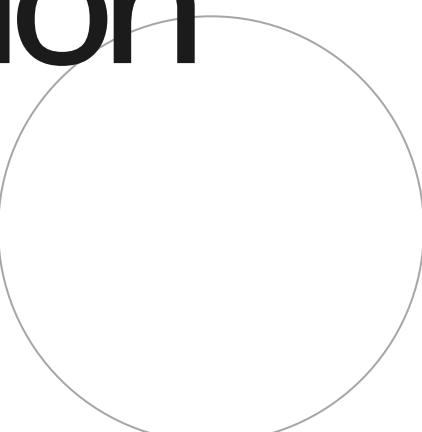
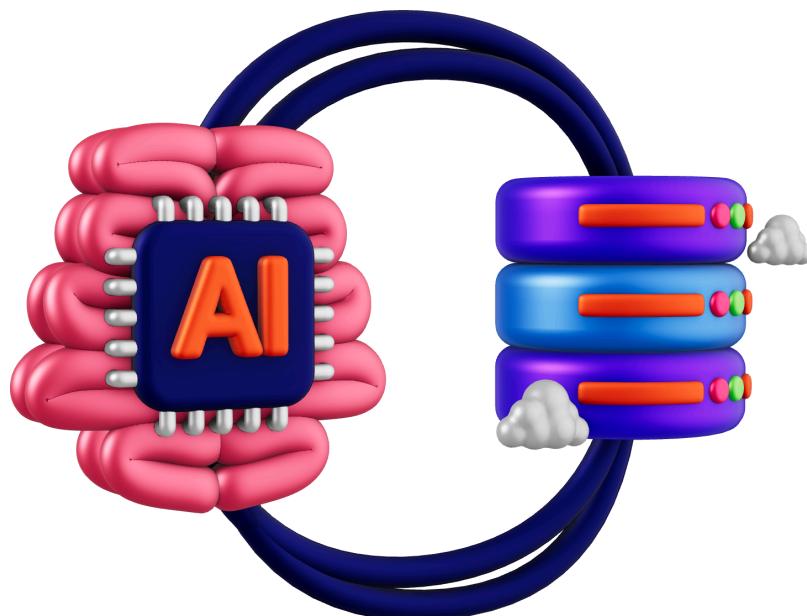
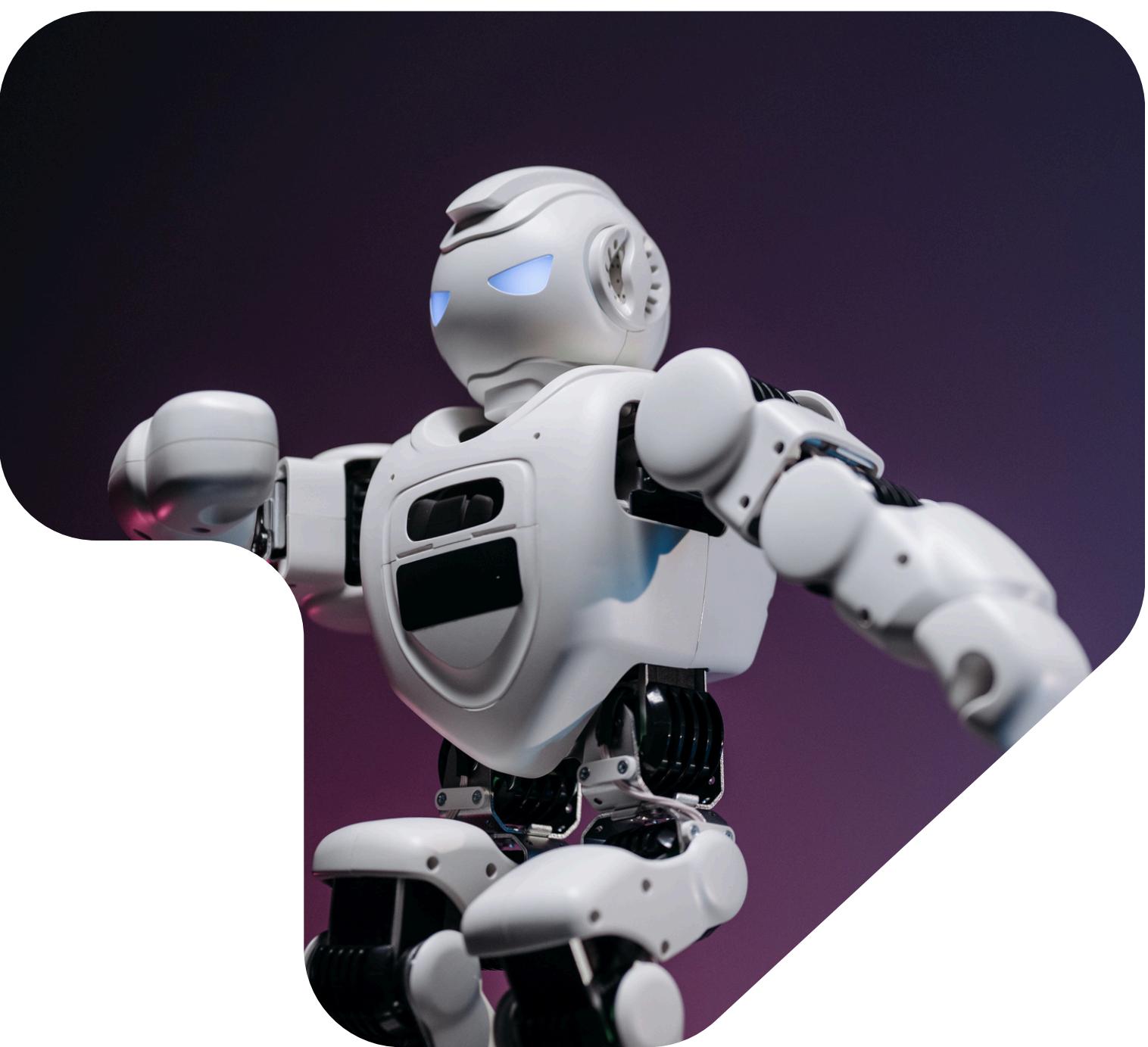


Automated AI-Based Financial Transaction Categorisation

Shaping the Future



|



Problem Statement

From Concept to Global Force



Financial applications must convert raw transaction texts—such as “Starbucks,” “HP Fuel,” or “Amazon Pay”—into clear categories like Dining, Fuel, or Shopping. Current systems often rely on expensive third-party APIs, which introduce privacy risks, higher latency, and limited flexibility in customizing categories.

To overcome these issues, there is a need for an independent, in-house AI-based categorisation system that provides accurate classification, allows easy updates to category taxonomy, explains its predictions, and continuously improves through user feedback.

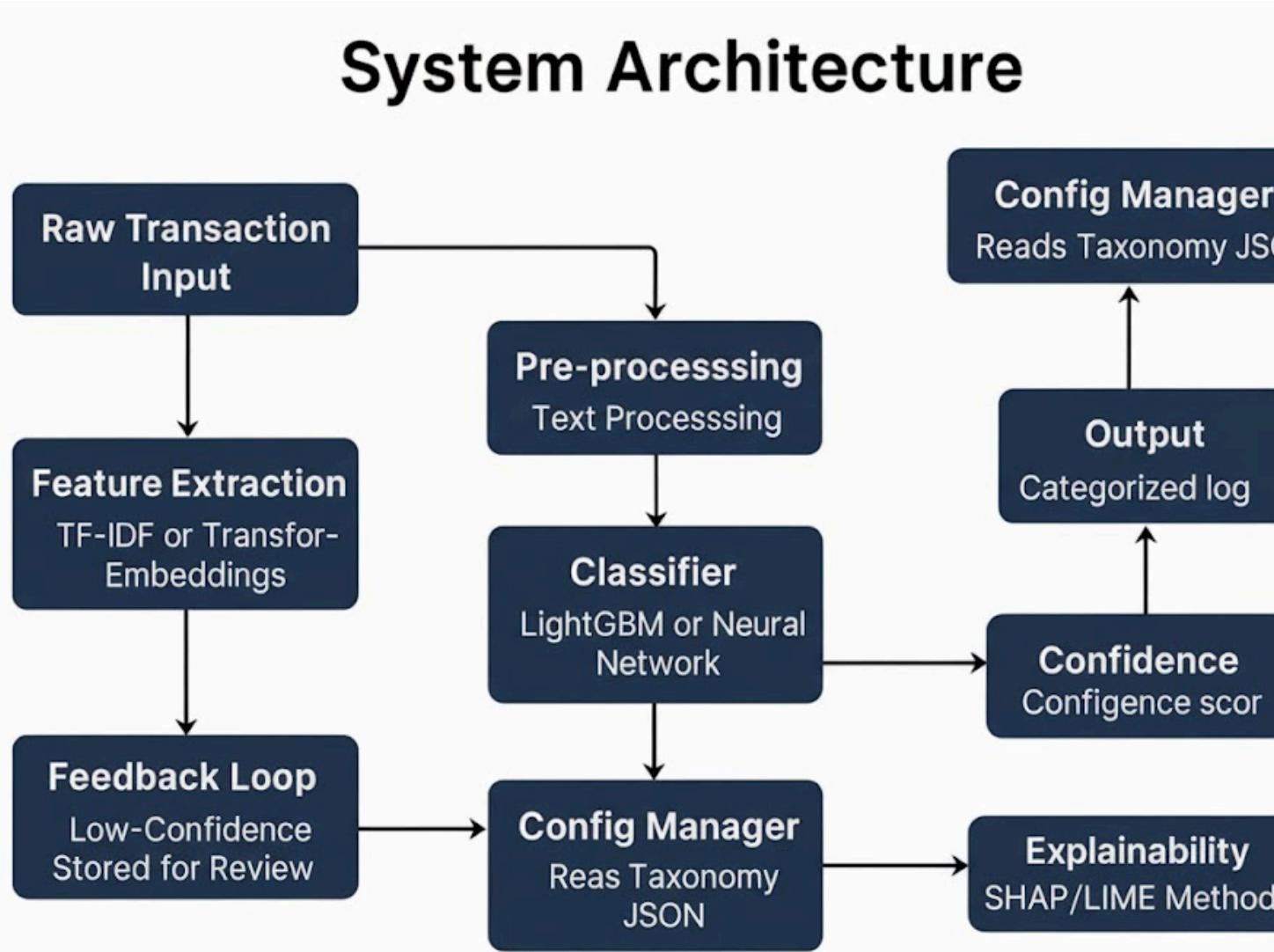
Technology Stack

Seamlessly Integrated
Into Our World



- **Programming Language:** Python
- **ML Frameworks:** Scikit-learn, LightGBM, TensorFlow (optional)
- **API Layer:** FastAPI / Flask
- **Storage:** PostgreSQL / SQLite
- **Configuration:** JSON / YAML (for dynamic taxonomy updates)
- **Explainability Tools:** SHAP / LIME
- **Deployment:** Docker, Kubernetes
- **Version Control:** GitHub / Google Drive Repository

System Architecture



(a) Input Preprocessing Layer

- Cleans raw transaction text
- Normalizes casing, removes special characters
- Expands abbreviations (e.g., “MC DLD” → “McDonald's”)
- Handles spacing, numeric noise, and merchant variations

(b) Feature Encoding Layer

- Converts cleaned text to numerical vectors
- Options:
 - TF-IDF (lightweight, fast)
 - Transformer embeddings (semantic understanding)

(c) Classification Layer

- Supervised ML model
- LightGBM / Logistic Regression / Shallow Neural Network
- Outputs:
 - Predicted category
 - Confidence score

(d) Config Manager

- Loads category taxonomy from JSON/YAML
- Allows editing categories without altering code

(e) Inference Pipeline

- Generates prediction + confidence
- Logs all input/output
- Sends low-confidence predictions for human review

(f) Explainability Module

- Uses SHAP/LIME
- Shows key words influencing the prediction

(g) Feedback Loop

- Users can correct wrong predictions
- These corrections are stored
- Model retrains periodically for better accuracy

Data Model & Storage

Transaction Table

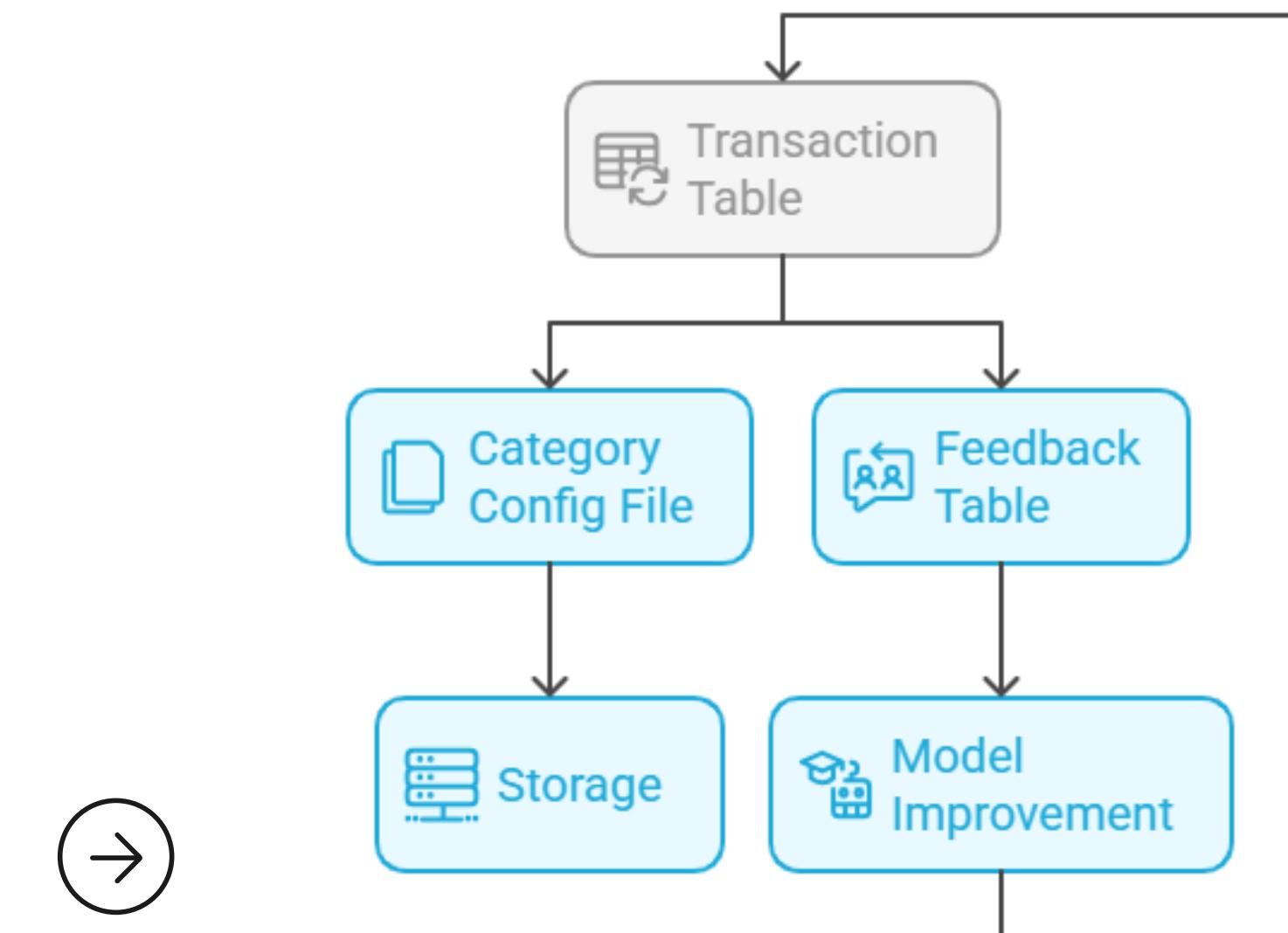
- ID
- Raw text
- Cleaned text
- Predicted category
- Confidence score
- Timestamp

Feedback Table

- Transaction ID
- Model prediction
- User-corrected category
- Review status
- Timestamp

Config File (JSON / YAML)

- Category list
- Subcategories (optional)
- Keywords or rules (optional)



Storage Layer

- SQLite for local/academic usage
- PostgreSQL for production deployment

Preprocessing

- Lowercasing
- Stopword removal
- Regular expression cleaning
- Abbreviation expansion
- Merchant text normalization

Feature Engineering

- TF-IDF vectors OR
- Sentence/embedding vectors

Model

- Supervised learning classifier
- Trained on mixed dataset
- Target metric: ≥ 0.90 Macro F1-score

Automation

- Automatic retraining pipeline
- Confidence-based routing
- Logging & monitoring
- Explainability generation for each inference



AI / ML / Automation Components

Security & Compliance

Local inference → No third-party API dependency

Encryption:

- At rest (database encryption)
- In transit (SSL/HTTPS)

Access Control:

- Role-based permissions for reviewers/admins

GDPR-Friendly Design:

- User data deletion supported
- Transparent & auditable logs

No external data leakage



Security and Compliance Measures

Local Inference Pipeline

The inference pipeline operates locally without external API calls.

Encryption in Transit

Data is encrypted during transmission using HTTPS/SSL.

GDPR Compliance

The system is designed to comply with GDPR regulations.

No Third-Party Data Leakage

Data is protected from leakage to third parties.



Encryption at Rest

Data is encrypted when stored in the database.

Role-Based Access Control

Access is controlled based on user roles.

Explainable AI

AI processes are transparent and understandable.

Scalability & Performance

Supports:

- Real-time classification
- High-volume batch processing

Optimized for CPU-only inference

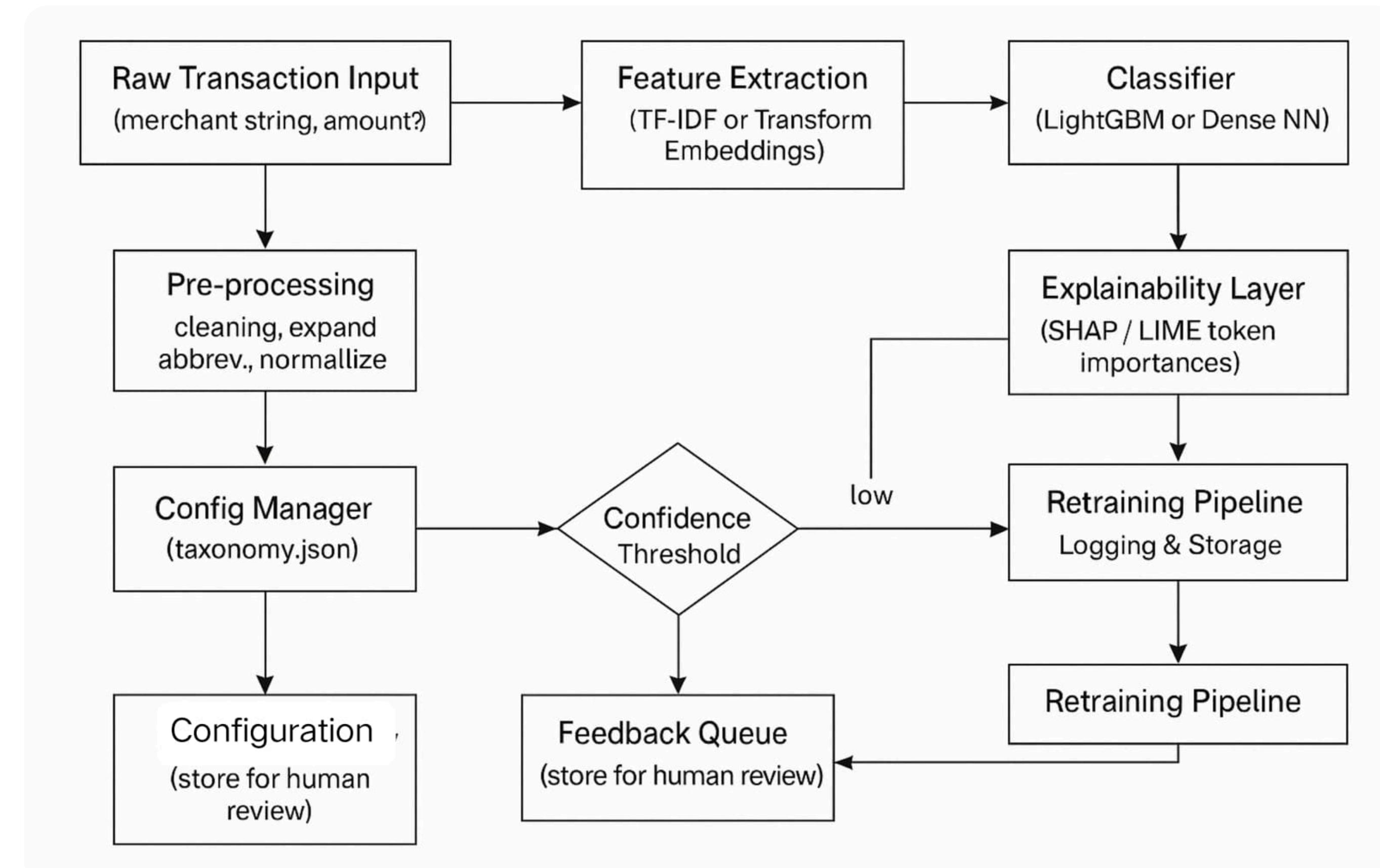
Can scale horizontally using:

- Docker containers
- Kubernetes

Supports model versioning and rolling updates

Low latency prediction pipeline

Pipeline diagram



Conclusion

The proposed InHouseTX – Automated AI-Based Transaction Categorisation System delivers a scalable, transparent, and cost-effective solution for classifying financial transactions using machine learning. By performing all inference locally, the system removes reliance on third-party APIs, ensuring higher privacy, lower latency, and zero recurring costs.

With a modular architecture, explainability features, dynamic taxonomy configuration, and a built-in feedback loop, the model can adapt to new transaction patterns and continuously improve over time. The design supports both real-time and batch processing, making it suitable for consumer banking apps, fintech platforms, and enterprise financial tools.

Overall, the system demonstrates a responsible and efficient approach to AI-driven financial categorisation, offering high accuracy, operational flexibility, and strong data security, making it a reliable solution for modern financial applications.





Thank You

Links

GitHub Repository:
[YOUR_REPO_LINK]

Demo Video (YouTube / Drive):
[YOUR_VIDEO_LINK]
xt