

A Gas Well Production Prediction Model Based on Time-convolutional Neural Network

Long Zhang

School of Petroleum Engineering, Xi'an Shiyou University, Xi'an, Shaanxi 710065, China

Abstract: This paper explores gas well production prediction, a crucial aspect of gas field development planning and analysis. Traditional methods like decline curve analysis and numerical simulation have limitations due to their reliance on simple mathematical models. In contrast, machine learning offers a more flexible approach. The study applies a Temporal Convolutional Network (TCN) model to predict production for two gas wells. The model's performance was evaluated using MSE, MAE, RMSE, and R², with results showing the TCN method achieving better accuracy. The TCN model presents a promising new approach for gas well production forecasting.

Keywords: Production Forecasting; Machine Learning; TCN.

1. Introduction

Gas well production is a key objective in gas field development, and accurate prediction of production changes is essential for designing development plans and conducting dynamic analyses. Traditional prediction methods, such as decline curve analysis and numerical simulation, each have limitations [1]. Decline curve analysis, while requiring minimal data and offering fast calculations, relies on overly simplified assumptions, making it less effective for wells with complex production dynamics [2]. Numerical simulation, on the other hand, demands extensive geological, petrophysical, and reservoir data, resulting in high workloads and time-consuming processes [3]. In contrast, Temporal Convolutional Networks (TCNs), a deep learning model designed for time series data, can better capture temporal patterns and enhance prediction accuracy in time-dependent data like gas well production [4]. This makes TCNs a valuable and practical approach for improving production forecasting.

2. Temporal Convolutional Neural Network

2.1. Basic Component

2.1.1. Causal Convolution

Causal convolution in Temporal Convolutional Networks (TCNs) ensures that the model processes time series data in a causal order. This means that when predicting the output at any given point in time, the model relies solely on past information, without access to future data. By enforcing this constraint, TCNs prevent data leakage and ensure that predictions are based strictly on historical data, making them particularly well-suited for time series forecasting tasks where preserving the temporal sequence is essential [5].

In Temporal Convolutional Networks (TCNs), the convolution operation at time step t is performed only on data points preceding t , ensuring that the result at t is not influenced by future information. For an input time series $[x_1, x_2, x_3, \dots, x_{t-1}, x_t]$ and a convolution kernel $f = [f_1, f_2, \dots, f_{k-1}, f_k]$ of size kt , the convolution operation at time t is defined as:

$$f(x_t) = \sum_{i=l}^k f_i x_{t-k+i} \quad (1)$$

This operation ensures that only past values up to t are considered in the computation. Similar to standard convolutional neural networks, TCNs employ padding (by adding zeros) to maintain the input and output sequence lengths, ensuring that the length of the output sequence matches that of the input.

2.1.2. Dilation Convolution

Expansion convolution (or dilated convolution) enables Temporal Convolutional Networks (TCNs) to broaden their receptive field without increasing the network's depth by introducing a dilation factor into the convolutional kernel. This mechanism allows TCNs to capture long-range temporal dependencies, which is critical for recognizing complex patterns in time series data [6].

In a traditional convolutional network with n layers and a kernel size of k , the receptive field y is calculated as: $y = n * (k - 1) + 1$.

This shows a linear relationship between the number of layers and the size of the receptive field. However, with expansion convolution, the relationship becomes exponential, allowing the network to increase the receptive field size more efficiently, even with fewer layers. This not only improves the model's ability to capture long-range dependencies but also simplifies optimization and accelerates convergence.

When expansion convolution is applied, the receptive field size y changes, and the operation becomes:

$$f(x_t) = \sum_{i=1}^k f_i x_{t-d_n(k-i)} \quad (2)$$

where d_n is the dilation factor for the n -th layer, calculated as:

$$d_n = 2^{n-1} \quad (3)$$

This exponential growth in the dilation factor allows the model to effectively handle long sequences and complex dependencies, all while maintaining a relatively shallow network architecture.

2.1.3. Residual Connections

Shortcut connections in the residual block allow the input to bypass the convolutional layer and be directly added to the output, mitigating the gradient vanishing problem and facilitating the training of deeper networks. This approach helps improve both optimization and convergence. The residual connection can be mathematically expressed as:

$$Output = ReLU(Input + F(Input)) \quad (4)$$

where $F(Input)$ represents the transformation applied to the input through the convolutional layer and its activation function. The input is then summed with this transformed result, and a ReLU activation is applied to generate the final output. This mechanism allows the network to preserve information from earlier layers, aiding the training process, particularly in deeper architectures.

2.2. Assessment of indicators

In this paper, the performance of the model is evaluated using four metrics: Mean Squared Percentage Error (MSPE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2). MSPE measures the average of the squared relative errors between the predicted and actual values, providing insight into the percentage error. RMSE quantifies the deviation between predicted and true values, reflecting how far off predictions are on average. MAE calculates the average of the absolute differences between predicted and true values, focusing on the magnitude of errors. For these three metrics, smaller values indicate higher prediction accuracy, with closer predictions to the true values. Their formulas are as follows:

$$MSPE = \frac{1}{m} \sum_{i=1}^m \left(\frac{y_i - \hat{y}_i}{y_i} \right)^2 \quad (5)$$

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2} \quad (6)$$

$$MAE = \frac{1}{m} \sum_{i=1}^m |y_i - \hat{y}_i| \quad (7)$$

where m is the number of gas well data points, y_i is the true value of gas well production, and \hat{y}_i is the predicted value.

The coefficient of determination (R^2) measures the goodness of fit of the regression model. It represents the proportion of variance in the true values that is explained by the predictions. The closer R^2 is to 1, the better the model's fit.

The formula for R^2 is:

$$R^2 = 1 - \frac{\sum_{i=1}^m (y_i - \hat{y}_i)^2}{\sum_{i=1}^m (y_i - \bar{y})^2} \quad (8)$$

where \bar{y} is the average of the true production values.

3. Model

3.1. Model Buiding

1. Dividing the Dataset: The dataset is split into a training set and a test set, with the first 80% of the data reserved for training and the remaining 20% for testing. The training set is used to train the model, while the test set evaluates its generalization ability.

2. Constructing the TCN Model Architecture: The TCN model consists of an input layer, a TCN layer, and an output layer. The input layer has a shape that matches the time window. The TCN layer includes two one-dimensional convolutional layers, both utilizing dilated convolutions to expand the receptive field. The model starts with 10 filters and a 3x3 convolutional kernel, and uses various dilation factors ('dilations') to capture patterns at multiple temporal scales. The output layer is a dense layer that generates the final predictions.

3. Compiling the Model: The model is compiled using the Adam optimizer and the Mean Squared Error (MSE) loss function. Adam is an adaptive learning rate optimization algorithm that adjusts the learning rate for each parameter based on the first and second moments of the gradients. The MSE loss function is chosen to measure the difference between predicted and actual values.

4. Training the Model: The model is trained on the training set over multiple epochs. During each epoch, the model updates its weights to minimize the loss function. To monitor model performance and prevent overfitting, 20% of the training data is used as a validation set [6].

3.2. Model Parameter

Based on the grid search method, a set of parameters with the best prediction results were finally identified, as shown in Table 1 [7].

Table 1. Model Setup Parameters

Model Parameter	Value
batch_size	64
epochs	200
filter_nums	10

3.3. Results

3.3.1. Results of Well A

In Figure 1, the blue line shows the training set predictions, which are very close to the original data, especially in the first half of the time series. This indicates that the training model captures the variation patterns of the original data very well. The green line is the test set yield prediction, which is basically highly fitted to the test set data.

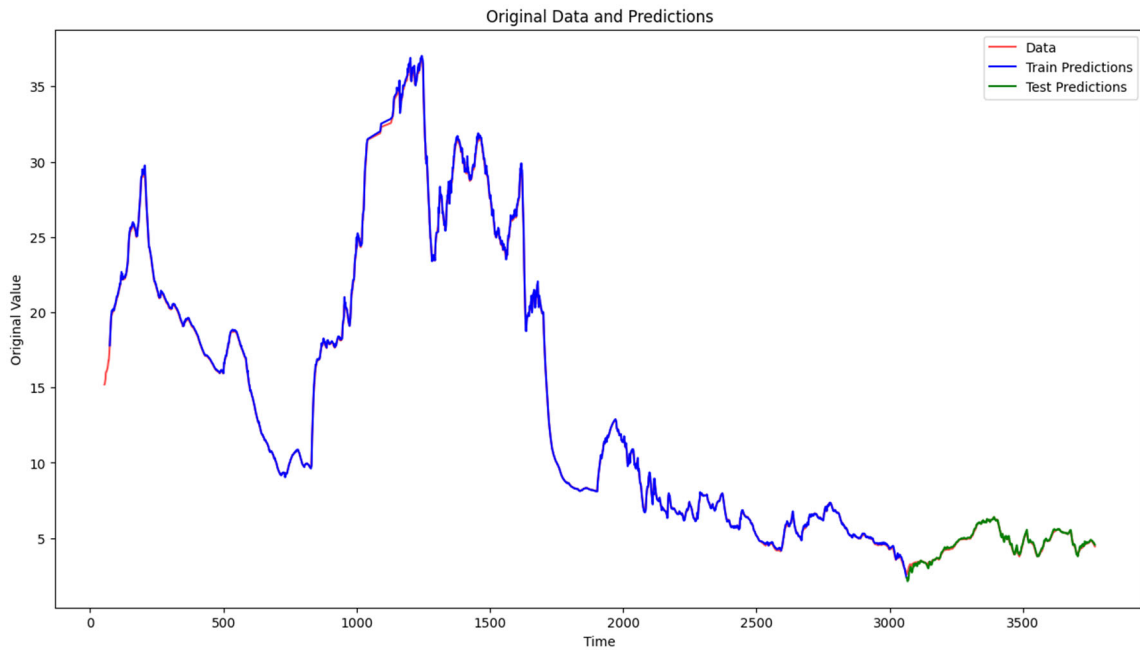


Figure 1. Well A Historical Forecast

Zooming in on the test set section, the production data and predicted values for the test set are shown in Figure 2:

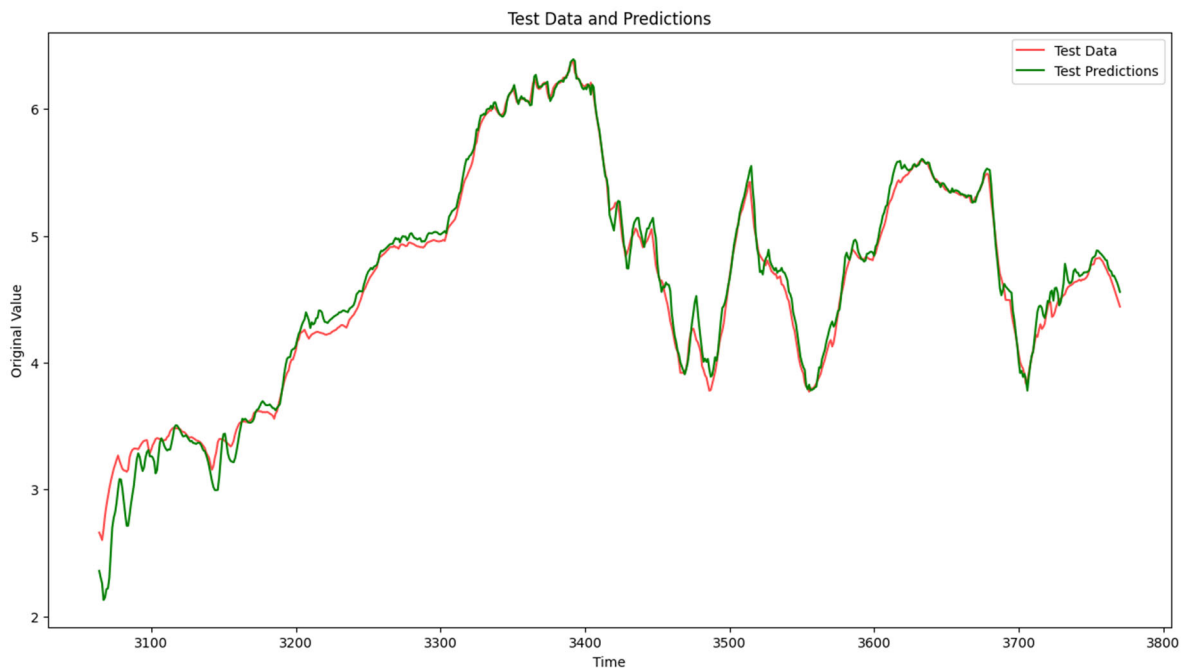


Figure 2. Well A Test Set Prediction

The red line represents the actual data of the test set and the green line represents the predicted data of the test set. As can be seen from the figure, the two curves are very close to each other most of the time, indicating that the predictive model fits the actual data very well. However, in some places, such as around time point 3230 as well as 3620, the predicted values are slightly higher than the actual values; while around time point 3150, the predicted values are slightly lower than the actual values. Overall, the predictive model has a high degree of accuracy, but there is still room for minor improvements. The test set predictions were further evaluated by outputting test set metrics, as shown in Table 2.

Table 2. Test set output metrics

Output metrics	Value
Test MSPE	0.0109
Test MAE	0.0022
Test RMSE	0.0034
R^2	0.9817

The MSPE, MAE, and RMSE values are very small, indicating that the model demonstrates high prediction accuracy. Additionally, the R^2 value is close to 1, signifying a good fit to the data, with only about 1.8% of the variance in the data not explained by the model. Consequently, the TCN

model is highly effective in predicting daily gas production from well A.

3.3.2. Results of Well B

In Figure 3, the blue line illustrates the predictions for the

training set, the green line represents the predictions for the test set, and the red line depicts the complete historical production data. The red line is nearly invisible in the figure, suggesting that the trained model performs exceptionally well in predicting gas well production.

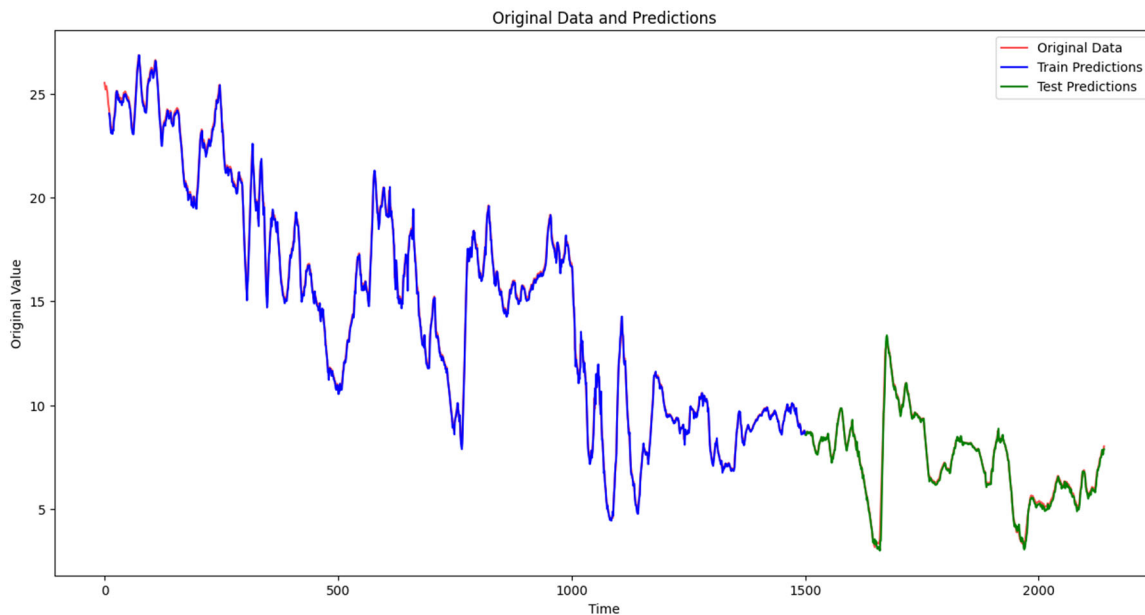


Figure 3. Well B Historical Forecast

Focusing on the test set, Figure 4 provides a closer look at the test set data alongside the corresponding prediction data. This zoomed-in view allows for a more detailed comparison

between the actual production values and the model's predictions for the test set, further demonstrating the model's effectiveness.

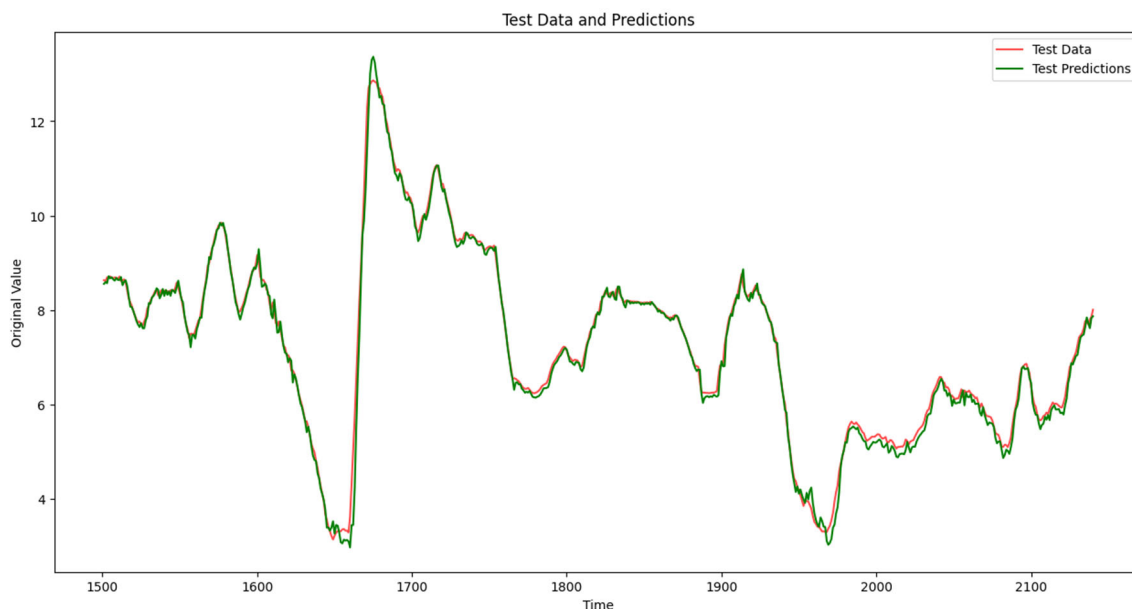


Figure 4. Well B Test Set Prediction

The graph clearly shows that the predicted and actual values closely align for most of the observed period, indicating a very high degree of agreement. Notably, between the time periods of 1600 and 1700, the predicted values closely track the actual values, demonstrating exceptional accuracy. However, around the time point of 2000, the predicted line dips slightly below the actual line, suggesting

that the model may be overestimating future values at this stage.

Overall, the model exhibits impressive accuracy in predicting gas well production. To provide a more comprehensive evaluation, the performance of the model on the test set is further assessed through various metrics, which are presented in Table 3.

Table 3. Test set output metrics

Output metrics	Value
Test MSE	0.0389
Test MAE	0.0047
Test RMSE	0.0076
R ²	0.9915

The values for MSE, MAE, and RMSE are all close to zero, indicating that the deviation between the predicted and actual values is extremely small. This reflects the model's very high prediction accuracy. Additionally, the coefficient of determination R² is notably high at 0.9915, demonstrating that the model explains over 99.15% of the variability in the data. This indicates an excellent fit, further reinforcing the model's effectiveness in predicting gas well production.

4. Conclusion

This paper examines the gas well production prediction method using Temporal Convolutional Networks (TCN) and highlights the advantages of TCNs in improving prediction accuracy. A TCN model was constructed for two gas wells, utilizing the same production data for predictions. The model's performance was evaluated using four metrics: MSE, MAE, RMSE, and (R²). The results demonstrate that the TCN method yields lower values for MSE, MAE, and RMSE on the test set, indicating superior prediction accuracy and model stability. Overall, the TCN method presents a novel approach for gas well production prediction, showcasing significant potential for widespread application.

References

- [1] Arps, J.J. Analysis of Decline Curves. *Trans. AIME* 1945, 160, 228–247.
- [2] Song, X.; Liu, Y.; Xue, L.; Wang, J.; Zhang, J.; Wang, J.; Jiang, L.; Cheng, Z. Time-Series Well Performance Prediction Based on Long Short-Term Memory (LSTM) Neural Network Model. *J. Pet. Sci. Eng.* 2020, 186, 106682.
- [3] Qiao, X. Status Analysis and Development of Reservoir Numerical Simulation Technology. *IOP Conf. Ser. Earth Environ. Sci.* 2021, 631, 012052.
- [4] Zhen, Y.; Fang, J.; Zhao, X.; Ge, J.; Xiao, Y. Temporal Convolution Network Based on Attention Mechanism for Well Production Prediction. *J. Pet. Sci. Eng.* 2022, 218, 111043
- [5] Lara-Benítez, P.; Carranza-García, M.; Luna-Romera, J.M.; Riquelme, J.C. Temporal Convolutional Networks Applied to Energy-Related Time Series Forecasting. *Appl. Sci.* 2020, 10, 2322.
- [6] Wan, R.; Tian, C.; Zhang, W.; Deng, W.; Yang, F. A Multivariate Temporal Convolutional Attention Network for Time-Series Forecasting. *Electronics* 2022, 11, 1516.
- [7] Li, D.; Wang, Z.; Zha, W.; Wang, J.; He, Y.; Huang, X.; Du, Y. Predicting Production-Rate Using Wellhead Pressure for Shale Gas Well Based on Temporal Convolutional Network. *J. Pet. Sci. Eng.* 2022, 216, 110644.
- [8] Ma, X.; Hou, M.; Zhan, J.; Zhong, R. Enhancing Production Prediction in Shale Gas Reservoirs Using a Hybrid Gated Recurrent Unit and Multilayer Perceptron (GRU-MLP) Model. *Appl. Sci.* 2023, 13, 9827.