International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A New Machine Learning Model for Detecting levels of Tax Evasion Based on Hybrid Neural Network

Abeer Shujaaddeen[1*], Prof. Fadl Mutaher Ba-Alwi[2], Prof. Ghaleb Al-Gaphari[3]

Submitted: 05/11/2023 **Revised**: 24/12/2023 **Accepted**: 02/01/2024

Abstract*:* Tax fraud is a general term that refers to the efforts of organizations or individuals to legally defraud, such as concealing the true status of the taxpayer to the tax authorities so that the value of the tax is reduced and included. The submission of false tax reports, such as declaring earnings that are undervalued. In other words, tax fraud is the lying on a tax return form to reduce tax liability. Therefore, detecting tax fraud is one of the main priorities of the tax authorities. Most of the recent works and modern business in tax fraud detection in many countries around the world rely on machine learning techniques that make use of labelled data. unsupervised learning has main advantage, that enabled this techniques e to deal with problems like detecting Tax Froud, which can be considered as a challenge in terms of decision.Also supervised learning is capable of classify the data. Therefore, this research paper aims to detect tax fraud and determine its level by building an optimal machine learning model by a new hybrid neural network technique that depend on two type of learning unsupervised learning and supervised learning for detecting tax fraud and determine its level. The proposed model is validated based on available machine learning techniques, it outperforms previous techniques in term of effort, computational time and cost reduction. The datasets used for validation and verification of the proposed model is given from the Tax Authority of Yemen. It consists of 1083 attributes.

Keyword: *Machine Learning, Unsupervised Learning, Supervised Learning, Hybrid Neural Network, Tax, Tax Fraud, Dataset.*

1. Introduction

Machine learning is a field within artificial intelligence (AI) and computer science. It uses algorithms and data to simulate human learning processes and enhance its accuracy over time [1]. In machine learning, there are three main types of algorithms: supervised, unsupervised, and reinforcement learning. Supervised learning algorithms aim to uncover the connection between input and target attributes, resulting in a model that represents this relationship. These models provide a broader understanding of phenomena not readily apparent in the dataset and can be beneficial for predicting the value of the dependent variable when the independent variable's value is known [2]. Machine learning, also known as inductive learning or classifier building, involves the process of acquiring a set of rules from examples or instances in a training set. This enables the creation of a classifier that can generalize from new instances [3]. On the other hand, unsupervised learning entails discovering interesting properties within a given set of instances without explicit guidance [5]. The objective of unsupervised learning is to learn how systems can identify specific input patterns that replicate the statistical structure of the entire set of input patterns. Unlike supervised or reinforcement learning, unsupervised learning lacks clear target outputs or environmental evaluations related to all inputs. Instead, it relies on inherent biases and is more akin to the structure of the human brain, making it significant in its own right. The primary focus of unsupervised learning methods is the observed input patterns, represented as xi [4]. Reinforcement learning, on the other hand, is a systematic learning process that utilizes previously collected data during specific application procedures. It processes and organizes feedback information from a specific part to establish a closed loop for data processing. Reinforcement learning generally involves expanding data collection through dynamic learning and statistical methods and is primarily employed to solve control problems in robotics. Representative learning algorithms in reinforcement learning include the Q-learning algorithm and the Temporal Difference learning algorithm [5]. In terms of machine learning types, supervised learning encompasses classification and regression tasks, unsupervised learning includes clustering, association, anomaly detection, and dimensionality reduction, and reinforcement learning stands as its own category as shown in figure1.

1 Sana'a University, Yemen
ORCID ID: 0009-0004-4993-3132
2 Sana'a University, Yemen
3 Sana'a University, Yemen
ORCID ID: 000-0002-6007-7638
** Corresponding Author Email: Abeer41036@gmail.com*

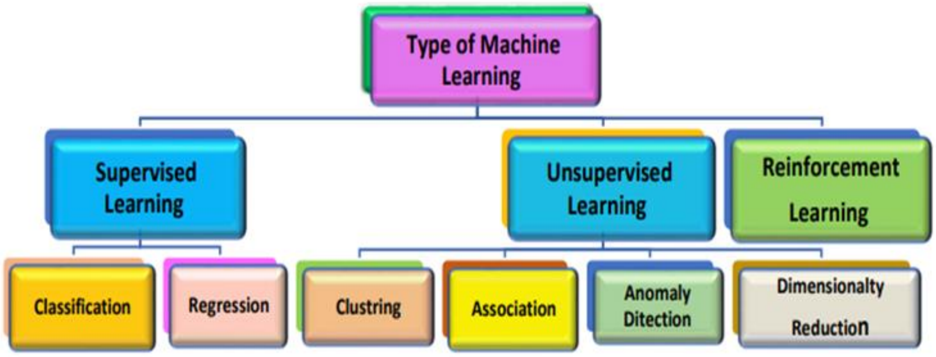

Fig. 1 Type of Machine Learning

The following are some of the techniques used in machine learning: - Decision Trees and Rules - Logistic regression - Bayesian nets / Naive Bayes - Random Forest (RF) – Multi Layer Neuronal networks - Nearest neighbour – The genetic algorithms depicted in the illustration. ML has a wide range of uses, including the following: Flexible web pages computing that is emotional Bioinformatics Brain-computer interfaces- Cheminformatics: DNA sequence classification - Computer-based marketing Finance computation - Object recognition using computer vision - Detection of credit card fraud Playing video games and retrieving information - Monitoring for online fraud - Artificial perception - Medical evaluation Processing of natural language - and optimization-ta heuristic - Advisory systems Robotic movement and search engines - Sentiment (or opinion) analysis - Software engineering - Sequence mining - Recognition of speech and handwriting - Stock market research Monitoring the condition of structures - Tax fraud detection and syntactic pattern recognition[1] .

One of the most significant objectives of tax authorities in many nations is to identify and uncover tax fraud. They made advantage of machine learning methods. These methods eliminate the need for the traditional methods of fraud detection, saving time, money, and effort. We aimed to create a brand-new machine learning model in this research. The Tax Authority in Yemen then used it to detect and gauge the amount of tax fraud. Along with tax authorities that face comparable issues, several financial corporations and banks could profit from this study.

2. Methodology

The full methodology of the proposed study is to build a new machine learning model based on a new hybrid neural network that used two types of learning to detect tax evasion and determine the level of tax evasion on the datasets provided by the Tax Authority of Yemen is shown in Figure2.

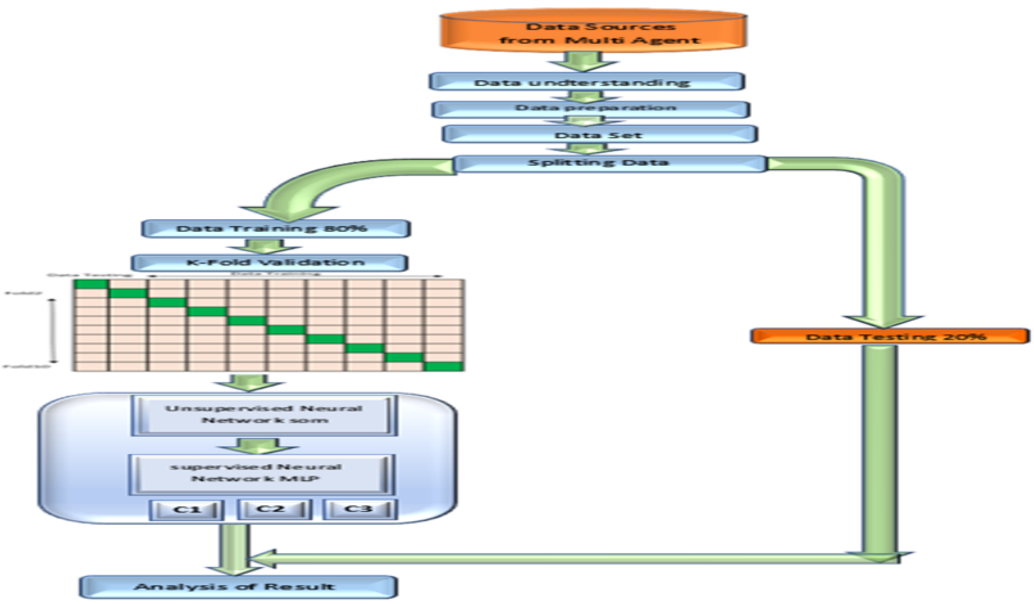

Fig 2: The Proposed Model

2.1 Data collection:

Data collection is the most important part of the research. We collected data from the Tax Authority of Yemen. We have taken one type of tax, which is the commercial and industrial profits tax because this type of tax is the backbone of the rest of the other types of taxes. The data were described and the meanings of the fields were clarified and the extent of their impact on other variables from the tax accounting side as shown in table1.

Table 1: The main Column in profits tax

No	Name of variable	The description	Type
1	TIN	Tax Number	Integer/number
2	TN	Trade Name	Var Char
3	LE	legal entity	Var Char
4	TP	Tax period	Integer/number
5	T_type	Tax type	Var Char
6	BN	Business Number	Integer/number
7	Tax	Tax	Integer/number
8	Punder	Payable under account	Integer/number
9	Dtax	Due tax	Integer/number
10	Fine	Fines	Integer/number
11	Damount	Deserved amount	Integer/number
12	Tax_L div BN_L	Tax rate to turnover	Integer/number
13	Tax_C div BN_C	Tax rate to turnover	Integer/number
14	Ratio_C	Tax rate to turnover for the previous year	Integer/number

We maintained the necessary features and eliminated unnecessary features that do not help in the decision-making process from the training dataset. This step is very important in reducing the dimensions of the input, which reduces the execution time and increases the prediction accuracy, and eliminates confusion from the data in the case of adding unnecessary features and variables. In our study, we had 14 column and we reduced the dimensions to reach 6 columns as shown in the table2.

Table 2: Features Selection

No	Name of variable	The description	Type
1	TIN	Tax Number	Integer/number
2	BN	Business Number	Integer/number
3	TAX	Tax	Integer/number
4	Paid under	Payable under account	Integer/number
5	Fine	Legal Fine	Integer/number
6	Ratio_C	Tax rate to turnover for the previous year	5Integer/number

2.2 Data preparation

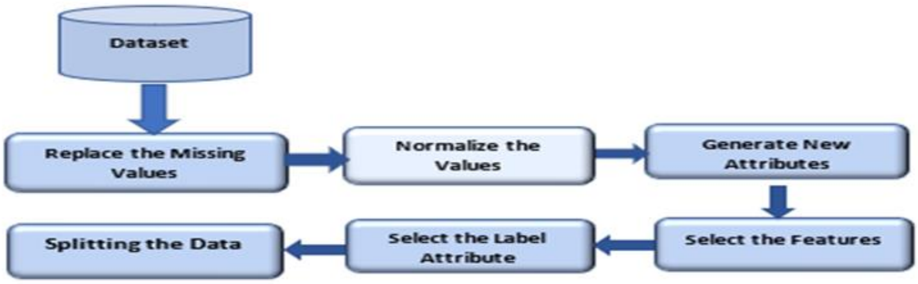

Fig 3: The Data Preparation Model

It is a set of operations used to modify the raw data as follow:

Data cleaning is very important because of their impact on the accuracy of classification, and the decision-making process is compromised when the data set is missing or incorrect Data cleaning in our data we replaced the missing data with zero. And we made the normalization method to let the values between (0,1), then we generated new attributes (computed attributes) for adoption as tax risk criteria

Table 3: Generated New Attributes

No	Name of Attributes	Function Expressions
1	Tin	
2	Finecode_C	if([Fine_C]>0,20,0)
3	Taxcode_C	if([Taxdue_C]<0,20,0)
4	BN_C*1	BN_C /100
5	Paid and BN_C	if([Paid Under_C]>0 && [BN_C]<=0,15,0)
6	Paid and Tax_C	if([Paid Under_C]>[Taxdue_C],15,0)
7	Tax_C div BN_C	[Taxdue_C] /[BN_Cu]
8	BN_Cu	if([BN_C] == 0,1, [BN_C])
9	Taxcode_L	if([Taxdue_L]<0,20,0)
10	Finecode_L	if(Fine_L>0,20,0)
11	BN_La	if([BN_L]==0,1,[BN_L])
12	Tax_L div BN_L	[Taxdue_L] /[BN7]
13	Ratio_C	if([Tax_C div BN_C]<[Tax_L div BN_L],10,0)
14	BN and Tax_C	if([BN_C*1]<[Taxdue_C],20,0)
15	Sum_C	)[BN and Tax_C] +[Paid and BN_C]+[taxcode_C]+[finecode_C]+[Paid and Tax_C]+[Ratio_C](
16	Class_C	if([sum_C]>=65,"F", if([sum_C]>=25,"PT", if([sum_C]>=10,"OT","E")))

Then we selected features (feature reduction) (to choose the appropriate features). for the second time to work with in model. As shown in table4.

Table 4: Select features for the second time

No	Attributes
1	BN and Tax_C
2	Paid and Bn18_C
3	Paid and Tax_C
4	Ratio_C
5	Class_C
6	Finecode_C
7	Sum_C
8	Taxcode_C

After cleaning the data, normalizing, generating new attributes, selecting features, and devising new features, we choose the attribute to be a label for the model.

The Actual data after processing

In our study we divided the behaviours of taxpayers into four categories bases on the level of evasion as the follow:

-Complete evasion

-Partial evasion

-Simple evasion

-Tax committed

Each of these cases has a specific treatment. Complete evasion is examined by comprehensive examination at the taxpayer's headquarters, and partial evasion is examined by partial examination at the taxpayer's headquarters, and simple evasion is examined by a desk examination at the tax administration, and the obligated taxpayer is not examined or visited, but may be give him a set of tax concessions. We made label for each case of treatment as follow: -

comprehensive examination= F

partial examination=PT

desk examination =OT

obligated taxes=E

After the pre-processing, we get the next chart as shown in Fig4.

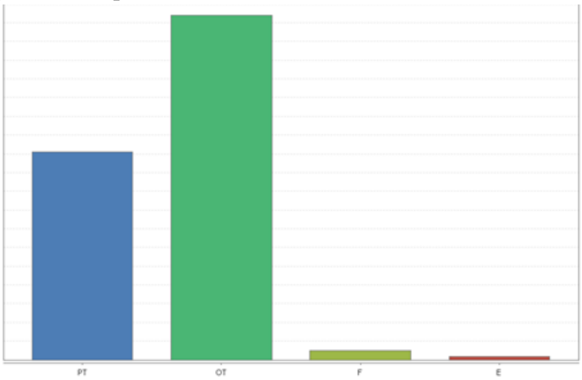

Fig 4: The Actual data after processing

Then it was the time for training and testing. The training data is used to train the model, and the dependent variable is known Test Data. The test data is used to make the predictions from the model that is already trained on the training data. The data was trained using the K-Fold Validation technique.

2.3. Splitting data

The data was divided into a training dataset of 80% and a testing dataset of 20%. The data was trained using the K-Fold Validation technique. We used this technique because our dataset is unbalanced, so this technique is the best to split data.

a. K-Fold Validation technique: -

Cross-validation is a resampling process used for evaluating machine learning models, on a data sample that is limited.

The process has a single parameter called k, which refers to the number of groups a given data sample is to be split into. This procedure is often called k-fold cross-validation. When choosing a specific value for k, it can be used instead of k, in the reference to the model, like k=10 becoming 10-fold cross-validation. Cross-validation is used in applied machine learning for estimating the skill of a machine learning model, on data that is unseen. That is, using a limited sample for estimating how the model is expected to perform in general. When used for making predictions on data, that is not used during the training of the model. It is a popular method, because it is simple to understand, and because it generally results in a less biased, or less optimistic estimate for the model skill than other methods. Such as a simple train/test split [6].

2.4. The algorithms that use for the Model2

a. Unsupervised neural network Self Organizing Map (SOM)

The Self Organizing Map is one of the most popular neural models. It belongs to the category of the competitive learning network. The SOM is based on unsupervised learning, which means that is no human intervention is needed during the training .It used for feature detection, clustering and dimension reduction[7]. The structure of SOMs is composed of two layers fully attached to each other: input layer and Kohonen layer. Kohonen layer is also the layer where the map is formed. Hidden layers are not in question as in prediction or classification studies. Neurons in the Kohonen layer are generally arranged two dimensionally. The number of neurons in the input layer is equal to the number of variables used. Each neuron in the input layer is connected to each neuron in Kohonen layer as feed forward [7].

SOM algorithm will consist of from inputs vectors and three neurons as three clusters.

We will put the training dataset into unsupervised neural network (SOM) to divided the dataset into clusters, this neural network based on the competitive learning, and consists of 2 layers as the following:

-Input layer.

- Output layer.

b- Supervised Neural Network (MLP).

The output of SOM will go to another algorithms. Each output will play as a new input to supervised neural network MLP.

BPN Algorithm

The algorithm for BPN is as classified int four major steps as follows:

1. Initialization of Bias, Weights.

2. Feedforward process.

3. Back Propagation of Errors.

4. Updating of weights & biases.

The Back-propagation (BP)/Generalized Delta Rule

The back-propagation (BP) algorithm is a generalization of the delta rule that works for networks with hidden layers. It is by far the most popular and most widely used learning algorithm by ANN researchers. Its popularity is due to its simplicity in design and implementation.

The Hyper parameters that use in the model are the follow

1- Number of Layers.

2-Number of Neurons.

3- Activation Functions.

4-learning rate.

For supervised NN we will control in all hyper parameters and we use two activation function as the follow

-ReLu function in hidden layers.

-Softmax function in output layer.

We chosed ReLu function because it has several advantages Since the ReLU does not compute exponentials or divisions, its main benefit is faster processing, which increases overall computation speed. Another property of the ReLU is that introduces sparsity in the hidden units as it squishes the values between zero to maximum [8].

ReLu function

Calculations in the rectified linear unit activation function are easily understandable. It depends on the output to activate the neurons; for example, if the output is less than zero, the neurons will be disconnected from the network.

$$\boldsymbol{f(x) = \max(0, x)} = \begin{cases} xi, & if\ xi \geq 0 \\ 0, & xi, if\ \ xi < 0 \end{cases} \quad (1)$$

Soft Max function

The Softmax is another AF that is used in neural computing. It is used for computing probability distribution from a vector of real numbers. It produces an output that is a range of values between 0 and 1, with the sum of probabilities equal to 1[9]. The SoftMax function is computed by

$$\boldsymbol{SofMax} = \frac{exp(z_i)}{\Sigma_j\, exp(z_j)} \qquad (2)$$

The SoftMax is used in multi classes models, that return the probabilities of each class, with the target class having the highest probability. It appears in all the output layers of the DL architectures. The main difference between the Softmax AF and Sigmoid is the Softmax is used for multivariate classification tasks while the Sigmoid is used in binary classification [9].

4-Larning Rate.

For unsupervised NN it doesn't have activations function but learning rate as a hyper parameter.

Self-organizing map is a bit different from standard ANNS. Each node in a low-dimensional map is given a weight that has the same dimensionality as the input data. This is how it operates. Then training the map by adjusting these weights to the input data, which eventually creates regions on the map dependent on the structure of the data.

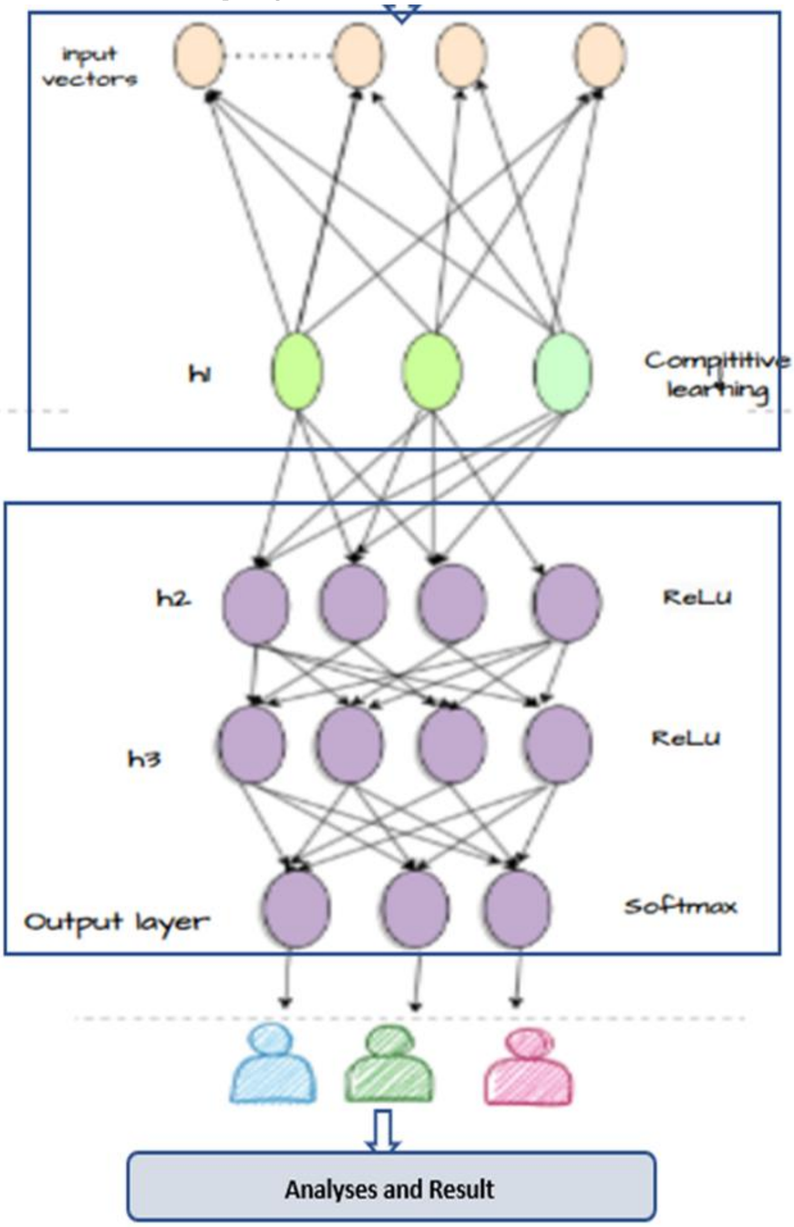

Fig 5: The Proposed Approach

These algorithms work as the following.

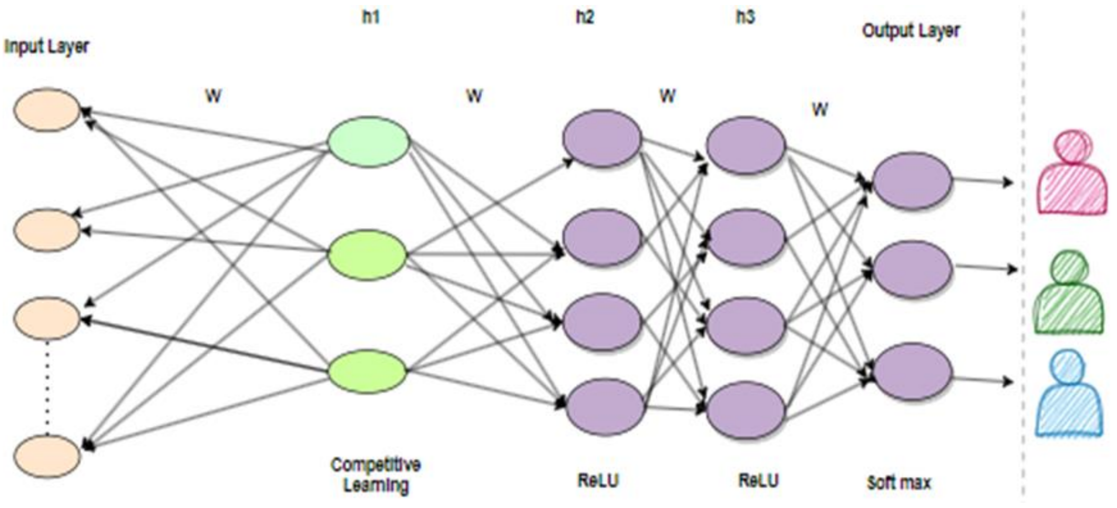

Fig 6: How the algorithm work

How Does the new approach work?

The input vectors will go to the first layer, in this layer the net will work in competitive learning as follow

Step 1: Initializing the Weights

There is no activation function in SOMs. Weights are a component of the nodes in this instance. The output node's weights in a SOM are its characteristic. Rather than being the sum of the weights, the output node in a SOM carries the weights as its coordinates. It bears these weights when it enters the input area. The initial values of the weights were selected at random; they are almost 0 but not quite

Step 2: Calculating the Best Matching Unit

Step 3: Estimating the neighbourhood's size around the BMU

After identifying the BMU, the next step in each iteration is to find out whether other nodes are nearby. The subsequent phase will modify the weight vectors of each of these nodes. Upon determining the radius of the neighbourhood, Pythagoras is utilized to promptly ascertain if every node is within the radial distance. An exponential decay function is responsible for the BMU's surrounding neighbourhood’s shrinkage. It gets smaller on each cycle until it is reduced to just the BMU.

$$\boldsymbol{\sigma(T) = \sigma_0\, exp\left(-\frac{t}{\lambda}\right)} \quad (\mathbf{4})$$

Step 4: Adjusting the Weights

The following equation modifies the weight vector of every node near the BMU, including the BMU:

Input Vector – Old Weights = New Weights = Old Weights + Learning Rate

Examining the dataset is the next step. We will determine which node in the dataset is most likely to be closest to each row. We shall retrieve the value of the first row for each of the three columns. Finding the output node that is closest to that row is the next step. To determine the best matching unit, one method is to iterate through all the nodes and compute the Euclidean distance between each node's weight vector and the current input vector. The node whose weight vector is closest to the input vector is recognized as the BMU. Given as follows is the Euclidean distance:

$$\textbf{Distance} = \sqrt{\sum(\boldsymbol{xi - wi})\mathbf{2}} \quad \textbf{(3)}$$

$$\mathbf{W(t+1) = W(t) + L(t)\big(V(t) — W(t)\big)} \quad (5)$$

The output of the first layer will go to the second layer. In the second hidden layer the net will work as follow

Step 1: Initially, one input node for each feature must be entered as the first observation from the first layer.

Step 2: Forward-Propagation: As the neurons fire, the weights restrict each neuron's activation's impact as it propagates from left to right. propagates the activations in order to achieve the desired result, y.

Step 3: Analyse how the actual and expected outcomes differ from one another. VIA the computation of the error.

Step 4: Back-Propagation: From left to right, the error is spreading backward. Changing the weights based on how much they contributed to the error. The learning rate determines how often we update the weights.

Step 5: Steps 1 through 5 should be repeated, updating the weights after each observation.

Step 6: The creation of an epoch occurs when the ANN has finished processing the whole training set, then going to the next epoch.

How the new model works

Initially, a self-organizing neural network will be trained to analyse data and find unexpected patterns. Based on these patterns, a multi-layered neural network will be trained to analyse the patterns more precisely and in-depth. Stated differently, the multi-layered neural network analyses and classifies the data depending on the patterns that the self-organizing neural network has identified.

Combining the two networks improves the accuracy of the analysis and increases the effectiveness of the work done in detecting and quantifying tax evasion.

What is the role of the new model play in detecting tax evasion?

1- Using a self-organizing neural network to analyse the data and find unexpected patterns, then using a multi-layered neural network to analyse the patterns that are found in a more thorough and accurate way.

2- Data classification more accurately, where the self-organizing neural network will be used to classify the unclassified data based on the patterns detected in it and the multi-layered neural network will be used to classify similar data in the same group or category,

3- Faster and more effective data analysis because the two neural networks split the job and may use their capacity for self-learning and data adaptation to analyse data more quickly and effectively.

4- Enhancing the quality of analysis and service, as deep analysis and artificial intelligence approaches are combined with the two neural networks to enhance the field of tax evasion detection analysis and service quality.

In general, the new model can be used in the field of tax evasion detection to improve the accuracy of analysis, quality of service, and increase work efficiency in this field.

3. Model Evaluation

The purpose of this task is to provide an explanation of the defined model's outcomes and evaluation parameters. A confusion matrix with values for true positives (correct classifications) and false positives (incorrect classifications) is used to compare the number of data points correctly and incorrectly classified. This evaluation parameter is used to compare the various classification models.

3.1. Evaluation and Performance Module After fitting

Using a variety of metrics, including accuracy, precision, recall, F-1 scores, and confusion matrix, we test each classifier's performance and integrity before adding it to the system model. Our performance evaluation activities have been conducted using these assessment matrices as their foundation.

3.2 Evaluation Metrics

When developing a machine learning model, it's critical to evaluate the model's quality using specific metrics. This is used to forecast the accuracy of the model for future data.

Confusion Matrix:

The confusion matrix is a matrix for evaluating the classification model's performance [10]. The amount of right and incorrect guesses is the key to the confusion matrix, which is summarized using count values and broken down by class. Although it is simple to comprehend, the parameter employed is perplexing.

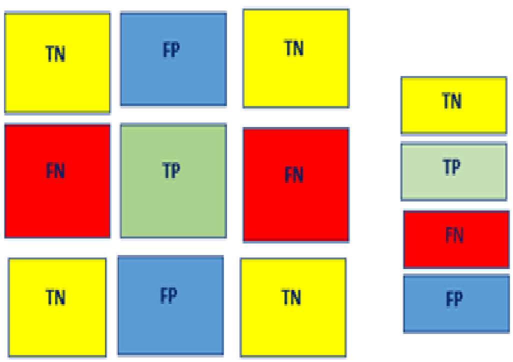

Fig 7: Confusion Matrix for unbalanced

The following are the conditions:

True Positive (TP): When the model classified the actual value and it is True Negative (TN): When the model classified the actual value and it is negative.

False Positive (FP): When the classifier predicts the news is true but actually its false.

False Negative (FN): When the classifier predicts the news is false but actual its true.

Precision Metrics: Precision metrics tell us how many of the correctly predicted cases turned out to be positive. These metrics determine whether the model is reliable or not [11].

$$\boldsymbol{Precision} = \frac{TP}{TP+FP} \qquad (6)$$

Recall Metrics: Recall metrics shows the number of really positive cases that could be predicted correctly using the model [11].

$$\boldsymbol{Recall} = \frac{TP}{TP+FN} \qquad (7)$$

D. F-Score

F-measure (F1 score) is defined as the mean of precision and recall. It is a measure combines accuracy and recall into a single performance measure. Averaging recall and accuracy yielded the F1-score. Precision and recall contribute equally to the F1-score[12].

$$\boldsymbol{Fscore} = \frac{2*(Precision*Recall)}{Precision+Recall} \qquad (8)$$

F1 Score: F1 gives us the combined idea about precision and Recall metrics.

That means when we try to upgrade the value of precision Recall goes down and vice-versa.

Accuracy: Accuracy is the fraction of correct predictions and total predictions made by the classifiers [12]

$$\boldsymbol{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \qquad (9)$$

We have noticed that our data is unbalanced, so the accuracy measure will not help us much, and the Recall measure is concerned with detecting each level of tax evasion separately,

and this is important, but it may include some taxpayers who are tax-compliant or whose level of tax evasion is from a certain category If this happens, this will lead to the dissatisfaction of the tax-obligated group or the category whose level of evasion is lower than the other, and thus will lead to a lack of satisfaction and transparency between both the tax administration and the public of taxpayers. On the other side, if the Precision measure is concerned with identifying taxpayers, then this will lead to identifying taxpayers more efficiently, but this measure may include some tax evaders from other categories, and this will lead to increased tax evasion and dissatisfaction by the tax administration

Therefore, we care about F-score measure, because it is concerned with identifying categories of tax evaders as well as taxpayers. Therefore, we will try to raise the efficiency of this criterion, and we have noticed through the experiments that some models that use the technology of multiple neural networks with a data set of Taxes had a low F-score level, so we built a new model that raises the F-score level in order to achieve tax auditing and determine the level of tax compliance in a better and more accurate manner.

4.Experments

We tested our proposed model on the data provided by the Tax Authority of Yemen, which numbered 1,083 attributes, after preprocessing the data to be suitable for use in building the model.

We divided the data into training dataset 80% and testing dataset 20%, then training the model as follows. Use the k_fold validation technique to divide the training data into 10 folds, then use the proposed technique.

The technology depends on controlling the higher parameters of each of the first hidden layer that uses competitive learning and the other hidden layers that use activation functions.

In the end, we chosed the network that gave greater accuracy and a higher score in the F1-score criterion, because what we care about here is getting a high F-score because our data is unbalanced data.

Experment 1:

When we used SOM The hyper parameters and performance as follow

Table5: SOM result

		2DM
SOM	Net Size	3
	Learning Rate	0.6
	Learning Rate end	0.01
	Radius start	1
	Radius end	1
	performance	76000

Experment2:

When we used different techniques with different hyper parameter the training result show as follow

Table 6: The result of training different techniques

	NB	RF	DT	SVM	KNN	MLP
Accuracy	99.72	96.54	97.34	94.11	99.65	97.33
Recall	99.905	59.96	66.67	33.29	99.88	66.63
Precision	99.48	98.82	65.75	32.27	96.67	64.56
F1-score	99.69	74.63	66.20	32.77	98.24	65.57

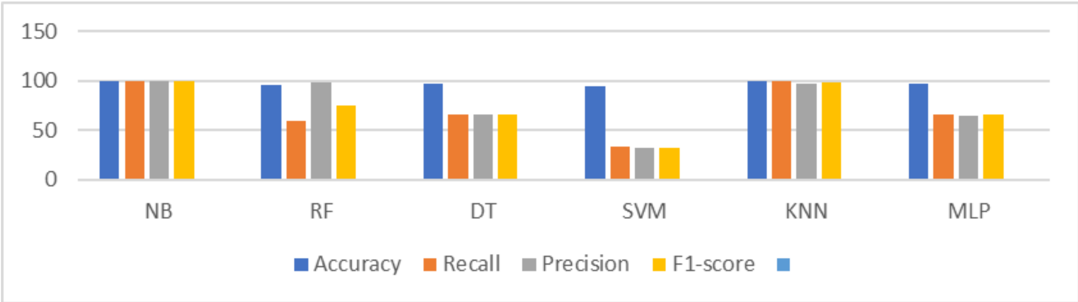

Fig 8: Confusion Metrix Chart for Training

The testing result show as follow:

Table 7 The result of testing different techniques

	NB	RF	DT	SVM	KNN	MLP
Accuracy	100	95.35	98.14	95.35	98.14	95.35
Recall	100	33.33	66.67	33.33	66.67	33.33
Precision	100	31.78	66.03	31.78	66.03	31.78
F1-score		32.53	66.34	32.53	66.34	32.53

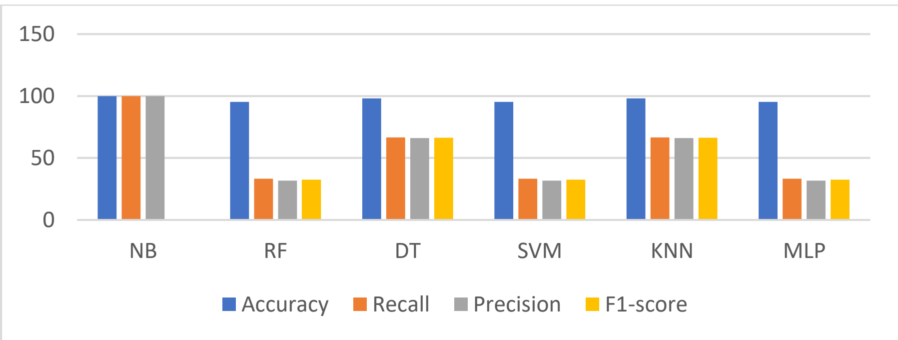

Fig 9: Confusion Metrix Chart for Testing

Experment3:

When we used MLP with multi different hyper parameter the training result and testing result show as the follow

Mlp1

Table 8 The result of MLP1

MLP	No_Hidden layer	2	Performance Measures		Training	Testing
	No_Neoron	20		Accuracy	97.23	95.35
	Activation function for hidden layer	sigmoid		Recall	66.63	33.33
	Activation function for hidden layer	sigmoid		Precision	64.56	31.78
	Training round	200		F1-score	65.57	32.53

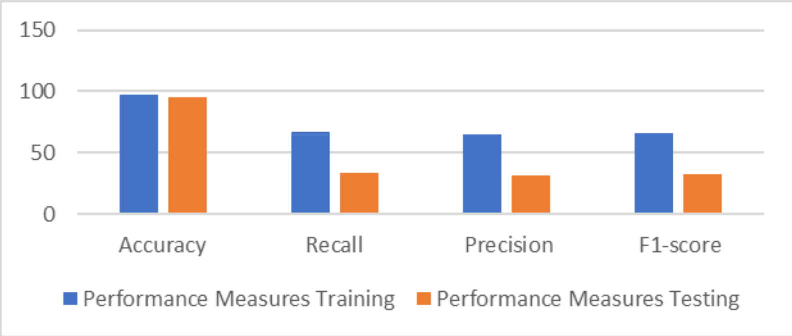

Fig10: Confusion Metrix Chart for Training and Testing MLP

MLP2

Table 9 The result of MLP2

:MLP	No_Hidden layer	2	Performance Measures		Training	Testing
	No_Neoron	20		Accuracy	97.34	95.35
	Activation function for hidden layer	sigmoid		Recall	66.67	33.33
	Activation function for hidden layer	sigmoid		Precision	65.75	31.78
	Training round	300		F1-score	66.20	32.53

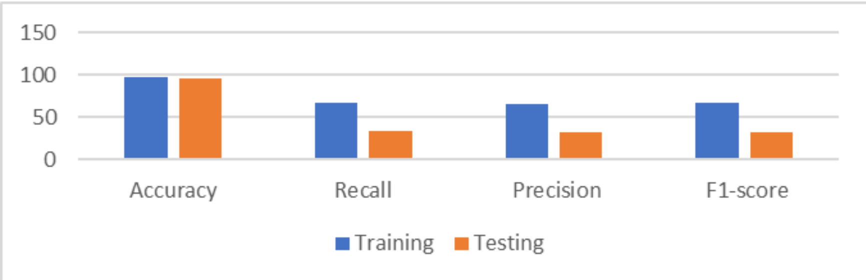

Fig1 1: Confusion Metrix Chart for Training and Testing MLP

MLP3

Table 10: The result of MLP3

MLP			Performance Measures		Training	Testing
	No_Hidden layer	2				
	No_Neoron	20		Accuracy	99.77	98.14
	Activation function for hidden layer	Relu		Recall	99.72	66.67
	Activation function for output layer	SoftMax		Precision	97.42	66.03
				F1-score	97.56	66.34

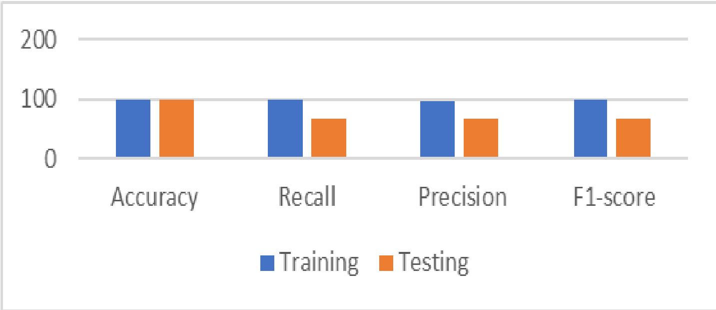

Fig 12: Confusion Metrix Chart for Training and Testing MLP

MLP4

Table 11: The result of MLP4

MLP			Performance Measures		Training	Testing
	No_Hidden layer	2				
	No_Neoron	10		Accuracy	99.88	95.35
	Activation function for hidden layer	Relu		Recall	89.96	33.33
	Activation function for output layer	softmax		Precision	89.44	31.78
	Training round	10		F1-score	89.69	32.53

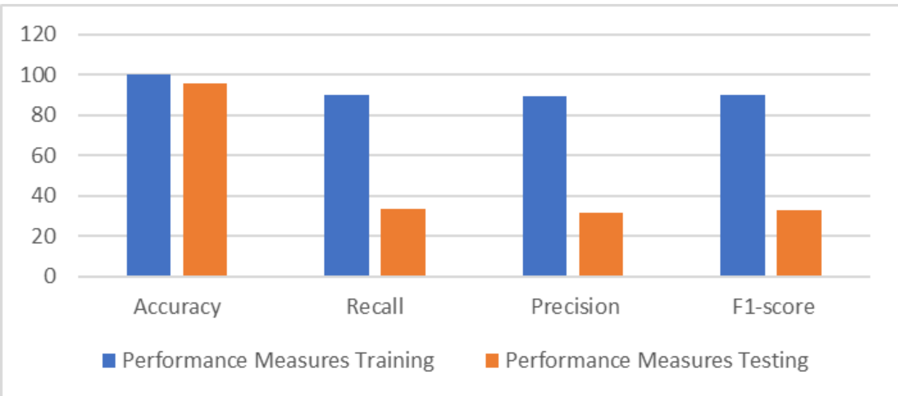

Fig 13: Confusion Metrix Chart for Training and Testing MLP

Experment4:

When we used the new techniques with multi different hyper parameter training and testing result show as in the follow

SMLP1

Table 12: The result of

SOM		2DM	MLP	No_Hidden layer	2	Performance Measures		Training	Testing
	Net Size	10		No_Neoron	40		Accuracy	99.65	99.07
	Learning Rate	0.8		Activation function for hidden layer	ReLU		Recall	99.88	99.67
	Learning Rate end	0.01		Activation function for hidden layer	Soft Max		Precision	96.67	88.89
	Radius start	10		epoch	10		F1-score	98.24	93.97
	Radius end	1							
	Training round	30							

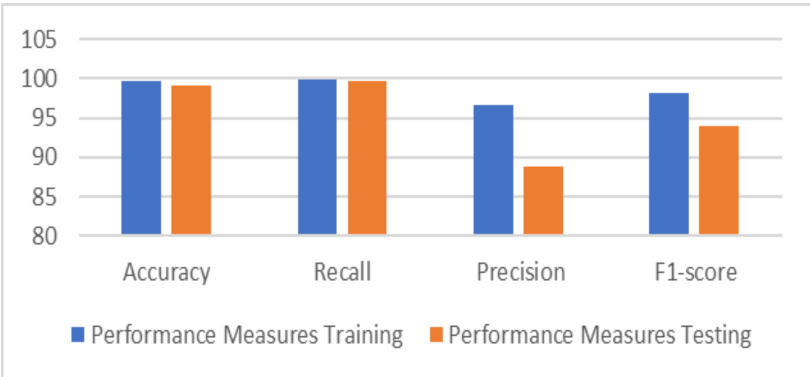

Fig 14: Confusion Metrix Chart for Training and Testing SMLP

SMLP2

Table 13: The result of Somdeep2

SOM		2DM	MLP	No_Hidden layer	2	Performance Measures		Training	Testing
	Net Size	10		No_Neoron	20		Accuracy	99.65	98.14
	Learning Rate	0.8		Activation function for hidden layer	ReLU		Recall	99.88	66.67
	Learning Rate end	0.01		Activation function for hidden layer	Soft Max		Precision	96.67	66.03
	Radius start	10		epoch	20		F1-score	98.24	66.34
	Radius end	1							
	Training round	30							

Fig 15: Confusion Metrix Chart for Training and Testing SMLP

SMLP3

Table 14: The result of Somdeep3

SOM		2DM	MLP	No_Hidden layer	2	Performance Measures		Training	Testing
	Net Size	3		No_Neoron	40		Accuracy	96.77	99.07
	Learning Rate	0.6		Activation function for hidden layer	ReLU		Recall	66.62	99.67
	Learning Rate end	0.01		Activation function for hidden layer	Soft Max		Precision	64.21	88.89
	Radius start	1		epoch	20		F1-score	65.39	93.97
	Radius end	1							
	Training round	30							

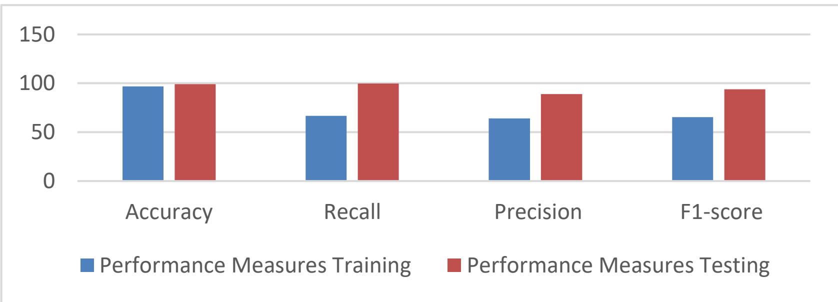

Fig 16: Confusion Metrix Chart for Training and Testing SMLP

SMLP4

Table 15: The result of Somdeep4

SOM		2DM	MLP	No_Hidden layer	2	Performance Measures		Training	Testing
	Net Size	5		No_Neoron	20		Accuracy	99.54	95.81

	Learning Rate	0.6		Activation function for hidden layer	sigmoid		Recall	99.84	66.18
	Learning Rate end	0.01		Activation function for hidden layer	sigmoid		Precision	95.70	51.42
	Radius start	1		epoch	300		F1-score	97.72	57.87
	Radius end	1							
	Training round	30							

Fig 17: Confusion Metrix Chart for Training and Testing SMLP

SMLP5

Table 16: The result of Somdeep5

SOM		2DM	MLP	No_Hidden layer	2	Performance Measures		Training	Testing
	Net Size	5		No_Neoron	20		Accuracy	99.19	95.35
	Learning Rate	0.6		Activation function for hidden layer	sigmoid		Recall	99.71	33.33
	Learning Rate end	0.01		Activation function for hidden layer	sigmoid		Precision	93.14	31.78
	Radius start	1		epoch	200		F1-score	96.31	32.53
	Radius end	1							
	Training round	30							

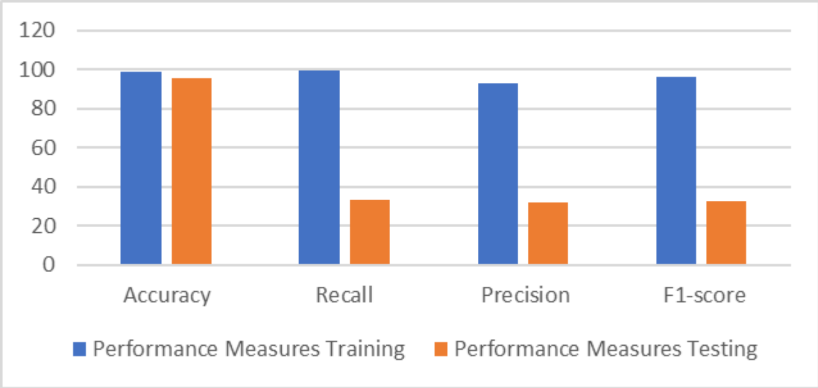

Fig 18: Confusion Metrix Chart for Training and Testing SMLP

5. Discussion the Results

We noticed from previous experiments that the hyperparameter control process has a significant impact on network formation and selection of the appropriate network, and we noticed that our proposed model achieved better results than machine learning models and techniques as follows (RF, DT, SVM, KNN, MLP.etc) for which we used the same data set. We also noticed that the difference in the hyperparameter control in the proposed model gives different results as shown in the table 17. We noticed through the table that exp 1,2 gave the best result with the same score for F_Score and accuracy then exp4 in training the model. On the other hand, when we tested the models, we noticed that the exp1 and exp3 gave the best and the same result in accuracy and f-score as shown in table 18. We care about f-score for the reasons that we mentioned, so we will take the proposed model with any hyperparameters either in exp1 or exp3.

Table 17: The summaries of training proposed model

	SOM						MLP					Performance measures			
	NS	LR	LRE	RS	RE	TR	N_H	N_N	AFH1	AFH2	TC	AC	RE	PR	F1_S
1	10	0.08	0.01	10	1	30	2	40	ReLU	SoftMax	10	99.65	99.88	96.67	98.24
2	10	0.08	0.01	10	1	300	2	20	ReLU	SoftMax	20	99.65	99.88	96.67	98.24
3	3	0.06	0.01	1	1	30	2	40	ReLU	SoftMax	20	96.77	66.62	64.21	65.39
4	5	0.6	0.01	1	1	30	2	20	sigmoid	sigmoid	300	99.54	99.84	95.70	97.72
5	5	0.6	0.01	1	1	30	2	10	sigmoid	sigmoid	200	99.19	99.71	93.14	96.31

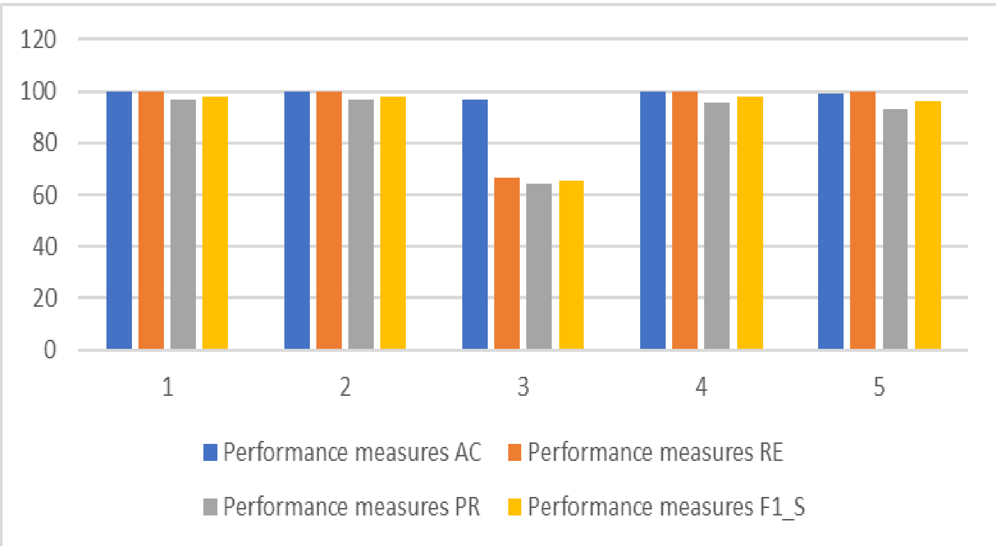

Fig 19: Confusion Metrix Chart for Training SMLP

Table 18: The summaries of testing proposed model

	AC	Recall	Precision	F-Score
1	99.07	99.67	88.89	93.97
2	98.14	66.67	66.03	66.34
3	99.07	99.67	88.89	93.97
4	95.81	66.18	51.42	57.87
5	95.35	33.33	31.78	32.53

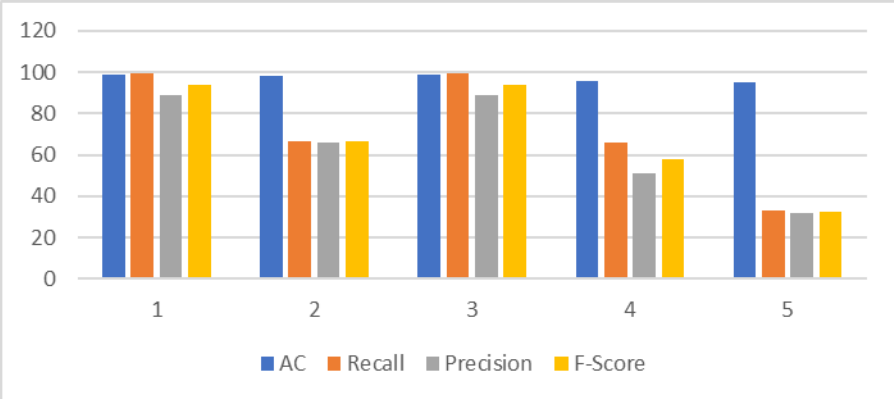

Fig 20: Confusion Metrix Chart for Testing SMLP

6.Conclusion and future work

Due to the importance of taxes in raising the level of the country and developing its infrastructure, which is considered one of the largest sources for both countries, including ours. In view of what countries suffer from tax evasion attempts at different levels, we have built a new model based on new technology in order to try to identify and detect levels of tax evasion. We analysed the results and took the f_ score criterion as the best and most important criterion. Because our data is unbalanced, we care about the accuracy standard in general and the F_score standard in particular. We really care a lot about the standard of accuracy, recall, or precision, because if the criterion of recall is high, this will lead to identifying, for example, the most evasive category. And if the standard of precision is high, then this leads to focusing on taxpayers, for example, from a less evading category, regardless of whether there is a group of taxpayers who are more evading, which leads to increased tax evasion and dissatisfaction with the tax administration. Here, F_ score criterion is very important because it performs the work of balancing and achieving satisfaction between both the tax administration and the public of taxpayers.

References

[1] P. Dönmez, “Introduction to Machine Learning, 2nd ed., by Ethem Alpaydın. Cambridge, MA: The MIT Press2010. ISBN: 978-0-262-01243-0. $54/£ 39.95 + 584 pages.,” Nat. Lang. Eng., vol.19, no.2, pp.285–288,2013,doi: 10.1017/s1351324912000290

[2] Li Kanghua, Jiang Shan. “Machine Learning and Cultural Production Reform Based on the Perspective of the Development of AI Technology [J]”. Journal of Xiangtan University (Philosophy and Social Sciences), 2020, 44 (01): 74-79.

[3] Jiang Na, Yang Haiyan, Gu Qingtuan, Huang Jiya. “Machine learning and its algorithm and development analysis [J]”. Information and Computer Science (Theoretical Edition), 2019 (01): 83-84 + 87.

[4] D. Mariam and R. Irakli, “Hybrid of Supervised and Unsupervised Learning Algorithms for Human Activity Recognition,” Trans. Georg. Tech. Univ. Autom. Control Syst., vol. 1, no. 25, pp. 31–40, 2018.

[5] S. V. Mahadevkar et al., “A Review on Machine Learning Styles in Computer Vision - Techniques and Future Directions,” IEEE Access, vol. 10. Institute of Electrical and Electronics Engineers Inc, pp.107293–107329, 2022.doi:10.1109/ACCESS.2022.3209825.

[6] A. N. Tarekegn, K. Michalak, and M. Giacobini, “Cross-Validation Approach to Evaluate Clustering Algorithms: An Experimental Study Using Multi-Label Datasets”, SN Compute. Sci, vol.1, no.5, pp.1–9,2020, doi:10.1007/s42979-020-00283-

[7] M. Bata, R. Carriveau, and D. S.-K. Ting, “Short-term water demand forecasting using hybrid

supervised and unsupervised machine learning model," Smart Water, vol. 5, no.1,2020, doi:10.1186/s40713-020-00020- y.

[8] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, "Activation Functions: Comparison of trends in Practice and Research for Deep Learning," no. December 2020, 2018, [Online]. Available: http://arxiv.org/abs/1811.03378

[9] H. Dahrouj et al., "An Overview of Machine Learning-Based Techniques for Solving Optimization Problems in Communications and Signal Processing," IEEE Access, vol. 9, pp. 74908–74938, 2021, doi: 10.1109/ACCESS.2021.3079639.

[10] A. Tasnim, M. Saiduzzaman, M. A. Rahman, J. Akhter, and A. S. M. M. Rahaman, "Performance Evaluation of Multiple Classifiers for Predicting Fake News," J. Comput. Common., vol.10, no.09, pp.1–21,2022, doi: 10.4236/jcc.2022.109001.

[11] Brownlee, J. (2020, January 3) How to Calculate Precision, Recall, and F-Measure for Imbalanced Classification. https://machinelearningmastery.com/precision-recall-and-f-measure-for-imbalance

[12] A. Tasnim, M. Saiduzzaman, M. A. Rahman, J. Akhter, and A. S. M. M. Rahaman, "Performance Evaluation of Multiple Classifiers for Predicting Fake News",J.Comput. Common., vol.10, no.09, pp.1–21,2022, doi: 10.4236/jcc.2022.109001.