


A Deep Learning Approach for Loan Default Prediction Using Imbalanced Dataset


Ebenezer Owusu, University of Ghana, Ghana*

 <https://orcid.org/0000-0002-4670-1342>

Richard Quainoo, University of Ghana, Ghana

Solomon Mensah, University of Ghana, Ghana

Justice Kwame Appati, University of Ghana, Ghana

 <https://orcid.org/0000-0003-2798-4524>

ABSTRACT

Lending institutions face key challenges in making accurate predictions of loan defaults. Large sums of money given as loans are defaulted and this causes a substantial loss in business. This study addresses loan default in online peer-to-peer lending activities. Data for the study was obtained from the online lending club on the Kaggle platform. The loan status was chosen as the dependent variable and was classified discretely into “default” and “fully paid” loans. The dataset is preprocessed to eliminate all irrelevant instances. Due to the imbalanced nature of the dataset, the adaptive synthetic (ADASYN) oversampling algorithm is used to balance the data by oversampling the minority class with synthetic data instances. Deep neural network (DNN) is used for prediction. A prediction accuracy of 94.1% is realized and this emerged as the highest score from several trials with variations in batch sizes and epochs. The result of the study clearly shows that the proposed procedure is very promising.

KEYWORDS

Adaptive Synthetic (ADASYN) algorithm, Deep neural network, Imbalanced dataset, Loan-default, Prediction

1. INTRODUCTION

Lending loans by financial institutions is a strategic business decision to generate funds to remain sustainable and competitive. However, the exercise carries great risk of financial losses to the lending institutions if borrowers fail to honor their obligations. For most developing countries the risk is so high that it can be compared to lottery. Lack of dependable data from borrowers and poor technology make it difficult to track defaulters even if they are closed. This means that for lending institutions to remain viable, they must maintain a portfolio of financial excellence in loan repayment and the easiest way to do that is to hype the interest rate to mitigate losses. This, however, is not a prudent decision in economic sense. We know that certain factors have the tendency to increase the rate of loan default and therefore, it is important to advance effective predictive mechanisms to forecasts the defaults. Several studies have been conducted about prediction of defaults and here are a few reviews.

A study conducted by Cheng et al. (2019) on the prediction of loan repayment patterns on behaviors based on the use of mobile phones. It assessed the low-cost approach of both profitable and

DOI: 10.4018/IJIT.318672

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

non-profitable investments using data acquired, and the outcome of the study showed that, for a set of loan repayment and guarantor networks, there is a high risk of default. For a peer-to-peer lending (P2P) which is an online platform that is not involved of a third party, the default rate will be far high. According to Byanjankar et al. (2015), several approaches have been used for predictions in the past but machine learning and data mining methods are predominantly, indicating a more successful trend. Thus, this study inclines on the same success trend to improve the prediction accuracies. Thus, given a set of n loan defaulters $\{l_1, l_2, \dots, l_n\}$, the study seeks to preprocess and balance the dataset prior to setting up a prediction model to estimate the status of each data instance to *loan default* or *fully-paid*.

We aim at improving prediction performance by using an imbalanced dataset of loan defaulters extracted from Kaggle. However, very few studies focused in dealing with the imbalanced problem (Chen et al., 2018). We make use of a method based on Adaptive Synthetic (ADASYN) oversampling method to balance the loan default dataset and implements a Deep neural network (DNN) algorithm for prediction. Thus, we seek to setup a DNN algorithm to learn from a given balanced dataset (with relatively enough synthetic data instances generated using ADASYN) so it can be applied for predicting the status of loan defaulters. This will assist the financial institutions to be in the known for potential loan defaulters thereby reducing expected losses or overhead cost.

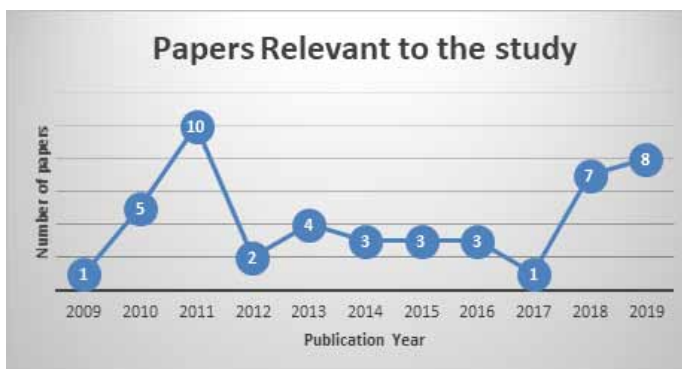
The remaining part of this paper is structured as follows: Section 2 briefly reviews related articles on loan default prediction. Section 3 gives a brief introduction of the proposed approach. Sections 4 and 5 discuss the data preprocessing and prediction techniques in detail. Section 6 discusses the results and analysis. Finally, the conclusion is drawn in section 7.

2. RELATED WORK

Several studies from 2009 to 2019 are analyzed in this section. Figure 1 shows the trend of the relevant articles reviewed per year of publication within the decade. Terms used in the search of the articles involved keywords and phrases such as loan default, prediction, neural network, deep learning, Peer-to-peer, and imbalanced dataset. We focused on those studies that have attempted to compare techniques, measures, or evaluation criteria to build the best possible default prediction model. All articles and journals irrelevant to the study were overlooked.

Due to the extensive attention drawn by peer-to-peer (P2P) lending, Chen et al. (2018) conducted a study into a real credit data from a famous P2P online lending market. According to them, a traditional credit risk prediction models fail to meet the demand of P2P companies for default risk prediction due to uneven distribution of credit data. In that study, a complimentary algorithm of DNN and Linear Regression (LR) was applied to boost the prediction accuracy. Though it seemed to show great

Figure 1. Number of papers per year in review



improvement, the result, indeed cannot be reliable or appropriate because the presence of imbalanced data in the study would cause biases and therefore interferes with the performance obtained.

Another study conducted by Ma et al. (2018) into a new aspect of peer-to-peer online lending default prediction used phone metadata to generate the necessary variables from phone usage. In this study predictions were made using classification models that was developed by demographic information acquired from borrowers. This type of data, however, is insufficient in making accurate predictions since some borrowers do not have bank accounts or access to credit facilities.

Further studies into peer-to-peer lending analysis (Omarina, 2018; Feng et al., 2015), led to the discovery of deep learning for prediction. It was identified that there are several studies that focus mainly on default prediction also known as credit scoring or profitability prediction (PD). Nevertheless, while credit scoring methods appear to demonstrate positive results in reducing the risk of investment, they do not completely address the true goal of P2P lending.

Another study conducted by Serrano-Cinca and Gutiérrez-Nieto (2016) on profit scoring used the internal rate of return as a measure of profitability. The lenders used this model in selecting borrowers with the highest rate of return as a measure of their profitability. In this study, the major flaw was that authors failed to consider the imbalanced problem in their internal rate of the return distribution.

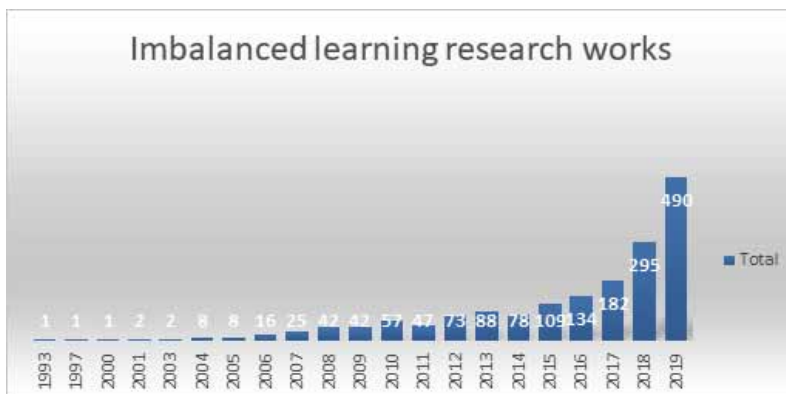
Bastani et al. (2019) also proposed an integrated two-stage approach to predict the probability of default, to determine non-default loans from the listings. Now even with such an innovative model, it is proven that the imbalanced nature of the data is critical to improve accuracy. From the data used, it was identified that 15% of loans fell into default whereas 85% were non-default.

The Hidden Markov Model has also been used earlier for a loan service department that is concerned with reducing the response time and increasing reactivity to deal with defaulted loans (Lee et al., 2011) The model was supposed to automate the process by monitoring payment patterns and generate a signal to the system user, when the probability of default is greater than a pre-specified threshold. However, the developed system was not automated; thus, making the process of predicting the default loans manually among many active loans, a time-consuming and erroneous.

Amin et al. (2016) used basic sampling method for under/over-sampling for loan default prediction. This approach is usually referred to as the manual approach and is the simplest form to deal with under-sampling or oversampling. It involves either eliminating samples from the majority class or duplicating samples from the minority class. However, there were some major errors in the under-sampling approach, and this leads to the discarding of potentially important samples from the majority class, whereas oversampling by duplicating the data would also take a longer time to complete.

The massive attention given to imbalanced learning problems has improved the accuracies. Figure 2 for instance shows an estimation of the number of publications on imbalanced learning over

Figure 2. Imbalanced learning publication trend over the years



the past two decades based on the Institute of Electrical and Electronics Engineers (IEEE) alone. An increasing trend in the activity of publications in this field is explosive.

In previous studies, attempts were made at solving the imbalance problem by using hybrid under-sampling techniques, that combined clustering, stochastic sensitivity measure, and the radial basis function of neural networks (See Figure 2: (2016), (2018), (2019)). However, those approaches have serious flaws for imbalanced datasets because the classifier would classify the entire minority class as part of the majority class when the proportion of the majority class is very high and leads to a biased prediction.

The complex characteristic of an imbalanced dataset requires more research. The main issue with imbalanced learning problems is the ability of imbalanced data to significantly compromise the performance of most standard learning algorithms. These algorithms expect balanced class distributions or equal misclassification costs. Therefore, when presented with complex imbalanced data sets, they fail to properly represent the distributive characteristics of the data (Vluymans, 2019).

The multi-class imbalanced data classification techniques can also adopt ensemble-based methods for resolving imbalanced problems (Bi, & Zhang, 2018) but are more complex compared to the binary imbalance learning. Multi-class imbalance learning extends binary imbalance classifiers to multi-class data using the decomposition methods or adapts to the inherent method of constructing decision trees.

The research gap in most of the reviewed articles is the use of imbalance dataset for setting up the classification models. This leads to bias results since the classification models tend to predict more of the *non-default* class instances as compared to the *default class* instances.

3. PROPOSED METHOD

First, in this study, we developed a method based on adaptive synthetic sampling (ADSAYN) to resolve the imbalanced problem of loan dataset. The method is based on the idea of adaptively generating minority data samples according to the data distribution. The procedure is chosen because it has a tendency of generating more synthetic data for minority class samples that are harder to learn, compared to other minority class samples that are easier to learn. In doing so, it does not only reduce the learning bias introduced by the imbalanced dataset but can also adaptively shift the decision boundary to focus on those difficult to learn samples.

Although there is a similar method previously created known as the Synthetic Minority Oversampling Technique (SMOTE) (Pears, Finlay, & Connor, 2014; Duan, 2019), this approach generates an equal number of synthetic samples for each minority data sample. The approach used in ADASYN, however, does not only provide a balanced representation of the data distribution but force the learning algorithm to focus on those difficult to learn samples. Thus, it is more efficient than the likes of SMOTE (Pears, Finlay, & Connor, 2014), SMOTE Boost (Chawla, Lazarevic, Hall, & Bowyer, 2003) and Data Boost-IM (Ceballes-Serrano et al., 2012) which all rely on the evaluation of hypothesis performance to update the distribution function. We then used a deep neural network (DNN) to train and test for improved prediction accuracy.

4. METHODOLOGY

4.1. Data Description

The dataset obtained for the analysis was extracted from an American Lending Club and it comprises of loan default records from 2007-2015. The Lending Club an online peer-to-peer lending platform with its headquarters in San Francisco, California. Further details of this secondary data are found at Kaggle¹ (where dataset was extracted) and we provide a sample of the dataset provided in Figure 3.

This data consists of 74 features and 887,371 instances, which did not include loans whose status was “Current” (Jagannatha et al., 2010). The genetic search algorithm which is an optimization

Figure 3. Sample of online P2P lending dataset

# loan_amnt	# funded_amnt	# funded_amnt...	Δ term	# int_rate	# installment
2500	2500	2500	36 months	13.56	84.92
30000	30000	30000	60 months	18.94	777.23
5000	5000	5000	36 months	17.97	180.69
4000	4000	4000	36 months	18.94	146.51
30000	30000	30000	60 months	16.14	731.78
5550	5550	5550	36 months	15.02	192.45
2000	2000	2000	36 months	17.97	72.28
6000	6000	6000	36 months	13.56	203.79
5000	5000	5000	36 months	17.97	180.69
6000	6000	6000	36 months	14.47	206.44
5500	5500	5500	36 months	22.35	211.05
28000	28000	28000	60 months	11.31	613.13
11200	11200	11200	36 months	8.19	351.95
6500	6500	6500	36 months	17.97	234.9
22000	22000	22000	60 months	12.98	500.35
3500	3500	3500	36 months	16.14	123.3
7000	7000	7000	36 months	12.98	235.8

technique was used for selecting the relevant features for the classification model. It uses a biological evolution principle to initially generate a population set, followed by a cross-over mutation resulting in new offspring feature set from existing parent features. The offspring or new feature set generated or selected for modeling are *loan amount*, *interest rate*, *installment*, and *annual income of the borrower*; all of which have a numerical datatype. Note that these five are the independent features or variables.

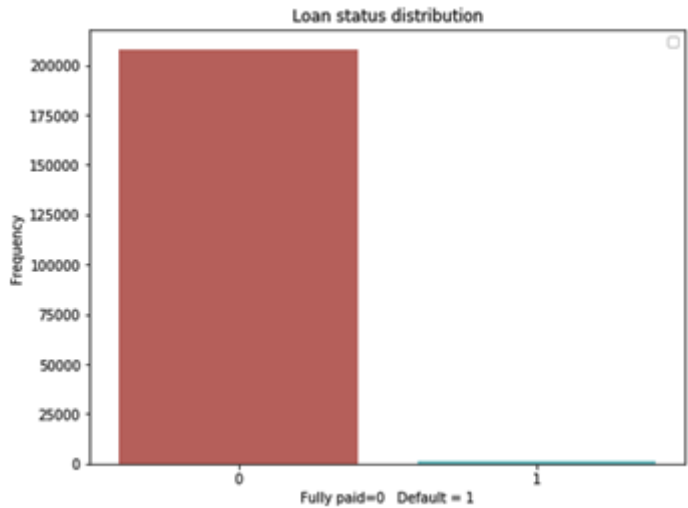
The dependent or target variable used for the analysis is the loan status, a categorical data type, that has been classified into two main categories namely, “Default” and “Fully Paid”. From Table 1, the classification identifies that 99% of loans are Fully paid whereas 0.7% are defaulted. This is an indication of good risk management in the lending club. Though prediction accuracy may be very high, the output of the model will be inaccurate due to the problem of overfitting that will happen in the training phase. The loan distribution is shown in Figure 4.

The nature of the loan dataset in terms of loan status was highly imbalanced with the majority class as fully paid, whereas the minority class as the default. This calls for preprocessing the data by using effective approaches such as cleaning and selection of some appropriate features for classification.

Table 1. Loan status classification

Class	Loan status	No of loans	(%)	Amount	(%)
1	Default	1,219	0.58	18,521,300.00	0.66
0	Fully Paid	207,723	99.42	2,772,344,050.00	99.34

Figure 4. Class distribution of imbalanced dataset – loan status



4.2 Data Preprocessing

4.2.1 Data Cleaning

All data with features of missing values less than 80% were dropped to avoid misclassifications. Data samples in the dependent feature belong to Class 0 (Fully paid) and Class 1 (Default). The algorithm for the procedure is presented in Algorithm 1.

The dataset has the *member_ID* attribute with missing values due to confidentiality reason. From Algorithm 1, we set all missing values across the feature set to NULL prior to setting up the prediction model. Note that, dropping instances with NULL values did not significantly affect the prediction modelling.

The data cleaning algorithm has three steps. Step 1 identifies all cells both those with null and non-null values. The last step 3 follows step 2 which computes the percentage of missing values in the dataset based on the expression provided in step 2. The expression finds the ratio of the product of the number of cells with null values and non-null values to the overall number of cells.

In step 3, we drop features with more than 80% NULL values. Thus, any feature which has null values 80% and above was not considered for the modelling since it will affect the performance of the model. It should be noted that, none of the 5 selected features was dropped after the computation of NULL values.

First Algorithm

Algorithm 1 Data cleaning algorithm
<i>Step 1: Identify all cells in the dataset that have no null values and those with null values</i>
<i>Step 2: Identify the percentage of missing values in the dataset by using the expression,</i> $\left[\left(N_{(null)} \times S_{(null)} / L_{(data)} \right) 100 \right] \%$ <i>, where $N_{(null)}$ represents all cells in the dataset that have no null values, $S_{(null)}$ represents total number of cells with null values, and $L_{(data)}$ represents total number of records in the dataset.</i>
<i>Step 3: Drop features with more than 80% NULL values.</i>

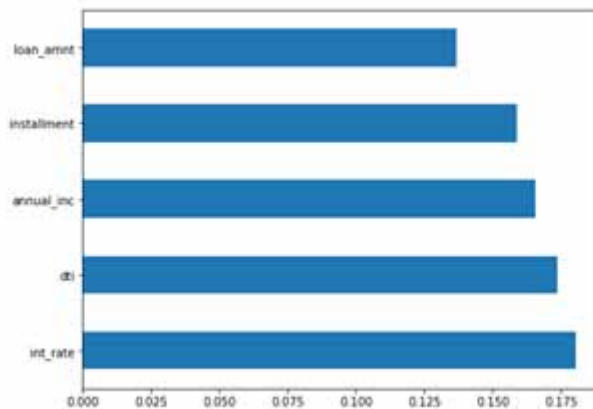
4.2.2 Feature selection

The non-informative predictors are removed to improve the performance of the prediction model. Five features were selected from the dataset. The dependent feature is the loan status classified into default and fully paid loans; the remaining features are the independent features. Except for the member id column, which was used as a unique identifier, all independent features chosen are numerical, whereas the dependent feature is categorical. The loan dataset consists of 74 features out of which the dependent and independent features were selected for the training the algorithm. The univariate feature selection, feature importance and the correlation matrix methods were used in the selection process. The feature importance gives a score for each feature of the data and it is demonstrated in Figure 5. The higher the score the more relevant the feature towards a good output.

Table 2. Features with the highest scores selected using univariate selection

Features	Score
Annual income	4.27
loan amount	4.21
installment	7.07
Interest rate	2.08

Figure 5. Histogram of selected features based on feature importance



A correlation matrix was employed as part of the feature selection process, from which six features were selected. All the variables chosen have related correlation with the target variable. Therefore, we build the model with all chosen features except for member id since it is a way of identifying a client and therefore does not influence whether a client would falter or not. Although there exist about 0.41 correlation between loan status and the member id, it just serves as identification. It was identified and established that “loan_amnt” and “installment” had the highest correlation (0.95). The selected features consist of data with data types of integers and floats, with only one categorical feature which is selected as the dependent variable (loan status). The correlation matrix is shown in Figure 6 and the variables are described in Table 3.

Figure 6. Correlation matrix of selected features



Table 3. Variables for model prediction

Features	Description	Data type	Variable Type
Loan amount	The listed amount of the loan applied for by the borrower.	Integer	Independent
Interest rate	Interest Rate on the loan	Float	Independent
installment	The monthly payment owed by the borrower if the loan originates.	Float	Independent
Annual income	The self-reported annual income provided by the borrower during registration.	Float	Independent
Loan status	Current status of the loan	Object (categorical)	Dependent

5. DATA MODELING

5.1. Balancing of the Imbalanced Data Set

The major problem of loan default predictions of previous models is the issue of imbalanced datasets. The loan default prediction problem can be classified as an instance of imbalanced classification tasks. This is because the classifier will be overwhelmed by the majority class and in so doing completely ignore the minority class. This dominance of the majority class over the minority class leads to the data being highly under fitted, since the model may not be able to classify the minority class successfully. One major objective of this research is to apply an oversampling technique to bring balance to the data set using the Adaptive synthetic (ADSYN) oversampling method for imbalanced data.

5.2. The ADSYN Algorithm

Let total dataset size be represented by D such that $\{x_i, x_j\}$ is a sample data. Let also the minority and the majority class samples be represented by D_i and D_j respectively, such that:

$$D_i < D_j \text{ and } D = \sum_{i=1, j=1}^T D_{ij}.$$

The algorithm is then presented as follows in Algorithm 2.

Second Algorithm

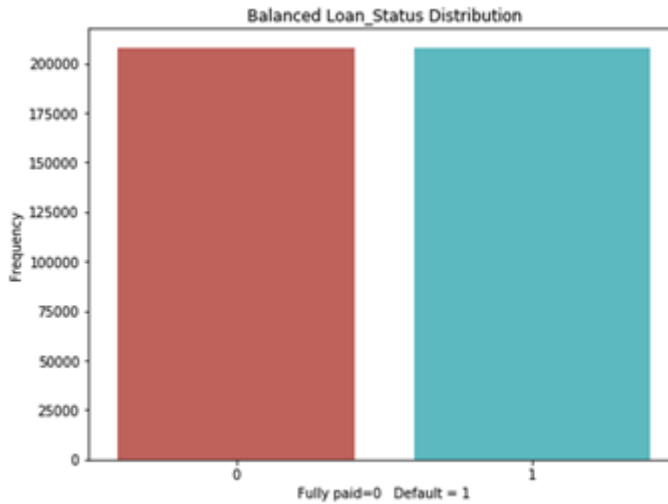
Algorithm 2 The proposed imbalanced ADASYN algorithm
Step 1: Compute $d_B = D_i / D_j$
Step 2: If $d_B < d_p$, where d_p is current threshold for maximum imbalance,
Step 3: Do:
a. Compute $N = \{D_j - D_i\} \mathcal{A}$ where \mathcal{A} represents the balance level of synthetic samples
b. Determine the K-nearest neighbor and compute $n_i = \Delta_i / K$
c. Determine $n_x \leftarrow n_i / \sum n_i$, where n_x is a normalized density distribution
d. Compute each synthetic sample generated for each minority data point p_i , such that $p_i = n_x \sum S_d$, where S_d is one synthetic feature.
e. For $\forall x_i$, generate S_i synthetic data samples for every minority class as follows:
i. Choose 1 minority data, x_m from the K-nearest neighbors for data x.
ii. Compute the synthetic data S_d as $S_d = (x_i + d_f) \gg$, where d_f is the difference vector in n-dimensional space and $\gg \in \{0, 1\}$ is a random number.
Step 4: Apply step (e) to the minority data of random state of 420 and K = 5
Step 5: Fit the variables to the training dataset and compare the results to the initial dataset.

The ADASYN technique was chosen over the popularly used SMOTE technique because of the slight improvement in the nature by which synthetic minority samples are generated, by focusing more on minority samples that are difficult to learn and generating fewer synthetic samples of data that are easier to learn. Again, the method is used because it tackles imbalanced classification problems in generating synthetic data, other than the popularly used synthetic minority under-sampling technique (SMOTE). The imbalanced loan distribution is shown in Figure 7.

5.3 Deep Neural Network Classification

A Deep Neural Network (DNN) is employed, to test for improved prediction accuracy using the balanced dataset. The activation function is used to aid in the generation of binary results. The rectifier

Figure 7. Class distribution of the balanced dataset – loan status



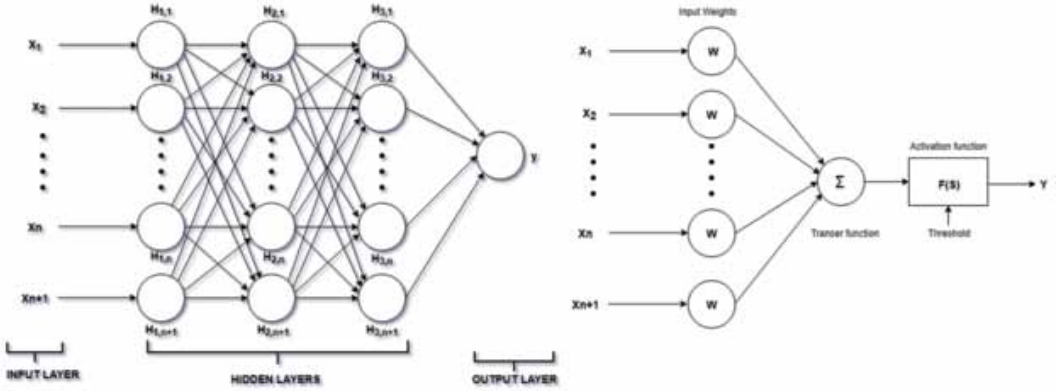
activation function is used for the input layers whilst sigmoid is used for the output layer. The purpose of the rectifier function is to classify that the output variable is binary in the hidden layers, whereas the sigmoid function determines the probability of the output (1 or 0). The number of nodes is changed to one since a binary output is expected. The optimizer, loss, and metrics functions are the necessary parameters used to compile the model. The optimizer function is used for identifying the optimal set of weights that have been initialized for the DNN model (Wang, Pan, & Ahsan, 2020). The chosen optimizer for this function is the Adam stochastic gradient descent algorithm. A binary output is expected so the cross-entropy logarithmic function is used. The dataset was split into training and testing by leave-one-out cross validation, of which 80% of the balanced data consisting of 332217 features were used for training and 20% consisting of 83055 were reserved for testing. To avoid biases in prediction due to anchoring to large values, the loan amount feature of the chosen dataset comprising of large values and not in the same range as that of the interest rate and installments are scaled within the range of -1 and 1. The proposed DNN learning model is represented in Figure 7. We used 3 input layers, 3 hidden layers and 1 output layer.

The weights were initialized randomly to small numbers close to zero. The first observation of the loan dataset is put in the input layer, with the ratio of a single feature to an input node. Forward propagation is done so that the neurons are activated in a manner that their activation is limited by weights. The activations are propagated until the predicted results y is generated. The next step is to measure the error generated, which is the difference between comparing the predicted results with the actual results. The error generated is then back propagated (Kumar, Gaidhane, & Mittal, 2020). The weights are updated according to their relation to the error and then the learning rate decides how much to update the weights. The process is repeated, and the weights are updated after a batch observation.

5.3.1 The DNN algorithm

Let the training data be represented by $x(n)$ and the output data by $y(n)$, whilst n is an integer given by $n = 1, 2, 3, \dots, n, n + 1$. And let the number of layers in the DNN be L . Then the computations of the DNN are executed by computing the weighted sum at each neuron; then the a transfer function $F(S)$ is applied to the weighted sum to determine the neurons output value. The final output y is expressed as a function of the input values and network weights.

Figure 8. Proposed DNN architecture



The sum of the weight, S is computed as follows:

$$S = w_1 x_1 + w_2 x_2 + \dots + w_n x_n + w_{n+1} x_{n+1} = \sum_{i=1}^j w_{n+1} x_{n+1} \quad (1)$$

The output y of the k -th neuron at the l -th layer at input $x(n)$ is then computed as:

$$s(z) = \sum_{i=0}^{p_{l-1}} w_{l,k,i} y_{i-1} i(z) \quad (2)$$

where $W_l^{[q]}$ is a matrix representing the weights q -th iteration. $W_l^{[q]}$ is a matrix computed as:

$$W_l^{[q]} = \begin{bmatrix} w_{l,1,0}^{[q]} & w_{l,1,1}^{[q]} & \dots & w_{l,1,n_l-1}^{[q]} & w_{l,1,n_l}^{[q]} \\ w_{l,2,0}^{[q]} & w_{l,2,1}^{[q]} & \dots & w_{l,2,n_l-1}^{[q]} & w_{l,2,n_l}^{[q]} \\ \dots & \dots & \dots & \dots & \dots \\ w_{l,n_l,0}^{[q]} & w_{l,n_l,1}^{[q]} & \dots & w_{l,n_l,n_l-1}^{[q]} & w_{l,n_l,n_l}^{[q]} \\ w_{l,n_{l+1},0}^{[q]} & w_{l,n_{l+1},1}^{[q]} & \dots & w_{l,n_{l+1},n_l-1}^{[q]} & w_{l,n_{l+1},n_l}^{[q]} \end{bmatrix}_{n_{l+1} \times (n_l+1)} \quad (3)$$

The weights and the biases are adjusted so that the hidden nodes are activated by the rectifier activation function.

5.3.2 Label Encoding

From the loan data set, the dependent variable and the loan purpose is identified as categorical data. These data are encoded to ensure the machine identifies them as numerical values and uses them in the model. It operates by creating separate columns for each variable and assigns them a binary value of either 0 or 1 to classify the loan status as either default or fully paid. The encoded algorithm is represented as follows:

Third Algorithm

Algorithm 3 Encoded algorithm of loan status
<i>Step 1: Import and assign the label encoder</i>
<i>Step 2: Assign the encoded column to the variable</i>
<i>Step 3: Drop the initial loan status column from the table</i>
<i>Step 4: Add the newly encoded label to the table and assigned the same name</i>

6. RESULT AND ANALYSIS

We fitted the DNN to the training set with variations in epochs and batch sizes to gain the highest accuracy. The training process stopped at 100 epochs, 50 units and 32 batch size.

The output of this algorithm resulted in a model with an accuracy score of 0.94. Further studies were undertaken using variations in batch sizes and Epoch. Results showed that using 3 hidden layers with a batch size of 10 and an epoch of 100, the accuracy score was 0.75, whereas using a batch size of 128 and an epoch of 100, the accuracy score was 0.92. The output in the form of probabilities is generated after prediction is made. In order to efficiently classify them, the condition ($y_pred > 0.5$) is set. This condition ensures that the result probabilities fall in the category of either True or False.

6.1 Performance Evaluation

Due to the uneven distribution of classes within our dataset, a predictive model undertaken without balancing the dataset would have a bias in the accuracy of its results. Results obtained when tested on the original dataset showed an accuracy of 99% and an error rate of 1%. The Accuracy of the results is questionable due to the imbalanced nature of the dataset. It is highly likely that the majority class may dominate the minority class, resulting in a model, which is highly under-fitted. The Adaptive Synthetic (ADASYN) oversampling approach is undertaken in order to balance the dataset through minority class oversampling. This is done through the generation of synthetic data from samples in the minority class. The use of ADASYN led to an almost even distribution of data, from an original target variable distribution of {0: 207723, 1: 1219} to an oversampled target variable distribution of {0: 207723, 1: 207549}. The shift in the dataset included an increase in the total number of rows in the dataset from 208,942 to 415,27, leading to an almost even distribution of the majority and minority class.

Although the accuracy is not as high as it would be with the original unbalanced dataset, the accuracy score of the balanced dataset is more realistic. The evaluation of the model made use of the other above stated metrics for assessing its performance. Table 4 shows the output of precision, sensitivity and the specificity. Precision measures the positively classified sets that were truly relevant, sensitivity refers to how good a test is by accurately detecting the positives, and specificity refers to the accuracy of rejecting false alarms in a test. The equations for these metrics are shown in Eq. (4), Eq. (5), and Eq. (6).

$$\text{Sensitivity (recall)} = \frac{tp}{tp + fn} \quad (4)$$

$$\text{Specificity} = \frac{tn}{tn + fp} \quad (5)$$

$$\text{Precision} = \frac{tp}{tp + fp} \quad (6)$$

Where tp , tn , fp , fn are true positives, true negatives, false positives and false negatives respectively. The overall performance of prediction or recognition is denoted by,

$$\text{Accuracy} = \frac{tp + tn}{(tp + fp + tn + fn)} \quad (7)$$

The results from the computation of the sensitivity (recall), specificity and precision are shown in Table 4. The result is compared against some methods that used the same dataset and the analysis is shown in Table 5 and Figure 9. We found that the proposed hybrid approach (ADASYN + DNN) yielded improved prediction accuracy of 94.1% against all benchmark techniques. Thus, Figure 9 and Table 5 illustrates the comparison between the proposed hybrid approach (ADASYN + DNN) compared to other benchmark models. It was observed that the proposed approach yielded in 94.1% accuracy in predicting a loan defaulter which was relatively better than the other benchmark techniques.

Table 4. Output of evaluation metrics

Precision	Sensitivity (Recall)	Specificity	Accuracy	
0.972	0.960	0.823	0.941	

Table 5. Analysis of loan default predictions on different methods

Model	Prediction accuracy
Logistic Regression (Turiel, & Aste, 2019)	81.0
SVM (Turiel, & Aste, 2019)	69.7
Logistic Regression (Feis et al., 2016)	88.0
LDA (Feis et al., 2016)	92.1
Decision tree (CRT) (Jin, & Zhu, 2015)	71.2
Decision tree (CHAID) (Jin, & Zhu, 2015)	70.1
SVM in IBM SPSS modeler (Jin, & Zhu, 2015)	72.1
Linear SVM (Feis et al., 2016)	89.0
Classic RBF (Feis et al., 2016)	68.1
MLP with one hidden layer (Feis et al., 2016)	71.2
MLP with one hidden layer (Byanjankar, Heikkilä, & Mezei, 2015)	63.7
AdaBoost (Feis et al., 2016)	91.7
MLP with three hidden layers (Duan, 2019)	93.2
Proposed method (ADASYN + DNN)	94.1

Figure 9. Proposed approach against benchmark techniques



7. CONCLUSION

The study introduces a hybrid approach (ADASYN + DNN) for addressing the imbalanced data problem by oversampling the minority class (loan defaulters) which intend generates a relatively balanced class distribution. The approach then uses the DNN to learn from the balanced dataset to classify new instances into the loan default class or the non-default class. The ADASYN minority oversampling approach resulted in an improved, and less biased prediction results. From literature, the study identified 13 different techniques for loan default prediction. The proposed approach was benchmarked against these techniques it was observed that the introduced approach can predict loan defaulters with about 94% performance accuracy.

The result suggests that the models used are effective in increasing the accuracy of prediction of loan default, however, the study considers only the categories of fully paid and default loans, not taking into consideration risky loans. Thus, further studies need to be conducted in the latter category to assess its behavior.

REFERENCES

- Amin, A., Anwar, S., Adnan, A., Nawaz, M., Howard, N., Qadir, J., Hawalah, A., & Hussain, A. (2016). Comparing oversampling techniques to handle the class imbalance problem: A customer churn prediction case study. *IEEE Access : Practical Innovations, Open Solutions*, 4, 7940–7957. doi:10.1109/ACCESS.2016.2619719
- Bastani, K., Asgari, E., & Namavari, H. (2019). Wide and deep learning for peer-to-peer lending. *Expert Systems with Applications*, 134, 209–224. doi:10.1016/j.eswa.2019.05.042
- Bi, J., & Zhang, C. (2018). An empirical comparison on state-of-the-art multi-class imbalance learning algorithms and a new diversified ensemble learning scheme. *Knowledge-Based Systems*, 158, 81–93. doi:10.1016/j.knosys.2018.05.037
- Byanjankar, A., Heikkilä, M., & Mezei, J. (2015, December). Predicting credit risk in peer-to-peer lending: A neural network approach. In *2015 IEEE Symposium Series on Computational Intelligence* (pp. 719–725). IEEE. doi:10.1109/SSCI.2015.109
- Ceballes-Serrano, C. C., Garcia-Lopez, S., Jaramillo-Garzón, J. A., & Castellanos-Domínguez, G. (2012, September). A strategy for classifying imbalanced data sets based on particle swarm optimization. In *2012 XVII Symposium of Image, Signal Processing, and Artificial Vision (STSIVA)* (pp. 218–222). IEEE. doi:10.1109/STSIVA.2012.6340585
- Chawla, N. V., Lazarevic, A., Hall, L. O., & Bowyer, K. W. (2003, September). SMOTEBoost: Improving prediction of the minority class in boosting. In *European conference on principles of data mining and knowledge discovery* (pp. 107–119). Springer. doi:10.1007/978-3-540-39804-2_12
- Chen, Y. Q., Zhang, J., & Ng, W. W. (2018, July). Loan Default Prediction Using Diversified Sensitivity Undersampling. In *2018 International Conference on Machine Learning and Cybernetics (ICMLC)* (Vol. 1, pp. 240–245). IEEE. doi:10.1109/ICMLC.2018.8526936
- Cheng, D., Zhang, Y., Yang, F., Tu, Y., Niu, Z., & Zhang, L. (2019, November). A dynamic default prediction framework for networked-guarantee loans. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (pp. 2547–2555). doi:10.1145/3357384.3357804
- Duan, J. (2019). Financial system modeling using deep neural networks (DNNs) for effective risk assessment and prediction. *Journal of the Franklin Institute*, 356(8), 4716–4731. doi:10.1016/j.jfranklin.2019.01.046
- Feis, A., Mehta, A. V., Morris, S., Solitario, J., & de Graaf, C. V. (2016). P2P loan selection. *Stanford University Algorithmic and Big Financial Data Report*.
- Feng, Y., Fan, X., & Yoon, Y. (2015). Lenders and borrowers' strategies in online peer-to-peer lending market: an empirical analysis of ppdai. Com. *Journal of Electronic Commerce Research*, 16(3), 242.
- Jin, Y., & Zhu, Y. (2015, April). A data-driven approach to predict default risk of loan for online peer-to-peer (P2P) lending. In *2015 Fifth International Conference on Communication Systems and Network Technologies* (pp. 609–613). IEEE. doi:10.1109/CSNT.2015.25
- Kumar, N., Gaidhane, V. H., & Mittal, R. K. (2020). Cloud-based electricity consumption analysis using neural network. *International Journal of Computer Applications in Technology*, 62(1), 45–56. doi:10.1504/IJCAT.2020.103917
- Lee, H., Gnanasambandam, N., Minhas, R., & Zhao, S. (2011, December). Dynamic loan service monitoring using segmented hidden markov models. In *2011 IEEE 11th International Conference on Data Mining Workshops* (pp. 749–754). IEEE. doi:10.1109/ICDMW.2011.71
- Ma, L., Zhao, X., Zhou, Z., & Liu, Y. (2018). A new aspect on P2P online lending default prediction using meta-level phone usage data in China. *Decision Support Systems*, 111, 60–71. doi:10.1016/j.dss.2018.05.001
- Omarini, A. E. (2018). Peer-to-peer lending: business model analysis and the platform dilemma.
- Pears, R., Finlay, J., & Connor, A. M. (2014). Synthetic Minority over-sampling technique (SMOTE) for predicting software build outcomes. *arXiv:1407.2330*.
- Reddy, M. J., & Kavitha, B. (2010, February). Neural Networks for Prediction of Loan Default Using Attribute Relevance Analysis. In *2010 International Conference on Signal Acquisition and Processing* (pp. 274–277). IEEE. doi:10.1109/ICSAP.2010.10
- Serrano-Cinca, C., & Gutiérrez-Nieto, B. (2016). The use of profit scoring as an alternative to credit scoring systems in peer-to-peer (P2P) lending. *Decision Support Systems*, 89, 113–122. doi:10.1016/j.dss.2016.06.014
- Turiel, J. D., & Aste, T. (2019). P2P Loan acceptance and default prediction with Artificial Intelligence. *arXiv:1907.01800*.
- Vueyans, S. (2019). *Dealing with imbalanced and weakly labelled data in machine learning using fuzzy and rough set methods*. Springer International Publishing. doi:10.1007/978-3-030-04663-7
- Wang, H., Pan, T., & Ahsan, M. K. (2020). Hand-drawn electronic component recognition using deep learning algorithm. *International Journal of Computer Applications in Technology*, 62(1), 13–19. doi:10.1504/IJCAT.2020.103905

ENDNOTES

- ¹ <https://www.kaggle.com/datasets/adarshsng/lending-club-loan-data-csv>