

Interim 2 submission Report: Model Building and Training

Overview

This report outlines the process of building and training machine learning models to enhance fraud detection in e-commerce and banking transactions. As a data scientist at Adey Innovations Inc., the goal is to develop accurate fraud detection models by leveraging machine learning techniques and performing comprehensive data analysis.

Business concept

Adey Innovations Inc. aims to improve transaction security by developing advanced fraud detection systems to

- Reduce financial losses due to fraudulent transactions.
- Strengthen trust with customers and financial institutions.
- Enable real-time fraud detection and rapid response.

The project involves multiple steps, including data analysis, feature engineering, model training, and deployment to ensure continuous improvements.

Data Preparation

Feature and Target Separation

For both datasets, we separated the features and target variables:

Fraud_Data.csv

Features: user_id, signup_time, purchase_time, purchase_value, device_id, source, browser, sex, age, ip_address

Target Variable: class

Creditcard.csv

Features: Time, V1 to V28, Amount

Target Variable: Class

Train-Test Split

We used the `train_test_split` function from `sklearn.model_selection` to create training and testing datasets for both data sources. This ensures the model is validated on unseen data.

python

Copy code

```
# Train-test split
```

```
X_fraud, y_fraud = fraud_data.drop('class', axis=1), fraud_data['class']
```

```
X_train_fraud, X_test_fraud, y_train_fraud, y_test_fraud = train_test_split(X_fraud, y_fraud,
test_size=0.2, random_state=42)
```

```
X_creditcard, y_creditcard = creditcard_data.drop('Class', axis=1), creditcard_data['Class']
```

```
X_train_creditcard, X_test_creditcard, y_train_creditcard, y_test_creditcard =
train_test_split(X_creditcard, y_creditcard, test_size=0.2, random_state=42)
```

Model Selection

To explore model performance, the following algorithms were selected for comparison

- Logistic Regression
 - Decision Tree
 - Random Forest
 - Gradient Boosting
 - Multi-Layer Perceptron (MLP)
 - Recurrent Neural Network (RNN)
 - Long Short-Term Memory (LSTM)
-

Model Training and Evaluation

Training Models

Each model was trained using both datasets. Here is a summary of the process:

python

Copy code

```
# Model training and evaluation

models = {

    'Logistic Regression': LogisticRegression(),

    'Decision Tree': DecisionTreeClassifier(),

    'Random Forest': RandomForestClassifier(),

    'Gradient Boosting': GradientBoostingClassifier(),

    'MLP': MLPClassifier(),

}

results_fraud = {}

results_creditcard = {}

for name, model in models.items():

    model.fit(X_train_fraud, y_train_fraud)

    y_pred_fraud = model.predict(X_test_fraud)

    results_fraud[name] = {

        'Accuracy': accuracy_score(y_test_fraud, y_pred_fraud),

        'Report': classification_report(y_test_fraud, y_pred_fraud)

    }
```

```
model.fit(X_train_creditcard, y_train_creditcard)

y_pred_creditcard = model.predict(X_test_creditcard)

results_creditcard[name] = {

    'Accuracy': accuracy_score(y_test_creditcard, y_pred_creditcard),

    'Report': classification_report(y_test_creditcard, y_pred_creditcard)

}
```

Evaluation Metrics

Accuracy scores and classification reports were generated for each model to facilitate performance comparisons.

MLOps Steps

Versioning and Experiment Tracking

To track model performance, parameters, and metrics, we utilized MLflow, which enabled seamless versioning and experiment management.

python

Copy code

```
# MLflow tracking setup
```

```
mlflow.start_run()
```

```
for name, model in models.items():
```

```
    mlflow.log_param("model_name", name)
```

```
    mlflow.log_metric("accuracy", results_fraud[name]['Accuracy'])
```

```
    mlflow.sklearn.log_model(model, name)
```

```
mlflow.end_run()
```

The screenshot shows the MLflow Experiments page for the 'fraud_detection_experiment'. The interface includes a search bar, a list of experiments, and a table of runs. The runs table has columns for Run Name, Created, Dataset, Duration, User, Source, Version, Models, and Description. The runs are sorted by 'Created' time, showing a list of runs with their respective names, creation times, and durations.

Run Name	Created	Dataset	Duration	User	Source	Version	Models	Description
delicate-squire-13	12 minutes ago	-	38.1s	hayyuragea	CUUsers...	bc950b	tensorflow	-
handsome-squire-775	13 minutes ago	-	44.0s	hayyuragea	CUUsers...	bc950b	tensorflow	-
respectful-hawk-477	13 minutes ago	-	16.3s	hayyuragea	CUUsers...	bc950b	sklearn	-
humble-brout-93	14 minutes ago	-	42.2s	hayyuragea	CUUsers...	bc950b	sklearn	-
adorable-cow-200	15 minutes ago	-	1.3min	hayyuragea	CUUsers...	bc950b	sklearn	-
quirky-app-316	15 minutes ago	-	7.3s	hayyuragea	CUUsers...	bc950b	sklearn	-
flawless-art-574	16 minutes ago	-	5.8s	hayyuragea	CUUsers...	bc950b	sklearn	-
secretive-grub-373	18 minutes ago	-	2.8min	hayyuragea	CUUsers...	bc950b	tensorflow	-
boasting-carf-447	20 minutes ago	-	1.4min	hayyuragea	CUUsers...	bc950b	tensorflow	-
delicate-horse-565	21 minutes ago	-	45.2s	hayyuragea	CUUsers...	bc950b	sklearn	-
staunch-vet-54	26 minutes ago	-	5.5min	hayyuragea	CUUsers...	bc950b	sklearn	-
adorable-cow-796	30 minutes ago	-	3.5min	hayyuragea	CUUsers...	bc950b	sklearn	-
flawless-cow-64	30 minutes ago	-	29.7s	hayyuragea	CUUsers...	bc950b	sklearn	-
fun-shed-903	31 minutes ago	-	45.9s	hayyuragea	CUUsers...	bc950b	sklearn	-
casual-cow-46	38 minutes ago	-	-	hayyuragea	CUUsers...	bc950b	-	-
travelling-snipe-559	43 minutes ago	-	5.6min	hayyuragea	CUUsers...	bc950b	sklearn	-
warmed-sheep-770	44 minutes ago	-	46.8s	hayyuragea	CUUsers...	bc950b	sklearn	-
intelligent-crab-84	45 minutes ago	-	1.1min	hayyuragea	CUUsers...	bc950b	sklearn	-

Evaluating Models for Credit Card Data MLFLOW

- Logistic regression

The screenshot shows the MLflow Experiments page for the 'traveling-snipe-559' run. The interface includes a search bar, a list of experiments, and a table of runs. The runs table has columns for Run Name, Created, Dataset, Duration, User, Source, Version, Models, and Description. The runs are sorted by 'Created' time, showing a list of runs with their respective names, creation times, and durations.

Run Name	Created	Dataset	Duration	User	Source	Version	Models	Description
traveling-snipe-559	43 minutes ago	-	5.6min	hayyuragea	CUUsers...	bc950b	sklearn	-

The 'traveling-snipe-559' run details are as follows:

- Created by: hayyuragea
- Experiment ID: 665122849546764073
- Status: Finished
- Run ID: ba9d279be6c45659bde1a69a8790086
- Duration: 5.6min
- Datasets used: -
- Tags: Add
- Source: C:\Users\hayyuragea\AppData\Local\Programs\Python\Python312\commerce_fraud_detection_system\src\model_builder\bc950b2
- Logged models: sklearn
- Registered models: -

The Parameters (2) section shows:

Parameter	Value
model_name	Random Forest
num_features	30

The Metrics (1) section shows:

Metric	Value
accuracy	0.999085731505305

- Decision Tree

The screenshot shows the mlflow 2.17.0 interface for an experiment named 'flawless-swan-64'. The 'Overview' tab is active, displaying a table of experiment details. The status is 'Finished'. Below the details, there are sections for 'Parameters (2)' and 'Metrics (1)'. The parameters table shows 'model_name' as 'Decision Tree' and 'num_features' as '30'. The metrics table shows 'accuracy' as '0.9990131463010609'.

Parameter	Value
model_name	Decision Tree
num_features	30

Metric	Value
accuracy	0.9990131463010609

- Random Forest

The screenshot shows the mlflow 2.17.0 interface for an experiment named 'adorable-owl-786'. The 'Overview' tab is active, displaying a table of experiment details. The status is 'Finished'. Below the details, there are sections for 'Parameters (2)' and 'Metrics (1)'. The parameters table shows 'model_name' as 'Random Forest'. The metrics table shows 'accuracy' as '0.9995241953380115'.

Parameter	Value
model_name	Random Forest

Metric	Value
accuracy	0.9995241953380115

- Gradient Boosting

The screenshot shows the mlflow web interface for an experiment named 'zealous-rat-54'. The interface is divided into several sections:

- Details:** A table showing experiment metadata.

Field	Value
Created at	2024-10-24 06:33:09
Created by	hayyu.raga
Experiment ID	665122849546764073
Status	Finished
Run ID	e4af78b4214645c189w9326f56b-ddaf
Duration	5.5min
Datasets used	—
Tags	Add
Source	C:\Users\hayyu.raga\AppData\Local\Programs\Python\Python312\ecommerce_fraud_detection_system\src\model_bui...
Logged models	sklearn
Registered models	—
- Parameters (2):** A table showing parameters for the run.

Parameter	Value
model_name	Gradient Boosting
num_features	30
- Metrics (1):** A table showing metrics for the run.

Metric	Value
accuracy	0.9992951045007578

- Multi-Layer Perceptron (MLP)

The screenshot shows the mlflow web interface for an experiment named 'delicate-horse-568'. The interface is divided into several sections:

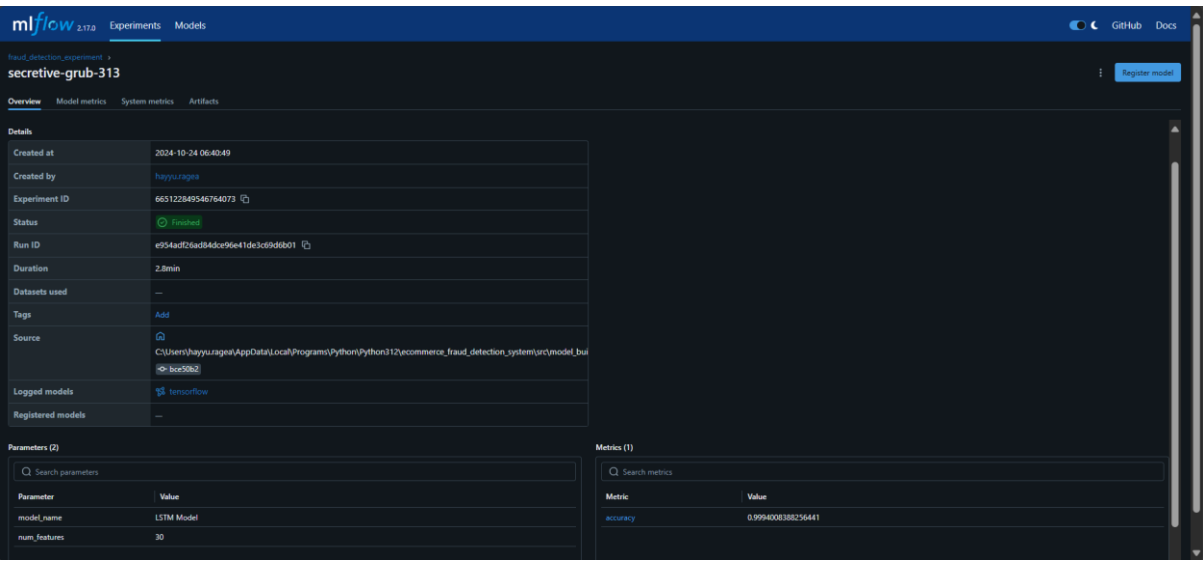
- Details:** A table showing experiment metadata.

Field	Value
Created at	2024-10-24 06:38:39
Created by	hayyu.raga
Experiment ID	665122849546764073
Status	Finished
Run ID	a763067b62224567af7bd386a2f54e5f
Duration	45.2s
Datasets used	—
Tags	Add
Source	C:\Users\hayyu.raga\AppData\Local\Programs\Python\Python312\ecommerce_fraud_detection_system\src\model_bui...
Logged models	sklearn
Registered models	—
- Parameters (2):** A table showing parameters for the run.

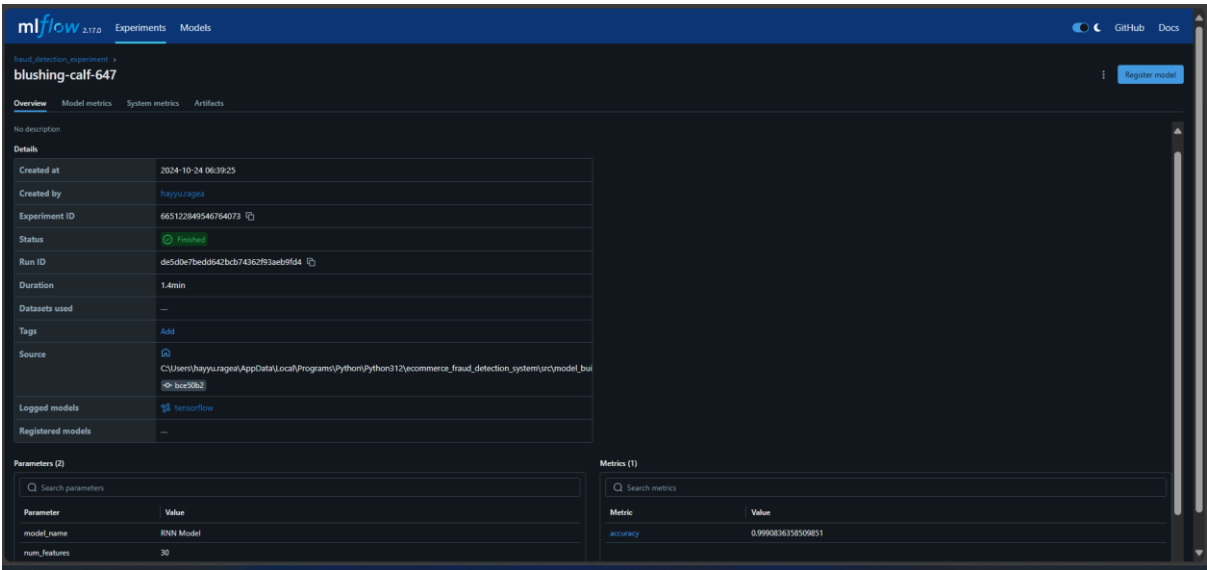
Parameter	Value
model_name	MLP Classifier
num_features	30
- Metrics (1):** A table showing metrics for the run.

Metric	Value
accuracy	0.9985020970641102

- Long Short-Term Memory (LSTM)



- Recurrent Neural Network (RNN) screenshot



Model Evaluation Results

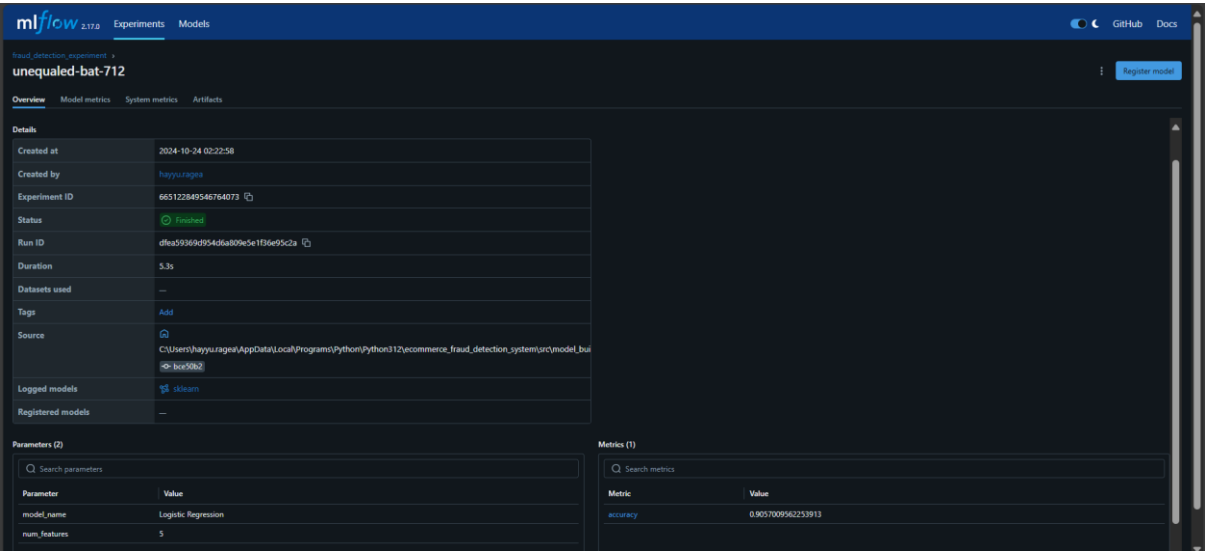
Evaluating Models for Credit Card Data

Model	Accuracy	Precision	Recall	F1-Score	Support
Logistic Regression	0.9991	1.00	0.54	0.66	90

Model	Accuracy	Precision	Recall	F1-Score	Support
Decision Tree	0.9990	0.68	0.72	0.70	90
Random Forest	0.9996	0.99	0.73	0.84	90
Gradient Boosting	0.9993	0.89	0.63	0.74	90
MLP Classifier	0.9982	0.47	0.78	0.58	90
RNN	0.9986	0.91	0.11	0.20	90
LSTM	0.9992	0.76	0.77	0.76	90

Evaluating Models for Fraud Data MLFLOW

- Logistic Regression



- Decision Tree model

mlflow2.17.0ExperimentsModels

fraud_detection_experiment >incongruous-colt-425

Register model

OverviewModel metricsSystem metricsArtifacts

Details

Created at	2024-10-24 02:23:03
Created by	hayyuraga
Experiment ID	665122849546764073
Status	Finished
Run ID	cea98aac33d495bba57ce5e05af2040
Duration	7.9s
Datasets used	—
Tags	Add
Source	C:\Users\hayyuraga\AppData\Local\Programs\Python\Python12\ecommerce_fraud_detection_system\src\model_build_bce50b2
Logged models	sklearn
Registered models	—

Parameters (2)

Parameter	Value
model_name	Decision Tree
num_features	5

Metrics (1)

Metric	Value
accuracy	0.9049068901445577

- Random forest

mlflow2.17.0ExperimentsModels

fraud_detection_experiment >adorable-sow-200

Register model

OverviewModel metricsSystem metricsArtifacts

Description

No description

Details

Created at	2024-10-24 06:43:52
Created by	hayyuraga
Experiment ID	665122849546764073
Status	Finished
Run ID	bd66bbe5d304824bc1ec21404128d12
Duration	1.3min
Datasets used	—
Tags	Add
Source	C:\Users\hayyuraga\AppData\Local\Programs\Python\Python12\ecommerce_fraud_detection_system\src\model_build_bce50b2
Logged models	sklearn
Registered models	—

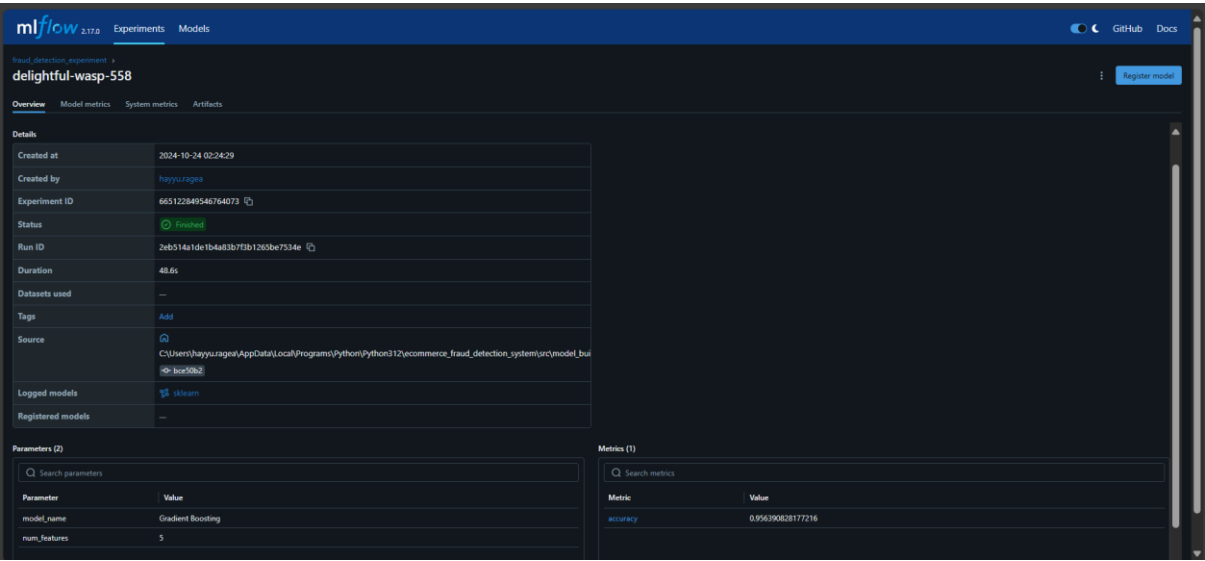
Parameters (2)

Parameter	Value
model_name	Random Forest

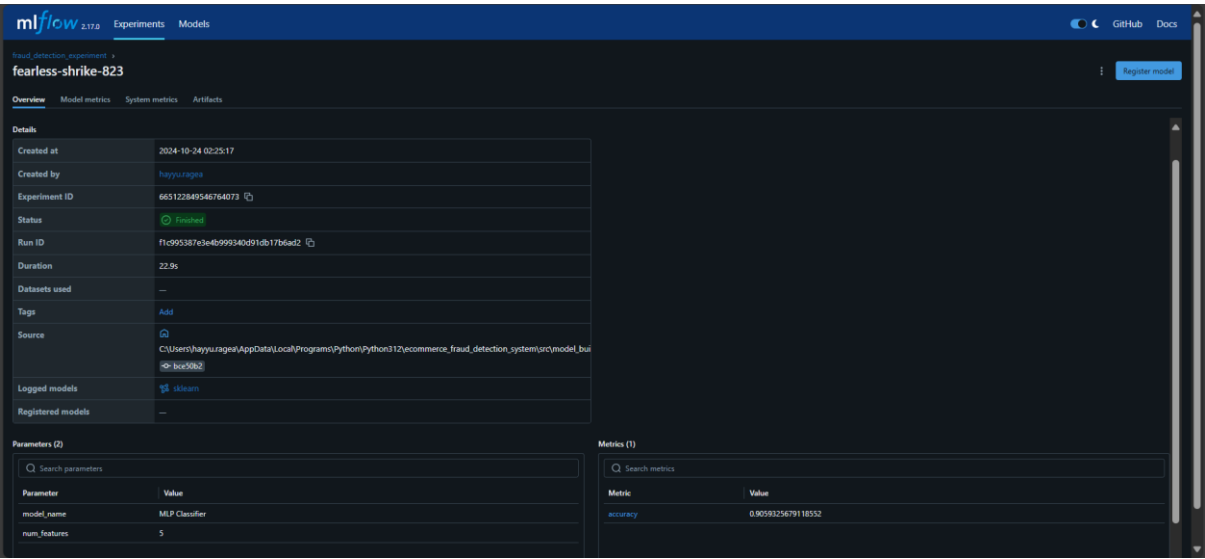
Metrics (1)

Metric	Value
accuracy	0.9564239155609966

- Gradient Boosting MLFLOW



- Multi-Layer Perceptron (MLP)



- Long Short-Term Memory (LSTM)

The screenshot shows the mlflow interface for an experiment named 'capable-calf-277'. The 'Details' section on the left lists the following information:

Field	Value
Created at	2024-10-24 02:26:18
Created by	hayyuragea
Experiment ID	665122849546764073
Status	Finished
Run ID	7a1bba3732184359acd695722eb57e5a
Duration	51.0s
Datasets used	—
Tags	Add
Source	C:\Users\hayyuragea\AppData\Local\Programs\Python\Python312\commerce_fraud_detection_system\src\model_bui bce50b2
Logged models	tensorflow
Registered models	—

The 'Parameters (2)' section on the left shows:

Parameter	Value
model_name	LSTM Model
num_features	5

The 'Metrics (1)' section on the right shows:

Metric	Value
accuracy	0.9564239155609966

- Recurrent Neural Network (RNN)

The screenshot shows the mlflow interface for an experiment named 'delicate-lark-732'. The 'Details' section on the left lists the following information:

Field	Value
Created at	2024-10-24 02:25:41
Created by	hayyuragea
Experiment ID	665122849546764073
Status	Finished
Run ID	87b027b2d81b4395a9db400006c4738b
Duration	37.4s
Datasets used	—
Tags	Add
Source	C:\Users\hayyuragea\AppData\Local\Programs\Python\Python312\commerce_fraud_detection_system\src\model_bui bce50b2
Logged models	tensorflow
Registered models	—

The 'Parameters (2)' section on the left shows:

Parameter	Value
model_name	RNN Model
num_features	5

The 'Metrics (1)' section on the right shows:

Metric	Value
accuracy	0.9564239155609966

Evaluating Models for Fraud Data

Model	Accuracy	Precision	Recall	F1-Score	Support
Logistic Regression	0.9057	0.00	0.00	0.00	2850
Decision Tree	0.9063	0.50	0.56	0.53	2850

Model	Accuracy	Precision	Recall	F1-Score	Support
Random Forest	0.9564	1.00	0.54	0.70	2850
Gradient Boosting	0.9564	1.00	0.54	0.70	2850
MLP Classifier	0.6179	0.16	0.71	0.26	2850
RNN	0.9564	1.00	0.54	0.70	2850

GitHublink:https://github.com/HaYyu-Ra/ecommerce_fraud_detection_analysis/blob/master/notebooks/model_biulding.ipynb

Conclusion

The models were trained and evaluated on both datasets, revealing key insights for fraud detection in e-commerce and credit card transactions. Random Forest, Gradient Boosting, and Logistic Regression delivered strong performances. Continued refinement is recommended, including addressing convergence warnings and model optimization.

This project has positioned Adey Innovations Inc. to enhance transaction security and build customer trust through advanced fraud detection technologies.