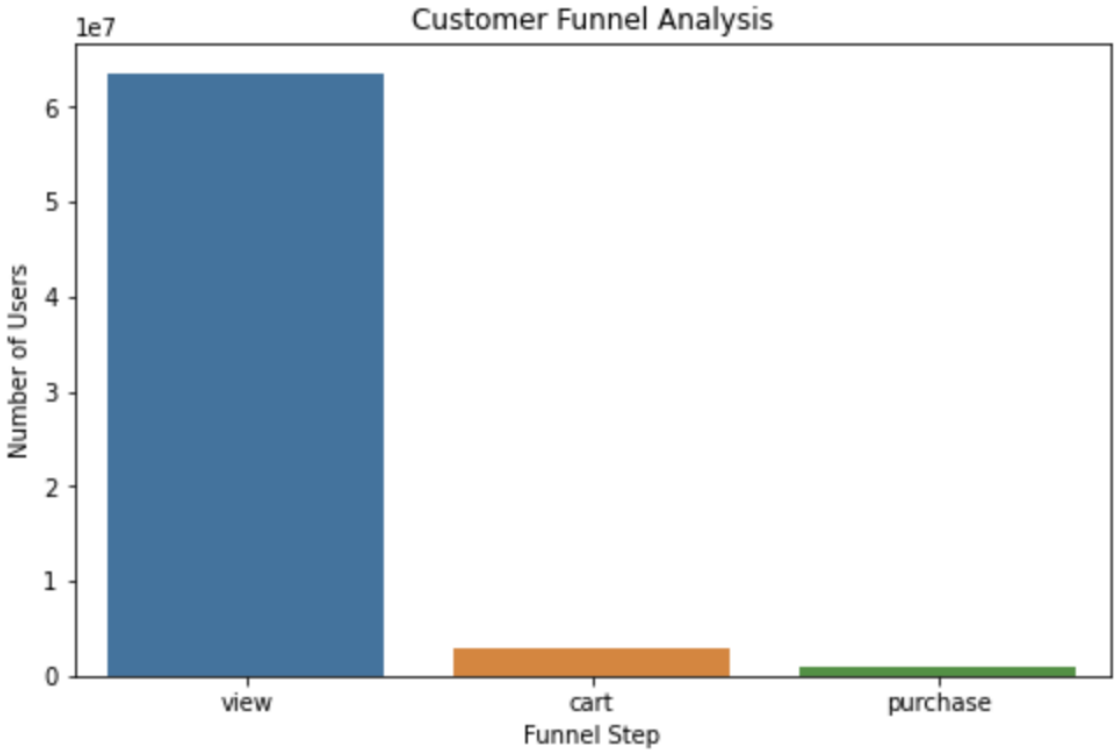
- **Price Distribution**: A histogram of purchase prices shows the distribution of item prices. This information can be helpful in pricing strategies.



- **Conversion Rates**: The conversion rate, indicating the percentage of users who made a purchase after visiting the website, was calculated and visualized. This metric is essential for

understanding user engagement. It shows that about 12% of customers purchased the item after visiting the website.

- **Funnel Analysis**: A funnel analysis was performed to track user progression through various stages, including viewing products, adding them to the cart, and making a purchase. This analysis helps identify drop-off points in the user journey.



Feature Engineering

In the feature engineering section, I created additional features to enhance our predictive model's performance. These features include:

- **Purchases per Session**: I calculated the number of purchases made per user session and added this as a feature.
- **Views per Session**: The count of views per user session was calculated and incorporated.
- **Carts per Session**: I computed the number of items added to the cart per user session and included it in our dataset.
- **Label Encoding**: Categorical variables such as 'brand,' 'category_code,' and 'price_category' were label-encoded to prepare them for modeling.

- **Datetime Transformation**: I converted the 'event_time' column to datetime objects and extracted the 'hour' feature from it.

- **Standard Scaling**: Numeric features like 'day,' 'hour,' and 'price' were standardized using the StandardScaler to ensure they have a similar scale for modeling.

Modeling

In the modeling section, I employed the following machine learning techniques:

- **XGBoost with Spark**: I initially trained an XGBoost classifier using Spark, which allowed us to work with a large dataset. The model was trained with predefined hyperparameters, and predictions were made on the test set. Key metrics such as accuracy, precision, recall, ROC-AUC, log-loss, and the confusion matrix were calculated to evaluate the model's performance.

- **Optuna Hyperparameter Tuning:** I used the Optuna library to optimize hyperparameters for the XGBoost classifier. Through several trials, Optuna found the best set of hyperparameters that maximize accuracy. We retrained the XGBoost model with these optimized hyperparameters and evaluated its performance.

- **Logistic Regression**: A logistic regression model was trained on the data, and its performance was evaluated using metrics like accuracy, confusion matrix, and classification report.

- **Optuna Hyperparameter Tuning for Logistic Regression**: I utilized Optuna again to optimize hyperparameters for logistic regression. The best hyperparameters were used to create an optimized logistic regression model, and its performance was assessed.

Results and Findings

The accuracy of XGBoost and Logistic regression models on 10% of data was the same. Confusion matrix and classification report also matched.

Accuracy: 0.9851788659500654

Confusion Matrix:

```
[[1992076       0]
 [  29969       0]]
```

Classification Report:

	precision	recall	f1-score	support
0	0.99	1.00	0.99	1992076
1	0.00	0.00	0.00	29969
accuracy			0.99	2022045
macro avg	0.49	0.50	0.50	2022045
weighted avg	0.97	0.99	0.98	2022045

Conclusion

Both models showed good performance on the data.

Best hyperparameters for XGBoost:

{'learning_rate': 0.014253246285117174, 'max_depth': 4, 'n_estimators': 256, 'subsample': 0.8837953460731236, 'colsample_bytree': 0.9942632456168128, 'gamma': 1.5822088270626555, 'lambda': 3.797855512226913}

Best hyperparameters for Logistic regression:

{'C': 0.1, 'solver': 'saga', 'class_weight': None}

My findings provide valuable insights for businesses operating in the eCommerce sector.