VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

Predicting Online Shoppers' Purchasing Intention

Under the Guidance of

Assistant Prof.Bincy Ivin

Team Members

Raj Padvekar(39)
Kaustubh Pukale(45)
Pranav Rahuvanshi(46)

OBJECTIVES AND PROBLEM STATEMENT

- Problem: E-commerce businesses need to distinguish potential buyers from casual browsers.
- Objective: Develop a machine learning model to predict purchase intent using browsing behavior data.
- This will enable businesses to personalize the shopping experience and increase conversion rates.

DATASET DESCRIPTION

Dataset Description

- Dataset: Online Shoppers Purchasing Intention Dataset from UCI Machine Learning Repository.
- 12,330 user sessions with 18 features.
- Features include:
  - Numerical (e.g., page visits, time spent)
  - Categorical (e.g., month, browser)
  - Boolean (e.g., weekend, revenue)
- Target Variable: Revenue (Purchase or No Purchase).
- Imbalanced dataset: 15.5% purchases, 84.5% no purchases.

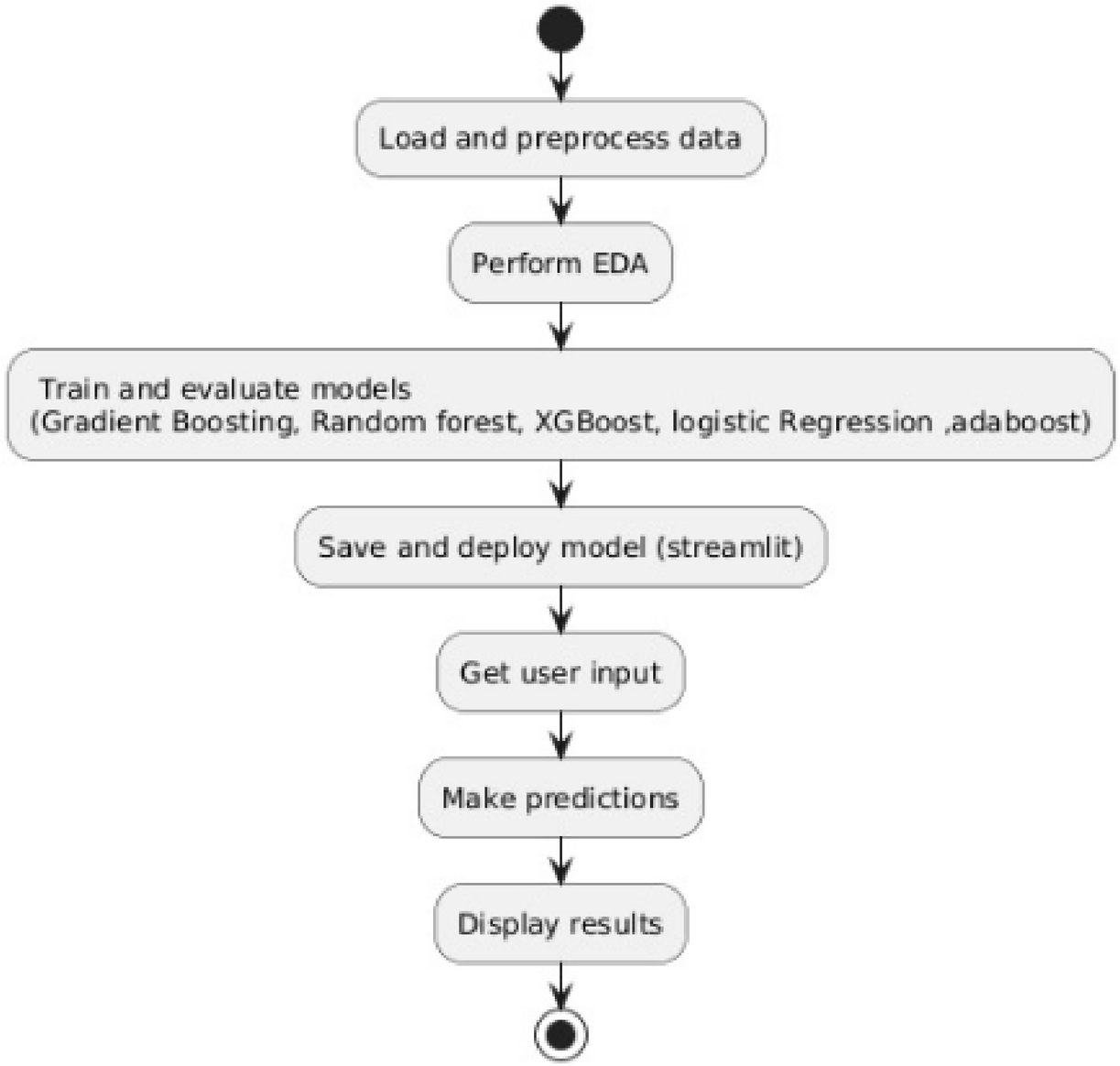
METHODOLOGY

- Exploratory Data Analysis (EDA): Understanding data patterns.
- Data Preprocessing:
  - Handling duplicate values.
  - Encoding categorical variables.
  - Feature engineering (e.g., log transformation, IQR for outlier handling).
  - Normalization of numerical features.
  - SMOTE for class imbalance.
- Model Selection and Training: Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, AdaBoost.

METHEDOLOGY

- Dataset Composition: Analyzed 12,330 user sessions with 18 features from the UCI Machine Learning Repository.
- Class Imbalance: Only 15% of sessions resulted in purchases, indicating a significant class imbalance.
- Key Predictors: Higher PageValues are strongly associated with increased likelihood of transactions.
- Feature Distributions: Many numerical features exhibit right-skewed distributions, suggesting that while most users engage minimally, a few have significantly higher interactions.
- User Behavior: Users spending more time on product-related pages and having higher page views are more inclined to make purchases.
- Negative Correlations: Higher ExitRates and BounceRates correlate negatively with PageValues, indicating that sessions with quick exits or bounces reflect less engaged users, leading to fewer conversions.

ARCHITECTURE



Tools & Technologies Used

1. Data Preprocessing & Analysis
- pandas: For loading data, cleaning, and manipulation (e.g., handling missing values, encoding).
- NumPy: For numerical operations and efficient data handling.

2. Exploratory Data Analysis (EDA)
- seaborn & matplotlib: For visualizing data distributions, correlations, and insights using plots like histograms, box plots, and heatmaps.

3. Model Building & Evaluation
- scikit-learn (sklearn):
  - Model training: Logistic Regression, Random Forest, Gradient Boosting, AdaBoost.
  - Model evaluation: Accuracy, precision, recall, F1 score, etc.
  - RandomizedSearchCV: For hyperparameter tuning and model optimization.

4. Model Saving
- joblib: To save and load trained machine learning models efficiently.

5. Model Deployment
- Streamlit: To build an interactive web app where users can input data, make predictions, and view results.
- Gemini API: Integrated a chatbot using Gemini API to enhance user interaction by answering queries, providing insights, and assisting with navigation within the application.

Chatbot Integration Using Gemini

Overview: This project involves the development of a chatbot system that leverages the capabilities of the Gemini Large Language Model (LLM). The chatbot is designed to accept user queries in JSON format, process them via the Gemini API, and return accurate, context-aware responses.

Key Functionalities:

•Accepts input in structured JSON format:jsonCopyEdit{ "question": "What is overfitting in machine learning?"

•Utilizes the Gemini API for natural language understanding and response generation.

•Delivers coherent, informative answers suitable for applications such as education, customer support, and knowledge retrieval systems.

•Easily integrable into various platforms, including web and mobile applications

RESULTS

Train Set:

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	0.8820	0.8866	0.8761	0.8813
Adaboost	0.8892	0.8998	0.8758	0.8876
Gradient Boosting	0.9260	0.9249	0.9273	0.9261
XGBoost	0.9218	0.9226	0.9209	0.9217
Logistic Regression	0.8580	0.8907	0.8162	0.8518

Train Set Classification Report

Test Set

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	0.8695	0.5420	0.8040	0.6475
Adaboost	0.8831	0.5815	0.7711	0.6630
Gradient Boosting	0.8930	0.6149	0.7546	0.6776
XGBoost	0.8943	0.6178	0.7637	0.6830
Logistic Regression	0.8779	0.5656	0.7821	0.6564

Test Set Classification Report

ROC Curve Insights

- All models show AUC scores around 0.92–0.93, indicating strong classification ability.
- Gradient Boosting and XGBoost again lead slightly with AUC = 0.93, confirming their reliability across multiple metrics.

CONCLUSION AND FUTURE WORK

CONCLUSION

- Gradient Boosting and XGBoost are the top-performing models across both training and test sets.
- These two models maintain a balance between high accuracy and generalization (not overfitting too much)..

FUTURE WORK

- Integration with live e-commerce systems.
- Advanced feature engineering.
- Hyperparameter tuning and AutoML.
- Enhanced user interface.
- Scalability and cloud deployment.

Thank You