PREDICTION OF CO_2 GAS SATURATION DISTRIBUTION BASED ON DEEP LEARNING USING DEEP NEURAL NETWORK (DNN) ALGORITHM

Eki Komara[1], Zahrotuts Tsaniyah[1], Widya Utama[1]
Geophysical Engineering Department, Faculty of Civil Planning and Geo Engineering, Institut Teknologi Sepuluh Nopember
e-mail : komara@its.ac.id

Abstract. *Multiphase flow analysis is essential for resolving subsurface flow issues in CO_2 capture and storage (CCS) systems. Predicting the distribution of CO_2 gas saturation is one example that is quite useful for evaluating multiphase flow. Multiphase flow simulation is typically performed using numerical simulations, such as the TOUGH2 simulator. Ordinary numerical simulations, on the other hand, have some limitations, such as high grid spatial resolution and significant processing costs. One option for estimating the distribution of CO_2 gas saturation is to employ deep learning with specific algorithms. A deep neural network (DNN) is a highly effective deep learning approach. A deep neural network is a network structure made up of three interconnected layers: input, hidden, and output. DNN learns from the input data about the previously constructed architecture. As input, DNN requires a significant amount of data train. The trained DNN model is then used to automatically estimate the distribution of CO_2 gas saturation. This algorithm is capable of dealing with complex data patterns, particularly gas saturation in multiphase flow issues. The reconstruction loss results revealed that the loss value lowers as the number of epochs grows. Furthermore, the model with 5 epochs and 0.001 regularization weight had the least error value 0.43. As a result, while this model is adequate for predicting the distribution of CO2 gas saturation, additional research is required to achieve more ideal outcomes.*

Keywords: *Multiphase Flow; Deep Neural Network; CO_2 Gas Saturation*

INTRODUCTION

Multiphase flow in porous media is an important consideration in Carbon Capture and Storage (CCS) systems. This is because multiphase flows can be employed to solve transportation and subsurface flow problems (Pachauri et al, 2014). Variations in permeability and capillary pressure can be caused by subsurface geological heterogeneity (Pini et al, 2012).

Effective permeability is vital in multiphase systems for accurate modeling and predicting fluid flow by assessing the fluid's ability to flow (Yang et al, 2023). The presence of a low gas saturation value causes the low permeability value. Low saturation, on the other hand, can diminish the effect of gravitational forces on high permeability layers (Wen and Benson, 2019).

Numerical simulations are typically used to simulate multiphase flows (Pruess, 2005). The TOUGH2 simulator is the most often utilized numerical simulation. Ordinary numerical simulations, on the other hand, have various limitations, including high grid spatial resolution (Doughty, 2010; Wen and Benson, 2019) and significant processing costs (Khebzegga et al, 2020).

The application of the Deep Neural Network (DNN) algorithm is one way devised to overcome the shortcomings of numerical simulation. DNN is a type of artificial neural network with three layers: an input layer, a hidden layer, and an output layer (Addo et al, 2018). According to Wen et al (2021), one of the benefits of utilizing the DNN technique is that the results of the DNN model may produce CO_2 migration forecasts with the same level of accuracy as traditional numerical simulations.

The loss reconstruction function must be used while creating the DNN model to indicate the extent to which the model can accurately rebuild data. The Mean Square Error (MSE) is a popular loss function. This function calculates the difference between the expected and actual values. The lower the values, the higher the model's ability to reconstruct the input data. Given a batch of train data, the parameters are updated to minimize the loss function, and the method is essentially similar to utilizing the objective function in the inverse problem. This function was

Artikel diterima 31 Juli 2023, Revisi 6 Agustus 2023 *Online* 31 Agustus 2023
http://dx.doi.org/10.12962/j25023659.v9i2.18089

choosen because increased CO_2 gas saturation in multiphase flows is frequently associated with higher mobility, which necessitates high precision in plume prediction (Wen et al, 2021).

DNN modeling, on the other hand, employs a 3D temporal architecture that is intended to extract temporal information from the model to be forecasted. This architecture includes a temporal layer that can simulate the depth of the temporal convolution kernel, making it effective for acquiring temporal information in both the short and long term (Diba et al, 2017).

Thus, the objective of this research is to use th Deep Neural Network (DNN) algorithm approach to predict the distribution of CO_2 gas saturation. So it is envisaged that our method would be able to meet the growing demand for analyzing CO_2 storage.

METHODOLOGY

This research uses several software, such as Anaconda (with Python 3.6 environment), Jupyter Notebook, Microsoft Word, and Microsoft Excel. In addition, numerous libraries are used in this study to create the needed DNN architecture. The libraries utilized, as well as their applications, are listed in Table 3.1 below.

Table 3.1**.** Libraries used in Python and their uses

Libraries	**Utilities**
Numpy	For processes that use numbers and arrays
Tensorflow	Assists all numerical computing processes associated with neural networks
Keras	Simplifies the implementation of deep learning algorithms
Matplotlib	Plotting
H5py	Supports readable .HDF5 format in Python

There are various stages in this investigation that lead to the ultimate outcome of CO_2 migration prediction. The workflow corresponds to the research steps depicted in Figures 3.1, 3.2, and 3.3 below.

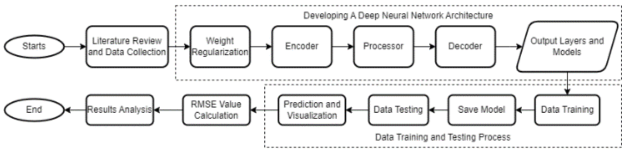

Figure 3.1. Flowchart of overall research

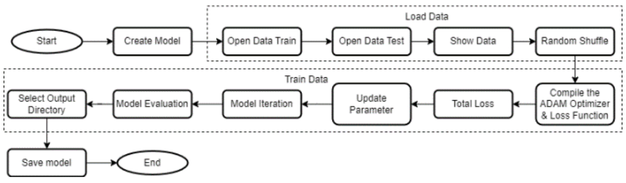

Figure 3.2. Flowchart of data training process

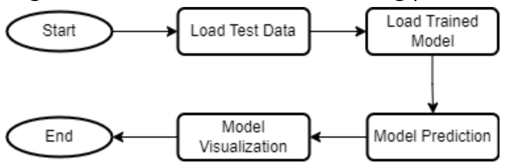

Figure 3.3. Flowchart of data prediction and visualization

Preliminary Studies

This stage includes preliminary studies on CO_2 capture and storage systems (CCS), multiphase flows, gas saturation, as well as Deep Neural Network (DNN) architectures and their constituent components. In addition, at this stage a preliminary study of the application of DNN was also carried out to predict the distribution of CO_2 gas saturation. This stage is useful as suporting data for data interpretation.

Developing DNN Architecture

This stage begins with importing the necessary libraries so that the script can run as needed. Then, define several values and layers that will be used, such as regularization weights, 3D convolution layers (convolution, reflection padding, batch normalization, and ReLU activation function), convolution residual layers (add layer), and non-linear batch nordeconvolution layers (deconvolution, batch normalization, ReLU activation function).

After that, the process of developing the architecture is continued by compiling predetermined layers to form the VAE (Variational Autoencoder) model. Then the step of developing this architecture ends with making the output model.

http://dx.doi.org/10.12962/j25023659.v9i2.18089

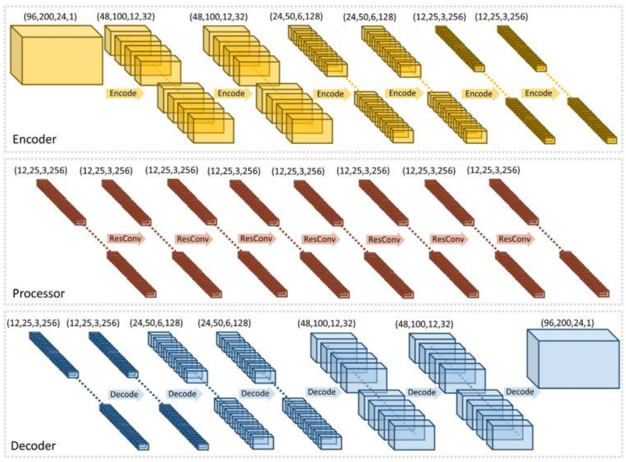

Figure 3.4. Schematic model of 3D temporal architecture

Data Training Process

The training data process begins by importing the appropriate library and inputting the shape based on the architecture model that was created previously. Then proceed with loading the train data and test data, as well as shuffle the data. In this study, X and Y data are included in each train and test data set. The reservoir conditions (initial pressure, temperature, and formation thickness), geological model (permeability), and injection design (injection rate, injection duration, and perforation thickness) data sets comprise the train_x and test_x data sets. Meanwhile, the data train_y and test_y are separating processing data from the Eclipse (e300) software in the form of CO2 gas saturation data with 0 to 1 values. The data set is available at https://drive.google.com/drive/folders/1SVZFkaxkAIjcGKew3rzGTmKW5tSBUGf7?usp=sharing.

After loading the data sets, the process is continued with the process of defining the loss function and determining the training specifications. The training process is continued by carrying out iterations for each epoch and batch, which ends with storing the model resulting the model resulting from the training data.

Data Prediction and Visualization Process

The initial stage in the data prediction and visualization process is to import the required library. Then the process is continued by loading test data to retrieve test data that will be predicted by the model. After that, load the trained model as a result of the previous training process. The prediction process is carried out by taking test data and data from the trained model. Finally, a plot of the predicted results is performed to display the visualization of the model.

RESULT AND DISCUSSION

The number of parameters obtained throughout the DNN model construction process can be utilized to analyze the model. The greater the number of parameters in a layer, the more complex the layer. When the model includes complicated layers, it can learn complex features from the data. Overfitting can also be caused by layers that are excessively complicated. Overfitting occurs when a model memorizes too much data and is unable to generalize to new data. The highest number of parameters that may be handled in this study is 40,399,489, with a total number of parameters that can be trained of 40,386,817. Because the model includes numerous layers, it is more prone to overfitting, according to the data.

Meanwhile, the ratio of train and test data used, and the number of epochs are among the hyperparameter values that may be studied using this DNN model. The ratio between train and test data used in this study is 70:30. This is due to the bulk of the train and test data being too enormous for the computer to load all of it.

The number of epochs used can also have an impact on the training process and outcomes. The more epochs employed, the more patterns in the training data set the model may learn through suitable repetition. As a result, in this study, epochs 3 and 5 were utilized, considering the amount of data, period of research, and computer performance.

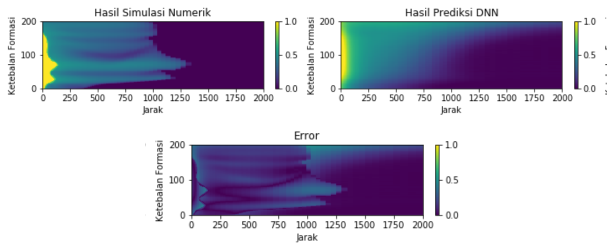

Figure 4.1. Comparison between numerical simulation output, prediction output by DNN, and error with epoch 3 and regularization weight 0.00001

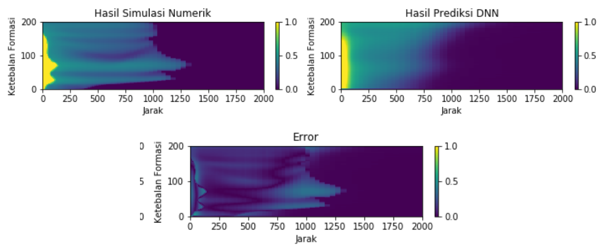

Figure 4.2. Comparison between numerical simulation output, prediction output by DNN, and error with epoch 5 and regularization weight 0.00001

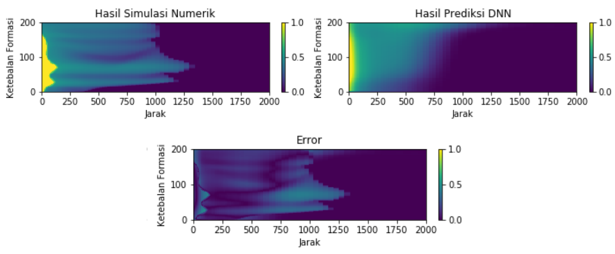

Figure 4.3. Comparison between numerical simulation output, prediction output by DNN, and error with epoch 3 and regularization weight 0.001

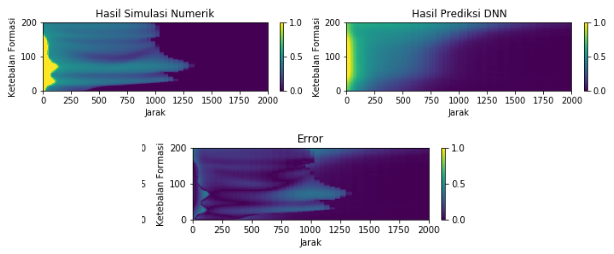

Figure 4.4. Comparison between numerical simulation output, prediction output by DNN, and error with epoch 5 and regularization weight 0.001

The script used in this investigation to forecast the distribution of CO_2 gas saturation is a version of the script developed by Wen et al (2021) and can be downloaded at the following link (https://github.com/gegewen/ccsnet_v1.0). The distribution visualization findings in Figure 4.1 reveal that high saturation is marked in yellow on the left. The saturation distribution then expands to the right, with the lower saturation value moving to the right.

Likewise, the saturation distribution visualization findings in Figure 4.2 reveal that the saturation distribution expands to the right, with high saturation values on the left. Because the number of epochs employed differs just slightly, the predicted model results in Figures 4.1 and 4.2 appear identical at first glance. Similarly, because the variation in the number of epochs is not too great, the visualization results of the distribution in Figures 4.3 and 4.4 often look comparable.

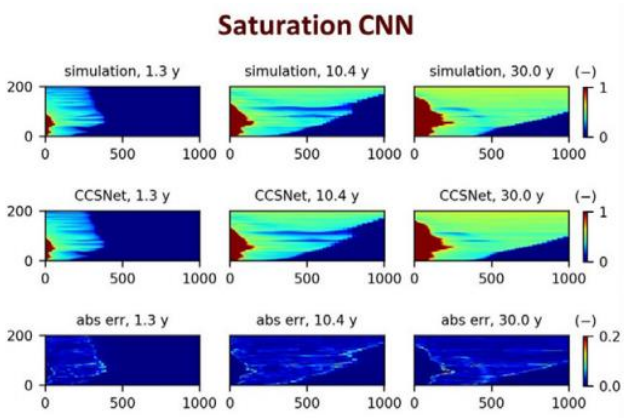

Figure 4.5. Visualization of prediction results in the research of Wen et al (2021)

Figure 4.5 shows three groups of result images: the first is the projected result for 1.3 years, the second is the predicted result for 10.4 years, and the third is the predicted result for 30 years. There are three resulting images in each of these image groups: numerical simulation results (top), Convolutional Neural Network (CNN) prediction outcomes (middle), and errors (bottom).

Figure 4.1 shows the results of this visualization when compared to the results in the manuscript produced by Wen et al (2021), there is a substantial difference. The manuscript's representation of the distribution prediction findings looks exactly like numerical simulation results; hence the error is small. In this investigation, however, the prediction outcomes were very different from the initial input. The variation in results is due to differences in numerous parameters used. Differences in the number of epochs, regularization weight values, batch sizes, and test numbers are examples of these differences. This demonstrates that adjusting the hyperparameter influences the prediction results.

Quantitative analysis necessitates the usage of parameters often found in deep learning. The parameter employed in this study is the value of reconstruction loss. To estimate the value of train reconstruction loss and assess reconstruction loss, reconstruction loss is computed using Mean Square Error (MSE) methods. Tables 4.1 and 4.2 show the calculation results with the regularization of weight of 0.00001.

Table 4.1. Train and eval reconstruction loss values in a model with 3 epochs and a regularization weight of 0.00001

Epoch	Train Reconstruction Loss	Eval Reconstruction Loss
1	3418.031988	2184.481201
2	2180.490850	1595.676147
3	1923.211903	1344.174072

The value of the reconstruction loss should ideally decrease as the epoch rises. A decent reconstruction loss value is also close to zero (0). In general, a model with three epochs has a lower train loss and eval loss value as the epoch number grows. According to trend of the value of the train rebuilding loss, the loss's worth is diminishing. The trend in the value of the eval reconstruction loss is similar, with the loss value decreasing. Furthermore, Table 4.1 shows that the eval reconstruction loss value is less than the train reconstruction loss value. This means that the model performed better on the test data.

Table 4.2. Train and eval reconstruction loss values in a model with 5 epochs and a regularization weight of 0.00001

Epoch	Train Reconstruction Loss	Eval Reconstruction Loss
1	3709.065782	1923.692627
2	1957.295876	1120.472412
3	1632.884357	982.791260
4	1183.218892	1053.554077
5	1057.201247	1369.580933

In general, a model with 5 epochs has a lower train loss and eval loss value as the epoch number grows. According to the trend of the value of the train reconstruction loss, the loss's worth is diminishing. However, the value of the eval reconstruction loss declined until epoch 3 and then increased. This demonstrates that there are signs of overfitting in epochs 4 and 5. Nevertheless, a more thorough assessment of the models and data used is required to confirm that the data is actually overfitting.

The reconstruction loss value derived from the loss values computation throughout the training and testing procedure with a regularization weight of 0.001 is shown below. The loss values results are shown in Tables 4.3 and 4.4 below.

Table 4.3. Train and eval reconstruction loss values in a model with 3 epochs and a regularization weight of 0.001

Epoch	Train Reconstruction Loss	Eval Reconstruction Loss
1	4002.747070	5528.802246
2	2555.539062	1777.246582
3	2211.471924	1751.809570

In general, a model with three epochs has a lower train loss and eval loss value as the epoch number increases. According to the trend of the value of the train reconstruction loss, the loss's worth is diminishing. The trend in the value of the eval reconstruction loss is similar, with the loss value decreasing. Furthermore, Table 4.3 shows that the value of the eval reconstruction loss is greater than the value of the train reconstruction loss in the first epoch. This demonstrates overfitting in the first epoch.

Table 4.4. Train and eval reconstruction loss values in a model with 5 epochs and a regularization weight of 0.001

Epoch	Train Reconstruction Loss	Eval Reconstruction Loss
1	4122.240723	5819.709961
2	2955.869141	2628.541016
3	2000.375488	1719.439087
4	1738.730957	1258.112061
5	1695.363647	1176.779785

In general, a model with 5 epochs has a lower train loss and eval loss value as the epoch number grows. According to the trend of the value of the train reconstruction loss, the loss's worth is diminishing. The trend in the value of the eval reconstruction loss is similar, with the loss value decreasing. Furthermore, Table 4.4 shows that the value of the eval reconstruction loss is greater than the value of the train reconstruction loss in the first epoch. When the results in Table 4.4 are compared to the results in Table 4.2, there is no sign of overfitting in the 4th and 5th epochs.

The error value is another metric that may be used to determine the amount of accuracy of a

Artikel diterima 31 Juli 2023, Revisi 6 Agustus 2023 *Online* 31 Agustus 2023
http://dx.doi.org/10.12962/j25023659.v9i2.18089

model that has been created. In deep learning models, the Root Mean Square Error (RMSE) is a regularly used error value calculation. In simplicity, the RMSE value is calculated by taking the square root of the squared difference between the expected and actual values of the test data. The RMSE value is produced using these calculations, as demonstrated in Table 4.5 below:

Table 4.5. RMSE calculation results of the model

Model	Regularization Weight	Epoch	RMSE
1	0.00001	3	0.4530
2		5	0.4566
3	0.001	3	0.4449
4		5	0.4386

Depending on the RMSE calculation findings in Table 4.5 above, the model with a regularization weight of 0.00001, thus the model with epoch 3, has the minimum error value. This is consistent with the loss value results, which show that the model with epoch 5 has an indication of overfitting on the fourth and fifth repetitions. Consequently, the error value in the model with epoch 5 is larger than the error value in the model with epoch 3. Meanwhile, the model with a regularization weight of 0.001, thus the model with epoch 5 has the minimum error value. This is consistent with the loss value results, where the loss value trend in the model with epoch 5 is smaller than the loss value trend in the model with epoch 3. If the error value from all models is compared, the model with a regularization weight of 0.001 and epoch 5 has the lowest error value. When compared to the other three models, model 4 is the most similar to the numerical simulation findings.

CONCLUSIONS AND RECOMMENDATIONS

As a result of the research, it is possible to infer that the model is adequate for predicting the distribution of CO_2 gas saturation. The resulting reconstruction loss value, which reduces as the number of epochs grows, demonstrates this. Furthermore, the obtained RMSE values ranged from 0.43 to 0.45. However, additional refinement of this model is required to achieve more ideal results.

Future work should consider using field data as input data to better represent the actual situation. The authors also expect future work to consider the use of more than 700 training data, 300 test data, and 5 epochs, as well as a regularization weight of 0.001.

ACKNOWLEDGMENTS

The author would like to thank the ITS for the internal institution funding to carry out this research, as well as all parties who cannot be mentioned one by one.

REFERENCES

Addo, P. M., Guegan, D., and Hassani, B. (2018), "Credit risk analysis using machine and deep learning models", *Risks*, Vol.6, hal. 1-20.

Diba, A., Fayyaz, M., Sharma, V., Karami, A.H., Arzani, M.M., Yousefzadeh, R., and Gool, L.V. (2017), "Temporal 3D ConvNets: New architecture and transfer learning for video classification", arXiv preprint arXiv:171108200. https://doi.org.10.48550/arXiv.1711.08200.

Doughty, C. (2010), "Investigation of CO2 plume behaviour for a large-scale pilot test of geologic carbon storage in a saline formation", *Transport in Porous Media*, Vol.81, No.1, hal. 49-76.

Khebzegga, O., Iranshahr, A., and Tchelepi, H. (2020), "Continuous relative permeability model for compositional simulation", *Transport in Porous Media*, Vol.134, No.1, hal. 139-172.

Pachauri, R.K., Allen, M.R., Barros, V.R., Broome, J., Cramer, W., Christ, R., Church, J.A., Clarke, L., Dahe, Q., Dasgupta, P., Dubash, N.K., Edenhofer, O., Elgizouli, I., Field, C.B., Forster, P., Friedlingstein, P., Fuglestvedt, J., Gomez-Echeverri, L., Hallegatte, S., Hegeri, G., Howden, M., Jiang, K., Jimenez Cisneroz, B., Kattsov, V., Lee, H., Mach, K.J., Marotzke, J., Mastrandrea, M.D., Meyer, L., Minx, J., Mulugetta, Y., O'Brien, K., Oppenheimer, M., Pereira, J.J., Pichs-Madruga, R., Plattner, G.K., Portner, Hans-Otto, Power, S.B., Preston, B., Ravindranath, N.H., Reisinger, A., Riahi, K., Rusticucci, M., Scholes, R., Seyboth, K., Sokona, Y., Stavins, R., Stocker, T.F., Tschakert, P., Van Vuuren, D., and Van Ypserle, J.P. (2014), "*Climate change 2014: Synthesis report", Contribution of Working Groups I, II and III to the Fifth Assessment Report of* *the Intergovernmental Panel on Climate Change*, IPCC, Geneva, Switzerland.

Pini, R., Krevor, S.C., and Benson, S.M. (2012), "Capillary pressure and heterogeneity for the CO2/water system in sandstone rocks at reservoir conditions", *Advances in Water Resources*, Vol.38, hal. 48-59.

Pruess, K. (2005), "ECO2N: A THOUGH2 fluid property module for mixtures of water, NaCl, and CO2", Earth Sciences Division, Lawrence Berkeley National Laboratory, University of California, Berkeley, CA.

Wen, G., and Benson, S.M. (2019), "CO2 plume migration and dissolution in layered reservoirs", *International Journal of Greenhouse Gas Control*, Vol.87, hal. 66-79.

Wen, G., Hay, C., and Benson, S.M. (2021), "CCSNet: A deep learning modeling suite for CO2 storage", *Advances in Water Resources*, Vol.155, hal. 1-16.

Wen, G., Tang, M., and Benson, S.M. (2021), "Towards a predictor for CO2 plume migration using deep neural networks", *International Journal of Greenhouse Gas Control*, Vol.105, hal. 1-18.

Yang, Z., Syabani, M.I., Solano, N., Ganizadeh, A., and Clarkson, C.R. (2023), "Experimental determination of gas-water relative permeability for ultra-low-permeability reservoirs using crushed-rock samples: Implications for drill cuttings characterization", *Fuel*, Vol.347, hal. 1-19.

Artikel diterima 31 Juli 2023, Revisi 6 Agustus 2023 *Online* 31 Agustus 2023
http://dx.doi.org/10.12962/j25023659.v9i2.18089