A Comparison of Data Sampling Techniques for Credit Card Fraud Detection

Abdulla Muaz[1], Manoj Jayabalan[2*], Vinesh Thiruchelvam[3]
School of Computing, Asia Pacific University of Technology and Innovation, Kuala Lumpur, Malaysia[1, 3]
Faculty of Engineering & Technology, Liverpool John Moores University, Liverpool, UK[2]

***Abstract*—Credit Card fraud is a tough reality that continues to constrain the financial sector and its detrimental effects are felt across the entire financial market. Criminals are continuously on the lookout for ingenious methods for such fraudulent activities and are a real threat to security. Therefore, there is a need for early detection of fraudulent activity to preserve customer trust and safeguard their business. A major challenge faced in designing fraud detection systems is dealing with the class imbalance issue in the data since genuine transactions outnumber the fraudulent transactions typically account less than 1% of the total transactions. This is an important area of study as the positive case (fraudulent case) is hard to distinguish and becomes even harder with the inflow of data where the representation of such cases even decreases further. This study trained four predictive models, Artificial Neural Network (ANN), Gradient Boosting Machine (GBM) and Random Forest (RF) on different sampling methods. Random Under Sampling (RUS), Synthetic Minority Over-sampling Technique (SMOTE), Density-Based Synthetic Minority Over-Sampling Technique (DBSMOTE) and SMOTE combined with Edited Nearest Neighbour (SMOTEENN) was used for all models. The findings of this study indicate promising results with SMOTE based sampling techniques. The best recall score obtained was with SMOTE sampling strategy by DRF classifier at 0.81. The precision score for this classifier was observed to be 0.86. Stacked Ensemble was trained for all the sampled datasets and found to have the best average performance at 0.78. The Stacked Ensemble model has shown promise in the detection of fraudulent transactions across most of the sampling strategies.**

Keywords*—Data imbalance; credit card fraud; sampling techniques

I. Introduction

Transactions using credit cards have become an important aspect of our daily lives. Purchase of goods and services are no longer a chore that requires physical activity, rather it is initiated with a touch of a button on our smartphone or personal computers. The authorization of transactions is rigorous and secure although such conveniences are brought about by compromising the proof of identity checks which require personal identification documents, authorized signature and physical presence. The basis of the identity proof in such transactions is the information on the card along with digital identification tied to the cardholder.

The conveniences brought about by digital transactions makes it a target for fraudsters who employ elegant tactics for theft and illicit use. Credit card fraud is generally an unauthorized movement by an individual who is not authorized to perform the said account operation. It can be also classified where a person transacts with a card, without the explicit permission of the owner of the cardholder or card issuer [1]. The most common form of credit card frauds are stolen or lost cards, fraudulent applications, counterfeit card fraud, non-receipt fraud, card not present (CNP) and account takeovers [2]. According to the European Central Bank [3], on the composition of fraudulent credit card transactions for the year 2016, 73% of the fraudulent transactions were a result of CNP, where payments are made via the internet or telephone.

Billions worth of transactions is lost worldwide every year due to fraudulent credit card transactions. According to the Nelson Report on global payment systems, the amount of losses due to credit card fraud is $22.8 billion, and this indicated a 4.4 percent increase from the year 2015. It was also highlighted that 38.6% of this global credit card frauds are accounted for from frauds in the United States. The Nelson Report also projects that the credit card fraud losses are expected to grow by over $10 billion over the next three years [4].

With the increasing amounts of loss due to such illicit activities costing institutions and individual's huge amounts of money, tackling this issue has become a priority over the past decade and various studies have been conducted to address this problem. Financial institutions are constantly on the verge of upgrading their fraud detection systems. Association of Certified Fraud Examiners (ACFE), suggests that proactive data analysis and continuous monitoring of real-time activity as the key for minimizing and preventing fraudulent credit card transactions [2].

Financial institutions and credit card issuers collect and store a vast amount of transaction data. Every credit card transaction composes key attributes such as the card identifier, transaction date, recipient and amount of transaction, which are stored in the databases. Fraud Detection Systems (FDS) implements various layers of validation to flag potential frauds using such datasets.

Although, machine learning and predictive analytics might not answer the question of exactly the type of fraud that may occur, it has the potential to flag suspicious activities and identify potential frauds with the help of a trained model, on historical data combined with expert analysis. Such systems can equip institutions with proactive insights into the future, to enable them to better cope and mitigate fraudulent transactions.

*Corresponding Author

Real-world implementation of FDS cannot reliably check all transactions as it is constrained by the human labour required to validate the sheer number of alerts raised the by system. It mainly relies on fraud investigators who are used as a confirmatory layer whereby flagged fraudulent activities (alerts) are verified and validated by the designated investigators.

Transactions which are then reported by the customer during this window are flagged or labelled as fraudulent and the unreported transactions labelled genuine transactions. To summarize, there are two ways FDS samples the data; immediate feedback samples (transactions with investigator feedback) and delayed samples (transactions whose fate are known only after a set reaction-time period). This is a crucial distinction to be considered when implementing an accurate FDS as every transaction is not immediately labelled either fraud or genuine [5].

Fraudulent labels in the dataset can, therefore, be safely assumed to be verified and validated by the investigators. However, there are other challenges in designing accurate machine learning techniques for such data. Firstly, the non-stationary data distribution (fraudulent and genuine transactions share similar profile). Often at times fraudsters mimic the cardholders spending behaviours, which makes the profiles of the fraudster and cardholder very similar in such cases and different in the other cases. This changing dynamics between genuine and fraudster profiles also known as concept drift, makes it particularly challenging for machine learning algorithms to accurately predict fraudulent transactions [5]–[9].

Secondly, the skewness or the class imbalance in these datasets poses a considerable challenge in building accurate machine learning models. This is the case for a variety of real-world applications where the true class or the interested observations tend to be a fraction of the total cases. Credit card fraud detection has this distinctive characteristic as majority of the transactions are genuine while the concerned cases (fraud activity) has very few transactions. This is known as the class imbalance and it is significant because the positive class is often the rare class and predicting this class becomes harder as the number of false class keeps on increasing. Machine learning models typically work on the assumption of an equal class balance and equal cost of misclassification, therefore adequate measures have to be taken in order to address this issue of class imbalance [5], [6], [10]–[12].

Detection of credit card fraud is classified as a cost-sensitive problem, where there is an associated cost incurred for incorrectly classifying a genuine transaction as fraudulent and incorrectly classifying fraudulent transaction as genuine. In the absence or no occurrence of fraud, there is no associated administrative costs incurred by the financial institution. However, failure to detect the fraud is a loss of the particular transaction amount. It is thus, an important proposition to incorporate in to the FDS, particularly in the development of models on class imbalanced datasets [9].

A. Contributions

The research contributes both theoretically and practically. The significance in terms of both means are summarized as follows.

This paper provides an overview of the most recent literature on credit card fraud detection strategies which focused on the newest Machine Learning techniques while addressing the major challenges faced by the traditional FDS. The research offers an up to date perspective on trends in the credit card fraud detection domain, model evaluation metrics that offer the best results and outlines limitations of existing FDS. Researchers can find this paper helpful as it is a good starting point, to kickstart a research on implementing machine learning techniques for credit card fraud detection.

The practical contributions of this research are to provide a sound and realistic model that articulates the classification problem pertaining to the domain of credit card fraud detection. Sampling strategies proposed to be implemented in this paper shall enable researchers to promptly use and adopt this technique which best serves their research goal. Various sampling techniques shall be implemented to generate and train different machine learning models, and conclusively summarize experimental results of the built models using a multitude of relevant model evaluation metrics.

The paper addresses key challenges faced in building machine learning models for FDS, and experimentally prove strategies to mitigate or minimize such challenges. Therefore, it is an invaluable contribution to the financial sector, with the contribution of a predictive model able to accurately predict fraudulent credit card transactions.

II. Related Works

This section synthesizes the contents and ideas in the existing studies and encompasses key subject matters regarding the domain of credit card fraud detection. These subjects include, machine learning techniques, sampling techniques, visual data analytics, feature engineering and model evaluation metrics.

A. Machine Learning Techniques

Credit card fraud detection studies on the use of predictive analytics have shown that researchers adopted various methods such as Artificial Neural Networks, k-Nearest Neighbour (kNN), Logistic Regression (LR), AdaBoost, Naïve Bayes (NB) and many more [6], [13]–[17].

In [6], used NB, kNN and LR on the European card holders dataset. This dataset contains anonymized transaction data of European credit card holders which were collected for a period of two days and contains 284,807 samples. The results of this study conclude that kNN produced the best results for accuracy, sensitivity and specificity. Although the authors argue that this potentially could be caused by the generation of synthetic samples using Synthetic Minority Over Sampling which uses KNN.

One study proposed an improved method of sampling to produce a better performance, which referred to as Moving to Adaptive Samples in Imbalanced (MASI) dataset. The study implemented Random Forest (RF), Support Vector Machines (SVM) and C 5.0 Decision Tree algorithm to conclude that SVM produced the best results [18].

In another study using the same dataset implemented LR, KNN, Linear SVM, RBFSVM decision trees, RF, and NB algorithms [17]. Although, both [18] and [6] implemented the same models, the sample size was 350, which was a result of random under sampling. The highest sensitivity score achieved for the study was SVM with a score of 94%.

Random Forest is implemented by majority of the researchers [6], [18], [19] with varying degree of results. In [20] experimented on a weight assignment approach to the RF, using out-of-bag error to compute the weights while other researchers typically opted for using various sampling techniques.

Deep Learning techniques such as ANN, Recurrent Neural Networks (RNN), Long Short-term Memory (LSTM) and Gated Recurrent Units (GRU) was implemented [21]. The LSTM and GRU outperformed the traditional ANNs, however the shortcomings are that the training was not conducted to achieve optimal model stability. It was cited that "performance improved whenever network size was increased", and future recommendation was made to identify an optimal stopping point.

Restricted Boltzmann Machines (RBM) was another topology of Deep Learning which was implemented by past researchers [22]. The researcher used a novel approach with the use of unsupervised machine learning techniques (Stacked Auto Encoder) to identify optimum weights, which was then applied to a supervised machine learning model RBM achieved an accuracy of 91.5%.

B. Issues of Class Imbalance

Numerous studies have shown different approaches to deal with this issue in the context of implementing accurate prediction model which are aimed at improving the detection rate of fraudulent transactions [6], [18], [23], [24].

The most common method implemented in the existing studies to handle the problem at data level, where the data is subjected to various sampling techniques. Random under sampling (RUS) is implemented where the majority class instances are removed [10], [17] or random oversampling (ROS) is used where minority class instances are added by replicating training samples with the same class representation. Some advanced methods were also used to oversample with techniques such as Synthetic Minority Oversampling Technique (SMOTE) which creates new synthetic instances of the minority class using kNN. Synthetic instances which are created using this technique have been shown to perform better, than simply using random oversampling or replication of instances [1], [25], [26]. Alternative methods to the SMOTE, was implemented by [23] and [18]. The drawbacks of the SMOTE sampling technique such as loss of potential information and potential for model overfitting for the synthetic samples.

In [18] proposed an improved method of sampling using an approach which the author refers to as Moving to Adaptive Samples (MASI) in Imbalanced dataset and obtained comparatively better performance against other sampling techniques such RUS, ROS and SMOTE. While SMOTE, resampling generates new instances and increase the data size prior to the implementation of the classifier, MASI adaptively creates synthetic samples which are created based on the density distribution of original data and up-samples the minority class by changing class labels. The researchers indicate this reduces the bias of the classifier as it moves the samples in minor class closer to the decision boundary.

Alternative to tackling the imbalance issue on the dataset, ensemble learning handles class imbalance issue at the algorithmic level. Ensemble methods typically include bagging and boosting that primarily aims to lower the variance in the data by using multiple classifiers. In bagging method, multiple weak classifiers are trained on different subsets of the majority class and minority class before combined final classifier is built using all the weak classifiers. AdaBoost employs similar strategy and can be implemented for many classification problems and it eliminates the need for exploring an optimum class balance ratio while alleviating the information loss which can be caused by RUS, and overfitting issue caused by ROS and SMOTE methods [25], [27].

One study implemented a new oversampling strategy which combined k-means clustering with genetic algorithm to oversample the minority class. The researchers propose this solution as opposed to SMOTE and other sampling strategies highlighting the potential for information loss and overfitting [23].

C. Feature Engineering

Fraudsters constantly change their behaviours and implement new ways to commit frauds, which renders traditional expert rules. Machine learning methods are also prone to this type of problems, however adoption of new strategies can assist to counter. Feature engineering is a method which can be used extensively to counter this effect, whereby new features are created based on the card holder's behaviour over time. These new features aids the machine learning models to distinguish patterns from the normal card holder behaviour [9], [28].

Feature engineering is proven to be an important aspect of predictive analytics for detection of credit card frauds. Financial institutions obtain and store large amounts of data related to transactions such as transaction amount, account holder details, time of transaction and more. While these collected data serve as good predictors in a classifier setting, it has the potential to be enriched with new information such as card holder spending habits in a set time frame, average amount spent in different geographical areas or product and service types. For example, a card holder can be profiled by his spending habit at home, but this may differ completely with his spending habit on a vacation in India. Such features could potentially be able to discover patterns and solve the concept-drift problem where card holder and fraudster behaviour is distinguished with the help of new data dimensions [9], [23].

It is also noteworthy, that single transaction information is typically insufficient for the purpose, rather aggregate measures which combines to form new features are ideal [9].

D. Evaluation Metrics

Evaluation metrics are an important aspect to understand the performance of the machine learning models. Detection of credit card fraud is classified as a cost-sensitive problem, where there is an associated cost incurred for incorrectly classifying a genuine transaction as fraudulent and incorrectly classifying fraudulent transaction as genuine [9]. As such, the choice of evaluation metric must be carefully chosen and shall be relevant in terms of the objective of the study and available data.

The existing studies have, adopted various evaluation metrics for binary classifiers such as Area Under the Curve (AUC), Sensitivity (also referred to as Recall), Precision and F1 score [10], [18], [21], [22] [14], [16], [21].

Machine learning models work on the assumption of equal class distribution and equal cost of misclassification. Using accuracy metric for evaluating a model is not suitable for datasets with class imbalance issue as it would bias the model towards majority class since the accuracy metric calculates the total of correct predictions [20], [10].

Area Under the Curve (AUC), is a measure of the probability that the model or classifier will choose a random positive instance higher than a random negative instance. AUC is a metric; many researchers have adopted [22], [26], [29] and gives a good indication of the overall predictive performance of the model across various probability threshold settings and is very well suited for the class imbalanced modelling.

Precision is the percentage of true positives among all positive predictions, while recall indicates the total correctly predicted positive classes over the total predictions for both correctly predicted positive class and falsely predicted positive class. F1 measure is the mean of sensitivity and precision. Out of all these metrics used in this study, the most useful metric which was able to give a clear indication of the best classifier was sensitivity or recall metric.

III. Methods and Techniques

This section briefs the research methodology that will be adopted to achieve the objectives of this research. The section includes an overview and key processes involved in the methodology, such as dataset summary, sampling techniques, and machine learning algorithms.

The dataset collected for this study is secondary data consist of transaction data of European credit card holders which were collected for a period of two days and contains 284,807 observations with 31 variables out of which 28 variables are anonymized using principal component [30].

The three non-anonymised variables are transaction time, amount and the class label (fraudulent or not fraudulent transaction). The class label indicates ‘0’ for non-fraudulent transaction and ‘1’ for fraudulent transaction. The dataset is highly imbalanced as the percentage of fraud instances accounts to 0.172%. The dataset does not contain any missing values and outliers, therefore pre-processing techniques on the dataset shall not be required. Table I describe the features in dataset. Features V1 to V28 are aggregated to single description for ease of reading.

TABLE I. Dataset Description

Features	Description
Time	Number of seconds elapsed between this transaction and the first transaction in the dataset
V1,V2,V3,V4,V5,V6,V7,V8,V9,V10,V11,V12, V13,V14,V15,V16,V17, V18,V19,V20,V21, V22,V23,V24,V25,V26, V27, V28	Result of a PCA Dimensionality reduction to protect user identities and sensitive features
Amount	Transaction amount
Class	1 for fraudulent transactions, 0 otherwise

A. Sampling Techniques

A reliable FDS with detecting all frauds is vital as well as reducing false flags where genuine transactions are misclassified as fraudulent. The associated costs are much higher, when a fraudulent transaction pass through the system undetected (False Negative). However, it is also an important issue when false flags are raised for non-fraud transactions (False Positive), which hurts the customer sentiment as well as an added cost of allocating investigative resources needlessly. Maximizing recall score, is thus significantly important as high recall scores indicate a higher ability for the classifier to detect True Positives (Frauds). Precision scores is also important as the FDS shall avoid or minimize misclassifying genuine transactions as frauds. Therefore, various sampling strategies were adopted, and four different classifiers implemented to conclusively deduce the best and most effective sampling strategies and classifiers best suited for the dataset.

1) Random Under Sampling (RUS): Random Under Sampling is one of the most commonly used sampling techniques, where the majority class is down sampled or reduced to the same number of minority class by randomly removing instances of the majority class. The major problem with RUS is that it is randomly removing data which leads to potential loss of important information which may have been captured.

2) Synthetic Minority Oversampling Technique (SMOTE): SMOTE create synthetic instances of the minority class. These data points are created by assessing the nearest neighbours for each of the minority sample and creating new synthetic instances in the feature space until the minority class is balanced to the given ratio.

3) Density-Based Synthetic Minority Oversampling Technique (DBSMOTE): DBSMOTE algorithm relies on a clustering algorithm called Density-Based Spatial Clustering of Applications with Noise (DBSCAN), which is widely used clustering algorithm used for data mining and machine learning applications. DBSCAN works by grouping together a set of data points based on how close together the points are packed in terms of a distance measurement such as the

Euclidean distance and a given number of minimum points to operate on the bi-dimensional space. DBSMOTE essentially implements the DBSCAN clustering algorithm to form a cluster of the minority class, which is then used to up-sample the minority class. The minimum samples specify the number of data points required to form the dense region.

4) Synthetic Minority Oversampling Technique with Edited Nearest Neighbor (SMOTEENN): SMOTEEN is another variant of SMOTE which is basically a combination of SMOTE and Edited Nearest Neighbour (ENN). The ENN is an effective method which is used to remove noise from the dataset. For any given data point of either class, ENN removes the data point which differs by at least half of the given k-Nearest Neighbour.

B. Machine Learning Techniques

This section, details and justifies the different benchmark machine learning models which are proposed in FDS, highlights the strengths of the machine learning models used, and lists out the evaluation metrics to be used focusing on the class imbalance nature of the dataset.

The machine learning algorithms are Gradient Boosting [25], [27], Stacked Ensemble [31], Artificial Neural Network – Multilayer Perceptron [21], [32] and Random Forest [13], [20].

The model shall primarily be evaluated with Recall score as it is more important to the FDS to accurately detecting the fraudulent transactions (increasing TPR). The precision although not as significant as the recall score, still has associated costs for an FDS and thus the second metric to consider shall be the precision score.

1) Stacked ensemble: Stacked Ensemble model have shown promising improvements in terms of classification accuracy when combined with diverse set of classifiers. In a study by [31], Stacked Ensemble was used for an imbalanced dataset and proved to have gained maximum performance among the other models. Modern applications of machine learning quite often must deal with imbalanced classification as is the case with this study. The current ensemble techniques offer a modification to the traditional ensemble models to allow for maximum performance on imbalanced learning. The Stacked Ensemble model allows for customization of parameters that are designed specifically to handle class imbalance issues [33]. The SE is a combined model of chosen base models of and uses General Linear Model (GLM) as a default meta learner to enhance the model performance.

2) Gradient boosting machine: Gradient Boosting Machine can be used for either regression or classification models. It is an ensemble learning method which operates on the concept of Boosting where weak learners are built gradually to allow for maximum prediction accuracy with each iteration. Unlike Random Forests which use Bagging, and trees are built independent of one another, Boosting aims to build trees which are built based on the results of previously built trees. Boosting although improves accuracy it is slower and has reduced interpretability than other traditional models.

This study shall use gradient boosting model to allow for a diverse set of classifiers where four different categories of learning is considered, namely, Bagging, Boosting, Deep Learning and Super Learning and gradient Boosting Machine [33].

3) Random forest: Random Forest is essentially and ensemble model consisting of many decision trees all of which are made from the same input dataset. The high prediction accuracy of random forests is due to the fact that a combined output is obtained in random forest by comparing outputs from all decision trees. Essentially multiple training subsets are built from the dataset and a decision tree is constructed for each of these training subsets. With each tree contributing towards voting and eventually majority of the votes determine the final class. This technique is known as random split and the trees are known as random trees.

For the purpose of this study, Random Forest shall be chosen to build a predictive classifier model, as this model gives the best classification accuracy and also due to the high speed of classification, interpretation ability of the knowledge or classifications, and model parameter handlining as indicated by [34].

4) Artificial neural network: Multilayer Perceptron (MLP) is a technique which is trained by the backpropagation algorithm. Essentially a MLP neural network composed of three layers namely; input layer, output layer and many hidden layers. The architecture is a densely connected network where every neuron in a layer is connected to neurons in prior and next layers. The other feature of this network is that there is no activation function in the input layer, but every neuron in the hidden and output layer has an activation function.

The initialization of weights is a random process in the MLP, however the network trains by working out the difference between the computed output and the actual output and adjusts the weights iteratively to this cause of minimizing the residual.

IV. Experimental Results and Discussions

This section briefs the model development stage and details the techniques implemented for this study. The sampling methods including RUS, SMOTE, DBSMOTE and SMOTEENN are implemented, since the dataset is highly imbalanced with 0.17% of positive instances. These methods are based on previous research conducted on the domain and is selected to offer diversity in terms of adopted sampling algorithm and attempts to find out which sampling strategy works best for the given dataset. The aim is also to understand in terms of the strengths of various classifiers and their ability to tackle each sampling strategy.

The class distribution after the dataset was split in to 70% for training set and 30% for holdout set using stratified random sampling. The holdout set contains 148 samples of the fraudulent transactions and will be used to evaluate the performance of all models to maintain consistency in scoring and model benchmarking. A separate hold out set is also the best strategy to adopt to avoid data leaks, which can be a

problematic and frequently occurs, while using cross validation along with oversampling. The training set contains 199,020 non-fraud and 344 fraudulent transactions. This set will be used for both data over sampling using SMOTE based techniques, as well as under sampling using RUS.

The Table II shows a summary of class counts after implementing sampling techniques on the original training dataset. In all the cases the final class counts are equal other than the SMOTEEN technique with unequal class counts. This is due to removal of noise using ENN.

TABLE II. SUMMARY OF CLASSES AFTER SAMPLING

Sampling Strategy	Fraudulent	Non-fraud	Total
RUS	344	344	688
SMOTE	199,020	199,020	398,040
DBSMOTE	199,020	199,020	398,040
SMOTEENN	195,374	190,186	385,560

A. Comparision of Sampling Techniques over Different Classifers

The training dataset was used to produce four different sampled datasets which were used to train each classifier. Unsampled dataset was used as a baseline for each of the classifier. The evaluation metrics used are F1 score, Precision and Recall.

Distributed Random Forest (DRF), Artificial Neural Network (ANN), Gradient Boosting Machine (GBM), and Stacked Ensemble (SE) are the four classifiers which have been trained on the four different sampling strategies (RUS, SMOTE, DBSMOTE, SMOTEENN). The results provided in this section includes classifier specific results for all the employed sampling strategies. Each classifier is concluded with an overall summary of key evaluation metrics F1 score, Precision and Recall score. These metrics, along with the confusion matrix facilitate to understand how the classifier performed with each of the sampling strategies.

1) Artifical Neural Network (ANN): The ANN architecture in this case was set at 30 in the input layer and 200 neurons in a single hidden layer followed by two layers in the output layer. The activation function used was Rectified Linear Unit (ReLU). The model reached optimal performance at 61 epochs with each epoch iterating over the training dataset. A drop out of 40% was used in the hidden layer so that the model automatically drops the neurons in the hidden layer. The learning rate used for the model was set at 0.005.

The highest F1 score is at 0.8116 on unsampled dataset, which is ideal in the case where precision and recall are of equal importance or significance as the F1 measure is a harmonic mean between the two metrics. However, in the case of FDS, recall is of much more importance then precision. The sampling method, RUS had the highest recall of 0.8311, and the lowest precision score among all the sampled datasets. ANN with unsampled data produced the highest f1 score of 0.8116, although its recall is lower than the second highest recall for ANN with SMOTE at 0.7635 and significantly better precision of 0.8370. Therefore, a better model for ANN is with SMOTE sampling.

2) Distributed Random Forest (DRF): Number of trees was set to 43 with a maximum tree depth of 20. The low tree depth helps lower model complexity while avoiding overfitting. The min rows parameter set as 5 specifies that a minimum of 5 observations is used for each leaf. The sample rate specifies the rate for row sampling which was set at 63%. Column sample rate was set to 0.8, which takes in 80% of columns to construct an individual tree. Lowering the column sample rate will aid in producing diverse trees, which are able to regularize well.

The highest F1 score was produced by the unsampled dataset for the DRF, which was influenced by the highest precision provided at 0.9286. Recall as we consider as the more important and significant metric is at the highest for SMOTE sampling at 0.8176 with a reasonable precision of 0.86. SMOTE sampling is, therefore, the best model for DRF considering the high recall score. It can also be noted that SMOTEENN sampling produced the second-best recall score for DRF classifier. SMOTEENN technique performs data reduction or noise removal using Edited Nearest Neighbour technique which removes any sample which is misclassified by its three nearest neighbours. It is proven with this result, that the noise removal is not very effective as it produced a lower recall score than the original SMOTE sampling.

3) Gradient Boosting Machine (GBM): The number of trees was set to 116 with a maximum tree depth of 15. This allows for reduced model complexity and prevents model from overfitting. Minimum rows to sample for the creation of each tree was set to 100 and column sampling rate set at 0.8, which means that 80% of the columns will be used for each tree.

It is observed that the highest recall score was produced by two sampling methods SMOTE, and SMOTEENN at 0.81. In this case, where two classifiers produce similar recall score, F1 score could be used as a deciding factor since it reflects the model with the best precision. Reducing false flags (False Positive) is an important aspect of an FDS, and thus the model with the highest recall and precision is preferred. Therefore, for the GBM classifier the best model is using SMOTE sampling which resulted in 0.81 recall score and 0.90 precision score. The model with the highest F1 score (DBSMOTE) at 0.86 cannot be considered the best model as it has lower recall score at 0.79 compared to the previously mentioned models, although precision is at the highest at 0.94.

4) Stacked Ensemble (SE): The Stacked Ensemble has very little parameters to define. The SE is a combined model of all trained models (30 models), using a General Linear Model (GLM) as a meta learner to enhance the model performance. The meta learner folds was set to 5 to create a 5-fold cross validated model training with stratified sampling.

Stacked Ensemble model is a Super Learner based on the combinations of ANN, GBM and DRF. The Random Under sampling (RUS) method scores the lowest for the key metric at

0.68 as well as offered the lowest precision score. Highest observed recall score was for SMOTEENN with a combination of SMOTE oversampling and noise removal is using the Edited Nearest Neighbour (ENN) technique. Since this model also offers a reasonable precision score of 0.85 it can be considered as the best sampling strategy for the Stacked Ensemble. Unsampled dataset offered the highest precision score of 0.94 as a result of less noise since it is based on 100% of original data and no synthetic samples were introduced.

B. Summary

The results from all classifiers for each of the sampling methods employed were consolidated based on the performance metrics. The key metric for the domain of FDS are recall which is of the highest priority while also addressing minimal False Positives (FP); i.e.; higher precision. To this end, the primary metric which will be considered is the recall as it is the key metric which is indicative of the total True Positives (fraud cases) detected while minimizing False Negatives (fraudulent transactions classified as non-fraudulent).

The evaluation results were assessed from two perspectives; i) Optimal sampling strategy, ii) Optimal classifier for the domain. The Table III summarizes how various sampling strategies performed comparatively.

TABLE III. PERFORMANCE COMPARISON ACROSS SAMPLING TECHNIQUES

Sampling	Model	F1-score	Recall	Precision
UNSAMPLED	ANN	0.8116	0.7568	0.8750
	DRF	**0.8540**	**0.7905**	0.9286
	GBM	0.8284	0.7500	0.9250
	SE	0.8433	0.7635	**0.9417**
RUS	ANN	0.4184	**0.8311**	0.2795
	DRF	**0.7653**	0.7162	0.8217
	GBM	0.7352	0.6284	**0.8857**
	SE	0.7566	0.6824	0.8487
SMOTE	ANN	0.7986	0.7635	0.8370
	DRF	0.8403	**0.8176**	0.8643
	GBM	**0.8541**	0.8108	**0.9023**
	SE	0.8459	0.7973	0.9008
DBSMOTE	ANN	0.8029	0.7568	0.8550
	DRF	0.8467	0.7838	0.9206
	GBM	**0.8603**	**0.7905**	**0.9435**
	SE	0.8509	**0.7905**	0.9213
SMOTEENN	ANN	0.7985	0.6959	**0.9364**
	DRF	0.8351	0.8041	0.8686
	GBM	**0.8451**	**0.8108**	0.8824
	SE	0.8305	**0.8108**	0.8511

The key metrics recall score is considered as a first step for identifying the best sampling strategy. RUS has the highest observed recall score of 0.83 with ANN classifier. However, this was not chosen to be the best model since it offered very little precision of 0.27. This means that while most of the fraudulent transactions are detected by the system it also falsely flagged several genuine transactions as fraudulent. Fraud Detection System is mostly concerned with increasing True Positives it must also consider to be precise in this detection by reducing the number of False Positive.

The second highest recall score was then considered with SMOTE sampling strategy by DRF classifier at 0.81. Precision score for this classifier is observed to be 0.86, which is significantly better than the RUS by ANN. Therefore, SMOTE method can be considered a better sampling strategy to adopt. It is also observed that SMOTE with GBM classifier also offers a high recall which was the third highest recorded at 0.81 while offering even higher precision then the SMOTE with DRF at 0.90.

SMOTEENN sampling is another technique which offered promising results and performed consistently with all classifiers except for ANN. The recall scores for most of the models been at 0.81 while yielding a good precision score above 0.85 in all the cases.

Assessing the average performance of the sampling strategy across various classifiers gives an indication of the best overall sampling strategy to adopt. In a diverse classifier domain such as FDS the average performance of the sampling strategy is very much indicative of its generalizability in terms of adopting well for other datasets in the field. Adopting no sampling strategy resulted in the worst average recall scores while SMOTEENN sampling strategy offered the best average recall score at 0.79. SMOTE and DBSMOTE have the same average score of 0.78, although SMOTE produced the best classifier. The average score considerably dropped for SMOTE due to a very low recall of 0.76 with ANN.

V. CONCLUSION

Detection of credit card fraud is classified as a cost-sensitive problem, where there is an associated cost incurred for incorrectly classifying a genuine transaction as fraudulent and incorrectly classifying fraudulent transaction as genuine. In the absence or no occurrence of fraud, there is no associated administrative costs incurred by the financial institution. However, failure to detect the fraud is a loss of the particular transaction amount. It is thus, an important proposition to incorporate in to the FDS, particularly in the development of models on class imbalanced datasets. There is an associated cost with False Positives, where genuine transactions are flagged as fraud. However, the cost associated with the inability to identify a fraudulent transaction can be immense in contrast. Therefore, recall score was used as key metric as the target of the FDS is to maximize the True Positive Rate.

A base model was implemented using an unsampled dataset, followed by the implementation of four different sampling strategies. Four different classifiers including a Super learner (Stacked Ensemble) was used for each of the sampled datasets to train the models. Distributed Random Forest (DRF),

Artificial Neural Network (ANN), Gradient Boosting Machine (GBM) and Stacked Ensemble (SE) are the four classifiers which have been trained on the four different sampling strategies (RUS, SMOTE, DBSMOTE, SMOTEENN). Each classifier is evaluated based on the overall summary of key evaluation metrics F1 score, Precision and Recall score.

The findings of this study indicate promising results with SMOTE based sampling techniques. The best recall score obtained was with SMOTE sampling strategy by DRF classifier at 0.81. Precision score for this classifier was observed to be 0.86. Therefore, SMOTE method can be considered a better sampling strategy to adopt.

Stacked Ensemble was trained for all the sampled datasets and found to have the best average performance at 0.78 with the second-best average for GBM classifier. ANN suffered with the worst recall score, which may be due to the high level of noise generated by the synthetic samples. The Stacked Ensemble model has shown promise in the detection of fraudulent transactions across majority of the sampling strategies.

A. Future Recommendation

Although the study was conducted to address the major problems in the domain of predicting fraudulent transactions, the limitations of the study with respect to time and resources contributed to selection of limited number of sampling strategies. Several other sampling strategies may be considered as an avenue for further research to improve the classifier performance.

Although un-supervised machine learning was not covered within the scope of this study it is still a promising area to be explored. This study may further be improved with the implementation of semi-supervised or un-supervised learning techniques such as one-SVM, k-means clustering and Isolation Forests.

Research can also be further expanded in identifying optimum thresholds for identifying the cut-off points to maximize the Recall score while finding the right balance between Precision and Recall could also yield potentially good results.

REFERENCES

[1] S. Manlangit, S. Azam, B. Shanmugam, K. Kannoorpatti, M. Jonkman, and A. Balasubramaniam, “An Efficient Method for Detecting Fraudulent Transactions Using Classification Algorithms on an Anonymized Credit Card Data Set,” in Intelligent Systems Design and Applications, 2017.

[2] K. T. Hafiz, S. Aghili, and P. Zavarsky, “The use of predictive analytics technology to detect credit card fraud in Canada,” Iber. Conf. Inf. Syst. Technol. Cist., vol. 2016-July, 2016.

[3] European Central Bank, “Fifth report on card fraud, September 2018,” 2018.

[4] AP NEWS, “SmartMetric Reports Worldwide Payment Card Fraud Losses Reach a Staggering $24.26 Billion While the USA Accounts for 38.6% of Global Card Fraud Losses,” BusinessWire, New York, Jan-2019.

[5] A. Dal Pozzolo, G. Boracchi, O. Caelen, C. Alippi, and G. Bontempi, “Credit card fraud detection and concept-drift adaptation with delayed supervised information,” Proc. Int. Jt. Conf. Neural Networks, vol. 2015-Septe, 2015.

[6] J. O. Awoyemi, A. O. Adetunmbi, and S. A. Oluwadare, “Credit card fraud detection using machine learning techniques: A comparative analysis,” Proc. IEEE Int. Conf. Comput. Netw. Informatics, ICCNI 2017, vol. 2017-Janua, pp. 1–9, 2017.

[7] A. Dal Pozzolo, O. Caelen, Y. A. Le Borgne, S. Waterschoot, and G. Bontempi, “Learned lessons in credit card fraud detection from a practitioner perspective,” Expert Syst. Appl., vol. 41, no. 10, pp. 4915–4928, 2014.

[8] C. Jiang, J. Song, G. Liu, L. Zheng, and W. Luan, “Credit Card Fraud Detection: A Novel Approach Using Aggregation Strategy and Feedback Mechanism,” IEEE Internet Things J., vol. 5, no. 5, pp. 3637–3647, 2018.

[9] A. Correa Bahnsen, D. Aouada, A. Stojanovic, and B. Ottersten, “Feature engineering strategies for credit card fraud detection,” Expert Syst. Appl., vol. 51, no. January, pp. 134–142, 2016.

[10] R. A. Bauder, T. M. Khoshgoftaar, and T. Hasanin, “An Empirical Study on Class Rarity in Big Data,” Proc. - 17th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2018, pp. 785–790, 2019.

[11] C. X. Zhang, S. Xu, and J. S. Zhang, “A novel variational Bayesian method for variable selection in logistic regression models,” Comput. Stat. Data Anal., vol. 133, pp. 1–19, 2019.

[12] S. H. Ebenuwa, M. S. Sharif, M. Alazab, and A. Al-Nemrat, “Variance Ranking Attributes Selection Techniques for Binary Classification Problem in Imbalance Data,” IEEE Access, vol. 7, pp. 24649–24666, 2019.

[13] G. E. Melo-Acosta, F. Duitama-Munoz, and J. D. Arias-Londono, “Fraud detection in big data using supervised and semi-supervised learning techniques,” 2017 IEEE Colomb. Conf. Commun. Comput. COLCOM 2017 - Proc., 2017.

[14] S. C. Satapathy, K. Srujan Raju, J. K. Mandal, and V. Bhateja, “Proceedings of the second international conference on computer and communication technologies: IC3T 2015, Volume 1,” Adv. Intell. Syst. Comput., vol. 379, pp. 681–689, 2016.

[15] C. K. Maurya, D. Toshniwal, and G. V. Venkoparao, “Online anomaly detection via class-imbalance learning,” 2015 8th Int. Conf. Contemp. Comput. IC3 2015, pp. 30–35, 2015.

[16] A. Charleonnan, “Credit card fraud detection using RUS and MRN algorithms,” 2016 Manag. Innov. Technol. Int. Conf. MITiCON 2016, pp. MIT73–MIT76, 2017.

[17] A. Kumar and G. Gupta, “Fraud Detection in Online Transactions Using Supervised Learning Techniques,” 2018.

[18] L. T. Nghiem, T. T. Thu, and T. T. Nghiem, “MASI: Moving to adaptive samples in imbalanced credit card dataset for classification,” 2018 IEEE Int. Conf. Innov. Res. Dev. ICIRD 2018, no. May, pp. 1–5, 2018.

[19] S. Xuan, G. Liu, Z. Li, L. Zheng, S. Wang, and C. Jiang, “Random forest for credit card fraud detection,” ICNSC 2018 - 15th IEEE Int. Conf. Networking, Sens. Control, pp. 1–6, 2018.

[20] S. Xuan, G. L. B, and Z. Li, “Refined Weighted Random Forest and Its Application to Credit Card Fraud Detection,” vol. 11280, pp. 343–355, 2018.

[21] A. Roy, J. Sun, R. Mahoney, L. Alonzi, S. Adams, and P. Beling, “Deep learning detecting fraud in credit card transactions,” 2018 Syst. Inf. Eng. Des. Symp. SIEDS 2018, pp. 129–134, 2018.

[22] A. M. Mubalaike and E. Adali, “Deep Learning Approach for Intelligent Financial Fraud Detection System,” UBMK 2018 - 3rd Int. Conf. Comput. Sci. Eng., pp. 598–603, 2018.

[23] I. Benchaji, S. Douzi, and B. El Ouahidi, Using genetic algorithm to improve classification of imbalanced datasets for credit card fraud detection, vol. 66. Springer International Publishing, 2019.

[24] H. He and E. Garcia, “Learning from imbalanced data,” Ieee Trans. Knowl. Data Engin, vol. 21, no. 9, pp. 1263–1284, 2009.

[25] Di. S. Sisodia, N. K. Reddy, and S. Bhandari, “Performance evaluation of class balancing techniques for credit card fraud detection,” IEEE Int. Conf. Power, Control. Signals Instrum. Eng. ICPCSI 2017, pp. 2747–2752, 2018.

[26] P. Xenopoulos, “Introducing DeepBalance: Random deep belief network ensembles to address class imbalance,” Proc. - 2017 IEEE Int. Conf. Big Data, Big Data 2017, vol. 2018-Janua, pp. 3684–3689, 2018.

[27] K. Randhawa, C. K. Loo, M. Seera, C. P. Lim, and A. K. Nandi, “Credit Card Fraud Detection Using AdaBoost and Majority Voting,” IEEE Access, vol. 6, pp. 14277–14284, 2018.

[28] K. Fu, D. Cheng, Y. Tu, and L. Zhang, "Credit Card Fraud Detection Using Convolutional Neural Networks," J. Soc. Mech. Eng., vol. 90, no. 823, pp. 758–759, 2017.

[29] H. Liu and M. Zhou, "Decision tree rule-based feature selection for large-scale imbalanced data," 2017 26th Wirel. Opt. Commun. Conf. WOCC 2017, pp. 1–6, 2017.

[30] ULB Machine Learning Group, "Credit Card Fraud Detection," 2016.

[31] U. R. Salunkhe and S. N. Mali, "Classifier Ensemble Design for Imbalanced Data Classification: A Hybrid Approach," Procedia Comput. Sci., vol. 85, no. Cms, pp. 725–732, 2016.

[32] M. Zamini and G. Montazer, "Credit Card Fraud Detection using autoencoder based clustering," 2018 9th Int. Symp. Telecommun., pp. 486–491, 2019.

[33] H2O AI TEAM, "H2O AI," 2016.

[34] S. J. Omar, K. Fred, and K. K. Swaib, "A state-of-the-art review of machine learning techniques for fraud detection research," 2018 IEEE/ACM Symp. Softw. Eng. Africa, pp. 11–19, 2018.