Churn Prediction System

Deepanshu

Abstract

This project presents the development of a machine learning-based Customer Churn Prediction System using the Telco Customer Churn dataset. The objective is to identify customers likely to discontinue services and enable businesses to take proactive retention measures. The dataset underwent extensive preprocessing, including handling missing values, encoding categorical variables, and removing anomalies. Various classification models were trained and evaluated, including Logistic Regression, Random Forest, XGBoost, Gradient Boosting, and Voting Classifiers. The system's performance was assessed using ROC-AUC, precision-recall metrics, and F1/F2 scores. The final solution was deployed via a user-friendly Streamlit application, allowing real-time prediction based on customer input.

Introduction

Customer churn—the loss of clients or subscribers—is a critical metric for subscription-based industries such as telecommunications, banking, and SaaS. Predicting churn accurately allows companies to implement targeted strategies to retain high-risk customers, thus improving profitability and customer satisfaction.

In this project, we utilize historical customer data from a telecom company to build a predictive churn classification model. The Telco Customer Churn dataset from Kaggle includes detailed customer demographics, service usage patterns, contract information, and payment behavior. The goal is to predict whether a customer is likely to churn based on these attributes.

To achieve this, we applied several machine learning algorithms, including tree-based models and ensemble techniques, and evaluated their performance using key classification metrics. The project concludes with the deployment of the model in a Streamlit web application for interactive use.

Data Source and Preprocessing

Data Source

The dataset used for this project is the Telco Customer Churn dataset, publicly available on Kaggle. It contains 7,043 records of customer information, including demographic data, account details, and usage behavior. The target variable is Churn, indicating whether a customer has left the service (Yes) or not (No).

Key features include:

gender, SeniorCitizen, Partner, Dependents

tenure, MonthlyCharges, TotalCharges

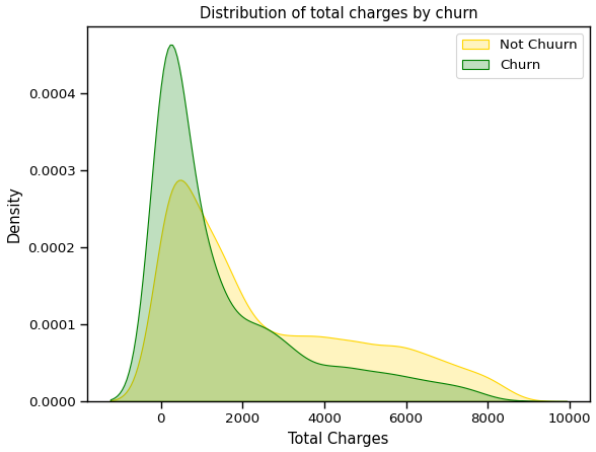
PhoneService, InternetService, StreamingTV

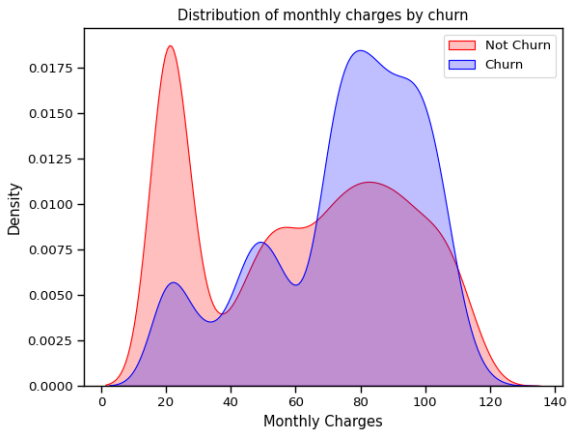
Contract, PaymentMethod, and more

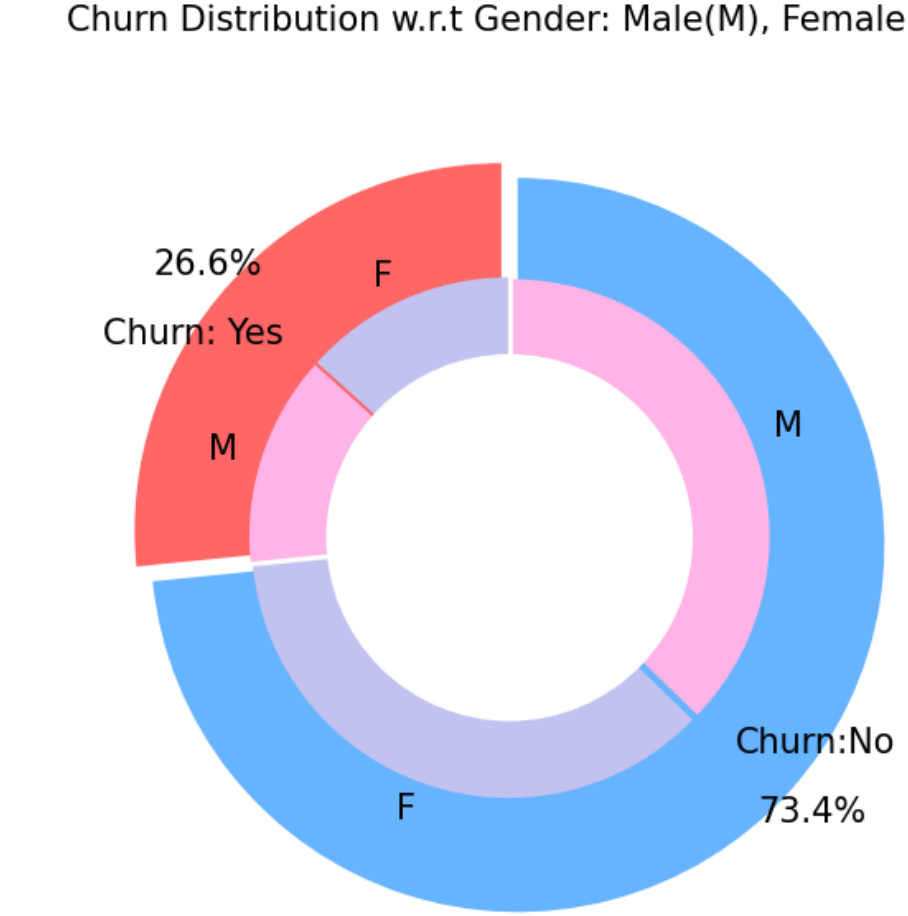
These features collectively offer valuable insights into customer profiles and behaviors relevant to churn prediction.

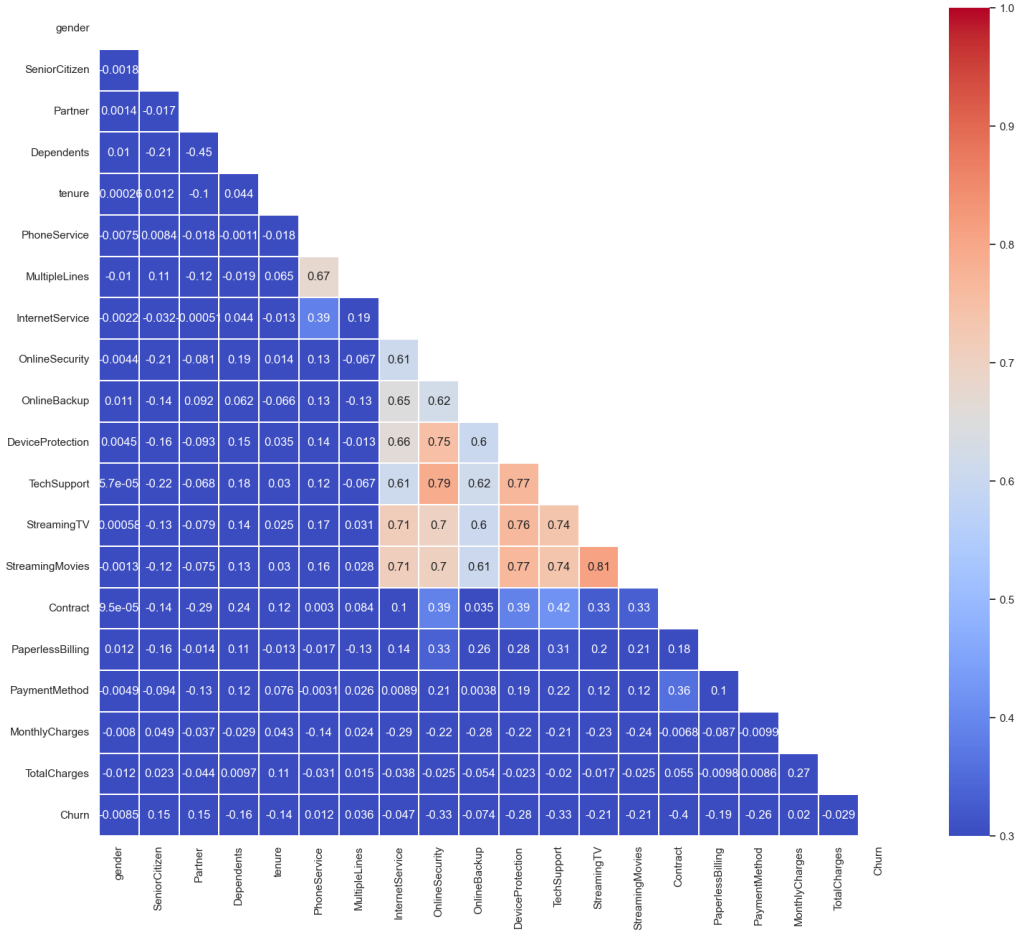
Preprocessing Steps

1. Removal of Non-Predictive Fields: The customerID column was removed as it serves only as a unique identifier and carries no predictive value.
2. Handling Missing and Invalid Values:
   The TotalCharges column contained blank strings for some records, which were converted to NaN and handled using mean imputation.
   11 records with missing or invalid values were identified and either imputed or dropped.
3. Zero Tenure Records: Customers with a tenure of 0 months were considered unrealistic or incomplete and were removed from the dataset.
4. Data Type Conversion: The TotalCharges field was converted from string to numeric format to allow proper numerical analysis.
5. Label Normalization: The SeniorCitizen field, originally encoded as 0 and 1, was mapped to No and Yes for better interpretability.
6. Encoding Categorical Features: Categorical variables were encoded using Label Encoding or One-Hot Encoding, depending on their cardinality and model requirements.
7. Feature Scaling: Numerical features like tenure, MonthlyCharges, and TotalCharges were scaled using StandardScaler to ensure uniformity across input features, especially for distance-based algorithms like KNN.






Churn Distribution w.r.t Gender: Male(M), Female(F)
26.6%
Churn: Yes
F
M
M
F
Churn:No
73.4%


gender
SeniorCitizen -0.0018
Partner 0.0014 -0.017
Dependents 0.01 -0.21 -0.45
tenure 0.00026 0.012 -0.1 0.044
PhoneService -0.0075 0.0084 -0.018 -0.0011 -0.018
MultipleLines -0.01 0.11 -0.12 -0.019 0.065 0.67
InternetService -0.0022 -0.032 0.0005 0.044 -0.013 0.39 0.19
OnlineSecurity -0.0044 -0.21 -0.081 0.19 0.014 0.13 -0.067 0.61
OnlineBackup 0.011 -0.14 0.092 0.062 -0.066 0.13 -0.13 0.65 0.62
DeviceProtection 0.0045 -0.16 -0.093 0.15 0.035 0.14 -0.013 0.66 0.75 0.6
TechSupport 5.7e-05 -0.22 -0.068 0.18 0.03 0.12 -0.067 0.61 0.79 0.62 0.77
StreamingTV 0.0005 -0.13 -0.079 0.14 0.025 0.17 0.031 0.71 0.7 0.6 0.76 0.74
StreamingMovies -0.0013 -0.12 -0.075 0.13 0.03 0.16 0.028 0.71 0.7 0.61 0.77 0.74 0.81
Contract 9.5e-05 -0.14 -0.29 0.24 0.12 0.003 0.084 0.1 0.39 0.035 0.39 0.42 0.33 0.33
PaperlessBilling 0.012 -0.16 -0.014 0.11 -0.013 -0.017 -0.13 0.14 0.33 0.26 0.28 0.31 0.2 0.21 0.18
PaymentMethod -0.0049 -0.094 -0.13 0.12 0.076 -0.0031 0.026 0.0089 0.21 0.0038 0.19 0.22 0.12 0.12 0.36 0.1
MonthlyCharges -0.008 0.049 -0.037 -0.029 0.043 -0.14 0.024 -0.29 -0.22 -0.28 -0.22 -0.21 -0.23 -0.24 -0.0068 -0.087 -0.0099
TotalCharges -0.012 0.023 -0.044 0.0097 0.11 -0.031 0.015 -0.038 -0.025 -0.054 -0.023 -0.02 -0.017 -0.025 0.055 -0.0098 0.0086 0.27
Churn -0.0085 0.15 0.15 -0.16 -0.14 0.012 0.036 -0.047 -0.33 -0.074 -0.28 -0.33 -0.21 -0.21 -0.4 -0.19 -0.26 0.02 -0.029
gender
SeniorCitizen
Partner
Dependents
tenure
PhoneService
MultipleLines
InternetService
OnlineSecurity
OnlineBackup
DeviceProtection
TechSupport
StreamingTV
StreamingMovies
Contract
PaperlessBilling
PaymentMethod
MonthlyCharges
TotalCharges
Churn
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3

Modeling Approach

The goal of this project is to predict whether a customer is likely to churn, using structured customer data from a telecom company. To accomplish this, we followed a structured machine learning pipeline comprising the following key stages:

1. **Data Understanding and Cleaning:** The dataset was first explored to understand feature distributions, data types, and missing values. Irrelevant columns like `customerID` were removed, and inconsistencies in numeric fields (e.g., blank `TotalCharges`) were cleaned.
2. **Feature Engineering and Encoding:** Categorical features were encoded using label encoding and one-hot encoding as appropriate. Numerical variables were scaled using StandardScaler to normalize values for models sensitive to distance.
3. **Train-Test Split:** The dataset was divided into training and test sets to evaluate model generalization. Stratification was used to ensure a balanced distribution of churn classes in both sets.
4. **Model Training:** A variety of classification models were trained, including Logistic Regression, Decision Tree, Random Forest, XGBoost, Gradient Boosting, Support Vector Machines, K-Nearest Neighbors, Naive Bayes, and Voting Classifiers. Each model was evaluated using cross-validation.
5. **Model Evaluation:** Models were assessed using several metrics including ROC-AUC, Accuracy, Precision, Recall, F1 Score, and F2 Score. ROC and confusion matrix plots were also used to visualize performance.
6. **Hyperparameter Tuning:** Grid Search with cross-validation (`GridSearchCV`) was applied to improve the performance of ensemble models like Random Forest and Gradient Boosting by tuning parameters such as number of estimators and tree depth.
7. **Model Selection:** Final model selection was based on multiple performance indicators, prioritizing F2 Score and ROC-AUC due to the business goal of maximizing the correct identification of potential churners.
8. **Deployment:** The best-performing model was saved using `pickle` and deployed using `Streamlit`, allowing real-time churn predictions through a user-friendly web interface.

Evaluation and Results

The models were evaluated using 10-fold cross-validation on the training data and further validated using a held-out test set. Key evaluation metrics include ROC-AUC, Accuracy, Precision, Recall, F1 Score, and F2 Score. The F2 score was especially prioritized to give more weight to Recall, which is critical in churn prediction to identify as many potential churners as possible.

Model Comparison Summary

Table 1 summarizes the mean ROC-AUC and Accuracy of the models across all folds. Voting Classifier, Gradient Boosting, and Adaboost consistently delivered the highest performance, with ROC-AUC scores above 84 and accuracy around 80%.

Table 1: Model Performance Comparison (ROC-AUC and Accuracy)

Algorithm	**ROC AUC Mean**	**ROC AUC STD**	**Accuracy Mean**	**Accuracy**
Voting Classifier	84.82	1.35	79.95	2.08
Gradient Boost Classifier	84.61	1.45	79.36	2.05
Adaboost	84.39	1.41	79.93	1.89
Logistic Regression	84.30	1.27	74.64	1.66
SVC	82.94	1.32	79.07	1.44
Random Forest	82.92	2.02	78.83	1.94
Gaussian NB	82.19	2.20	75.38	1.60
Kernel SVM	79.68	1.66	79.34	1.86
K-Nearest Neighbors	77.23	2.33	75.86	1.78
Decision Tree Classifier	65.42	2.48	72.57	2.43

Detailed Classification Metrics

Table 2 reports detailed metrics from the test set. Among all models, the Voting Classifier and Adaboost achieved the best F1 and F2 scores while maintaining high accuracy. Naive Bayes achieved the highest recall, which is favorable for identifying churners, but at the cost of lower precision.

Table 2: Detailed Evaluation Metrics on Test Set

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**	**F2 Score**
Adaboost	0.810	0.683	0.535	0.600	0.559
Voting Classifier	0.810	0.679	0.540	0.602	0.563
Gradient Boost	0.805	0.671	0.524	0.589	0.549
Kernel SVM	0.794	0.659	0.463	0.544	0.493
Logistic Regression	0.805	0.658	0.558	0.604	0.573
Random Forest	0.797	0.651	0.508	0.571	0.531
K-Nearest Neighbours	0.789	0.629	0.504	0.559	0.525
SVM (Linear)	0.789	0.629	0.501	0.557	0.522
Naive Bayes	0.757	0.531	**0.731**	0.616	**0.680**
Decision Tree	0.733	0.497	0.504	0.501	0.503

Key Observations

- **Voting Classifier** and **Adaboost** had the highest F1 and F2 scores, making them the best choices overall.
- **Naive Bayes** achieved the highest recall (0.731) but with the lowest precision, indicating a high number of false positives.
- **Decision Tree** underperformed across all metrics, likely due to overfitting.
- Models like Logistic Regression and Gradient Boosting provided a good balance of precision and recall.

ROC Curve Analysis

Figure 1 compares the ROC curves of all evaluated models. The area under the ROC curve (AUC) serves as a summary metric of each model's ability to distinguish between churn and non-churn customers.

- **Naive Bayes** achieved the highest AUC of **0.749**, indicating strong performance in identifying churners despite its simplicity.
- **Voting Classifier** and **Logistic Regression** followed closely with AUC scores of **0.724** and **0.726**, respectively.
- **Gradient Boosting** and **Adaboost** also performed competitively, with AUC values of **0.716** and **0.723**.
- **Decision Tree Classifier** had the lowest performance with an AUC of **0.660**, likely due to overfitting on the training data.

Overall, ensemble methods (Voting, Boosting) and probabilistic models (Naive Bayes, Logistic Regression) demonstrated superior classification capability compared to single-tree models and K-Nearest Neighbors.

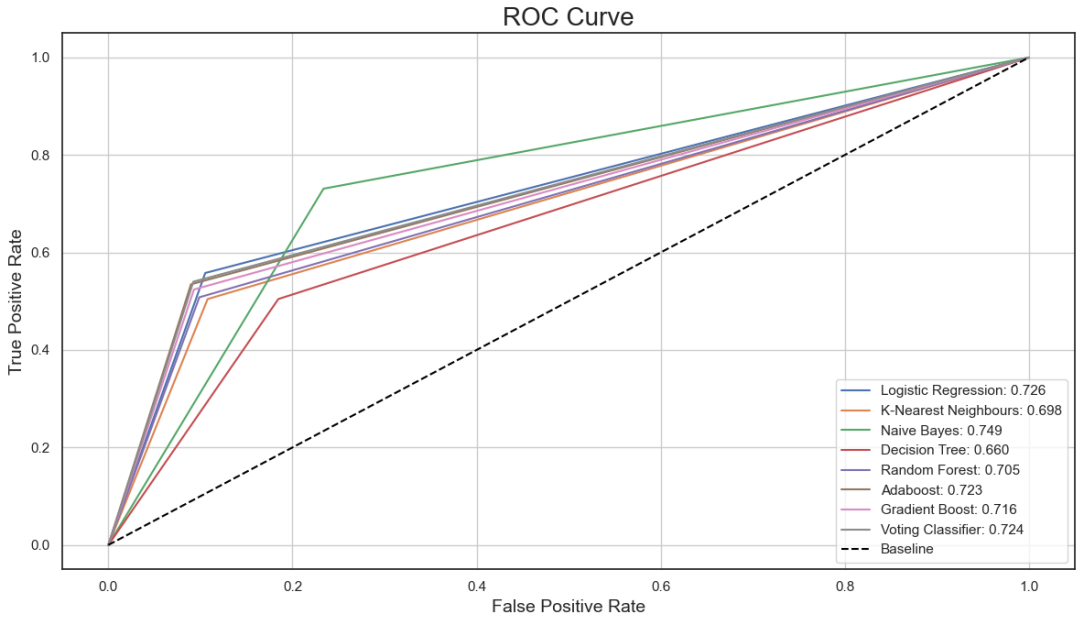
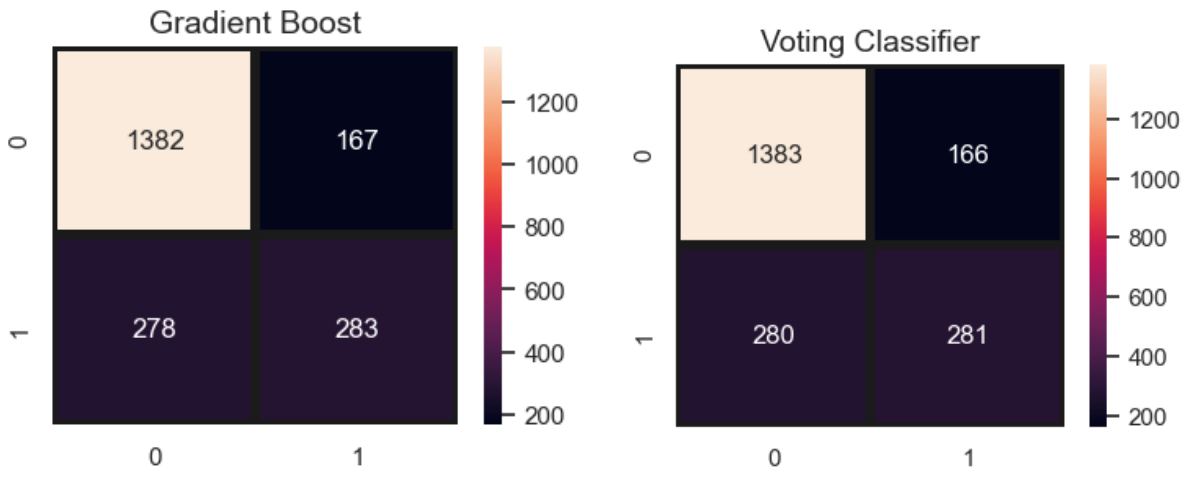
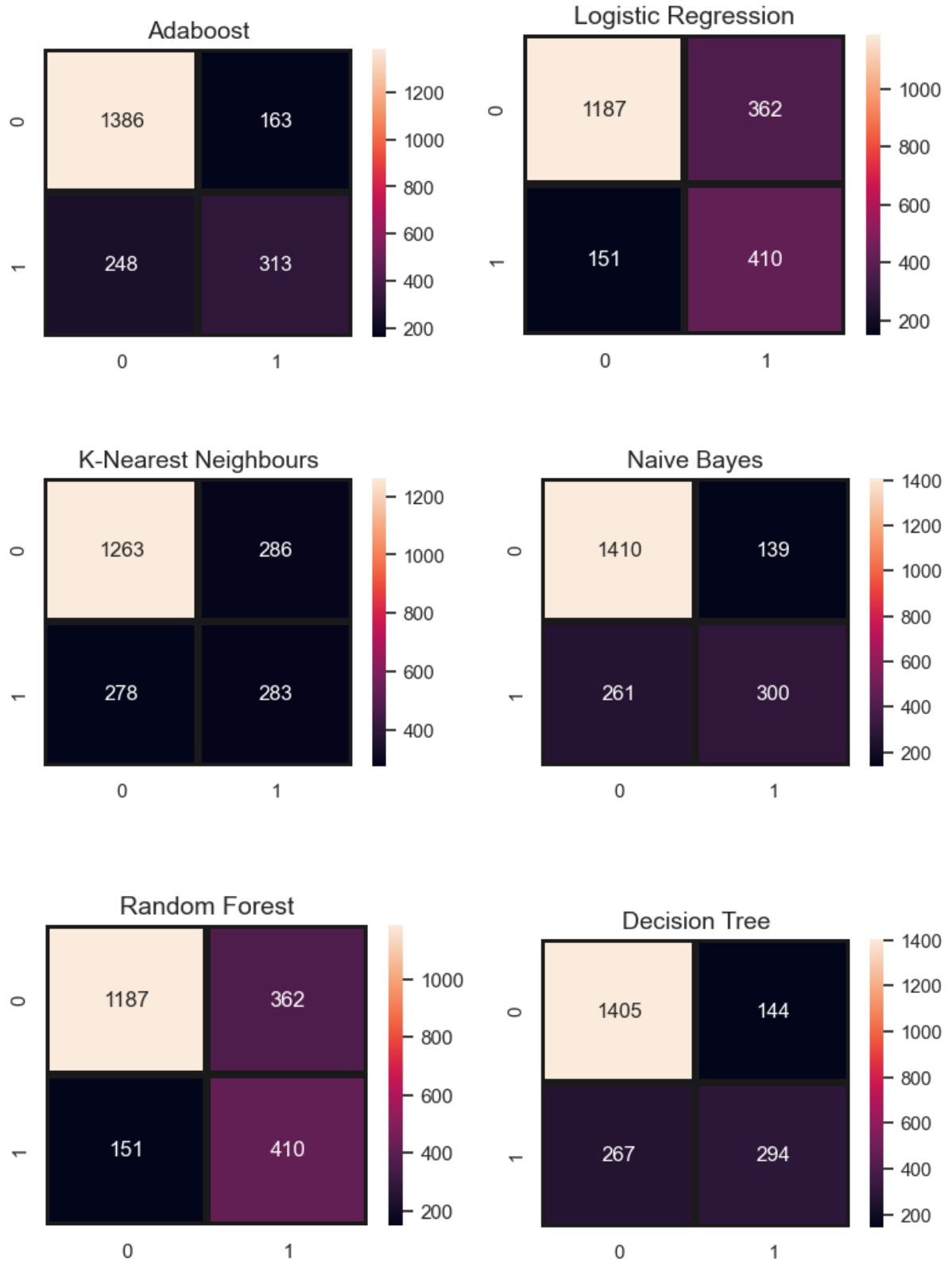


Figure 1: ROC Curve Comparison for All Models





Conclusion

In this project, we successfully developed a machine learning-based Customer Churn Prediction System using the Telco Customer Churn dataset. Through systematic preprocessing, feature engineering, and model evaluation, we explored and compared a range of classification algorithms.

Among all models, ensemble methods such as the Voting Classifier, Gradient Boosting, and Adaboost delivered the highest performance in terms of ROC-AUC, F1, and F2 scores. These models were particularly effective in balancing precision and recall—key for minimizing false negatives in churn prediction.

The final model was deployed through a Streamlit web application, allowing for real-time predictions based on customer input. This provides a practical and interactive tool that can be used by non-technical stakeholders to identify high-risk customers and take proactive retention measures.

Future Work

While the current system performs well, several improvements and extensions can be explored:

- **Explainable AI:** Incorporating SHAP or LIME to interpret model predictions and understand individual churn drivers.
- **Business Integration:** Connecting the system to CRM tools or customer databases for automated alerts or retention campaigns.
- **Real-time Scoring:** Deploying the model as an API for real-time scoring of incoming customer data.
- **Model Monitoring:** Implementing drift detection and model retraining pipelines to maintain performance over time.
- **Expanded Features:** Enriching the dataset with behavioral, transactional, or customer feedback data for improved accuracy.
- **Cost-based Evaluation:** Incorporating financial cost into the evaluation to prioritize retention actions with the highest ROI.

Overall, this system lays the groundwork for a data-driven churn management strategy that can be scaled and enhanced over time.
