



# Predictive Modeling of NPA Accounts in Retail Lending

Final Submission - PSB
Hackathon Series 2025

Tailwinders

# Problem Statement

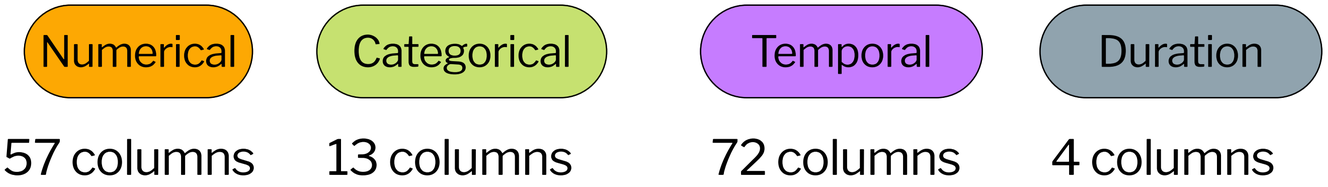
- **Phase-1 Objective – Predictive Modeling**
  - Predict retail loan accounts likely to become NPAs (SMA-2 stage) in the next month.
  - Binary classification using customer demographics, credit bureau scores, and transaction patterns.
  - **Goal:** Early detection to minimize credit losses and optimize recovery efforts.

- **Phase-2 Objective – Digital Tracking Extension**
  - Track identified high-risk accounts using public digital footprints (LinkedIn, Twitter, UPI trends).
  - Enhance risk profiling with real-time behavioral signals.
  - Ensure data privacy with anonymized, compliant data usage.

# Data Overview

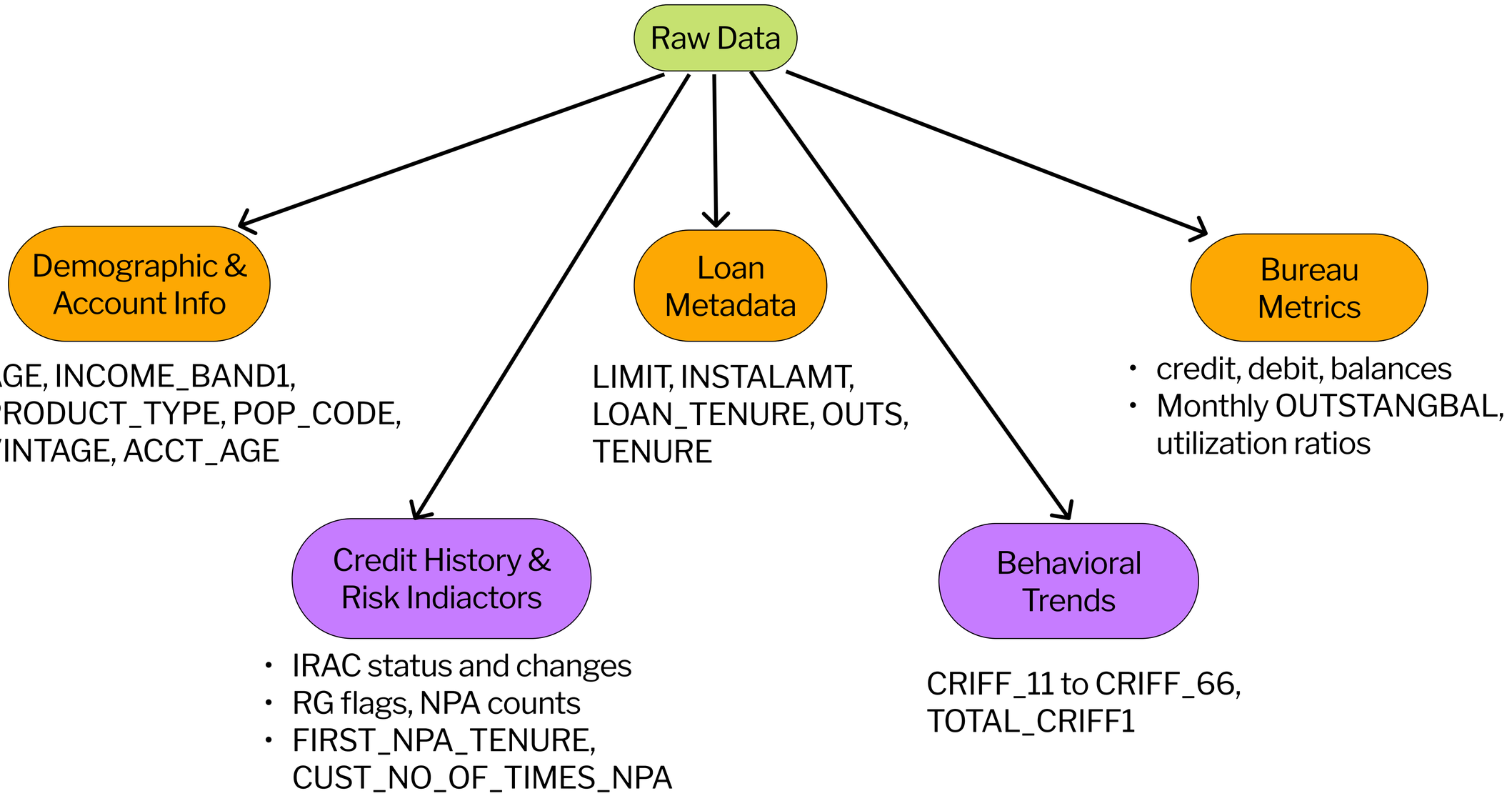
The data present can be briefly presented as follows:

- Dataset Dimensions: 330,000 rows × 100+ columns
- Target Column: TARGET (Binary: 0 = Non-defaulter, 1 = Defaulter)
- Major Datatypes are:



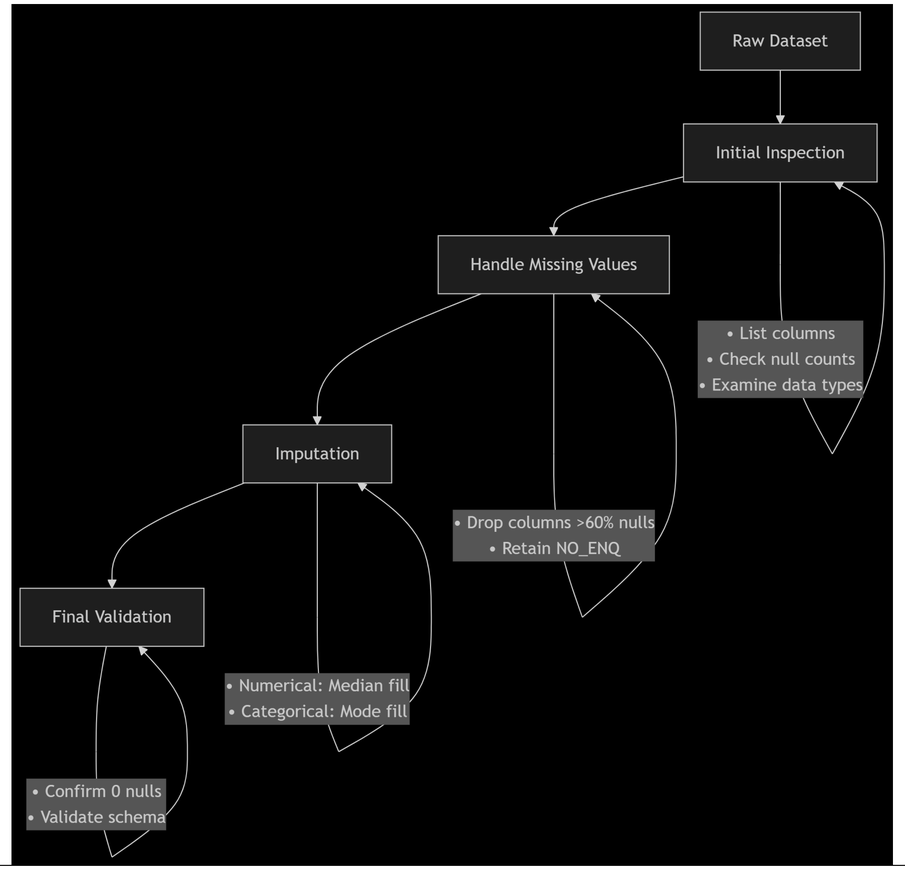

- **Data Source:**
  - Historical retail loan data from PSBs
  - CRIF Bureau scores and IRAC status logs
- **Challenges:**
  - Presence of noise and missing values
  - Highly imbalanced target variable (<10% defaulters)
  - Mixed data types requiring different preprocessing techniques
- **Opportunities:**
  - Rich time-series behavior across 12 months
  - Detailed credit risk indicators and account history
  - Room for innovative feature engineering and domain knowledge application

# Major Feature Domains



## Approach & Methodology

# The process of obtaining a cleaned dataset from the available data can be clearly summarized as shown in the flowchart below:



# Feature Engineering

Feature Engineering can be done for different categories of data some of those examples are provided below:

- **Utilization Ratio:**
  - **Formula:** OUTS / LIMIT
  - **Purpose:** Measures how much of the sanctioned credit limit is being used.
  - Higher values indicate over-leverage, increasing default risk.
- **Installment to Limit Ratio:**
  - **Formula:** INSTALAMT / LIMIT
  - **Purpose**: Represents EMI burden relative to credit limit.
  - Helps assess repayment capacity under credit constraints.
- **Residual Tenure Ratio:**
  - **Formula:** ACCT_RESIDUAL_TENURE / LOAN_TENURE
  - **Purpose:** Indicates how much of the loan tenure remains.
  - Accounts nearing end-of-tenure with high outstanding balance are riskier.

**Model Strategy**

# Baseline Model:

## Logistic Regression

- Quick and interpretable.
- Established initial performance benchmark.
- Limitation: Couldn’t capture complex non-linear feature interactions.

# Final Model:

## XGBoost Classifier

- Handles non-linear relationships effectively.
- Better performance on imbalanced datasets.
- In-built handling of missing values & regularization.

| Criteria | Logistic Reg. | Random Forest | XGBoost | LightGBM |
|---|---|---|---|---|
| Interpret-ability | High | Medium | Medium | Medium |
| Speed | Fast | Moderate | Slow | Fast |
| Performance | Baseline | Good | Very Good | Excellent |
| Imbalance Handling | Manual Weights | Medium | Built-in | Built-in |
| Feature Importance | Coefficients | Gini | Gain/SHAP | Gain/SHAP |

# Model Tuning & Evaluation Metrics

## Model Tuning

**Tuning Approach:**

- RandomizedSearchCV with Stratified K-Fold Cross-Validation.
- Efficiently explored a wide parameter space with fewer iterations.

**Parameters Tuned:**

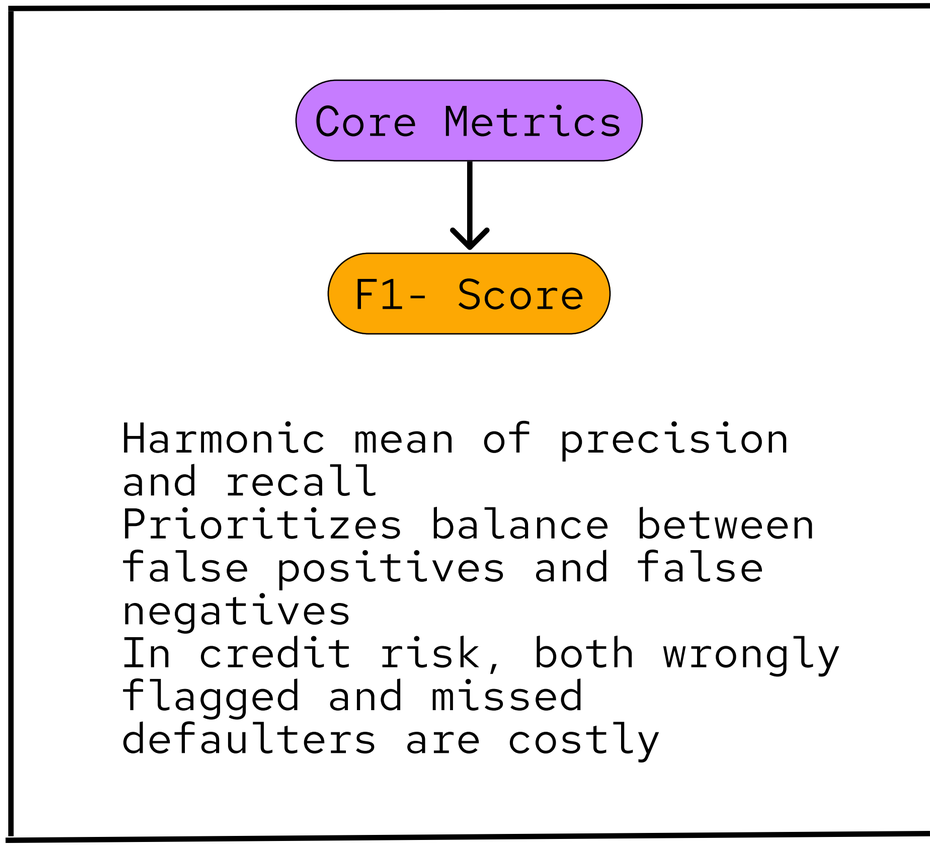
- n_estimators, max_depth, learning_rate, subsample, scale_pos_weight.

**Goal:**

- Improve model generalization.
- Handle class imbalance effectively.

**Threshold Tuning (Decision Boundary Optimization)**

- Default threshold (0.5) resulted in high false positives.
- Evaluated Precision-Recall tradeoff across multiple thresholds.
- Optimal Threshold Chosen: **0.7**
- Significantly reduced false positives.
- Maintained high recall to capture maximum defaulters.
- Aligned with business objective to minimize operational costs of false alarms.



# Model Comparison:

| Criteria | Logistic Reg. | Random Forest | XGBoost | LightGBM |
|---|---|---|---|---|
| Accuracy | .86 | .87 | .85 | .88 |
| F1 Score | .40 | .47 | .55. | .48 |

# Threshold Tuning:

| Threshold | Precision | Recall | F1- Score | Accuracy |
|---|---|---|---|---|
| .5 | .43 | .82 | .57 | 87% |
| .7 | .56 | .68 | .61 | 91% |

# Phase- 2 Objective & Approach

**Objective :**

- Enhance the predictive model by developing a Digital Monitoring Layer to track defaulters' digital activity.
- Support recovery teams with real-time behavioral signals derived from publicly accessible social media data.

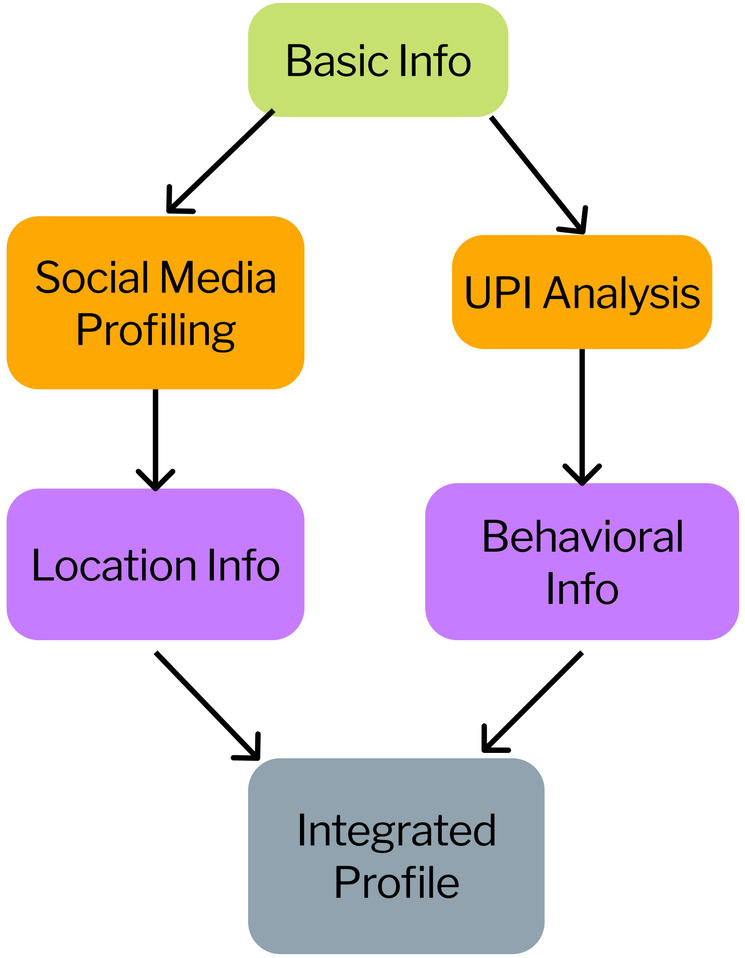
**Tracking Layers Implemented:**

**a. Social Media Scraping (Instagram & Facebook):**

- Search & extract publicly visible data using:
  - Name, Phone Numbers, Email IDs (bank has this data).
- Target Insights:
  - Account Activity Status (active/inactive).
  - Bio/Location Tags.
  - Public Posts indicating lifestyle or financial changes.
- Challenge: Private accounts limit deep data extraction.

**b. UPI Transaction Monitoring:**

- Utilize UPI transaction summaries (shared with banks via NPCI) for real-time financial activity insights.
- Classify transactions into Merchant vs Individual categories.



# UPI Transaction Analysis & Location Approximation

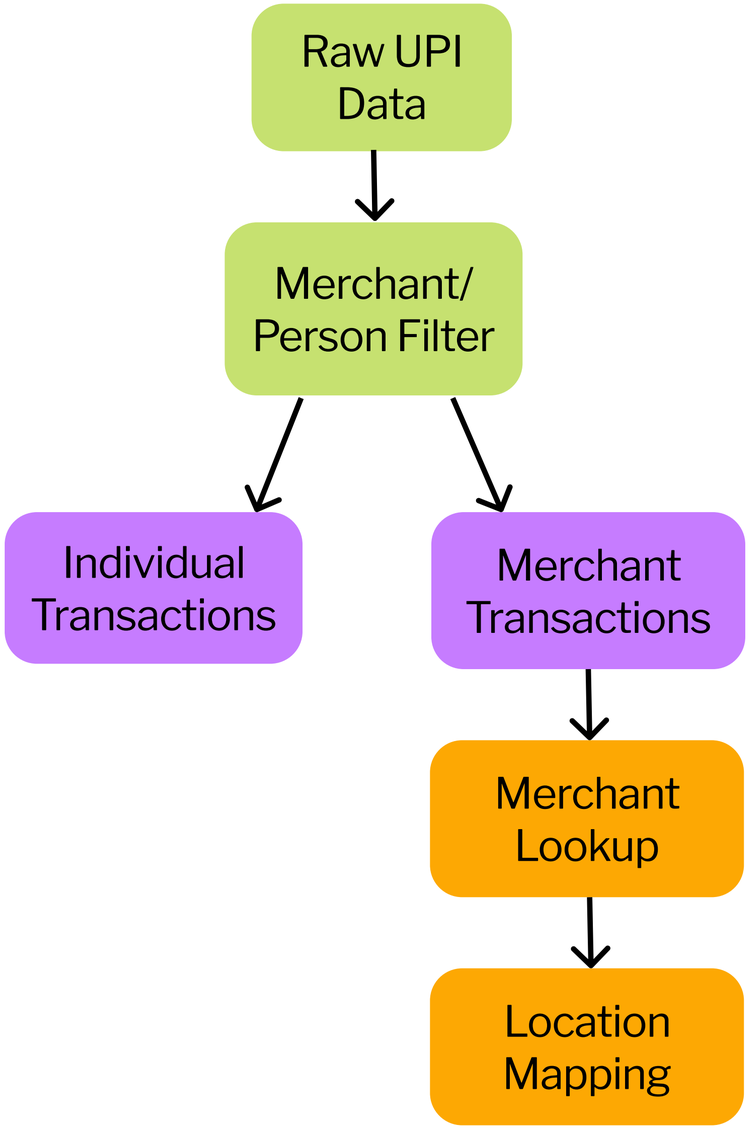
1. **UPI Transaction Pattern Analysis:**

   - **Detect financial stress signals:**
     - Drop in transaction frequency.
     - Irregular payment patterns.
   - **Merchant vs Individual:**
     - Classify UPI receivers (using handle patterns & keywords).

2. **Geo-Mapping of Merchants:**

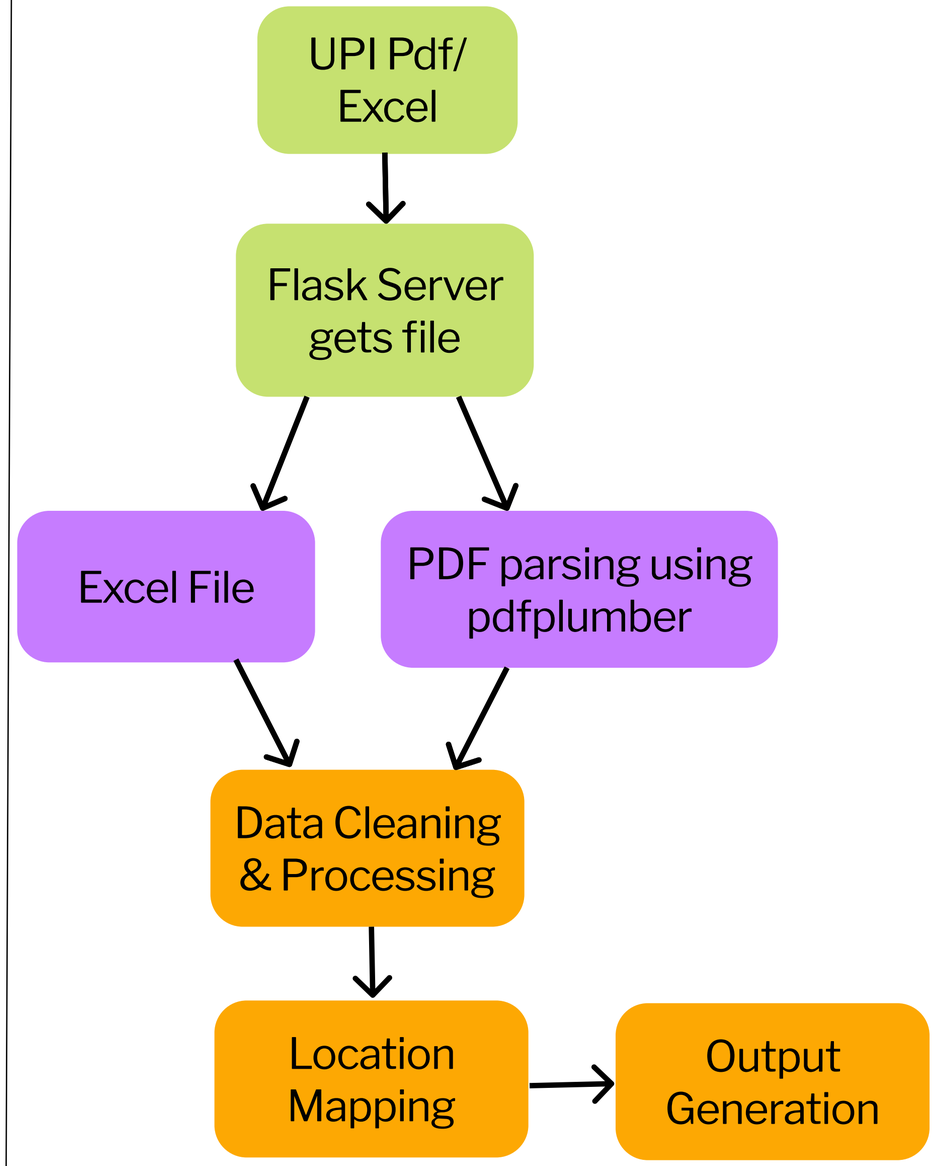
   - **For Merchant Transactions:**
     - Fetch merchant details from business directories.
     - Use Geolocation APIs to map merchant addresses.

   - **Approximate borrower's current locality based on their recent payments.**



# Prototype Working

- **Simple Web Interface** — Upload UPI statements (PDF) and get instant analysis, no technical skills needed.
- **Python + Flask Backend** — Lightweight app using pdfplumber for parsing and pandas for data processing.
- **Recipient Classification** — Categorizes top recipients as people or locations via googlesearch.
- **Outputs Generated** — Cleaned CSV, detailed PDF summary, and URLs for location-based recipients.
- **Easy Deployment** — Runs locally with minimal setup; users just upload and get results automatically.



# Compliance & Ethical Data Usage

## Data Privacy Measures:

- No APIs used for Social Media Scraping — Only publicly visible data is manually fetched.
- No unauthorized data collection or web crawling.
- UPI transaction summaries are obtained through authorized bank-NPCI channels.
- Geolocation APIs are used solely to fetch merchant location details (no individual location tracking).
- All data usage is aligned with RBI privacy norms.
- Entire system designed as an internal risk monitoring tool, not for external exposure.

| Aspect | Compliance Measure |
|---|---|
| Social Media Data | Publicly available information only, no scraping APIs used |
| UPI Transaction Data | Bank-authorized via NPCI channels |
| Location Tracking | Merchant location mapping only, no direct customer GPS |
| Data Security | All processing within bank's secure infrastructure |

# Summary

- **Phase-1: NPA Prediction Model**
  - Built an XGBoost model to predict SMA-2 stage NPAs.
  - Feature Engineering: Utilization Ratio, Installment-to-Limit Ratio, Residual Tenure Ratio.
  - Hyperparameter & Threshold Tuning (Optimal Threshold: 0.7).
  - Achieved 61% F1-Score with improved precision-recall balance.
- **Phase-2: Digital Monitoring Layer**
  - Scraped public Instagram & Facebook data for activity tracking.
  - Analyzed UPI transaction patterns (via NPCI data) to detect financial stress.
  - Used Merchant classification & Geolocation APIs to approximate borrower locations.
  - Developed a simple web tool for UPI data processing & reporting.
- **Compliance & Impact**
  - Public data only, no unauthorized API usage.
  - Aligned with RBI guidelines.
  - Enables proactive defaulter tracking and smarter recovery actions.

# Thank You

Tailwinders