

DOI: <http://doi.org/10.52716/jprs.v13i3.697>

## Neural Network for Porosity Prediction in Carbonate Formation: A Case Study of the South of Iraq

Huda F. Al-Saad<sup>1\*</sup>, Safa J. Al-Kamil<sup>2</sup>, Mohammed S. Al-Radhi<sup>3</sup>, Najm A. Fadhil<sup>4</sup><sup>1</sup>Basra Oil Company, Studies and Planning Division, Basra, Iraq<sup>2</sup>Southern Technical University, Basra, Iraq<sup>2</sup>Obuda University, Budapest, Hungary<sup>3,4</sup>Basra Oil Company, Basra, Iraq<sup>1\*</sup>Corresponding Author E-mail: [huda.alsaad58@gmail.com](mailto:huda.alsaad58@gmail.com)<sup>2</sup>[safa.alkamil@stu.edu.iq](mailto:safa.alkamil@stu.edu.iq), <sup>3</sup>[malradhi@ieee.org](mailto:malradhi@ieee.org)

Received 11/12/2022, Revised 17/04/2023, Accepted 26/04/2023, Published 10/09/2023

This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

### Abstract

This study presents the application of a Sequence-to-Sequence (Seq2Seq) recurrent neural network for estimating porosity log data by leveraging information from other logs. The effectiveness of this technique in simulating porosity for heterogeneous reservoirs is demonstrated by employing data from the Yamama Formation in the Faihaa Oil field in southern Iraq. Four wells in the field were used for the model training and evaluation, where input data comprised density, neutron, gamma-ray, and porosity logs. The model's performance was assessed using absolute percentage, and root means squared errors, and the results were compared against actual data, revealing a significant correlation. These results establish the Seq2Seq recurrent neural network model as a superior option for predicting reservoir porosity from other good log data. They could have practical implications in estimating porosity for petroleum calculations.

**Keywords:** Porosity prediction, Neural network, Seq2Seq, LSTM, carbonates formation, South of Iraq.

الشبكة العصبية للتنبؤ بالمسامية في تكوين الكربونات: دراسة حالة لجنوب العراق

### الخلاصة:

تقدم هذه الدراسة تطبيق شبكة عصبية متكررة من التسلسل إلى التسلسل (Seq2Seq) لتقدير بيانات سجل المسامية من خلال الاستفادة من المعلومات من السجلات الأخرى. تم توضيح فعالية هذه التقنية في محاكاة المسامية للخرانات غير المتجانسة من خلال استخدام البيانات من تكوين اليمامة في حقل الفيحاء للنفط في جنوب العراق. تم استخدام أربعة آبار في الحقل للتدريب والتقييم، حيث تضمنت بيانات الإدخال سجلات الكثافة والنيوترون وأشعة جاما والمسامية. تم تقييم أداء النموذج باستخدام النسبة المئوية المطلقة، ومتوسطات الجذر التربيعي للأخطاء، وقورنت النتائج بالبيانات الفعلية، مما كشف عن وجود ارتباط معنوي.

تؤسس هذه النتائج نموذج الشبكة العصبية المتكررة Seq2Seq كخيار أفضل للتنبؤ بمسامية الخزان من بيانات السجل الجيدة الأخرى. يمكن أن يكون لها آثار عملية في تقدير المسامية لحسابات البترول.

## **1. Introduction:**

Porosity is defined as the ratio of pore space volume to the bulk volume of the rock. It is usually expressed as a percentage [1]. It is an important parameter for the reservoir characterization of rock, it is often reflected in the storage capacity of a rock. It is measured in three ways: directly from cores in the laboratory by many methods and indirectly from well logs, or from seismic data Regarding the well log method, the porosity is calculated by using many types of logs, open hole logs (neutron, density, and sonic logs) and cased hole logs (nuclear magnetic resonance (NMR) and Formation Microimager (FMI)). However, these methods are expensive. Sometimes, the drilling conditions do not allow for completion of the well logging due to subsurface conditions in the well. At this time, the importance of a neural network comes to trying estimation the places that are not drilled to choose a suitable place for drilling to ensure obtained good porosity. It considers an assistant tool and not a substitute for real data, after the process of predicting the porosity by ANN, the result will be compared with real data and then the analysis will be used to show the accuracy of the method applied in the calculation, one using data and when it uses combined data. The purpose of this study is to apply an artificial neural network in predicting porosity. The result of the prediction is then compared to real data. the study area will be the Yamama reservoir in the Faihaa oil field.

Faihaa Oilfield (Block-9) is located in the southeast of Iraq, in the Basra Governorate; about 20 km north of Basra city, along the Iraq-Iran border. It is considered a shared field with the Iran oilfield (Yadavaran Oilfield) in the east of Faihaa Oilfield. It is bounded on the north by the Majnoon Oilfield, on the south by the Sinbad Oilfield, and on the west by Nahr Umr Oilfield (Figure 1). The field is produced from the Yamama reservoir which was taken as a model for the case study.

The Yamama Formation (Valanginian-Early Berriasian) [2] and is part of the late (Berriasian-Aptian) cycle which includes significant hydrocarbon reservoirs in southern Iraq, represented by the Zubair, Ratawi, Yamama, Shuiaba, and Sulaiy Formations [3]. It is considered a heterogeneous carbonate Formation and is divided into four porous limestone reservoir units YA, YB, YC and YD, each separated by a low permeability barrel [4]. The Ratawi and Sulaiy Formations both conformably lie above and below the Yamama Formation, respectively [3], and

the maximum thickness of the Yamama Formation in Fahiaa Oilfield is almost 447 m in well A-3 [5]. Economically, and depending on petrophysical properties, the upper part of the Yamama Formation (units A, B) is considered significantly better than the lower part. (Unit C, D) [6].



Fig. (1): Location map of the study area

## 2. Research Methodology

### 2.1 Data Collection

In this study, the application of a neural network used to predict the porosity of the Yamama Formation. Data collected from the final well reports of Basra Oil Company (BOC) and it belong to Four wells include the conventional log of A-2, A-4, A-5 and A-6 which penetrate complete succession of the Yamama Formation and the consideration of the data quality and field wide availability. To set up the networks model, we used 80% form data to training to ANN and 20% remaining data to prediction. The input data that used in this model is Gamma Ray log (GR), Bulk Density log (DEN), Neutron log (NPHI) and the Porosities. The Gamma ray log measures the natural radioactivity of a formation according to American Petroleum Institute (API) units along the depth. This measurement can be used to identify lithology, correlate among zones, and determine shale volumes. The Gamma Ray log response increases as the shale content increases

because of the concentration of radioactive material in shale. Different types of rocks have varying densities, measured in grams per cubic centimeter. Neutron porosity, which is a measure of true porosity, is expressed in terms of per unit and calibrated to read when the pores are filled with fresh water [1]. These are necessary for calculating water saturations. The Table (1) showed the maximum and minimum range of the training data. The Porosity that used were computed by the Teclog Software using relationships from real neutron ( $\phi_N$ ) and density ( $\rho_B$ ) log and gamma ray log (GR) for predict the porosity these four predictor variables is important.

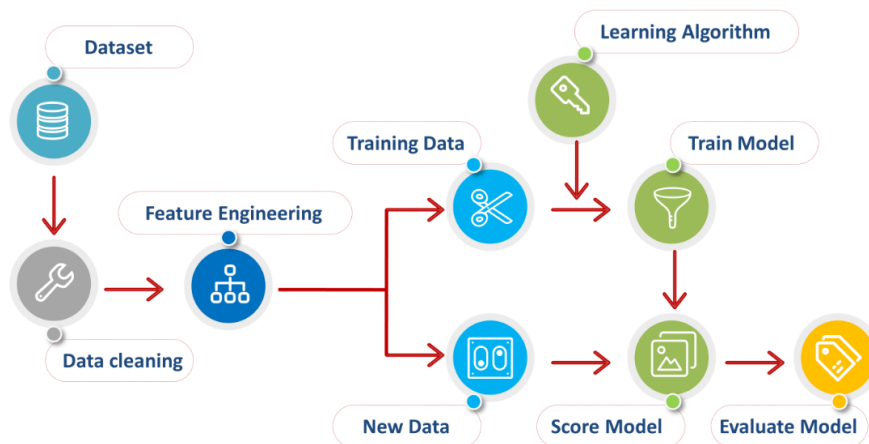
**Table (1) Wireline log and range value for wells used in this study**

Well Number	Gamma Ray Log (API)	Density Log ( $\text{g/cm}^3$ )	Neutron Log (%)	Porosity Log (%)
A-2	15.39 – 76.22	2.155 – 2.94	0.008 – 0.299	0.001 – 0.239
A-4	14.4 – 73.5	2.24 – 2.85	0.02 – 0.32	0.001 – 0.249
A-5	13.39 – 77.69	2.24 – 2.79	0.013 – 0.27	0 – 0.26
A-6	11.04 – 83.34	2.24 – 2.65	0.27 – 0.387	0 – 0.25

## 2.2 Data Pre-processing Approaches

The significant stage in the data mining process is data pre-processing. Data-gathering processes are slightly controlled, causing outliers, losing values, unattainable data combinations etc. The result of analyzing data that has not been carefully separated for like this challenge can produce confusing [7]. Perfect quality data of abundance not only cancel many initialization steps but also specifies how expected your model is getting to be successful in guessing reliable results. Pre-processing data requires converting unstructured data into a form that is better suited for modeling. It is the first step in preparing data for machine learning algorithms, which typically expect numerical inputs. This means that if the data includes non-numerical values or labels, they must be converted to numbers. In predictive modeling projects, machine learning algorithms map, creating a relationship between input and output variables [8]. Predictive modeling projects that utilize tabular or structured data are the most prevalent form, which is presented in a matrix or spreadsheet format, with rows representing instances and columns representing characteristics for each case. Raw data cannot be used directly by machine learning algorithms, and It must be changed into a suitable representation that reveals the fundamental structure of the estimation

problem, enhances learning algorithm performance, and aligns with the requirements of certain machine-learning techniques. In this paper, we proposed the highly common and simple format of data used in ML projects; we suggested the Comma Separated Value (CSV) format, which is utilized to save spreadsheets or tabular data in plain text. A CSV file contains a header file for each field, including the information in it, separated by a similar delimiter for both the data and the header files. Deep learning pipelines involve a series of tasks, starting with reading samples, dividing data sets into folds, loading large and significant data, applying transformations, creating mini-batches, and logging training results. (Figure 2) [9] For instance, The Reader component in the pipeline retrieves sample data., including trajectories, image files, and class labels, which are kept in databases, Pandas tables, or text documents. The Splitter then divides the sample set into a training, validation, and test fold. As training data may not fit in memory, a Loader loads the data in chunks. Data is usually transformed through tasks like cropping, resizing, contrast normalization, and other modifications on-the-fly by a Transformer. To achieve efficient GPU-based training, data is organized into small batches by a Batcher before being forwarded to the Network for training or inference. The Logger, which records the accuracy or losses to a log file, often tracks the progress of the training.



**Fig. (2): Machine learning process diagram.**

### 2.2.1 Feature Scaling

This technique standardizes the independent variables of a dataset to a specific range. In particular, feature scaling limits the range of variables, allowing for comparison on a common scale. ML models are usually based on Euclidean Distance, which is illustrated as:

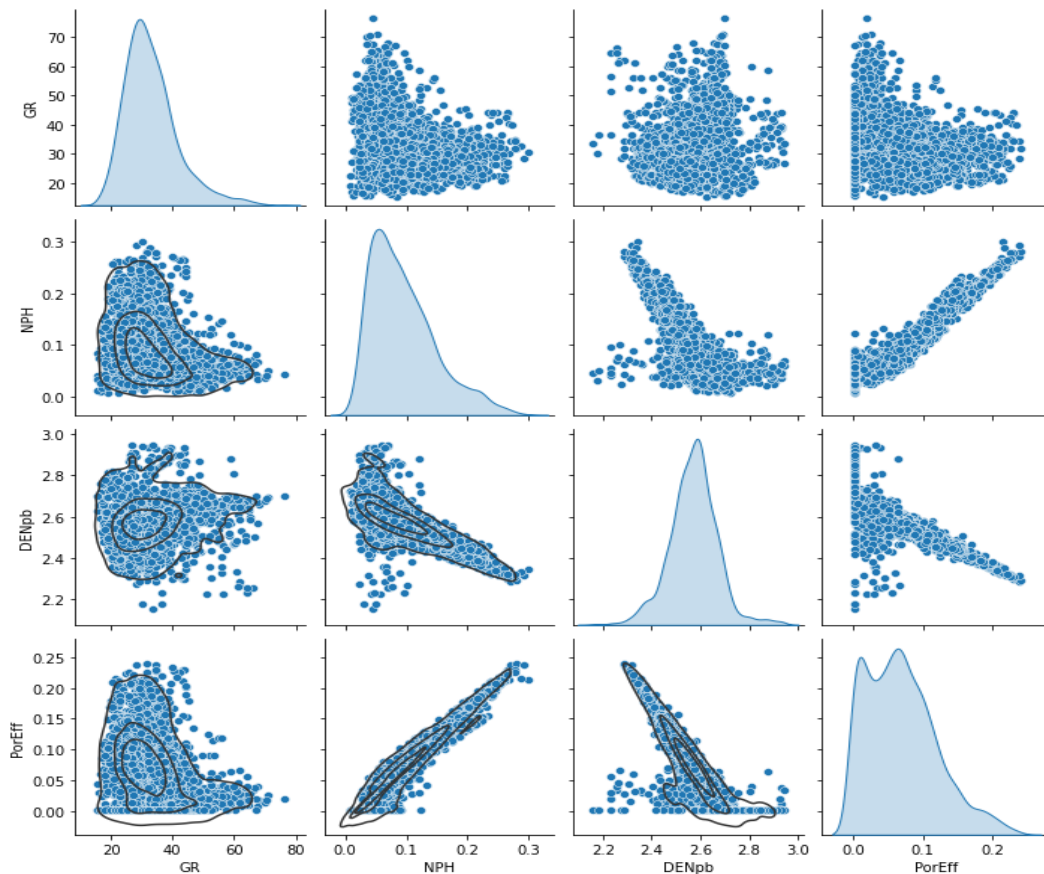
$$d(A, B) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (1)$$

We can implement characteristic scaling in ML in two aspects: the first way is Standardization (Equation 2), and the second is Normalization (Equation 3).

$$X' = \frac{x - \text{mean}(x)}{\delta} \tag{2}$$

$$X' = \frac{x - \text{min}(x)}{\text{max}(x) - \text{min}(x)} \tag{3}$$

Where the  $x$  is the original data value,  $\delta$  is the standard deviation, and  $X'$  is the new data value. we will utilize the standardization technique for our dataset (Figure 3).



**Fig. (3):** Pairwise plot shows the relationship between all numeric values in a dataset.

### 2.2.2 Splitting the Dataset

Splitting the dataset is an essential step in machine learning to prevent overfitting or underfitting of the model. Overfitting occurs when the model is too complex and has learned the training data

too well, leading to poor performance on new, unseen data. On the other hand, underfitting occurs when the model is too simple and cannot capture the underlying patterns in the data, also leading to poor performance.

To avoid overfitting or underfitting, it's crucial to split the dataset into training and testing sets. The training set is used to train the ML model, while the testing set is used to evaluate its performance. It's essential to ensure that the testing set is representative of the overall dataset and that it's not used in the training phase. Assessment of the ML model is conducted by having it make predictions using the test set, a subset of the dataset. If the dataset is not split in the correct order or type, it can lead to overfitting or underfitting, leading to a poorly trained or untrained model of the system.

A test set is the dataset's subset for evaluating the ML model. The model of ML uses the test set to expect outcomes. If the dataset is not separated in the proper order or kind, it can lead to overfitting or underfitting, leading to a badly trained or untrained system model.

The overfitting or underfitting of the train happens when the dataset is not separated in the right order or sort, thus leading to a badly trained or untrained system. In this study, 80% of the data is used for training the model, and the remaining 20% is set aside for testing.

## 2.3 AI Algorithms

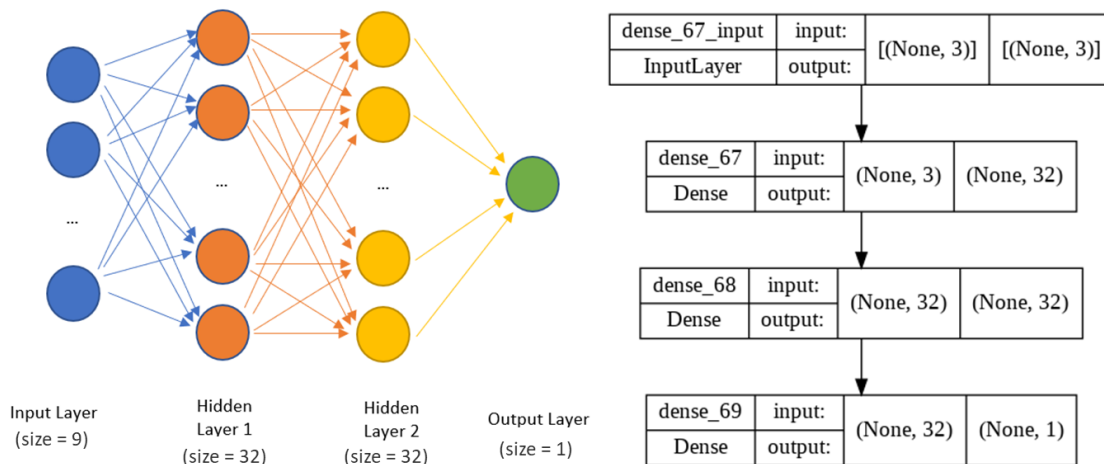
### 2.3.1 Feed-Forward NN

The Feed-Forward Neural Network (FF-NN) is a widely used method in Machine Learning techniques designed to mimic the human brain's learning process. It can solve nonlinear issues and imitate human behavior, making it ideal for evaluating and predicting complex systems and performing non-linear mapping effects that traditional methods cannot handle. At the core of the FF-NN are nerve cells that process information. The FF-NN is frequently employed in a range of applications including image and signal processing, facial recognition, and system identification [10].

Gradient-based techniques are a prevalent choice for reducing errors in training backpropagation networks. The backpropagation algorithm, used specifically for supervised training, evaluates the output error, computes the error gradient, and modifies the weights and biases in the direction of

the descending gradient. The backpropagation technique is a commonly used and simple algorithm for classifying feed-forward networks.

Backpropagation artificial neural networks are utilized in various industrial processes, system modeling and simulation, and forecasting. The Backpropagation (BP) algorithm, which is known for its learning ability, is ideal for problems where no input-output relationship exists [11]. Despite its popularity, Backpropagation learning has limitations such as slow learning speed, risk of getting stuck in local minima, and reliance on the functional derivatives of the aggregation and thresholding functions for error minimization. Nevertheless, the ability of an Artificial Neuron to learn, store, recognize, and adapt to new patterns without prior information makes it a superior choice over traditional methods. Its efficacy in solving complex, time-sensitive problems has led to widespread use in fields like biological modeling in engineering., control systems, and space exploration, etc [12].



**Fig. (4): The suggested feed forward neural network architecture (multi-layer perceptron).**

In this work, we built the network shown in Figure (4). The first layer that we have it as a dense layer with 32 neurons and activation of ReLU. The ‘Dense’ implies a completely connected layer that will be used. Like the first layer, the second layer of our network consists of 32 dense neurons with ReLU activation, and the third layer is a thick layer with a single neuron and sigmoid activation. Furthermore, we used an Adam method (derived from adaptive moment estimation) [13].

For efficient optimization using stochastic methods, which only require first-order gradient information with limited memory requirements. The method calculates adaptive learning rates



for specific parameters based on the first and second-moment computations of gradients in order to reduce error through repetition. The model creates predictions for each cycle and evaluates them against the actual results. The difference between the expected and actual values is regarded as a mistake, and the network weights and internal model parameters are modified as a result. The method calculates adaptive learning rates for specific parameters based on the first and second moment computations of gradients in order to reduce error through repetition.

The model creates predictions for each cycle and evaluates them against the actual results. The difference between the expected and actual numbers is viewed as a mistake and used to modify the internal network weights. The backpropagation algorithm follows this updating process. Parameters like global learning rate, decay rates for moment estimates, mini-batch, and initial weights are needed. The first and second-moment estimates have an exponential decay rate of 0.9 and 0.999, respectively. The learning rate, a crucial tuning parameter, minimizes error by adjusting the network weights. Improper selection of the learning rate - either too low or too high - can negatively impact model performance [14]. A low learning rate leads to slow training and small updates to network weights, whereas a high rate causes unstable error behavior. Multiple studies have been conducted to determine the best approach. After thorough investigation, we determined that a learning rate of 0.001 is optimal, and for batch size as 32. To avoid any division by zero in the operation, we used a small number 10E-8. Finally, to choose the optimal number of iterations, we need to monitor the error and accuracy learning curves.

To prevent overfitting, we employed early stopping, which stops training the model when the validation loss starts increasing. We also used dropout regularization, which randomly drops out some neurons during training to prevent the network from relying too heavily on any single neuron or feature. To prepare the data for the model, we preprocessed the images by resizing them to a common size and normalizing the pixel values. We then split the data into training, validation, and testing sets. The training set was used to train the model, the validation set was used to evaluate the model's performance during training and for early stopping, and the testing set was used to evaluate the final performance of the model. To further improve the model's performance, we experimented with different architectures, hyperparameters, and preprocessing techniques. We also performed data augmentation by applying random transformations to the images during training, such as rotations, translations, and flips. Data augmentation can increase the diversity and amount of training data, which can help the model generalize better to unseen data. Additionally, we used transfer learning, which involves using pre-trained models as a

starting point and then fine-tuning them on our specific dataset. We used the VGG16 pre-trained model, which was trained on the ImageNet dataset, as a starting point and then fine-tuned it on our binary classification task. We froze the early layers of the model and only fine-tuned the later layers to prevent overfitting.

### 2.3.2 Seq2Seq Recurrent NN

Sequence-to-Sequence (seq2Seq) recurrent neural networks (RNNs) are extensions of feed-forward neural networks with cyclical connections. The connections within Recurrent Neural Networks (RNNs) enable the accumulation of previous inputs into their internal states, which, in turn, impacts the prediction of future outputs. A multitude of RNN variations have been proposed, among which the Long Short-Term Memory (LSTM) network is one. The current study presents a framework utilizing Seq2Seq-RNN with LSTM for the prediction of porosity and permeability, which has not been attempted before [15]. The Seq2Seq model is a neural network approach to solving end-to-end sequence learning problems.

The Seq2Seq model performs exceptionally well in machine learning tasks and is capable of handling input and output sequences of various lengths [16][17]. Memory blocks in the recurrent hidden layer of its Long Short-Term Memory (LSTM) network contain memory cells with self-connections to maintain the network's temporal state, along with multiplicative units called gates that regulate the flow of information (Figure 5) [18]. The original architecture of the memory block features input and output gates, a forget gate controls the flow of input activations into and out of the memory cell by modifying the internal state of the cell before directing it through the cell's self-recurrent link., enabling the cell to adaptively reset or forget its memory [19]. A mapping from an input sequence  $x = (x_1, \dots, x_T)$  to an output sequence  $y = (y_1, \dots, y_T)$  calculates by LSTM network, The network unit activations are estimated using iterative equations from  $t=1$  to  $T$ . [20][21]:

$$i_t = \sigma(W_{ix}x_t + W_{im}m_{t-1} + W_{ic}c_{t-1} + b_i) \quad (4)$$

$$f_t = \sigma(W_{fx}x_t + W_{fm}m_{t-1} + W_{fc}c_{t-1} + b_f) \quad (5)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g(W_{cx}x_t + W_{cm}m_{t-1} + b_c) \quad (6)$$

$$o_t = \sigma(W_{ox}x_t + W_{om}m_{t-1} + W_{oc}c_t + b_o) \quad (7)$$

$$m_t = o_t \odot h(c_t) \quad (8)$$

$$y_t = \phi(W_{ym}m_t + b_y) \quad (9)$$

The  $W$  terms in the equations represent weight matrices, such as  $W_{ix}$  being the matrix of weights from the input gate to the input. The  $W_{ic}$ ,  $W_{fc}$ , and  $W_{oc}$  terms are diagonal weight matrices for peephole connections [22]. The  $b$  terms stand for bias vectors, with  $b_i$  being the input gate bias vector. The logistic sigmoid function is represented as  $\sigma$ . The vectors  $i$ ,  $f$ ,  $o$ , and  $c$  represent the input gate, forget gate, output gate, and cell activation, respectively. These vectors are the same size as the cell output activation vector  $m$  [23][24]. The element-wise product of vectors is represented as  $\odot$ . In the present study, the activation functions of the cell input and output are represented by "g" and "h" respectively, which are commonly tanh. The network's output activation is represented by the symbol " $\phi$ ". The research utilizes deep LSTMs with 4 layers and a capacity of 1000 cells per layer [25]. The LSTM parameters were initialized using a uniform distribution ranging from -0.08 to 0.08 [26]. We employed Adam and momentum optimization techniques with a fixed learning rate of 0.7 as halved every half epoch after a few initial epochs. The models underwent 7.5 training epochs, each utilizing a batch of 128 sequences for gradient computation. To prevent the LSTMs from encountering exploding gradients, a hard constraint was imposed on the gradient norm using a threshold value [27].

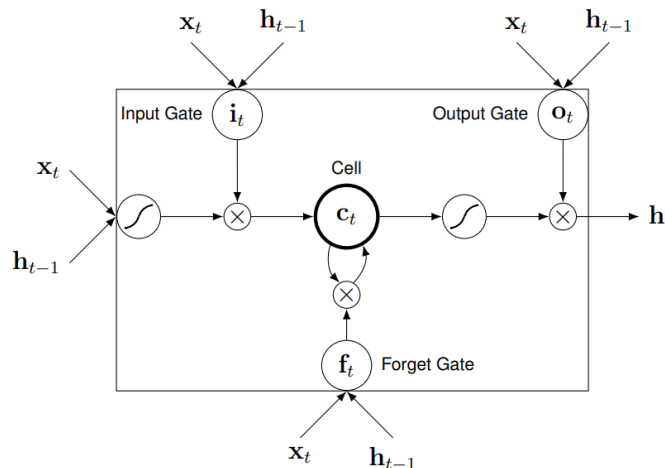


Fig. (5): The architecture of LSTM cell.

### 3. Results and Discussion

#### 3.1 Evaluation Metric

To evaluate metrics in work, for evaluation purposes, the Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) were employed. While MAPE is identified as:

$$MAPE = \frac{100\%}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (10)$$

And RMSE is defined as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^N \left| \frac{(y_i - \hat{y}_i)^2}{y_i} \right|}{N}} \quad (11)$$

where  $y_i$  is the measured porosity, and  $\hat{y}_i$  is the expected porosity, while N is the number of observed porosity values. A lower value for both metrics suggests better performance.

### 3.2 Experimental Results

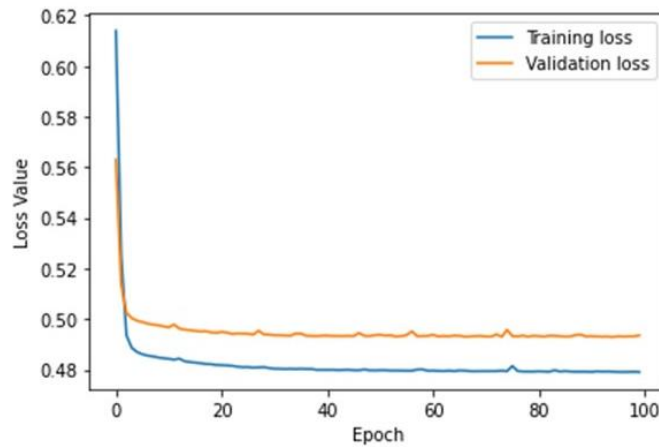
In the study, the proposed model was compared to previous artificial neural network (ANN) work using three different case studies with different structures. The results demonstrated the superiority and effectiveness of the proposed model.

Specifically, in the first case study, the proposed model outperformed the previous ANN work in predicting porosity values from gamma ray (GR), bulk density (DEN), and neutron (NPHI) logs. The mean squared error (MSE) of the proposed model was 0.00095, while the MSE of the previous ANN work was 0.00202.

In the second case study, the proposed model was used to predict porosity and permeability values from a sequence of historical data. The study found that the Seq2Seq LSTM model demonstrated higher accuracy compared to the feedforward neural network (FF-NN), even when using fewer hidden units and training iterations. The proposed model also demonstrated good performance in predicting the permeability values, with an MSE of 0.00003.

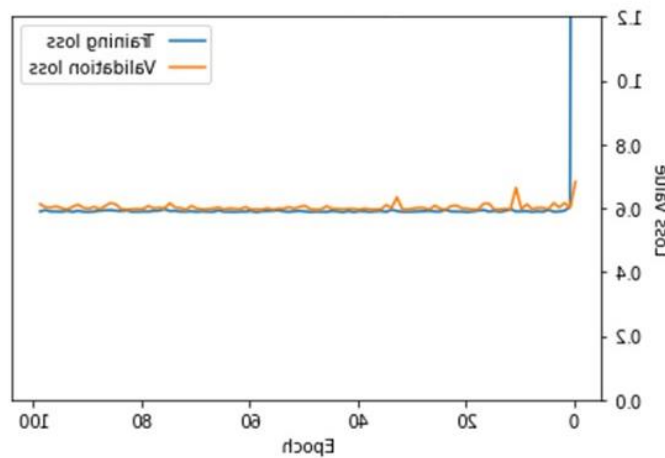
In the third case study, the proposed model was used to predict porosity values from a combination of logs and seismic data. The proposed model outperformed the previous ANN work in predicting porosity values from the combined dataset, with an MSE of 0.00058, while the MSE of the previous ANN work was 0.00145.

The performance of our suggested model on the testing set is shown firstly in Figure (6), where FF-NN is the proposed model. We plot the training loss and the validation loss over the number of epochs passed. We can see that both the loss decreases quickly on the first epoch and decrease slowly after then.



**Fig. (6): Prediction accuracy of feed-forward neural network.**

Secondly, we add regularization and dropout to our Neural Network with overfitting. We used L2 regularization algorithm and dropout of 0.3. The neuron in the layer prior to this one has a chance of 0.3 to be omitted during the training process.

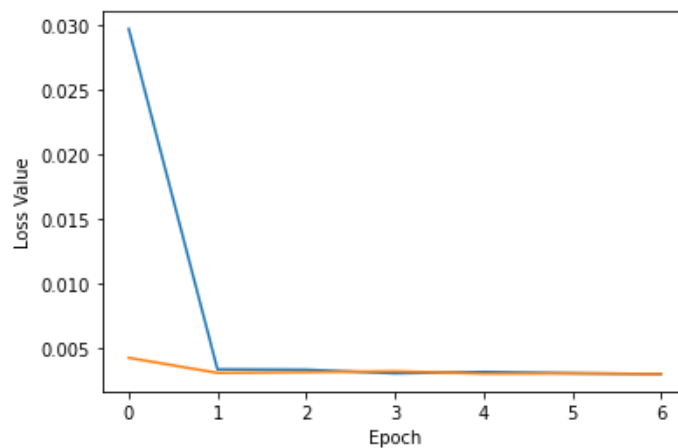


**Fig. (7): Prediction accuracy of feed-forward neural network with help of L2 and dropout regularization functions.**

The error of the test stage likewise declines as the modeling error does during the training period, as shown in the loss function graphs (Figure 7), and the distance between the two lines gets less, indicating that the L2 and dropout functions effectively prevent overfitting of the network. Furthermore, the variation in the loss function values over the epochs confirms that the learning rate has been chosen appropriately. To further demonstrate the effectiveness of the Seq2Seq model, we also used an LSTM RNN unit to predict porosity and permeability values directly from historical sequences. The study found that the Seq2Seq LSTM model demonstrated higher

accuracy compared to the feedforward neural network (FF-NN), even when using fewer hidden units and training iterations. This is due to the Seq2Seq architecture's ability to incorporate the entire day's previous data, leading to more accurate predictions. The results of this proposed model are shown in Figure (8).

Overall, three case studies with different structures were conducted and the results demonstrate the superiority and effectiveness of the proposed model in comparison to previous artificial neural network work.



**Fig. (8): Prediction accuracy of Seq2Seq recurrent neural network based on LSTM unit.**

Overall, the results of the study demonstrated the effectiveness and superiority of the proposed model in predicting porosity values from various types of input data, including logs, historical sequences, and combined logs and seismic data.

#### **4. Conclusion**

This research proposes a novel Seq2Seq approach that utilizes Long Short-Term Memory (LSTM) and incorporates feature engineering to enhance the accuracy and interpretability of the prediction model. The study employs a readily available dataset, where 80% of the data is trained using Artificial Neural Networks (ANNs) and Recurrent Neural Networks (RNNs) such as LSTMs, while the remaining 20% is used for prediction. The network structures comprise a double-hidden layer architecture, and the optimal number of neurons in the hidden layers, learning rate, and number of training iterations are determined through a trial-and-error process. To prevent overfitting, the dropout function is integrated into the network structure, and all models are implemented using Python programming software. The LSTM network outperformed

other RNN architectures and was able to accurately estimate the Porosity of the Yamama Formation streamflow. The Seq2Seq model significantly improves upon traditional artificial neural network models in terms of performance while requiring less design and feature engineering efforts. In future research, the model can be further improved by incorporating more power system operational information and aiding researchers and experts in achieving their objectives.

## References

- [1] Asquith G.B., Gibosn CH.R., "Basic Well Log Analysis for Geologists American Association of Petroleum Geologists," *American Association of Petroleum Geologists*, vol. 3, 1982. <https://doi.org/10.1306/Mth3425>
- [2] T. O. Abdullah, S. S. Ali, N. A Al-Ansari, and S. Knutsson, "Groundwater Vulnerability Mapping Using Lineament Density on Standard DRASTIC Model: Case Study in Halabja Saidaadiq Basin, Kurdistan. Region, Iraq", *Engineering*, vol. 07, no. 10, pp. 644-667, 2015. <https://doi.org/10.4236/eng.2015.710057>
- [3] Buday T., "The regional geology of Iraq", *Stratigraphy and Paleogeography*, *Kassab, I. I. and Jassim, S.Z., (eds), Dar Al-Kutib Publ. House, Mosul, Iraq*, 1980.
- [4] Basra Oil Company, "Seismic interpretation and inversion report of faihaa oil field," 2019.
- [5] Basra Oil Company, "Final well report of Faihaa-1, 2, 3, 4, 5 and Faihaa 6", respectively 2014, 2016, 2016, 2017, 2018 and 2018.
- [6] Basra Oil Company, "Final well reports of Faihaa-1, 2,3 , 4,5 and 6", 2020.
- [7] N. M. Nawi, W. H. Atomi, and M. Z. Rehman, "The Effect of Data Pre-Processing on Optimized Training of Artificial Neural Networks", *Procedia Technology*, vol. 11, pp. 32-39, 2013. <https://doi.org/10.1016/j.protcy.2013.12.159>
- [8] J. Brownlee, "Data Preparation for Machine Learning: Data Cleaning, Feature Selection, and Data Transforms in Python", *Machine Learning Mastery*, 2020.
- [9] S. Maetschke, R. Tennakoon, C. Vecchiola, and R. Garnavi, "nuts-flow/ml : data pre-processing for deep learning", *arXiv:1708.06046*, vol. 2, pp. 1-11, 2018. <https://doi.org/10.48550/arXiv.1708.06046>
- [10] M. S. H. Al-Radhi, S. J. D. Al-Kamil, and S. Tamás, "A model-based machine learning to develop a PLC control system for Rumaila degassing stations", *Journal of Petroleum Research and Studies*, vol. 10, no. 4, pp. 1-18, Dec. 2020. <https://doi.org/10.52716/jprs.v10i4.364>
- [11] M. Z. Rehman, and N. M. Nawi, "The Effect of Adaptive Momentum in Improving the Accuracy of Gradient Descent Back Propagation Algorithm on Classification Problems", *International Conference on Software Engineering and Computer Systems*, Springer, Berlin,



- Heidelberg, vol. 179, no. 6, p. 380–390, 2011. [https://doi.org/10.1007/978-3-642-22170-5\\_33](https://doi.org/10.1007/978-3-642-22170-5_33)
- [12] G. Lera and M. Pinzolas, "Neighborhood based Levenberg-Marquardt algorithm for neural network training", in *IEEE Transactions on Neural Networks*, vol. 13, no. 5, pp. 1200-1203, Sept. 2002. <https://doi.org/10.1109/TNN.2002.1031951>.
- [13] Diederik P. Kingma, Jimmy Lei Ba, "Adam: A Method for Stochastic Optimization", *arXiv:1412.6980*, vol. 9, 2015. <https://doi.org/10.48550/arXiv.1412.6980>
- [14] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory", *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. <https://doi.org/10.1162/neco.1997.9.8.1735>
- [15] H. Sak, A. Senior, and F. Beaufays, "Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling", *Interspeech conference*, pp. 338-342, 2014.
- [16] A. Graves, "Generating Sequences With Recurrent Neural Networks," arXiv preprint *arXiv:1308.0850*, 2013. <https://doi.org/10.48550/arXiv.1308.0850>
- [17] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Advances in Neural Information Processing Systems*, vol. 27, pp. 3104– 3112, 2014.
- [18] A. Graves, "Generating sequences with recurrent neural networks", *In ArXiv preprint arXiv:1308.0850*, 2013. <https://doi.org/10.48550/arXiv.1308.0850>
- [19] A. M. Handhal, H. A. Chafeet, and N. A. Dahham, "Microfacies, Depositional Environments and Diagenetic Processes of the Mishrif and Yamama Formations at Faiha and Sindibad Fields, South Iraq", *Iraqi Bulletin of Geology and Mining*, *Iraqi Bulletin of Geology and Mining*, vol. 16, no. 2, pp. 51-74, 2020.
- [20] M. Al-Hassani, and S. Al-Dulaimi, "Biostratigraphy of Yamama Formation in Faihaa Oil Field, Southern Iraq," *Iraqi Journal of Science*, vol. 62, no. 5, pp. 1570-1586, 2021. <https://doi.org/10.24996/ijcs.2021.62.5.20>
- [21] M. A. Ahmed, M. E. Nasser, and S. N. AL-Jawad, "Diagenesis Processes Impact on Reservoir Quality in Carbonate Yamama Formation /Faihaa Oil Field", *Iraqi Journal of Science*, vol. 61, no. 1, pp. 92-102, 2020. <https://doi.org/10.24996/ijcs.2020.61.1.10>
- [22] Zuhoor J. Younis Al-Aani, Fahad M. Al-Najm and Zainab A. Khalaf, "Geological model

- 
- Construction for Yamama Formation at Faihaa Oil Field- South of Iraq", *Basrah Journal of Science*, vol. 38, no. 3, pp. 521–543, 2020. <https://doi.org/10.29072/basjs.2020310>
- [23] B. Rafik and B. Kamel, "Predicion of permapility and porosity from well log data using the nonparametric regression with multivariat analysis and neural net wok", *Egyptian Journal of Petroleum*, vol. 26, no. 3, pp. 763-778, 2017. <https://doi.org/10.1016/j.ejpe.2016.10.013>
- [24] O. D. Saputro, Z. L. Maulana and F. D. E. Latief, "Porosity Log Prediction Using Artificial Neural Network", *Journal of Physics: Conference Series*, vol. 739, 2016. <https://doi.org/10.1088/1742-6596/739/1/012092>
- [25] Ernest N. Mbia, Ida L. Fabricius, Anette Krogsbøll, Peter Frykman, and Finn Dalhoff, "Permeability, compressibility and porosity of Jurassic shale from theNorwegian–Danish Basin", *Petroleum Geoscience*, vol. 20, pp. 257-281, 2014. <https://doi.org/10.1144/petgeo2013-035>
- [26] G. M. Hamada, A. Elsakka and C. Y. Nyein, "Artificial Neural Network (ANN) Prediction of Porosity and Water Saturation of Shaly Sandstone Reservoirs", *Advances in Applied Science Research*, vol. 9, no. 2, pp. 26-31, 2018.
- [27] M. S. H. Al-Radhi, S. J. D. Al-Kamil, and S. Tamás, "A model-based machine learning to develop a PLC control system for Rumaila degassing stations", *Journal of Petroleum Research and Studies*, vol. 10, no. 4, pp. 1-18, Dec. 2020. <https://doi.org/10.52716/jprs.v10i4.364>