CREDIT CARD FRAUD DETECTION

USING LSTM & AUTOENCODER

Name: Dazzle Vincent

Batch : A

Roll No.: 43

Project Guide : Ms.Joice T

OBJECTIVE:

- The objective is to develop a system for credit card fraud detection using LSTM and Autoencoders.
- The system accurately identifies suspicious transactions and prevents fraud.It analyzes transaction patterns to detect unusual behavior that may indicate fraudulent activity.
- LSTM,XGBoost,Random Forest and Autoencoders are leveraged to provide quick and reliable detection.
- The system aims to:
- Minimize financial losses.
- Enhance payment security.
- Build trust among users.
- The goal is to reduce detection errors, ensuring proper handling of both fraud cases and genuine transactions.

ABOUT LSTM & AUTOENCODER:

- **LSTM** is a type of neural network that excels at understanding patterns in sequences over time, like a series of transactions made by a user. It can learn long-term dependencies, making it ideal for spotting unusual behaviors in transaction histories.
- **Autoencoders**, on the other hand, are used to identify anomalies. They work by learning to compress data into a smaller representation (encoder) and then reconstruct it back to the original (decoder).
- **Random Forest**, is an ensemble learning algorithm that improves classification by combining multiple decision trees. Each tree is trained on a random subset of data, and the final prediction is based on the majority vote. This approach reduces overfitting and enhances model robustness, making it effective for fraud detection.
- **XGBoost**,is a powerful algorithm known for its speed and accuracy. It builds decision trees sequentially, optimizing each new tree to correct previous errors. With built-in handling for imbalanced data, XGBoost efficiently detects fraudulent transactions.

EXISTING SYSTEM:

- **Detects fraud using predefined rules set by experts:** The system relies on manually crafted rules based on expert knowledge to flag potentially fraudulent transactions.
- **Utilizes traditional machine learning models to classify transactions as fraud or normal:** It applies basic machine learning models like decision trees or logistic regression to categorize transactions into fraudulent or legitimate.
- **Learns from previous fraud cases to improve detection:** The system improves over time by learning from previous fraud cases and using that knowledge to detect future fraud.
- **Assigns a fraud risk score based on transaction behavior:** The system gives a fraud risk score, but this score may not be highly accurate due to the limitations of the detection methods used.

PROPOSED SYSTEM:

- **Replaces predefined rules with AI-based models (LSTM, Random Forest, XGBoost) for better fraud detection:** The system uses advanced AI models that are more accurate, allowing for better identification of fraud patterns and improving detection accuracy.
- **Combines multiple advanced algorithms to improve classification accuracy:** Using a mix of algorithms, the system increases the overall accuracy by choosing the best-performing model for each transaction.
- **Improves fraud detection by learning from transaction sequences and patterns over time (using LSTM):** LSTM models can capture time-based patterns and trends, leading to more accurate fraud detection as they adapt to evolving behaviors.
- **Enhances fraud risk scoring using advanced predictive models for more precise decision-making:** The system uses improved risk scoring models, providing more accurate fraud risk assessments, which leads to better decision-making.

SYSTEM DESIGN:

1. **Data Processing Module**: Clean and prepare transaction data for the model.Handle missing values, normalize transaction amounts, extract features (e.g., transaction time, amount, location), and process the data into a suitable format for model training.
2. **Training Module:** Train the model to detect fraud using LSTM,Random Forest,XGBoost and Autoencoder.Split data into training and testing sets, select the models (LSTM for sequential data and Autoencoders for anomaly detection), and train them on the processed data.
3. **Testing Module**: Evaluate the performance of the combined LSTM,Random Forest and XGBoost models.Test the models on unseen transaction data, calculate accuracy, precision, recall, and F1 score, and analyze the confusion matrix to assess performance.

WORK DONE SO FAR:

- Analyzed a dataset sourced from Kaggle, containing over 8,000 transaction entries.
- Studied the dataset's structure, features, and patterns to identify potential fraudulent behavior.
- Conducted an in-depth study of: LSTM,Random Forest and XGBoost
- Autoencoders: Effective at detecting anomalies.Gained a strong understanding of the dataset and advanced machine learning techniques.

TOOLS & TECHNOLOGIES:

Dataset:CreditCardFraudDetectionDataset(Kaggle:https://www.kaggle.com/datasets/rupakroy/online-payments-fraud-detection-dataset)

Programming Language: Python

Libraries: pandas, numpy, scikit-learn, matplotlib, tensorflow, keras

Algorithms: LSTM (Long Short-Term Memory), Autoencoders,Random Forest,XGBoost

Deployment: PYQT for real-time fraud detection interface

DATASET:

	A	B	C	D	E	F	G	H	I	J	K
1	step	type	amount	nameOrig	oldbalanceOrg	newbalanceOr	nameDest	oldbalanceDes	newbalanceDi	isFraud	isFlaggedFraud
2	1	PAYMENT	9839.64	C1231006815	170136	160296.36	M1979787155	0	0	0	0
3	1	PAYMENT	1864.28	C1666544295	21249	19384.72	M2044282225	0	0	0	0
4	1	TRANSFER	181	C1305486145	181	0	C553264065	0	0	1	0
5	1	CASH_OUT	181	C840083671	181	0	C38997010	21182	0	1	0
6	1	PAYMENT	11668.14	C2048537720	41554	29885.86	M1230701703	0	0	0	0
7	1	PAYMENT	7817.71	C90045638	53860	46042.29	M573487274	0	0	0	0
8	1	PAYMENT	7107.77	C154988899	183195	176087.23	M408069119	0	0	0	0
9	1	PAYMENT	7861.64	C1912850431	176087.23	168225.59	M633326333	0	0	0	0
10	1	PAYMENT	4024.36	C1265012928	2671	0	M1176932104	0	0	0	0
11	1	DEBIT	5337.77	C712410124	41720	36382.23	C195600860	41898	40348.79	0	0
12	1	DEBIT	9644.94	C1900366749	4465	0	C997608398	10845	157982.12	0	0
13	1	PAYMENT	3099.97	C249177573	20771	17671.03	M2096539129	0	0	0	0
14	1	PAYMENT	2560.74	C1648232591	5070	2509.26	M972865270	0	0	0	0
15	1	PAYMENT	11633.76	C1716932897	10127	0	M801569151	0	0	0	0
16	1	PAYMENT	4098.78	C1026483832	503264	499165.22	M1635378213	0	0	0	0
17	1	CASH_OUT	229133.94	C905080434	15325	0	C476402209	5083	51513.44	0	0
18	1	PAYMENT	1563.82	C761750706	450	0	M1731217984	0	0	0	0
19	1	PAYMENT	1157.86	C1237762639	21156	19998.14	M1877062907	0	0	0	0
20	1	PAYMENT	671.64	C2033524545	15123	14451.36	M473053293	0	0	0	0
21	1	TRANSFER	215310.3	C1670993182	705	0	C1100439041	22425	0	0	0
22	1	PAYMENT	1373.43	C20804602	13854	12480.57	M1344519051	0	0	0	0
23	1	DEBIT	9302.79	C1566511282	11299	1996.21	C1973538135	29832	16896.7	0	0
24	1	DEBIT	1065.41	C1959239586	1817	751.59	C515132998	10330	0	0	0
25	1	PAYMENT	3876.41	C504336483	67852	63975.59	M1404932042	0	0	0	0
26	1	TRANSFER	311685.89	C1984094095	10835	0	C932583850	6267	2719172.89	0	0
27	1	PAYMENT	6061.13	C1043358826	443	0	M1558079303	0	0	0	0
28	1	PAYMENT	9478.39	C1671590089	116494	107015.61	M58488213	0	0	0	0
29	1	PAYMENT	8009.09	C1053967012	10968	2958.91	M295304806	0	0	0	0
30	1	PAYMENT	8901.99	C1632497828	2958.91	0	M33419717	0	0	0	0
31	1	PAYMENT	9920.52	C764826684	0	0	M1940055334	0	0	0	0
32	1	PAYMENT	3448.92	C2103763750	0	0	M335107734	0	0	0	0
33	1	PAYMENT	4206.84	C215078753	0	0	M1757317128	0	0	0	0
34	1	PAYMENT	5885.56	C840514538	0	0	M1804441305	0	0	0	0
35	1	PAYMENT	5307.88	C1768242710	0	0	M1971783162	0	0	0	0
36	1	PAYMENT	5031.22	C247113419	0	0	M151442075	0	0	0	0
37	1	PAYMENT	24213.67	C1238616099	0	0	M70695990	0	0	0	0
38	1	PAYMENT	8603.42	C1608633989	253	0	M1615617512	0	0	0	0
39	1	PAYMENT	2791.42	C923341586	300481	297689.58	M107994825	0	0	0	0

REFERENCES:

1. Robert B. Cleveland, William S. Cleveland, Jean E. McRae, Irma Terpenning. STL: A seasonal-trend decomposition procedure based on loss. Journal of Official Statistics, Vol 6., No. 1, 1990, pp. 3-73. http://www.nniiem.ru/file/news/2016/stl-statistical-model.pdf
2. A Tutorial on Deep Learning Part 2: Autoencoders, Convolutional Neural Networks and Recurrent Neural Networks http://robotics.stanford.edu/~quocle/tutorial2.pdf
3. Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735-1780.Hinton, G. E., & Salakhutdinov, R. R. (2006).
4. Reducing the dimensionality of data with neural networks. Science, 313(5786), 504-507.

THANKYOU