



UnionVision.AI

Top 3 Machine Learning Use Cases for Prototyping for UA Locals

1. Outlier Detection of Remittance Transactions

Why?

- Requires minimal labeled data.
- Can be implemented using unsupervised learning techniques.
- Provides immediate business value by detecting anomalies in financial transactions.

Approach:

- Data Required: Historical remittance transactions (amounts, timestamps, contractor IDs).
- ML Techniques: Isolation Forest, Autoencoders, One-Class SVM.
- Implementation Steps:
 1. Collect past remittance transaction data.
 2. Normalize and preprocess transaction amounts.
 3. Train an anomaly detection model to identify transactions deviating from normal patterns.
 4. Visualize flagged transactions for manual review.

Expected Outcome:

- A model that flags unusual transactions in real-time, reducing fraud and errors.
-

2. Member Churn/Attrition Risk Prediction

Why?

- Well-defined problem with historical churn patterns.
- Can be implemented using basic classification models.
- Helps in proactive member engagement.

Approach:

- Data Required: Member work history, benefits usage, previous churn cases.
- ML Techniques: Logistic Regression, Decision Trees, Random Forest.
- Implementation Steps:
 1. Gather historical data of active and churned members.
 2. Engineer relevant features (e.g., tenure, benefit usage frequency).
 3. Train a classification model to predict the likelihood of churn.
 4. Deploy model to generate risk scores for active members.

Expected Outcome:

- Early warning system to prevent member drop-offs.



UnionVision.AI

3. Contractor Performance Analysis

Why?

- Uses structured, available data (payment consistency, disputes, safety records).
- Can be implemented as a scoring system using regression models.
- Provides actionable insights to improve contractor reliability.

Approach:

- Data Required: Contractor payment history, dispute rates, on-time completion records.

- ML Techniques: Linear Regression, Gradient Boosting, Ranking Models (LambdaMART).

- Implementation Steps:

1. Collect contractor performance indicators.
2. Define weighted scoring metrics (e.g., payment timeliness, number of disputes).
3. Train a ranking model to generate performance scores.
4. Display scores in a dashboard for management insights.

Expected Outcome:

- A contractor rating system that aids in decision-making and partnership management.

Next Steps

1. Data Preparation: Identify and preprocess required data for each use case.
2. Prototype Development: Build simple ML models and evaluate performance.
3. Validation & Testing: Ensure accuracy using real-world data.
4. Deployment Strategy: Implement dashboards or API endpoints for integration.

These three prototypes offer high business impact with relatively low implementation complexity.