



Long Short-term Memory (LSTM) Networks for Forecasting Reservoir Performances in Carbon Capture, Utilisation, and Storage (CCUS) Operations

Utomo Pratama Iskandar and Masanori Kurihara

¹Waseda University, Graduate School of Creative Science and Engineering,
Department of Earth Sciences, Resources and Environmental Engineering,
3-4-1 Okubo, Shinjuku, Tokyo 169-8555, Japan

Corresponding author: utomo@ruri.waseda.jp
(Received June 02, 2022 ; Accepted June 13, 2022)

ABSTRACT - Forecasting reservoir performances during the carbon capture, utilization, and storage (CCUS) operations is essential to monitor the amount of incremental oil recovered and CO₂ trapped. This paper proposes predictive data-driven models for forecasting oil, CO₂, and water production on the existing wells and future infill well utilizing long short-term memory (LSTM) networks, a deep learning variant for time series modeling. Two models are developed based on the number of phases referred to: 3-phases (3P) and 1-phase (1P), one interest phase at a time. The models are trained on the dataset from multiple wells to account for the effect of interference of neighboring wells based on the inverse distance to the target well. The performance of the models is evaluated using walk-forward validation and compared based on quality metrics and length and consistency of the forecasting horizon. The results suggest that the 1P models demonstrate strong generalizability and robustness in capturing multivariate dependencies in the various datasets across eight wells with a long and consistent forecasting horizon. The 3P models have a shorter and comparable forecasting horizon. The 1P models show promising performances in forecasting the fluid production of future infill well when developed from the existing well with similar features to the infill well. The proposed approach offers an alternative to the physics-driven model in reservoir modeling and management and can be used in situations when conventional modeling is prohibitively expensive, slow, and labor-intensive.

Keywords: carbon capture, utilization, and storage (CCUS), deep learning, time series forecasting, long short-term memory (LSTM) networks.

© SCOG - 2022

How to cite this article:

Utomo Pratama Iskandar and Masanori Kurihara 2022, Long Short-term Memory (LSTM) Networks for Forecasting Reservoir Performances in Carbon Capture, Utilisation, and Storage (CCUS) Operations, Scientific Contributions Oil and Gas, 45 (1) pp., 35-50. DOI.org/10.29017/SCOG.45.1.943

INTRODUCTION

Carbon capture, utilization, and storage (CCUS) in the upstream oil and gas industry involve the capture of carbon dioxide (CO₂) from industrial processes, the transport of this CO₂ via pipeline, and the injection into a depleted oil reservoir to recover the remaining oil. The utilization of CO₂ for enhanced oil recovery (EOR) sustains oil production and meets the climate goals. It is a viable emissions reduction technology that can be applied to Indonesia owing to the availability of reservoirs and CO₂ captured

ready from the gas processing plant (Iskandar & Syahrial, 2009).

Forecasting reservoir performance during the CCUS operations is essential to monitor the amount of incremental oil recovery and CO₂ trapped. Additional oil recovery can amount from 5% to 20% of the original oil in place (OOIP) depending on the characteristics of the hydrocarbon and the reservoir conformance (Green & Willhite, 2003). Meanwhile, approximately 40% of the injected CO₂ remains trapped in reservoirs during the CO₂ injections

(Faltinson & Gunter, 2011). Forecasting entails estimating oil, water, and gas production over the life of wells. It enables decision-making for economic evaluation and field development planning (Tadjeer et al., 2021).

Reservoir simulation model is the most popular tool for forecasting petroleum field production and assisting in decision making. It is one of the most effective physics-driven modeling methods for analyzing and predicting reservoir behavior. The drawbacks of reservoir simulation are: (i) it requires a large amount of reservoir data and is labor and time-intensive; (ii) it necessitates extensive effort to comprehend and analyze the field; and (iii) uncertainties in reservoir input data that affect reservoir performance prediction (Negash & Yaw, 2020). As a result, reservoir simulations are not always practical for field development plan and reservoir management studies, especially in a constantly and rapidly evolving environments, such as the volatility of oil prices, disruptive technologies, and pandemics

With the industrial revolution 4.0, the oil and gas industry is embracing artificial intelligence (AI), predictive analytics, and automation to facilitate effective decision-making, cost-cutting on inefficient operations, and better understanding well and reservoir performances (Khan & Louis, 2021). Many researchers are using deep learning algorithms for production forecasting. Al-Shabandar et al. (2021) proposed the novel architecture of a deep gate recurrent neural network, which is an advancement to the standard recurrent neural network for the prediction of oil production. The architecture can run for a long time without using the memory unit and can train well in less time. Similarly, Song et al. (2020) presented long short-term memory (LSTM) for oil forecasting and a particle optimization algorithm to optimize the architecture of LSTM. de Oliveira Werneck et al. (2022) developed a novel data-driven system N-th Day for predicting several outputs by utilizing machine learning techniques. Additionally, the authors evaluated four deep learning architectures for time-series data prediction, including LSTM and gated recurrent unit (GRU) layers. Wei et al. (2021)) has utilized three algorithms i.e., recurrent neural network (RNN), LSTM, and GRU to forecast the pore-water pressure. The multi-layer perceptron (MLP) is used for comparison to the RNN, LSTM, and GRU.

This study proposes a predictive data-driven model for forecasting oil, CO₂, and water produc-

tion on the existing wells and future infill well using LSTM networks. The significant contributions of this work are: i) the proposed approach has never been implemented in the CCUS operations; ii) it considers well-to-well interactions and the number of phases; iii) the data required to develop the deep learning model is relatively limited; iv) and it uses the knowledge of existing wells to forecast the performance of future infill well. Two models are developed considering the number of phases, 3-phase (3P) and 1-phase (1P) models. They are trained on the dataset from multiple wells to account for the effect of interference of neighboring wells based on the inverse distance to the target well. Performance of the models is evaluated using walk-forward validation and compared based on quality metrics and length and consistency of the forecasting horizon. While not intended to substitute for conventional reservoir simulation, the proposed approach to reservoir modeling and management can be used when conventional modeling is slow, cost, and workforce prohibitive.

Data-driven Models

A. Recurrent Neural Network (RNN)

Recurrent neural network (RNN) is a type of artificial neural networks (ANNs) that has temporal loop beside input, hidden, and output units. A directional loop may aid in remembering when to make a choice, what the current node's inputs are, and what it has learnt from prior inputs. RNN is able to perform effectively with time series since its internal memory can recall the prior input received (Tariq et al., 2021). This may assist the RNN generate accurate predictions. Fig. 1 displays the RNN framework for modeling time-series observations.

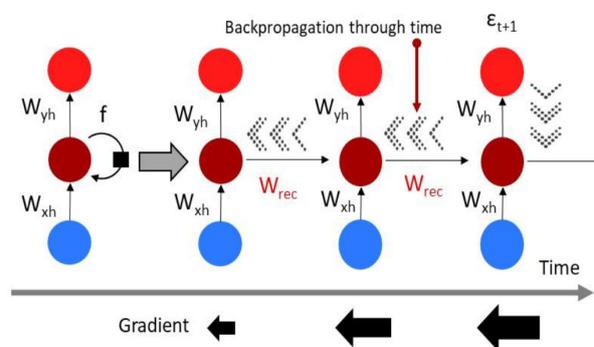


Figure 1.
Unfolded RNN structure at each time step

The characteristics of RNN are: (i) feed the output of activation function back to the same neuron; (ii) have short term memory because recent behavior has more influence on the current behavior; and (iii) great for predicting something in short contexts (Yu et al., 2019). However, several critical problems of RNN include: (i) Has a difficulty in learning long range dependencies; (ii) Severely difficult to train as the number of parameters become extremely large; and (iii) Vanishing gradient or exploding gradient problems.

Particularly the latter occurs when network is too deep. To illustrate, for the error to backpropagate the chain rule must be applied (Fig. 1). The multiplication of the same exact weight multiple times occurs many times as needed to go through the temporal loop. If at 100 epochs the weight cannot be finished updated, the overall network will be not properly trained. Furthermore, if any one of the gradients approached to zero, all the gradients would rush to zero exponentially due to the multiplication.

B. Long Short-term Memory (LSTM) Networks

The long short-term memory (LSTM) network is a recurrent neural network (RNN)-based machine learning technique (Hochreiter & Schmidhuber, 1997). LSTM is the most widely used for time series forecasting. It may retain knowledge learned over a short time and utilize it for long-term training. In capturing long dependencies, LSTM outperforms RNNs because RNNs suffer from the vanishing gradient issue, which consists of the gradient diminishing as the number of layers increases (Dama & Sinoquet, 2021). The difference between LSTM and RNN is the transmission of an internal unit state together with the hidden state.

LSTM contains units called memory blocks in the hidden layer. These blocks receive the input sequence and then determine if it is dynamic using a gate activation unit. This activity modifies the state and adds information that conditionally flows through the block (Tariq et al., 2021). Gates in the memory blocks are superior to traditional neurons since it enables them to remember current streams.

During the training phase, the weight of the gates may be improved. It makes the LSTM unit flexible because the gating function regulates the input, retains the content of the internal state variables, and manages the output, as shown in Fig. 2 (Abdel-Nasser & Mahmoud, 2019).

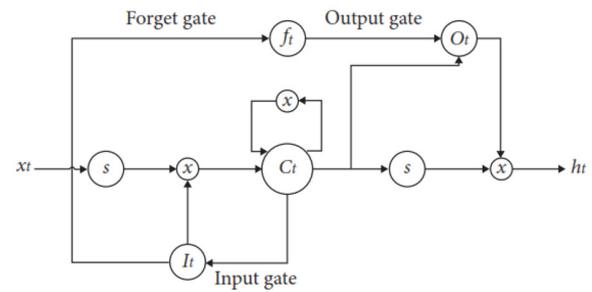


Figure 2.
LSTM unit and its components

In brief, these gates fall into three categories: (i) an input gate determines conditionally which input values contribute to updating the memory state; (ii) a forget gate decides what information will be removed from the module; (iii) an output gate decides the output based on module input and module memory state (Dama & Sinoquet, 2021). A gate may or may not become active based on the sigmoid activation function.

Similar to RNN, LSTM hyperparameters include the number of hidden layers, the number of units in each layer, network weight initialization, activation functions, learning rate, momentum values, the number of epochs, batch size (minibatch size), decay rate, optimization algorithms, sequence length for LSTM, gradient clipping, gradient normalization, and dropout (Sezer, Gudelek, & Ozbayoglu, 2020). The optimal choice of these hyper-parameters is essential, as these values greatly influence the prediction results obtained by the network.

The feedforward calculation used the hyperparameters of LSTM is shown in Eqs. 1 – 5 (Sezer et al., 2020):

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \quad (1)$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \quad (3)$$

$$c_t = f_t * c_t + i_t * \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \quad (4)$$

$$h_t = o_t * \sigma_h(c_t) \quad (5)$$

Backpropagation through time is used to train LSTM networks, which helps avoid the vanishing gradient problem.

METHODS

A. Reservoir Model

To construct the deep learning model, a PUNQ-S3 reservoir model was used to generate a dataset containing the reservoir properties as features and fluid production as targets. The compositional simulator of Computer Modelling Group Ltd. (CMG)-GEM was used for simulating the CCUS process.

The synthetic PUNQ-S3 model depicts a realistic geological model for flow simulations and has been used in several reservoir simulation investigations (Kovscek & Cakici, 2005; Lyons & Nasrabadi, 2013). It comprises $19 \times 28 \times 5$ grid cells, 1,761 of which are active. The size of each grid block is 180×180 meters. Table 1 provides a summary of the PUNQ-S3 reservoir model properties.

Table 1.
Reservoir properties of the PUNQ-S3 model

Parameter	Value	Unit
Pressure at the reference depth	28,300	kPa
Bottomhole flowing pressure	19,810	kPa
Saturation pressure	18,858	kPa
Pressure Fracture	42,450	kPa
Reservoir temperature at the reference depth	110	°C
Free water level	2,395	m
Free oil level	2,355	m
Initial water saturation	0.2	fraction
Initial oil saturation (oil zone)	0.8	fraction
Initial oil saturation (gas cap)	0.3	fraction
Initial gas saturation (gas cap)	0.5	fraction
Reference depth	2,355	m

Floris et al. (2001) described the PUNQ-S3 model as a five-layer model (Fig. 3). The PUNQ-S3 model was developed from a reservoir engineering study on a real field-performed Elf Exploration Production. It was qualified as a small-size industrial reservoir engineering model. The top depth of the PUNQ-S3 reservoir is 2,430 m. It has a dip angle of about 1.5 degrees and is bounded by a fault to the east and south. A relatively strong aquifer on the north and west provides pressure support. There is also a small gas

cap in the PUNQ-S3 reservoir model in layer 1. Each layer of PUNQ-S3 has different thicknesses and facies comprising channel fill, lagoonal shale, and mouthbar.

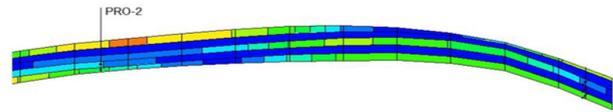


Figure 3.
Cross-sectional view of the PUNQ-S3 reservoir model

In this study, well locations have been re-configured as shown in Fig. 4. The permeability distributions in each layer guided the placement of each producing wells (PRO-1 through PRO-8). As a result, each well has several completions in the productive zone to contribute to oil production. In addition, two CO₂ injection wells (INJ-1 and INJ-2) were positioned in the northern and southern portions of the reservoir to sweep the remaining residual oil.

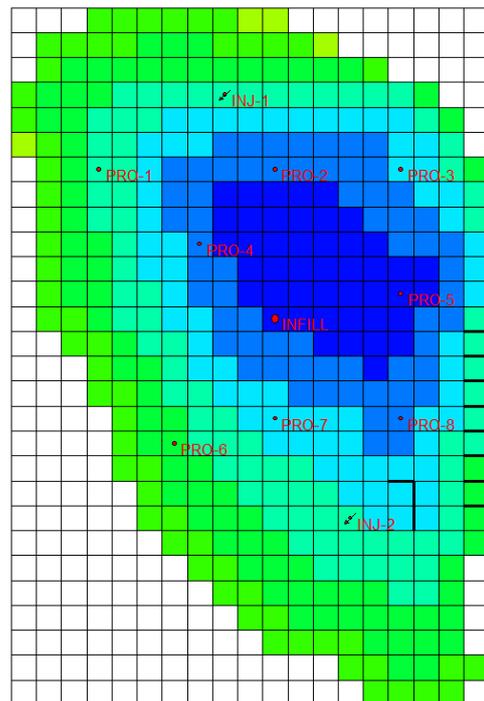


Figure 4.
Well configurations on the PUNQ-S3 reservoir model to generate dataset for deep learning model

To produce a reliable CO₂ miscible injection process, this study utilized the reservoir fluid data from a crude oil data bank containing more than 5000 PVT and gas injection data curated by Jaubert et al, (2002). Since the fluid data contains many hydrocar-

bon components that are computationally inefficient for compositional reservoir simulation, they were lumped into several pseudocomponents using the method suggested by Jessen & Stenby (2007). This method can capture significant information regarding the mixture compositions that will form during the gas injection process.

The minimum miscibility pressure (MMP) was computed using CMG-Winprop PVT simulator by employing cell-to-cell simulation. The procedure involves mixing a solvent with a primary gas. Then, a solvent is added to the oil such that the solvent-to-oil molar ratio increases by a specified value for each mixture. Flash calculations are performed for a maximum of 100 mixtures of solvent and oil. If no two-phase region is detected, the process is judged to be first-contact miscible, and the calculations stop. If a two-phase region is encountered, the calculation procedure proceeds by removing all liquid in the previous step. The remaining gas is combined with the original oil in the gas-oil ratio to form a tie line in the ternary diagram. A flash calculation is performed, and the liquid is removed. The procedure is repeated. This simulates a multi-contact vaporizing or extraction process. As a result, the MMP is achieved at 26,750 kPa by backward contacts–condensing gas drive.

The field development plan is illustrated in Fig. 5. First, the reservoir underwent natural depletion for five years, followed by continuous miscible CO₂ injection for ten years with the injection pressure and rate of 32,545 kPa and 7,000 m³/day, respectively. Finally, the infill well was drilled in the year-15.



Figure 5. Timeline of the field development plan of PUNQ-S3 reservoir to generate dataset for deep learning model

B. Dataset

A dataset encompassing production data from eight producing wells (PRO-1 to PRO-8) was used to build the deep learning models. A decade of A decade of production history from January 2004 to December 2013 was recorded with 3,653 daily

time series observations. These time-series data are oil production (m³/day), gas production (CO₂-kg/day), and water production (m³/day) (Fig. 6). The gas production is expressed in mass rate to provide a simple estimate the amount of CO₂ sequestered during the injection process with the injection rate is 3,308 tonnes/day (as comparison Weyburn field ~5,000 tonnes/day) where the composition of CO₂ in the reservoir fluid is miniscule, around 0.7% mole.

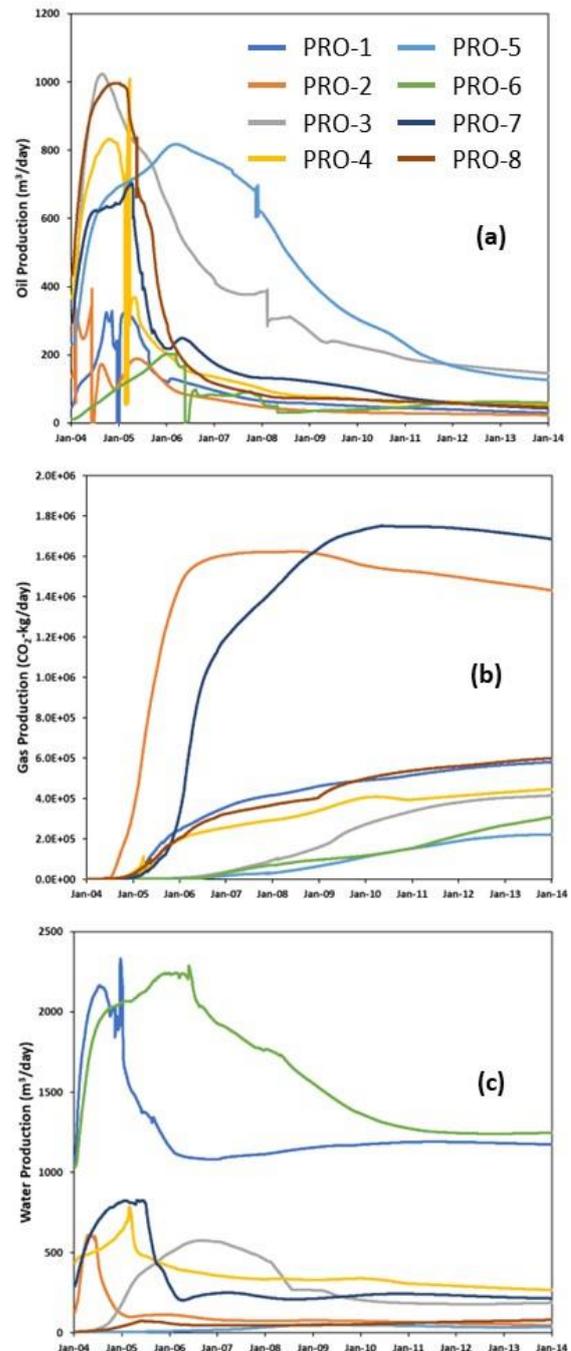


Figure 6. Time series data used to train the model: (a) oil; (b) gas; and (c) water production

C. Features and Targets

Features used for training the deep learning model comprise static parameters: porosity, permeability (mD), formation thickness (m), well location x-axis (i), and well location y-axis (j); and dynamic parameters: bottom hole flowing pressure (BHFP-kPa). These features (Table 2) were selected as they influence the fluid flow most. Besides that, these parameters were selected due to the following reasons: (i) they are readily available from field measurements and do not require processing and acquisition from the lab; (ii) they represent the capability of the well to produce a fluid; (iii) each well has its unique values which help to distinguish the characteristics between wells: as a result, the data-driven model can learn and correlate the selected parameters to production characteristics of each well. Because certain wells were completed in several layers, features such as porosity and permeability could not be simply averaged without considering the thickness of each layer. Consequently, the features such as porosity and permeability were engineered to storage capacity and flow capacity to improve the quantitative correlation between reservoir parameters on each layer and production data. Storage capacity indicates the amount of hydrocarbons can be stored in the rock, while flow capacity represents the rate at which hydrocarbons may be produced (Fanchi & Christiansen, 2016). Meanwhile, the targets of the models are the time series data of oil production, water, and gas production.

Two approaches were experimented with to teach the deep learning model the concepts of fluid displacement, pressure interference between wells, the impact of the operational constraints, and reservoir characteristics on production. The first approach simultaneously considers three phases (oil, water, and gas) on each well, whereas the second method focuses only on one interest phase at a time. Furthermore, since each well affects the neighboring wells at a different rate, the inverse distance method (Eqs. (6) – (8)) was utilized to account for this impact (Abedini & Nasser, 2008).

$$D_{A-B} = \sqrt{(i_A - i_B)^2 + (j_A - j_B)^2} \tag{6}$$

$$W_{A-B} = \frac{1}{(D_{A-B})^f} \tag{7}$$

$$WF_{A(o,w,g)} = W_{A-B} \times F_{A(o,w,g)} \tag{8}$$

The notions above were formulated by combining the data from surrounding wells with the target well. For a number of old observations 3 (N_old_obs = 3), these formulations are shown in Fig. 7 and Fig. 8 for 1-phase (1P) and 3-phase (3P) models, respectively.

In Figs. 7 and 8, Gradients (G) are the difference between observations and suggest a more favorable curve. The input data were prepared as batches in a high-dimensional tensor.

Table 2.
Features for building a deep learning model

Well	Avg. Porosity (fraction)	Avg. Permeability (mD)	Storage Capacity (m)	Flow Capacity (mD-m)	Avg. Fm Thickness (m)	BHP (kPa)	i	j
PRO-1	0.43	988.0	1.462	3,359.2	3.40	20,500	4	22
PRO-2	0.43	804.0	1.634	3,055.2	3.80	19,100	11	22
PRO-3	0.43	690.3	1.658	2,551.9	3.90	20,000	16	22
PRO-4	0.42	963.0	1.932	4,429.8	4.60	18,700	8	19
PRO-5	0.32	370.0	1.632	1,887.0	5.10	19,800	16	17
PRO-6	0.40	897.7	1.956	4,375.6	4.93	20,800	7	11
PRO-7	0.34	536.0	1.298	1,961.8	3.97	20,100	11	12
PRO-8	0.24	370.0	0.784	1,231.7	3.65	19,500	16	12
INFILL	0.41	906.0	2.132	4711.2	5.2	20,000	11	16

$$\begin{bmatrix} (O_{T1}, G_{T1})_{w1} & \dots & \dots & (O_{T1}, G_{T1})_{w8} & \dots & \dots \\ (O_{T2}, G_{T2})_{w1} & \dots & \dots & (O_{T2}, G_{T2})_{w8} & \dots & \dots \\ (O_{T3}, G_{T3})_{w1} & (Cn, Cr)_{w1} & \dots & (O_{T3}, G_{T3})_{w8} & (Cn, Cr)_{w8} & \dots \end{bmatrix} \rightarrow \begin{bmatrix} \dots \\ \dots \\ (O_{T4})_{w1} \end{bmatrix}$$

Figure 7.
The input and output of 1-phase model for N_old_obs = 3

$$\begin{bmatrix} (O_{T1}, G_{T1})_{w1} & (g_{T1}, G_{T1})_{w1} & (w_{T1}, G_{T1})_{w1} & \dots & \dots & (O_{T1}, G_{T1})_{w8} & (g_{T1}, G_{T1})_{w8} & (w_{T1}, G_{T1})_{w8} & \dots & \dots \\ (O_{T2}, G_{T2})_{w1} & (g_{T2}, G_{T2})_{w1} & (w_{T2}, G_{T2})_{w1} & \dots & \dots & (O_{T2}, G_{T2})_{w8} & (g_{T2}, G_{T2})_{w8} & (w_{T2}, G_{T2})_{w8} & \dots & \dots \\ (O_{T3}, G_{T3})_{w1} & (g_{T3}, G_{T3})_{w1} & (w_{T3}, G_{T3})_{w1} & (Cn, Cr)_{w1} & \dots & (O_{T3}, G_{T3})_{w8} & (g_{T3}, G_{T3})_{w8} & (w_{T3}, G_{T3})_{w8} & (Cn, Cr)_{w8} & \dots \end{bmatrix} \rightarrow \begin{bmatrix} \dots \\ \dots \\ (O_{T4})_{w1} \end{bmatrix} \begin{bmatrix} \dots \\ \dots \\ (g_{T4})_{w1} \end{bmatrix} \begin{bmatrix} \dots \\ \dots \\ (w_{T4})_{w1} \end{bmatrix}$$

Figure 8.
The input and output of 3-phase model for N_old_obs = 3

D. Data Pre-processing

The normalization technique was utilized as data pre-processing to scale the six features. Data pre-processing is common in developing a deep learning model because features have different units and magnitudes. Such feature scaling helps to accelerate the model learning process to converge faster. The min-max normalization is a general and effective procedure in the scientific literature (Wei et al., 2021). This process is represented as follows (Eq. (9)):

$$x_{new} = \frac{x_{old} - x_{min}}{x_{max} - x_{min}} \quad (9)$$

E. Model Development

The model was developed utilizing the Keras framework with TensorFlow as the backend. The model development process is depicted in (Fig. 9). Initially, the dataset was split into several segments by honoring their temporal order. These segments consist of: the training set (40%) for the first four years, the validation set (20%) for the following two years, and test sets (40%) for the rest four years.

The model development (Fig. 9) began with creating a baseline model. The objective is to obtain a decent model as fast as possible to make baseline predictions. This default starting point may not produce the best possible model as the model structure and topology were configured based on the rule of thumb in the machine learning application. The number of hidden layers and neurons is relatively small, and the hyperparameters were not optimized.

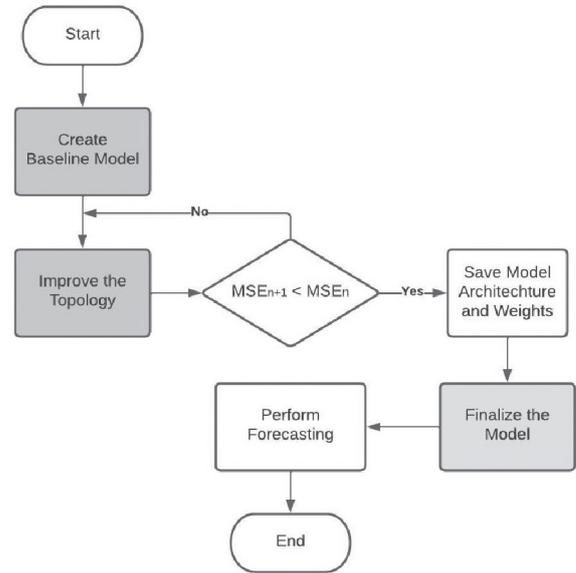


Figure 9.
Model development flowchart

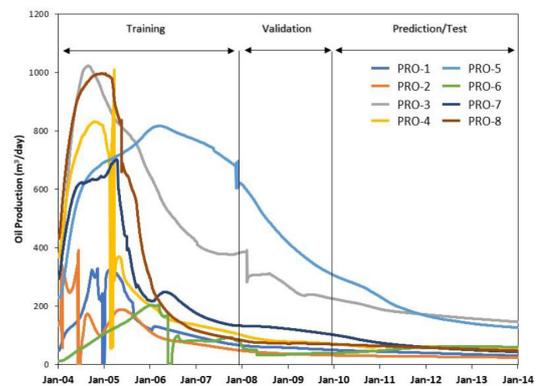


Figure 10.
Data split for training, validating and testing the deep learning LSTM model

The model was trained by applying stochastic gradient descent to minimize the loss function, the difference between the predicted value, and actual observation. During this step, 50 epochs were used to train the model. Mean squared error (MSE) (Eq. 10) was selected as the loss function:

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \tag{10}$$

Walk-forward validation (WFOV) method was adopted to evaluate the model on the validation set and perform prediction on the test set. However, during the process, the model was not updated. It is the standard method for evaluating time series models using past data and honoring the historical sequence. The procedure of WFOV is described as follows:

1. Step 1: Starting at the beginning of the validation set, the last set of observations in the training set is used as input of the model to predict the next set of data (the first set of true values in the validation set).
2. Step 2: The model makes a prediction for the next time step.
3. Step 3: Get true observation and add to history for predicting the next time.
4. Step 4: The prediction is stored and evaluated against the true observation.
5. Step 5: Go to step 1.

The next step involves improving the topology of the baseline model. The most significant leverage is by examining deeper and wider topology. It can be achieved by gradually increasing the number of hidden layers and the number of neurons. If there is overfitting due to this process, a dropout will be implemented. These hyperparameters are not learned and are fixed values inside the model equations. Grid search was used to evaluate the different combinations of hyperparameters.

Table 3 summarizes, based on early tests, the ranges of feasible values and pertinent hyperparameters (Torres, Hadjout, Sebaa, Martínez-Álvarez, & Troncoso, 2021).

Table 3. Hyperparameters used for grid search

Hyperparameters	Description	Range
Hidden layers	It determines the depth of the neural network.	[4, 5, 6, 7, 8]
Dropout	It eliminates certain connections between neurons in each iteration. It is used to prevent over-fitting.	[0.01, 0.001, 0.0001]
L1/L2 Regularization	It prevents over-fitting, stopping weights that are too high so that the model does not depend on a single feature.	[0.0001, 0.00001, 0.000001, 0.0000001]
Units	It determines the level of knowledge that is extracted by each layer. It is highly dependent on the size of the data used.	[128, 256, 512, 1024]

Finally, the model was ready to be deployed to forecast the time series. Two prediction methods were used: walk-forward (WF) and walk-forward over data (WFOD), as shown in Fig. 12. WF prediction uses the last number of actual observations to make the next prediction. Once the last observation is used, the model uses observations generated by the model in the previous time step. In contrast, WFOD uses the actual observations of the current period to predict subsequent periods or generates forecasts in a rolling fashion.

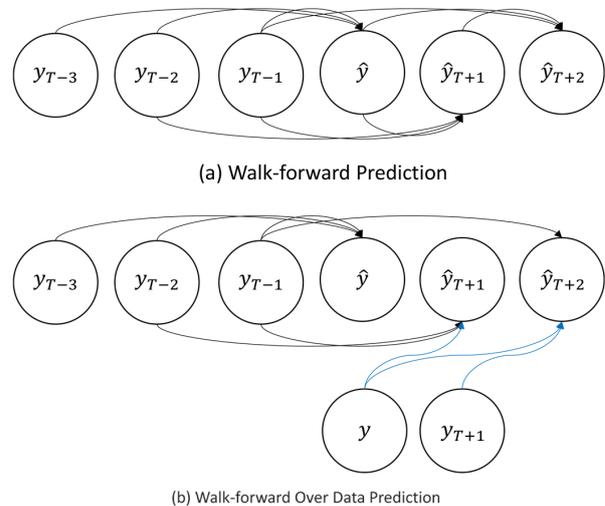


Figure 11. Prediction methods to forecast the time series

Mean Absolute Percentage Error (MAPE), Eq. 11, was used as the performance metric to evaluate the prediction results. Since the datasets comprise different time series with different units, MAPE is convenient for this case because the error values are presented in percentages. It enables a direct comparison of the accuracy of various time series datasets.

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left(\frac{|\hat{y}_i - y_i|}{y_i} \right) \times 100\% \quad (11)$$

Similar methods were used for infill well predictions. Eight existing wells were used to predict the infill well future time series (Fig. 11). Each well represents a distinct deep learning model.

The number of old observations (N_{old_obs}) determines the number of time steps used to compute the initial data. Subsequently, these data were weighted proportional to the inverse distance of existing wells to the infill well. To make prediction, the features of the infill well (e.g., well location and storage capacity) were fed to each of the existing well models. The walk-forward prediction (WF) method was used to forecast the oil, gas and water production of future infill well by utilizing N_{old_obs} time

series data. As a result, each existing well generated one forecast for each time series data. Since there are eight existing wells and three time series data, the total time series data generated are 24 time series representing 24 deep learning models. In addition, two time series data were generated for each time series using weighted and simple averaging of individual time series.

Instead of employing MAPE as described above, each prediction generated from each model was evaluated to the actual observation using average

$$\sigma_{average} = \frac{|y_i - \hat{y}_i|}{(\max(y_i) - \min(y_i))} \times 100\%, \quad (12)$$

The AND is a more reliable evaluation if the actual observation value is close to zero.

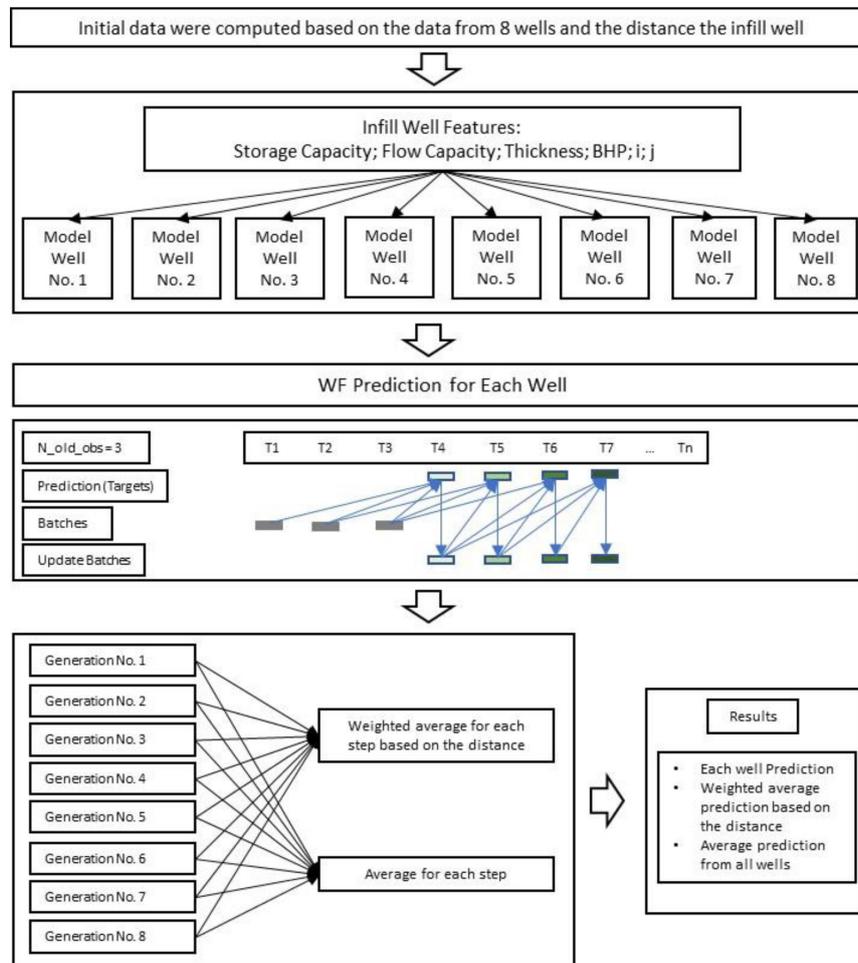


Figure 12. Prediction method for infill well

RESULTS AND DISCUSSIONS

A. Final Model Topology

The results for developing both 3-phases (3P) and 1-phase (1P) LSTM models are shown in Table 4. The number of hidden layers and units varies for the 1P models. This variation was attributed to the time series used for training each model. Each phase has a different trend; for instance, the gas production rate generally has an upward trend. Consequently, the weights and biases were adjusted to the corresponding target. Meanwhile, the 3P models require many units to handle the complexity of three distinct

time series simultaneously. As the number of hidden layers or units increases in deep learning, the neural network will provide power and flexibility, improving accuracy (Hornik et al., 1989).

During training, it was observed that overfitting occurred. Therefore, dropout was introduced to enhance the model generalization capability. This strategy was applied to all developed models, which is a common practice due to the increase in model topology. As a result, the network becomes less sensitive to the specific weights of neurons, forcing the network to distribute its learning.

Table 4.
Results of topology and hyperparameter tuning

Model	Parameter	3P Model	1P Model		
			Oil	Gas	Water
LSTM	Factor for inverse distance	20	20	20	20
	Number of previous observations	5	5	5	5
	Optimizers	Adam	Adam	Adam	Adam
	Activation function	ReLU	ReLU	ReLU	ReLU
	No. of hidden layers	6	7	8	8
	No. of units	1024	256	512	512
	Dropout	0.01	0.001	0.0001	0.001
	Kernel regularizator	L1	L1	L1	L1
	Kernel regularizator rate	0.0001	0.0001	0.0000001	0.0000001
	Activity regularizator	L2	L2	L2	L2
	Activity regularizator rate	0.00000001	0.0000001	0.0000001	0.0000001

B. Existing Well Forecasting

3P Model

The 3P model predicts the values of time-dependent targets based on the flow of oil, gas, and water from each well to the inverse distance of the target well. The predictive performance is evaluated using the test dataset of 1,460 observations. The predictions of 3P models for each time-series data are shown in Fig. 13. Owing to space constraints, only PRO-1 well is shown.

The purple line on the graph is the WFOD prediction that forecasts one time step in advance based on actual data. The model could generally predict the downward slopes and flat lines for a certain period before deviating from the actual observations. For example, the prediction of oil production shows a good agreement with the actual data for several months before it underestimates the trend. Meanwhile, the

model can forecast upward trends, and the prediction is accurate through the completion of the project for gas production. In contrast, the prediction of water production is relatively shorter compared to other time series. It may be because the water production profile comprises fluctuated portions at the tail of the time series.

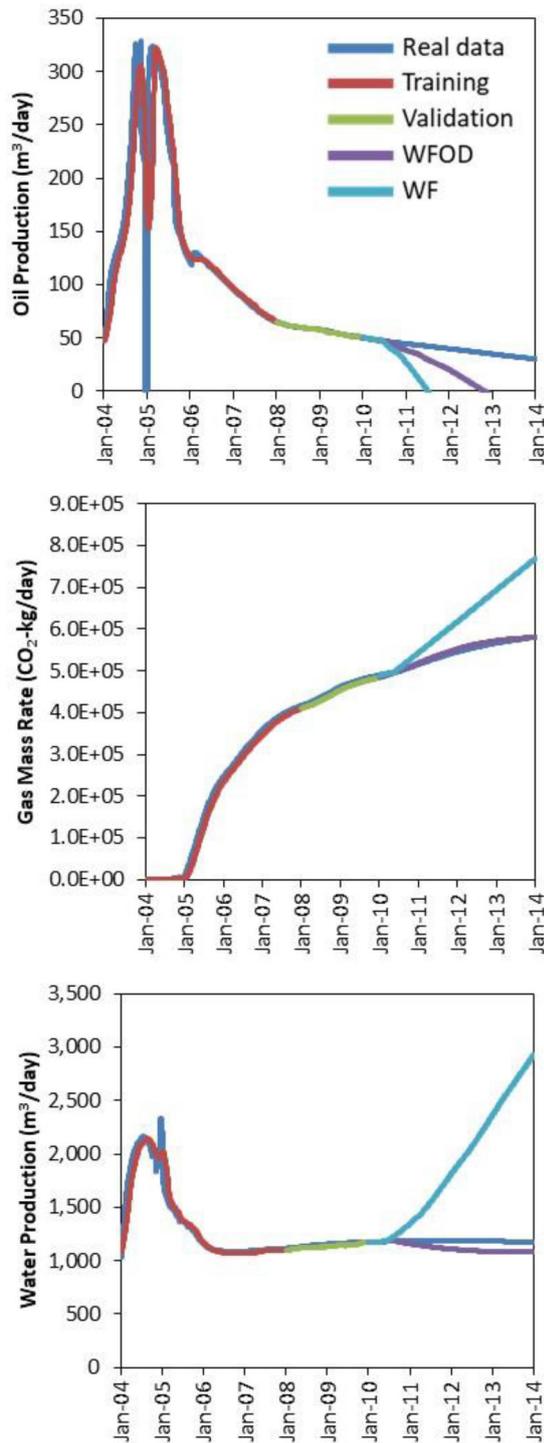


Figure 13. Prediction results of 1P models PRO-1 for each time series

The WF prediction, shown by a light blue line, demonstrates a significant premature deviation from the actual observations. The forecasting horizon is shorter than the WFOD prediction method. The prediction could not be anticipated in certain wells as it is gradually off from the historical data. In general, the model struggles to forecast downward slopes and plateau trends in oil and water production and upward trends in gas production. However, such deviation can be solved by updating the model after the inclusion of new data in order to apply the most recent trend character. Nonetheless, having WF prediction gives access to forecast several time steps in the future, thus allowing decision-makers to see the future trends and optimize their actions throughout the course.

1P Model

The forecasting results for all time series and wells for the 1P models are shown in Fig. 14. It can be observed that the WFOD predictions in most time series resulted in a longer forecasting horizon than the 3P models. Although some prediction errors exist in a few wells, the model can generally represent the multivariate relationship between reservoir parameters and production data. In particular, reproduction of the oil and water patterns showed substantial improvements.

Likewise, the WF predictions perform better in most wells and time series. The forecasting horizon is extended, and sudden jumps in the prediction are reduced significantly. However, the forecasting results tend to overestimate gas production in some wells after certain periods. Meanwhile, the prediction trends in oil and water production agree with the actual observations. Overall, the non-linear approximation holds for short-term forecasts, but the deviation is more pronounced when the forecasting horizon becomes large.

C. Comparing 3P Models vs. 1P Models

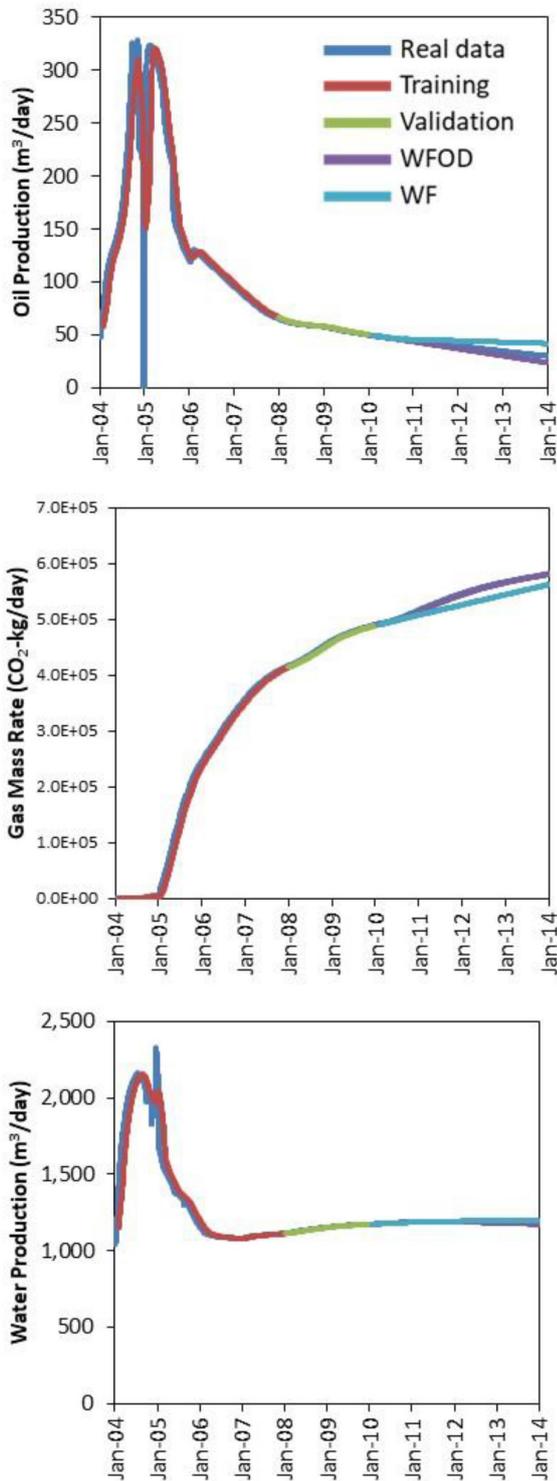


Figure 14. Prediction results of 1P models PRO-1 for each time series

The accuracy between 3P and 1P models is compared to evaluate model predictive capability. Table 5 presents the average MAPE values for the entire wells and time series. These MAPE values were computed over four years. The reduction of features significantly impacts the accuracy of the 1P model for both prediction methods. Reducing the number of features improved the predictive performance of the 1P models, as observed in most time series. As the number of features decreases, the input sequence becomes shorter. Particularly for LSTM, a short input sequence facilitates the model for capturing long-term multivariate dependencies of data (Abdullayeva & Imamverdiyev, 2019).

In addition to accuracy, the model performance was evaluated based on the forecasting horizon. The longer the forecasting horizon reproduces actual well performance, the more preferable. Table 6 presents the average number of days for each prediction method with less than 5% error. An error less than 5% denotes that the model has an accuracy of 96% in a specific forecasting horizon. For instance, the walk-forward (WF) prediction of the 1P model for oil rate has an average of 176 days. This value indicates that from day 0 of prediction up to day 176, the difference between actual observations vs. predicted observations within this horizon is maximum 4%. In other words, the model enables prediction for 176 days with an accuracy of 96%.

The feature reduction also influences predictive power on the forecasting horizon. WFOD prediction typically has a larger forecasting horizon than WF prediction since it uses the most recent time step to anticipate the next time step. Generally, the forecasting horizon of the 1P models outperforms the 3P models in most time series and both prediction methods. However, the discrepancies of both models are not statistically different. The shortest forecasting horizon on 3P and 1P models are oil and water production, around 150 days and 170 days, respectively.

Consistency in producing long-term forecasts was evaluated according to the standard deviation of the average number of days (Table 7). These figures represent the mean of eight wells for each model type and prediction method. The 1P models have more consistent in producing a long forecasting horizon across eight wells, as indicated by the lower standard deviation. This implies that the 1P models have a high degree of generalizability and are robust at capturing multivariate relationships across various datasets.

Table 5.
Average MAPE of eight wells for four years

Model	WFOD			WF		
	Oil Rate	Gas Mass Rate (CO ₂)	Water Rate	Oil Rate	Gas Mass Rate (CO ₂)	Water Rate
LSTM – 3P	41.46%	3.86%	31.25%	142.96%	15.38%	72.19%
LSTM – 1P	14.84%	5.63%	7.15%	28.87%	12.06%	10.05%

Table 6.
The average number of days with prediction error < 5% for eight wells

Model	WFOD (Days)			WF (Days)		
	Oil Rate	Gas Mass Rate (CO ₂)	Water Rate	Oil Rate	Gas Mass Rate (CO ₂)	Water Rate
LSTM – 3P	180	269	261	158	177	172
LSTM – 1P	269	284	278	176	187	183

Table 7.
The standard deviation of the average number of days with prediction error < 5% for eight wells

Model	WFOD (Days)			WF (Days)		
	Oil Rate	Gas Mass Rate (CO ₂)	Water Rate	Oil Rate	Gas Mass Rate (CO ₂)	Water Rate
LSTM – 3P	12	10	11	13	12	12
LSTM – 1P	7	9	9	13	11	15

D. Future Infill Well Forecasting

The performance of future infill well was forecasted using 1P models derived from knowledge of existing wells. The available information on the infill well includes well location and static and dynamic

features. Such information was fed to the models trained on the existing wells. Only the most accurate of five years performances of infill well for all time series are presented in Fig. 15.

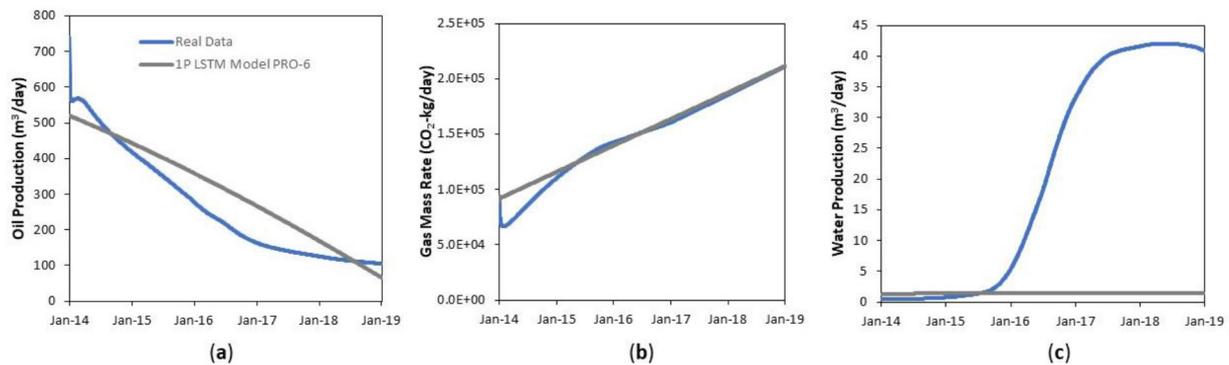


Figure 15.
Infill forecasting performance using LSTM 1P model PRO-6 well

Fig. 15 shows the prediction comparisons generated from the model and real dataset. The 1P model using PRO-6 well can mostly reproduce the trend of real datasets. In the first year, the model was prone to underestimate the oil production while overestimating the gas production. Afterward, it closely followed the upward trend

of gas production. The profile of water production could not be perfectly reproduced for the whole well's lifetime, especially the oscillating part. However, it matched the plateauing trend for the first two years.

As the models generated multiple forecasts from the existing wells, the prediction error for each

time series data was computed using average normalized difference (AND). Fig. 16 presents the range of uncertainty generated from the models for each time series. The prediction can be anticipated to underestimate 105% at the end of the project for gas production. On the other hand, in the commencement of the infill well, it can be expected that the model overestimates the water

production by 188% of the actual production. In the traditional numerical simulation, where various scenarios are simulated with significant uncertainties, the range of uncertainty is considered excellent if within $\pm 100\%$ of the actual production. Nonetheless, this range provides the degree of confidence to determine if the well location is optimal.

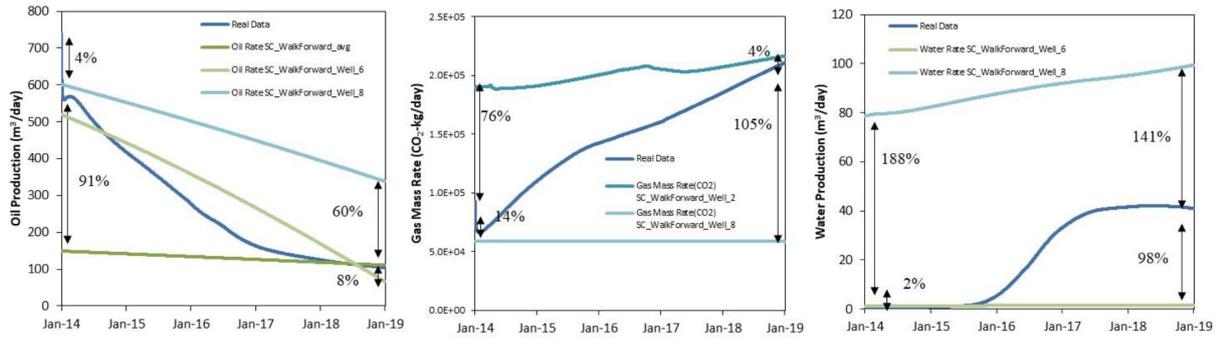


Figure 16. Range of uncertainty 1P flow models for eight wells

Furthermore, MAPE was utilized to quantify the discrepancy between model-generated predictions and actual data. Additionally, the curvature was

assessed using shape difference (Table 8). Overall, the 1P flow model PRO-6 well can satisfactorily predict the performance of the infill well.

Table 8. The 1P model PRO-6 evaluation results

Time Series	MAPE	% Shape Difference
Oil Rate	28%	28%
Gas Mass Rate (CO ₂)	5%	6%
Water Rate	98%	17%

The PRO-6 well has similar features to the infill well with an average difference of 6%, which may contribute to the satisfactory predictions (Table 9). The model trained on PRO-6 can

capture the time series trend and characterize the dependent relationship of time sequence data with the features. This finding can assist in field development plans from an economic standpoint.

Table 9. Features similarity of the existing wells compared to the infill well

Well No.	Storage Capacity (m)	Flow Capacity (mD-m)	Fm. Thickness (m)	BHP (kPa)	% Total Difference
Infill Well	2.132	4,711.2	5.2	20,000	-
PRO-1	1.462	3,359.2	3.40	19,100	24%
PRO-2	1.634	3,055.2	3.80	20,000	22%
PRO-3	1.658	2,551.9	3.90	18,700	23%
PRO-4	1.932	4,429.8	4.60	19,800	8%
PRO-5	1.632	1,887	5.10	20,800	22%
PRO-6	1.956	4,375.6	4.93	20,100	6%
PRO-7	1.298	1,961.8	3.97	19,500	30%
PRO-8	0.784	1,231.7	3.65	19,100	42%

CONCLUSIONS

Deep learning models utilizing long short-term memory (LSTM) networks were developed and investigated to fit the static and dynamic features of PUNQ-S3 reservoir models. The models were developed by considering three phases simultaneously (3P) and focusing on only one phase of interest (1P), and the interference of neighboring wells based on the inverse distance. The findings suggest that 1P models outperform the 3P models in most time series and in walk-forward over data (WFOD) and walk-forward (WF) prediction methods. The 1P models have a high degree of generalizability and are robust at capturing multivariate relationships across various datasets. Reducing the number of features improved the predictive performance of the 1P models on the accuracy and the forecasting horizon.

The performance of the 1P models in forecasting the fluid production of future infill wells is encouraging. By selecting the similar features of the existing wells, the performance of the infill well can be satisfactorily predicted.

Future works may involve observing model improvements by experimenting with a different training size. In addition, the actual data from the field should be considered to test the model's capability in handling more variable datasets.

GLOSSARY OF TERMS

Symbol	Definition	Unit
W_{xh}	Weight between input and hidden layer	-
W_{yh}	Weight between hidden and output layer	-
h	State	-
f	Recurrence formula	-
t	Time step	Day
x_t	Input vector to the LSTM	-
f_t	Forget gate activation vector	-
i_t	Input gate activation vector	-
o_t	Output gate's activation vector	-
h_t	Output vector of the LSTM unit	-
c_t	Cell state vector	-
σ_g	Sigmoid function	-
σ_c	Hyperbolic tangent function	-
σ_h	Hyperbolic tangent function	-
W	Weight matrices need to be learned	-
U	Weight matrices need to be learned	-
b	Bias vector parameters that need to be learned	-
WF	Weighted fluid	-

F_A	Fluid of well-A	$m^3/day,$ $CO_2\text{-kg/day}$
D_{A-B}	Distance from well-A to well-B	-
W_{A-B}	Weight of well-A and well-B	-
f	Factor for inverse distance	-
(i,j)	Well location	-
C_n	Constants [BHFP, thickness, permeability, porosity]	-
Cr	Coordinates	-
G	Gradients for respective phase [goil, ggas, gwater]	-
x_{new}	The normalized data points	-
x_{old}	The real values	-
x_{min}	The lower data constraints	-
x_{max}	The upper data constraints	-
n	The number of samples	-
Y_i	Target values	-
\hat{Y}_i	Predicted values	-
\hat{y}_{Ti}	Predicted value at i^{th} time step	-
y_{Ti}	Observation value at i^{th} time step	-
\hat{y}_i	Predicted values	-
y_i	Target values	-
$\sigma_{average}$	Average normalized difference	-
<i>Subscript</i>		
o	Oil	
w	Water	
g	Gas	
w_i	Well number i^{th}	
$T-i$	Time step i^{th}	

REFERENCES

- Al-Shabandar, R., Jaddoa, A., Liatsis, P., & Hussain, A. J. (2021). A deep gated recurrent neural network for petroleum production forecasting. *Machine Learning with Applications*, 3, 100013.
- Dama, F., & Sinoquet, C. (2021). Time Series Analysis and Modeling to Forecast: a Survey. *arXiv preprint arXiv:2104.00164*.
- de Oliveira Werneck, R., Prates, R., Moura, R., Gonçalves, M. M., Castro, M., Soriano-Vargas, A., Ferreira, A. (2022). Data-driven deep-learning forecasting for oil production and pressure. *Journal of Petroleum Science and Engineering*, 210, 109937.
- Faltinson, J. E., & Gunter, B. (2011). Net CO₂ stored in North American EOR projects. *Journal of Canadian Petroleum Technology*, 50(07), 55-60.
- Floris, F. J., Bush, M., Cuypers, M., Roggero, F., & Syversveen, A. R. (2001). Methods for quantifying the uncertainty of production forecasts: a comparative study. *Petroleum Geoscience*, 7(S), S87-S96.
- Green, D. W., & Willhite, G. P. (2003). *Enhanced Oil Recovery*, SPE Textbook Series Vol. 6, the Society of Petroleum Engineers Inc. USA doi, 10.

- Hochreiter, S., & Schmidhuber, J.** (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735-1780. doi:10.1162/neco.1997.9.8.1735
- Hornik, K., Stinchcombe, M., & White, H.** (1989). Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5), 359-366.
- Iskandar, U. P., & Syahrial, E.** (2009). Carbon Capture And Storage (Ccs)-Enhanced Oil Recovery (Eor): Global Potential In Indonesia. *Scientific Contributions Oil and Gas*, 32(3), 228-238.
- Khan, H., & Louis, C.** (2021). An Artificial Intelligence Neural Networks Driven Approach to Forecast Production in Unconventional Reservoirs—Comparative Analysis with Decline Curve. Paper presented at the International Petroleum Technology Conference.
- Kovscek, A., & Cakici, M.** (2005). Geologic storage of carbon dioxide and enhanced oil recovery. II. Cooptimization of storage and recovery. *Energy conversion and Management*, 46(11-12), 1941-1956.
- Lyons, J., & Nasrabadi, H.** (2013). Well placement optimization under time-dependent uncertainty using an ensemble Kalman filter and a genetic algorithm. *Journal of Petroleum Science and Engineering*, 109, 70-79.
- Negash, B. M., & Yaw, A. D.** (2020). Artificial neural network based production forecasting for a hydrocarbon reservoir under water injection. *Petroleum Exploration and Development*, 47(2), 383-392.
- Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M.** (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied soft computing*, 90, 106181.
- Song, X., Liu, Y., Xue, L., Wang, J., Zhang, J., Wang, J., Cheng, Z.** (2020). Time-series well performance prediction based on Long Short-Term Memory (LSTM) neural network model. *Journal of Petroleum Science and Engineering*, 186, 106682.
- Tadjar, A., Hong, A., & Bratvold, R. B.** (2021). Machine learning based decline curve analysis for short-term oil production forecast. *Energy Exploration & Exploitation*, 39(5), 1747-1769.
- Tariq, H., Hanif, M. K., Sarwar, M. U., Bari, S., Sarfraz, M. S., & Oskouei, R. J.** (2021). Employing Deep Learning and Time Series Analysis to Tackle the Accuracy and Robustness of the Forecasting Problem. *Security and Communication Networks*, 2021.
- Torres, J. F., Hadjout, D., Sebaa, A., Martínez-Álvarez, F., & Troncoso, A.** (2021). Deep learning for time series forecasting: a survey. *Big Data*, 9(1), 3-21.
- Wei, X., Zhang, L., Yang, H.-Q., Zhang, L., & Yao, Y.-P.** (2021). Machine learning for pore-water pressure time-series prediction: Application of recurrent neural networks. *Geoscience Frontiers*, 12(1), 453-467.