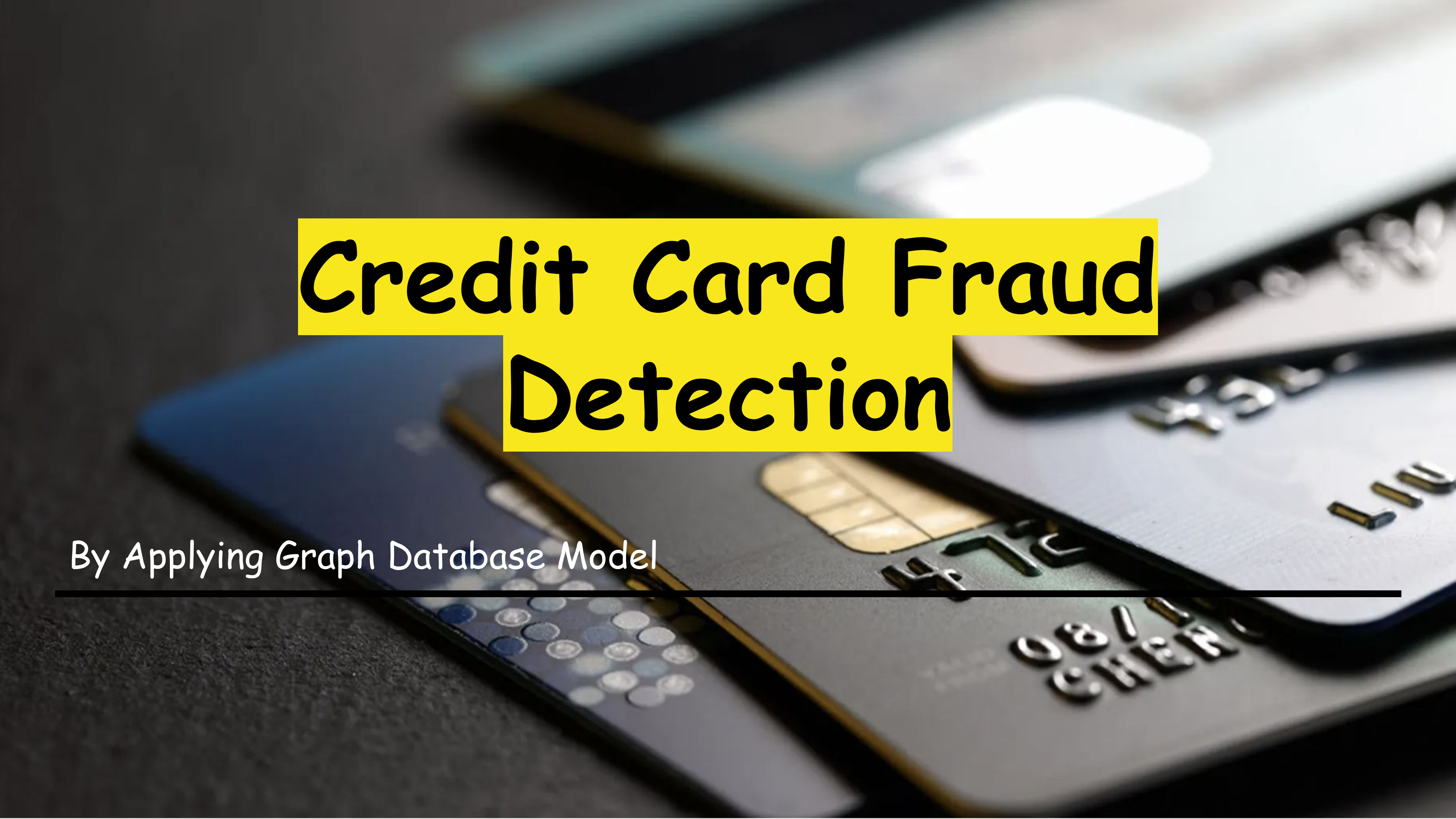


Credit Card Fraud Detection

By Applying Graph Database Model

Types of Frauds:

- Credit Card Transactions
- Insurance claim fraud
- Opinion fraud
- Social security fraud
- Scientific fraud

Types of Credit Card Fraud Activities

Traditional card related fraud

Merchant-Related fraud

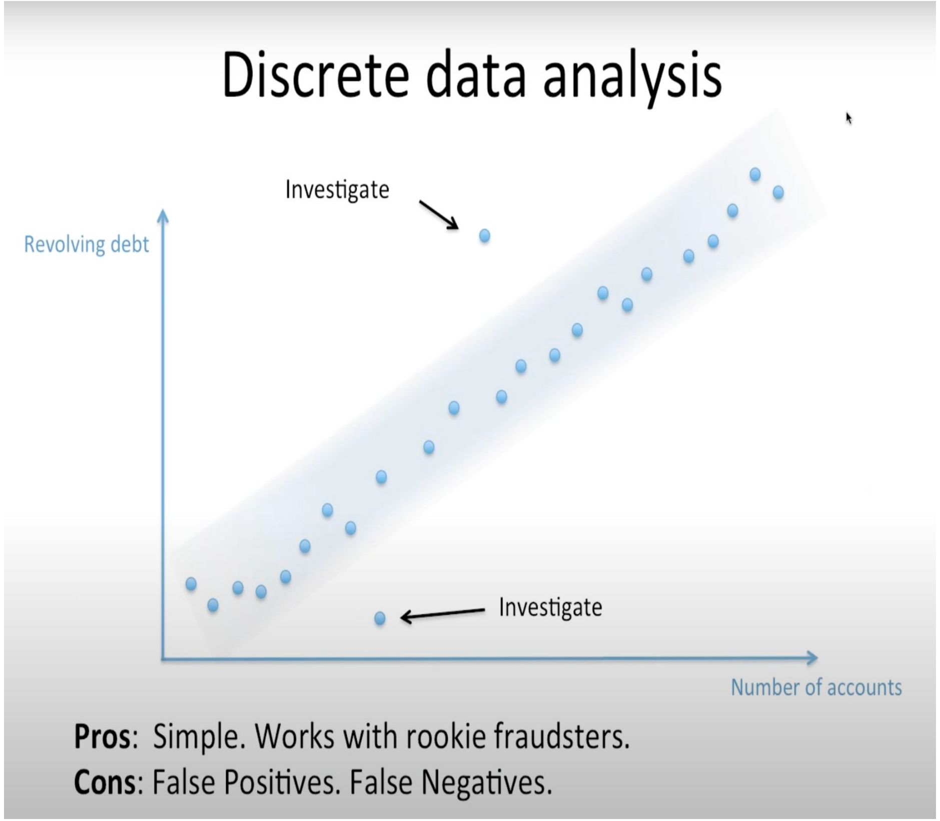
Internet-Related fraud

Skimming and phishing

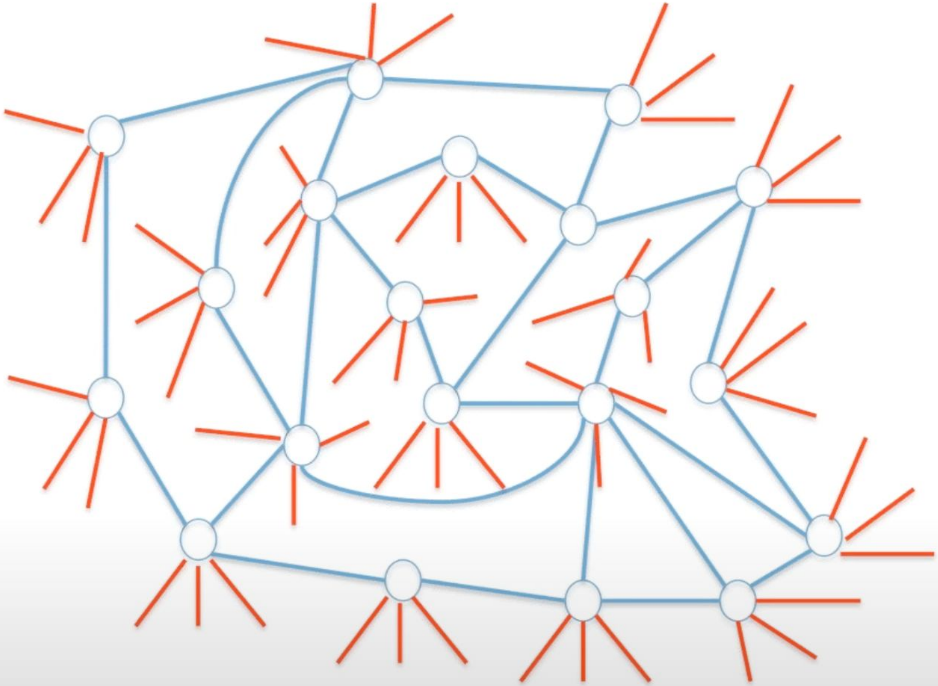
Idea of Fraud:

- An outlier value, an abnormal behavior or characteristic in a data set might often indicate that the person perpetrates suspicious activities.

- At a dataset level it looks something like this



Connected Analysis



Value: Effective in detecting some of the most impactful attacks: from organized rings.

*For example a **ten-person fraud bust-out is $1.5M**, assuming 100 false identities and 3 financial instruments per identity, each with a $5K credit limit.*

Challenge: Extremely difficult with traditional technologies

Problem Statement:

The performance metrics of the machine learning classification algorithms are evaluated initially without incorporation of any graph features by all the previous authors.

Objective of the paper:

The objective of this study is to extract the graph features from the graph model and then to study the performance of five supervised classification algorithms such as decision tree, random forest, k-nearest neighbour (k-NN), multilayer perceptron (MLP) and support vector machine (SVM) and two unsupervised algorithms such as Local Outlier Factor (LOF) and Isolation Forest (IF) applied on them.

Use of Graph Database and Neo4j tool

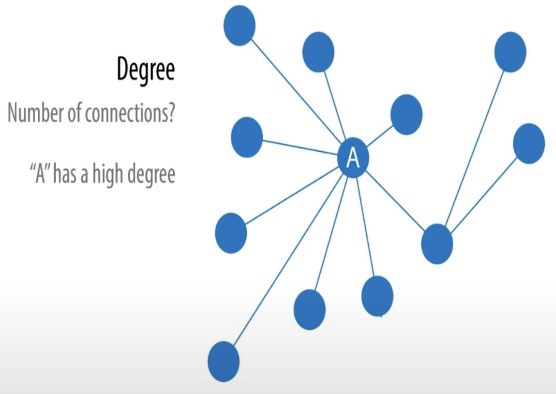
In this study the fraud detection has been carried out based on the number of records representing transactions using credit cards and a graph is being developed, where node represents the transaction type and edges represent the relationship between them.

To extract the graph feature from the given dataset, author has used Neo4j, as it has various graph algorithm libraries which can easily be applied to extract a graph feature. Graph algorithms used in this study are :

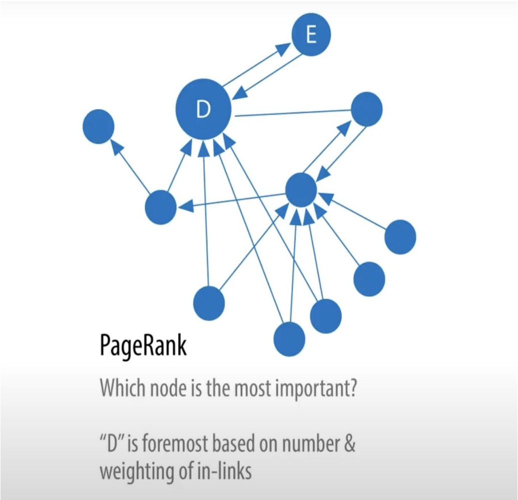
- Degree Centrality Algorithm
- PageRank Algorithm
- Label Propagation Algorithm

Visualization of graph algorithm:

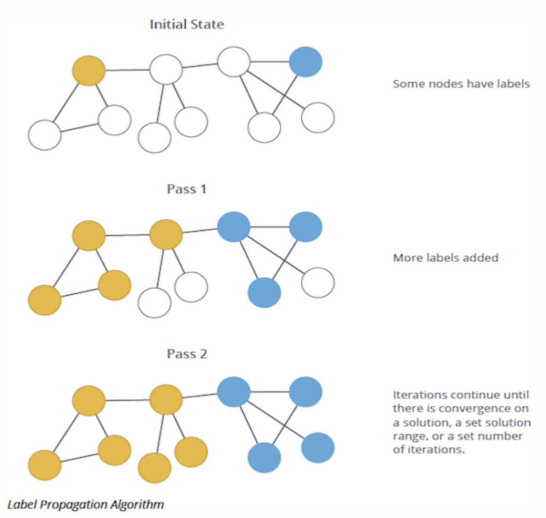
Degree Centrality



Page Rank

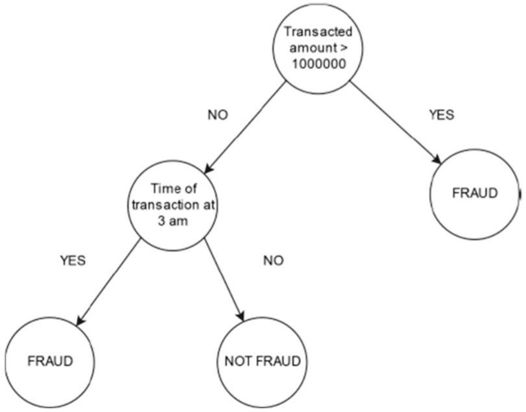


Label Propagation

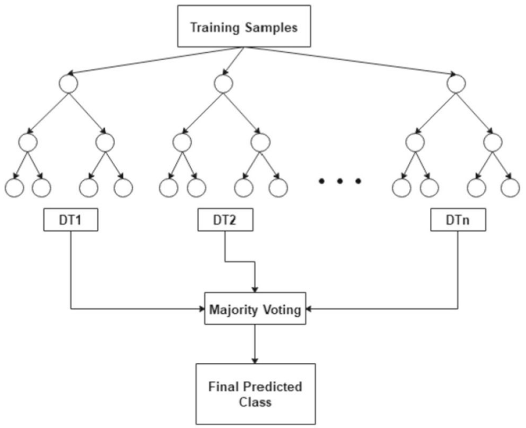


Machine Learning Algorithms used:

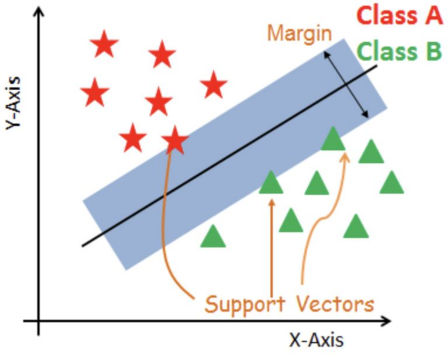
Decision Tree Algorithm



Random Forest Algorithm



SVM Algorithm



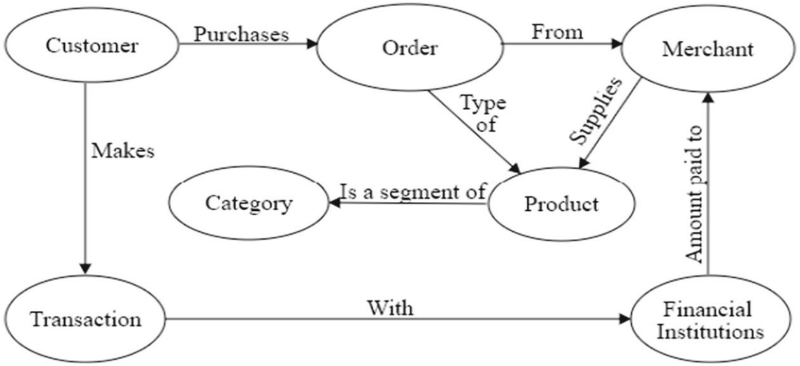
5 supervised ML algorithms used: Decision-Tree, Random Forest, SVM, K-NN, Multilayer perceptron algorithms.
2 unsupervised ML algorithms used: Local Outlier Factor(LOF), Isolation Forest(IF)

Dataset Used:

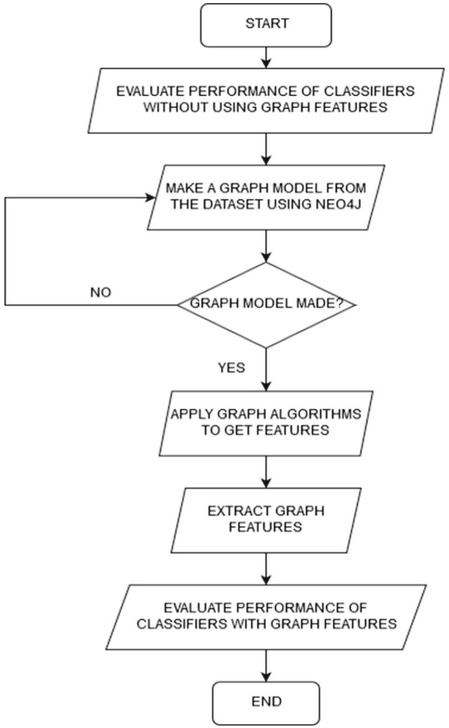
- A synthetic dataset from a financial payment system has been considered for analysis in this study. This is a synthetic dataset generated by the BankSim payment simulator and available on Kaggle.
- BankSim was run for 180 steps (approx. for six months); several times and the parameters were calibrated in order to obtain a distribution that is close enough to be reliable for testing and fraudulent transactions were injected into it.
- Thus, it has 594643 records in total, out of which 587443 (98.79%) transactions are normal payments and 7200 (1.21%) are fraudulent transactions.

# step	A customer	A age	A gender	A merchant	A category	# amount	# fraud
0 – 179	4112 unique values	'2' 31% '3' 25% Other (260202) 44%	'F' 55% 'M' 45% Other (1693) 0%	'M1823072687' 50% 'M348934600' 35% Other (89524) 15%	'es_transportation' 85% 'es_food' 4% Other (63270) 11%	0 – 8.33k	0 – 1
0	'C1093826151'	'4'	'M'	'M348934600'	'es_transportation'	4.55	0
0	'C352968107'	'2'	'M'	'M348934600'	'es_transportation'	39.68	0
0	'C2054744914'	'4'	'F'	'M1823072687'	'es_transportation'	26.89	0
0	'C1760612790'	'3'	'M'	'M348934600'	'es_transportation'	17.25	0
0	'C757503768'	'5'	'M'	'M348934600'	'es_transportation'	35.72	0
0	'C1315400589'	'3'	'F'	'M348934600'	'es_transportation'	25.81	0
0	'C765155274'	'1'	'F'	'M348934600'	'es_transportation'	9.1	0

Proposed Graph Model:



Flowchart of the method used:



Performance Parameters:

- **Accuracy**:

  (TP + TN) / (TP + TN + FP + FN)

- **Precision**:

  TP / (TP + FP)

- **Recall or Sensitivity:**

  TP / (TP + FN)

- **F-1 Score:**

  2TP / (2TP + FP + FN)

- **MCC**:

  (TP * TN - FP * FN) / (TP + FP) * (TP + FN) *(TN + FP) *(TN + FN)^1/2

- **ROC - AUC Score:**

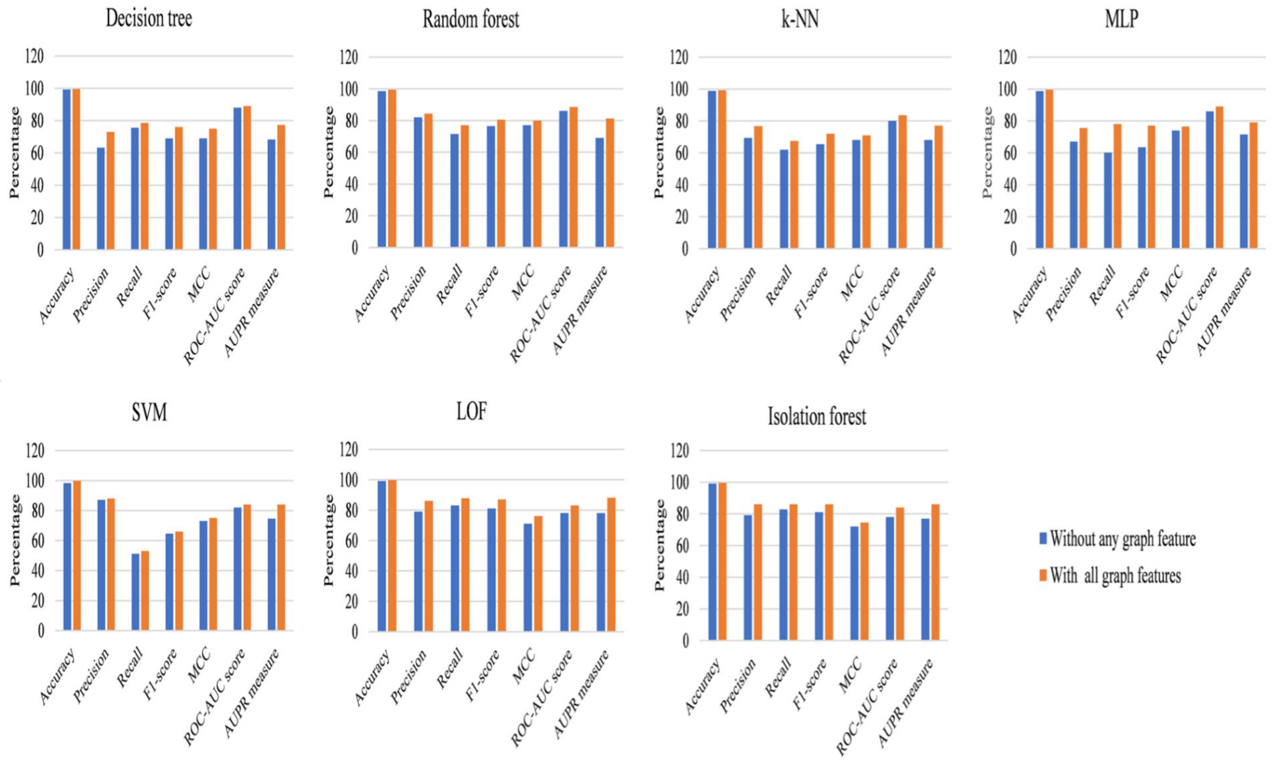
  TPR = T P/(T P + FN)
  FPR = F P/(T N + F P)

- **AUPR Measure:**

Comparison of performance Parameters:

Classification model	All graph features	Accuracy	Precision	Recall	F1-score	MCC	ROC–AUC score	AUPR measure
Decision tree	Without graph features	99.365	63.333	75.727	68.978	68.778	88.038	68.232
Decision tree	With graph features	99.477	73.097	78.545	75.723	75.253	89.081	77.332
Random forest	Without graph features	98.451	82.025	71.472	76.386	76.850	86.250	69.119
Random forest	With graph features	99.534	84.311	77.222	80.611	80.286	88.512	81.283
K-NN	Without graph features	98.789	69.266	62.121	65.500	68.136	80.012	68.009
K-NN	With graph features	99.326	76.689	67.538	71.823	71.196	83.623	77.198
MLP	Without graph features	98.672	67.184	60.178	63.488	74.138	85.925	71.440
MLP	With graph features	99.423	75.424	77.873	76.629	76.286	88.782	79.196
SVM	Without graph features	98.192	87.192	51.222	64.533	72.942	82.128	74.658
SVM	With graph features	99.789	87.922	53.134	66.238	75.218	84.129	83.997
LOF	Without graph features	99.184	78.991	83.192	81.037	71.022	78.190	78.238
LOF	With graph features	**99.753**	86.210	**87.791**	86.993	75.991	83.220	88.007
IF	Without graph features	99.070	79.205	82.771	80.948	71.776	78.213	77.237
IF	With graph features	99.539	86.003	86.021	86.012	74.390	84.128	85.774

Comparison Analysis of performance metrics using all the 3 graph features



Conclusion:

- Thus, it may be considered that the inclusion of graph features from the graph algorithms helps to improve the performance of the machine learning algorithms
- LOF algorithm has the improved accuracy result and highest recall value with all the graph features.

Future Research :

Other graph algorithms such as closeness centrality, betweenness centrality, Louvain modularity, node clustering coefficient and average clustering coefficient may be considered in future to extract features and applying deep learning algorithms to classify mere critically, the transactions into fraudulent and authentic transactions.

Thank you for your attention!