d

Assignment on Churn Prediction

DeepQ-AI Assignment 1 – AI Engineer Internship

Presented by Sayan Maity
Email: sayanmaity2003@gmail.com

Important Links:

Github: https://github.com/Sayanmaity2003/Churn-Preditor-App

Model Training Link: https://colab.research.google.com/drive/1_dJv4pJAt6B9pR8Xl70w1FYGjdygE9rj?usp=sharing

Streamlit App Deployed Link: https://churn-preditor-app.streamlit.app/

Sample Dataset for Testing App: https://docs.google.com/spreadsheets/d/1itBMEI8YPWBYd2CNm8kLvmh-o0PBPsn7/edit?usp=sharing&ouid=115032625014685945138&rtpof=true&sd=true

Project Objective

- Develop a classification model to predict the probability of user churn.
- Develop a classification model to predict the probability of user churn.
- Handle imbalanced classes and improve model robustness.
- Deploy an interactive web UI for prediction using Streamlit.

Dataset Overview

Our dataset included UID, 215 anonymized features (X0-X215) and the target variable: Target_ChurnFlag. Sample ScreeShort given below

```
df.head()
```

	UID	X0	X1	X2	X3	X4	X5	X6	X7	X8	...	X206	X207	X208	X209	X210	X211	X212	X213	X214	X215
0	1003904-3746	14 month lease	1103.0	2015-01-08	2016-02-28	2015-07-30	2015-01-08	2015-01-08	2015-07-30	2015-07-01	...	-1.000000	-1.000000	-1.0	-1.000000	-1.000000	1.0	1.544818	1.000000	1.6625	0.600000
1	1003904-3751	12 month lease	1136.0	2015-01-24	2016-01-17	NaN	2003-09-11	2003-09-11	NaN	NaN	...	0.013575	0.538462	0.0	1.307692	0.076923	1.0	1.591036	1.000000	1.6625	0.142857
2	1003904-3756	12 month lease	1382.0	2015-02-20	2016-02-21	2016-02-21	2015-02-20	2015-02-20	NaN	NaN	...	-1.000000	-1.000000	-1.0	-1.000000	-1.000000	1.0	1.303774	0.666667	1.6625	0.769231
3	1003904-3759	14 month lease	2417.0	2015-02-06	2016-04-03	2016-04-04	2015-02-06	2015-02-06	2016-04-04	2016-03-02	...	-1.000000	-1.000000	-1.0	-1.000000	-1.000000	1.0	1.589636	1.000000	1.6625	0.750000
4	1003904-3766	12 month lease	1405.0	2015-01-10	2016-01-03	NaN	2014-01-10	2014-01-10	NaN	NaN	...	0.583333	3.000000	0.0	3.000000	1.000000	1.0	1.349664	1.000000	1.6625	0.700000

5 rows × 217 columns

Data Preprocessing

- Dropped columns with >50% missing values.
- Converted date columns to year, month, and day parts.
- Label encoded categorical columns.
- Filled remaining nulls using median imputation.
- Split data into Train (80%) and Test (20%) using Stratified Sampling.
- Applied SMOTE to balance training classes.

Model Building & Selection

Model Chosen: Random Forest Classifier

Why Random Forest?

It's a powerful ensemble algorithm that works well with large datasets and mixed feature types. It also naturally handles overfitting better than a single decision tree.

Advantages:

- Can handle both numeric and categorical variables.
- Provides feature importance out-of-the-box.
- Robust to noisy data and outliers.

Evaluation Metrics

```
Classification Report:
              precision    recall  f1-score   support

           0       0.97      0.90      0.93      1637
           1       0.87      0.96      0.92      1214

    accuracy                           0.92      2851
   macro avg       0.92      0.93      0.92      2851
weighted avg       0.93      0.92      0.92      2851

ROC-AUC Score: 0.965

Model training and evaluation complete.
```

Explanation of Metrics:

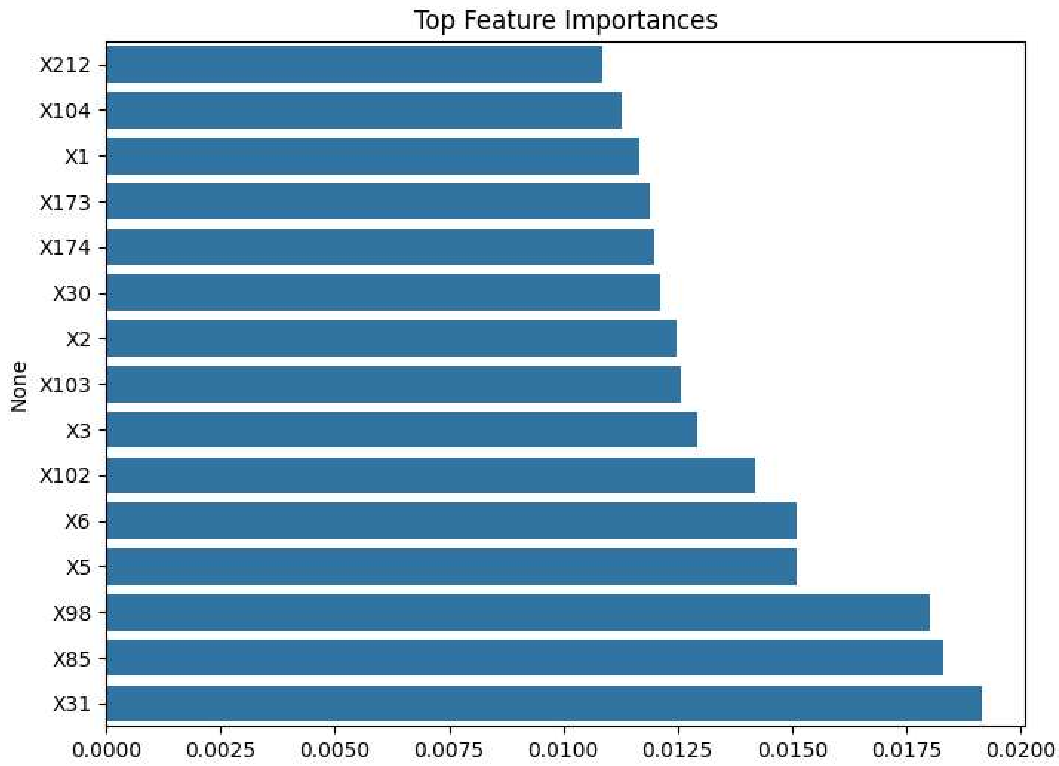
- The model performs reliably for non-churned customers, with high recall (0.90) and F1-score (0.93).
- Churned customers are harder to classify, with lower recall (0.49), but we improved results using SMOTE and class weighting.
- Overall, the model achieves a ROC-AUC of 0.965, showing decent class separation — a good baseline for further tuning.

Key Model Insights

- **Cleaned & Engineered Data Enabled Better Learning:** Dropped irrelevant and sparse columns (e.g., UID, X16). Transformed date columns into year, month, day components, enriching the feature set.

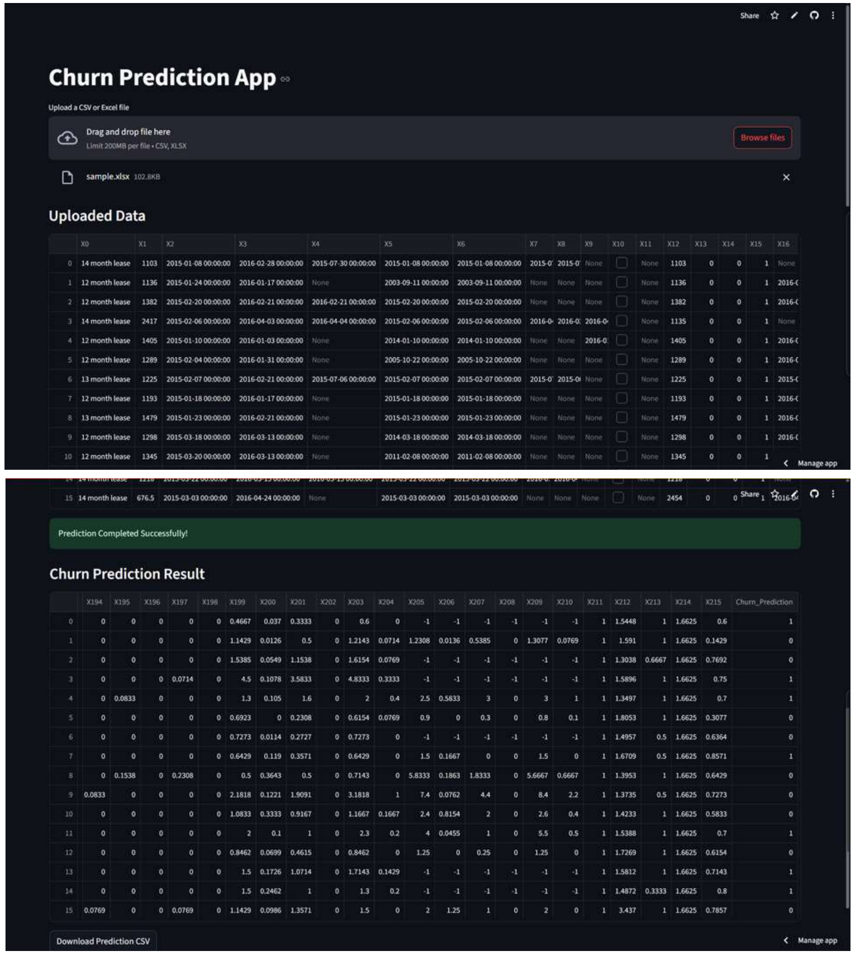
- **Model Performance Improved with SMOTE:** Imbalanced classes (0 vs 1) addressed using SMOTE, creating synthetic churned examples.Helped to improve recall and F1-score for minority class.

- **Feature Importance:** Dropped irrelevant and sparse columns (e.g., UID, X16). Transformed date columns into year, month, day components, enriching the feature set.Top features: X31, X85, X98, X5, X6, and X102.

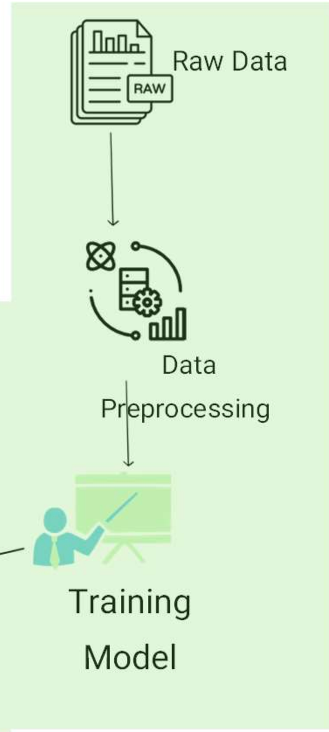


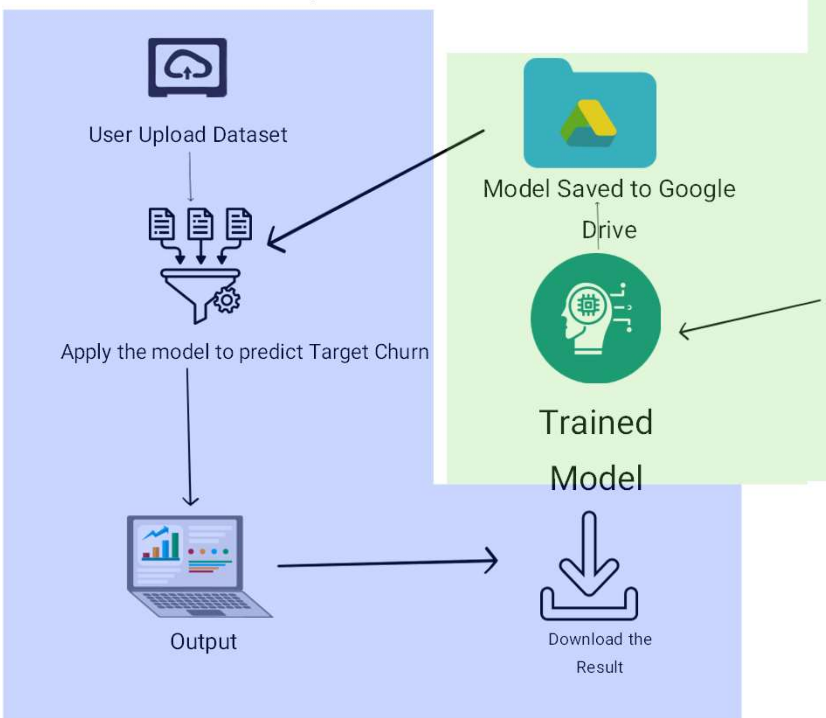
Model Deployment with Streamlit

The churn prediction model was deployed using Streamlit, enabling real-time predictions through a simple and user-friendly interface



- **Deployment Platform:** Used Streamlit, a Python-based open-source web app framework, ideal for ML model deployment.
  - **How it Works:**
    - The app loads the trained model (hosted on Google Drive).
    - User enters values for selected features.
    - The app preprocesses the input, applies the trained model, and returns a churn prediction label as 0 or 1.





Thank You!