Deep Learning vs. traditional Machine Learning algorithms used in Credit Card Fraud Detection

Sapna Gupta
X14115824
Masters in Data Analytics
School of Computing
National College of Ireland

***Abstract*—With the continuing growth of E-commerce, credit card fraud has evolved exponentially, where people are using more on-line services to conduct their daily transactions. Fraudsters masquerade normal behaviour of customers to achieve unlawful gains. Fraud patterns are changing rapidly where fraud detection needs to be re-evaluated from a reactive to a proactive approach. In recent years Deep Learning has gained lot of popularity in image recognition, speech recognition and natural language processing. This paper seeks to understand how Deep Learning can be helpful in finding fraud in credit card transactions and compare Deep Learning against several state of the art algorithms (RF, GBM, GLM) and sampling methods (Over, Under, Hybrid, SMOTE and ROSE) used in fraud detection. The results show that Deep Learning performed best with the highest Recall (accuracy of identifying fraudulent transactions), which means lowest financial losses to the company. However, Deep Learning achieved the lowest Precision rate (classified more legitimate transactions as fraudulent), which can cause customer dissatisfaction. Among other chosen classifiers, oversampling method performed best in terms of AUC, precision was highest for GLM and F-Score was highest for model trained using ROSE sampling method. Recall and Precision both have high cost, so there cannot be any trade of one against the other. Selecting the best classifier to identify fraud is based on the business goal.**

Keywords*—*Imbalanced class, Data mining, Sampling methods, H2O, Gradient Boosting, Random Forest, Generalized Linear Models, Deep Learning, Grid search, Hyper parameter optimization, Ensemble methods.

I. Introduction

In today's fast moving world, where millions of financial transactions happen every day, the probability of financial fraud is very high. In 2015, £755 million of losses were calculated by Financial Fraud Action UK (FFA UK), which is 26% higher than in 2014. Fraud losses on UK-issued cards totalled £567.5 million in 2015, an 18% increase from £479 million in 2014; the fourth consecutive year of increase [1]. At the same time, in the US around $20 billion was lost as a result of credit card fraud with 12.7 million US victims of this fraud. This indicates that credit card fraud is a worldwide problem, affecting both financial institutions and customers [2].

There are various ways in which a fraud can be committed. One approach is stolen identity, in which a synthetic id is created by a fraudster and injected into the bureau, to get a new credit card issued. Fraudster uses the received newly issued card to make fraudulent transactions and then moves on to next financial institution to get another credit card to repeat the process. These fraudsters build a false credit history and end up cheating multiple banks. The second way to commit fraud is using a Counterfeit credit cards. Third way to commit fraud is using stolen credit card. As per on-line statistics, 45% of fraud happened due to card not being present, 37% due to counterfeit cards and 14% accounted to lost or stolen cards [2].

There are many challenges faced by researchers performing any study on credit card transactions. Under data protection laws, financial institutions do not share their customers credit card transactions with researchers. To find a real dataset is a big challenge. Second challenge is to keep pace with changing behaviour of fraudsters. Once acquired, a dataset is highly imbalanced i.e ratio of legitimate transactions is very high as compared to fraudulent transactions [10] [6] [12].

Out of millions of credit card transactions happening in a fraction of time, it is not possible to manually identify fraudulent transactions, so there is a need for automated fraud detection systems [29] [28]. Data mining provides an automated and quicker way of finding fraud in millions of transactions without any human intervention. Credit card fraud detection is a binary classification problem, where outcome is either that the transaction is fraudulent or the transaction is legitimate. There are three main considerations while analysing credit card transactions. As it is hard to find fraudulent transactions in a highly imbalanced dataset, there is a need of smart mining solution to balance the dataset without losing any important information. Second consideration is to find a machine learning algorithm, which can learn from such imbalanced dataset with high accuracy. Finding false positives is a lesser financial loss to the company than finding false negatives (fraudster identified as legitimate), which can lead to heavy financial losses in a very short time. So the third main consideration is to use the best model performance metrics for assessing the results of trained models.

Previous studies [10] [6] [14] have proved that sampling methods like Oversampling, Undersampling, Hybrid, ROSE and SMOTE have improved overall classification performance compared to the imbalanced data sets. Along with sampling methods, three very common Ensemble machine learning techniques used in fraud detection are Bagging, Boosting and Stacking. These ensemble machine learning methods use multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms [31]. Deep learning has been getting a lot of attention lately with breakthroughs in image classification and speech recognition. However, its application to fraud detection has not yet been explored. Due to high popularity

of deep learning in the field of machine learning, we chose deep learning for comparing against Sampling, Bagging(RF), Boosting(GBM) and Stacking ensemble methods commonly used in credit card fraud detection applications.

A. The layout of the paper consists of five sections

Section II, "Related work" will discuss previous technologies used in the field of credit card fraud detection. The section will also discuss how other researchers have dealt with class imbalance problem in credit card fraud transactions. Section III, "Design" documents the choosing of the dataset, finding the challenges and designing different approaches. Section IV, "Implementation" continues by examining different approaches to deal with different challenges in credit fraud detection. Section V, "Evaluation" assess the accuracy and performance of different approaches implemented. Section VI, "Conclusions" summarises the findings discovered in this paper and the direction and suggestions for future work.

II. Related Work

Credit card fraud is a well researched topic, most of the studies are based on either learning customer buying behaviour or handling class imbalance problem using various data mining techniques. These studies are discussed under two subsections.

A. Studies based on learning normal customer spending behaviour

Most of the studies found are based on learning customer buying behaviour using various data mining techniques and using these learning to predict unseen credit card transactions. Jha et al. [27] aggregated customers historical transactions and built a logistic regression model to learn customer buying behaviour and to predict the new transaction as legitimate or fraudulent. Panigrahi et al. [26] developed a four step process to catch fraud. In the first step he used a rule based filter to determine suspicion level of each transaction, in the second step he used Dempster-Shafer's theory to learn customer behavior, in the third step, based on the past behaviour transaction is classified as normal, abnormal or suspicious and then in the fourth step, he used Bayesian theory on suspicious transactions to label them as fraudulent or legitimate.

Wong et al. [21] and Brabazon et al. [32] investigated application of Artificial Immune System (AIS), a biological technique to find fraud in credit card transactions. AIS learns the normal pattern of the customer buying behaviour and behaves normally if the new transaction is legitimate and behaves abnormally as soon as a fraudulent transaction enters the system. Credit card transactions are dynamic in nature as fraudsters continually adapt their strategies in response to the increasing sophistication of detection systems. Artificial Immune System can be useful to flag non standard transactions without having seen previous examples of such transactions during training of the algorithm [32]. Wong et al. talk about comparing AIS against logistic regression model, but no results are presented in the paper which compare the AIS with logistic regression model. AIS has three methods: Unmodified Negative selection, Modified negative selection and clonal selection and these methods have been compared with each other, but none of these three methods have been compared with logistic regression model. The study also shows it is a time consuming process and needs more research to be done to improve the performance of the training process.

Syeda et al. [22] developed a parallel processing neural network GNN for finding fraud in a very short time and proved that higher accuracy in prediction required more data for training. Detection error rate is higher for actual fraudulent transactions. Sanchez et al. [25] proposed a fuzzy association rule based engine by learning normal behaviour of the customer. Performance of this model is still a question where number of transactions are growing exponentially. Seyedhossein and Hashemi [24] aggregated daily amounts spent of individual cardholders in a given time window to find fraud in time. For increased prediction accuracy, seasonal customer buying behaviour is also incorporated in the model.

Srivastava et al. [23] built a model using Hidden Markov Model (HMM) based on normal customer spending behaviour. Transactions rejected by HMM are classified as fraudulent. Results show that training time of the model increased as number of transactions increased. Seeja et al. [10] proposed a new framework 'FraudMiner', which used frequent Itemset Mining approach (Apriori) to learn customers normal buying behaviour. All attributes were given equal importance in order to handle the anonymous nature of the dataset. Every new incoming credit card transaction was tagged as fraudulent or legitimate using a matching algorithm. The authors compared performance of 'FraudMiner' with SVM and RF and found that 'FraudMiner' performed best.

B. Studies dealing with class imbalance problem in fraud

Credit card fraud is a binary classification problem and data is highly skewed towards the legitimate transactions which leads to class imbalance problem. Brennan proposed two methods to deal with class imbalance problem: algorithmic centric and data centric [6]. Data centric approach was applied using Oversampling, Undersampling and SMOTE. Under algorithmic approach, the author used Naive Bayes, ID3, C4.5, KNN, RF and RIPPER on the various samples of the three different data sets. The algorithmic method focused on choosing best learner by measuring misclassification cost, using Metacost procedure or probability thresholds. As per the results of data centric approach, the oversampling method shows the best performance and undersampling of majority class showed the poorest performance. Results of the algorithmic approach shows that F-measure of RF algorithm was the best. The author also stated that "Training models using balanced dataset but not testing the performance on the balanced dataset can lead to an incorrect assessment" [6].

Dal et al. [9] proposed a data mining solution to process non stationary real-time credit card fraud transactions by creating a new model every time a new chunk of data arrived in the system and this solution could also handle class imbalance problem . The experimental setup compared several state of the art algorithms (RF, SVM, NNET), Sampling methods (Under, SMOTE, Easy-Ensemble) and modeling techniques on a real data set. Models updated on higher frequency (Daily or more than once in a day) performed better than models updated on lower frequency (Weekly or once in 15 days or monthly). Results show that frequency of updating the models is very crucial in a non-stationary environment. The author

demonstrated that Random Forest performed better than Neural Network and Support Vector Machine.

Abdulla et al. [7] introduced a three stage hybrid approach to detect fraud in credit card transactions. The first stage was pre-processing, in which anonymous transactions were removed. In the second stage, a genetic algorithm was used for feature selection and in the last stage support vector machine (SVM) was used for classification. As per the study, feature selection using K Nearest Neighbour approach and a model trained using SVM showed better accuracy. However, there are no clear results to prove the proposed hybrid approach worked well. Lee et al. [14] presented in his study a new sampling method called Oversampling via randomly imputed features (ORIF), to deal with huge volume of imbalanced e-commerce transactions paid by credit card. ORIF generates artificial instances for minority classes (fraudulent) and does not impose any restructuring on the data compared to SMOTE, which adds artificial neighbors to the minority class. The author stated that "ORIF is very easy and fast to implement. ORIF works well with all types of data sets and does not require the distance metric on the feature space which is hard to define when it is a mixture of numerical and categorical variables" [14]. There was no evidence to prove the hypothesis that it performed better than SMOTE.

With the constant evolution of new technology like H2O and new evolving machine learning techniques like Deep Learning, there is a constant need to study the application of these new techniques in the field of fraud. In this paper we are trying to compare deep learning with other traditional classification techniques used in credit card fraud detection.

III. Design

A. System Architecture

Initial analysis was carried out on manually created Ubuntu Virtual Machine (VM), where all required tools were installed manually. Integration of these tools was a very big challenge, so final benchmarking exercise was carried out using Data Science Studio (DSS). DSS is a collaborative data science software platform that enables data analysts, data scientists and data ops to explore, prototype, build and deliver their own data products more efficiently [3]. Fig 1 shows an overview of the system configuration and bigdata tools used in building the fraud detection application.

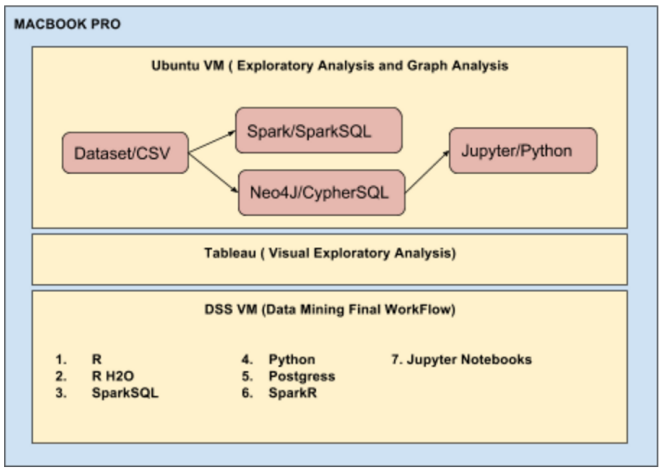

Fig. 1: Overview of the hardware and software used during building process of the Fraud detection application

Model training was first tried using Rapid-minor installed on the local machine and then using Spark(Mlib) on Ubuntu Machine. Model training using these systems was very slow, the system kept crashing and required lots of technical skills. H2O integration with R is an easy to use interface, so models were trained using H2O library in R. A single node H2O cluster with 5GB memory was launched on local machine for training various models. H2O was chosen to build all machine learning models because it has built-in APIs to carry out deep learning, boosting and bagging ensemble techniques. Among the data scientist community H2O is gaining a lot of popularity due to its state-of-art open source programming engine which provides an easy application for machine learning and deep learning on any type of dataset. Over 200 corporations are using H2O [18]. A key feature of H2O is that we do not need to do sampling of data, but still predictions of data can be done quickly. H2O was used because it can compress data in memory and handle billions of rows in memory. H2O provide interfaces for all programming languages e.g. R, Python, Scala, Java ,Json and java scripting. H2O was run in a standalone mode and was deployed in minutes. H2O has a scalable engine with various modes and is able to process data in parallel.

B. Data Source

The dataset used for analysis is hosted from UCSD-FICO Data mining contest 2009 [4] and is also used by other studies [10] [7] [11] [12] [30]. There were two data sets: an easy set with nineteen attributes including the label class and a hard set with twenty attributes including the label class. The hard set comes with customer id and email address, whereas the easy set had domain id instead of customer id. These two data sets had different column structures, so there was no possibility of merging these datasets. Easy set had no customer id, so we decided to use only the hard set for this analysis [5].

The hard set is further comprised of two data sets: training set and testing set. The hard training dataset contains 100,000 transactions and hard testing dataset contains 50,000 credit card transactions. Twenty attributes found in the hard set are class labels, amount, hour1, state1, zip1, custAttr1, field1,

custAttr2, field2, hour2, flag1, total, field3, field4, indicator1, indicator2, flag2, flag3, flag4 and flag5. Names of the attributes were anonymised, looking at the values in column custAttr1, it was mapped to customer id and custAttr2 was mapped to email id. The attributes corresponding to state and email information are categorical features. The testing set contains nineteen attributes the same as the training set except for Class labels. SPARK SQL was used to analyse these training and testing data sets to find any common information in both the data sets. There were no common transactions found in the training and testing data sets. Out of 100,000 transactions in training data set only 2654 cases were fraudulent, so the training dataset had around 2.6% fraudulent cases. The class distribution in the dataset is highly imbalanced. In this case, standard classifiers tend to have a bias in favor of the larger classes and ignore the smaller ones.

Class imbalance is a very common problem in data mining applications. There are various proven data mining techniques tested by other practitioners on the chosen dataset [10] [7] [11] [12]. Deep Learning is gaining a lot of popularity and has not been tested on an imbalanced transactional dataset. The objective of this paper is to compare Deep learning against other proven sampling and ensemble data mining techniques, using the above same dataset.

C. Sampling Methods

Oversampling, Undersampling, hybrid, SMOTE and ROSE sampling methods were chosen to resolve the problem of class imbalance and models trained using these samples are compared against Deep Learning. In Oversampling, minority class observations are duplicated to obtain a balanced dataset. In Undersampling, majority class observations are dropped to obtain a balanced dataset. Hybrid is using both Oversampling and Undersampling to get a balanced dataset from imbalanced dataset. ROSE is synthetic generation of data to get a balanced dataset. The data generated using ROSE is considered to provide a better estimate of the original data. SMOTE is another very popular sampling method, which generates new synthetic data by randomly interpolating pairs of nearest neighbours. It is a widely used sampling method, which creates artificial database features similar to minority samples. It generates a random set of minority class observations to shift classifier learning bias towards minority class. SMOTE uses bootstrapping and K-nearest neighbours. No sampling is where all data points from majority and minority training sets are used. These sampling methods have some drawbacks. In Oversampling representation of minority class may lead to Overfitting or Underfitting and there is loss of significant information.

D. Ensemble Methods

Ensemble machine learning methods use multiple learning algorithms to obtain better predictive performance than could be obtained from any of the single learning algorithms [12]. In addition to sampling methods(along with decision tree), ensemble techniques like bagging , boosting and stacking have been used for comparison against deep learning [13]. Random Forest (RF) is a widely used Bagging Ensemble machine learning algorithm, which generates a forest of classification trees, rather than a single classification tree. Each of these trees generates a classification for a given set of attributes [30]. The classification results from each tree is aggregated to give the final prediction. Gradient Boosted Model (GBM) is a forward-learning ensemble method consisting of either regression or classification tree models that obtains predictive results through gradually improved estimations. Boosting is a flexible non-linear regression procedure that improves the accuracy of trees through sequentially applying weak classification algorithms to the incrementally changed data. The results generate a series of decision trees that produce an ensemble of weak predication models [17]. GBM uses poorly performing functions iteratively to generate a highly predictive output which is binary in nature. This technique uses a set of parameters to be passed as input to these functions. All these functions are simple to understand. There is no limitation in GBM as to type of data, as GBM can accept unclean data [16]. Stacking ensemble is a two step process, wherein a base algorithm is applied on the training set to get a level one data set matrix. All of these level one dataset matrices are then passed through a meta algorithm to generate final output.

E. Deep Learning

Deep learning currently provides the best solutions to many problems in image recognition, speech recognition and natural language processing [33]. The objective of this paper is to compare Deep Learning's performance to other existing machine learning techniques used in finding credit card fraud. Deep learning is a biologically inspired model of a human neurons [15]. Deep learning architecture is composed of multilevel hidden layers of nonlinear processing units, where each neuron may send data to a connected neuron within hidden layers [20]. A weighted combination of all input signals is aggregated and then an output signal transmitted by the connected neuron. Deep learning with default parameters using H2O library automatically handles missing values, data standardisation, load balancing, cross-validation, checkpointing and gridsearch [20]. Deep Learning models are built through assessing different representations of raw data with exhibited high performance on complex data such as images, text and speech. There are many hyper parameters in deep learning which are used to tune the models [15].

F. Performance Metrics

Fraud detection is a binary classification problem, output label is either 'fraudulent' or 'legitimate'(non-fraudulent). There are multiple performance metrics to measure the performance of any binary classification algorithms. Different machine learning algorithms used in designing fraud detection applications are evaluated using performance measures like Precision, Recall (also known as sensitivity), F-Measure, Mathews Co-relation Coefficient and Area Under Curve (AUC) [8].

These performance measures are based on four factors [10]:

True Positive (TP): class was positive and predicted positive
True Negative (TN): class was negative and predicted negative
False Positive (FP): class was negative but predicted positive
False Negative (FN): class was positive but predicted negative

Here fraud is a positive class and legitimate is negative class.

Precision is defined as the number of true positive (fraudulent) case predictions compared to the total number of positive

predictions.

$$Precision = \frac{TP}{(TP + FP)} \quad (1)$$

Recall or sensitivity is defined as the number of true positive (fraudulent) predictions compared to the total number of fraudulent transactions. In fraud detection, the most important measure is Recall or fraud detection rate, as a higher value of recall means a lowest financial loss to the company.

$$Recall = \frac{TP}{(TP + FN)} \quad (2)$$

F-Measure is the weighted harmonic mean of Precision and Recall [10].

$$F - Measure = 2 * \frac{Precision * Recall}{(Precision + Recall)} \quad (3)$$

Mathews Co-relation Coefficient(MCC) is used to measure the quality of binary classifications. This coefficient takes into account true and false positives and true negatives and false negatives [10].

$$MCC = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (4)$$

Accuracy is another metric used for model evaluation and is defined as the number of correct predictions made divided by the total number of predictions made.

$$(Accuracy = (TP + TN)/(TP + TN + FP + FN)). \quad (5)$$

Accuracy of the model can be misleading in case of credit card fraud detection, where the numbers of fraudulent transactions is much lower than the legitimate transactions and the dataset is highly imbalanced.

Receiver Operating Characteristic curve (ROC) shows the sensitivity of the classifier by plotting the rate of true positives (fraudulent classified as fraudulent) to the rate of false positives (non fraudulent classified as fraudulent). The area under curve (AUC) summarizes the ROC curve into a single number and high value of AUC is a better prediction.

Selecting the right performance metrics depends on the business objective because one measure can help to prevent financial losses and the other can help to gain customer satisfaction.

IV. Implementation

Before building any machine learning models, it is important to explore and clean your data. Dataset was explored by plotting various charts using Tableau to understand relationship between different attributes. Attributes 'amount' and 'total' were found to have similar values, so it was decided to drop attribute 'total' from the analysis. Similarly, 'hour1' and 'hour2' attributes also had similar information and it was decided to drop 'hour2' attribute from the analysis. Finally only 18 attributes in training data set were considered for further analysis. It was also found that data is from 50 states in US and majority of data belongs to California (CA) state.There were 254 cases of fraud from a single zip code 708 in state LA. Email id 'zwzihwgzxohnq@cbbtr.com' was used in 207 fraudulent transactions. Maximum fraudulent transactions happened during midday. There was no missing information from any of the attributes of the dataset.

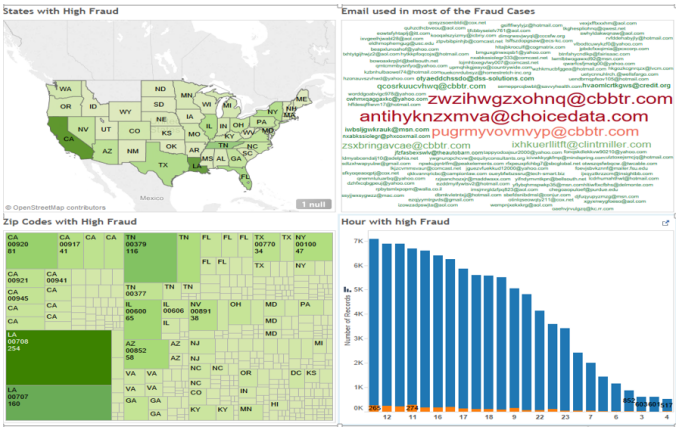

Fig. 2: Visual exploratory analysis using Tableau

In Credit card transactions, attributes like Social Security Number(SSN), mailing address, name and email address are very important to uniquely identify a customer. But multiple occurrences of same attributes like email or SSN for different credit card holders can raise an alert flag of fraud or possible fraud. To accomplish all this, dataset was loaded into Neo4j graph database for graph analysis. As compared to traditional relational data bases, Neo4j easily uncovers difficult to detect hidden fraud patterns found in the dataset. 3D graphs of fraud rings were created using cypher query and using python Jgraph library as shown in Fig 2. Lot of customers were found to be using same email address. There were over 100 customers using same email id 'antihyknzxmva@choicedata.com', which is kind of application fraud where same person is creating multiple identities to get credit cards issued and use them to carry out financial transactions. Primary attribute which was used to find fraud rings is email address, so multiple transactions done by different customers, having same email address were shown in fraud rings as potential fraud cases. Total monetary value of these transactions is the total possible risk or loss to the financial institution. Cypher query used to build the graph is as follow:

```
MATCH (c:custid)-[r1:HAS_EMAIL]->(e:emailid)
WHERE c.class = '1'
RETURN c.name as id, r1, e
```

Fig. 3: Fraud Rings: Same email id used by more then one customer

The main objective of this study was to compare Deep Learning against Sampling methods (Undersampling , oversampling, hybrid, SMOTE and ROSE) and Ensemble methods (RF and GBM). For this purpose, after exploratory and graph analysis data was moved into DSS. Fig 4. shows a complete workflow designed to test all the models built for benchmarking.

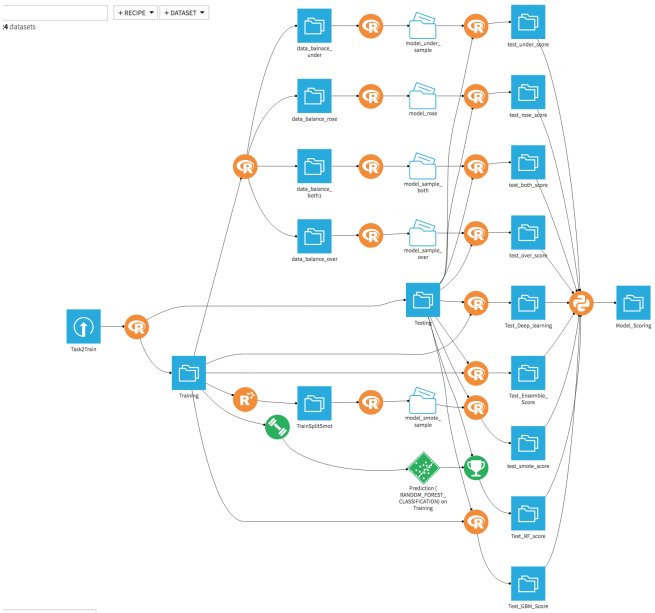

Fig. 4: WorkFlow for compareing different ML algorithms in DSS

In the designed workflow, first task was to divide dataset into 70:30 ratio of training and testing sets. Second step was to train models on training set of 70,000 rows and tested on testing set of 30,000 rows. Third and last step was to validate the trained models. Implementation of the techniques used to train and test the chosen dataset are discussed in three sub sections: Model training using Sampling techniques, Model training using Ensemble techniques and Model training using Deep Leaning.

A. Model training using Sampling methods

Sampling was carried out using 'ROSE' and 'DMwR' libraries in R. 'Ovun.sample' function in 'ROSE' library enabled Oversampling, Undersampling and hybrid sampling. 'ROSE' function in the ROSE library, was used for generating a balanced synthetic sample. 'SMOTE' function under 'DMwR' library, was used to generate artificially new samples of the minority class using the nearest neighbors of the minority class and majority class is under-sampled, leading to a more balanced dataset. Distribution of the label class in the new samples, is shown in the Table I. The new samples generated using above sampling methods were trained using 'rpart' function in the 'rpart' library in R. Models were trained using default parameters. 'predict' function from 'ROSE' package was used to apply the trained models on testing dataset to generate the predicted labels.

TABLE I: Class distribution using different sampling methods

Sampling Methods	Legitimate Transanctions	Fraudulent Transactions
Oversampling	68162	68130
Undersampling	1870	1838
hybridsampling	2564	2436
Smotesampling	3676	3676
Rosesampling	34919	35081

B. Model training using Ensemble techniques

Random forest (RF) ensemble algorithm is commonly used as a starting point in any classification problem. RF with 100 trees was trained using RF modelling API in DSS. Model training using DSS is made very easy, does not require any programing knowledge, but comes with limited model selection. H2O integration with R is well known among data scientists, so models were trained using H2O library in R. First H2O package was installed in R and then models were trained on training dataset using RF bagging technique through 'h2o.rf' function. GBM boosting ensemble methods was implemented using 'h2o.gbm' function in H2O. For stacking ensemble algorithm GLM, RF, GBM and Deep learning was used as base learners and GLM was used as metalearner from h2o.ensemble library.

```
Base-learners <- ("h2o.glm.wrapper",
                  "h2o.randomForest.wrapper",
                  "h2o.gbm.wrapper",
                  "h2o.deeplearning.wrapper")
Meta-learner <- "h2o.glm.wrapper"
```

All models were trained using default parameters. Models were then tested on testing dataset using 'h2o.predict' function.

C. Model training using Deep Leaning

For training model using Deep Learning 'h2o.deeplearning' function was used from H2O library. Model was built using 6 hidden layers of 50 neurons each , 500 epochs and Rectifier activation function. 'Balance_Classes' parameter is set to true in case of imbalance classification problem. Model was also trained using 5 fold cross validation by just adding two more parameters 'nfolds=5' and 'fold_assignment="Stratified"'. For leveraging the power of

H2O Deep Learning model was then trained using Random Hyper-Parameter Grid Search. 'Grid search' is training model using different combinations of list of parameters provided, this leads to more training time. Early stopping parameters (stopping_metric="logloss", stopping_tolerance=1e-2, stopping_rounds=2, score_duty_cycle=0.025) were set for improving model performance in the grid search. Hyper parameters used in Grid search are listed below.

```
activation=c("Rectifier","Tanh","Maxout",
             "RectifierWithDropout",
             "TanhWithDropout",
             "MaxoutWithDropout")
hidden=list(c(20,20),c(50,50),c(30,30,30),
            c(25,25,25,25),
            c(50,50,50,50,50,50))
input_dropout_ratio=c(0,0.05)
l1=seq(0,1e-4,1e-6)
l2=seq(0,1e-4,1e-6)
```

V. Evaluation

Trained models were first evaluated by calculating AUC, Recall, MCC, Precision and F-Score performance metrics using Python's pandas and Sklearn library. Results of these performance metrics for each model selected for comparing models are listed in Table II.

TABLE II: Summary of the Performance Metrics

Modelling Methods	AUC	Recal	MCC	Precision	F Score
OverSampling	0.825	0.681	0.487	0.375	0.483
UnderSampling	0.807	0.707	0.319	0.175	0.281
HybridSampling	0.802	0.695	0.316	0.175	0.279
SmoteSampling	0.808	0.708	0.323	0.178	0.285
RoseSampling	0.818	0.645	0.652	0.678	0.661
Random Forest(DSS)	0.696	0.395	0.54	0.761	0.52
Random Forest(H2O)	0.75	0.518	0.467	0.45	0.482
GBM	0.722	0.451	0.518	0.62	0.522
GLM	0.608	0.218	0.419	0.832	0.346
Deep Learning	0.722	0.722	0.159	0.068	0.123
Ensemble	0.774	0.554	0.617	0.708	0.622

Table II shows that oversampling method performed best in terms of AUC value (0.822). Deep learning could correctly identify most fraudulent transactions with highest Recall value (0.725). MCC score (0.652) was highest for ROSE sampling. Precision value (0.832) was highest for GLM model trained using H2O. F-Score (0.661) was highest for ROSE sampling method. Model trained using random forest(using H2O library) showed better AUC and Recall value as compared to model trained using random forest (using DSS API)

Performance metrics were also plotted in the line chart as shown in the Fig. 5 for better visualization of results.

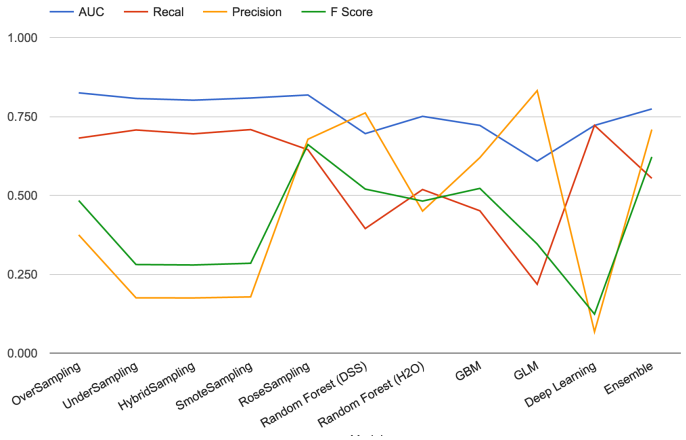

Fig. 5: Line Graph: Visual comparison of the Performance Metrics

Fig 5 shows comparison of performance and effectiveness of various models trained on the same dataset. F-score and precision for most sampling methods is less than 50 percent but AUC and Recall is significantly high. Ensemble methods performed bast in terms of AUC, Fscore and precision but time taken to train the model using ensemble technique is significantly very high. Deep learning has the lowest precision and highest recall values. Random Forest models performed average.

Models were also evaluated using, Type I error and Type II error. Type I error is when a legitimate transaction is marked as fraudulent, which by convention corresponds to a false positive(FP). Type I errors leads to customer dissatisfaction. Bar chart in Fig 6 shows that Ensemble model has lowest Type I error (36) and model trained using deep learning has highest Type I error (8134).

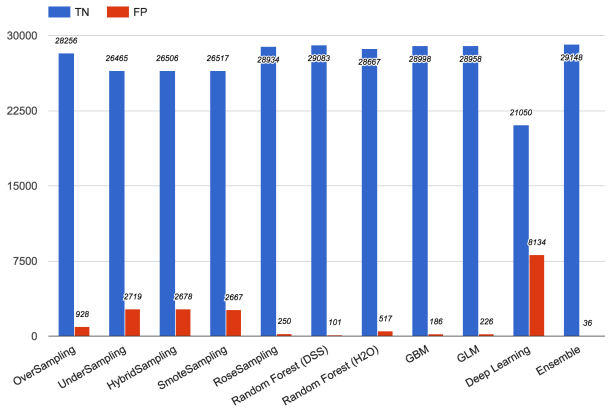

Fig. 6: Type I error plot

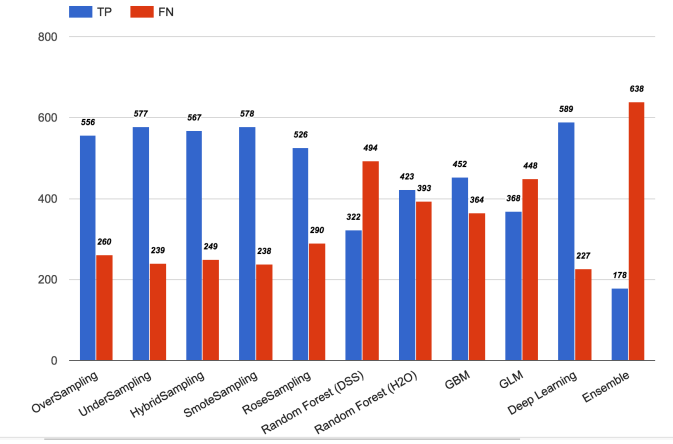

Fig. 7: Type II error plot

Type II error is when a fraudulent transaction is marked as legitimate, which by convention corresponds to a false negative(FN). Type II errors lead to financial loss to the company. Bar chart in Fig 7 shows that Ensemble model has highest Type II error (638) and Deep Learning has lowest Type II error(227). Results shows that, H20 Deep Learning model is the most suitable model, as it is able to find frauds with accuracy of around 72 percent and financial loss to the company is lowest, although customer satisfaction is lowest among all the other models.

For better illustrating the power of Deep learning , models trained using 5-fold cross-validation and hyper-parameter grid search were also evaluated. Model trained using H2O deep learning with default parameters is named as 'Deep Learning DF'. Model trained using 5-fold cross-validation feature of H2O deep learning is named as 'Deep Learning CV' and model trained using hyper-parameter grid search is named as 'Deep Learning Grid'. Grid search summery is listed in the Fig 8. In grid search model-id 'dl_grid_random_model_4' with lowest logloss (0.79) was chosen as best model.

model_ids	epochs	l2	l1	input_dropout_ratio	hidden	activation	logloss
dl_grid_random_model_4	500	0.000004	0.000045	0.05	25,25,25,25	Maxout	0.7989345929
dl_grid_random_model_2	500	0.000045	0.000029	0	30,30,30	Tanh	1.000089482
dl_grid_random_model_3	500	0.000094	0.000036	0	25,25,25,25	Rectifier	1.035693499
dl_grid_random_model_1	500	0.00006	0.000087	0.05	50,50	Tanh	1.080510091
dl_grid_random_model_5	500	0.000003	0.000008	0.05	50,50,50,50,50,50	Tanh	1.198024979
dl_grid_random_model_0	500	0.000045	0.00007	0.05	30,30,30	TanhWithDropout	1.471085054

Fig. 8: Hyper-Parameter Search Summary: ordered by increasing logloss

Results of the AUC, Recall, Precision, MCC and F-Score performance metrics for each of the deep learning model trained are listed in Table III.

Table III shows that deep learning model trained using grid search improved AUC value by 0.753. Deep learning using default parameters, could correctly identify most fraudulent

TABLE III: Summary of Performance metrics of Deep Learning Models

Modelling Methods	AUC	Recal	MCC	Precision	F Score
Deep Learning DF	0.722	0.722	0.159	0.068	0.123
Deep Learning CV	0.75	0.62	0.238	0.125	0.209
Deep Learning Grid	0.753	0.689	0.207	0.095	0.167

transactions with highest Recall value (0.725). Precision value (0.125) , MCC score (0.238) and F-Score (0.209) was best for deep learning model with 5 fold cross validation. Precision , MCC and F-score of deep learning models was much lower than sampling methods shown in Table II.

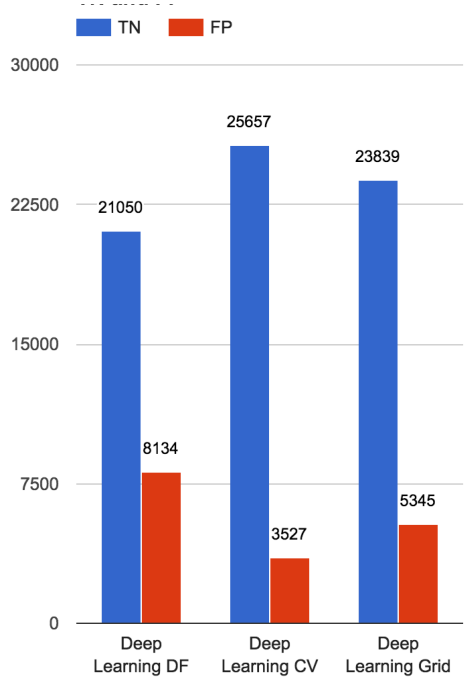

Fig. 9: Type I error plot for Deep Learning models

Bar chart in Fig 9 shows that Deep Learning CV has lowest Type I error (3527) and model trained using deep learning with default parameters has highest Type I error (8134).

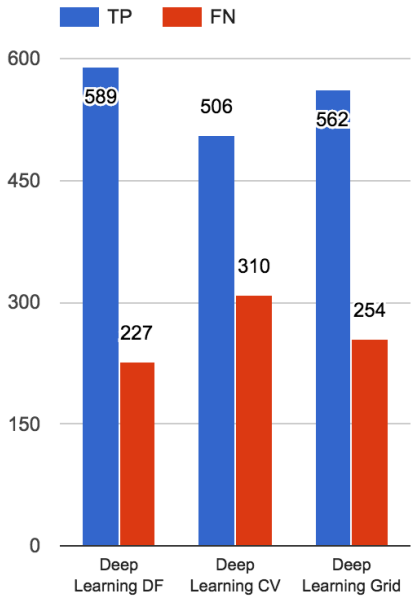

Fig. 10: Type II error plot for Deep Learning models

Bar chart in Fig 10 shows that Deep Learning-(CV) has highest Type II error (310) and Deep Learning-(DF) has lowest Type II error(227).

From all the above results, it is found that deep learning model trained using default parameters performed best in terms of classifying correct fraudulent transactions at the cost of customer satisfaction. Model tuning methods like 5-fold cross-validation and hyper-parameter grid search could not improve performance in terms of identifying more fraudulent transactions and achieving high customer satisfaction simultaneously.

VI. Conclusion

Credit Card Fraud is a growing problem, financial institutions are losing huge amount of money, researchers are implementing new strategies for finding fraud and preventing financial losses. Deep learning is a branch of machine learning and has gained a lot of success in the fields of image recognition, speech recognition and natural language processing [33]. In this study we are trying to compare Deep Learning against several state of the art algorithms (RF, GBM, GLM) and Sampling methods (Over,Under,Hybrid, SMOTE and ROSE), in the field of credit card fraud detection. The study was carried out using highly imbalanced and anonymous credit card transactions acquired from UCSD-FICO Data mining contest 2009. Models were trained on historical credit card transactions using R integration with H2O. Implementation using DSS, R and H2O proved to be the most efficient data science platform in terms of memory efficiency, processing speed and ease of implementation.

Results show that sampling methods perform better as compared to ensemble methods and Deep learning. Sampling method has chance of Overfitting or Underfitting, whereas Ensemble methods and Deep learning remove chances of Overfitting or lose of information. Choosing the right performance metrics is a big challenge in evaluating models trained using highly imbalanced dataset. Type I error is false positive, which can lead to fraudsters remaining undetected, allowing them to achieve their unlawful gains causing financial loss to companies. Type II error is a false negative, which can lead to legitimate customers being accused of being fraudsters and creating customer dissatisfaction. Recall is measure of Type I error and Precision is measure of Type II error. Findings show that a Deep learning model (trained using default parameters) had a very high recall value which means classifying correctly fraudulent transactions. Deep learning model tuning methods like 5-fold cross-validation and hyper-parameter grid search could not improve performance in terms of identifying more fraudulent transactions and achieving high customer satisfaction simultaneously.

In this study only the H2O deep learning library was used, but there is still a need to assess the power of deep learning in the field of credit card fraud detection application using other highly ranked deep learning libraries such as Torch, Theano, Caffe, Neon, TensorFlow, Keras, Deeplearning4J and Spark MLLib. Benchmarking of deep learning in classifying real time credit card transactions using fast GPU is still to be evaluated.

Fraud detection is a well researched topic. Due to new emerging patterns of fraud and the heavy financial losses to the industry each year, there is always need for more research to be done in this area. New emerging technologies like deep learning allow researchers to design smart solutions to address current and future industry challenges. The designed model approach documented in this paper is a strong foundation which can help researchers build towards advanced solutions in the ever evolving field of fraud detection using Deep Learning.

Acknowledgment

The author would like to thank Dr. Jason Roche for his aspiring guidance, invaluably constructive criticism and friendly advice during the project work. The author would also like to thank Ian Bassett , Eoin Gillen and Dr. Barry Haycock, who were a great support throughout the project development process.

References

[1] Katy Worobec: Fraud the facts 2016, https://fraudfacts16.financialfraudaction.org.uk/

[2] Tamara E. Holmes: Credit card fraud and ID theft statistics, http://www.nasdaq.com/article/credit-card-fraud-and-id-theft-statistics-cm520388

[3] Data, C., & Development, S. (n.d.). data iku, 1-9. http://www.dataiku.com/dss/

[4] UCSD: University of California, San Diego Data Mining Contest 2009, https://www.cs.purdue.edu/commugrate/data/credit_card/

[5] Tim Finin: UCSD Data Mining Contest 2009, http://ebiquity.umbc.edu/blogger/2009/05/24/ucsd-data-mining-contest/

[6] Brennan, P. (2012). A comprehensive survey of methods for overcoming the class imbalance problem in fraud detection, (June), 1-107.

[7] Abdulla, N., Rakendu, R., & Varghese, S. M. (2015). A Hybrid Approach to Detect Credit Card Fraud, 5(11), 304-314.

[8] Caruana, R., Niculescu-Mizil, a, Crew, G., & Ksikes, a. (2011). Ensemble Selection from Librairies of Models. Icml, 34, 1-21. doi:10.1145/1015330.1015432

[9] Dal Pozzolo, A., Caelen, O., Le Borgne, Y. A., Waterschoot, S., & Bontempi, G. (2014). Learned lessons in credit card fraud detection from a practitioner perspective. Expert Systems with Applications, 41(10), 4915-4928. doi:10.1016/j.eswa.2014.02.026

[10] Seeja, K. R., & Zareapoor, M. (2014). FraudMiner: A Novel Credit Card Fraud Detection Model Based on Frequent Itemset Mining. TheScientificWorldJournal, 2014(SEPTEMBER 2014), 252797. doi:10.1155/2014/252797

[11] West, J., & Bhattacharya, M. (2016). Some Experimental Issues in Financial Fraud Detection: An Investigation, 7-10. Retrieved from http://arxiv.org/abs/1601.01228

[12] Yang, H., & King, I. (2009). Ensemble learning for imbalanced e-commerce transaction anomaly classification. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 5863 LNCS(PART 1), 866âĂŞ874. doi:10.1007/978-3-642-10677-498

[13] Zang, W., Zhang, P., Zhou, C., & Guo, L. (2014). Comparative study between incremental and ensemble learning on data streams: Case study. Journal Of Big Data, 1-16. doi:10.1186/2196-1115-1-5

[14] Lee, M., & Ham, S. (n.d.). E-commerce Transaction Anomaly Classification, 6-10.

[15] Lanford, J., Ledell, E., Parmar, V., Arora, A., & View, M. (2015). Deep Learning with H2O.

[16] Click, C. (2015). Gradient Boosted Machines with H2O, (August).

[17] Nykodym, T., Nykodym, T., Rao, A., Wang, A., Kraljevic, T., Lanford, J., & Hussami, N. (2015). Generalized Linear Modeling with H2O.

[18] Aboyoun, P., Aboyoun, P., Aiello, S., Fu, A., & Lanford, J. (2015). Fast Scalable R with H2O.

[19] Malohlava, M., Mehta, N., & Iyengar, V. (2016). Machine Learning with Sparkling Water : H2O + Spark, (March).

[20] Miskuf, M., & Zolotova, I. (2016). Comparison between multi-class classifiers and deep learning with focus on industry 4.0. 2016 Cybernetics & Informatics (K&I), 1-5. http://doi.org/10.1109/CYBERI.2016.7438633

[21] Wong, N., Ray, P., Stephens, G., & Lewis, L. (2012). Artificial immune systems for the detection of credit card fraud: An architecture, prototype and preliminary results. Information Systems Journal, 22(1), 53-76. http://doi.org/10.1111/j.1365-2575.2011.00369.x

[22] Syeda, M., Zhang, Y.-Q., & Pan, Y. (2002). Parallel granular neural networks for fast credit card fraud detection. 2002 IEEE World Congress on Computational Intelligence. 2002 IEEE International Conference on Fuzzy Systems. FUZZ-IEEE 02. Proceedings (Cat. No.02CH37291), 1, 572-577. http://doi.org/10.1109/FUZZ.2002.1005055

[23] Srivastava, A., Kundu, A., Sural, S., & Majumdar, A. K. (2008). Credit card fraud detection using hidden Markov model. Ieee Transactions on Dependable and Secure Computing, 5(1), 37-48. http://doi.org/10.1109/tdsc.2007.70228

[24] Seyedhossein, L., & Hashemi, M. R. (2010). Mining information from credit card time series for timelier fraud detection. 2010 5th International Symposium on Telecommunications, IST 2010, 619-624. http://doi.org/10.1109/ISTEL.2010.5734099

[25] Sanchez, D., Vila, M. A., Cerda, L., & Serrano, J. M. (2009). Association rules applied to credit card fraud detection. Expert Systems with Applications, 36(2 PART 2), 3630-3640. http://doi.org/10.1016/j.eswa.2008.02.001

[26] Panigrahi, S., Kundu, A., Sural, S., & Majumdar, A. K. (2009). Credit card fraud detection: A fusion approach using Dempster-Shafer theory and Bayesian learning. Information Fusion, 10(4), 354-363. http://doi.org/10.1016/j.inffus.2008.04.001

[27] Jha, S., Guillen, M., & Christopher Westland, J. (2012). Employing transaction aggregation strategy to detect credit card fraud. Expert Systems with Applications, 39(16), 12650-12657. http://doi.org/10.1016/j.eswa.2012.05.018

[28] Sculley, D., Otey, M. E., Pohl, M., Spitznagel, B., Hainsworth, J., & Zhou, Y. (2011). Detecting adversarial advertisements in the wild. Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD âĂŹ11, 274. http://doi.org/10.1145/2020408.2020455

[29] Phua, C., Lee, V., Smith, K., & Gayler, R. (2010). A Comprehensive Survey of Data Mining-based Fraud Detection Research, 14. http://doi.org/10.1016/j.chb.2012.01.002

[30] Zareapoor, M., & Shamsolmoali, P. (2015). Application of credit card fraud detection: Based on bagging ensemble classifier. Procedia Computer Science, 48(C), 679âĂŞ686. http://doi.org/10.1016/j.procs.2015.04.201

[31] Yale, R. M. (2013). International Journal of Advanced Research in Computer Science and Software Engineering, 3(7), 1356âĂŞ1360.

[32] Brabazon, A., Cahill, J., Keenan, P., & Walsh, D. (2010). Identifying online credit card fraud using Artificial Immune Systems. IEEE Congress on Evolutionary Computation, 1âĂŞ7. http://doi.org/10.1109/CEC.2010.5586154

[33] Liu, L. (2015). Learning Discriminative Feature Representations for Visual Categorization, (February).