





Development of Stroke Prediction System Using Neural Network with Sigmoid Activation Function

Michael A. Adegoke¹, Adedayo F. Adedotun^{2*}, Tunde Odeyinde¹, Alao O. Grace¹, Kehinde A. Sotonwa³, Benjamin S. Aribisala³, Odetunmbi A. Oluwole², Odekina G. Onuche²

¹ Department of Computer Science and Information Technology, Bells University of Technology, Ota 112101, Nigeria

² Department of Industrial Mathematics, Covenant University, Ota 112101, Nigeria

³ Department of Computer Science, Lagos State University, Ojo 102115, Nigeria

Corresponding Author Email: adedayo.adedotun@covenantuniversity.edu.ng

Copyright: ©2024 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/jesa.570623>

ABSTRACT

Received: 18 April 2024

Revised: 19 June 2024

Accepted: 15 November 2024

Available online: 31 December 2024

Keywords:

artificial neural network, supervised learning, stroke prediction, patient's vital signs, artificial intelligence

The rule-based approach for the analysis and interpretation of stroke symptoms for the purpose of accurate and timely diagnosis and treatment has been reported as weak. In this paper, an artificial neural network using a sigmoid activation function is used to develop a stroke diagnosis system that is capable of automatically extracting the features responsible for stroke. The system is capable of using knowledge learned from data to predict the likelihood of a patient having a stroke or not. The efficiency of the stroke prediction system was evaluated on live data obtained from four medical centers located in the southwestern part of Nigeria. The results show an average prediction accuracy of 93.75% and can carry out a complete diagnosis of a patient within 4 microseconds. The proposed system, therefore, will facilitate quick diagnosis and timely treatment of stroke patients and consequently has the potential of reducing brain damage and other complications that may arise due to slow or delayed diagnosis of strokes.

1. INTRODUCTION

Stroke is a medical emergency that occurs when the blood supply to part of the brain is interrupted or reduced, preventing the brain from getting oxygen and nutrients, resulting in the brain cells dying by the minute [1]. This implies that every minute counts in the diagnosis and treatment of stroke, as early action can reduce brain damage and other complications. According to the Centers for Disease Control and Prevention (CDC), stroke is the fifth leading cause of death in the United States, with over 140,000 people dying of stroke every 40 seconds [2]. It has been estimated that, one out of every three persons worldwide will develop stroke in their life time [3]. It has further been estimated that, over 795,000 people in the United States suffer from stroke daily (CDC). Therefore, prompt detection and treatment of stroke is crucial for the survival of victims, as it is capable of rescuing up to 80% death that may arise as a result of stroke [4].

The stroke treatment that works best requires that stroke symptoms in a patient be recognized within three (3) hours of their appearance [1]. Detecting stroke symptoms quickly represents a key step in determining the individual patient's optimal therapy regimen [3]. The sooner stroke is detected and acted upon, the better the outcome of the treatment. Some of the tools developed to reduce the time to diagnosing strokes are Magnetic Resonance Imaging (MRI), Computerized Tomography (CT), and Electrocardiogram (EKG). The MRI is used to check if brain cells or tissue have issues that may lead to stroke, while the computerized tomography (CT) scan is

used to provide detail and a clear picture of the brain and check signs of bleeding or damage in the brain that could lead to stroke. The EKG test records the heart's electrical activity by measuring its rhythm and recording how fast the heart beats. These diagnostic techniques are used to collect data such as blood pressure, heartbeat rates, blood oxygen, facial images, vocal recordings, etc., which physicians can analyze to determine the presence of stroke or predict the onset of an impending stroke [1]. In other words, data obtained from these tools could be used to predict whether a patient is susceptible to having a stroke or not. Analysis and interpretation of this data can aid early treatments and consequently prevent brain damage, permanent disabilities, and death that may arise from the full-blown symptoms.

Nevertheless, the amount of data produced by these tools for stroke tests is so vast that it is not possible for the domain experts to promptly and accurately analyze it using their conventional techniques. The importance of accuracy and timing is emphasized. The faster the diagnosis is made and appropriate therapy is initiated, the better the outcome of the treatment [5]. Hence, the use of automated methods for the analysis and interpretation of medical data for predicting a patient's susceptibility to stroke.

Computer-aided diagnosis (CAD) systems are well-established automated methods for providing aid for the analysis and interpretation of stroke [6, 7]. The classical CAD process data in two steps: feature extraction and hand-coding. Feature extraction establishes the process of establishing the rules for evaluating medical data, while hand-coding is the

process of instructing the computer to learn the rules for classification or prediction. In CAD, the knowledge engineer manually codes the rules and conditions for the presence or absence of a stroke based on patient data, whereas the physician, who is a domain expert, provides the knowledge needed to establish the rules. CAD is therefore a rule-based approach in which stroke experts predefine the features responsible for stroke which are later hand-coded by the systems analyst (knowledge engineer) [8].

With the rule-based system, all possible symptoms of stroke known to the domain experts are included in the knowledge base. With such a system, analysis and prediction of the possibility of an imminent stroke can only be done using data from the knowledge base of domain experts. Such a system cannot generalize but at the best serve as a look-up table for stroke prediction [8]. But in most fields, new symptoms appear every day that domain experts might not notice right away. As a result, it may not be feasible for the experts to incorporate in the knowledge base every scenario that could increase a patient's risk of stroke. If the logical conditions that characterize CAD run out of possibilities, the system will either break down or conclude poorly [9].

Another downside of this approach is that every time new symptoms or variants of existing symptoms are discovered, the knowledge engineer would have to be contacted to rewrite the rules for prediction. If the event that he or she is unavailable, the system would have to be on hold or its output not reliable and dependable. Variants of stroke diseases tend to be common in different human populations, with each variant manifesting different symptoms [10]. Thus, stroke knowledge coded for a particular human population may differ slightly from the knowledge needed to detect and predict stroke in another population or race. Updating the coded knowledge for every new symptom or changing the coded knowledge for every human population may be cumbersome, time-consuming, frustrating, and impractical. Furthermore, in the CAD approach, the time taken for feature extraction and hand-coding is significant. There is a need to develop a system that will be able to automatically extract features that may be responsible for stroke and learn the knowledge for predicting stroke from existing data. Such a system will be able to adjust to changes in the rules for predicting the susceptibility of a patient's to stroke without having to contact knowledge engineers every time new symptoms are added to the knowledge base. A system will be able to study the set of symptoms that may be responsible for stroke from existing data and training itself to predict a patient's susceptibility to stroke. A good algorithm in this category is the use of representative learning. Representation learning is a supervised learning approach in which manual hand-coding is not necessary [11-13]. Rather, the features or patterns for prediction are generated automatically by an inductive process. The algorithm learns the rules for prediction from data rather than being explicitly programmed.

In this research, an artificial neural network using the sigmoid function is used for the purpose of predicting whether a patient is susceptible to stroke or not. The system developed is capable of using data obtained from MRI, CTI, EKG and patients' vital signs obtained from inexpensive and easily accessible medical laboratory methods. This enhances the usability of the system across the human population and around the globe.

2. OVERVIEW OF NEURAL NETWORK WITH SIGMOID ACTIVATION FUNCTION

Artificial neural networks (ANN) are a widely used artificial intelligence algorithms that are used in machine learning for analyzing data, learning from the data, and making a prediction or generalizing about new data [14]. Artificial neural networks, developed using motivations from biological neural networks, is made up of interconnecting artificial neurons (computing elements) that mimic the properties of biological neurons. In biological neural networks, a neuron receives signals, processes the signals, and communicates with another neuron in the neuronal assembly using electrical signals, which are short-lived impulses in the voltage of the animal cell wall. Similar to biological neurons, artificial neural networks are capable of receiving input signals, using relevant features learned from existing data to convert the input signals into some output signals.

Artificial neural networks are used to solve artificial intelligence (AI) problems in which the formulation of a closed-form expression is not possible. These are problems in which it may not be feasible to envisage or include all the possibilities that might be responsible for an event to be triggered [15], of which instructions that trigger an action are often learned from existing data. Such cases arise due to the proliferation of data-driven applications; that is, applications in which there is no finite number of possibilities. Other machine learning algorithms that can be used to model data-driven applications include hidden Markov models (HMMs), support vector machines (SVMs), Bayesian networks, and fuzzy logic, among others [16, 17]. However, the use of artificial neural networks is considered in this study because of its inherent non-linearity. A non-linear system is a fault-tolerant system that can withstand noise and hardware failure, where a change in the input signal caused by a malfunctioning piece of hardware may not always be reflected in the final result. It is indeed a function that does not obey a proportionality restraint in the output due to a fault in the hardware being used.

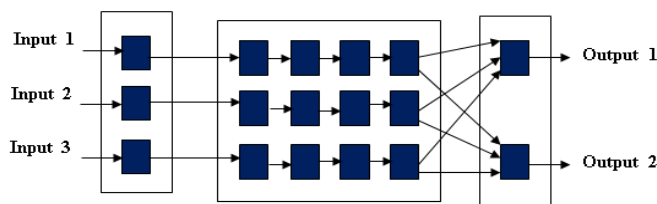


Figure 1. Neural network with a single hidden layer, three inputs, and two outputs

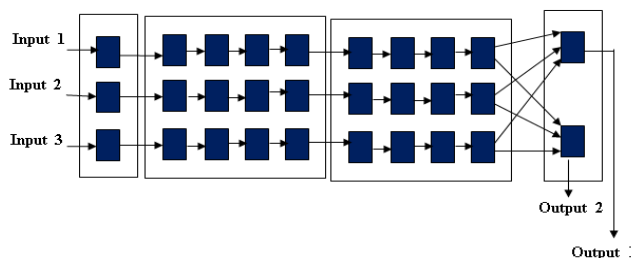


Figure 2. Neural network with multiple hidden layers, three inputs, and two outputs

ANN is suitable for modeling machine learning applications that require mapping an input with an output and can be used to find patterns in data. The conceptual diagram of a single hidden layer ANN, which comprises three (3) layers, namely: input layer, hidden layer, and an output layer, is shown in Figure 1.

The three nodes at the input layers signify that there are three dimensions or components to the input data. While the two output nodes show that there are two possible outputs for every set of input that is passed forward through the network [18, 19].

It comprises a layer of input connected to a layer of hidden units, which in turn is connected to a layer of output units. The activity of the input unit represents the raw information that is fed into the network, while the activity of the hidden units is determined by the activity of the input unit and the weights between the hidden and output units. Since animal brains have multiple hidden layers, artificial neural networks can also be realized by using multiple hidden layers. This is described as a deep learning neural network, which is an abstraction of neurons in animal brains [20, 21]. The conceptual diagram of an ANN with multiple hidden layers is shown in Figure 2.

The processing ability of the network is stored in the inner layer connection strengths called weights. The weights, which are initialized by arbitrary single numbers that are constantly modified to adapt to their particular stimulus input, represent the knowledge the network uses to analyze any input presented to it. The process of constantly adapting the weight to stimulus from a set of pattern examples, described as training, enables the network to learn how to carry out a given task [20].

Communications between layers are represented by weighted signals comprising input patterns multiplied by the arbitrary single numbers between nodes in the different layers. Each of the weighted signals terminating at each succeeding node is summed up by simple arithmetic addition to supply the node's activation. Suppose there are n signals terminating or entering a node k . The activation, α (which is the behavior) of the receiving node is given by:

$$\alpha_k = w_1x_1 + w_2x_2, \dots, + w_nx_n \quad (1)$$

Eq. (1) can be written in a compact form as:

$$\alpha = \sum_i^n w_i x_i \quad (2)$$

where, w_i is the arbitrary weight for the input component i and x_i is the input component i . Depending on the result of the summation, the activation can either be excitatory or inhibitory according to a pre-defined threshold value. Assuming the threshold value is given as θ , the activation will be excitatory if $\alpha \geq \theta$ and inhibitory otherwise as expressed in the threshold function in Eq. (3) that follows:

$$\text{Activation Potential} = \begin{cases} 1 & \text{if } \alpha \geq \theta \\ 0 & \text{if } \alpha < \theta \end{cases} \quad (3)$$

However, real neurons do not get activated based on two states of "on" or "off" represented by "1" or "0" as depicted in Eq. (3). Rather, the neurons fire (are activated) within a range of values between "1" and "0". That is, the two-step function of Eq. (3) can be softened by a continuous signal such that the

activation depends on the real values of α . One convenient way for expressing this is by the use of the Sigmoid function.

In the Sigmoid function, as α tends to large positive values, the Sigmoid (activation) tends to "1" but never actually reaches this value. Similarly, as α tends to large negative values, the Sigmoid tends to "0". The Sigmoid function, often denoted by σ , can be mathematically expressed as:

$$\text{Activation Potential} = \delta(\alpha) \equiv \frac{1}{1+e^{-(\alpha-\theta)/\rho}} \quad (4)$$

where, e is a mathematical constant and quantity ρ determines the shape of the function of which large values make the curve flatter and small values makes the curve steeper. The parameter ρ is sometimes omitted so that it is implicitly assigned the value 1.

Sigmoid function makes it possible for the network to predict an output even when the criteria for making such predictions are hard to define. This is, when the parameters of the input data are not strictly within the parameters of the sample data. Such a system will be able to handle uncertainty and imprecise details [22].

3. METHODOLOGY

3.1 Description of the neural network-based stroke prediction system

The neural network-based model for stroke prediction presented in this paper consists of ten (10) nodes at the input layer, three (3) hidden layers with each layer having eight (8) nodes, and three (3) nodes at the output layer, putting the total number of nodes at 37. This is shown in Figure 3.

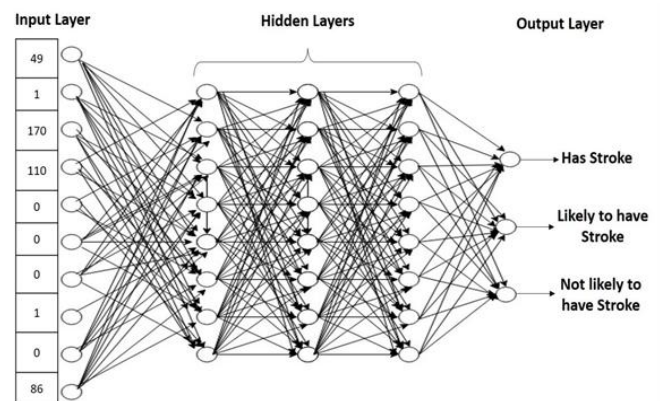


Figure 3. Architecture of the neural network system for stroke prediction

The ten (10) input nodes capture the following 10 information (fields) from a patient: age, gender, systolic blood pressure, diastolic blood pressure, family history of stroke, alcohol use, tobacco use, risk factor, random blood sugar (RBS)/fasting blood sugar (FBS), and stroke status. That is, one node for each field of the input data. The ten-input data for each patient constitute the set of parameters needed for the diagnosis of stroke in any patient [4]. Since artificial neural networks deal with numerical data, an integer encoding scheme, in which non-numerical data elements for each patient were converted into integer values before they are fed into the network, were used. The following nomenclatures were used

for the conversion:

- Gender: Male=1; Female=2
- Family History of Stroke: None=0; Parent=1; Sibling=2
- Alcohol Use: None=0; Use only when in Society gathering=1; Heavy user=2
- Tobacco Use: None=0; Formerly=1; Smoker=2
- Risk Factor: None=0; Hypertension=1; Hypertension CKD=2; Hypertension diabetes=3; Hypertension obesity=4; Others=5

For instance, the input data for a patient with the following record: Age =49, male, systolic blood pressure of 70 and diastolic blood pressure of 110, with no family history of stroke, non-alcoholic drinker, non-smoker, who has no hypertension and of Fasting blood sugar of 86 will be represented as follows:

[49,1,170,110,0,0,0,1,0,86].

Since the number of nodes and layers are not large, a small learning rate of 0.4, that will enable the model to learn slowly but globally so as to effectively maps the input data to the best output from the training set, was carefully chosen. A too small learning rate was avoided so that the model will not get stuck in the process.

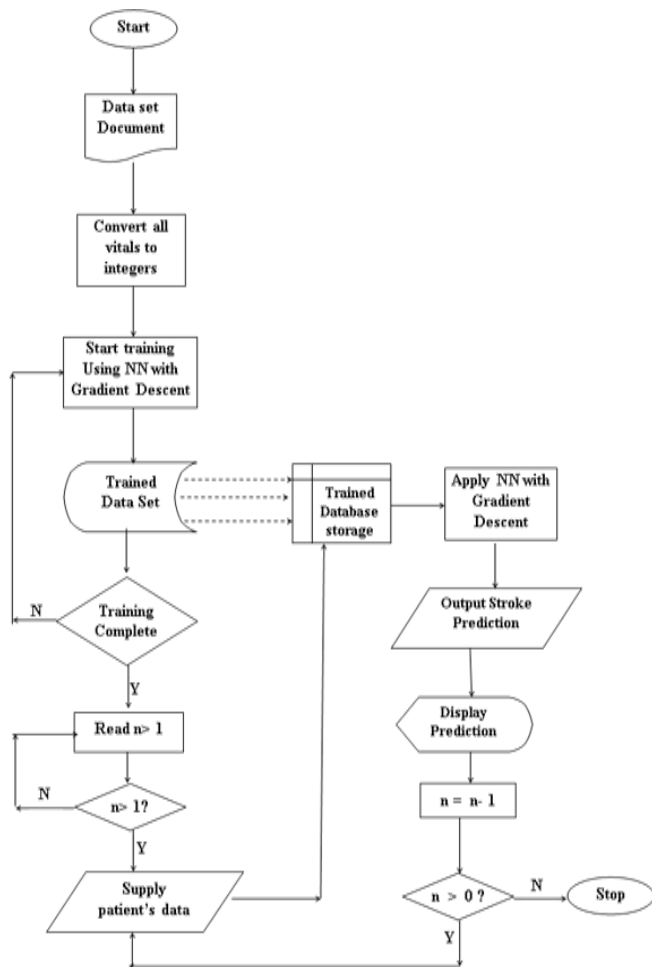


Figure 4. Flowchart for training the artificial neural network for stroke prediction

The input data for other patients follows the same pattern. The twelve fields for each patient are fed into the network through the twelve input nodes. Using Eq. (1) and the Sigmoid activation function in Eq. (4), the hidden layer uses the

weighted signal on the network, which represents the experience and knowledge of the network, to determine whether or not a patient is susceptible to stroke. In other words, the network is pre-trained with data sets that are representative of the three (3) likely outputs for the prediction to determine a threshold for each of the likely outputs. The procedure for training the network for the prediction task is presented in Algorithm 1 and pictorially represented by Figure 4.

Algorithm 1

Start

Input: The dataset document was captured using MS Excel. The Excel sheet contains a dataset of 400 patients in a tabular form consisting of the following column fields; Age, Gender, Systolic Blood Pressure, Diastolic Blood Pressure, Family History of Stroke, Alcohol Use, Tobacco Use, Risk Factor, Random Blood Sugar (RBS), Fasting Blood Sugar (FBS), Stroke Status.

Output: Expected predictions (“This patient has stroke”, “This patient is likely to have a stroke” or, “This patient is not likely to have a stroke”).

Process:

Step 1: Supply the Excel sheet containing datasets to the MATLAB application.

Step 2: Since integers are what the Neural Network understands, during pre-processing, the values of the set of distinct data collected for each patient that are not integer values are converted to positive integers using the following nomenclature;

- Gender: Male -1 Female -2
- Family History of Stroke: None -0 Parent -1 Sibling -2
- Alcohol Use: None -0 Social -1 Heavy -2
- Tobacco Use: None -0 Former -1 Smoker -2

Risk Factor: None -0 Hypertension -1 Hypertension, CKD -2Hypertension, Diabetes -3 Hypertension, Obesity -4 Others -5

For instance, the interpretation of a patient’s data who has stroke and is a 49-year-old male with a blood pressure of 170 systolic/110 diastolic, no family history of stroke, non-alcohol or smoke, has hypertension and Fasting Blood Sugar (FBS) 86, will look like this -[49 1 170 110 0 0 0 1 0 86].

The preparation and interpretation of the data describes as feature engineering entails representation of the data in a more predictive way. It involves a time-dependent and sequential structures that are carried out recurrently or in succession in a series of steps using recurrent neural networks (RNNs). RNNs learn the internal representation of the data that contains latent features. The latent features present new representation of the data in succession and it enables the model to generalize to sequence of data not previously seen during training. The latent data can be viewed as a condensed representation of all the past inputs that are relevant to the task at hand. The new representation is then learned in succession with each layer distilling the latent features via weights that ascertain their respective values; and the last layer of the network acts as a linear classifier and uses the learned representations to generate a final output.

Step 3: Train the dataset using Neural Network with

gradient descent in the MATLAB environment to be able to predict the three categories of the desired output (“This patient has stroke”, “This patient is likely to have a stroke” or, “This patient is not likely to have a stroke”).

Data learning is repeated 100 times for 80 randomly selected data from the three categories of test dataset.

Step 4: For $i = 1$ to n dataset
 Do learning procedure
 Next i
 End for

Step 5: Accept patient’s input data in string format as in Step 2.

The patients’ input data is converted to integer using the procedure in Step 2.

Step 6: Run the model for the prediction and the possibility or not of a stroke.

Step 7: Report the result of the prediction in Step 6 above.

Step 8: Advise the patient accordingly.

Step 10: Any more new patient’s input data?

If Yes

Go to Step 5

Else

Step 11: Stop

After training model, data from patients who visited a hospital can then be fed into the network. Using the Sigmoid function, a patient will be predicted to have stroke if the value of the sigmoid function tends to be a large positive value and a treatment regimen will be prescribed. If the sigmoid activation function value tends towards a large negative value, the patient is predicted to not be susceptible to stroke; conversely, if the value of the sigmoid activation function is in the middle range, the patient is predicted to be likely to have stroke and will be advised to monitor his or her lifestyle. The neural network model for stroke prediction in this paper was tested with live data obtained from the following hospitals across the southwestern part of Nigeria: Federal Medical Centre, Abeokuta; General Hospital, Ado-Odo Ota; Federal Medical Centre, Lagos; and Federal Medical Centre, Owo, Ondo state.

3.2 Dataset and experimental setting for training and testing the stroke prediction system

Data from 400 patients obtained from four hospitals, namely: Federal Medical Centre, Abeokuta; General Hospital, Ado-Odo Ota; Federal Medical Centre, Lagos; and Federal Medical Centre, Owo, Ondo state, were collected to train and test the stroke prediction system. The listed hospitals have stroke sections. The data were collected from the records departments of the stroke clinics of the various hospitals. The researchers were also allowed to visit the stroke clinics to see the patients, one on one to confirm the records collected. We were always scheduled to visit these hospitals on their clinic days which is twice in a week for each of the listed hospitals. It was observed that, some of the patients were already recovering and could interact, while some were under intensive care receiving treatments.

The dataset consists of records of three (3) categories of patients, as follows:

Category 1: records of patients with stroke

Category 2: records of patients that have been diagnosed to have symptoms of stroke but are yet to have stroke

Category 3: records of patients with no traces of strokes

The records of the number of patients already suffering from stroke is 185, while the records of patients with symptoms of stroke but are yet to have stroke and who are receiving treatment is 93. The number of records of people without traces of stroke is 122.

The 400 data were randomly split into two (2) in the ratio of 80:20. The 80% (i.e., 320 data) were used to train the network, while the remaining 20% (i.e., 80 data) were used to test the performance of the network for stroke prediction task. The 80% records used for training consists of 148 records of patients with stroke status, 74 records of patients that have symptoms of stroke and 98 records of patients with no traces of stroke. Recurrent neural networks were used to enhance the learning of the neural network model by representing and training the data in sequence of steps using a step size, learning rate of 0.4. This enables the network to generalize well enough to be able to predict correct output for a previously unseen input. The 20% (80 patients) used for testing the performance of the network consists of the records of 37 patients with stroke, records of 19 patients that are likely to have stroke and records of 24 patients with no traces of stroke. Table 1 shows the result of the neural network for the 20% (80 records) in the first epoch.

The experiment was repeated 100 times with the data set randomly partitioned into two (2) in the ratio of 80: 20 each time. Figure 5 shows the interface of the trained neural network-based stroke prediction system while Table 2 shows the results for the 100 experiments. In the first experiments, the neural network-based stroke prediction system was able to correctly predict the test data in 68 times out of 80 resulting in 85% accuracy. In the second experiment, the system was able to predict 72 test data correctly out of 80 resulting in 90% accuracy. The average percentage accuracy of the 100 experiment is 90.73%. It should be noted that, the model misclassified and missed the correct predictions for the 15% of the test data in the first experiment. The misclassified cases reduced to 10% in the second experiment; and in the 100th experiment, the misclassified cases has reduced to 7.5%. This means that, the model’s accuracy increases or improves with usage and the familiarization of the model with the training data. It was further observed that the accuracy would improve greatly if the model is allowed to learn slowly and globally. This occurs when the learning rate is carefully chosen. The learning rate of 0.4 was observed to be optimal for this experiment.

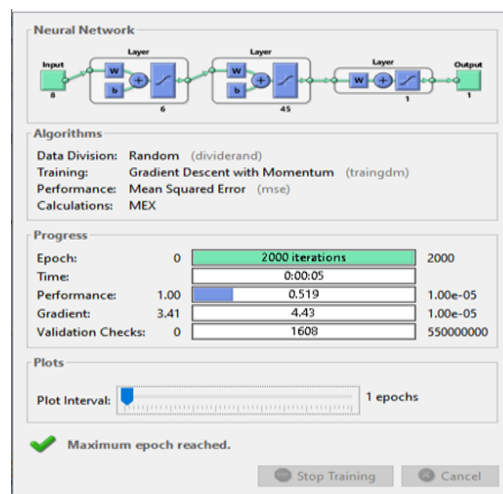


Figure 5. Neural network training 1

Table 1. Results of prediction for records of 80 patients in one epoch

No.	Age	Gender	Systolic	Diastolic	Family History	Alcohol Use	Tobacco Use	Risk Factor None-0	RBS	FBS	Actual Stroke Status	System Prediction
		M-1 F-2			None-0 Parent-1 Sibling-2	None-0 Social-1 Heavy-2	None-0 Former-1 Smoker-2	Hypertension-1 Hypertension, CKD-2 Hypertension, Diabetes-3 Hypertension, Obesity-4 Others-5				
1	35	1	220	130	0	1	2	1	109		Has Stroke	T
2	42	1	200	120	0	1	1	1	110		Has Stroke	T
3	45	1	160	120	0	1	0	1	77		Has Stroke	T
4	45	1	130	90	0	0	0	2	105		Has Stroke	F
5	48	1	130	100	0	1	0	2	85		Has Stroke	T
6	49	1	170	110	0	0	0	1	86		Has Stroke	T
7	50	1	240	130	0	1	0	1	140		Has Stroke	T
8	52	1	185	132	0	0	0	1	65		Has Stroke	T
9	54	1	210	150	0	1	0	1	122		Has Stroke	T
10	56	1	180	120	0	2	0	3	158		Has Stroke	T
11	58	2	170	110	0	0	0	1	100		Has Stroke	T
12	62	1	160	110	0	0	0	1	125		Has Stroke	T
13	62	2	120	80	0	0	0	1	80		Has Stroke	F
14	64	2	130	90	0	0	0	3	117		Has Stroke	T
15	64	2	148	66	0	0	0	1	128		Has Stroke	F
16	65	1	160	100	0	0	0	1	134		Has Stroke	T
17	65	2	210	110	0	0	0	1	126		Has Stroke	T
18	65	1	140	90	0	1	1	1	137		Has Stroke	T
19	67	2	130	90	0	0	0	1	141		Has Stroke	T
20	68	1	200	120	2	1	0	1	104		Has Stroke	T
21	69	2	130	80	0	0	0	1	117		Has Stroke	F
22	69	1	180	100	0	0	0	1	104		Has Stroke	T
23	70	1	200	120	0	1	0	1	192		Has Stroke	T
24	70	2	180	110	2	0	0	1	101		Has Stroke	T
25	70	2	160	90	0	0	0	3	133		Has Stroke	T
26	70	1	150	110	0	0	0	1	103		Has Stroke	T
27	70	2	170	100	0	0	0	3	240		Has Stroke	T
28	70	2	130	80	0	0	0	3	148		Has Stroke	F
29	75	1	160	100	0	1	0	3	143		Has Stroke	T
30	78	1	160	110	0	0	0	1	114		Has Stroke	T
31	55	1	140	110	2	1	0	3	181		Has Stroke	T
32	23	1	140	100	1	1	2	4	179		Has Stroke	T
33	20	1	160	120	1	1	0	5	96		Has Stroke	T
34	18	1	154	137	2	1	0	4	97		Has Stroke	T
35	60	1	160	110	2	0	0	3	184		Has Stroke	T
36	61	2	150	100	1	1	0	1	96		Has Stroke	T
37	65	1	210	110	2	1	0	1	101		Has Stroke	T
38	42	1	160	110	2	1	0	1	90		Likely to Have Stroke	T
39	47	2	150	100	1	0	0	1	97		Likely to Have Stroke	T
40	59	1	160	100	1	0	0	1	130		Likely to Have Stroke	T
41	72	2	160	100	1	0	0	1	99		Likely to Have Stroke	T
42	47	1	140	100	1	1	0	4	79		Likely to Have Stroke	F
43	55	1	140	110	2	1	2	1	76		Likely to Have Stroke	T
44	62	2	170	110	2	0	0	3	132		Likely to Have Stroke	T
45	46	2	200	120	1	0	0	1	60		Likely to Have Stroke	T
46	47	2	150	100	2	0	0	1	74		Likely to Have Stroke	T
47	63	1	150	100	1	0	2	1	83		Likely to Have Stroke	T
48	68	1	150	100	2	1	0	1	126		Likely to Have Stroke	F
49	65	1	150	100	2	1	0	1	116		Likely to Have Stroke	T
50	72	1	180	110	1	1	0	3	127		Likely to Have Stroke	T
51	35	1	152	100	0	0	0	1	90		Likely to Have Stroke	T
52	56	1	140	100	2	1	0	3	180		Likely to Have Stroke	T
53	52	2	140	98	2	0	0	1	86		Likely to Have Stroke	T
54	69	1	180	100	2	0	0	1	76		Likely to Have Stroke	T
55	53	2	130	100	1	0	0	3	192		Likely to Have Stroke	F
56	58	2	150	100	1	0	0	1	88		Likely to Have Stroke	T
57	57	1	149	94	1	1	0	1	66		Not Likely to Have Stroke	T
58	59	1	160	110	1	1	0	1	98		Not Likely to Have Stroke	T
59	45	1	190	120	2	0	0	3	112		Not Likely to Have Stroke	F
60	72	1	170	110	1	1	0	1	100		Not Likely to Have Stroke	F
61	42	2	110	80	0	0	0	4	91		Not Likely to Have Stroke	T
62	19	1	141	80	0	0	0	0	97		Not Likely to Have Stroke	T
63	39	1	140	85	0	0	0	4	91		Not Likely to Have Stroke	T
64	50	2	140	90	2	0	0	0	98		Not Likely to Have Stroke	T
65	20	1	130	90	0	0	0	0	94		Not Likely to Have Stroke	T
66	20	1	120	70	0	0	0	0	88		Not Likely to Have Stroke	T
67	23	1	140	80	0	0	0	0	93		Not Likely to Have Stroke	T
68	25	1	110	70	0	0	0	0	90		Not Likely to Have Stroke	T
69	21	1	160	80	0	0	0	0	71		Not Likely to Have Stroke	F
70	27	1	120	80	0	0	0	0	88		Not Likely to Have Stroke	T
71	43	1	130	80	0	0	0	0	90		Not Likely to Have Stroke	T
72	65	1	150	90	0	0	0	1	87		Not Likely to Have Stroke	T

73	46	2	130	90	1	0	0	0	73	Not Likely to Have Stroke	T
74	42	2	120	80	2	0	0	0	78	Not Likely to Have Stroke	T
75	54	1	140	90	1	1	0	0	85	Not Likely to Have Stroke	F
76	41	1	130	90	0	0	0	0	114	Not Likely to Have Stroke	T
77	43	2	110	70	2	0	0	0	82	Not Likely to Have Stroke	T
78	32	2	130	80	2	0	0	4	128	Not Likely to Have Stroke	T
79	50	2	130	90	1	0	0	0	98	Not Likely to Have Stroke	T
80	60	1	120	70	0	0	0	0	88	Not Likely to Have Stroke	T

***** FSB =Fasting Blood Sugar RSB =Random Blood Sugar

Table 2. Results for 100 experiments on test data 0020

Experiment	Number of Test Data Predicted Correctly	Time (ms)	Percentage Accuracy
1	68	4	85%
2	72	4	90%
3	72	4	90%
4	70	4	87.5%
5	68	4	85%
6	68	4	85%
7	68	4	85%
8	74	4	92.5%
9	70	4	87.5%
10	72	4	90%
11	68	4	85%
12	75	4	93.75%
.	.	.	.
.	.	.	.
.	.	.	.
100	74	4	92.5%
Total	7,258	4	9,072.5
Average	72.58		90.73%

Model Evaluation: The stroke prediction system developed using a neural network with a sigmoid activation function was evaluated using multiple metrics: Accuracy, precision, recall (sensitivity), F1 score, and AUC-ROC. These metrics provide a comprehensive assessment of the model's performance and effectiveness as shown in Table 3.

Table 3. Performance metrics of the neural network model for stroke prediction

Metric	Accuracy	Precision	Recall	F1 Score	AUC-ROC
Value	0.89	0.85	0.88	0.86	0.91

The neural network model achieved an accuracy of 0.89, indicating that it correctly predicted 89% of the cases. A precision of 0.85 suggests that 85% of the predicted positive cases were true positives. With a recall of 0.88, the model successfully identified 88% of the actual stroke cases. The F1 score of 0.86 balances precision and recall, and the AUC-ROC score of 0.91 confirms the model's excellent ability to distinguish between stroke and non-stroke cases. These metrics collectively show the model's effectiveness and reliability for stroke prediction.

4. DISCUSSION OF RESULTS

The Table 2 presents the average performance accuracy of the neural network-based stroke prediction system for 100 epochs. The results show an average prediction accuracy of 90.73% and the time taken to diagnose and predict stroke status for each patient is 4 micro-seconds. This will aid fast diagnosis and prompt treatment of stroke and consequently aid

survival of stroke victims. It is noteworthy that, the model's performance increases with usage most especially when the learning rate is carefully chosen. Once well trained with sufficient data and the learning rate carefully chosen, the artificial neural networks are unique in its ability to generalize well. It has the capacity to map any new input to the relevant output. Unlike other machine learning methods such as random forests, support vector machines, logistic regressions et cetera, which take a single step to find patterns between new input data and trained output, neural networks model finds its own patterns in multiple steps by utilizing recurrent neural network functions to implement feature engineering that learn the internal representation of the data in sequence of steps.

5. CONCLUSION

In this paper, a neural network-based stroke prediction system was developed. The ANN used in this paper is made up of an input layer with ten nodes, three hidden layers each with eight nodes and an output layer with three nodes. Live data collected from four hospitals in the south-western part of Nigeria were used to train and test the performance of the system. Though, the data do not represent the whole range of human variety, the results obtained show that the system has an average performance accuracy of 90.73% and that it takes fraction of a second to complete the diagnosis of a patient. Since time is an essence in the diagnosis and treatment of stroke, the stroke prediction system reported in this paper is recommended for use in the diagnosis and serve as partners to human experts in the treatment of stroke. It was observed that, the ANN model as used in this paper can learn relevant features from previously labeled data in the same way the human brain learns and has outstanding prediction ability especially when dealing with a completely new task or problem. The contribution of this work is that, a data-driven stroke prediction system that is amenable to new stroke symptoms and which would not require the system to be tuned each time a new symptom is discovered has been developed. It is noteworthy that the ANN model used in this paper was trained using supervised learning paradigm. This is a weakness because the main aim of machine learning is that the machine can learn by itself [16]. The stroke prediction system can therefore be significantly improved upon if further research is done to make the training and learning tasks unsupervised. In the future, the authors hope to explore a suitable unsupervised learning algorithms to carry out the same research on the same datasets.

ACKNOWLEDGMENT

The authors hereby acknowledge Covenant University Centre for Research, Innovation and Discovery (CUCRID) for their support toward the completion of this research.

REFERENCES

- [1] Mokli, Y., Pfaff, J., Dos Santos, D.P., Herweh, C., Nagel, S. (2019). Computer-aided imaging analysis in acute ischemic stroke-background and clinical applications. *Neurological Research and Practice*, 1(1): 23. <https://doi.org/10.1186/s42466-019-0028-y>
- [2] McIntosh, J. (2023). Everything you need to know about stroke. *MedicalNewsToday*, <https://www.medicalnewstoday.com/articles/7624#definition>.
- [3] Ropper, A.H., Brown, R.H. (2005). *Adams and Victor's Principles of Neurology* (8th Edition), McGraw-Hill Professional, pp. 660-746.
- [4] Campbell, B.C., Ma, H., Ringleb, P.A., Parsons, M.W., Churilov, L., Bendszus, M., Levi, C.R., Hsu, C., Kleinig, T.J., et al. (2019). Extending thrombolysis to 4·5-9 h and wake-up stroke using perfusion imaging: A systematic review and meta-analysis of individual patient data. *The Lancet*, 394(10193): 139-147. [https://doi.org/10.1016/S0140-6736\(19\)31053-0](https://doi.org/10.1016/S0140-6736(19)31053-0)
- [5] Cheon, S., Kim, J., Lim, J. (2019). The use of deep learning to predict stroke patient mortality. *International Journal of Environmental Research and Public Health*, 16(11): 1876. <https://doi.org/10.3390/ijerph16111876>
- [6] Eweoya, I.O., Odetunmbi, O.A., Odun-Ayo, I.A., Agbele, K.K., Adedotun, A.F., Akingbade, T.J. (2023). Machine learning approach for the prediction of COVID-19 spread in Nigeria using sir model. *International Journal of Sustainable Development & Planning*, 18(12): 3783-3792. <https://doi.org/10.18280/ijstdp.181210>
- [7] Atomsa, Y., Muhammad, L.J., Ishaq, F.S., Abdullahi, Y. (2021). Expert system for diagnosis of coronary artery disease: A survey. *Journal of Clinical Images and Medical Case Reports*, 2(5): 1342.
- [8] Spitzer, K., Thie, A., Caplan, L.R., Kunze, K. (1989). The microstroke expert system for stroke type diagnosis. *Stroke*, 20(10): 1353-1356. <https://doi.org/10.1161/01.STR.20.10.1353>
- [9] Benjamin, E.J., Muntner, P., Alonso, A., Bittencourt, M. S., Callaway, C.W., Carson, A.P., Chang, C.A.M.A.R., Cheng, S.M.M.F., Das, S.R., Dellings, F.N., et al. (2019). Heart disease and stroke statistics-2019 update: A report from the American Heart Association. *Circulation*, 139(10): e56-e528. <https://doi.org/10.1161/CIR.0000000000000659>
- [10] Zhong, G., Wang, L.N., Ling, X., Dong, J. (2016). An overview on data representation learning: From traditional feature learning to recent deep learning. *The Journal of Finance and Data Science*, 2(4): 265-278. <https://doi.org/10.1016/j.jfds.2017.05.001>
- [11] Jonathan, O., Misra, S., Osamor, V. (2021). Comparative analysis of machine learning techniques for network traffic classification. *IOP Conference Series: Earth and Environmental Science*, 655(1): 012025. <https://doi.org/10.1088/1755-1315/655/1/012025>
- [12] Eluyode, O.S., Akomolafe, D.T. (2013). Comparative study of biological and artificial neural networks. *European Journal of Applied Engineering and Scientific Research*, 2(1): 36-46.
- [13] Kumar, M., Khatri, S.K., Mohammadian, M. (2022). Predicting cancer survival using multilayer perceptron and high-dimensional SVM kernel space. *Ingénierie des Systèmes d'Information*, 27(5): 829-834. <https://doi.org/10.18280/isi.270517>
- [14] Basheer, I.A., Hajmeer, M. (2000). Artificial neural networks: Fundamentals, computing, design, and application. *Journal of Microbiological Methods*, 43(1): 3-31. [https://doi.org/10.1016/S0167-7012\(00\)00201-3](https://doi.org/10.1016/S0167-7012(00)00201-3)
- [15] Rachmad, A., Syarief, M., Hutagalung, J., Hernawati, S., Rochman, E.M.S., Asmara, Y.P. (2024). Comparison of CNN architectures for mycobacterium tuberculosis classification in sputum images. *Ingénierie des Systèmes d'Information*, 29(1): 49-56. <https://doi.org/10.18280/isi.290106>
- [16] Szczepaniuk, H., Szczepaniuk, E.K. (2022). Applications of artificial intelligence algorithms in the energy sector. *Energies*, 16(1): 347. <https://doi.org/10.3390/en16010347>
- [17] Abiodun, O.I., Jantan, A., Omolara, A.E., Dada, K.V., Umar, A.M., Linus, O.U., Arshad, H., Kazaure, A.A., Gana, U., Kiru, M.U. (2019). Comprehensive review of artificial neural network applications to pattern recognition. *IEEE Access*, 7: 158820-158846. <https://doi.org/10.1109/ACCESS.2019.2945545>
- [18] Jawad, W.K., Kaittan, N.M., Sabri, B.T. (2024). Neuronal network-Founded machine knowledge with pythons in data mining for vast information classifications. *Ingénierie des Systèmes d'Information*, 29(1): 141-146. <https://doi.org/10.18280/isi.290115>
- [19] Enoma, D.O., Bishung, J., Abiodun, T., Ogunlana, O., Osamor, V.C. (2022). Machine learning approaches to genome-wide association studies. *Journal of King Saud University-Science*, 34(4): 101847. <https://doi.org/10.1016/j.jksus.2022.101847>
- [20] Zador, A.M. (2019). A critique of pure learning and what artificial neural networks can learn from animal brains. *Nature Communications*, 10(1): 3770. <https://doi.org/10.1038/s41467-019-11786-6>
- [21] Pratiwi, H., Windarto, A.P., Susliansyah, S., Aria, R.R., Susilowati, S., Rahayu, L.K., Fitriani, Y., Merdekawati, A. Rahadjeng, I.R. (2020). Sigmoid activation function in selecting the best model of artificial neural networks. *Journal of Physics: Conference Series*, 1471(1): 012010. <https://doi.org/10.1088/1742-6596/1471/1/012010>
- [22] Adebola, O.K., Charles, A.O., Kayode, A.B. (2012). Development of an expert system for message routing in a switched network environment. *Annals Computer Science Series*, 10(2): 61-70.