



Prediction of well performance in SACROC field using stacked Long Short-Term Memory (LSTM) network

Palash Panja ^{a,b,c,*¹}, Wei Jia ^{a,c,d}, Brian McPherson ^{a,c,d}

^a National Energy Technology Laboratory, 626 Cochran Mill Road, Pittsburgh, PA 15236, USA

^b Department of Chemical Engineering, University of Utah, 50 S, Central Campus Dr., Room 3290 MEB, Salt Lake City, UT 84112, USA

^c Energy & Geoscience Institute, University of Utah, 423 Wakara Way, Suite 300, Salt Lake City, UT 84108, USA

^d Department of Civil and Environmental Engineering, University of Utah, 110 S, Central Campus Dr., Suite 2000, Salt Lake City, UT 84112, USA

ARTICLE INFO

Keywords:

Enhanced Oil Recovery (EOR)
Long Short-Term Memory (LSTM)
SACROC unit
Water Alternating CO₂ Injection
Mean Absolute Percentage Error (MAPE)
Production Performance

ABSTRACT

Forecasting oil production can be very challenging, especially for reservoirs with sparse data or other complexities. If traditional decline curve analysis or time series models fail to capture production rate variabilities, a machine learning model for time series data may be effective. A temporal machine learning model, a long short-term memory network model (LSTM) in specific, may be trained to predict oil, gas, and water production rates. We develop an LSTM for such an application and evaluate its efficacy with a case study of the Scurry Area Canyon Reef Operators Committee (SACROC) unit, an active CO₂ enhanced oil recovery (EOR) field in western Texas, USA. The monthly averaged production rates of oil, gas and water from 23 producers are obtained from simulations of the SACROC reservoir model that also includes 22 injectors for 5-spot injection of water and CO₂ (alternating annually). The bottom hole pressure (BHP) of the producers, BHPs of surrounding injectors, and historical production rates are used as input data for the LSTM. Blind predictions of LSTM test sets show promising outcomes for data that otherwise traditional time series models are not effective. Stacked LSTM models are efficient in multi-step-ahead forecasting. Such an LSTM approach may also be effective for quantitative analysis of unconventional oil and gas reservoirs like shales or other tight formations. Critical aspects of the LSTM workflow include optimization of machine learning parameters and quantification of the relative impacts of different variables on forecasted outcomes.

1. Introduction

Proxy models, also called surrogate models or response surface models, are mathematically- or statistically-defined functions or sets of functions that decrease the computational requirements of numerical models, albeit often with some sacrifice of accuracy. However, induced uncertainty associated with proxy models decreases as the resolution of calibration data improves. Proxy models are utilized to forecast production/injection and perform sensitivity and uncertainty analyses. Proxy models are typically a supervised learning method (Mishra and Datta-Gupta, 2017). Models are first calibrated or trained using selected training data (70–80% of total data) (Panja, 2018). The training process involves optimizing the model parameters while minimizing an objective function (e.g., mean squared error). Popular proxy model methods include response surface methodology (RSM), polynomial chaos

expansion (PCE), and artificial neural network (ANN).

Although proxy models are often developed for specific (even one-time) purposes, they are often limited to static or point data predictions. Such applications limit output to only a particular time step. Therefore, forecasting of time series data sometimes requires special treatment during the development of the proxy model. De Gooijer and Hyndman (De Gooijer and Hyndman, 2006) reviewed a selected set of time series forecasting models and subdivided these into eight categories, namely.

- (i) exponential smoothing (Muth, 1960; Gardner, 1985; Snyder, 1985);
- (ii) Autoregressive Integrated Moving Average (ARIMA) (Yule, 1927; Box and Jenkins, 1970);
- (iii) seasonal models (Dagum, 1982; Huyot, 1986);
- (iv) state space and structural models and the Kalman filter (Kalman, 1960; Schweppe, 1965; Shumway and Stoffer, 1982);

* Corresponding author.

E-mail addresses: ppanja@egi.utah.edu (P. Panja), wei.jia@utah.edu (W. Jia), b.j.mcpherson@utah.edu (B. McPherson).

¹ ORCID: 0000-0002-0631-9907.

Nomenclature	
Symbol	Description (Units)
RNN	Recurrent Neural Network (-)
σ	Sigmoid function (-)
RSM	Response Surface Methodology (-)
PCE	Polynomial Chaos Expansion (-)
MAPE	Mean Absolute Percentage Error (-)
ML	Machine Learning (-)
MLP	Multi-Layer Perceptron (-)
LSTM	Long Short-Term Memory (-)
IDE	Integrated Development Environment (-)
ANN	Artificial Neural Networks (-)
SACROC	Scurry Area Canyon Reef Operators Committee (-)
BHP	Bottom Hole Pressure (kPa)
ARFIMA	Autoregressive Fractionally Integrated Moving Average (-)
ARCH	Autoregressive Conditional Heteroskedasticity (-)
GARCH	Generalized Autoregressive Conditional Heteroskedasticity (-)
ARIMA	Autoregressive Integrated Moving Average (-)
DCA	Decline Curve Analysis (-)
PLE	Power Law Exponential (-)
SE	Stretched Exponential (-)
MSE	Mean Squared Error (Square of Unit)
RMSE	Root Mean Squared Error (Unit)
NRMSE	Normalized Root Mean Squared Error (-)
LGA	Logistic Growth Analysis (-)
PM	Particulate Matter (-)
RE	Relative Error (-)
PI	Bottom Hole Pressure of Injector (kPa)
PP	Bottom Hole Pressure of Producer (kPa)
R	Fluid Production rate (Unit)
R'	Predicted Production rate (Unit)
EOR	Enhanced Oil Recovery (-)
P	Producer Well (-)
I	Injector Well (-)
ReLU	Rectified Linear Unit (-)
W	Weight (-)
$Y_{\text{model},i}$	Modeled Value (Unit)
$Y_{\text{obs},i}$	Observed Data (Unit)
Adam	Adaptive Moment Estimation (-)
b	Bias (Unit)
h	Hidden Output (Unit)
C_t	Cell State (Unit)
i_t	Input State (Unit)
O_t	Output from Output Gate (Unit)
f_t	Forget Gate (Unit)
g_t	One Output from Input Gate (Unit)
Tanh	Hyperbolic Tangent Function (-)
\otimes	Pointwise Multiplication (-)

- (v) nonlinear models (Wiener, 1958; Volterra, 1930);
- (vi) long-range dependence models, including the family of Autoregressive Fractionally Integrated Moving Average (ARFIMA) models (Ray, 1993; Ray, 1993);
- (vii) Autoregressive Conditional Heteroskedasticity/Generalized Autoregressive Conditional Heteroskedasticity (ARCH/GARCH) models (Bollerslev et al., 1994; Taylor, 1987; Engle, 1982), and.
- (viii) count data forecasting (Willemain, 1994; Croston, (1970–1977), 1972).

Although each of these models is for a particular use, ARIMA is the most popular for general time series applications. These models are trained based on historical data of the output, and other factors are seldom incorporated. Another consideration is multi-step time series forecasting, i.e., the prediction of multiple time steps into the future. Traditional time-series models cannot predict more than a single time step for ‘out-of-sample’ data. Specifically, instead of using a test data set (if available), predicted values at previous steps are incorporated into the model to predict the value at the current time step in the future. Such an approach has more practical applications because the actual outcome may not be known during predictions.

Prediction of production rates of oil, gas, and water from a hydrocarbon reservoir is a time series forecasting problem. A decline curve model developed by Arps (Arps, 1945) is widely used to predict production rates from conventional reservoirs, such as limestone or sandstone reservoirs with high porosity and high permeability. The decline curve analysis (DCA) method uses a mathematical equation developed by regression of historical data (mainly flow production rate versus time). The fitted curve is then used to predict future production rates and cumulative production. For its inherent assumption of boundary-dominated flow, this empirical approach overestimates hydrocarbon recovery from unconventional (low permeability) reservoirs. Hence, recently-developed alternative techniques include the Power Law Exponential (PLE) (Ilk and Exponential vs. Hyperbolic Decline in Tight Gas Sands: Understanding the Origin and Implications for Reserve Estimates Using Arps’ Decline Curves, in SPE Annual Technical Conference and Exhibition, , 2008), Stretched-Exponential model (SE) (Valko,

2009), Logistic Growth Analysis (LGA) (Clark et al., 2011) and Duong Method (Duong, A.N., An Unconventional Rate Decline Approach for Tight and Fracture-Dominated Gas Wells, in Canadian Unconventional Resources and International Petroleum Conference, 2010). However, most of these DCA methods are not suitable for multiphase flow such as that in hydrocarbon reservoirs. Moreover, these methods are typically not capable of capturing complex production rate trends, such as sinusoidal type behavior. Other disadvantages of these techniques include the inability to predict from complex and heterogeneous reservoirs and sensitivity to production variation associated with changes in flowing bottom-hole pressures. Some specific machine learning (ML) models can address many of these disadvantages.

Time series forecasting is an important area of ML analysis. Proxy models that connect nodes along a temporal sequence (with new data integration) can be constructed via some subsets of artificial neural networks (ANN), such as a recurrent neural network (RNN) and its variants (e.g., the Long Short-Term Memory network, or LSTM) to predict time-dependent future performance. Specifically, LSTMs are capable of predicting output for several time steps with mutual dependence. A few other time series ML models developed recently include transformer (Vaswani and Attention is All you Need, in 31st Conference on Neural Information Processing Systems (NIPS 2017), I. Guyon, et al., , 2017), Temporal Convolutional Network (TCN) (Lea, C.,ECCV 2016 Workshops, , 2016), Convolutional Neural Networks LSTM (CNN-LSTM) (Yan, 2021). Neuro-fuzzy and its variants are becoming very popular for time series forecasting. Neuro-fuzzy systems were developed for chaotic time-series data that represent natural phenomena (Masulli et al., 1997). A k-nearest neighbor (k-NN) based neuro-fuzzy predictor was developed for time series prediction of stock closing prices (Wei et al., 2013; Peng, 2015). Some examples of LSTM applications include forecasting oil production rate (Song, 2020), predicting water table depth in agriculture areas (Zhang, 2018), describing dynamic features of wind (Qin, 2019), mid-to the long-term prediction of photovoltaic power generation (Han, 2019), estimating tourism rates over time (Li and Cao, 2018), forecasting spatiotemporal distributions of PM 2.5 (Tong, 2019), multi-step influenza outbreak forecasting (Kara, 2021), and a multi-time,

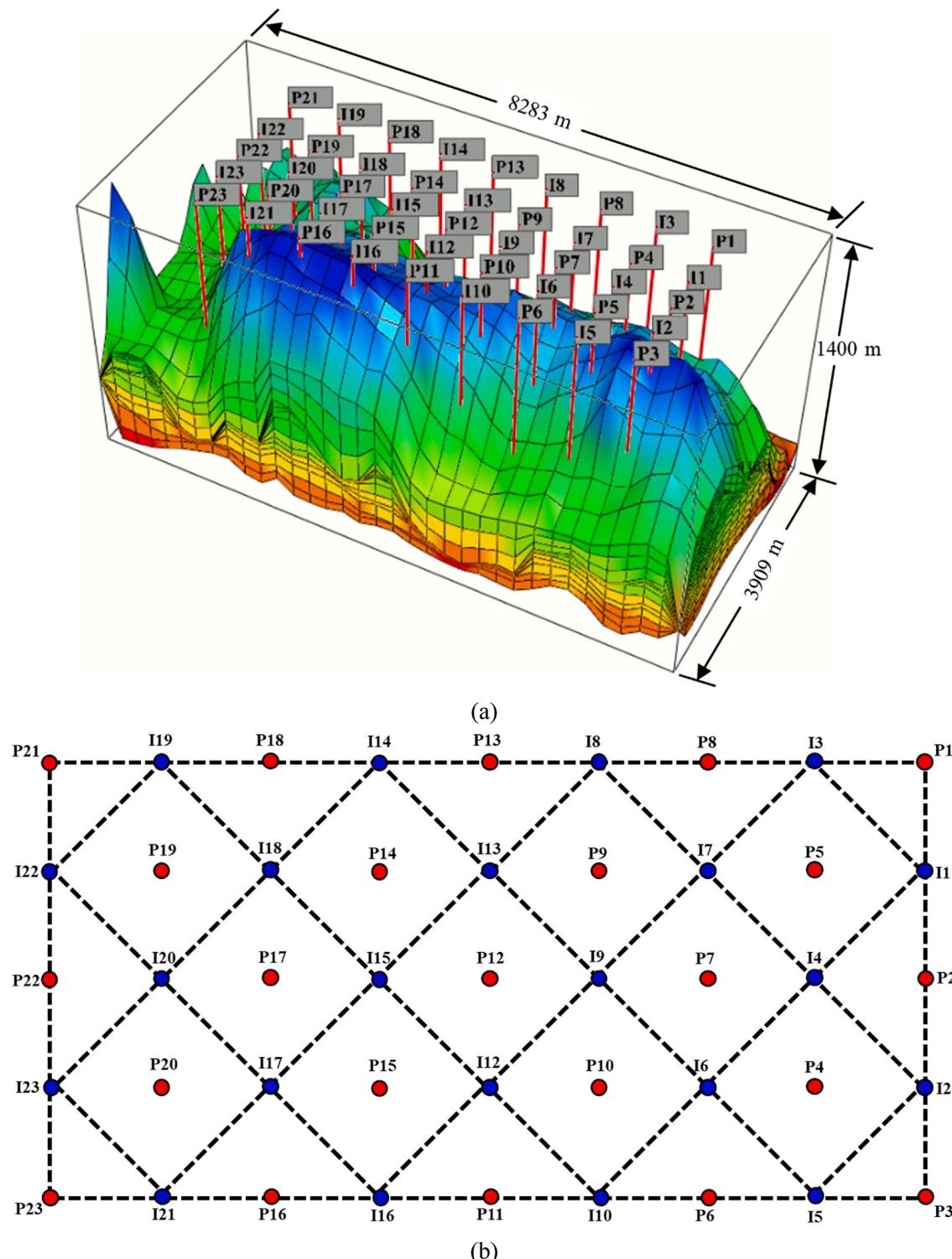


Fig. 1. (a) Three-dimensional reservoir model for the Northern Platform of SACROC (b) top view schematic of the 5-spot injection scheme with 23 producers and 22 injectors.

multi-site forecasting of Beijing's air quality (Yan, 2021). Several of these LSTM applications suggest that LSTM can be effective for quantifying the temporal dependence of sequential data.

Despite the success of LSTM application to time-series data, it is still challenging to predict the production rates of a reservoir with complex geology. For example, to forecast oil, water and gas production from a heterogeneous reservoir during enhanced oil recovery (EOR) often requires a network of LSTM models. Temporal memories of previous time

steps must be propagated to subsequent time, and this feature of LSTM models helps to preserve the variational nature of the data. Hence, an ML network consisting of LSTM networks is a viable approach to predict multi-phase flow (production rates) of oil, gas, and water. Two important variations of LSTM include bi-directional and stacked. In a stacked LSTM, multiple units of LSTM blocks are vertically assembled. In a bi-directional LSTM, temporal information is passed back and forth between two LSTM blocks. A primary objective of this study is to

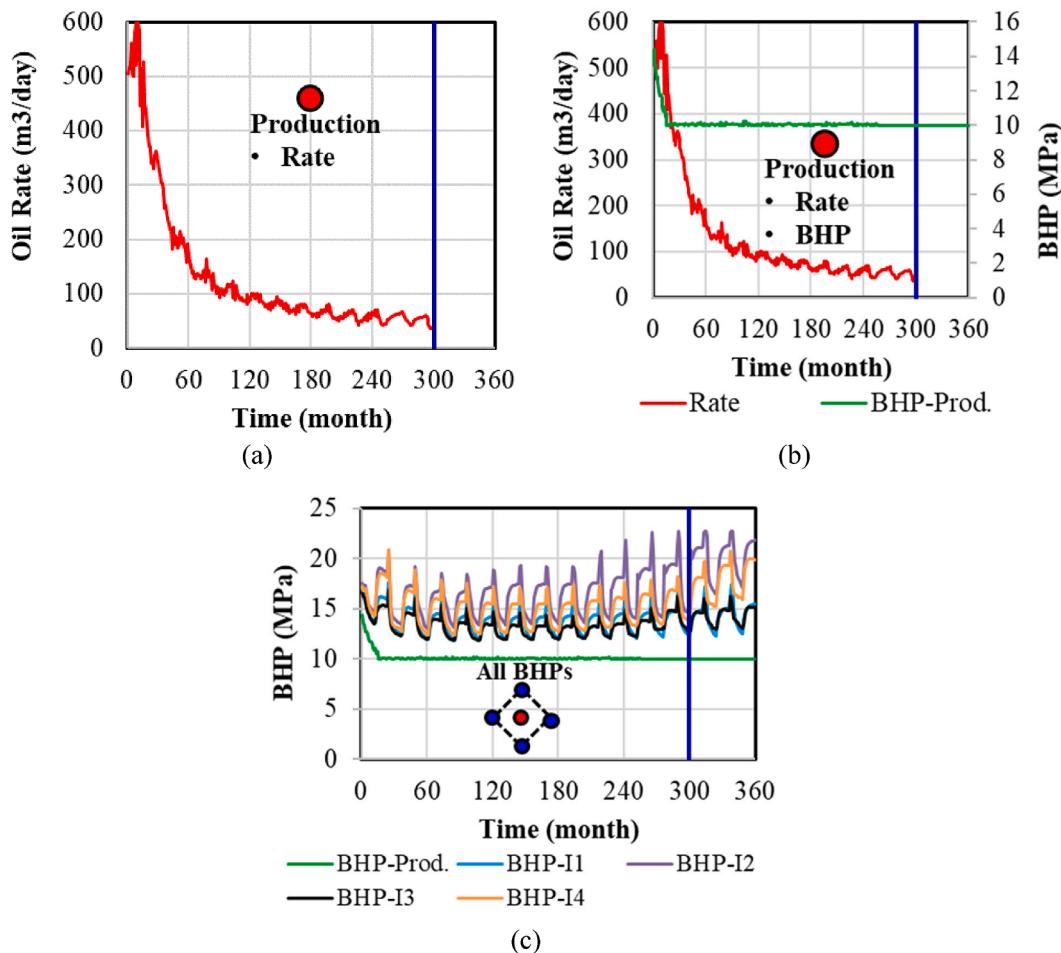


Fig. 2. Combination of input data for LSTM model to predict production rates of oil, gas, and water from producers (a) only production rate from producer (b) production rate and BHP from producer (c) BHP of producer and BHPs of surrounding four injectors.

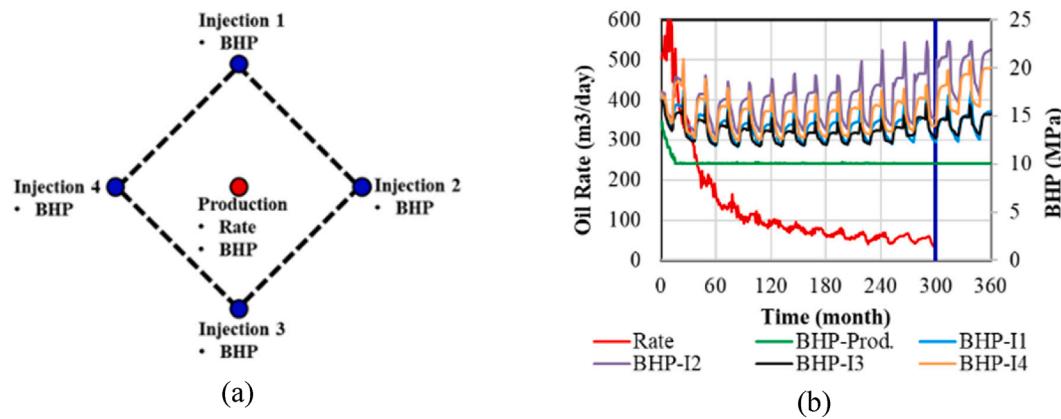


Fig. 3. (a) 5-spot injection scheme of a single producer (b) input data for training testing of the model.

demonstrate the capability of stacked LSTM models to forecast production rates from a complex oil reservoir, including associated complicating aspects of geology, operation, and field development. The traditional time-series model, ARIMA, is also developed to compare the

performances of the LSTM model. The Scurry Area Canyon Reef Operators Committee (SACROC) field with multiple production and injection wells as described in the next section is selected as a case study because of its heterogeneous geology and complex operations. Among SACROC's

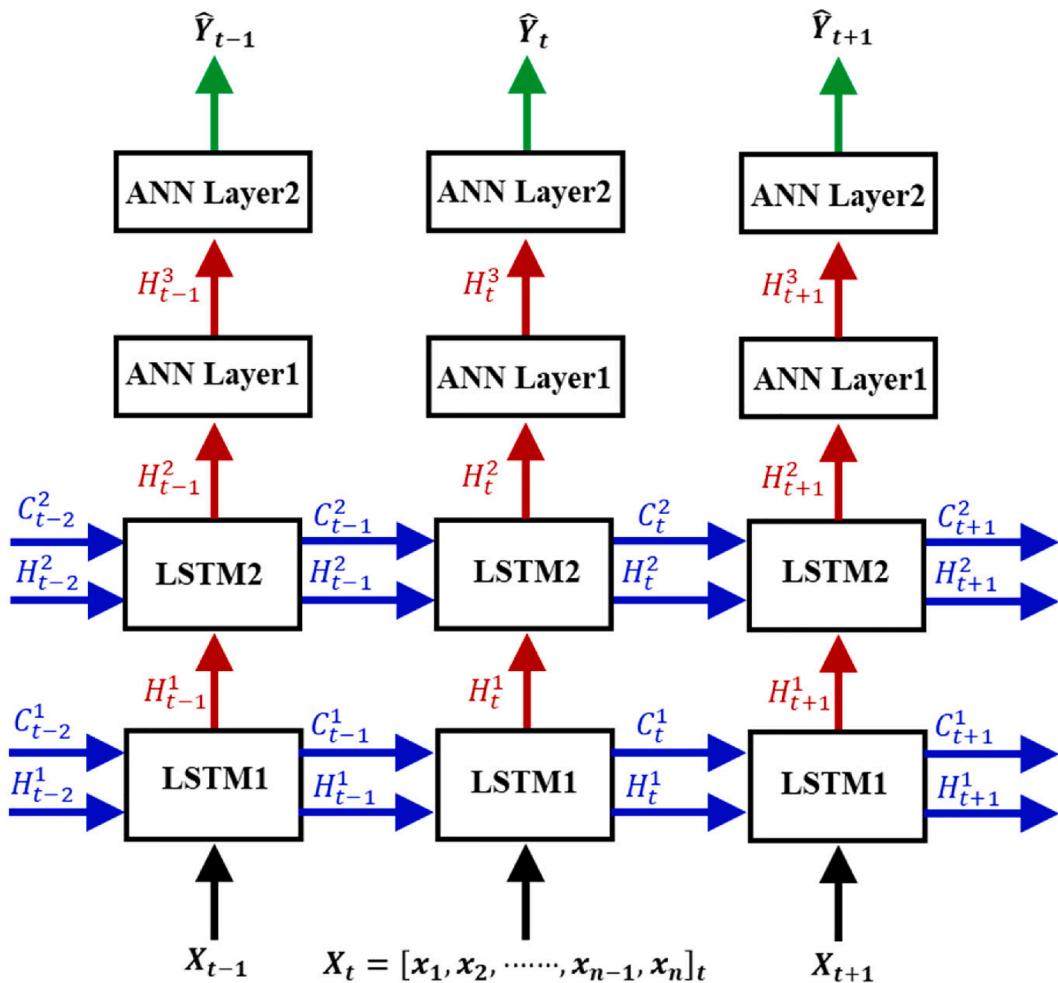


Fig. 4. The network structure of stacked LSTM model for prediction of production rates in SACROC.

Table 1
Number of ML parameters in each layer.

ML layer	Number of neurons	Number of parameters
LSTM 1	100	42,800
LSTM 2	100	80,400
ANN Layer1	100	10,100
ANN Layer2	1	101
Total	301	133,401

A total of 133,401 parameters of the ML model is obtained during the training process.

complexities, of interest is its five-spot injection scheme with water and CO₂ as alternating injection fluids and the randomized shut-in of producers. The LSTM model developed here appears to be a promising ML algorithm for predicting oil, gas, and water production rates from this and other similar complex fields.

2. Reservoir model

The Scurry Area Canyon Reef Operators Committee (SACROC) oil-field is located on the eastern edge of the Permian Basin, Texas, USA.

Enhanced oil recovery (EOR) by CO₂ injection was initiated at SACROC in 1972. A reservoir model of the Northern Planform of SACROC (Fig. 1a), is simulated with the CMG-GEM package (Computer Modeling Group: GEM and Unconventional, 2020) to generate data for an ML algorithm. The reservoir model here (Fig. 1a) is modified from previous research on SACROC performed as part of the Southwest Regional Partnership on Carbon Sequestration (SWP), a 20-year geologic CO₂ demonstration program sponsored by the U.S. Department of Energy. The SACROC project, including its geologic setting, model development and calibration and other associated details, is described in several publications, for reference (Han, 2010; Jia, 2018; Jia, 2016).

The SACROC model grid consists of 34 columns of cells in the x-direction (total 8,283 m), 16 columns of cells in the y-direction (total 3,909 m), and 25 layers in the z-direction (total 1,400 m). Its well patterns include 23 producers and 22 injectors. Within the 25 layers of the model, well perforations are limited to layers 19, 20, 21, and 22, based on simulated well data. All wells are placed in 5-spot injection patterns, as shown in a top view (Fig. 1b). Injected fluids are alternated between CO₂ and water, i.e., CO₂-WAG, on an annual basis. Producers are shut-in and opened randomly for a few months to mimic the actual field scenarios, based on the actual production data. The model duration is 30 years, representing production from 1 January 1972 to 1 January 2002.

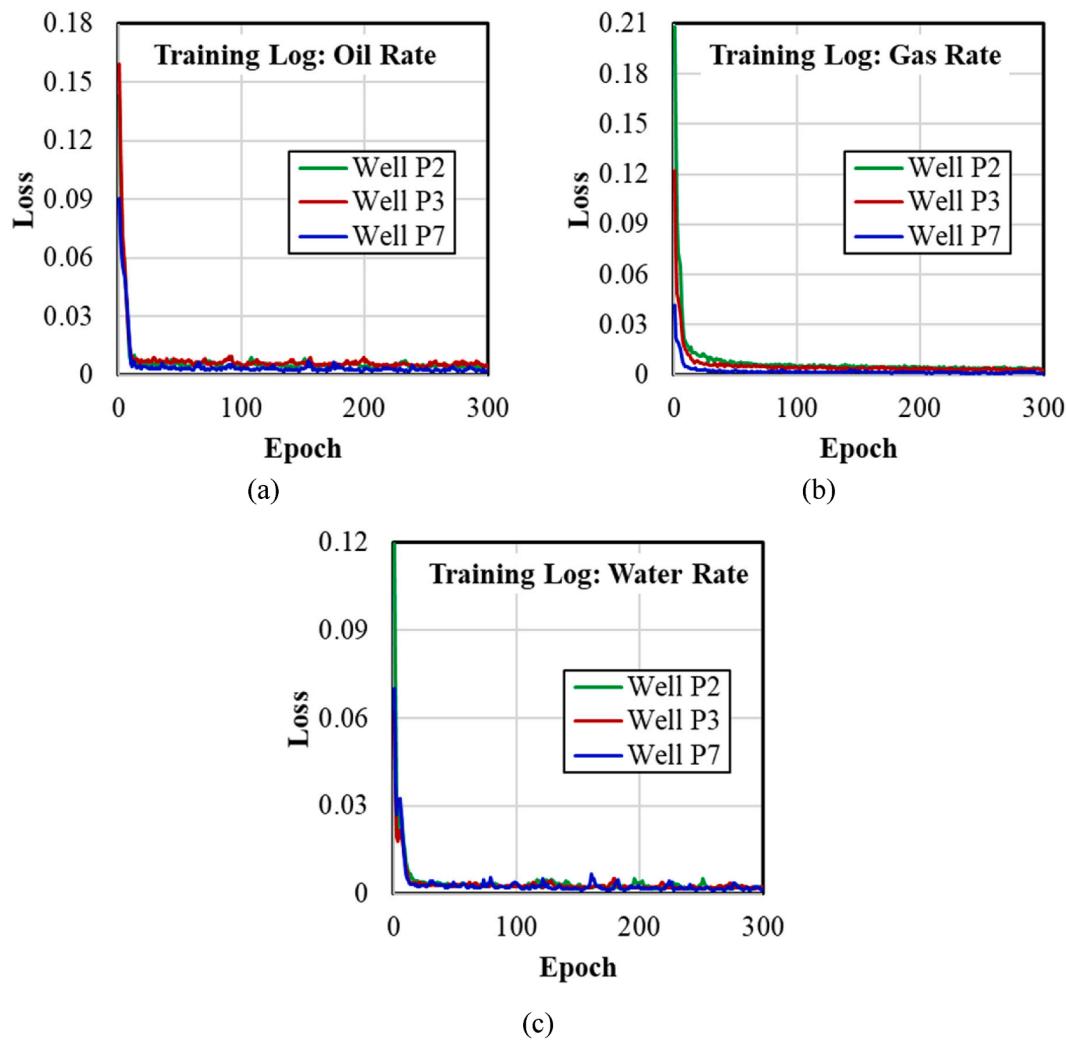


Fig. 5. Minimization of loss function during training of LSTM models (a) oil production rate (b) gas production rate (c) water production rate.

This reservoir model and associated data may be accessed at NREL's Energy Data eXchange (EDX) ([EDX: NREL's Energy Data eXchange, https://netl.doe.gov/edx](https://netl.doe.gov/edx), accessed in, 2020).

3. Machine learning model

3.1. Input selection and preparation

A feature selection method is employed to reduce the number of relevant features. This approach has many advantages including model simplification, faster training, increased interpretation ability, increased accuracy (based on our results), and reduction of overfitting. A few methods of feature selection include filter-based on a uni-variate metric (variance, chi-square, correlation coefficients, mutual information), wrapper method (exhaustive search, forward search, backward search), embedded method (decision tree, lasso regularization, ridge regularization) (Kumari and Swarnkar, 2011; Guyon et al., 2006; Khandelwal, 2019; Galelli, 2014; Fernando et al., 2009). Parameter choices affect the production and injection rates and thus require effective characterization of the hydrodynamics of SACROC under its operational variability.

In the 5-spot injection scheme (Fig. 1b), it is evident that the rate of production depends not only on the bottom hole pressure (BHP) of the producer but also on the BHPs of the surrounding 4-injectors. However, it is also possible that operations of other producers and injectors that are not immediate neighbors can influence production. Such may be caused by flow connection or pressure front propagation through high permeability zones, including faults or other similar features. Therefore, input selection for ML algorithms is typically not straightforward, especially for complex systems. In this study, input data are selected based on our knowledge of production operation in an oil & gas field.

It is also common practice to choose the history (older time steps) of the output as an input in an LSTM model. Therefore, one production rate and five BHPs (one producer and four injectors) are the potential input data for an LSTM. To quantify the dependency of these input parameters on the output, a few combinations of input parameters are investigated initially. Three such schemes tried (but not selected in the final model) are shown in Fig. 2.

The BHP of the producer alone could not predict the production rate from the well, because the BHP of the producer is often constant at the minimum allowable value in later stages of production, albeit the

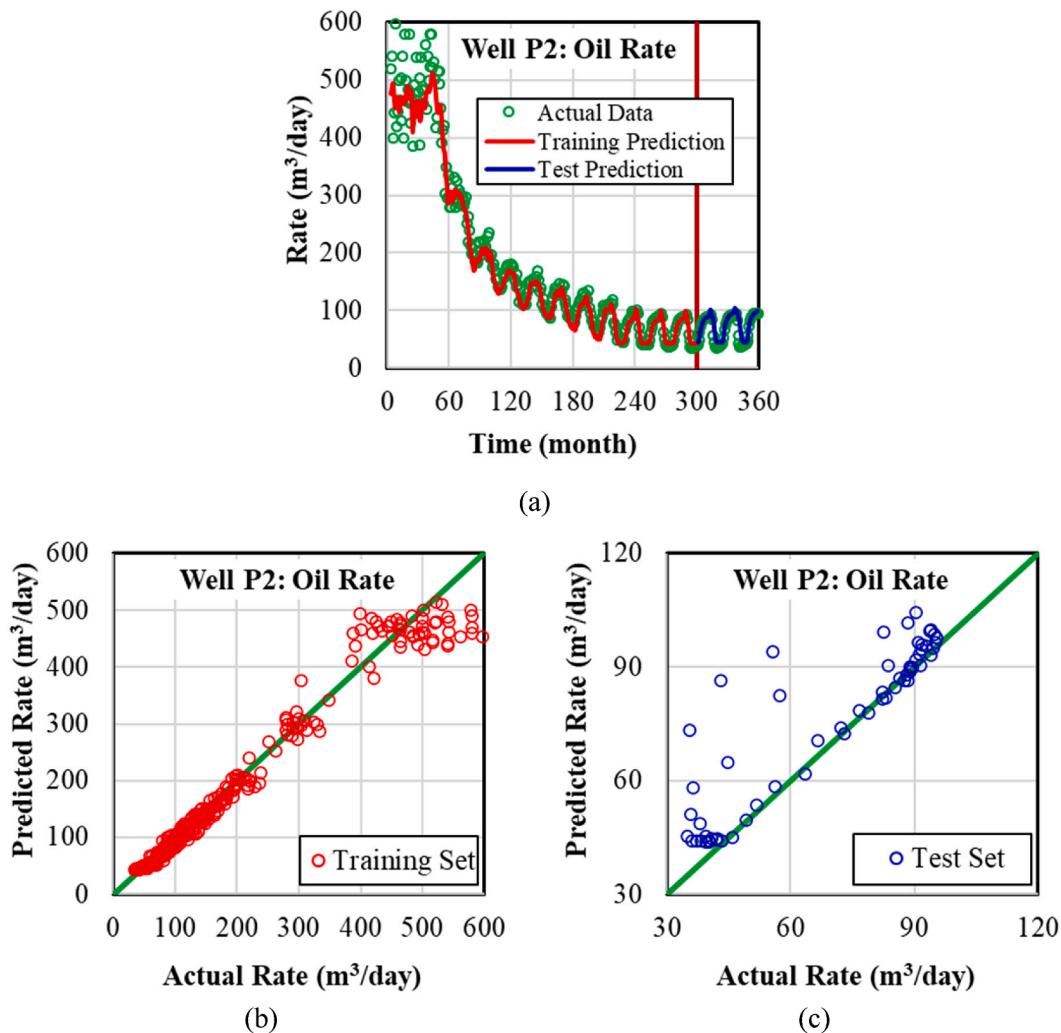


Fig. 6. Comparison of predicted oil production rates and simulated oil production rates from well P2 (a) time series production rates for the training and test sets (b) cross plot for the training set (b) cross plot for the test set.

production rate varied over time due to impacts from other wells. Another reason is that the BHP itself is not the primary driving force for fluid flow, but rather the drawdown, i.e., the difference between reservoir pressure and the BHP. However, it is not possible to include reservoir pressure in well-based LSTM models because it is spatially variable. Similarly, using both the BHP of the producer and the production rate as input data (Fig. 2b) could not predict the future production rate correctly for the same reason. Another analysis was conducted with all five BHPs used as input, but with production rate excluded (Fig. 2c), and this model failed to predict the future production rate. These results suggest that the inclusion of the production rate is critical (but not the only parameter as input) because it facilitates both capturing the complexity of the reservoir and describing the fluid flow. After these preliminary investigations, six input parameters (five BHPs and one production rate) are selected, as shown in Fig. 3.

Because the production rate is the output of the LSTM model as well, it is only used during the training of the model and not used during the blind testing, i.e., the model has never “seen” the testing production rate as shown in Fig. 3b. The vertical line (blue) separates the training and

testing data sets, and the production rate is discontinued in the testing zone. Only BHPs are used during the testing. However, as the model requires six input data (i.e., production rate as well) as it is trained, the predicted rates from previous time steps are fed back as input data.

Data in the majority of data analytic and machine learning tools are scaled for various reasons. There are two popular scaling methods, standardization and normalization. The standardization method is generally applied to normally distributed data to transform the distribution to another normal distribution with a mean of zero and a standard deviation of 1. In the second scaling method i.e., normalization, the data are linearly transformed between a given range of minimum and maximum values. The data normalization method is selected for scaling the time series data in this study. The minimum/maximum/range values depend on various factors including model parameters and the physical system. For example, negative values of some physical properties (for example, porosity) are meaningless. In the context of this study, production rates (without the direction consideration) and pressures are always positive. Different variables such as bottom hole pressure and production rates are scaled to the same range to minimize potential bias

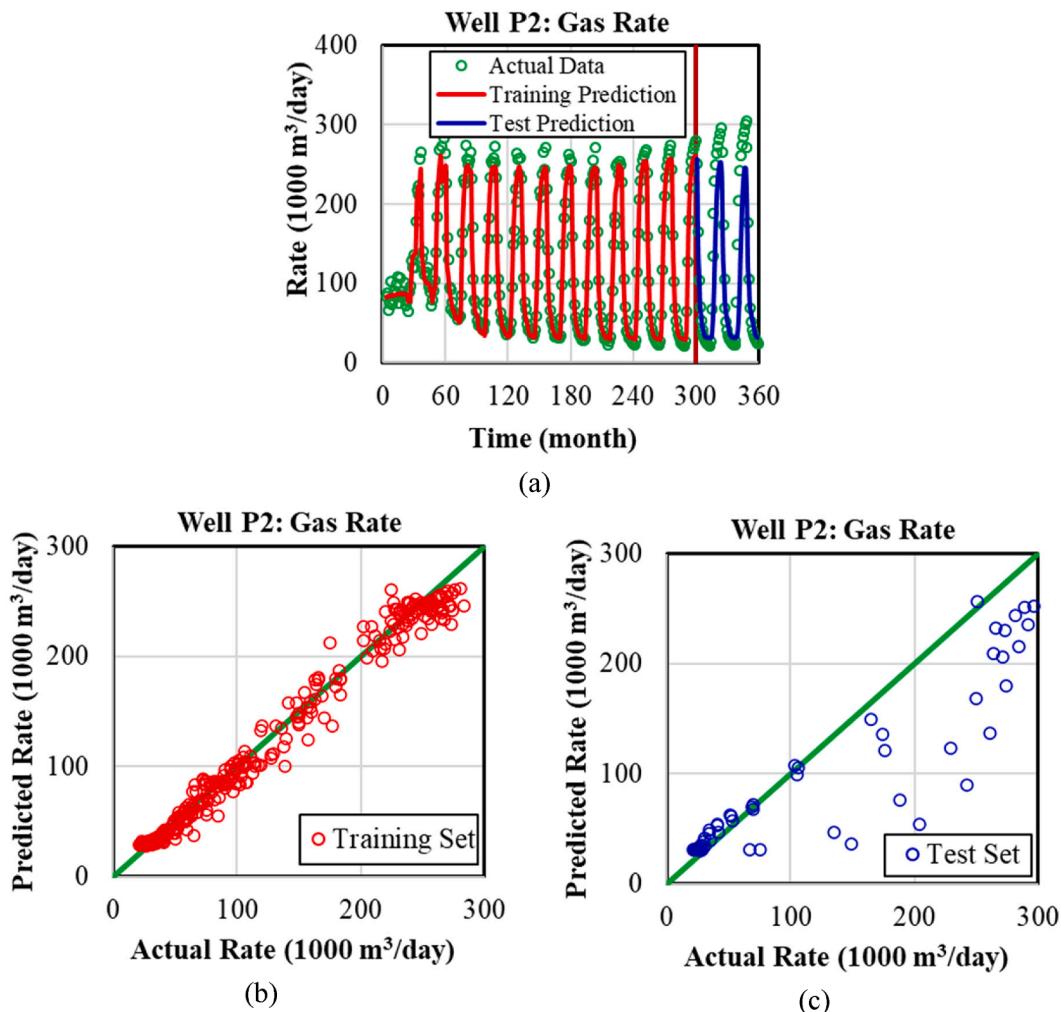


Fig. 7. Comparison of predicted gas production rates and simulated gas production rates from well P2 (a) time series production rates for the training and test sets (b) cross plot for the training set (b) cross plot for the test set.

during the training of a model. For example, the BHP is in the approximate range of 10,000–20,000 kPa, whereas the oil production rate ranges between about 0 – 600 m³/day. These time series must be normalized by scaling with the minimum and maximum values to weigh all values between 0 (minimum) to 1 (maximum). An activation function is used in machine learning algorithms to convert a piece of information. Various activation functions have different output ranges, for example, sigmoid: 0 to 1, hyperbolic tangent: -1 to 1, and, Rectified Linear Unit (ReLU): zero (for negative input) to positive. Therefore, the data of the LSTM model must lie within the range of the activation function. Consequently, bottom-hole pressures and production rates are scaled between 0 and 1 which is suitable for the activation function (ReLU) used in this study.

The current production rate is never used to predict the current production rate (itself) for obvious reasons. In the time series machine learning model, a few (usually three to five) previous time steps values, which are the output of models at previous time steps, are typically considered as input to predict the current output. Variation of production rate with the time is often a critical factor in choosing the number of data points of historical production rates to prepare an input sample for

the machine learning model. Typically, after primary production, the production rate is then varied with the injection schedule. Therefore, the frequency of production rate variation is one year which corresponds to the annual change of injection fluid (CO₂ and water). There are 12 data points (one for each month) in this part of the trend. Selecting too many historical points (greater than 12 in this case) would include different trends in the same input sample. Consequently, the model would be ineffective with respect to manifesting the variation in the production rate. Hence, it is essential to select optimum historical points in the input sample to represent the trend in the production rate.

In this study, an approach of 3-step lag values of production rate (i.e., one-fourth of data representing a certain trend) is adopted. Therefore, although only six primary input parameters are initially selected, there is a total of eight input values to the LSTM model, as written in Equation (1). The left side under braces is the input set, and the right-hand side under braces is the output set, where I₁ to I₄ denote the BHP of the four injectors, P the BHP of the producer, R, and R' the previous and predicted production rate of the producer, and the numbers in the parentheses denote the time steps.

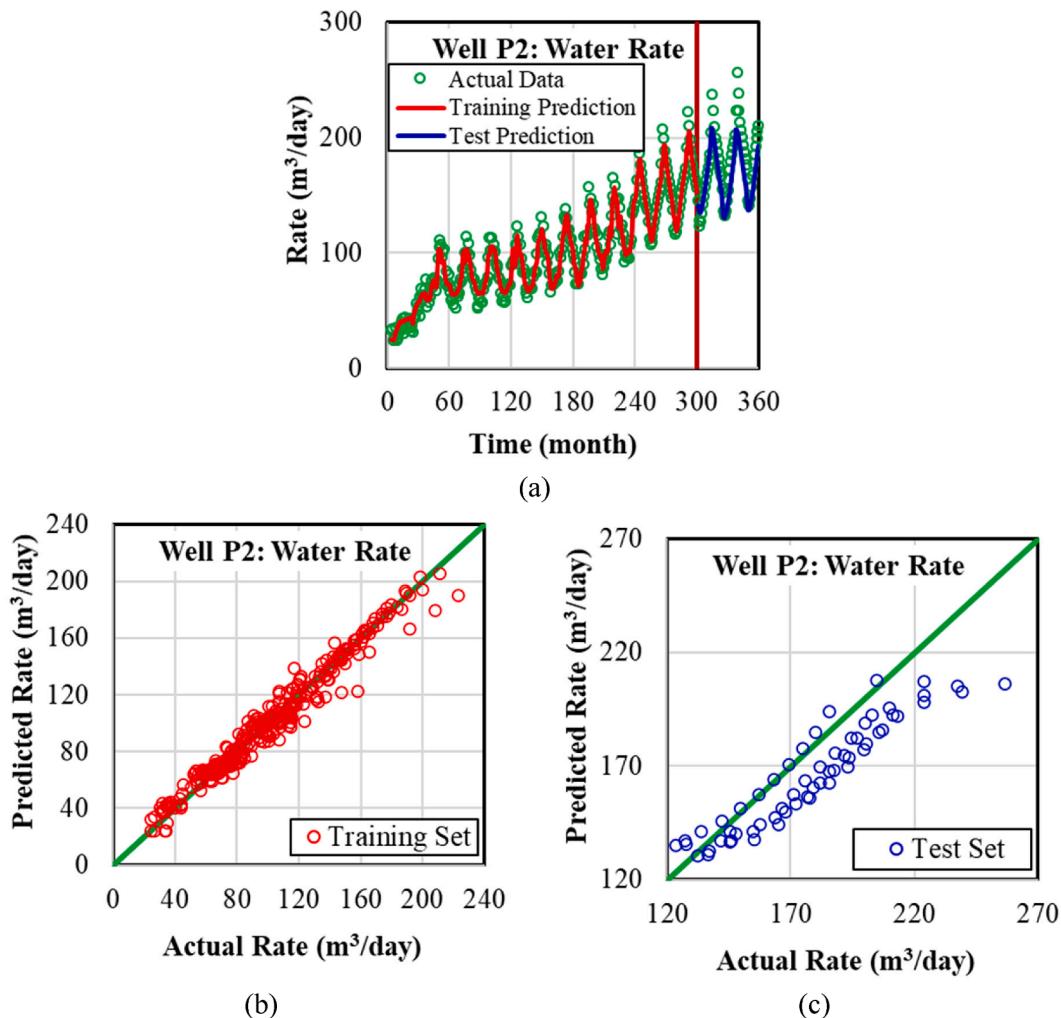


Fig. 8. Comparison of predicted water production rates and simulated water production rates from well P2 (a) time series production rates for the training and test sets (b) cross plot for the training set (b) cross plot for the test set.

$$\left\{ \begin{array}{ccccccccc} I1(4) & I2(4) & I3(4) & I4(4) & P(4) & R(1) & R(2) & R(3) \\ I1(5) & I2(5) & I3(5) & I4(5) & P(5) & R(2) & R(3) & R(4) \\ \vdots & \vdots \\ I1(t) & I2(t) & I3(t) & I4(t) & P(t) & R(t-3) & R(t-2) & R(t-1) \end{array} \right\} \overbrace{\left\{ \begin{array}{c} R'(4) \\ R'(5) \\ \vdots \\ R'(t) \end{array} \right\}}^{\text{Output}} \quad (1)$$

Because a 3-step lag is chosen, the production rate at the 4th-time step is predicted first during training. All BHP values at the current time step are fed as input data. The first row in Equation (1) shows that the five BHP at the 4th-time step and production rates at the 1st, 2nd, and 3rd-time steps are the input data to predict the production rate at the 4th-time step. A similar explanation applies to every row in this equation. Note that the last production rate is never used as input during training, and the prediction starts from the production rate at the 4th time step. The input preparation during testing is similar to training except for the production rate. The test production rates are kept hidden

from the model during testing, as shown in Fig. 3b. The input data during testing are shown in Equation (2). While the first row is similar to the training input set, the other rows are different. In the first row, the last three production rates from the training set along with the five BHPs are fed as input data. As a time step progresses in the second row, the predicted production rate from the previous step is used as one of the input data. Similarly, in the third row, two predicted production rates from previous time steps and three predicted production rates in the fourth row and onwards are used as input data.

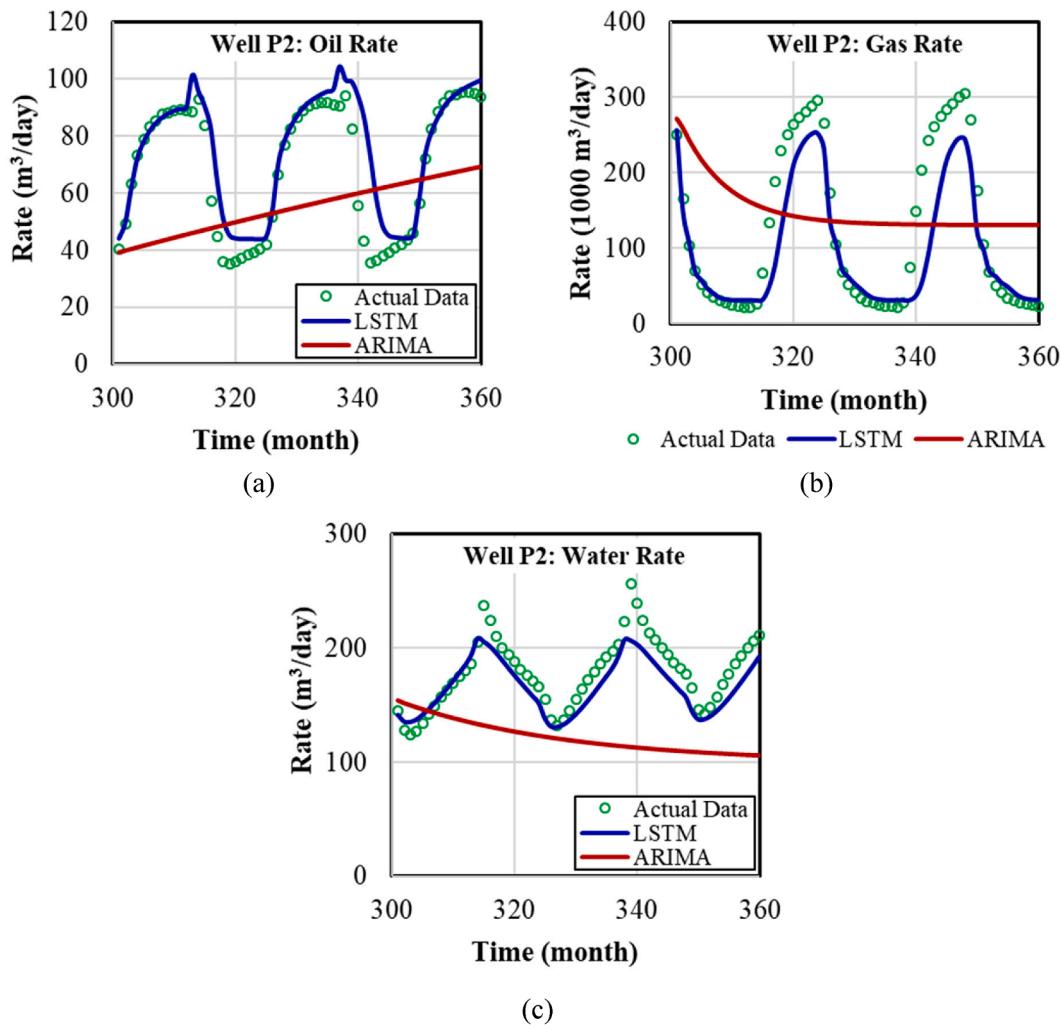


Fig. 9. Comparison of simulated production rates from well P2 with predicted production rates from LSTM and ARIMA models (a) oil production rate (b) gas production rate (c) water production rate.

$$\left\{ \begin{array}{ccccccccc} I1(t+1) & I2(t+1) & I3(t+1) & I4(t+1) & P(t+1) & R(t-2) & R(t-1) & R(t) & R'(t+1) \\ I1(t+2) & I2(t+2) & I3(t+2) & I4(t+2) & P(t+2) & R(t-1) & R(t) & R(t+1) & R'(t+2) \\ \vdots & \vdots \\ I1(t+n) & I2(t+n) & I3(t+n) & I4(t+n) & P(t+n) & R'(t+n-3) & R'(t+n-2) & R'(t+n-1) & R'(t+n) \end{array} \right\} \quad (2)$$

Note that not all producers are surrounded by four injectors (Fig. 1b). The well patterns may be divided into three types, depending on the accompanying injectors. Eleven wells (P4, P5, P7, P9, P10, P12, P14, P15, P17, P19, P20) are surrounded by four injectors, eight wells (P2, P6, P8, P11, P13, P16, P18, P22) by three injectors and four wells (P1, P3, P21, P23) by two injectors. The input vectors for training and testing in Equations (1) and (2) are adjusted accordingly.

3.2. Network structure and implementation

After selecting input parameters for the machine learning algorithm, a network structure for the stacked LSTM model is developed by adding multiple units. It is also important to optimize the number of total parameters (weights and biases) in the network for efficient training of the model. A possible ML network that is optimized for time to train the model with reasonable accuracy is depicted in Fig. 4.

In this machine learning architecture, two LSTM units are vertically

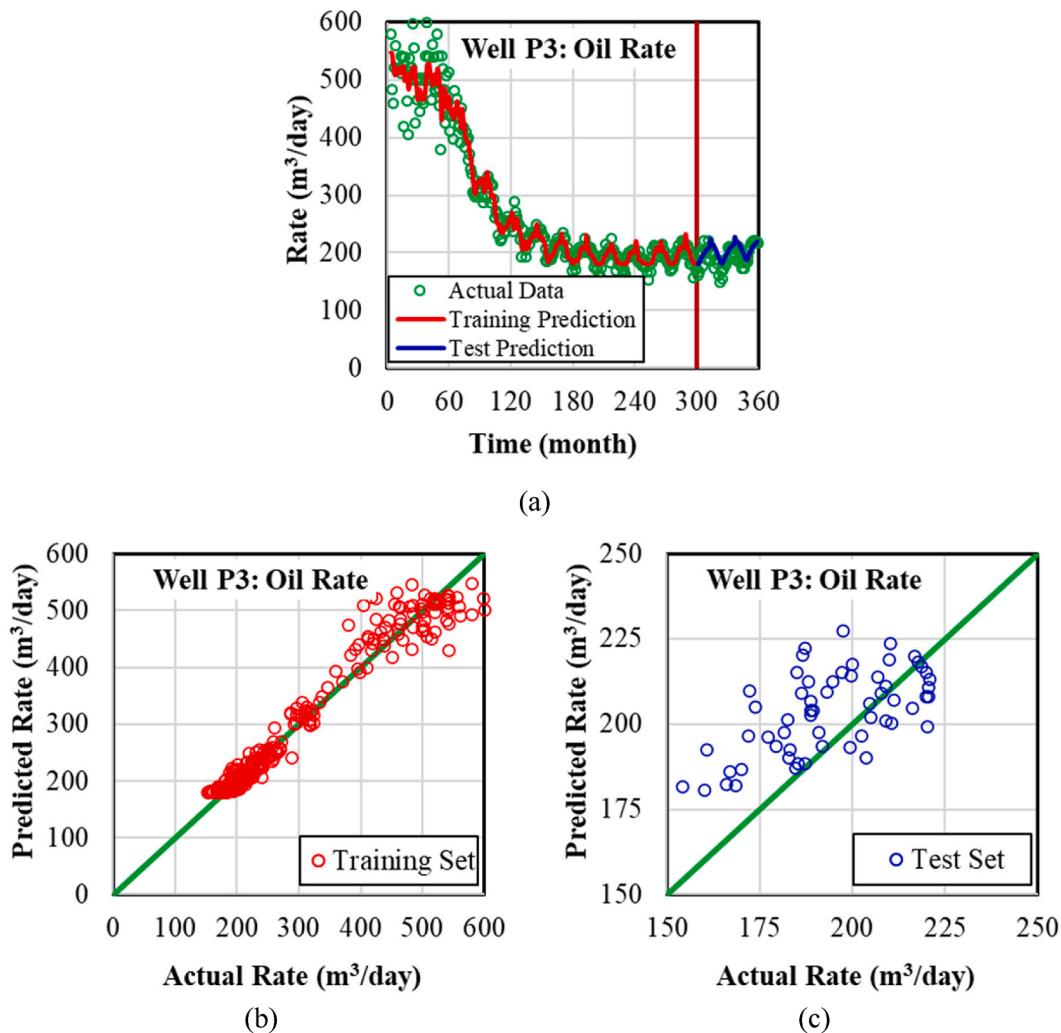


Fig. 10. Comparison of predicted oil production rates and simulated oil production rates from well P3 (a) time series production rates for the training and test sets (b) cross plot for the training set (b) cross plot for the test set.

stacked for the same time step. A summary of LSTM theory is provided in Appendix B. A vector of input data (X_t) for each time step that corresponds to an input row in Equation (1) during training (and Equation (2) during testing) is set in the bottom LSTM. The hidden outputs from the bottom LSTM block go to the upper LSTM block. On the other hand, both the hidden output and the cell states are reassigned sidewise to the adjacent LSTM for the next time step. Therefore, in a stacked LSTM, information is propagated unidirectionally upward and sidewise. The sidewise flow of information carries temporal “memories” from all previous time steps. Another layer, a Multi-Layer Perceptron (MLP) unit above the upper LSTM unit, is included to express the intricate relationship among the input data. Finally, an activation function, Rectified Linear Unit (ReLU), resides at the top of the ML network. The addition of an activation function at the end of the network in the LSTM algorithm is a common practice to reduce hidden output to the desired number of final output results. As described earlier, all input data and output are normalized to lie between 0 and 1 as part of the data preparation. Therefore, the ReLU activation function that always has positive output is consistent with the data in this ML algorithm. The final output (from

ReLU) for each time step is compared to the actual value to optimize the model parameters (described in more detail later in this paper). For practical purposes, the final output is often transformed back to real values.

The architecture is implemented in Spyder (Spyder, 2020), an open-source, cross-platform integrated development environment (IDE) for programming in Python from Anaconda Navigator (Navigator, 2020). On this platform, all necessary ML environments are loaded in the backend. Keras (Keras, <https://keras.io/>, accessed in, 2020), a machine learning Python library, is used to develop and evaluate the LSTM model in Spyder. Keras runs on top of an efficient numerical computation library, TensorFlow (TensorFlow, <https://www.tensorflow.org/>, accessed in, 2020).

Deciding how many neurons (Panja, 2018; Dinkelbach, 2012) that will reside in hidden layers in each block is an important component of ML model development. Too few neurons for a complicated system may not adequately capture the variation in a data set, and the data set suffers from underfitting. On the other hand, overfitting occurs when too many neurons are used. Moreover, computation time is rapidly

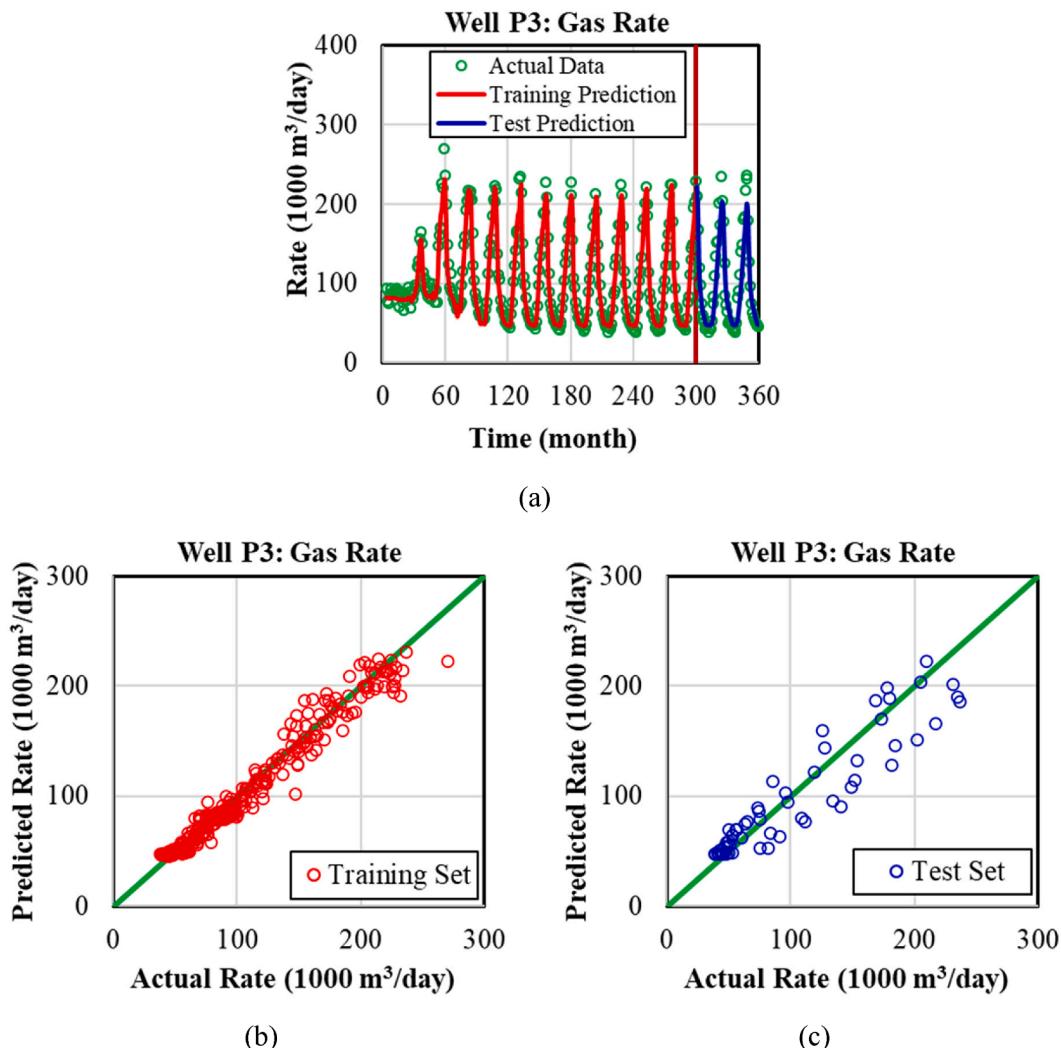


Fig. 11. Comparison of predicted gas production rates and simulated gas production rates from well P3 (a) time series production rates for the training and test sets (b) cross plot for the training set (b) cross plot for the test set.

increased with the number of neurons (Dinkelbach, 2012). Although there is no definite rule of thumb for choosing the number of hidden neurons, some formulas can be used as guidance. In this study, the number of neurons in each block (LSTM and MLP) is determined by trial and error. A total of 100 neurons in each layer are chosen after the initial screening with 20 to 200 neurons. Dropout is a popular regularization strategy to avoid overfitting, where randomly selected neurons are dropped out during training (Brownlee, 2018). The contributions of the dropped neurons to the downstream activation function are ignored. A dropout production rate of 20% is optimum for this system, determined by trial and error and varying between 10% and 40%. The number of neurons and parameters in each layer of the stacked LSTM network is summarized in Table 1.

The model is compiled using adaptive moment estimation (Adam optimizer) (Kingma, 2014) with mean squared error (MSE) as a loss function, as shown in Equation (3).

$$\text{MSE} = \frac{\sum_{i=1}^n (Y_{\text{obs},i} - Y_{\text{model},i})^2}{n} \quad (3)$$

Y_{obs} and Y_{model} are the simulated value and predicted value from the model, respectively; n is the number of data points (different for training and test sets) in the calculation.

3.3. Training & testing

The monthly averaged production rates of oil, gas and water from 23 producers were used for the ML algorithm. Therefore, 360 data points for each production rate were available for the 30 years of operation at SACROC (from 1st January 1972 to 1st January 2002). Approximately 83% of the total, or 300 months (from 1st January 1972 to 31st December 1996), were used as the training set, and the remaining 60 months (from 1st January 1997 to 1st January 2002) of production rates were used for testing. During the training process, the loss function (based on scaled values) is observed for a stable minimum value (Fig. 5).

It is important to find an optimum number of epochs which is defined as one complete cycle of all training data (forward pass) including error gradient (backward pass) through the neural networks (Afaq and Rao, 2020). Too few epochs can terminate the training process immaturely,

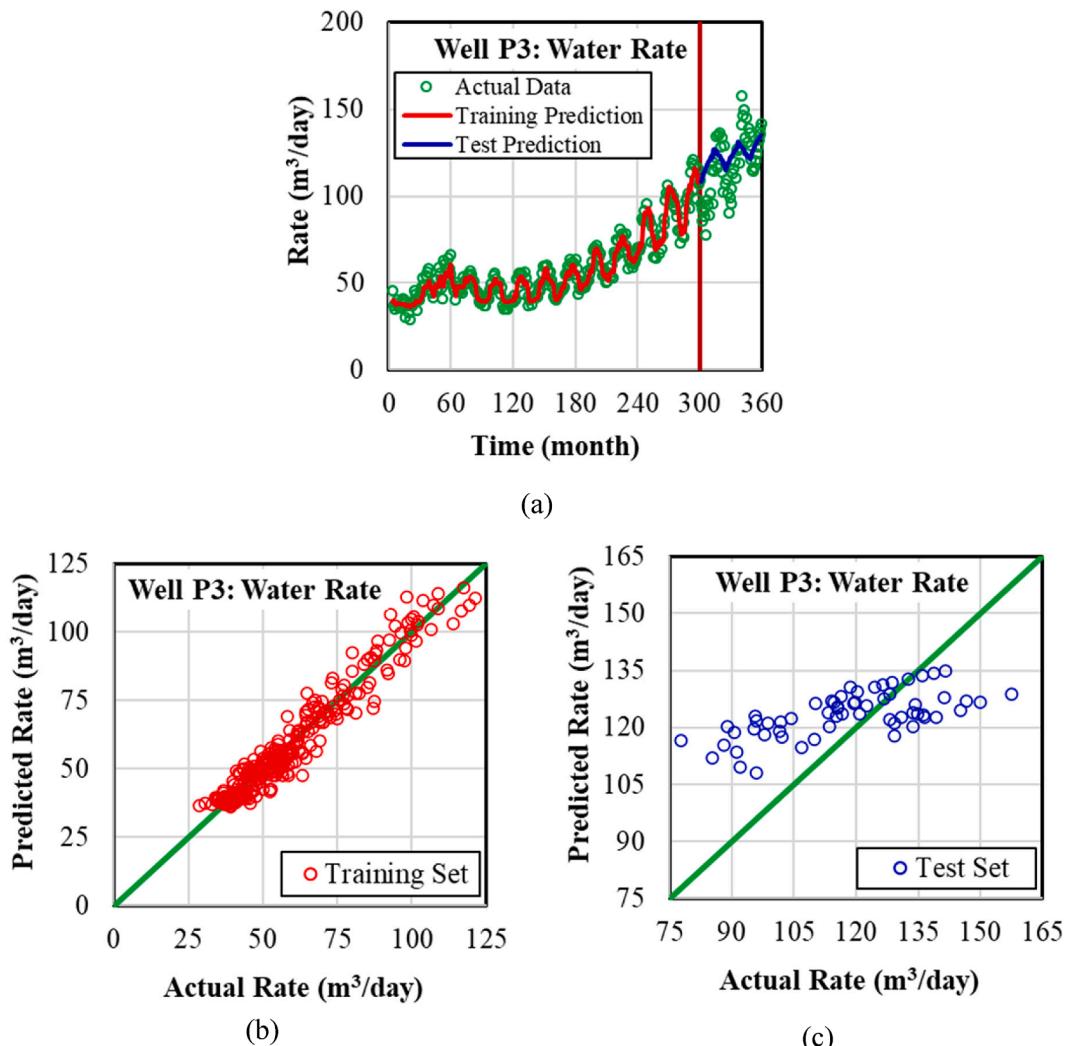


Fig. 12. Comparison of predicted water production rates and simulated water production rates from well P3 (a) time series production rates for the training and test sets (b) cross plot for the training set (b) cross plot for the test set.

and too many epochs require excessive time to finish the training. Therefore, 50–1000 epochs were tried initially, and it was found that 300 epochs are sufficient for all production rates. After the ML model was trained successfully with acceptable accuracy such as a minimum achievable value of loss function (mean squared error of less than 1%), it was used to predict the test data set that has been kept unseen by the model.

4. Results and discussions

Three models were developed for three production rates (oil, gas, and water) for each well. Therefore, a total of 69 models were trained for 23 producing wells. Instead of presenting the results of all 23 producers, only the results of three representative wells are selected: P7 (producer surrounded by four injectors), P2 (producer surrounded by three injectors), and P3 (producer surrounded by two injectors). However, the error analyses of all 23 wells (69 models) are presented later.

4.1. Production forecast

Simulated and predicted production rates of oil, gas, and water from each of the three selected wells (P2, P3, and P7) are plotted for the entire 360 months (Figs. 6–16). The production rate curves show unusual trends due to the irregular open and shut-in timing of all production wells and due also to annual changes in injection fluids (CO₂-WAG). The performance of each model is described with two types of plots: a time series plot and a cross plot between simulated and predicted production rates. Variations of relative errors with time are also summarized in Appendix A. The models inherently exhibit high fitness with training data because those data are already considered when (used for) developing the model. Out-of-sample (“unseen” test data) prediction provides a means of evaluating the model’s robustness. Therefore, predictions of training and testing sets are shown separately. In the time series plots, a vertical line at the 300th month separates the training and testing periods.

The production rates for well P2 are shown in Figs. 6–8. For higher initial reservoir pressure, the oil production rate is higher in the first 3–4

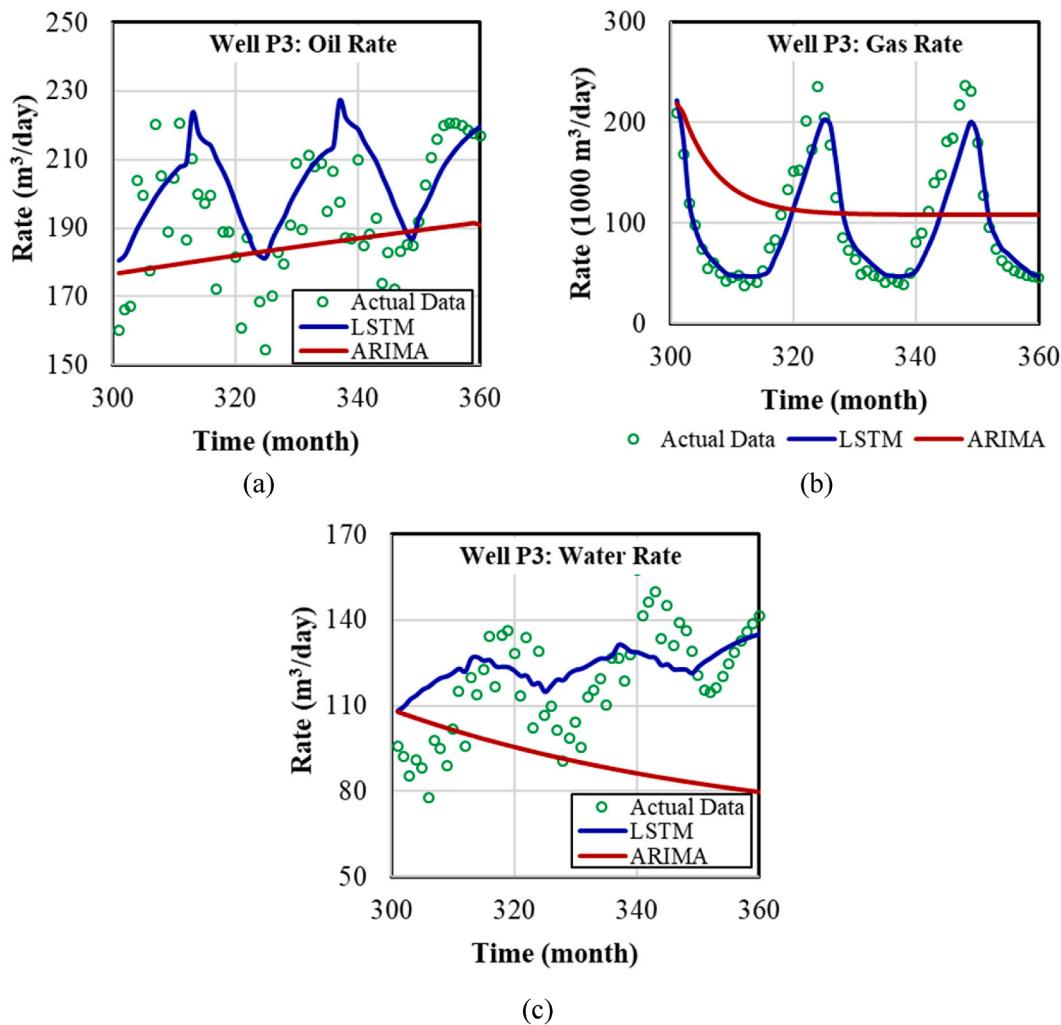


Fig. 13. Comparison of simulated production rates from well P3 with predicted production rates by LSTM and ARIMA models (a) oil production rate (b) gas production rate (c) water production rate.

years, and it gradually declines as the reservoir pressure drops (Fig. 6a). However, an operator may also maintain a higher drawdown by reducing the BHP to produce at an economical production rate. Additionally, injection in the 5-spot pattern also supports an increase in the production rate. The combined effects are observed in the first 5–6 years (60–72 months), as exhibited by Fig. 6a. Eventually, the BHP of the producer reaches a minimum allowable value (10,000 kPa), and the injection from surrounding wells dominates the flow rate from the producer. These effects are typically observed in the later stages of a well. Consequently, the production rate fluctuates following the pressure pattern in the injection wells and the type of injection fluid. This type of complex pattern of oil production rate is difficult to predict using a traditional time-series model. However, the LSTM models in this study show a promising result. The difference between the simulated and predicted values is higher at the start of the training set, and it diminishes at a later time.

It is evident from the cross-plot in Fig. 6 b and c that the predicted values (training and test both) do not deviate much from the simulated values. Most of the predictions of oil production rates from well P2 fall between $\pm 30\%$ relative errors, as shown in Fig. A.1a (Appendix A).

The gas production rate from well P2 starts increasing after an initial flat plateau until about 30 months, as shown in Fig. 7 a. The maximum and minimum production rates remain at stable values. The production of gas is from two sources, in-situ and injected. The amount of injected gas reaching the production well depends on the geology of the reservoir. Considering the heterogeneous nature of the SACROC field, an uneven distribution of gas towards production wells is expected. The annual injection schedule of CO₂ greatly affects the pattern in the gas production rate. Therefore, gas production rates from all 23 wells are different in amount and nature. The periodicity of injection and the variability in the production rate may not be in a synchronized state because of the time delay (time of migration) for the injected fluid to reach the production well. In addition to heterogeneity, fluid properties also have significant control over gas production. For example, water is supposed to take a longer time to reach the production well for its lower mobility, but on the other hand, gas migrates faster from an injection well to producers for higher mobility. Nevertheless, the LSTM model predicts the gas production rate and its variations with minimum errors. As expected, a higher error is induced for the test set (Fig. 7 d cross plot, and relative error in Fig. A.1b).

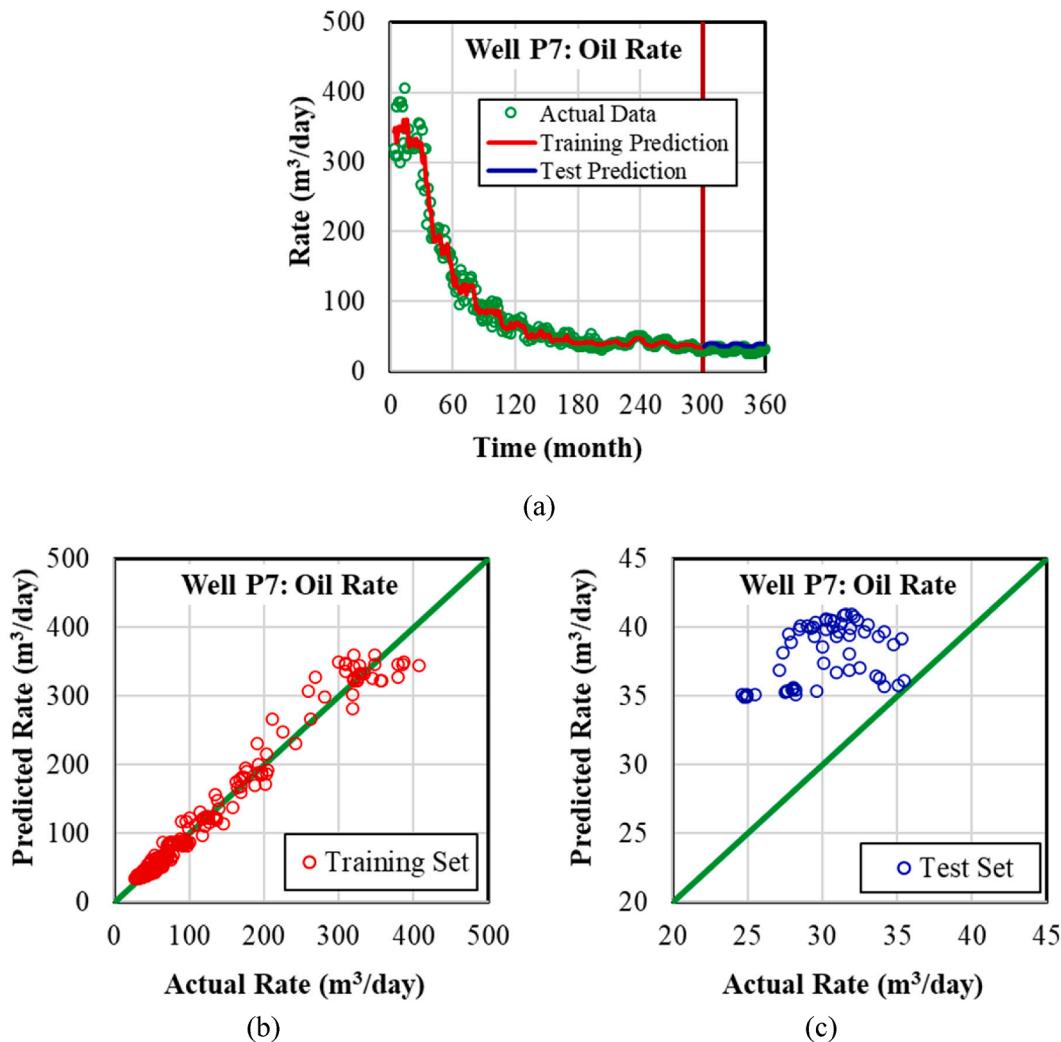


Fig. 14. Comparison of predicted oil production rates and simulated oil production rates from well P7 (a) time series production rates for the training and test sets (b) cross plot for the training set (b) cross plot for the test set.

The water production rate from well P2 exhibits a similar nature except that the maximum and minimum values continue increasing with time, likely because of the contribution of injected water. The LSTM model closely predicts the water production rates, as shown in Fig. 8a. The production rates also fall near the diagonal line (simulated = predicted) in the cross-plot Fig. 8 b and c.

Although high accuracy is achieved during training, the performance of an LSTM model is assessed during the multi-step-ahead blind prediction for the test set. Only test results (301–360 months) from Fig. 6a, Fig. 7a and Fig. 8a are expanded in Fig. 9. The out-of-sample prediction from ARIMA models is also added to these figures for comparison.

The blind out-of-sample predicted values by LSTM models for the test set are consistent with the simulated production rates. The LSTM model could capture the oscillating patterns in production rate, whereas ARIMA failed to match the production rate for out-of-sample prediction. The ARIMA model is effective apparently only for single-step prediction (Fig. 9). The first blind predictions, i.e., oil, gas, and water production rates at the 301st month by ARMA, are close to the simulated values. However, production rates start deviating from the 302nd month

because of the out-of-sample prediction, i.e., the predicted values instead of the simulated test values of the previous time steps are used to re-train the ARANA model as part of the method, suggesting the main advantage of LSTM models. The LSTM consistently predicts production rates with reasonable accuracy (i.e., low errors) and captures the oscillating nature of production rates in the test period.

Simulated and predicted oil, gas, and water production rates from wells P3 are compared in Fig. 10, Fig. 11 and Fig. 12. Similar to well P2, the oscillating natures of production rates from well P3 are apparent in the LSTM model results.

The expanded sections of the test period, along with results from ARIMA models from well P3, are shown in Fig. 13.

ARIMA models always predict the first production rates of the test set (301st month) close to the simulated production rates. However, the models then predict a monotonic increase or decrease in production rates with time for multi-steps ahead out-of-sample (i.e., blind).

Similar results and performances of LSTM and ARIMA models are observed for well P7, as shown in Figs. 14–17.

In some cases (for example Fig. 6c, Fig. 7c, Fig. 10c, and Fig. 14c),

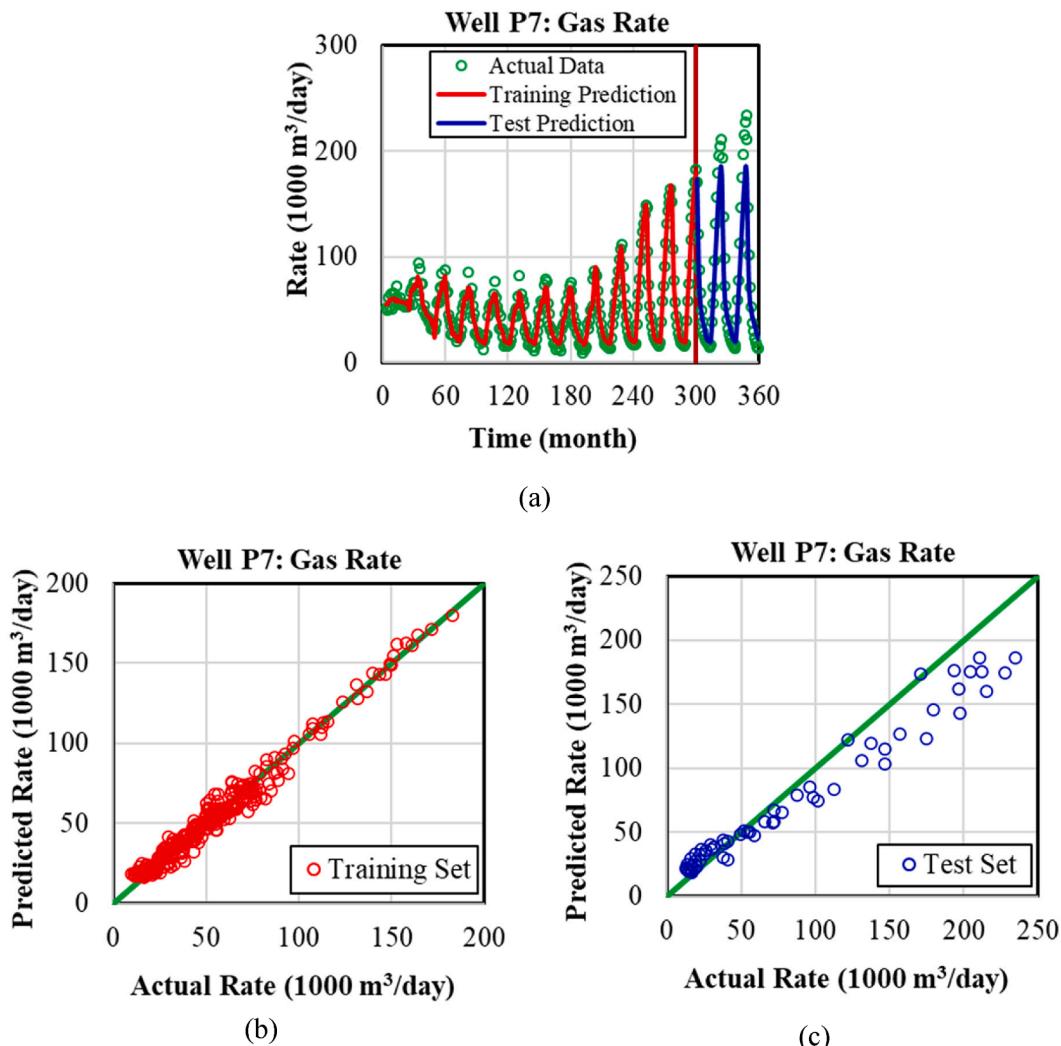


Fig. 15. Comparison of predicted gas production rates and simulated gas production rates from well P7 (a) time series production rates for the training and test sets (b) cross plot for the training set (c) cross plot for the test set.

more significant deviations are observed in the prediction for the test set. Despite a good match with oscillating trends in production rates, errors are greater due to several reasons. A small-time lag between the predicted and the simulated production rates is exhibited, causing larger errors between any given two points at different time alignments. Injected water or carbon dioxide (alternating annually) pushes oil towards producers at different production rates because mobility and sweeping efficiency are different. Additionally, the injection response takes some time to reach the producer (pressure migration rate is a function of hydraulic diffusivity), and hence such is reflected in the production rates that exhibit lag. Therefore, the effects of the injection are not “felt” by the producer immediately. However, the ML model predicts production rates using the bottom hole pressures of injectors at the same time step. Model performance might be better when the responses are faster in cases where the reservoir has higher permeability and porosity and is less heterogeneous.

The main reason for the success of these LSTM models in predicting the oil, gas, and water production rates and their oscillating natures is the selection of input parameters for the models. In this study, one

critical input is the historical production rate itself which manifests the dynamic changes in the reservoir pressure and fluid saturations. However, historical production rates alone are not sufficient to predict the dynamic and complex nature of production rates. Therefore, the BHPs of producers and injectors are also included, to help capture the oscillating nature of the production rate.

4.2. Model evaluation and uncertainty

There are many quantitative metrics to evaluate the model fitness, including mean squared error (MSE), root mean squared error (RMSE), normalized root mean squared error (NRMSE), mean absolute error (MAE), relative error (RE), and mean absolute percentage error (MAPE). The Nash–Sutcliffe model efficiency coefficient (NSE) (Nash and Sutcliffe, 1970) is often used to assess efficacy of time series prediction. The value of NSE close to unity suggests good model performance. In some cases, NSE values of negative are also observed, and thus a normalized Nash–Sutcliffe model efficiency coefficient (NNSE) is suggested to keep NSE values within the range of zero to one. Relative error (RE), mean

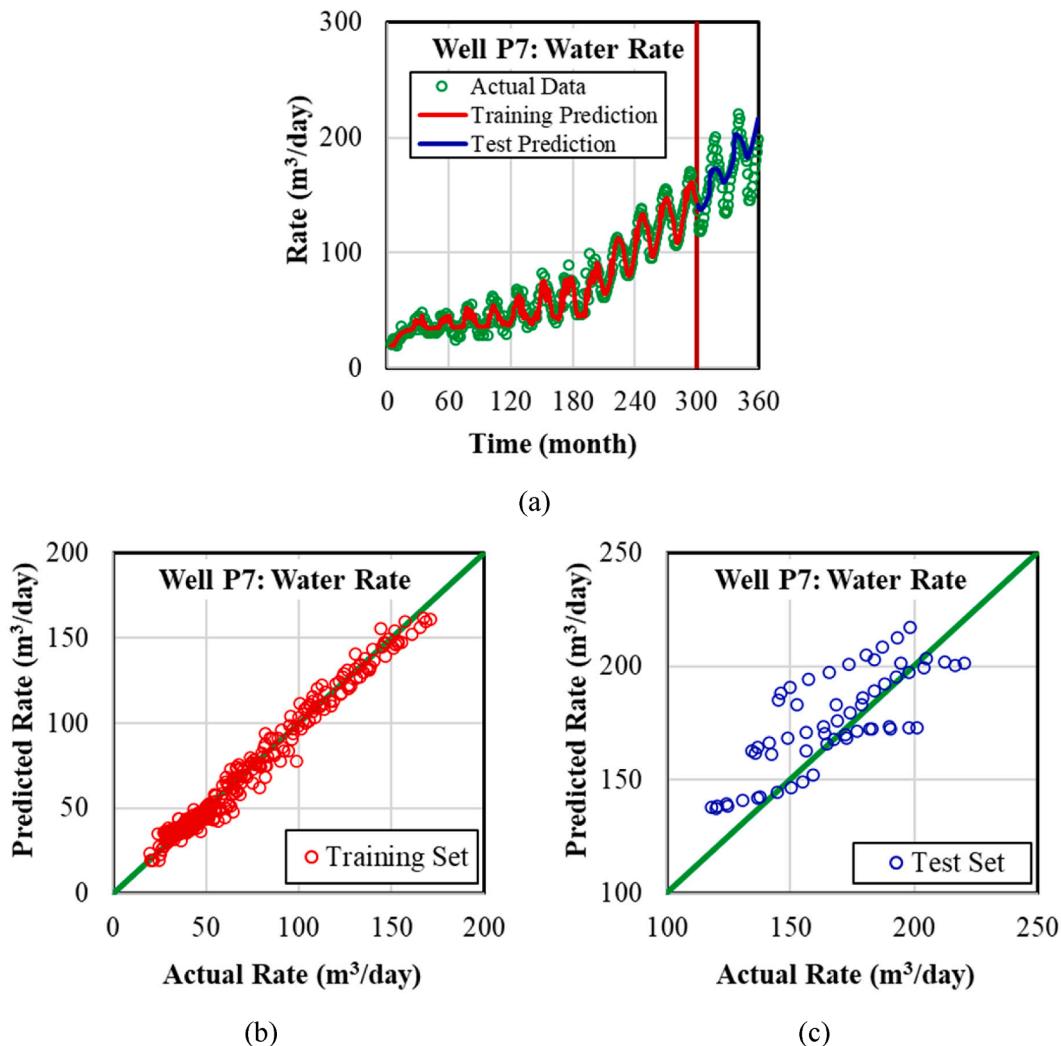


Fig. 16. Comparison of predicted water production rates and simulated water production rates from well P7 (a) time series production rates for the training and test sets (b) cross plot for the training set (b) cross plot for the test set.

absolute percentage error (MAPE) and NNSE are selected for assessing efficacy in this study. The relative error provides important information about the pointwise error and its variation with time, which assists in identifying regions of the higher and lower errors. However, MAPE and NNSE provide a single-valued error. The mean absolute percentage error (MAPE) is calculated as:

$$\text{MAPE} = \frac{1}{N} \sum_{i=1}^N \left| \frac{y_{i,\text{obj}} - y_{i,\text{model}}}{y_{i,\text{obj}}} \right| \times 100 \quad (4)$$

The Nash–Sutcliffe model efficiency coefficient (NSE) is calculated as:

$$\text{NSE} = 1 - \frac{\sum_{i=1}^N (y_{i,\text{obj}} - y_{i,\text{model}})^2}{\sum_{i=1}^N (y_{i,\text{obj}} - \bar{y}_{\text{obj}})^2} \quad \text{where } \bar{y}_{\text{obj}} = \frac{1}{N} \sum_{i=1}^N y_{i,\text{obj}} \quad (5)$$

The normalized NSE is calculated as:

$$\text{NNSE} = \frac{1}{2 - \text{NSE}} \quad (6)$$

Absolute values of errors are used to avoid any nullification (offsetting) of positive and negative errors. A low value of MAPE is a quantitative indication of a good match between simulated values (observed values) and predicted values.

Calculated MAPE values for predicted oil, gas, and water production rates from all 23 producers in SACROC are shown in Fig. 18. The MAPEs for training and test sets are separately plotted. Except for a few wells, MAPE values are under 10% for oil production rate predictions in the training set, whereas errors are higher (about 30%) in the test set. The resulting MAPE values for the test set for a few wells could not be calculated because wells were shut-in before 300 months, and no production was possible during the test period.

NNSEs for predicted oil, gas, and water production rates from all 23 producers in SACROC are shown in Fig. 19.

The values of NNSE for the training data set are close to one for oil, gas and water production rates from all wells, suggesting a good fit of the stacked LSTM models during the training. However, NNSE values for oil production rates vary from near zero to 0.8 for the test set. The NNSE values are relatively higher for gas and water production rates.

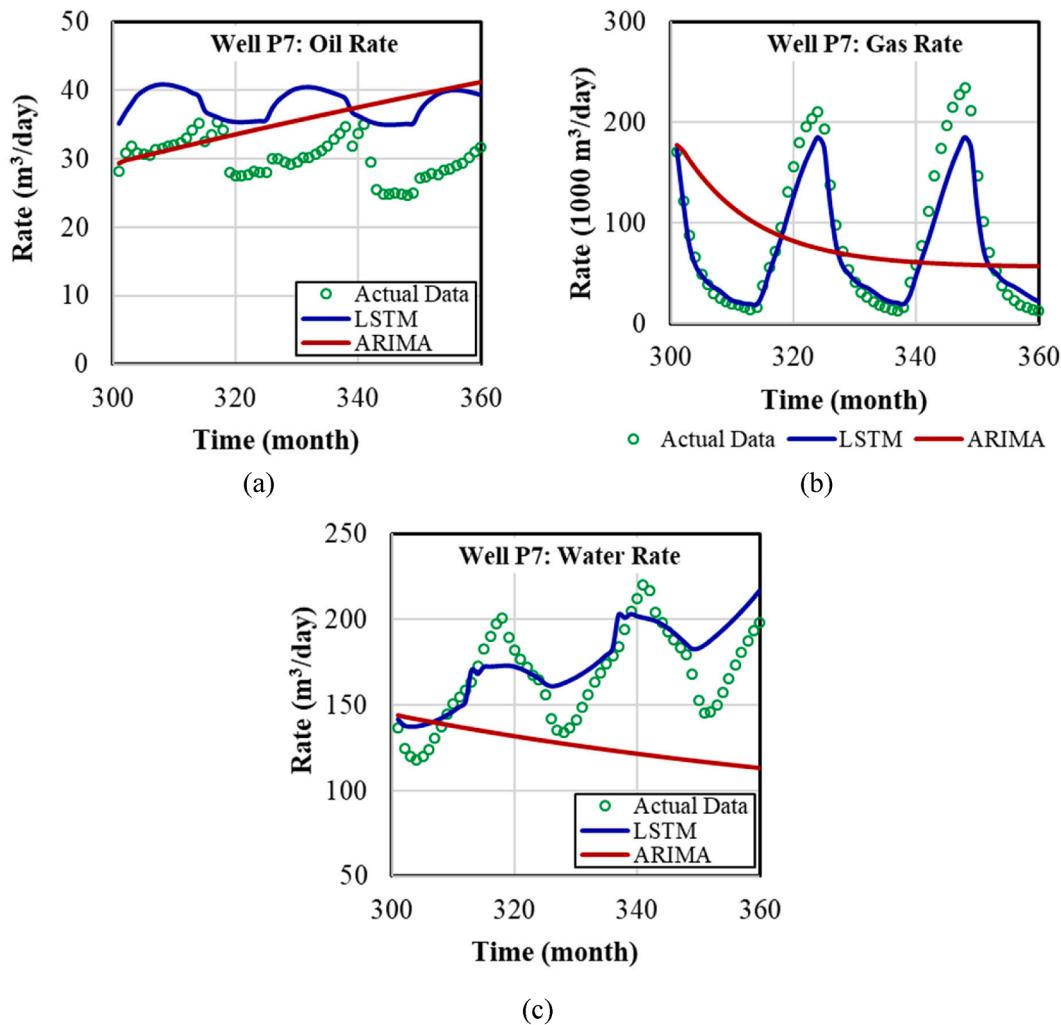


Fig. 17. Comparison of simulated production rates from well P7 with predicted production rates by LSTM and ARIMA models (a) oil production rate (b) gas production rate (c) water production rate.

Note that despite relatively higher errors (MAPE around 30% and NNSE from zero to unity) in predicted production rates, the trends in the production rates are correctly captured by the LSTM model. This may possibly be reduced by adding more geologic parameters in the model as input for a complex field like SACROC. As suggested by the highlighted MAPE values for wells P2, P3, and P7, the LSTM models may be effective for the prediction of oil, gas, and water production rates, at least for fields as complex as SACROC.

The uncertainty of a machine learning model such as LSTM may be quantified by the variations of errors in the test set (blind) when the model is trained several times. For sake of such assessment, the LSTM models for oil, gas, and water production rates from wells P2, P3, and P7 are trained repeatedly 110 times with the same number of neurons, dropout production rate, optimizer, and epoch. All MAPE values in test sets from these runs are stored and statistically analyzed. Results are summarized in Box-Whisker plots (Fig. 20).

The major components of this uncertainty are the median and mean values of MAPE. Except for the oil production rate from well P2, the uncertainties of the LSTM models are within an acceptable range (low error values). The higher uncertainty in the LSTM oil production rate

from P2 may be caused by the frequent overfitting of the model. The overfitting during training ensures minimal errors in the cost of bad prediction for the test set, but this may be overcome by adjusting the number of neurons and dropout production rate, or other ML parameters in the LSTM model. However, these adjustments were deliberately not implemented, to maintain the numbers of model parameters the same for all wells. For practical purposes, a custom LSTM model with an optimized number of neurons, dropout production rate, etc., for individual well may be developed.

It should be noted that LSTM has some inherent limitations. Typically, LSTM requires more time and more memory to train due to the higher number of parameters in an LSTM. Therefore, depending on data size, LSTM models can be computationally expensive and prone to overfitting. Another disadvantage is that the interpretation of models to understand their behavior can be challenging.

5. Conclusions

Prediction of multiphase production rates (oil, gas, and water) from a complex reservoir like SACROC is a challenging task because of complex

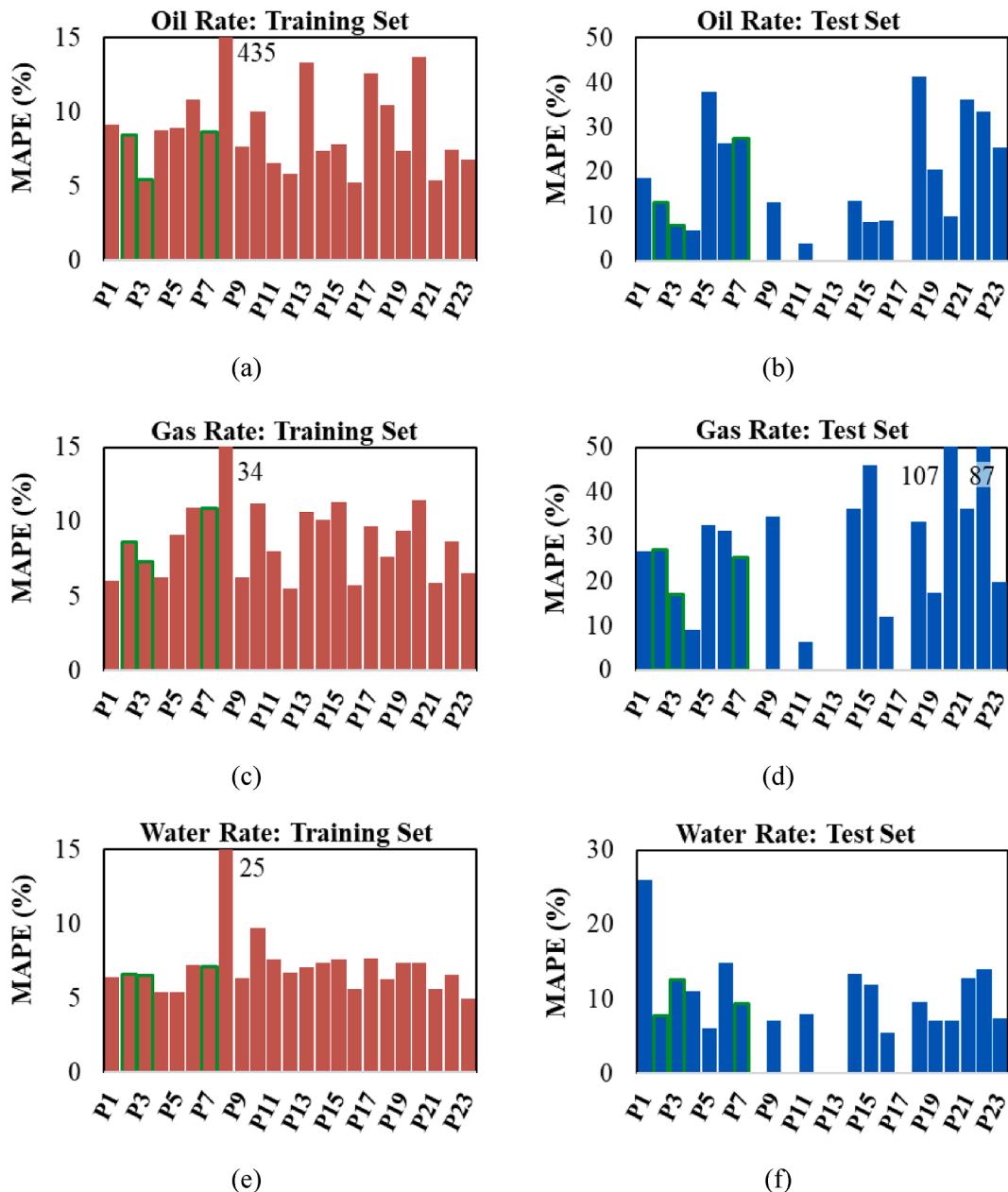


Fig. 18. Mean absolute percentage error (MAPE) for all 23 producers, (P2, P3, and P7 are highlighted) (a) oil production rate - training set (a) oil production rate - test set (c) water production rate - training set (d) water production rate - test set (e) water production rate - training set (f) water production rate - test set.

geological structure, heterogeneity in permeability and porosity, and sparse oil composition data. Consequently, production rates exhibit complicated patterns that are difficult, if not impossible, to predict by any traditional time series models (such as ARIMA). An alternative approach is a stacked long short-term memory (LSTM) machine learning model. Key conclusions of this study include:

- (1) The success of LSTM models for predicting the oil, gas, and water production rates is due to the selection of input parameters and the quality of data.
- (2) The primary factors that influence resulting LSTM forecasts include the BHP of a producer, BHPs of surrounding injectors,

and historical production rates (all of which are input data for LSTM models).

- (3) Unusual and/or oscillating trends in production rates may be captured with a stacked LSTM.
- (4) The LSTM modeling approach is a promising ML algorithm for time-series prediction such as oil, gas, and water production rates from complex fields (like SACROC) with the enhanced oil recovery operations.
- (5) Such LSTM models can be improved by preparing more complex network architecture (better heterogeneity information) and optimizing the number of neurons.

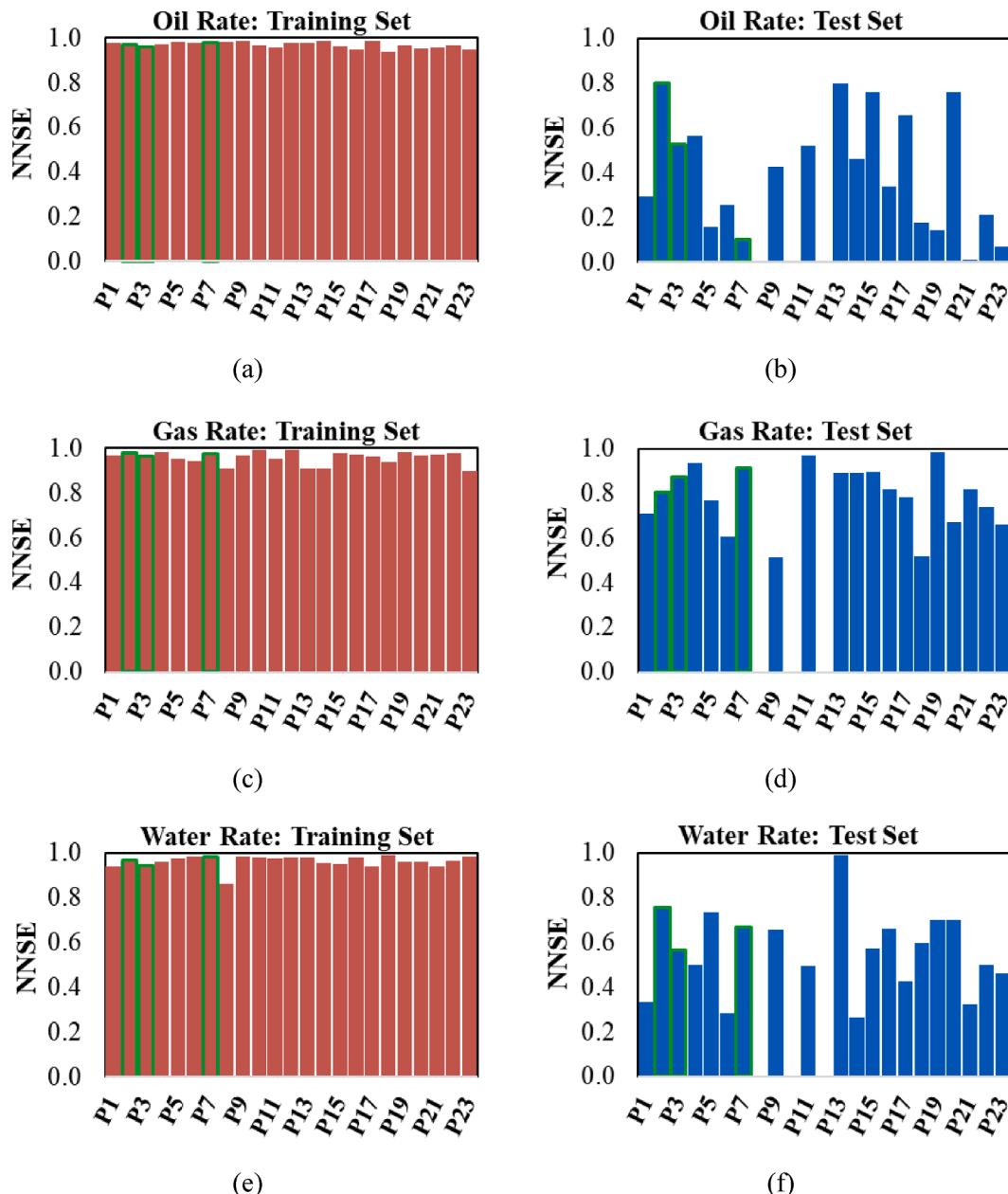


Fig. 19. Normalized Nash–Sutcliffe model efficiency coefficient (NNSE) for all 23 producers, (P2, P3, and P7 are highlighted) (a) oil production rate - training set (a) oil production rate - test set (c) water production rate - training set (d) water production rate - test set (e) water production rate - training set (f) water production rate - test set.

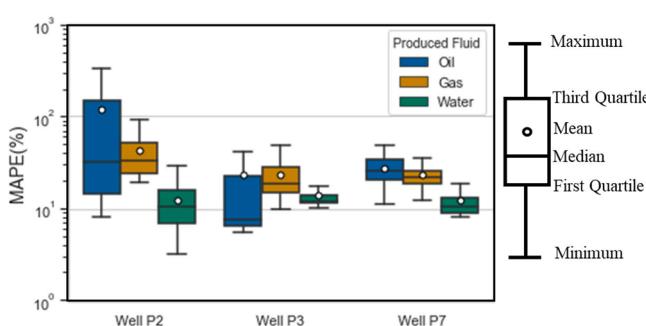


Fig. 20. Model uncertainty analysis using Box-Whisker plots of MAPE of oil, gas, and water production rates for testing data (a) producer P2 (b) producer P3 (c) producer P7.

CRediT authorship contribution statement

Palash Panja: Conceptualization, Methodology, Data curation, Formal analysis, Writing – original draft. **Wei Jia:** Data curation, Software, Investigation, Writing – review & editing. **Brian McPherson:** Supervision, Resources, Investigation, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was performed in support of the National Energy

Technology Laboratory's ongoing research under the RSS contract 89243318CFE0000003. The authors also acknowledge the academic license of CMG products from Computer Modeling Group, Calgary, Canada.

Disclaimer

This work was funded by the Department of Energy, National Energy Technology Laboratory, an agency of the United States Government, through a support contract with Leidos Research Support Team (LRST). Neither the United States Government nor any agency thereof, nor any

of their employees, nor LRST, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Appendix A. Error analysis

Relative error calculation is shown in Equation (A.1).

$$RE = \frac{Y_{\text{obs},i} - Y_{\text{model},i}}{Y_{\text{obs},i}} \quad (\text{A1})$$

Y_{obs} and Y_{model} are the simulated value and predicted value from the model, respectively, n is the number of data points (different for training and test sets) in the calculation.

Absolute values of errors sometimes do not provide sufficient information about how large the errors are. To avoid this pitfall, the relative error is measured at a particular point of a curve, as shown in Equation (A.2).

$$RE(\%) = \frac{y_{i,\text{obj}} - y_{i,\text{model}}}{y_{i,\text{obj}}} \times 100 \quad (\text{A2})$$

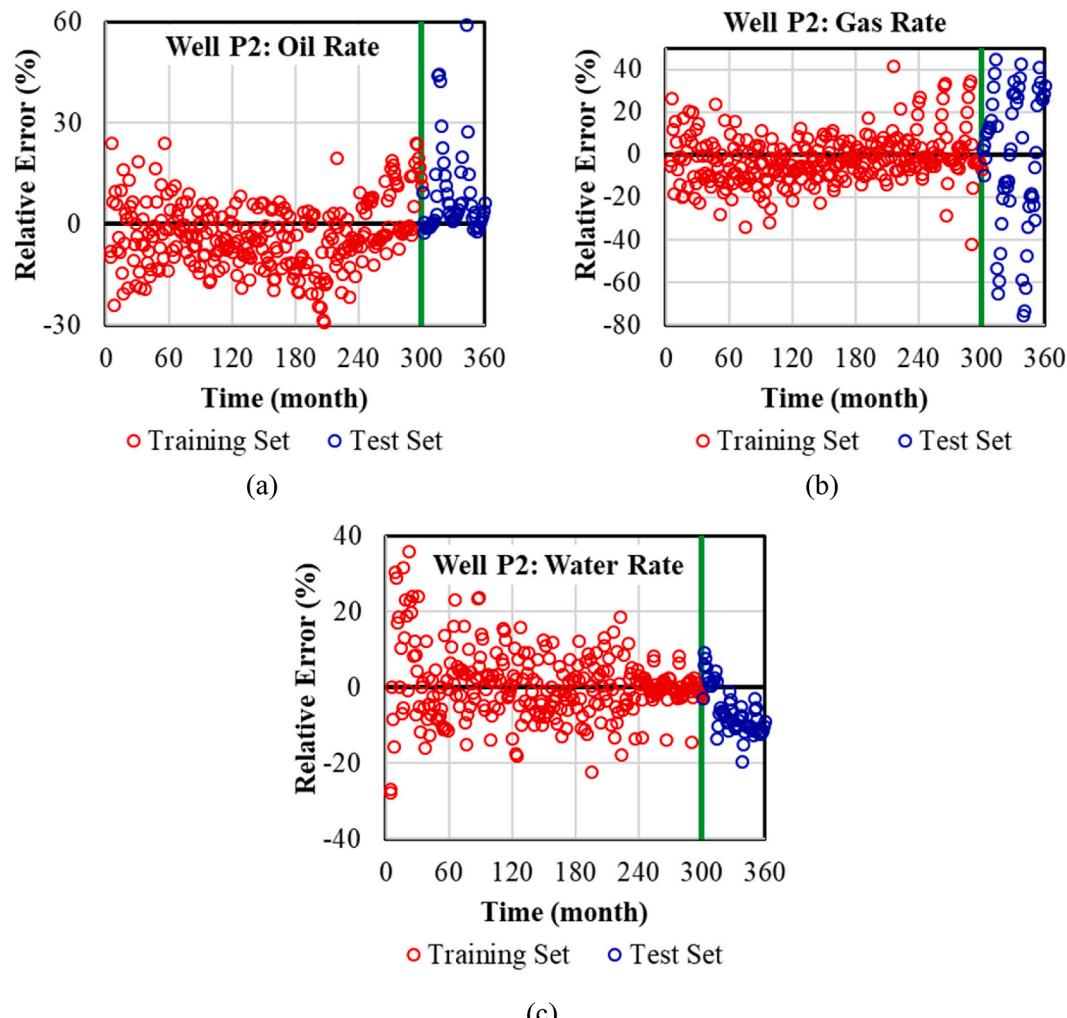


Fig. A1. Relative errors in predicted production rates from well P2 (a) oil production rate (b) gas production rate (c) water production rate.

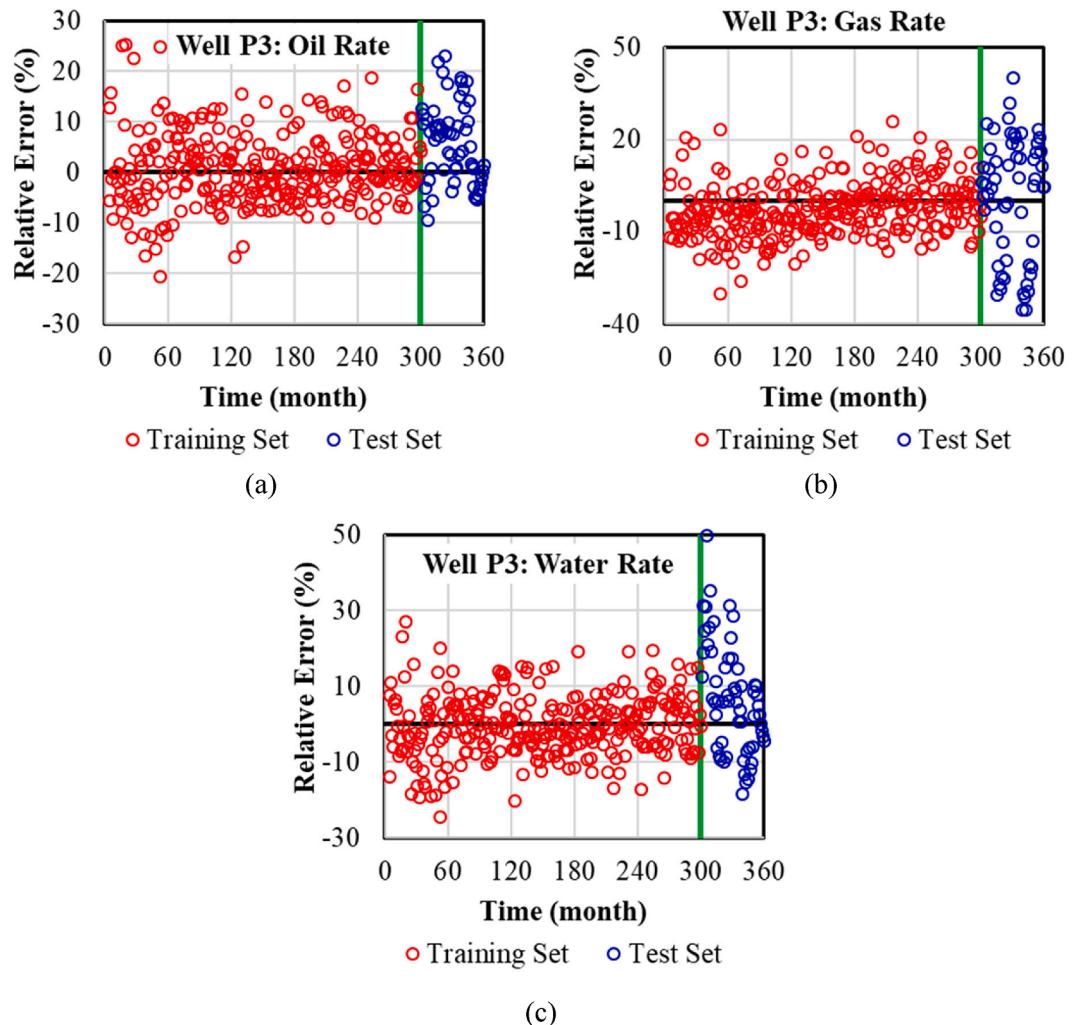


Fig. A2. Relative errors in predicted production rates from well P3 (a) oil production rate (b) gas production rate (c) water production rate.

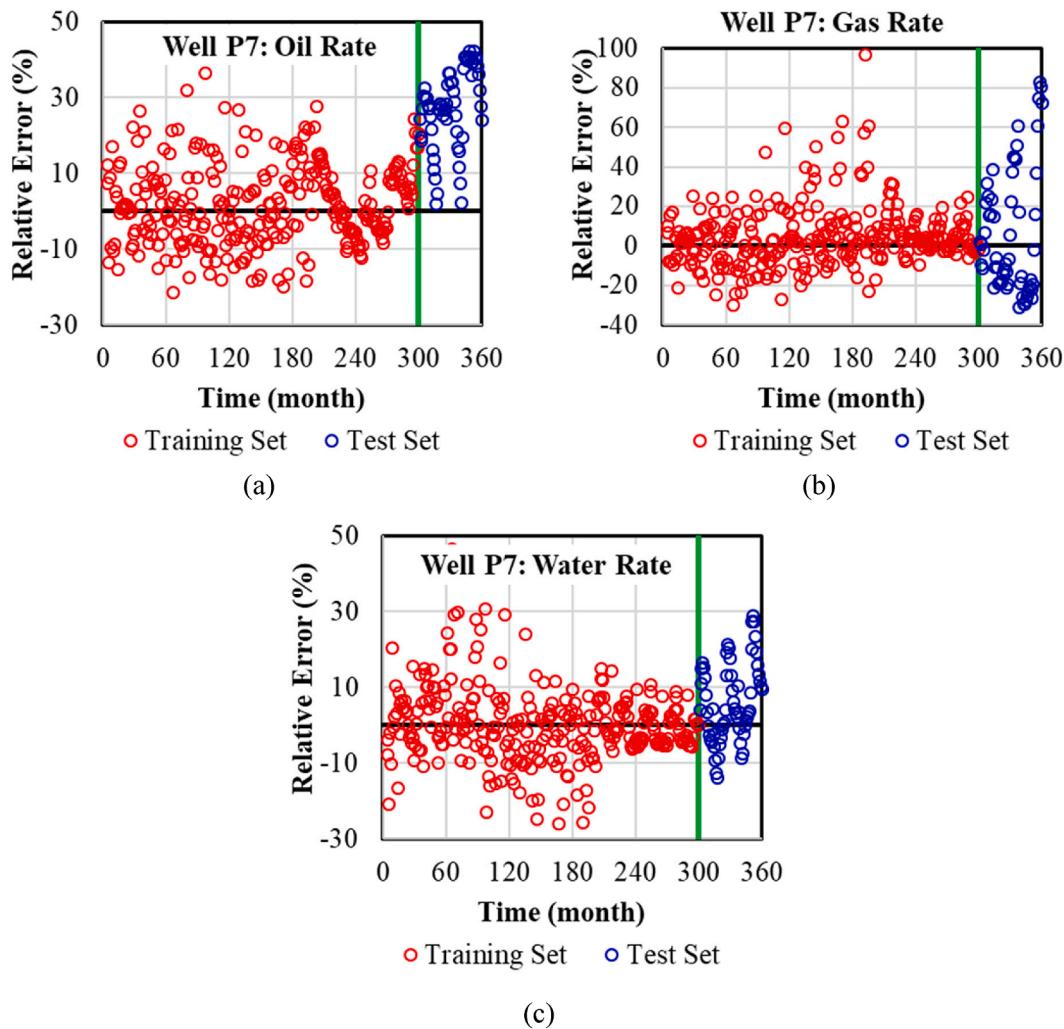


Fig. A3. Relative errors in predicted production rates from well P7 (a) oil production rate (b) gas production rate (c) water production rate.

Relative errors show the error in each point of a curve with respect to that particular point.
Appendix B. Brief theory of LSTM

Long Short-Term Memory (LSTM) for time series prediction is briefly discussed here. A correlation between temporal input data and parameters and the basic structure of LSTM is shown in Fig. B.1.

Each time series data point is the direct input to an LSTM block, and the output of that block are fed to the next LSTM block. Therefore, an LSTM block has three input data; one input is from the current time step, and two input data from the previous time step, namely cell state (C_{t-1}) and hidden output (h_{t-1}). The cell state is responsible for carrying the memories of all previous time steps. There are three gates; input gate, forget gate, and output gate, as shown in Fig. B.1. Equations B.1-B.6.

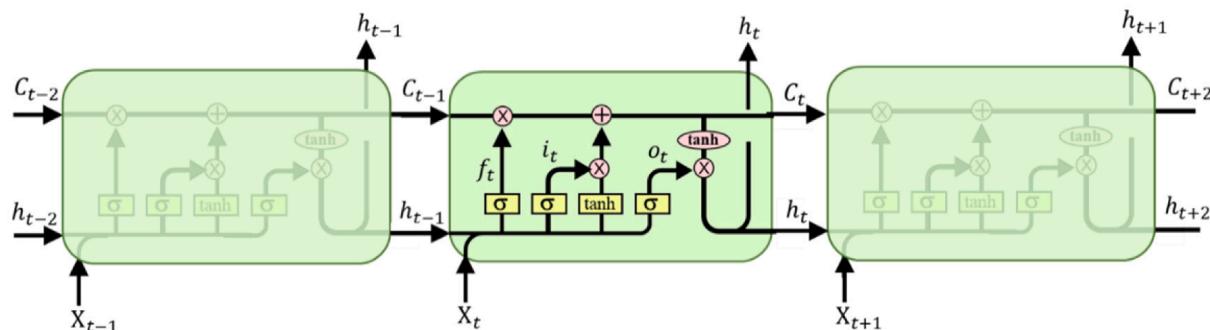


Fig. B1. A basic structure of the LSTM model and the temporal correlations parameters The Basic operations in a single LSTM module. Modified from (Networks, 2020).

- Input Gate

$$g_t = \tanh(W_{xg}^* X_t + W_{hg}^* h_{t-1} + b_g) \quad (B1)$$

$$i_t = \sigma(W_{xi}^* X_t + W_{hi}^* h_{t-1} + b_i) \quad (B2)$$

- Forget Gate

$$f_t = \sigma(W_{xf}^* X_t + W_{hf}^* h_{t-1} + b_f) \quad (B3)$$

- Output Gate

$$C_t = f_t \otimes C_{t-1} + i_t \otimes g_t \quad (B4)$$

$$O_t = \sigma(W_{xo}^* X_t + W_{ho}^* h_{t-1} + W_{co}^* C_t + b_o) \quad (B5)$$

$$h_t = O_t \otimes \tanh(C_t) \quad (B6)$$

All the output, i.e., h from each time step, are compared with the simulated values by calculating the mean squared error or other loss function. Finally, this loss function is minimized during the optimization of all the weights (W), and bias terms (b).

References

- Mishra, S., Datta-Gupta, A., Applied Statistical Modeling and Data Analytics: A Practical Guide for the Petroleum Geosciences. 1st Edition ed. 2017: Elsevier Inc.
- Panja, P., et al. (2018). Application of artificial intelligence to forecast hydrocarbon production from shales. *Petroleum*, 4(1), 75–89.
- De Gooijer, J. G., & Hyndman, R. J. (2006). 25 years of time series forecasting. *International Journal of Forecasting*, 22(3), 443–473.
- Muth, J. F. (1960). Optimal Properties of Exponentially Weighted Forecasts. *Journal of the American Statistical Association*, 55(290), 299–306.
- Gardner, E. S. (1985). Exponential smoothing: The state of the art. *Journal of Forecasting*, 4(1), 1–28.
- Snyder, R. D. (1985). Recursive estimation of dynamic linear models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 47(2), 272–276.
- Yule, G. U. (1927). On a method of investigating periodicities in disturbed series, with special reference to Wolfer's sunspot numbers. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 226, 267–298.
- Box, G.E.P., Jenkins, G.M., Time series analysis: forecasting and control. 1970: Holden-Day.
- Dagum, E. B. (1982). Revisions of time varying seasonal filters. *Journal of Forecasting*, 1 (2), 173–187.
- Huyot, G., et al. (1986). Analysis of revisions in the seasonal adjustment of data using X-11-Arima model-based filters. *International Journal of Forecasting*, 2(2), 217–229.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1), 35–45.
- Schweppke, F. (1965). Evaluation of likelihood functions for Gaussian signals. *IEEE Trans. Inf. Theory*, 11(1), 61–70.
- Shumway, R. H., & Stoffer, D. S. (1982). An approach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis*, 3(4), 253–264.
- Wiener, N. (1958). *Nonlinear problems in random theory*. [Cambridge]: Technology Press of Massachusetts Institute of Technology.
- Volterra, V., Theory of functionals and of integral and integro-differential equations. 1930: Blackie & Son Limited. 226.
- Ray, B. K. (1993). Modeling long-memory processes for optimal long-range prediction. *Journal of Time Series Analysis*, 14(5), 511–525.
- Ray, B. K. (1993). Long-range forecasting of IBM product revenues using a seasonal fractionally differenced ARMA model. *International Journal of Forecasting*, 9(2), 255–269.
- Bollerslev, T., Engle, R. F., & Nelson, D. B. (1994). Chapter 49 Arch models. In *Handbook of Econometrics* (pp. 2959–3038). Elsevier.
- Taylor, S. J. (1987). Forecasting the volatility of currency exchange rates. *International Journal of Forecasting*, 3(1), 159–170.
- Engle, R. F. (1982). Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation. *Econometrica*, 50(4), 987–1007.
- Willemain, T. R., et al. (1994). Forecasting intermittent demand in manufacturing: A comparative evaluation of Croston's method. *International Journal of Forecasting*, 10 (4), 529–538.
- Croston, J.D., Forecasting and Stock Control for Intermittent Demands. Operational Research Quarterly (1970–1977), 1972. 23(3): p. 289–303.
- Arps, J.J., Analysis of Decline Curves. Transactions of the AIME, 1945. SPE-945228-G.
- Ilk, D., et al., Exponential vs. Hyperbolic Decline in Tight Gas Sands: Understanding the Origin and Implications for Reserve Estimates Using Arps' Decline Curves, in SPE Annual Technical Conference and Exhibition. 2008, Society of Petroleum Engineers: Denver, Colorado, USA.
- Valko, P.P., Assigning value to stimulation in the Barnett Shale: a simultaneous analysis of 7000 plus production histories and well completion records, in SPE Hydraulic Fracturing Technology Conference. 2009, Society of Petroleum Engineers: The Woodlands, Texas. p. 19.
- Clark, A.J., L.W. Lake, and T.W. Patzek, Production Forecasting with Logistic Growth Models, in SPE Annual Technical Conference and Exhibition. 2011, Society of Petroleum Engineers: Denver, Colorado, USA.
- Duong, A.N., An Unconventional Rate Decline Approach for Tight and Fracture-Dominated Gas Wells, in Canadian Unconventional Resources and International Petroleum Conference. 2010, Society of Petroleum Engineers: Calgary, Alberta, Canada.
- Vaswani, A., et al., Attention is All you Need, in 31st Conference on Neural Information Processing Systems (NIPS 2017), I. Guyon, et al., Editors. 2017: Long Beach, CA, USA.
- Lea, C., et al. Temporal Convolutional Networks: A Unified Approach to Action Segmentation, in Computer Vision – ECCV 2016 Workshops. 2016. Cham: Springer International Publishing.
- Yan, R., et al. (2021). Multi-hour and multi-site air quality index forecasting in Beijing using CNN, LSTM, CNN-LSTM, and spatiotemporal clustering. *Expert Systems with Applications*, 169.
- Masulli, F. and L. Studer, Neuro-fuzzy system for chaotic time series forecasting, Proc. SPIE 3165, in Applications of Soft Computing. 1997: San Diego, CA, United States. p. 204–215.
- Wei, C.-C., Chen, T.-T., Lee, S.-J., k-NN Based Neuro-fuzzy System for Time Series Prediction, in 2013 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing. 2013: Honolulu, HI, USA. p. 569–574.
- Peng, H.-W., et al. (2015). Time series forecasting with a neuro-fuzzy modeling scheme. *Applied Soft Computing*, 32, 481–493.
- Song, X., et al. (2020). Time-series well performance prediction based on Long Short-Term Memory (LSTM) neural network model. *Journal of Petroleum Science and Engineering*, 186, Article 106682.
- Zhang, J., et al. (2018). Developing a Long Short-Term Memory (LSTM) based model for predicting water table depth in agricultural areas. *Journal of Hydrology*, 561, 918–929.
- Qin, Y., et al. (2019). Hybrid forecasting model based on long short term memory network and deep learning neural network for wind signal. *Applied Energy*, 236, 262–272.
- Han, S., et al. (2019). Mid-to-long term wind and photovoltaic power generation prediction based on copula function and long short term memory network. *Applied Energy*, 239, 181–191.
- Li, Y., & Cao, H. (2018). Prediction for Tourism Flow based on LSTM Neural Network. *Procedia Computer Science*, 129, 277–283.
- Tong, W., et al. (2019). Deep learning PM2.5 concentrations with bidirectional LSTM RNN. *Air Quality, Atmosphere & Health*, 12(4), 411–423.
- Kara, A. (2021). Multi-step influenza outbreak forecasting using deep LSTM network and genetic algorithm. *Expert Systems with Applications*, 180.
- Computer Modeling Group: GEM, Compositional & Unconventional Simulator, <https://www.cmg.ca/gem>. [cited 2020].
- Han, W. S., et al. (2010). Evaluation of trapping mechanisms in geologic CO₂ sequestration: Case study of SACROC northern platform, a 35-year CO₂ injection site. *American Journal of Science*, 310(4), 282.
- Jia, W., et al. (2018). Uncertainty quantification of CO₂ storage using Bayesian model averaging and polynomial chaos expansion. *International Journal of Greenhouse Gas Control*, 71, 104–115.
- Jia, W., et al. (2016). Probabilistic analysis of CO₂ storage mechanisms in a CO₂-EOR field using polynomial chaos expansion. *International Journal of Greenhouse Gas Control*, 51, 218–229.
- EDX: NETL's Energy Data eXchange, <https://edx.netl.doe.gov/edx>, accessed in 2020. [cited 2020; Available from: <https://edx.netl.doe.gov/>.

- Kumari, B., & Swarnkar, T. (2011). Filter versus wrapper feature subset selection in large dimensionality micro array: A review. *International Journal of Computer Science and Information Technologies*, 2(3), 1048–1053.
- Guyon, I., et al., Feature Extraction : Foundations and Applications. Studies in Fuzziness and Soft Computing. 2006, Berlin, Heidelberg, Germany: Springer.
- Khandelwal, R., Feature Selection : Identifying the best input features. 2019.
- Galelli, S., et al. (2014). An evaluation framework for input variable selection algorithms for environmental data-driven models. *Environmental Modelling & Software*, 62, 33–51.
- Fernando, T. M. K. G., Maier, H. R., & Dandy, G. C. (2009). Selection of input variables for data driven models: An average shifted histogram partial mutual information estimator approach. *Journal of Hydrology*, 367(3–4), 165–176.
- Spyder IDE, <https://www.spyder-ide.org/>, accessed in 2020. [cited 2020; Available from: <https://www.spyder-ide.org/>.
- Anaconda Navigator, <https://docs.anaconda.com/anaconda/navigator/>, accessed in 2020. [cited 2020; Available from: <https://anaconda.org/>.
- Keras, <https://keras.io/>, accessed in 2020. [cited 2020; Available from: <https://keras.io/>. TensorFlow, <https://www.tensorflow.org/>, accessed in 2020. [cited 2020; Available from: <https://www.tensorflow.org/>.
- Dinkelbach, H.Ü., et al. (2012). Comparison of GPU- and CPU-implementations of mean-firing rate neural networks on parallel hardware. *Network: Computation in Neural Systems*, 23(4), 212–236.
- Brownlee, J., How to Reduce Overfitting With Dropout Regularization in Keras. 2018.
- Kingma, D.P. and J. Ba Adam: A Method for Stochastic Optimization. 2014. doi: <https://doi.org/10.48550/arXiv.1412.6980>.
- Afaq, S., & Rao, S. (2020). Significance of epochs on training a neural network. *International Journal of Scientific & Technology Research*, 9, 485–488.
- Nash, J. E., & Sutcliffe, J. V. (1970). River flow forecasting through conceptual models part 1 — A discussion of principles. *Journal of Hydrology*, 10(3), 282–290.
- Understanding LSTM Networks, <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>, accessed in 2020. 2015 [cited 2020; Available from: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>.