Customer Churn Prediction

I. Introduction

The ability to predict customer churn, or the likelihood of a customer discontinuing a service, is an important factor in the success of many businesses. It is generally more cost-effective to retain an existing customer than to acquire a new one, making churn prediction a key business metric for industries ranging from telecommunications and Internet service providers to insurance firms and media service providers. The Customer Churn Prediction System is a sophisticated tool designed to aid in this task, leveraging a suite of advanced data science techniques to predict customer churn with a high degree of accuracy and interpretability.

II. Data Collection and Preprocessing

The data utilized in this project comprises a rich collection of customer attributes. Each row represents a distinct customer, with columns that encapsulate both demographic information such as age and gender, and account-specific details like service subscriptions, contract type, billing method, and monthly and total charges. The target variable is a binary 'Churn' column, indicating whether a customer left within the last month.

The first step in the process involves thorough data preprocessing and standardization. Raw data is often replete with inconsistencies, missing values, or skewed distributions that can adversely impact the performance of machine learning models. To combat this, our preprocessing pipeline includes data cleaning, handling missing values, feature engineering, and data transformation to ensure compatibility with downstream modeling processes.

III. Model Development and Evaluation

To predict customer churn, we employ an ensemble method - the Random Forest algorithm, chosen for its ability to handle non-linearity and class imbalance in the classification problem. The model was developed using an 80-20 data split for training and validation respectively, ensuring no data leakage. Hyperparameters for the Random Forest model were fine-tuned using Grid Search Cross-Validation to prevent overfitting.

Our model yielded an F1 score of 0.62 and an ROC-AUC of 0.85, indicating a robust performance. Further, we utilized feature importance analysis, which revealed the variables most influential in governing customer churn. These insights could prove valuable to stakeholders, guiding business decisions and customer retention strategies.

IV. Model Explainability

To facilitate transparency and interpretability, we integrated Explainable AI modules like Permutation Importance, Partial Dependence Plots, and SHapley Additive exPlanations (SHAP) values.

Permutation Importance measures feature importance by random shuffling of feature values and quantifying the degradation in model performance. Partial Dependence Plots depict how churn probability changes across a specific feature's range. SHAP values, a game-theoretic approach, breaks down the output of the machine learning model, shedding light on individual feature contributions to a particular prediction.

V. Survival Analysis

In addition to churn prediction, we conducted a survival analysis using the Cox-proportional hazard model. This statistical method calculates the expected customer lifetime value, generating a survival curve and a hazard curve that portray customer behavior over time. The insights from survival analysis provide a more holistic understanding of customer churn and can guide effective retention strategies.

VI. Application Development and Deployment

Our solution is not limited to a back-end churn prediction system. We've developed a user-friendly application using Flask, a micro web framework written in Python, to serve as an interface between the model and the end-user. This app takes in customer data, predicts the likelihood of churn, and offers a detailed explanation of the prediction using SHAP values.

The application is hosted on an AWS EC2 instance, with static files stored in an AWS S3 bucket. The front-end user interface was developed using React, a popular JavaScript library known for its efficiency and flexibility in building user interfaces.

The Customer Churn Prediction System represents a comprehensive solution to a prevalent business problem. Its advanced algorithmic backbone, commitment to model transparency, and friendly user interface ensure accurate predictions while maintaining a user-centric focus. By accurately predicting customer churn, this system equips businesses with the insight they need to proactively address customer dissatisfaction, boost customer loyalty, and ultimately enhance their bottom line.

Requirements :

The requirements for the Customer Churn Prediction System project can be classified into functional and non-functional requirements.

Functional Requirements:

Data Collection and Preprocessing:

- Ability to import data from different sources.
- Ability to handle missing data, erroneous entries, and inconsistencies during preprocessing.
- Implementation of feature engineering techniques for generating new features.

Model Development and Evaluation:

- Implementation of an ensemble machine learning model - Random Forest - to predict customer churn.
- Optimization of hyperparameters to achieve the best model performance.
- Evaluation of the model on testing data to assess its performance metrics (F1 Score, ROC-AUC).

Model Explainability:

- Integration of Explainable AI techniques (like SHAP values, Permutation Importance, and Partial Dependence Plots) to understand the contributions of different features to the model's predictions.

Survival Analysis:

- Implementation of the Cox-proportional hazard model for survival analysis.
- Calculation of expected customer lifetime value.

Application Development and Deployment:

- Development of a Flask backend to serve the model predictions.
- Development of a React frontend for user interaction.
- Deployment of the application on a cloud service like AWS.
-

Non-Functional Requirements:

Performance:

- The prediction system must provide results in a reasonable timeframe.
- The system must handle multiple requests simultaneously.

Reliability:

- The system must provide reliable and consistent prediction results.
- The deployed application should have high uptime and low latency.

Usability:

- The user interface should be intuitive and easy to navigate.
- The application should provide clear and understandable prediction explanations.

Security:

- Customer data must be handled securely.
- Deployment on AWS must adhere to best practices for security.

Scalability:

- The system should be designed to accommodate growing amounts of data.
- The application should be able to serve an increasing number of users.

Maintainability:

- The code should be well-structured and appropriately commented to allow for future modifications.
- The application should be designed in a modular way to allow for updates and improvements without major overhauls.

KDD :

Here is the KDD process for your Customer Churn Prediction project:

1. Data Selection: The first step involves collecting data relevant to the problem domain. This might come from your company's databases or external sources. For your project, you have customer data, which includes demographics, account information, and service details.

2. Data Preprocessing: This step addresses data cleaning and transformation. Data cleaning could involve handling missing data, removing outliers, and correcting inconsistent or erroneous data. Data transformation could involve normalization, aggregation, or generalization. For your project, you're preprocessing the raw dataset, performing feature engineering, transforming the data, and selecting the relevant features for your model.

3. Data Transformation: This step includes activities to transform or consolidate the data according to the needs of the chosen data mining algorithm. In your project, you use techniques such as TF-IDF and count vectorizer, and you're creating feature sets including unigrams and bigrams.

4. Data Mining: This is the crucial step where intelligent methods are applied to extract patterns in the data. You're using machine learning techniques, specifically, a Random Forest model for the prediction of customer churn.

5. Evaluation and Interpretation: The patterns and knowledge discovered from the data mining step are evaluated for their relevance and usefulness. Any non-trivial, implicit, previously unknown, and potentially useful patterns are interpreted into knowledge. In your project, you evaluate the model based on the F1 score and ROC-AUC metrics. You also use techniques such as Permutation Importance, Partial Dependence plots, and SHap values for explaining the model's output.

6. Use of Discovered Knowledge: The final step is to use the discovered knowledge in the business process. You're using the model predictions for customer churn and survival analysis insights for understanding customer behavior. These predictions are presented in a user-friendly Flask application, which makes the insights accessible to non-technical stakeholders.

This completes the KDD process for your project. Remember, this process is iterative, meaning you may need to go back and forth between steps based on your findings.

Business Intelligence :

In the Customer Churn Prediction project, the implementation of Business Intelligence (BI) transforms raw data into meaningful insights, enabling businesses to understand customer retention challenges and devise effective strategies to address them. Here our customer is a Telecom service provider who is offering different services to the customer here is how our BI is integrated into various aspects:

Descriptive Business Intelligence: By analyzing the historical data of customer churn rates and identifying trends, we gain insights into past customer behavior. This process allows us to understand which customer segments have been experiencing higher churn rates and when these churn events typically occur.

Diagnostic Business Intelligence: The churn prediction model's explainability features offer an understanding of why customers are churning. If certain services or demographics are strongly associated with higher churn rates, these factors indicate areas where the business can potentially improve to retain customers.

Predictive Business Intelligence: The churn prediction model is a pivotal part of our predictive BI strategy. It forecasts future churn rates, providing an opportunity for the business to anticipate and proactively manage customer churn.

Prescriptive Business Intelligence: The insights derived from descriptive, diagnostic, and predictive analysis empower telecom providers with effective strategies to mitigate customer churn. For instance, if a demographic is highly likely to churn, they can propose targeted retention strategies to engage these customers effectively.

Data Visualization and Reporting: An integral part of BI is presenting the data in a digestible format. Our Flask app's dashboard visualizes key metrics and insights, such as churn rates over time, high-risk customer segments, and the impact of various factors on churn. This visualization aids in simplifying complex data for non-technical stakeholders and facilitates informed decision-making.

Real-time Business Intelligence: Our system is designed to update predictions with incoming data, providing real-time insights into customer churn. This feature allows the business to swiftly react and implement retention strategies.

Through this BI approach, the Customer Churn Prediction System not only predicts customer churn but also provides valuable insights to inform and optimize business strategies. By understanding the factors contributing to customer churn, businesses can make data-driven decisions, leading to improved customer retention and increased profitability.

Data Science Algorithms & Features Used

The Customer Churn Prediction System project has been developed using a blend of data science algorithms and feature engineering techniques. Each algorithm and feature plays a vital role in creating a robust prediction system that can forecast customer churn with high accuracy, while also explaining the underlying reasons for the churn.

Modeling with a Tree-based Ensemble Method:

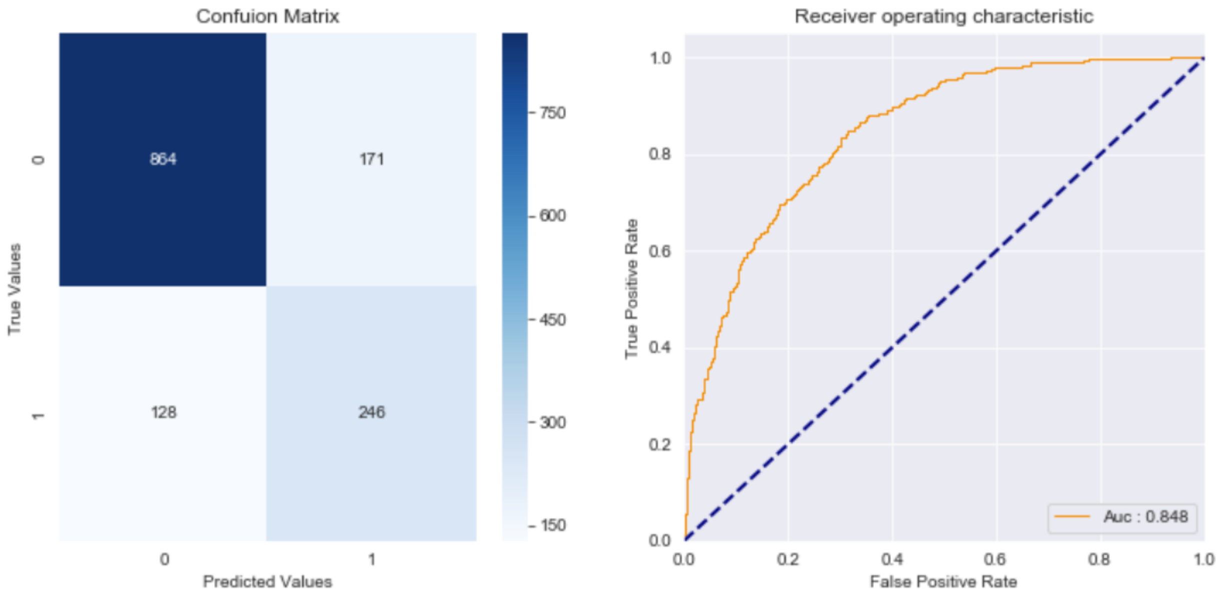
The backbone of the churn prediction system is the Random Forest model, a popular tree-based ensemble method. Unlike simpler algorithms that might assume linearity or other specific relationships between features and the target variable, Random Forest makes no such assumptions, making it an apt choice for this complex classification problem.

The ensemble approach of the Random Forest model combines the predictions of multiple decision trees to make a final prediction, reducing the chances of overfitting. This model was trained on 80% of the available data, and the remaining 20% was used for validation, ensuring the model's predictions were tested on unseen data to prevent data leakage.

Due to the inherent class imbalance present in the data (with a ratio of 1:3), an important step was to assign a class weightage of 1:3. This implies that false negatives (customers who were predicted not to churn but did) are considered three times costlier than false positives (customers who were predicted to churn but didn't). This approach ensures the model pays more attention to the minority class, thus improving its ability to predict churn.

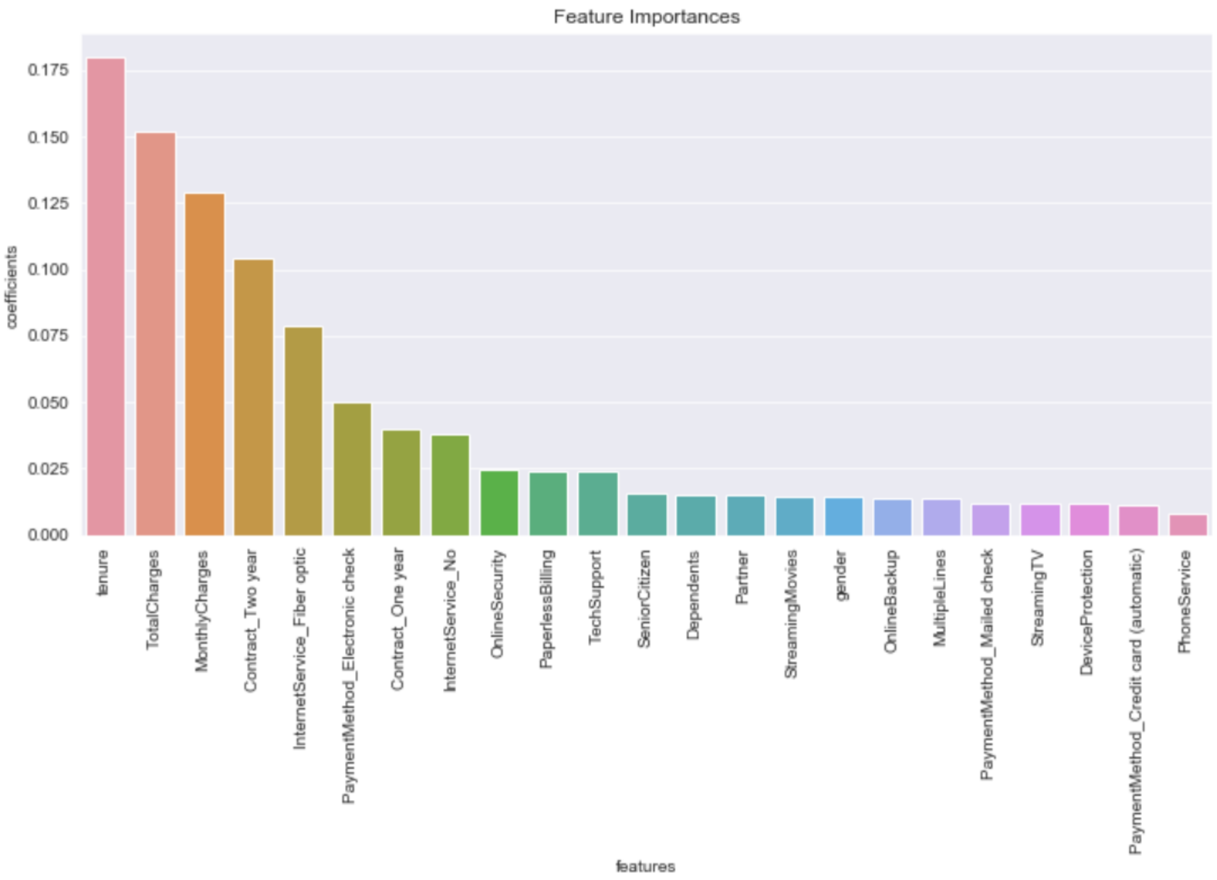
Hyperparameter Tuning:

Hyperparameter tuning is an essential process in machine learning to fine-tune the model's performance. In the case of the Random Forest model, several hyperparameters, such as the number of decision trees (n_estimators), the maximum depth of the trees (max_depth), and the minimum number of samples required to split an internal node (min_samples_split), were tuned. The tuning was performed using Grid Search Cross Validation to ensure the model did not overfit the data. The final model achieved a promising F1 score of 0.62 and a ROC-AUC of 0.85, indicating its high accuracy in predicting customer churn.



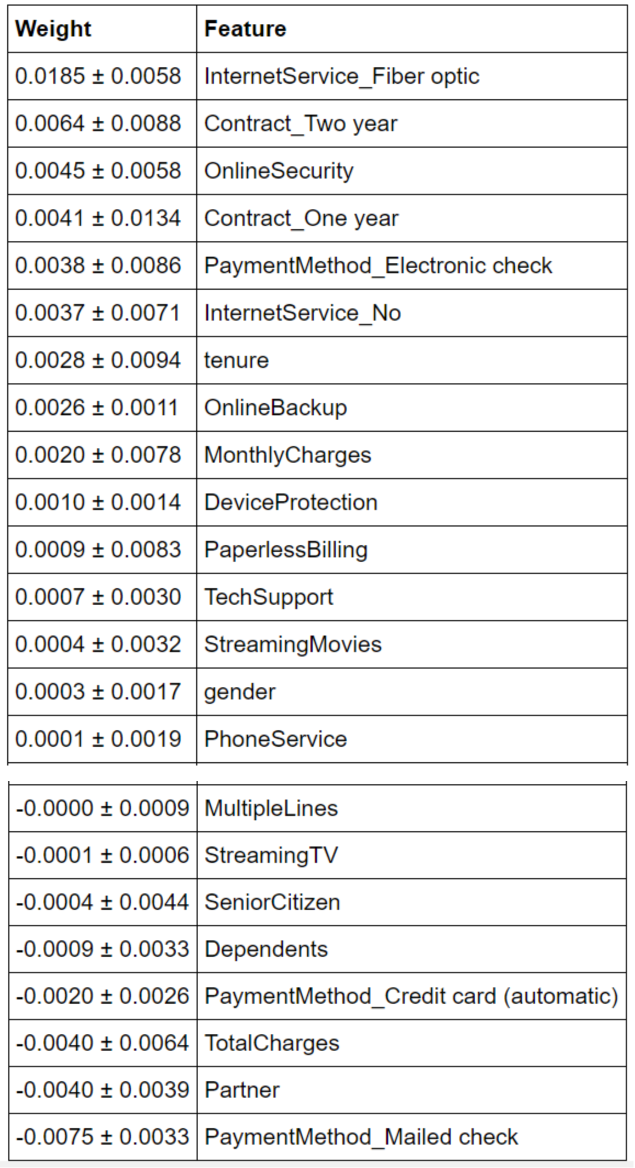
Feature Importance and Model Explainability:

Interpretability is crucial when using machine learning models to make real-world decisions. Thus, to understand the factors governing customer churn, the feature importance plot was generated. This plot ranked the features based on their importance in making accurate predictions, helping to identify the significant influencers of customer churn.



To further enhance the model's interpretability, Explainable AI modules such as Permutation Importance, Partial Dependence plots, and Shapley Additive Explanation (SHAP) values were used:

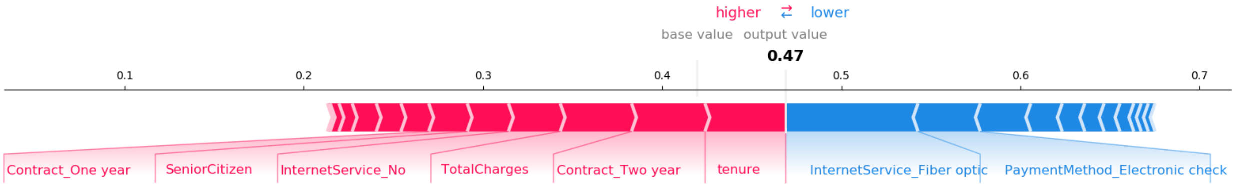
Permutation Importance: This technique measures how much the model's performance decreases when the values of a particular feature are randomly shuffled. The idea is that if a feature is important, then randomly changing its values should significantly degrade the model's performance.



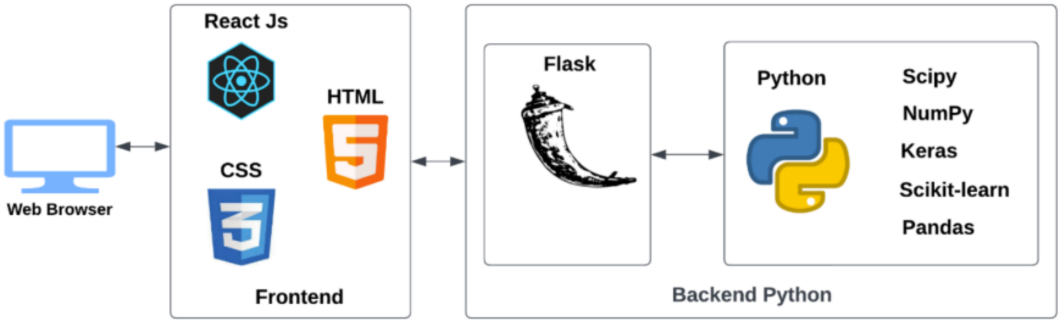
Weight	Feature
0.0185 ± 0.0058	InternetService_Fiber optic
0.0064 ± 0.0088	Contract_Two year
0.0045 ± 0.0058	OnlineSecurity
0.0041 ± 0.0134	Contract_One year
0.0038 ± 0.0086	PaymentMethod_Electronic check
0.0037 ± 0.0071	InternetService_No
0.0028 ± 0.0094	tenure
0.0026 ± 0.0011	OnlineBackup
0.0020 ± 0.0078	MonthlyCharges
0.0010 ± 0.0014	DeviceProtection
0.0009 ± 0.0083	PaperlessBilling
0.0007 ± 0.0030	TechSupport
0.0004 ± 0.0032	StreamingMovies
0.0003 ± 0.0017	gender
0.0001 ± 0.0019	PhoneService
-0.0000 ± 0.0009	MultipleLines
-0.0001 ± 0.0006	StreamingTV
-0.0004 ± 0.0044	SeniorCitizen
-0.0009 ± 0.0033	Dependents
-0.0020 ± 0.0026	PaymentMethod_Credit card (automatic)
-0.0040 ± 0.0064	TotalCharges
-0.0040 ± 0.0039	Partner
-0.0075 ± 0.0033	PaymentMethod_Mailed check

Partial Dependence Plots: These plots illustrate how the probability of churn changes across the range of a particular feature while keeping all other features constant. For instance, if tenure is an important feature, a partial dependence plot for tenure could show how to churn probability decreases as tenure increases.

Shapley Additive Explanation (SHAP) Values: SHAP values offer a game-theoretic approach to explain the output of machine learning models. They help understand the contribution of each feature to the prediction for each instance. Using SHAP values, we can identify which features are driving a particular customer's churn probability.



Client slide design :



React Application:
Home Component: This will serve as your landing page, containing a brief introduction to the service and its benefits. It could also include navigation to other parts of the application.

Data Input Component: This will be a form to collect user data. The form fields will correspond to the features your model requires for prediction. Once the user submits this form, it will trigger a POST request to the Flask API with the input data.

Prediction Results Component: After receiving the prediction response from the Flask API, this component will display the prediction results. It could display the probability of churn, the most influential factors, and the survival analysis.

About Component: This component provides information about the project, the technologies used, and the people behind it.

Flask API (Middleware between React Application and Machine Learning Model):

Data Preparation Endpoint: This endpoint will receive the POST request with the user input data from the React application, preprocess and standardize it as required by the model.

Prediction Endpoint: This endpoint will use the preprocessed data to run the prediction with your machine learning model, and return the prediction results. It will also calculate and return the SHAP values and survival analysis results.

Model Explainability Endpoint: This endpoint could provide further details about the model's decision using the SHAP values and survival analysis results, which will be displayed in the Prediction Results Component.

AWS Deployment:

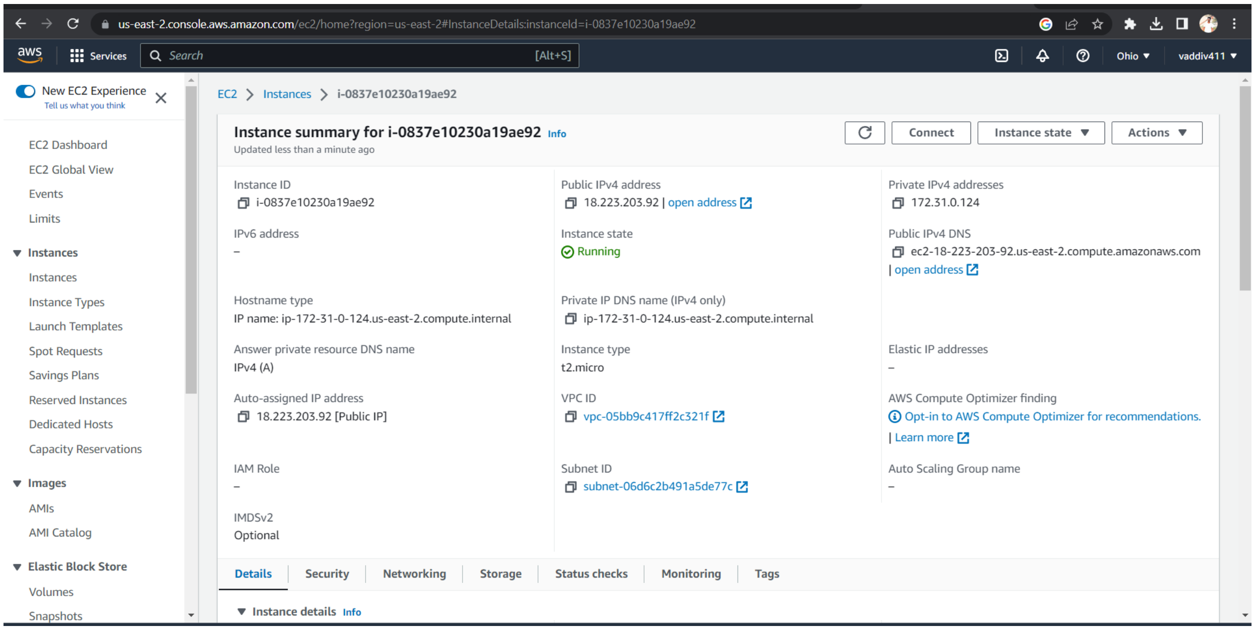
AWS EC2: Deploy your Flask API and Machine Learning Model on an AWS EC2 instance. EC2 provides secure, resizable compute capacity in the cloud and will host your Flask application.

Model Deployment with a Flask App:

The project deployed the tuned Random Forest model using a Flask web application into AWS.

this project serves as a user interface that showcases the churn probability, severity of churn, and SHAP values based on a customer's data. It enables a non-technical user to input customer data and receive an immediate churn prediction, accompanied by an explanation of the prediction. This application turns the churn prediction model into a usable tool that can be used directly by business stakeholders to make informed decisions.

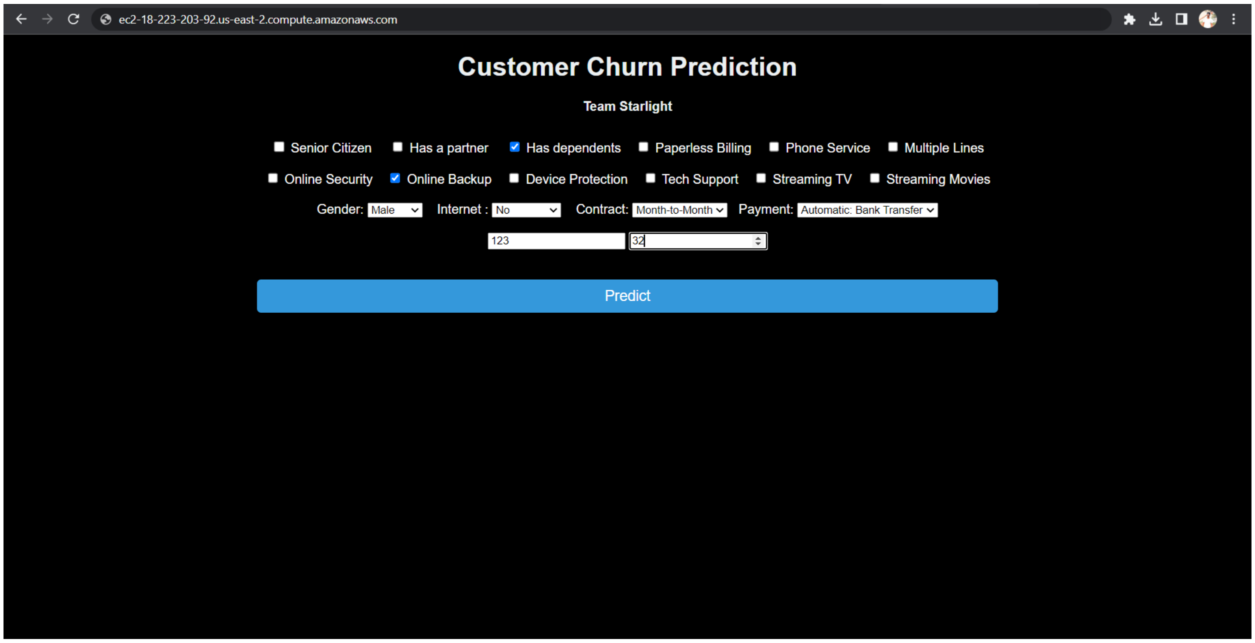
One of the key features of the Flask app is its integration with the SHAP library. The app uses the SHAP values calculated by the model to visually explain the reasons behind each churn prediction. This makes the model's decisions transparent, enabling stakeholders to understand why a particular customer is predicted to churn.



Instance summary for i-0837e10230a19ae92 Info

Updated less than a minute ago

Instance ID i-0837e10230a19ae92	Public IPv4 address 18.223.203.92 | open address	Private IPv4 addresses 172.31.0.124
IPv6 address –	Instance state Running	Public IPv4 DNS ec2-18-223-203-92.us-east-2.compute.amazonaws.com | open address
Hostname type IP name: ip-172-31-0-124.us-east-2.compute.internal	Private IP DNS name (IPv4 only) ip-172-31-0-124.us-east-2.compute.internal	
Answer private resource DNS name IPv4 (A)	Instance type t2.micro	Elastic IP addresses –
Auto-assigned IP address 18.223.203.92 [Public IP]	VPC ID vpc-05bb9c417ff2c321f	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role –	Subnet ID subnet-06d6c2b491a5de77c	Auto Scaling Group name –
IMDSv2 Optional		



Customer Churn Prediction

Team Starlight

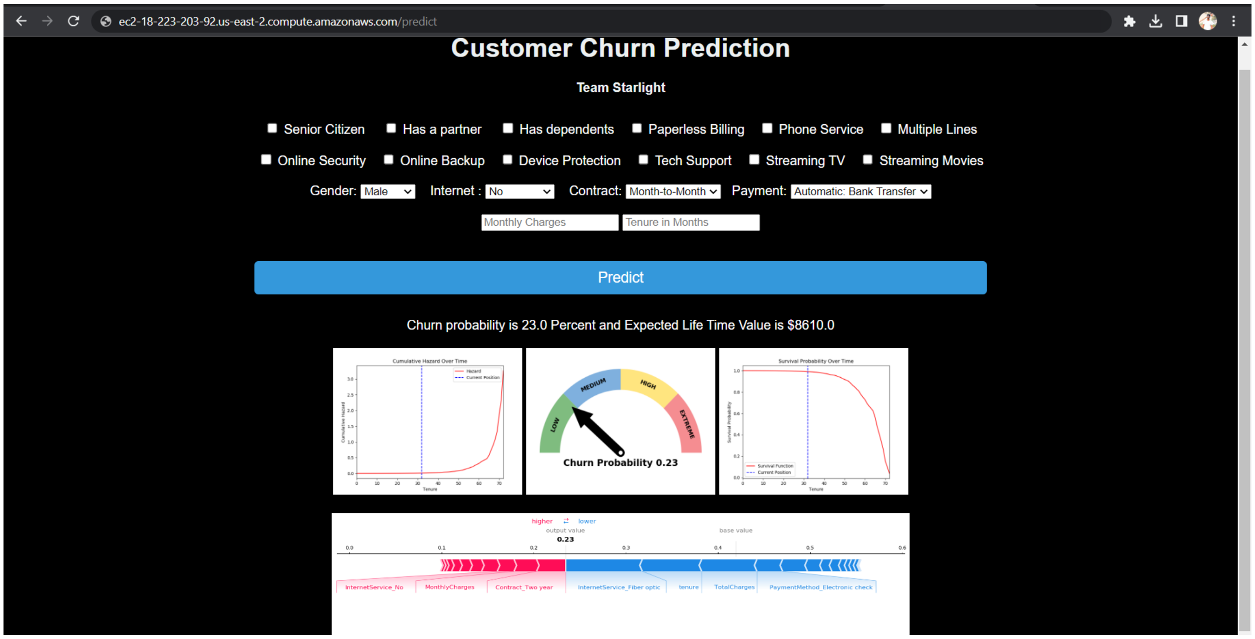
☐ Senior Citizen ☐ Has a partner ☑ Has dependents ☐ Paperless Billing ☐ Phone Service ☐ Multiple Lines

☐ Online Security ☑ Online Backup ☐ Device Protection ☐ Tech Support ☐ Streaming TV ☐ Streaming Movies

Gender: Male Internet : No Contract: Month-to-Month Payment: Automatic: Bank Transfer

123 32

Predict



Customer Churn Prediction

Team Starlight

□ Senior Citizen □ Has a partner □ Has dependents □ Paperless Billing □ Phone Service □ Multiple Lines

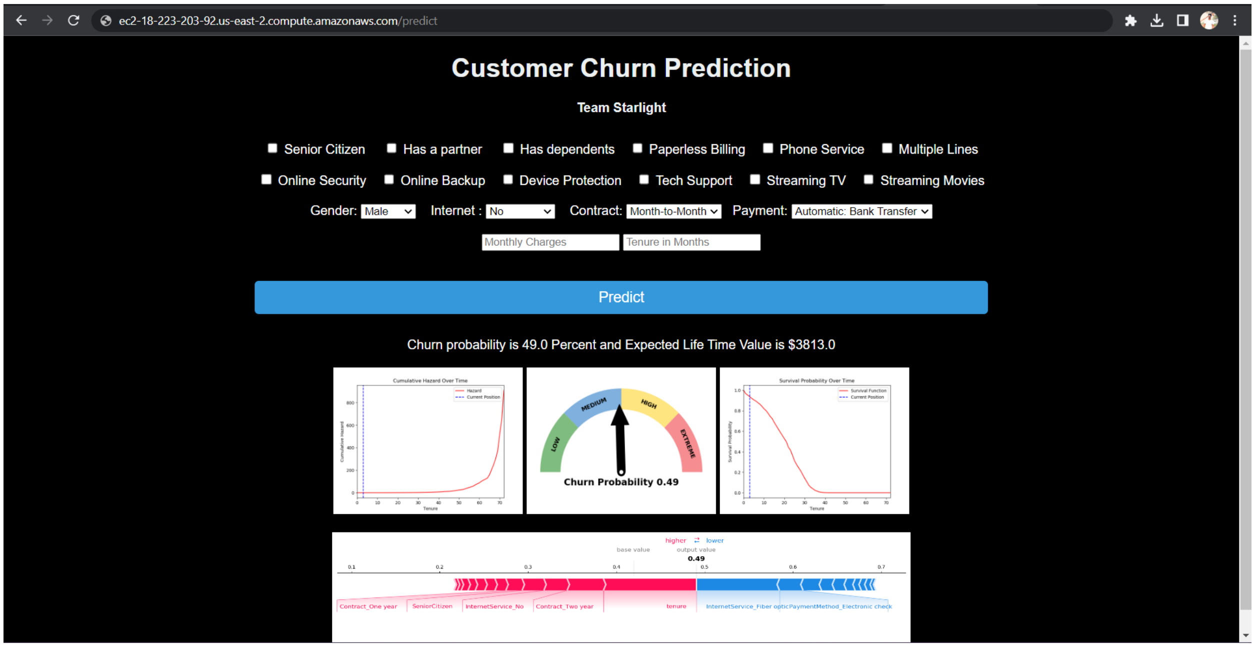
□ Online Security □ Online Backup □ Device Protection □ Tech Support □ Streaming TV □ Streaming Movies

Gender: Male Internet : No Contract: Month-to-Month Payment: Automatic: Bank Transfer

Monthly Charges Tenure in Months

Predict

Churn probability is 23.0 Percent and Expected Life Time Value is $8610.0



Customer Churn Prediction

Team Starlight

□ Senior Citizen □ Has a partner □ Has dependents □ Paperless Billing □ Phone Service □ Multiple Lines

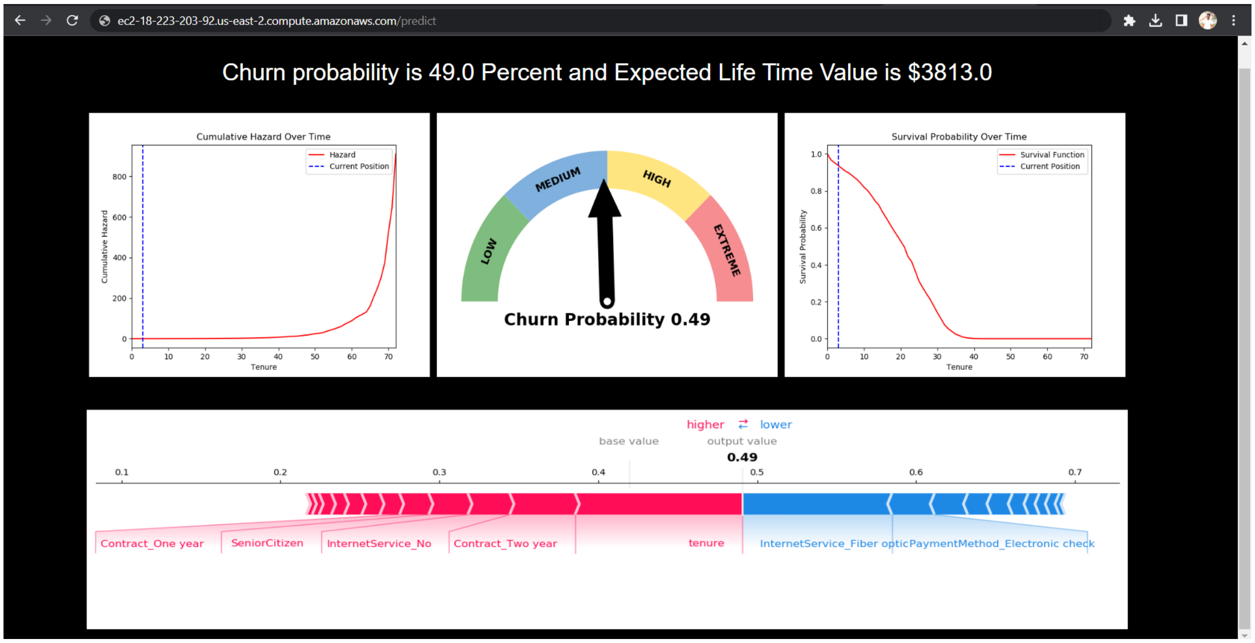
□ Online Security □ Online Backup □ Device Protection □ Tech Support □ Streaming TV □ Streaming Movies

Gender: Male Internet : No Contract: Month-to-Month Payment: Automatic: Bank Transfer

Monthly Charges Tenure in Months

Predict

Churn probability is 49.0 Percent and Expected Life Time Value is $3813.0



ec2-18-223-203-92.us-east-2.compute.amazonaws.com/predict

Churn probability is 49.0 Percent and Expected Life Time Value is $3813.0

Conclusion:

The Customer Churn Prediction System is a comprehensive solution that uses a suite of data science algorithms and features to predict customer churn with high accuracy and interpretability. The system provides valuable insights into customer behavior, enabling businesses to understand the significant factors influencing churn and devise data-driven strategies to enhance customer retention. The integration of the model into a user-friendly Flask app ensures that the insights provided by the model can be easily accessed and used by business stakeholders. Furthermore, the addition of survival analysis provides a deeper understanding of customer behavior over time, complementing the churn prediction model and providing a holistic view of customer churn.