**Fraud Detection using Autoencoder model**

**Project Objective**

The primary goal of this project is to **detect fraudulent transactions** from large financial datasets using an **unsupervised deep learning approach**—specifically, an **autoencoder** neural network. Traditional machine learning models often struggle with highly imbalanced data like fraud detection. Autoencoders can learn to reconstruct normal behavior, and any large reconstruction error is flagged as potential fraud.

**Why Autoencoder?  
Traditional supervised models struggle when frauds make up less than 1% of the data. That’s why autoencoders are effective—they don't need labeled fraud data to learn. Instead, they try to recreate normal patterns. If a transaction cannot be reconstructed well, it’s considered suspicious.”**

* **Imbalanced Dataset:** Fraudulent transactions are typically rare (~1% or less), making it difficult for supervised models to generalize.
* **Unsupervised Learning Power:** Autoencoders learn from **only the normal class**, making them ideal for anomaly detection.
* **Reconstruction Error-Based Detection:** The model learns to replicate normal transaction patterns. If a transaction deviates significantly, it is flagged as fraudulent.

**Dataset Overview  
We cleaned and scaled the data using StandardScaler. Categorical columns were label-encoded. The label column was kept aside and not used during training. This is key because the autoencoder is trained only on normal transactions.**

* **Size:** 535,178 transactions
* **Features:** 9 (after feature engineering)
* **Imbalance:**
  + Normal: 528,686 (98.8%)
  + Fraudulent: 6,492 (1.2%)

**Project Structure**

**1. Data Preprocessing**

* Loaded train.csv dataset.
* Removed irrelevant or redundant columns.
* Scaled all features using StandardScaler.
* Encoded categorical features using LabelEncoder (if needed).
* Separated out the label column for evaluation later.
* Removed labels before training the autoencoder.

**2. Autoencoder Architecture**

* **Input Layer:** Matches the number of features (9)
* **Encoding Dimension:** 14 (compressed representation)
* **Loss Function:** Mean Squared Error (MSE)
* **Optimizer:** Adam
* **Training Epochs:** 7
* **EarlyStopping:** Applied to avoid overfitting

**3. Training Summary**

Autoencoder Architecture:

- Input dimension: 9

- Encoding dimension: 14

- Total parameters: 275

Epoch-wise Loss:

- Epoch 1: 0.5398

- Epoch 2: 0.4907

- Epoch 3: 0.4883

- Epoch 4: 0.4824

- Epoch 5: 0.4790

- Epoch 6: 0.4886

- Epoch 7: 0.4866

**Thresholding and Evaluation**

* Calculated **reconstruction error** on training data.
* Threshold determined using:  
  threshold = mean + 3 × std deviation of reconstruction errors.

**Threshold Set:** 1.379744

**Model Evaluation on Training Data**

Confusion Matrix:

[[502251 26435]

[ 3915 2577]]

Fraud Detection Summary:

- True Negatives: 502251

- False Positives: 26435

- False Negatives: 3915

- True Positives: 2577

Model Accuracy: 94.33%

**Technologies Used**

* **Python**
* **NumPy**, **Pandas** – data handling
* **scikit-learn** – preprocessing, evaluation
* **TensorFlow / Keras** – deep learning model
* **Matplotlib** – plotting and visual analysis

Training Autoencoder on entire dataset...

🔹 Autoencoder Architecture:

Input dimension: 9

Encoding dimension: 14

Total parameters: 275

Epoch 1/7

**16725/16725** ━━━━━━━━━━━━━━━━━━━━ **30s** 2ms/step - loss: 0.5398

Epoch 2/7

**16725/16725** ━━━━━━━━━━━━━━━━━━━━ **41s** 2ms/step - loss: 0.4907

Epoch 3/7

**16725/16725** ━━━━━━━━━━━━━━━━━━━━ **30s** 2ms/step - loss: 0.4883

Epoch 4/7

**16725/16725** ━━━━━━━━━━━━━━━━━━━━ **29s** 2ms/step - loss: 0.4824

Epoch 5/7

**16725/16725** ━━━━━━━━━━━━━━━━━━━━ **41s** 2ms/step - loss: 0.4790

Epoch 6/7

**16725/16725** ━━━━━━━━━━━━━━━━━━━━ **30s** 2ms/step - loss: 0.4886

Epoch 7/7

**16725/16725** ━━━━━━━━━━━━━━━━━━━━ **29s** 2ms/step - loss: 0.4866

🔹 Training completed in 7 epochs

🔹 Calculating reconstruction errors...

**16725/16725** ━━━━━━━━━━━━━━━━━━━━ **18s** 1ms/step

🔹 Threshold set at: 1.379744

🔹 Normal data reconstruction error - Mean: 0.390362, Std: 0.432947

==================================================

🔹 MODEL EVALUATION RESULTS

==================================================

🔹 Confusion Matrix:

[[502251 26435]

[ 3915 2577]]

🔹 Fraud Detection Summary:

True Negatives: 502251

False Positives: 26435

False Negatives: 3915

True Positives: 2577

🔹 MODEL ACCURACY: 0.9433 (94.33%)

🔹 Detailed Performance Metrics:

Overall Accuracy: 0.9433 (94.33%)

Overall Precision: 0.9813

Overall Recall: 0.9433

Overall F1-score: 0.9607

🔹 Class-wise Performance:

Normal Class (0):

Precision: 0.9923

Recall: 0.9500

F1-score: 0.9707

Fraud Class (1):

Precision: 0.0888

Recall: 0.3970

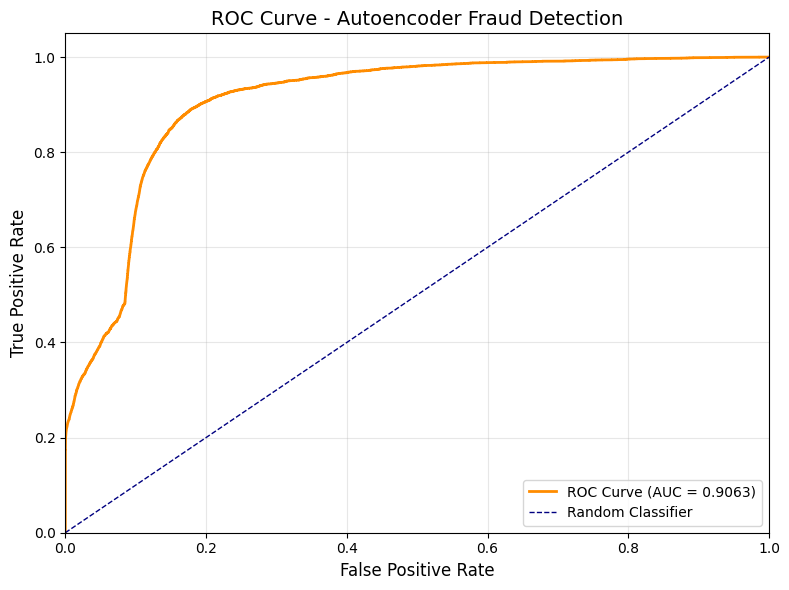
F1-score: 0.1452

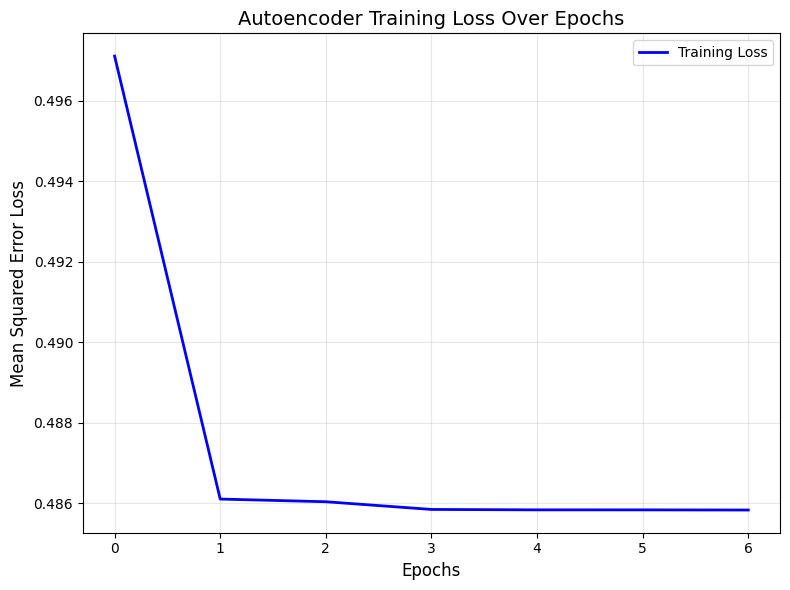
🔹 ROC-AUC Score: 0.9063

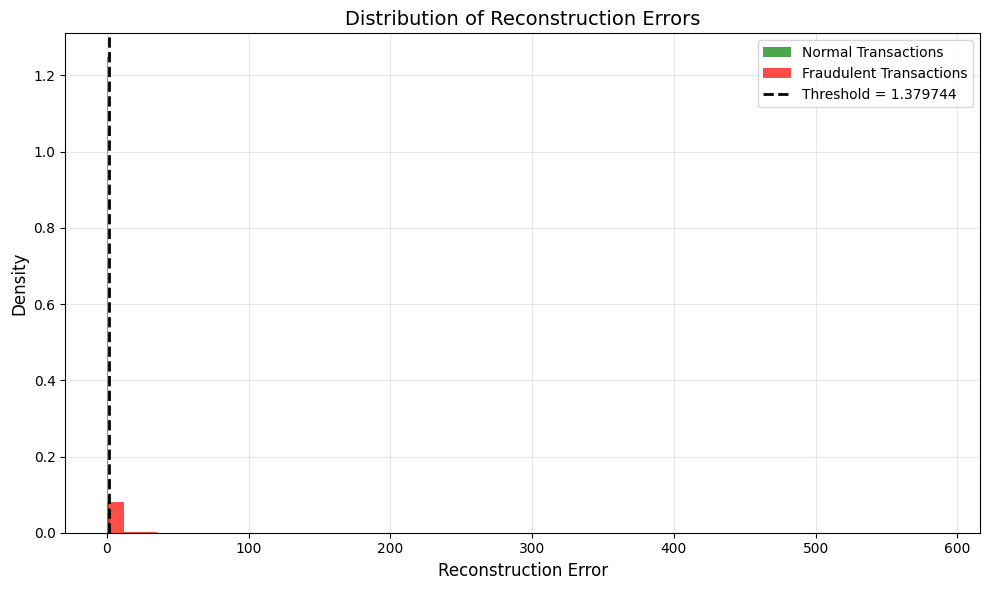
🔹 Additional Accuracy Metrics:

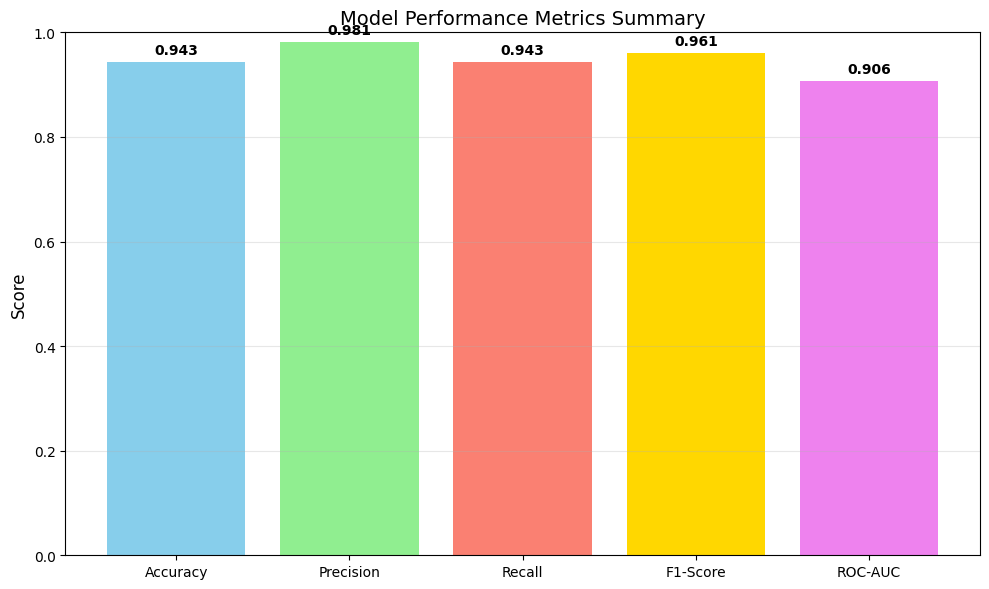
Sensitivity (Recall): 0.3970 (39.70%)

Specificity: 0.9500 (95.00%)









🔹 Final Model Summary:

==================================================

🎯 MODEL ACCURACY: 0.9433 (94.33%)

🔹 Model trained on 535178 total samples

🔹 Normal transactions: 528686 (98.8%)

🔹 Fraudulent transactions: 6492 (1.2%)

🔹 Correctly classified: 504828.0 out of 535178 samples