The International Journal Of Engineering And Science (IJES)
|| Volume || 3 || Issue || 4 || Pages || 31-40 || 2014 ||
ISSN (e): 2319 – 1813 ISSN (p): 2319 – 1805

Students' Admission placement in Nigerian Universities Using Artificial Neural Networks

[1]Mikailu Habila, [2]G. M. Wajiga, [1]P. B. Zirra
[1]Department of Computer Science, Adamawa State University, Mubi
[2]Department of Computer Science, Modibo Adama University of Technology, Yola

ABSTRACT

The study was carried out regarding University students' admission placement using artificial neural networks needed in the University. The objective was to automate the university screening and placement process. The back propagation algorithm technique was used to develop this system because of its ability to minimize error and maximize accuracy. The placement of candidates into Departments is based on candidate's choice when requirements are met; otherwise the system assigns an alternative course for the candidate. Each candidate must have at least five relevant O' level credits including Mathematics and English language, to qualify for the chosen course. Candidates' O' level grades are strictly used for the placement process, having met the UTME and other entry requirements. A visual C# compiler was used to develop the framework for this work. The result ensures even distribution of students into various courses of the universities. The result is very effective and efficient in admission screening and placement processes.

KEYWORDS - *UTME, input layer, Students Placement, and Nigerian Universities.*

Date of Submission: 20 March 2014 Date of Publication: 15 April 2014

I. BACKGROUND OF THE STUDY

Joint Admission and Matriculation Board (JAMB) and the University are saddled with the responsibility of admitting students into university courses each year. In this process students are allocated courses of their choice on meeting the requirements, or be assigned another course in which they have the requirement. Minimum UTME points requirements exist for each university, and only students having the prescribed points and grades in specific subjects are eligible to join a particular course of the university.

Predicting student outcomes is really a process of trying to determine what group an individual student belongs. Should the student be placed into English 1A or a developmental English course? Will the student be more likely to drop out or be put on probation due to poor academic performance during their first semester? Reliable answers to these questions, and others like them, could help colleges tailor services and interventions to target populations and thereby utilize their limited resources more efficiently [7].

According to [1], Master of Business Administration (MBA) program admission directors are carefully reexamining the admission criteria they use to admit students into their graduate programs. This is in response to the bleak enrollment prospects facing business schools across the country.

This research discusses the applicability of artificial neural networks in the process of placing students into any suitable course of the university. The research intends to reveal that artificial neural networks using Back Propagation Technique can be used to place and identify the right university course for students within or outside their chosen options.

According to [5], University placement is usually extensive because of the large number of students compared to the very limited number of slots at the institution. Further, this is made complex by the fact that the selections are done based on student choice and the available courses of the university. On the other hand, the university may be overwhelmed by the number of students demanding access against limited university spaces. This process is prone to bias, errors, or favour, leading to disadvantaging innocent students, and in the long run, poor performance, drop outs, and reduced university standard are the consequences.

The problem with the current system is that; candidates are often placed into courses or departments they have not qualified for; on the other hand some are denied their chosen course even though they are qualified. Furthermore, some students forged result to gain admission into the university. As a result of this, the university has to go back rechecking the validity of students' results. Students who are found to forged result are expelled immediately, which is cumbersome and embarrassing. This leads to poor performance and reduced completion rate.

This paper is focused on placing candidates into the department in which they applied for whenever a requirement is made. On the other hand assign an alternative course for those who fail to qualify for their own chosen course. It will further handle the screening processing prior to placement to ensure candidates have the prescribed requirement and also a valid one, in particular the UTME result.

The remaining part of this research is divided as: part 2. Review of related works, Part 3 System design, Part 4 Results and discussion, and part 5. Summary, conclusion and future work.

II. MATERIAL AND METHODS

II.I Prediction of Student Academic Performance

Prediction of student academic performance by an application of data mining techniques, by [9] the aims of the study was to apply the kernel method as data mining techniques to analyze the relationships between students' behaviour and their success and to develop the model of student performance predictors.
This is done by using smooth support vector machine (SSVM) classification and kernel k-means clustering techniques**.**

II.II Placement Chance Prediction

[4] proposed a generalized design framework for a placement chance prediction data mining problem. Here history data is collected and data cleaning is done. After that attribute dependency analysis using chi square statistical analysis is done to clearly understand which all attributes have importance in deciding the placement chance. Once this is properly understood attribute reduction and porting to model dependent data formats takes place. In this work it has been proved that the technology named data mining can be very effectively applied to the domain called employment prediction, which helps the students to choose a good branch that may fetch them placement.

II.III NEURONS

A neuron is the building block of a neural network that consists of input links (from other neurons), an input function, an activation function, output, and output links (to other neurons). Input links provide activation to the neuron from other neurons. Each input link is assigned a weight which determines both the activation strength of connection and the sign. The stronger the activation strength the more likely it will be for the option to be connected to for further processing. A weighted sum is calculated for each neuron, i, and then the weighted sum of its entire input links, which are then used in the activation function in order to derive its output, denoted by ai [5]. The activation function is designed such that it meets two conditions. Firstly, the neuron should be active (with output near +1) when the correct inputs are given to it, and inactive (with output close to 0) when the incorrect inputs are given.

Secondly, the activation function needs to be nonlinear, to keep the entire neural network from becoming a simple linear function. There are two possible choices of activation functions: threshold and sigmoid function. The sigmoid function is differentiable and therefore useful for a weight-learning algorithm. Figure 4 shows a neuron, that has N inputs (u1, u2, ..., uj … , un). The inputs are each assigned a weight W (w1, w2, … , wj, …, wn) having a threshold denoted by ,θ, which allows for optimization of the initial start values thus giving an activation function of

$$a_i = \sum_{j=1}^{n} (u_i w_j) + \theta \qquad (2.1)$$

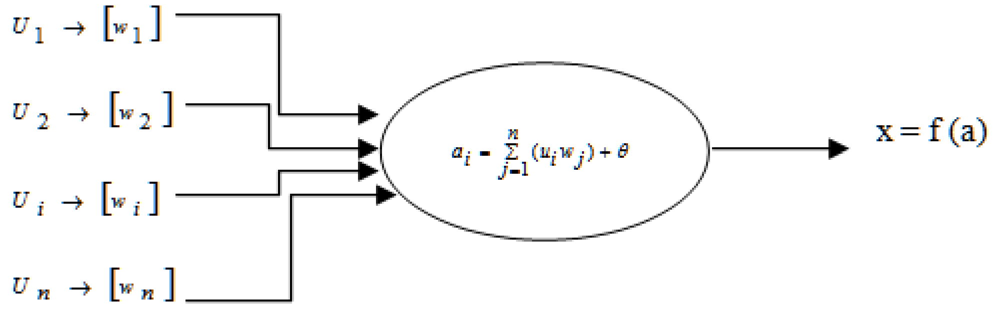

Fig. 2.4 Neuron showing input, weight, input function, activation function, and output [5]

Notice that in addition to the inputs and weights from the other neurons in the network, each neuron also gets a bias. This sets the actual threshold, θ, for the neuron. This allows us to adjust the threshold of "a" away from 0, where it would normally be [5]. The output value of the neuron is a function of its activation $x = f(a)$.

ANNs are mathematical entities or models based on the basic building block known as the artificial neuron and usually modeled after the biological neuron found in the brain. The artificial neurons making up the ANN are distributed processors and operate in parallel. According to [6], an ANNs resembles the brain in two aspects: it is able to acquire knowledge through learning and to store the knowledge through interconnection strengths known as synaptic weights.

The ANN process basically consists of input data intended to be processed in a certain desired way, computational processors comprising several neurons that are capable of extracting important features contained in the input data, output of computational processes, expected target values, and control mechanism for adjusting the parameters values (weights and biases) not determined by a students' performance in the exam. Fig. 2.7 below shows the conceptual model of ANN highlighting its main elements.

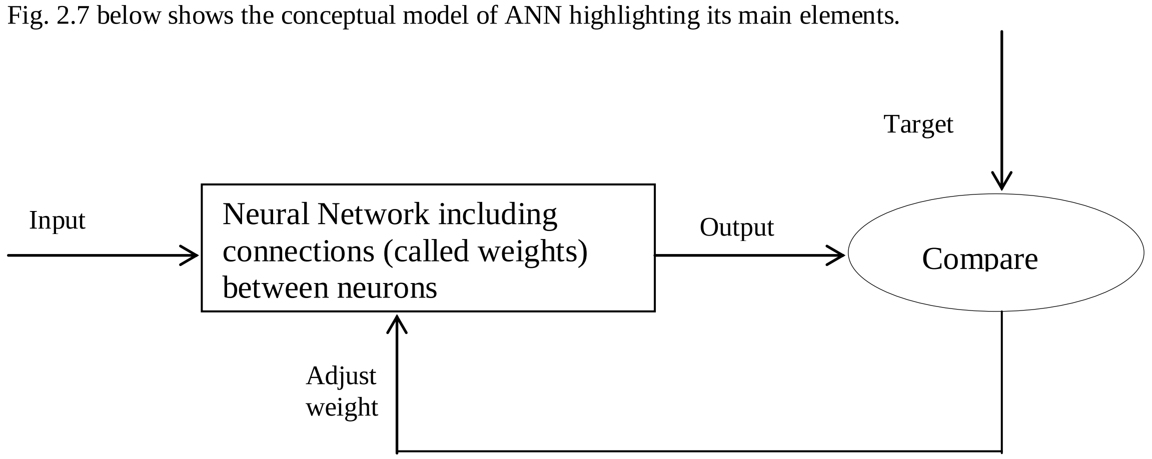

Fig.2.7: Conceptual model of artificial neural networks. [5]

The artificial neurons in ANNs are arranged in layers, each layer providing different functions to the ANNs, e.g., an input layer consisting of input neurons, an output layer consisting of output neurons etc.

II.IV The Back propagation Algorithm

The back propagation algorithm [2] is used in layered feed-forward ANNs. This means that the artificial neurons are organized in layers, and send their signals forward, and then the errors are propagated backwards. The network receives inputs by neurons in the *input layer*, and the output of the network is given by the neurons on an *output layer*. There may be one or more intermediate *hidden layers*. The back propagation algorithm uses supervised learning, which means that we provide the algorithm with examples of the inputs and outputs we want the network to compute, and then the error (difference between actual and expected results) is calculated. The idea of the back propagation algorithm is to reduce this error, until the ANN *learns* the training data. The training begins with random weights, and the goal is to adjust them so that the error will be minimal.

The activation function of the artificial neurons in ANNs implementing the Back propagation algorithm is a weighted sum (the sum of the inputs x_i multiplied by their respective weights w_{ij}):

$$A_j(\bar{x}, \bar{w}) = \sum_{i=0}^{n} x_i w_{ji} \tag{2.2}$$

This shows that the activation depends only on the inputs and the weights.
The most common output function is the sigmoidal function:

$$Oj(x, w) = \frac{1}{1 + e^{A(x,w)}} \tag{2.3}$$

Now, the goal of the training process is to obtain a desired output when certain inputs are given. Since the error is the difference between the actual and the desired output, the error depends on the weights, and there is a need to adjust the weights in order to minimize the error. The error function for the output of each neuron can be defined as:

$$E_j(\bar{x}, \bar{w}, d) = (O_j(\bar{x}, \bar{w}) - d_j)^2 \tag{2.4}$$

Taking the square of the difference between the output and the desired target in order to be always positive, and because it will be greater if the difference is big, and lesser if the difference is small. The error of the network will simply be the sum of the errors of all the neurons in the output layer:

$$E(\bar{x}, \bar{w}, d) = \sum_j (O_j(\bar{x}, \bar{w}) - d_j)^2 \tag{2.5}$$

The back propagation algorithm now calculates how the error depends on the output, inputs, and weights. After finding this, the weights can be adjusted using the method of *gradient descendent*:

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}} \tag{2.6}$$

This formula can be interpreted in the following way: The adjustment of each weight (ΔW_{ji}) will be the negative of a constant eta (η) multiplied by the dependence of the previous weight on the error of the network, which is the derivative of E in respect to w_i. The size of the adjustment will depend on η, and on the contribution of the weight to the error of the function. This is, if the weight contributes a lot to the error, the adjustment will be greater than if it contributes in a smaller amount. (2.6) is used until appropriate weights (minimal error) are found.

So, the derivative of E in respect to w_{ij} needs to be found only . This is the goal of the back propagation algorithm, since we need to achieve this backwards. First, how much the error depends on the output is calculated, which is the derivative of E in respect to O_j from (2.4).

$$\frac{\partial E}{\partial O_j} = 2(O_j - d_j) \tag{2.7}$$

And then, how much the output depends on the activation, which in turn depends on the weights from (2.2) and (2.3).

$$\frac{\partial O_j}{\partial w_{ji}} = \frac{\partial O_j}{\partial A_j} \frac{\partial A_j}{\partial w_{ji}} = O_j(1 - O_j)x_i \tag{2.8}$$

It is shown that from (2.7) and (2.8):

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial O_j} \frac{\partial O_j}{\partial w_{ji}} = 2(O_j - d_j)O_j(1 - O_j)x_i \tag{2.9}$$

And so, the adjustment to each weight will be from (2.6) and (2.9):

$$\Delta w_{ji} = -2\eta(O_j - d_j)O_j(1 - Oj)x_i \tag{2.10}$$

(2.10) can be used as it is for training an ANN with two layers. Now, for training the network with one more layer, some considerations have to be made. To adjust the weights (let's call them v_{ij}) of a previous layer, how the error depends not on the weight needs to be calculated, but in the input from the previous layer. This is easy, just change x_i with w_{ij} in (2.8), (2.9), and (2.10). But to see how the error of the network depends on the adjustment of v_{ik}, we write

$$\Delta v_{ik} = -\eta \frac{\partial E}{\partial v_{ik}} = -\eta \frac{\partial E}{\partial x_i} \frac{\partial x_i}{\partial v_{ik}} \qquad (2.11)$$

Where:

$$\frac{\partial E}{\partial w_{jk}} = 2(O_j - d_j)O_j(1 - O_j)w_{ji} \qquad (2.12)$$

And, assuming that there are inputs u_k into the neuron with v_{ik} from (2.8):

$$\frac{\partial x_i}{\partial v_{ik}} = x_i(1 - x_i)v_{ik} \quad [2] \qquad (2.13)$$

III. SYSTEM DESIGN

III.I The Placement Process

The placement is done via candidates' UTME score and O' level. Each candidate must have five (5) credits in course combination with Mathematics and English Language inclusive and also meet up with the minimum JAMB requirement. A1 – C6 are used to represent a pass, while D7 – F9 are used to represent a fail. The values of the three core subjects for each course are computed to determine which course will be assigned to candidate. This format is demonstrated below.

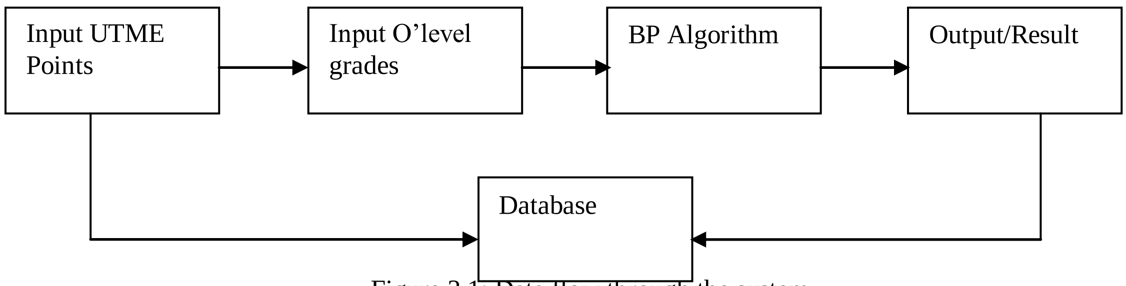

Figure 3.1: Data flow through the system.

The UTME points is entered for verification, on meeting requirement it is passed onto the next stage for further verification and then placement, else, the candidate's disqualification record is saved into the database for future reference. Next, the O' levels of the candidate is fed into the back propagation algorithm. The back propagation algorithm determines which department to assign a candidate based on the input data, and finally displays the result or output (department). The candidate's record is also stored into the database.

The input to the Network is demonstrated below.

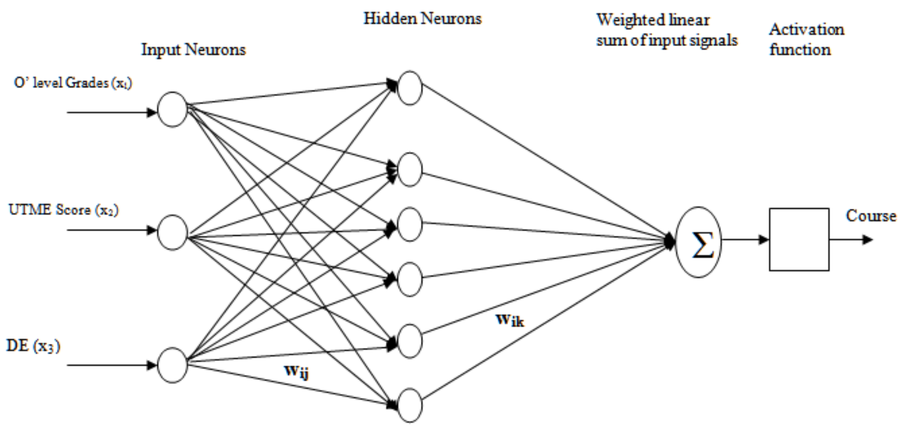

Figure 3.2: Inputs model to ANN algorithm.

During the implementation, the weights will be generated randomly within a specified range of 0.1 to 0.9, because they are the only parameters that can be adjusted by the network. Thus, giving our system the ability to adjust until the best configuration (best representation of the input-output relationship) is gotten. The back propagation algorithm consists of the following steps as derived from equation (2.1), (2.2) and (2.3):
Each Input is multiplied by a weight that would either inhibit the input or excite the input. The weighted sum of the inputs is then calculated.

First, it computes the total weighted input A_j, using the formula in equation (2.1).
Where x_i is the activity level of the jth unit in the previous layer and W_{ji} is the weight of the connection between the ith and the jth unit, as shown in Fig.3.2.
Then the weighed A_j is passed through a sigmoid function that would scale the output in between 0 and 1.
Next, the unit calculates the activity O_j using some function of the total weighted input. Typically we use the sigmoid function in equation (2.2). Once the output is calculated, it is compared with the required output and the total error (E) is computed.
Once the activities of all output units have been determined, the network computes the error E, which is defined by the expression in equation (2.3). Where O_j is the activity level of the ith unit in the top layer and d_j is the desired output of the ith unit.

III.II ANN Configuration and Design

The research work shows how we can input data into the network and the way we can read the output. The system will check to see if the candidate has the required O' levels credit to be admitted into the department and thereby place the candidate or not. This will be accomplished using the back propagation algorithm.

The subjects grade for a student can be any five to nine of the following: A1, B2, B3, C4, C5, C6, D7, E8 or F9 whose equivalent numerical points (scores) are 1, 2, 3, 4, 5, 6, 7, 8, 9 respectively. The subject combination of a student is a total score/points in those subjects indicated as core for a particular course. There are normally three core subjects with each having a possible maximum of 1 and minimum of 6. Real grades e.g. A1, C5, and D7 etc. were used for testing the network to ensure that the system can perform optimally.

The back propagation artificial neural network used in this paper has to be properly configured to best suit the nature of the problem. As seen above, the inputs nodes will be five (5) only corresponding to the number of subjects to be collected for proper placement of Course.

Again, the choice of the number of nodes to form the hidden layer is also crucial because the hidden layer allows our algorithm to converge smoothly, thus adding to the accuracy of our system. For the purpose of this research, ten (10) hidden nodes were deemed appropriate based on trial means. There is basically no formal means to determine the accurate or optimal number of nodes in any layer, although few unconventional means might be adapted.
That is why the only three (3) layers were used; input layer, hidden layer, and output layer.
The output layer chosen for this network is only one so that the result can be read distinctly without any ambiguity.

Table I: Used Parameters for Neural Network.

Parameters	Values
Number of Inputs	3
Number of outputs	1
Number of hidden layers	6
Error	0.02

Once the candidate fail to meet up with the required grade, the system will prompt the user for and alternative course else the system places the candidate.

The result of the computation is passed through the linear function, in this case 'sigmoid function'. This is to maintain the linear region of operation without running into saturation. The output is thereby compared with the target result, and the error is calculated and is propagated back to the network, where the weights are adjusted.

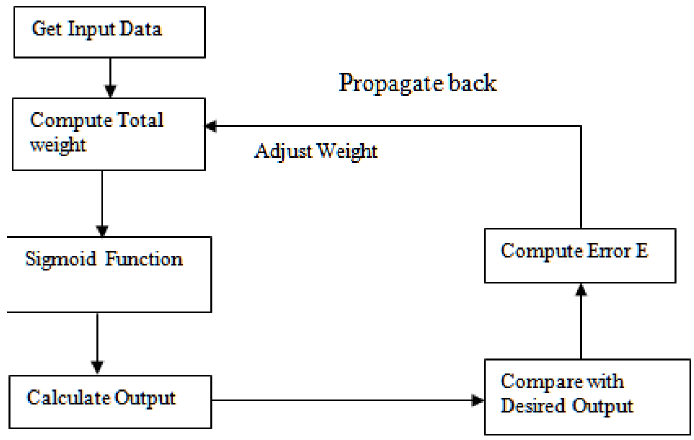

Figure.3.3: Back propagation network model

The user enters data into the network, in this case UTME points, O'level subjects and their respective grades and DE result. Each of the inputs has an associate weight that is computed. The result of the input weights is passed through the sigmoid function which output is then calculated. The output result of the sigmoid function is compared against the target (desired) point. The error is thereby calculated using equation 2.3, which is propagated back to the network until it has learned how to classify students into various courses of choice or an alternative courses.

III.III Implementation design

For this design, we used C# (visual studio 2010), this is to maximize the graphical user interface (GUI) of the compiler and make the system user friendly as opposed to the console application (command line) mostly used in implementing back propagation algorithm.

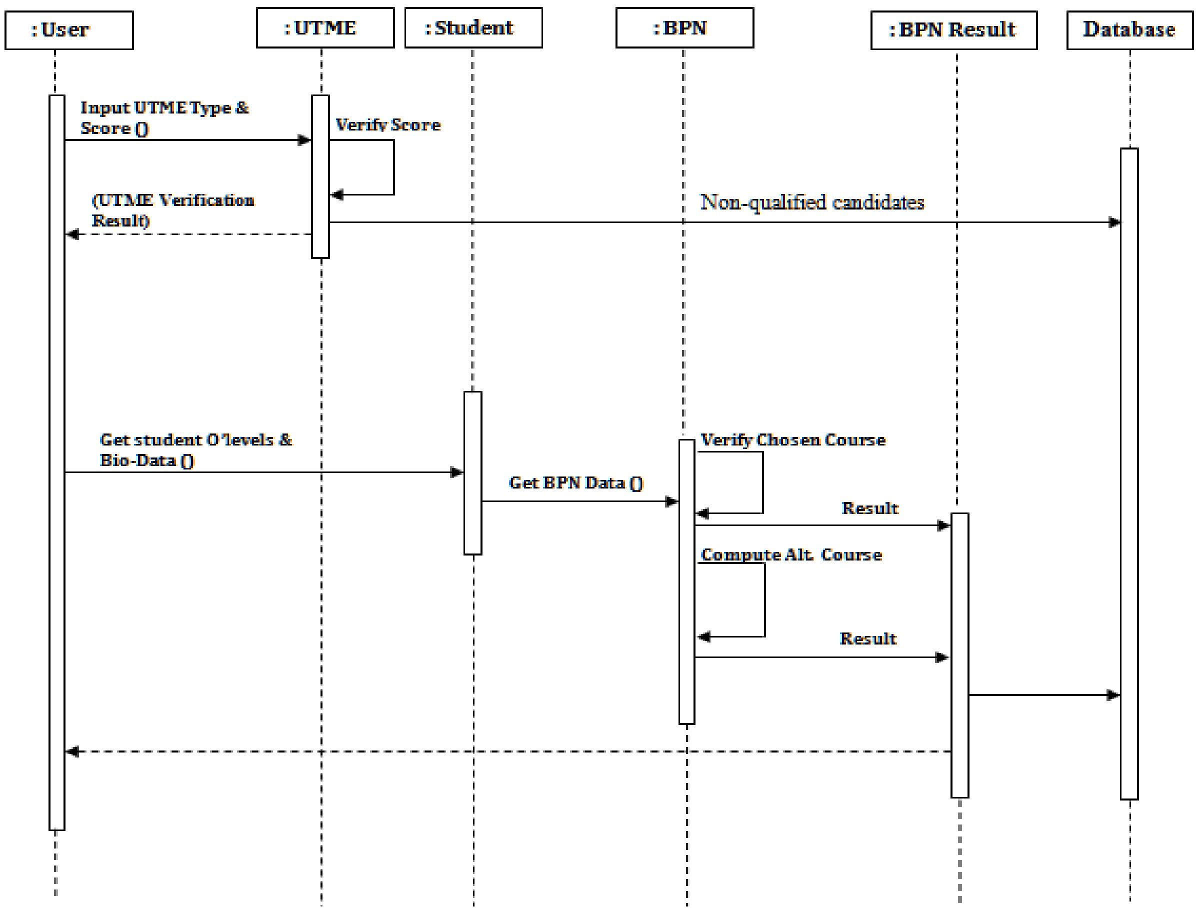

Fig.3.4: Sequence diagram for implementation

The diagram shows user entering mode of entry with its associate points or grade, the system thereby verifies if a candidate has a required points or grade for admission and of course the level. On meeting the requirement, candidates O' level will be required from the user and other information in order to proceed with the placement process. On the other hand if the candidate fail to meet up with required entry points, admission is refused and his/her record is saved to the database. The candidate's O' level is processed by the ANN to verify if the candidate has qualified for the applied course. If qualified, the applied course of choice is assigned, else, and alternate course in which he/she has the requirement is allocated. The worse case is when a candidate has not met requirement for any of the available courses; in that case no admission placement is offered. In which ever case, the candidates' records are saved into the database for the purpose of referencing.

IV. RESULT AND DISCUSSION

IV.I Result

Table II and Table III show the list of qualified candidates and list of disqualified candidates respectively.

Table II: List of Qualified Candidates

QUALIFIED CANDIDATES' RECORD

University ADSU

ID	CandidateName	CandidateNumber	University	Department	CandidateCourse	LevelAdmitted
14	VAISA PETER	0001	ADSU	MATHEMATICA...	COMPUTER SCI...	100
15	PETER WATO	0019	ADSU	MGT & BUSINES...	ACCOUNTING	200
16	AISHA BELLO	1004	ADSU	AGRICULTURE	FISHERIES	200
17	BILKISU SULE	1111	ADSU	AGRICULTURE	AGRIC ECONS &...	200
25	JOEL MADIKI	3030	ADSU	MGT & BUSINES...	ACCOUNTING	200
26	ELDAD TOMA	1000	ADSU	MGT & BUSINES...	BUSINESS ADMIN	100
27	MURTAL AHMED	1002	ADSU	SCIENCE	GEOGRAPHY	100
28	ATIKU ABBA	0003	ADSU	AGRICULTURE	AGRIC ECONS &...	100
29	JOEL ZIRRA	9292	ADSU	AGRICULTURE	BOTANY	200
30	FUNGO ELKA	4004	ADSU	SCIENCE	COMPUTER SCI...	200
31	JIBBRIL AMINU	0100	ADSU	SCIENCE	COMPUTER SCI...	200

Exit

Table III: List of Disqualified Candidate

DISQUALIFIED CANDIDATES' RECORD

Name	idNumber	University	Entry	Remark
VAISA PETER	A0001	ADSU	UME	UME-Points not met
PETER WATO	0019	ADSU	IJMB	IJMB-Points not met
AYUBA JOHN	9003	ADSU	IJMB	IJMB-Points not met
AHMED ALI	9000	ADSU	UME	UME-Points not met
BILKISU SULE	1111	ADSU	ND	O-Level not met
NURU IDI	3030	ADSU	UME	O-Level not met
SULE YAYA	2020	ADSU	UME	UME-Points forged
AMINA ABBAS	0011	ADSU	UME	UME-Points forged
KHADIJAT BULUS	2220	ADSU	UME	UME-No Record Found
PRINCE EKA	9002	ADSU	ND	O-Level not met
Udo John	0004	ADSU	UME	UME-Points forged

Exit

IV.II Discussion

Table II shows the list of admitted students into the various courses of Adamawa State University, Mubi, having met the entry requirements of UTME, IJMB, DE and O'level for each course. From Table II, it is clearly shown that students with candidateNumber 0001, 1000, 1002 and 0003 were admitted into 100 level of Computer Science, Business Administration, Geography, Agric. Economy and Extension respectively, having met O'level and UTME or other entry requirements. While students with candidateNumber 0019, 1004, 1111, 3030, 9292, and 4004, were admitted into 200 level of Accounting, Fisheries, Agric Economy and Extension, Accounting, Botany and Computer Science respectively, having met the O'level and DE requirements of the respective courses. Furthermore, the system assigns alternative courses to students who are not qualified for their chosen course but have the requirements into other courses of the University.

The result displayed in Table II proved that the proposed classification algorithm is able to admit students into various courses of their choices without favouritism, biasness, and selfinishness. The proposed algorithm also ensured even distribution of students into various courses of the Universities. This is in conformity with the undergraduate students' handbook (2012) of Adamawa State University, Mubi.

Table III displays the result of the disqualified candidates who have not met the requirements of admission into the University. It is clearly shown in the remark column of Table III that students with idNumber A0001, 0019, 9003 and 9000 were not qualified for admission into the University due to lack of required UTME points. Similarly, students with idNumber 1111, 3030 and 9002 were not offered admission because they could not meet the required O'level grades. Furthermore, students with idNumber 2020, 0011 and 0004 were not offered admission because of forgery of UTME points, and his/her remark column is tagged as "forged". Similarly, when UTME record of a candidate is not found in the database, as in the case of candidate with idNumber 2220, the candidate is rejected. This finding is in conformity with to Nigerians' Universities admission placement process.

It can then be deduced from Table III that the proposed classification algorithm proved to be secure from irregularities such as favouritism, biasness, and human error that is associated with the traditional method of students' admission placement.

V. CONCLUSSION AND RECOMMENDATION

It is shown that back propagation algorithm has the capability to palace candidate into the department or course in which he/she applied for having meet the requirement. On the other hand, the system can assign an alternative course for candidates that are not qualified for their own course of choice. The system could also verify the UTME entry point for each candidate seeking admission and also its authenticity before placement. This system could be used in the universities to eliminate error, favour and biasness that exist during admission process. Also only students with genuine UTME results are admitted into the university. The system is limited in that it cannot check the authenticity of the O'level result due of the fact that the body concern does not send students result to the universities' database. It is hoped that this will motivates the Nigerian universities to look at possibilities of making use of artificial intelligence in the university admission placement process.

Further studies in this area should focus on ranking of students during placement process. This will be beneficial in that, it will make it easy to select the best out of those qualified for the same course, in a situation where all candidates cannot be admitted.

REFERENCES

[1]. N. Bijayananda & R. Srinivasan (2004). Using Neural Network to Predict MBA Student Success. retrieved November 11, 2012 from http://www.eric.ed.gov/ERICWebPortal/ recordDetail?accno

[2]. G. Carlos (2012). *Artificial neural networks for beginners*. Retrieved September 4, 2012 from http://arxiv.org/ftp/cs/papers/0308/0308031.pdf.

[3]. R.C. Chakraborty (2010, June). Fundamentals of Neural Networks. AI Course lecture 37–38, notes, slides. Retrieved from www.myreaders.info/html/artificialintelligence. html

[4]. S. Elayidom, S. Idikkula & M. Alexander, J. (2011). A Generalized Data mining Framework for Placement Chance Prediction Problems. *International Journal of Computer Applications (0975 – 8887)* Volume 31– No.3, October 2011

[5]. W. Franklin, & M. M. Fullgence (2011). Student selection for University Course Admission at the Joint Admission Board Kenya using trained neural networks. *Journal of information technology education Volume 1o*,2011.

[6]. S. Haykin (2004). *Feedforward neural networks: An introduction*. Retrieved December 17, 2012 from http://media.wiley.com/product_data/excerpt/19/0471349 1/0471349 119.pdf.

[7]. M. Minsky & S. papert (1969). Perception without Awareness, Psychology of unconscious from conscious cognition. *Journal of experimental psychology.vol*.4 p. 11-22

[8]. RP Group. (2001). Predicting Student Outcomes Using Discriminant Function Analysis. *RP Group Proceedings publication.* Retrieved December 7, 2012 from http://www.rpgroup.org/.

[9]. S. Sajadin, M. Zarlis, D. Hartama, S. Ramliana & W. Elvi (2011). Prediction of student academic performance by an application of Data Mining Techniques. *International Conference on Management and Artificial Intelligence. IPEDR vol.6 (2011) © (2011) IACSIT Press, Bali, Indonesia*

Mikailu, Habila obtained his B. Tech. degree in Computer Science 2009 from Federal University of Technology, Yola and now Modibo Adama University of Technology, Yola. His currently pursuing his M,Sc. degree at Adamawa State University Mubi - Nigeria. His area of interest includes artificial neural networks.

Gregory MakshaWajiga is a Professor of Computer Science with Modibbo Adama University of Technology, Yola. He holds PhD in Computer Science from Abubakar Tafawa Balewa University, Bauchi in 2000 and MSc Industrial Mathematics (Statistics and Operational Research), Aston University, Birmingham, 1983. He also obtained his BSc Mathematics, Ahmadu Bello University, Zaria in 1979. Gregory MakshaWajiga is Editor-in-Chief, Bagale Journal of Pure and Applied Sciences, Modibbo Adama University of Technology, Yola. He is a member Nigeria Computer Society (NCS), Digital Equipment Computer Users' Society. He is happily married with three children.

Doctor Peter Buba Zirra is Lecturer with Adamawa State University, Mubi Nigeria. He obtained his Doctorate degree in Computer Science from Modibbo Adama University of Technology, Yola in 2012, MSc in Computer Science from Abubakar Tafawa Balewa University, Bauchi in 2006, MBA (Finance) from University of Maiduguri, Borno state in 2000 and had his B.Tech in computer science, 1994 same AbubakarTafawaBalewa University, Bauchi. His area of interest includes Computer Network and Security. He is happily married with two children.