Proc. Natl. Acad. Sci., India, Sect. A Phys. Sci. (April 2024) 94(2):201–212
https://doi.org/10.1007/s40010-024-00871-1

SCIENTIFIC RESEARCH PAPER

A Robust Framework for fraud Detection in Banking using ML and NN

Astha Vashistha[1] · Anoop Kumar Tiwari[1] · Priyanshi Singh[1] · Paritosh Kumar Yadav[1] · Sudhakar Pandey[1]

Received: 5 April 2023 / Revised: 11 October 2023 / Accepted: 17 January 2024 / Published online: 19 February 2024

Abstract Banking fraud is a problem that is becoming more and more serious, along with considerable monetary losses, damage to the bank's brand, loss of client and customer confidence. Fraud identification and prevention are major challenges for many financial organizations, retail firms, and e-commerce companies. Fraud detection is used to both identify and stop fraudsters from obtaining goods or bugs illegally. In the same vein, this research will conduct a feasibility study to determine the best fraud detection strategy. We provide a list of the tried-and-true methods for spotting fraud. To avoid fraud detection, many techniques like Deep Neural Network, Support Vector Machine, Multilayer Perceptron, K-Nearest Neighbors, Random Forest, XG Boost, LGBM, and Decision Tree were used. The dataset was built from 20,000 entries on Kaggle, each having 114 attributes. Before using machine learning and neural network approaches, the dataset is balanced using the Synthetic Minority Over-Sampling Method. Following the analysis of the dataset using a number of methods, it was determined that Random Forest, Decision Tree, XG Boost, and LGBM all had 100% accuracy. This demonstrates that the model outperformed other models by balancing the dataset.

Keywords Illegitimate financial activity · Machine learning · Neural network · Random forest · Decision tree · XG boost · LGBM

✉ Sudhakar Pandey
spandey96@gmail.com

[1] National Institute of Technology Raipur, Raipur, India

1 Introduction

With the development of cutting-edge technology and global communication, fraud has been discernibly rising. The two basic strategies to avoid fraud are detection and prevention. In the era of internet, fraud detection is a complex and sophisticated task [1]. To prevent fraud, expert systems and intelligent software are in use. Identification of fraud is essential for mitigating losses. However, fraudsters continue to invade by effectively circumventing current and newly created anti-fraud procedures.

The industry uses the effective technology of graph databases to identify fraud scenarios and other frauds that are comparable to them. Unfortunately, there is reportedly no efficient method of preventing fraud. The best way to proceed is to develop fraud detection methods. This can be done using both individual data elements and the relationships between them. These connections sporadically go unrecognized until it is too late [2]. The connections between various data elements are crucial for spotting fraud because they can provide the most illuminating hints. A bank transaction or pattern of activity that raises a red flag can be investigated by the bank's investigating team. For years, banks have improved their ability to detect fraud. According to the present scenarios, the majority of banks rely on machine learning tools from the previous generation that are configured to detect particular types of behaviors (e.g., large, aberrant transactions to unknown accounts).

1.1 Challenges of Fraud Detection

Fraud detection has many challenges over the organization, and there are challenges that complicate the fraud detection process [3]. These are the following:

(a) *Over-time changing of fraud patterns* It addresses that the fraudster is the toughest job because they always used to find the new innovative patterns to commit the act.
(b) *Imbalance in the class* It means imbalance in the classification of fraud detection models (to classify the transaction into fraud and non-fraud).
(c) *Interpretations of Models* Due to the fact that models usually assign a score indicating whether a transaction is likely to be fraudulent or not without providing an explanation, this limitation is related to the idea of explaining ability.
(d) *Time consumption due to Feature Generation* To produce a complete feature set, it may need a lot of time, which slows down the fraud detection process.

However, there are several solutions to these problems, and ensemble modeling and the Explain Ability Technique [3] are two of them. Deep learning (DL) is the most promising method for classifying data, and its main component is the Feedforward-Neural-Network (FNN), also known as the Multilayer Perceptron (MLP). Different structures, such as classification, attention mechanisms, and convolutional feature maps, have been developed using this important advancement [2]. Misuse of the system, such as fraud in financial systems conducted for personal or organizational financial gain, misrepresentation, or spoofing can cause serious problems for companies in this competitive environment. Credit fraud has become a major issue in recent years, costing banks and financial institutions billions of dollars annually.

1.2 Contributions

The goal of this study is to put fraud detection algorithms based on machine learning (ML) and neural networks (NN) into practice. The following are the paper's practical contributions:

(a) The dataset is unbalanced since fraud accounts for 26.5683 percent of the data and not for 73.43 percent of the data. Therefore, using the approaches directly will always get the best results.
(b) The Synthetic Minority Over-Sampling Technique (SMOTE) is employed in the over-sampling approach to balance the unbalanced dataset used for fraud detection in banks.
(c) To identify fraud, various techniques including MLP, DNN, SVM, RF, KNN, DT, XG Boost, and LGBM are utilized. Their Accuracy, Precision, Recall, and F1 Score are also evaluated.
(d) A machine learning model based on the Random Forest, XG Boost, Decision tree, and LGBM algorithms achieves the best accuracy of 100%.

1.3 Framework Organization

The objective of the paper is to detect fraud in the bank dataset. Examine some of the papers related to the bank fraud detection, then provide the related work of relevant research in Sect. 2. Section 3 provides the dataset description, proposed work, and description of all the algorithms which are applied in this paper. After that, Sect. 4 represents the results and finally concludes the paper in Sect. 5.

1.4 Statement of Significance

The research pioneer's novel fraud detection solutions by carrying out an extensive feasibility analysis and applying cutting-edge methods like DNN, SVM, MLP, KNN, RF, XG Boost, LGBM, and Decision Tree. The achievement of an unparalleled 100% accuracy rate with RF, DT, XGB, and LGBM is significant because it demonstrates their supremacy in reducing fraud risks, maintaining financial integrity, and fostering confidence in the banking industry.

2 Related Work

The different types of fraud observed in which they used supervised learning methods SVM with spark to represent normal and abnormal behavior of the customers [2]. But as a result, Black Propagation Networks given best performance compared to Support Vector Machine (SVM) where accuracy and average prediction reached its maximum when training data ratio arrives at 0.8.

With the help of different ML models like LR, KNN, DT classification, and RF, the transaction of the dataset is tested individually [4]. First, define the detection tasks which are attributes of the dataset, the metric choice, and any technique to control such unbalanced datasets. This leads to the fact that underlying pattern generating the dataset results in which random forest has given highest accuracy of 96.64 percent as compared to all other algorithms.

The effectiveness of two different random forest models was investigated [5]. This study makes use of a real-world B2C dataset for credit card transactions. There are still certain issues, like skewed data, even if random forest models produce good results on small sets of data. And compared to RF I, which had accuracy of 91%, they obtained the maximum accuracy in RF II, which was 96.77%.

Fraud is recognized using the Harmony Search Algorithm and ANN method [6]. The suggested strategy looks for hidden patterns in the data of legitimate and fraudulent

consumers. The findings of the suggested system demonstrate that it has an adequate capability in fraud detection given that fraudulent conduct could be identified and halted before it occurs. According to the comparative results, the German dataset provided the suggested system with the best accuracy, which is 86. Additionally, 87 is the greatest result for the same Recall criteria.

Using machine learning methods and neural networks, this study aims to develop the best model that can distinguish between fraudulent and legitimate transactions [7]. The project's goal is to develop complicated machine learning models for prediction and understanding of the dataset using classification ML algorithms, statistics, calculus (differentiation, chain rule, etc.), and linear algebra. They employed Logistic Regression (LR) to reach accuracy of 94.84%, naive Bayes to get accuracy of 91.62%, decision trees to achieve accuracy of 92.88%, and ANN (Artificial Neural Network) to achieve accuracy of 98.69% in deep learning, outperforming all other techniques.

A novel method for detecting fraud in which consumers are classified according to their transactions and behavioral patterns are extracted to create a profile for each cardholder [8]. They used local outlier factor, IF, SVM, LR, DT, and RF for fraud detection. They balanced the dataset by using SMOTE and discovered that the classifiers were working more effectively than previously. They used local outlier factor, IF, SVM, LR, DT, and RF for fraud detection; then finally, it was discovered that the algorithms for DT, RF, and LR provided the best results.

As in the banking industry, enormous amounts of data are constantly being produced [9]. Simulations combining programming software and data mining tools show that using association rules to classification algorithms like KNN (K-nearest neighbor) can significantly increase accuracy. In order to boost the accuracy of fraud detection in the electronic banking system, they used a new method of combination categorization employing clustering and association criteria [10]. Although accuracy in this algorithm increased because of their usage of techniques like KNN, association rules like the Apriority algorithm, and clustering transactions, they occasionally fell short of 100% accuracy. Rules for data classification and association can be updated, and additional techniques like NN are applied for future applications.

The Long Short-Term Memory (LSTM) [11] methodology is the foundation of a deep learning (DL)-based strategy for the identification of financial fraud. This model aims to improve both the efficiency and accuracy of the current detection methods in the context of large data. A real dataset of credit card frauds is used to assess the suggested model, and the outcomes are compared with a DL model already in existence called the Auto-encoder model and various other ML methods. The trial outcomes showed that the LSTM performed flawlessly, achieving 99.95% accuracy in less than a minute.

Whether model-free or model-based, the most fascinating aspect of Deep Reinforcement Learning (DRL) is to achieve reward prediction in terms of Bellman-like self-consistent equations [12]. It is still debatable whether the brain implements this prediction. If such a structure does exist, it has a wide range of consequences. DRL may serve as a foundation for decision-making in a complex world with numerous agents as well as for the recognition of the "self" in such a universe. Thus, by merging ideas and information from numerous study domains, including machine learning, control theory, and fraud detection in banking, the theory of DRL will advance.

To balance the weight of the fraudulent and legitimate transactions, consider using class weight-tuning hyperparameters [13], where specially used Bayesian optimization to optimize the hyperparameters while maintaining real-world problems like unbalanced data. And proposed the weight-tuning as a preprocess for unbalanced data, where Cat Boost and XG Boost also proposed to improve the Light GBM performance by accounting for the voting mechanism. Evaluated separately all the three algorithms using a five-fold cross-validation method. Then achieved the best level criteria in LGBM and XGB in ROC-AUC: 0.95, Recall: 0.80, Precision: 0.79, F1 Score: 0.79, and MCC: 0.79 but using the Bayesian optimization method to tune the hyperparameters then achieved ROC-AUC: 0.94, Recall: 0.82, Precision: 0.80, *F1* Score: 0.81, and MCC: 0.81.

The technique, dataset, and evaluation of the paper which deals with the detection of anomalies in blockchain networks and make use of ML or NN methods are described in Table 1. Comparison of applied datasets and evaluation of the paper which deals with the detection of anomalies in blockchain networks and make use of ML or NN methods are described in Table 2.

3 Proposed Work

3.1 Dataset Description

In proposed method, dataset of fraud cases of a bank is used, which consist of 20 K records with binary values as entries and total 114 columns in which 113 features and 1 column represents as target variable. It is a labeled dataset which is present in Kaggle [18].

Base Learner These are the members of the ensemble who are tactically combined as an individual or as a component. It must concentrate on effectively categorizing the examples with the highest weights while ardently avoiding over-fitting. The base learner of the suggested method is

Table 1 Comparison of applied datasets in related studies

Sr. no.	References	Method/model used	Bank sim dataset	Germen dataset	Resilient distributed dataset	Fraud card transaction dataset
1	[4]	LR, KNN, DT, RF	×	✓	×	×
2	[6]	ANN technique and Harmony search algorithm	✓	×	×	✓
3	[7]	LR, NB, DT, ANN Model	✓	×	×	×
4	[14]	SVM, Naive Bayes (NB), KNN, and LR	✓	✓	✓	✓
5	[8]	Local outlier factor, Isolation Forest, SVM, LR, DT, RF used with SMOTE	✓	✓	✓	×
6	[15]	SVM	✓	✓	×	×
7	[16]	RF	✓	✓	✓	✓
8	[9]	DT, K-means, NB, SVM	×	✓	×	✓
9	[10]	KNN	×	×	✓	✓
10	[17]	SVM	✓	✓	✓	×

Table 2 Literature analysis table

S. no.	References	Dataset	Methodology	Evaluation parameters	Result
1	[4]	Bank sim	LR, KNN, DT, RF	Accuracy	LR 91.38, KNN 92.66, DT 96.11, RF 96.6
2	[6]	German dataset	ANN technique and Harmony search algorithm	Training, testing, and Recall	80.53 ± 0.88 and 88 86%
3	[7]	Fraud card transaction from Europe	LR, NB, DT, ANN model	Accuracy, Precision, Recall	LR 94.84, 97.58, 92.00, NB 91.62, 97.09, 84.82, DT 92.88, 99.48, 86.34, ANN 98.69, 98.41, 98.98
4	[14]	Consist of fraud transactions log file and all transactions log file	SVM, Naive Bayes (NB), KNN, and LR	Accuracy	91%, 81%, 74% and 72%
5	[8]	Transaction contained by cardholder	Local outlier factor, Isolation Forest, SVM, LR, DT, RF used with SMOTE	Accuracy, Precision, Matthews correlation coefficient	0.8990, 0.0038, 0.0172, IF: 0.9011, 0.0147, 0.1047, SVM: 0.9987, 0.7681, 0.5257, LR: 0.9990, 0.875, 0.6766, DT: 0.9994, 0.8854, 0.8356, RF: 0.9994, 0.9310, 0.8268
6	[15]	Resilient distributed dataset	SVM	Training and testing accuracy	98.78% and 99.86%
7	[16]	Consist of fraudulent and legitimate B2C transactions	RF	Accuracy, Precision, Recall	98.67%, 32.68%, 59.62%
8	[9]	Germen credit, used for customer retention problem	DT, K-means, NB, SVM	Training and testing	72% and 98%
9	[10]	Legal and fraud transaction	KNN	Accuracy, Precision, Recall	98.5%, 100%, and 98%
10	[17]	Fraud transactional	SVM	Accuracy	80%

the target value, which is a binary value in the dataset and is used to determine whether a transaction is fraudulent or not.

In the dataset, 26.5683 percent of the data are fraud and 73.43 percent of the data are having no fraud. As in the Fig. 1, 0 is representing the non-fraud data and 1 is representing the fraud data. Figure 2 shows the unsampled count of data where 0 and 1 describe the fraud and non-fraud transaction. And Fig. 3 is the box plot graph of the dataset where the targets show the fraud transactions in respect to the "Unnamed: 0" feature of the dataset.

3.2 Proposed Method

The proposed data model is built on the bank transactional dataset in which target class shows the whole fraud transactional data and non-fraud transactional data. In the proposed algorithm, the first step is to import the dataset for detecting

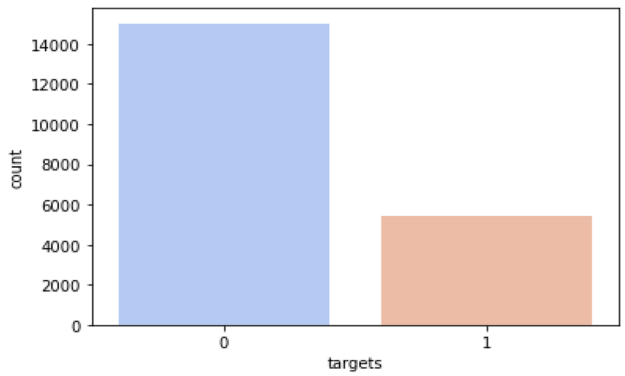


Fig. 1 Fraud and non-fraud dataset description

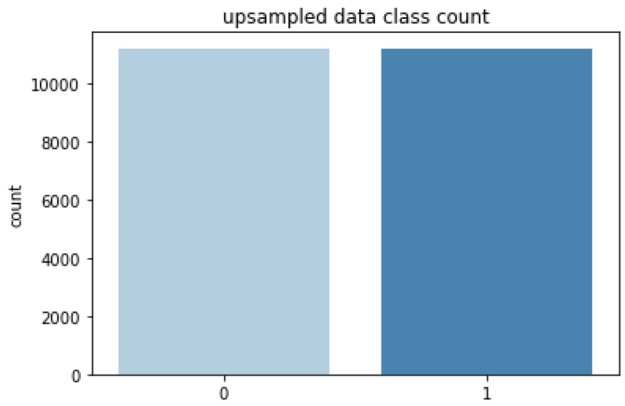


Fig. 2 Unsampled count of data class

the fraud as shown in Fig. 4, then in the next step, preprocess the dataset in which first removed all the NULL values and normalized it, but there is misbalancing in the dataset for that used the over-sampling method which is SMOTE due to which got balanced dataset. Then performed the training and testing in which 70% part of the dataset is taken for training and the remaining part isfor the testing. As per the presented paper [8], the training and testing part is divided into 70 and 30 percent, but while changing the ratio of the training and testing up to 80 and 20%, it is giving best accuracy compared to that paper. Then applied and saved the model after which determined the classification report in which analyzed the different parameters such as Accuracy, Precision, Recall, and *F*1 Score.

3.3 Over-sampling with SMOTE

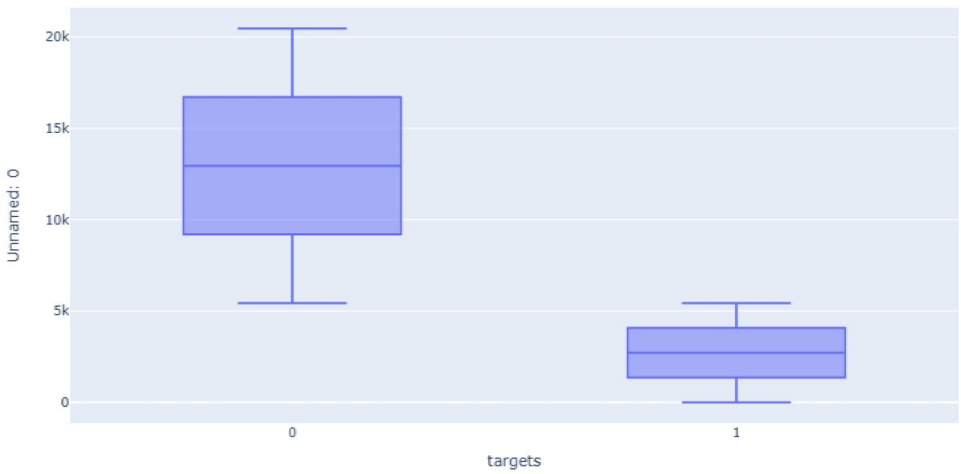
Synthetic Minority Over-Sampling Technique (SMOTE) is a better approach for handling imbalanced data in classification issues [19]. A categorical variable's observed frequencies are said to be unbalanced when they considerably differ from all of its possible values. There are typically many observations of one type and few of another.

It aims to balance the distribution of classes by randomly increasing minority class samples and duplicating them. SMOTE combines existing minority instances to produce new minority instances. It uses linear interpolation to produce virtual training records for the minority class. These synthetic training records are picked at random from the k-nearest neighbors for each example in the minority class. The data are recreated after the over-sampling process and can then be exposed to a variety of categorization models.

3.4 Support Vector Machine (SVM)

SVM, one of the strongest ML algorithms, was first introduced in the 1990s and is largely used for pattern recognition [20]. It is used to address a variety of pattern classification issues, including those involving false card detection, face identification, voice recognition, text categorization, and picture recognition, among others. A powerful technique for data separation in many fields, pattern recognition attempts to classify data based on either statistical

Fig. 3 Box plot of fraud and non-fraud transaction



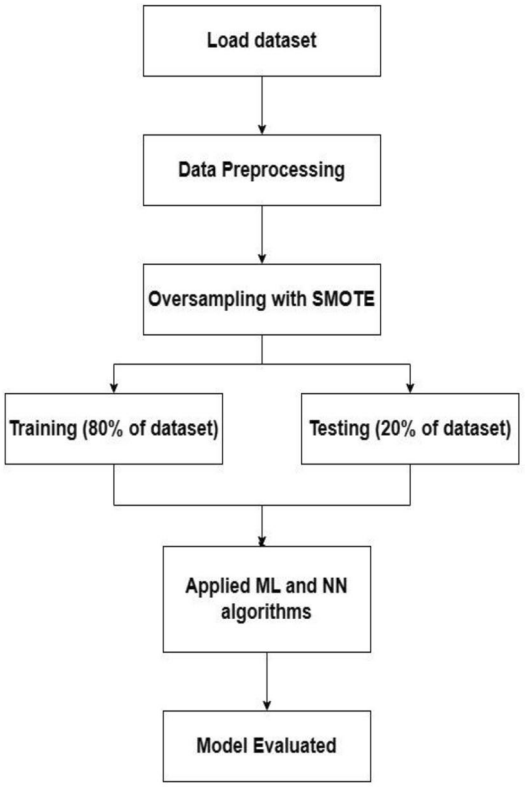


Fig.4 The overall workflow of the proposed scheme for the detection of bank fraud

information extracted from raw data or other factors [21]. SVM is a supervised machine learning technique. Given a collection of training examples, each of which has been classified as falling under one of the several categories, this training approach creates a model that predicts the category of the new example. SVM is having two types which are the following:

Linear SVM Dataset can be classified into some classes using straight line as shown in Fig. 5 which is also called as linearly separable.

Nonlinear SVM In this, dataset cannot be classified using the straight line as shown in Fig. 6 which is also called as nonlinearly separable.

Support vector, hyperplane, and marginal distance are present in SVM. The data points or vectors that are closest to the hyperplane and have the most impact on where the hyperplane is located are known as support vectors [18]. Because they support the hyperplane, these vectors are known as support vectors. Even though there may be numerous lines or other decision boundaries used to divide the classes in n-dimensional space, we still need to select the best boundary to help classify the data points.

The SVM hyperplane is this ideal boundary. Here, marginal distance sets this ML method apart from others.

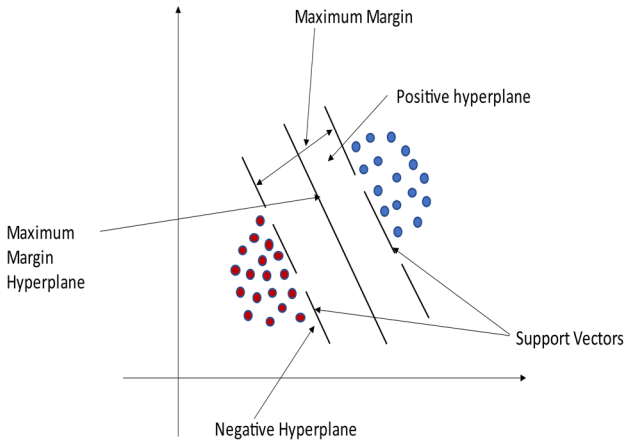


Fig. 5 Linear separable

Basically, it is the classifier used to predict and classify pattern. It was created using the Structural Risk Minimization principle. The SVM's decision function in a binary classification problem is depicted in Eq. (1) [21].

$$f(x) = sgn(x \cdot w) + b \tag{1}$$

where input vector is x which contains weight and constant is b. Equation (1) is used to determine the boundaries of a decision between two classes. The SVM must learn the parameter values for w and b during the training phase, and the value of b is determined by maximizing the margin of separation between the two classes.

The margin maximization between the two classes serves as the foundation for the SVM's criterion. To find the hyperplane between the two hyperplanes H: $y = w \cdot x + b = 0$.

Two hyperplanes are $H1$ [17]: $y = w \cdot x + b = +1$ and $H2$: $y = w \cdot x + b = -1$.

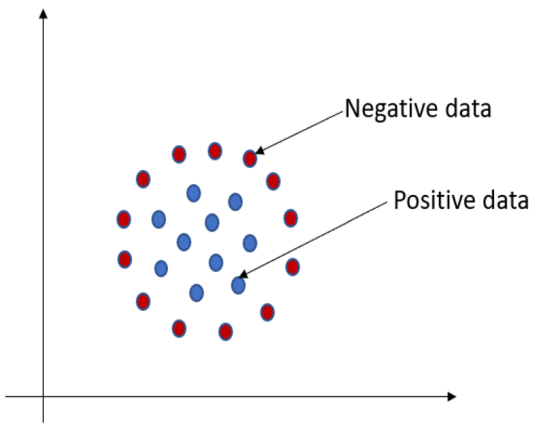


Fig. 6 Nonlinear separable

Here, the margin is 2/||w||, where w is the norm of vector w. The margin is soft in cases where separation is not absolute. There is a potential for classification inaccuracy. Errors in categorization should be kept to a minimum. It is minimized by introducing the slack variable ξ_i.

Equation (2) for the hyperplane:

$$Y_i * w^T x_i + b_i >= 1 \tag{2}$$

The optimization problem for the calculation of w and b can thus be defined by Eq. (3).

For finding error [20]:

$$(w*, b*) \min ||w||2/2 + ci \sum_{i}^{n} \xi_i \tag{3}$$

SVM uses SVM Kernels which is used to covert the low dimensional dataset to the high dimensional dataset which means it can easily classify the nonlinear separable hyperplane. The SVM function is called as kernel-aids in issue solving. They provide solutions for challenging calculations. The good thing about kernel is that it makes it possible to travel to higher dimensions and yet carry out calculations with ease.

3.5 Random Forest (RF)

The RF Algorithm, a very popular supervised machine learning technique, is used to address classification and regression problems [16]. A forest is made up of numerous different species of trees as described in Fig. 7, and the forest will be more vigorous the more trees there are. Similar to this, the number of trees in an RF algorithm increases the algorithm's accuracy and ability to solve problems. A classifier known as RF employs numerous DT on various subsets of the input data to boost the predicted accuracy of the dataset. It is based on the concept of ensemble learning, which is the practice of combining different classifiers to solve a difficult problem and improve the performance of the model.

3.6 K-Nearest Neighbor (KNN)

KNN is one of the first methods evaluated when there is little or no prior knowledge about the distribution of the data [24]. It is among the simplest and most fundamental classification methods. KNN classification was developed because of the requirement to perform discriminant analysis when precise parametric estimates of probability densities are unknown or difficult to determine.

The main goal of this algorithm is to find nearest neighbors of a given query point due to which class label assigned to that point. For determining the distance metrics, there are some distance measures like Euclidean, Manhattan distance, etc.

As K represents the number of nearest neighbors, so after determining the parameter K, calculate the distance between the neighbors, then sort it into the ascending order, and then it will follow the steps as shown in Fig. 8 and decide the class of unlabeled pattern based on majority vote.

3.7 XG Boost (XGB)

Extreme Gradient Boosting, also known as XGB, is a boosting technique based on gradient-boosted decision trees. One way that XGB differs from gradient boosting is by using a stronger regularization strategy to lessen over-fitting [24]. "XGB" which is an open-source package offers ML algorithms that use gradient boosting techniques. A class for classification that is compatible with the scikit-learn API is called XGB classifier. It is not able to learn from training set immediately; instead, it stores the dataset, and at the time of classification, it performs an action on the dataset due to which it is also known as lazy learner algorithm.

3.8 LGBM

The LGBM (light gradient boosting machine) is a DT-based gradient boosting technique that is used to raise the

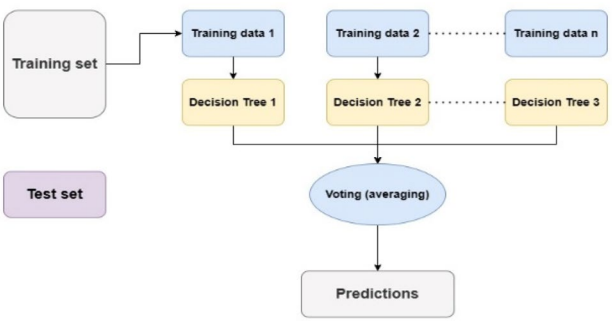


Fig. 7 Random forest flowchart [22]

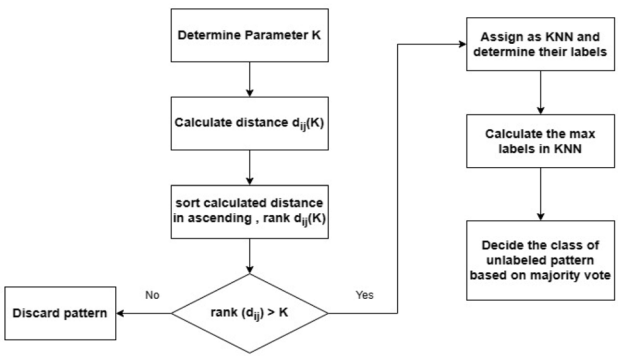


Fig. 8 KNN algorithm [23]

performance of a given classification model while using less memory [24]. It is utilized in a variety of machine learning applications, including ranking and classification. It is based on two cutting-edge strategies. The two methods, Exclusive Feature Bundling (EFB) and Gradient-based One Side Sampling (GOSS), were created to solve the shortcomings of the histogram approach used in Gradient Boosting Decision Tree (GDBT) models. The EFB and GOSS techniques are used to produce the properties of the LGBM model.

3.9 Decision Tree (DT)

Classification and regression issues can be resolved using the supervised learning technique known as a decision tree; however, this approach is frequently preferred [25]. It is a tree-structured classifier, where each leaf node represents the classification outcome and inside nodes represent the features of a dataset.

There are two nodes in this: the Leaf Node and the Decision Node as shown in Fig. 9. Decision nodes are used to make decisions and have many branches, whereas Leaf nodes are the outcomes of decisions and do not have any more branches. The test is run or judgments are made using the features of the provided dataset.

3.9.1 Deep Neural Network (DNN)

A neural network having more than two layers and a certain level of complexity is referred to as a DNN. It uses sophisticated mathematical models to handle data in complex ways [21]. In general, a neural network is a piece of software designed to mimic the functions of the human brain, particularly pattern recognition and the transmission of input across several layers of artificial neural connections. DNNs are networks with an input layer, an output layer, and at least one hidden layer in between, according to many experts as in Fig. 10. The process of sorting and arranging that each layer accomplishes is known as "feature hierarchy" by some. Dealing with unlabeled or unstructured input is one of the main applications of highly powerful neural networks. These deep neural networks are also referred to by the term "deep learning," which refers to a particular type of machine learning in which tools utilizing AI-related components attempt to classify and arrange data in ways that go beyond conventional input/output protocols.

3.9.2 Multilayer Perceptron (MLP)

A multilayer perceptron is a feedforward artificial neural network that creates a collection of outputs from a set of inputs (MLP) [26]. There are several levels of input nodes in the directed graph that connects an MLP's input and output layers. The network is trained by MLP using backpropagation.

A directed graph, in which the signal only moves in one way across the nodes, is what distinguishes a multilayer perceptron from other neural networks. Every node has a nonlinear activation function aside from the input nodes. An MLP employs the supervised learning technique of backpropagation. The layer-based organization of neurons makes it a deep learning technique. It is commonly employed in the study of computational neuroscience, distributed parallel computing, and supervised learning problems. Examples of applications include speech recognition, image recognition, and machine translation [27].

4 Results and Discussion

The performance of the ML and NN algorithms is examined in this section which are DNN, SVM, MLP, KNN, RF, XG Boost, LGBM, and Decision tree. Figure 11 describes the formula for calculating the performance, which are used for

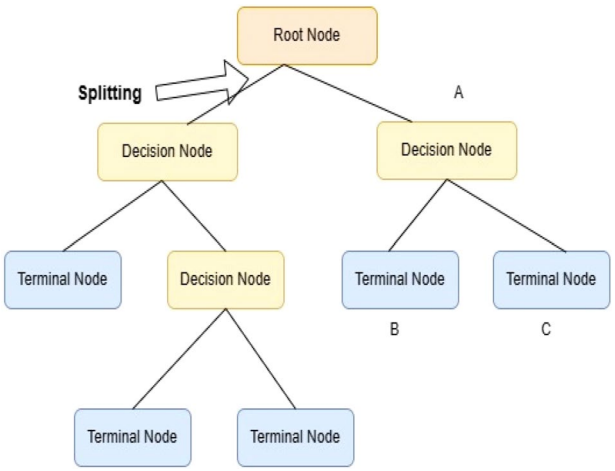


Fig. 9 Decision tree [22]

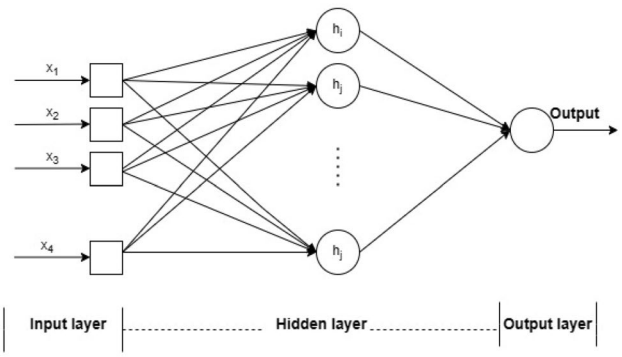


Fig. 10 DNN architecture

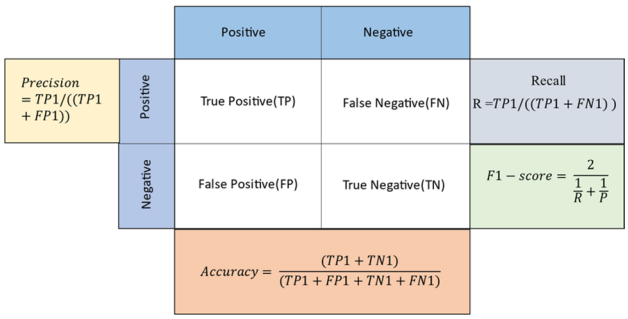


Fig. 11 Description of ML- and NN-based formula for calculation of performance parameters

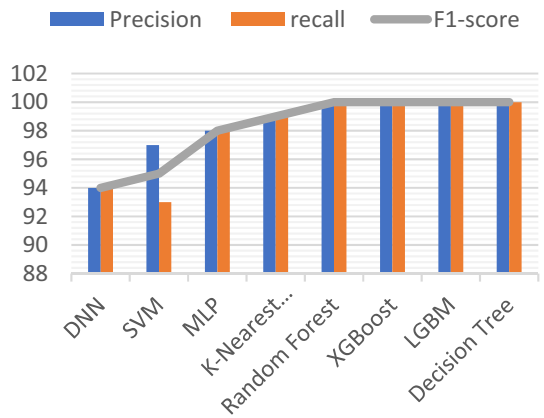


Fig. 12 Analysis of macro-average

the comparison of the algorithms. The different performance parameters are Accuracy, Precision, Recall, and *F*1 Score.

The comparison between Precision, Recall, and F1 Score is shown in Table 2 in terms of the macro-average, and Fig. 12 shows the analysis of it. For both scores to be equally important, the macro-average is the arithmetic mean between the F1 Scores of the two categories.

The weighted average of Precision, Recall, and *F*1 Score in Table 3 shows the comparison, and Fig. 13 shows the analysis of it. It is referred to as a weighted average when some quantities are more significant than others and do not contribute evenly to the outcome. When weight is added, it is a straightforward process to arrive at an average value between two or more quantities.

The comparison of positive and negative values in terms of support, examined in below Table 4. In the actual dataset, it is the sum of all entries for each class. It is the total of all the rows for each class.

Table 5 shows the accuracy comparison between the algorithms where the random forest, XG Boost, LGBM, and Decision tree getting the highest accuracy of 100 percent as compared to other ML and DL methods. And Fig. 14 describes the analysis of the algorithms in terms of accuracy.

Table 3 Macro-average comparison

Classifier	Precision	Recall	*F*1 Score
DNN	94	94	94
SVM	97	93	95
MLP	98	98	98
K-nearest neighbors	99	99	99
Random forest	100	100	100
XG Boost	100	100	100
LGBM	100	100	100
Decision tree	100	100	100

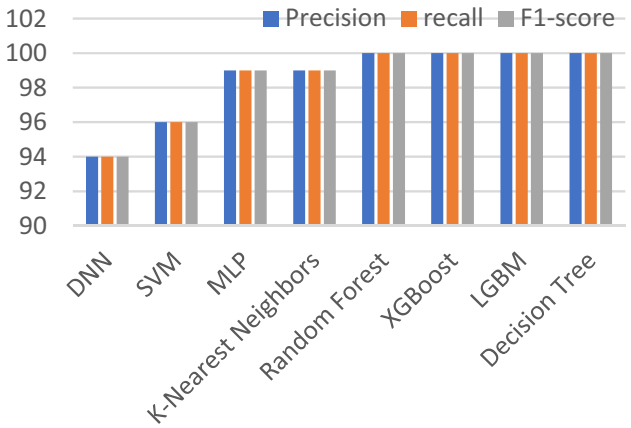


Fig. 13 Analysis of weighted average

Table 4 Weighted average comparison

Classifier	Precision	Recall	*F*1 Score
DNN	94	94	94
SVM	96	96	96
MLP	99	99	99
K-nearest neighbors	99	99	99
Random forest	100	100	100
XG Boost	100	100	100
LGBM	100	100	100
Decision tree	100	100	100

As RF is the extension of DT, here, DT itself giving the best accuracy because this algorithm is fit for the model and able to visualize the dataset, and RF is able to handle large dataset with high dimensionality which is going to enhance the accuracy and performance of the model.

XG Boost and LGBM both are efficient and easy to use algorithms which gives higher efficiency, lower memory usage, and better accuracy because XGB is having feature of handling missing values which allow to handle real-world data with missing values without requiring

Table 5 Comparison of positive and negative values in terms of support

Classifier	0	1
DNN	2971	3041
SVM	3006	1088
MLP	3016	1078
K-nearest neighbors	2994	3018
Random forest	2994	3018
XG Boost	2994	3018
LGBM	3007	1087
Decision tree	3007	1087

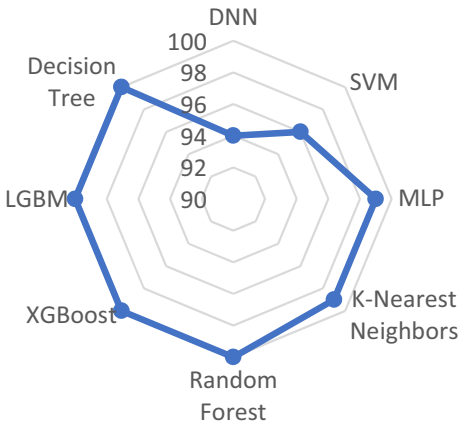


Fig. 14 Analysis of methods in terms of accuracy

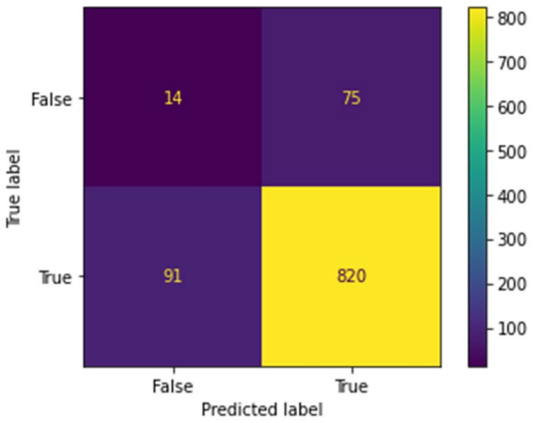


Fig. 15 Confusion matrix of Random forest

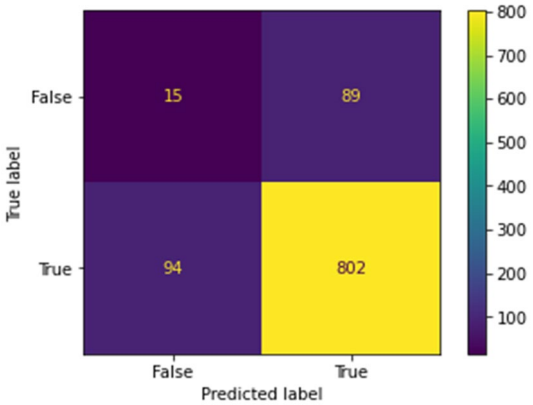


Fig. 16 Confusion matrix of XGB

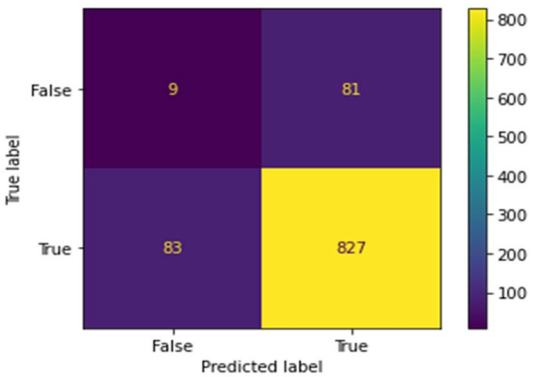


Fig. 17 Confusion matrix of KNN

significant preprocessing, and LGBM used different novel approaches like Gradient-based One Side Sampling and Exclusive Feature Bundling which comprise together to make the model work efficiently.

As confusion matrix examined the performance of the classification models, it is between the true positive, true negative, false positive, and false negative where Figs. 15, 16, 17, and 18 show all the values of the dataset and predicted it from the testing data by using the algorithms RF, XGB, KNN, and DT.

Comparison of the evaluation parameters such as Accuracy, Precision, and Recall, the algorithms used are shown in Table 6. As compared to the presented paper [6, 7, 9, 10, 12], proposed approaches are giving best results with 100 percent accuracy in RF, XGB, LGBM, DT. This shows the better performance of our approach to detect fraud in bank dataset. List of abbreviations and their definitions used are in Table 7.

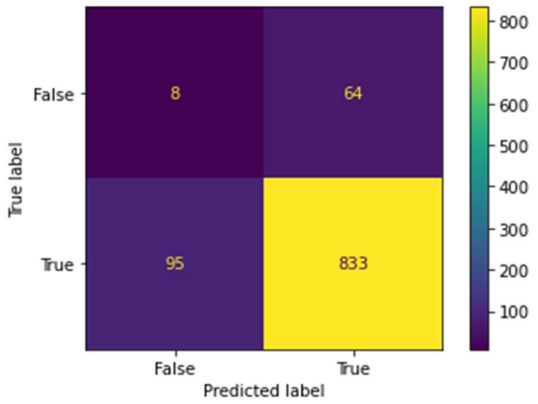


Fig. 18 Confusion matrix of Decision trees

Table 6 Analysis of classifier algorithms in terms of accuracy

Classifier	Accuracy
DNN	94
SVM	96
MLP	99
K-nearest neighbors	99
Random forest	100
XG Boost	100
LGBM	100
Decision tree	100

5 Conclusion

In contemporary times, financial frauds are augmenting at an agitated rate. The ability to avoid frauds relies on an accurate and effective method of detection. ML-based systems are better adapted for detecting fraud because they can identify thousands of patterns, unlike rule-based systems. So, more banks are implementing ML to detect fraud in the financial industry. However, a model that only uses supervised ML methods or algorithms would not be adequate to accurately detect fraud and offer crucial insights. Using a Kaggle dataset, various ML and NN methods are used in this paper to identify bank fraud. The dataset has a fraud percentage of 26.5683 percent and a non-fraud percentage of 73.43 percent. The paper compared ML and NN models, and some ML models are found to have the highest accuracy, where the highest accuracy of 100% is achieved by Random Forest, XG Boost, Decision tree, and LGBM models. However, the limitation of work is that ML and NN models can be costly in terms of computational resources, infrastructure, and skilled personnel required for model development and monitoring. Therefore, for future studies, it is suggested to propose using feature selection and extraction methods with all the different ML and NN methods to improve performance and enhance the time of the epochs in the proposed method.

Table 7 Comparison of results of proposed work with existing research papers

References	Method	Accuracy		Precision		Recall	
		Presented paper	Proposed approach	Presented paper	Proposed approach	Presented paper	Proposed approach
Varun Kumar et al. [7]	SVM	99.87	96	78.81	96	–	–
Khodabakhshi et al. [10]	KNN	98.5	99	32.68	99	59.62	99
Patil et al. [9]	RF	98.67	100	100	100	98	100
Hashemi et al. [13]	XGB	99.92	100	78.62	100	79.49	100
Hashemi et al. [13]	LGBM	99.91	100	75.34	100	79.90	100
Varun Kumar et al. [7]	DT	92.88	100	99.48	100	86.34	100

Author Contributions AV was involved in conceptualization, methodology, implementation, validation, and writing original draft. AT was responsible for resources, conceptualization, writing—reviewing and editing, and supervision. PKY and SP contributed to implementation.

Declarations

Conflict of interest The authors declare that they have no known competing financial interest or personal relationship that could have appeared to influence the work reported in this paper.

References

1. Lakshmi SVSS, Kavilla SD (2018) Machine learning for credit card fraud detection system. Int J Appl Eng Res 13(24):16819–16824
2. Sarma D, Alam W, Saha I, Alam MN, Alam MJ, Hossain S (2020) Bank fraud detection using community detection algorithm. In: 2020 second international conference on inventive research in computing applications (ICIRCA). p 642–646. IEEE
3. Hilal W, Gadsden SA, Yawney J (2022) Financial fraud: a review of anomaly detection techniques and recent advances. Expert Syst Appl 193:116429. https://doi.org/10.1016/j.eswa.2021.116429
4. Ranjan P, Santhosh K, Kumar A, Kumar S (2022) Fraud detection on bank payments using machine learning. In: 2022 International conference for advancement in technology (ICONAT). p 1–4. IEEE
5. Sadgali I, Sael N, Benabbou F (2019) Performance of machine learning techniques in the detection of financial frauds. Procedia computer science 148:45–54
6. Daliri S (2020) Using harmony search algorithm in neural networks to improve fraud detection in banking system. Comput Intell Neurosci. https://doi.org/10.1155/2020/6503459
7. Varun Kumar KS, Vijaya Kumar VG, Vijay Shankar A, Pratibha K (2020) Credit card fraud detection using machine learning algorithms. Int J Eng Res Technol (IJERT) 9(07):5–8
8. Dornadula VN, Geetha S (2019) Credit card fraud detection using machine learning algorithms. Procedia Comput Sci 165:631–641
9. Patil PS, Dharwadkar NV (2017) Analysis of banking data using machine learning. In: 2017 international conference on I-SMAC (IoT in social, mobile, analytics and cloud) (I-SMAC). p 876–881. IEEE
10. Khodabakhshi M and Fartash M (2016) Fraud detection in banking using knn (K-nearest neighbor) algorithm. In: International conf. on research in science and technology
11. Alghofaili Y, Albattah A, Rassam MA (2020) A financial fraud detection model based on LSTM deep learning technique. J Appl Secur Res 15(4):498–516
12. El Bouchti A, Chakroun A, Abbar H, Okar C (2017) Fraud detection in banking using deep reinforcement learning. In: 2017 seventh international conference on innovative computing technology (INTECH). p 58–63. IEEE
13. Hashemi SK, Mirtaheri SL, Greco S (2022) Fraud detection in banking data by machine learning techniques. IEEE Access 11:3034–3043
14. Thennakoon A, Bhagyani C, Premadasa S, Mihiranga S, Kuruwitaarachchi N (2019) Real-time credit card fraud detection using machine learning. In: 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence). p 488–493. IEEE
15. Gyamfi NK, and Abdulai JD (2018) Bank fraud detection using support vector machine. In: 2018 IEEE 9th annual information technology, electronics and mobile communication conference (IEMCON) p 37–41. IEEE
16. Xuan S, Liu G, Li Z, Zheng L, Wang S, Jiang C (2018) Random forest for credit card fraud detection. In: 2018 IEEE 15th international conference on networking, sensing and control (ICNSC). p 1–6. IEEE
17. Dheepa V, Dhanapal R (2012) Behavior based credit card fraud detection using support vector machines. ICTACT J Soft Comput 2(4):391–397
18. Fraud detection bank dataset 20K records binary (n.d.) Fraud Detection Bank Dataset 20K Records Binary|Kaggle. https://www.kaggle.com/datasets/volodymyrgavrysh/fraud-detection-bank-dataset-20k-records-binary
19. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357
20. Suthaharan S (2016) Support vector machine. Machine learning models and algorithms for big data classification. Integr Ser Inf Syst 36:1–12
21. Yu X, Li X, Dong Y, Zheng R (2020) A deep neural network algorithm for detecting credit card. In: 2020 international conference on big data, artificial intelligence and internet of things engineering (ICBAIE) p 181–183. IEEE
22. Machine Learning Random Forest Algorithm-Javatpoint. (n.d.). www.javatpoint.com. https://www.javatpoint.com/machine-learning-random-forest-algorithm
23. Yusnita MA, Paulraj MP, Yaacob S, Bakar SA, Saidatul A (2011). Malaysian English accents identification using LPC and formant analysis. In: 2011 IEEE international conference on control system, computing and engineering. p 472–476. IEEE
24. Priscilla CV, Prabha DP (2021) A two-phase feature selection technique using mutual information and XGB-RFE for credit card fraud detection. Int J Adv Technol Eng Explorer 8:1656–1668
25. Sahin Y, Bulkan S, Duman E (2013) A cost-sensitive decision tree approach for fraud detection. Expert Syst Appl 40(15):5916–5923
26. Mubarek AM, and Adali E (2017).Multilayer perceptron neural network technique for fraud detection. In: 2017 international conference on computer science and engineering (UBMK). p 383–387. IEEE
27. Zhang D, Wang H (2005) Disjoint directed quadrilaterals in a directed graph. J Graph Theor 50(2):91–104. https://doi.org/10.1002/jgt.20096