AWS ML Engineer Nanodegree

Capstone Proposal

Svetlana Bogdanova

December 13, 2024

Domain Background

Subscription-based businesses have become increasingly popular across industries, including entertainment, software, fitness, and e-commerce. However, one major challenge these businesses face is churn due to billing failures. Billing failures can occur for various reasons, such as expired payment methods, insufficient funds, or technical errors, and often lead to unintended subscription cancellations. According to industry reports, involuntary churn due to billing issues accounts for up to 20-40% of total subscription cancellations, representing a significant revenue loss for businesses. [1]

Academic research supports the application of ML in addressing billing-related issues. Studies in predictive analytics and churn modeling demonstrate how machine learning techniques, such as classification algorithms and time-series analysis, can enhance customer retention strategies [2].

This project proposes using ML predictions to address the problem of stopped subscriptions due to billing failures

Problem Statement

Billing failures in subscription-based models lead to significant revenue loss. Identifying which transactions are likely to recover after an initial billing failure is crucial for minimizing churn. The challenge is to develop a machine learning model that accurately predicts recovery likelihood, optimizing billing retries and reducing unnecessary operational costs.

Datasets and Inputs

The primary dataset for this project is the Bank Customer Churn Dataset [3], which provides information about customer demographics, account behavior, and churn status. While the dataset is designed to predict customer churn in banking, it will be adapted to represent a subscription context by modifying the target variable to indicate recovery outcomes after billing failure.

This table summarizes the input features, dropped fields, and synthesized fields used in the subscription churn prediction project. It highlights the actions taken to prepare the dataset to align with real-world constraints.

Category	*Field Name*	*Description*	*Action*
Input Features	**country**	Customer's geographical region (categorical).	Retained
Input Features	**gender**	Customer's gender (categorical).	Retained
Input Features	**age**	Customer's age (numeric).	Retained
Input Features	**tenure**	Duration of the customer relationship (numeric).	Retained
Input Features	**products_number**	Number of products held by the customer (numeric).	Retained
Input Features	**active_member**	Whether the customer is actively engaged (binary).	Retained
Input Features	**amount_due**	Amount owed during billing failures (numeric, repurposed from balance).	Repurposed
Input Features	**billing_failure_count**	Number of recent billing failures (numeric, synthesized).	Synthesized
Input Features	**engagement_score**	Overall customer engagement score, derived from tenure, products_number, and active_member.	Synthesized
Input Features	**subscription_tier**	Subscription level (e.g., Basic, Standard, Premium), based on estimated_salary (before dropping).	Synthesized
Dropped Fields	**customer_id**	Customer identifier, not useful for modeling.	Dropped
Dropped Fields	**credit_score**	Credit score, sensitive data unavailable in transactional systems.	Dropped
Dropped Fields	**estimated_salary**	Estimated salary, sensitive data unavailable in transactional systems.	Dropped

Data preprocessing steps will include feature selection, transformation, cleaning, and the generation of additional features related to subscription recovery (e.g., billing history, engagement metrics).

Feature engineering approach:

```
# Generate engagement score
df['engagement_score'] = (
    0.4 * df['tenure'] / df['tenure'].max() +  # Scale tenure
    0.4 * df['products_number'] / df['products_number'].max() +  # Scale
products_number
    0.2 * df['active_member']  # Active member contributes directly
) * 100  # Scale to a range of 0-100

# Define tiers based on salary percentiles
salary_bins = np.percentile(df['estimated_salary'], [33, 66])
df['subscription_tier'] = pd.cut(
    df['estimated_salary'],
    bins=[-np.inf, salary_bins[0], salary_bins[1], np.inf],
    labels=['Basic', 'Standard', 'Premium']
)

import numpy as np

# Normalize credit_score to range [0, 1]
normalized_credit_score = (df['credit_score'] - df['credit_score'].min())
/ (df['credit_score'].max() - df['credit_score'].min())

# Simulate billing_failure_count
df['billing_failure_count'] = np.random.poisson(
    lam=3 * (1 - normalized_credit_score) + 0.5 * (df['amount_due'] >
df['amount_due'].median()),
    size=len(df)
)
```

The dataset will be split into training, validation, and test sets for model development and evaluation

Solution Statement

This project will apply machine learning techniques to predict customer recovery likelihood after a billing failure. Using a repurposed bank customer churn dataset, we will simulate billing failure scenarios and train models to predict the likelihood of recovery. We will first develop a benchmark model (**logistic regression**) and then apply **AutoML** techniques to identify the best-performing model. The solution will help subscription-based businesses target recovery efforts efficiently, saving resources and improving customer retention

Benchmark Model

The benchmark model for this project is a **Logistic Regression classifier**. Logistic regression is a widely used, interpretable algorithm for binary classification problems and serves as a standard approach in domains such as customer churn prediction. The model will

predict the likelihood of recovery after a billing failure based on the adapted features from the Bank Customer Churn Dataset.

The benchmark model will provide a baseline for comparison with more complex models, which will be trained using **AutoML techniques**.

Evaluation Metrics

The model's performance will be evaluated using the following metrics:

Accuracy: Overall correct predictions.

Precision: Correctly predicted recoveries among all predicted recoveries.

Recall: True recoveries identified among all actual recoveries.

F1-Score: The harmonic mean of precision and recall, balancing both.

AUC-ROC: The area under the receiver operating characteristic curve, indicating the model's ability to distinguish between recoverable and non-recoverable transactions.

These metrics will be used to compare the benchmark model with AutoML-generated models and identify the best-performing solution.

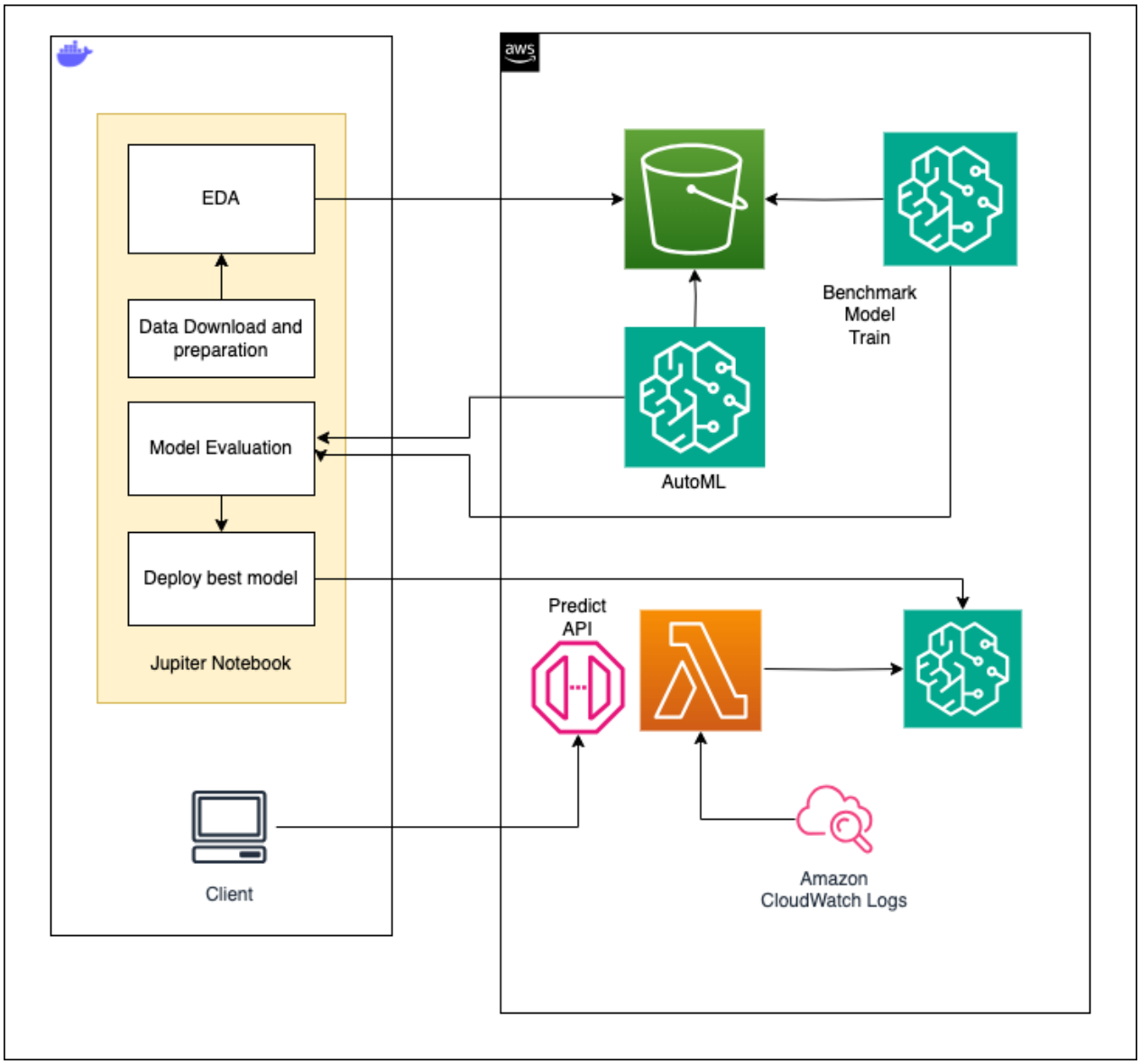
Project Design

The project will follow a structured workflow:

1. **Data Preparation**: Clean and preprocess the dataset, including feature selection, transformation, and splitting into training, validation, and test sets. It is done in the local Jupiter Notebook on local machine
2. **Exploratory Data Analysis** (EDA): Visualize and analyze the data to understand distributions, relationships, and potential issues like class imbalance. It is done in the local Jupiter Notebook on local machine
3. **Benchmark Model Development**: Train a logistic regression model and evaluate its performance using the defined metrics. Training will be done in SageMaker
4. **AutoML Model Development**: Use AutoML (AutoGluon) to automatically train and tune multiple models, optimizing performance. Hyperparameter tuning and training will be done in SageMaker
5. **Model Evaluation and Comparison**: Compare the performance of the benchmark and AutoML models, selecting the best model for deployment.Done in the local Jupiter Notebook
6. **Deployment Simulation**: The best model would function in a real-world subscription recovery scenario, inference API will trigger lambda call in realtime.

The project will apply data handling techniques such as class balancing, feature engineering, and hyperparameter tuning to ensure robust model development and evaluation.

Transaction recovery solution architecture



Resources

[1] Understanding Involuntary Churn and How to Address It

[2] Classifying variety of customer's online engagement for churn prediction with mixed-penalty logistic regression

[3] Bank Customer Churn Dataset