Computers and Chemical Engineering 197 (2025) 109068

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

A time series forecasting method for oil production based on Informer optimized by Bayesian optimization and the hyperband algorithm (BOHB)

Wu Deng [a], Xiankang Xin [a,b,c,*], Ruixuan Song [d], Xinzhou Yang [e], Weifeng Wang [e], Gaoming Yu [a,b,c]

[a] School of Petroleum Engineering, Yangtze University, Wuhan 430100, China
[b] Hubei Key Laboratory of Oil and Gas Drilling and Production Engineering, Yangtze University, Wuhan 430100, China
[c] School of Petroleum Engineering, Yangtze University: National Engineering Research Center for Oil & Gas Drilling and Completion Technology, Wuhan, 430100, China
[d] Department of Earth and Planetary Science, University of California, Santa Cruz, CA 95064, USA
[e] Shenzhen Branch of CNOOC Limited, Shenzhen 518000, China

ARTICLE INFO

Keywords:
Oil production forecasting
Deep learning
Time series forecasting
Informer
Bayesian optimization and hyperband algorithm

ABSTRACT

Oil production forecasting is essential in the petroleum and natural gas sector, providing a fundamental basis for the adjustment of development plans and improving resource utilization efficiency for engineers and decision-makers. However, current deep learning models often struggle with long-term dependencies in long time series and high computational costs, limiting their effectiveness in complex time series forecasting tasks. This paper introduced the Informer model, an enhancement over the Transformer framework, to address these limitations. For evaluation and verification, the Informer model and reference models such as CNN, LSTM, GRU, CNN-GRU, and GRU-LSTM were applied to publicly available time-series datasets, and the optimal hyperparameters of the model were identified using Bayesian optimization and the hyperband algorithm (BOHB). The experimental results demonstrated that the Informer model outperformed others in computational speed, resource efficiency, and handling large-scale data, showing potential for practical applications in the future.

1. Introduction

Oil production forecasting is essential for decision-making and development performance evaluation during the exploration and development phase. Accurate production forecasts can help engineers and decision makers better understand the pattern of production change, and establish a robust foundation for the rational development of production strategies and optimization of oil recovery schemes. However, production forecasting is a challenging task affected by multiple factors. The static factors mainly include geological characteristics (including reservoir type, reservoir thickness, porosity, permeability, etc.), reservoir pressure, and rock physical properties (e.g., pore structure of rocks, etc.). The dynamic factors mainly include production pressure change, water drive effect, reservoir exploitation technology, environmental factors and so on. Therefore, considering the influence of the aforementioned factors, along with the complexity of reservoirs, the uncertainty of underground conditions, and the specialized knowledge required in petroleum engineering, geology, and physics, accurately predicting oil production remains a significant challenge.

Decline curve analysis and reservoir simulation are well-established methods for analyzing well production in the petroleum industry. These approaches have been broadly utilized in the petroleum and natural gas industries to forecast oil and gas reservoir production dynamics (Ma et al., 2023; Niu et al., 2023). Decline Curve Analysis (DCA) is a well-established classical approach for predicting oil production (Arps, 1945; Jongkittinarukorn et al., 2021). This method presupposes that the decline in production from a well or reservoir comply with a specific mathematical function (exponential, hyperbolic or harmonic decline). Fitting the decline curve to production data enables the initial estimation of fundamental parameters such as decline rate and ultimate recovery, while also providing a straightforward and efficient means for forecasting future production. The mathematical equations of the DCA approach are straightforward and consist of just a few parameters, requiring minimal static and dynamic data, effective to characterize the decline in oil well production. Moreover, the calibration of these parameters requires solely production data (Khanamiri, 2010; Tadjer et al., 2021). Nevertheless, it has many flaws and limitations that impair its predictive ability, leading to a significant error in predictions, which

* Corresponding author.
E-mail address: xiankang.xin@hotmail.com (X. Xin).

https://doi.org/10.1016/j.compchemeng.2025.109068
Received 26 July 2024; Received in revised form 8 February 2025; Accepted 18 February 2025
Available online 22 February 2025

constrain its widespread application. On the one hand, due to the considerable variability observed in many production curves, the manual selection of the starting point for data fitting is paramount, This necessitates petroleum engineers to possess extensive experience, as the subjectivity of analysts can greatly influence the analysis results (Li et al., 2020). On the other hand, these empirical methods neglect to incorporate the actual formation factors such as real reservoir variables, the complex behavior of fluid phases, and other static factors (Li et al., 2022). It is noteworthy that these methods are confined to generating smooth and desirable production curves, whereas real production is generally more irregular (Song et al., 2020; Martínez and Rocha, 2023). Consequently, even with effective historical matching, forecasting results may be unreliable because of dynamic and complex production conditions as well as the influence of subjective factors.

In the realm of current reservoir engineering, reservoir numerical simulation stands out as the most sophisticated and efficient method for predicting petroleum production (Kazemi et al., 1976; Wang et al., 2017). These methods are typically based on mathematical models and physical laws, incorporating factors including reservoir geological structure, rock porosity characteristics, fluid properties, and well network layout. Through numerical calculation method, it simulates the fluid flow and interaction processes within the reservoir. Reservoir numerical simulation can accurately predict crucial parameters such as reservoir production, pressure distribution, and permeability variations, providing a scientific basis for oil field development and management (Nwaobi and Anandarajah, 2018). However, numerical simulation must consider numerous parameters compared to decline curve analysis (DCA), such as petrophysical properties, reservoir characteristic and other parameters. The complexity of data acquisition and characterization of rock and fluid properties is heightened by reservoir anisotropy and heterogeneity, thereby rendering the aforementioned parameters not universally available. Furthermore, constructing such models involves geological model development, numerical formulation, and history matching, which requires a strong technical background and a substantial time commitment. In conclusion, the accuracy and reliability of numerical simulation techniques largely affected by the quality of geological models and the precision of history matching. When confronted with challenges such as incomplete reservoir parameters and complex fluid flow dynamics, numerical simulation techniques struggle to provide precise production forecasts. Additionally, they are time-consuming and operationally complex (Negash and Yaw, 2020; Al-qaness et al., 2022).

Accurate production forecasting using conventional methods is challenging because of the intricate nature of geological and reservoir data, along with dynamic operational management events and rapid production decline (Sun et al., 2018). Therefore, the objective of this research is to establish faster, more precise, and easier to implement methods for predicting oil well production. Advances in machine learning theory have facilitated the application of AI algorithms in the realm of petroleum engineering, addressing the shortcomings of conventional production forecasting methods (Cao et al., 2022; Ng et al., 2023). Data-driven techniques for production forecasting provide benefits such as accelerated training times. Such as fast training speeds, low development costs, and minimal parameter requirements, and ease of implementation with less reliance on specialized knowledge, which are challenging to attain using traditional decline curve analysis methods and numerical simulation (Liu et al., 2020). Nevertheless, the production sequence in oil reservoirs falls into the category of time series data. In essence, oil production forecasting constitutes a problem of time series prediction, where the precision of the forecasting model is largely determined by whether it incorporates temporal relationships at the time level within the data (Song et al., 2020; Cao et al., 2022). Over the past few years, researchers have attempted to employ complex and sophisticated neural network to address the nonlinear behaviors and temporal characteristics of oil wells (Schmidhuber, 2015; Sheikhoushaghi et al., 2022). The literature on various oil well production forecasting methods is summarized in Table 1.

While RNNs are extensively applied in production forecasting, the issues of gradient explosion and vanishing gradients restrict their performance with long time series data. In contrast to RNN, CNN can process the entire sequence in parallel, leading to higher computational efficiency. However, its effectiveness in capturing the temporal dependencies within time series data is limited, resulting in struggle to effectively handle long time series problems (Lecun et al., 1998). As a variant of Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) is commonly applied to time series forecasting. Compared to traditional RNNs, LSTM possesses a stronger memory capability, enabling effective handling of long-term dependencies within time series. By simplifying the internal cell structure of LSTM, GRU shortens training time while preserving prediction accuracy. Despite the significant progress made by LSTM and GRU in effectively addressing the challenges of gradient vanishing and exploding in traditional RNNs,

Table 1
Summary of literature on common prediction models.

Literature	Methods	Contributions	Insights/Limitations
Mohd Razak et al. (2022)	RNN (Rumelhart et al., 1986; Williams and Zipser, 1989), Transfer Learning (Pan and Yang, 2010)	An RNN model was proposed using transfer learning to predict oil, water, and gas production under varying controls with minimal target well data.	Transfer learning in the study has certain limitations, including subjectivity and potential bias in assessing source-target relevance, challenges in generalizing to diverse fluid or geologic properties, and the risk of negative transfer when source datasets do not sufficiently match target tasks.
Ning et al. (2022)	ARIMA (Anderson et al., 1978), LSTM (Hochreiter and Schmidhuber, 1997), Prophet (Taylor and Letham, 2017)	ARIMA, LSTM, and Prophet models were systematically compared, with emphasis placed on their ability to generate rapid predictions without necessitating extensive reservoir knowledge. The models effectively captured declining trends, demonstrating strong representativeness.	Each model has limitations, including ARIMA's reliance on stationarity, LSTM's complexity in selecting optimal architectures, and Prophet's tendency to potentially overestimate seasonal effects, despite their robust performance in capturing declining trends and seasonality.
Ng et al. (2022)	SVR, FNN (Rumelhart et al., 1986), RNN / PSO	Models for forecasting oil production in the Volve oil field were developed using SVR, FNN, and RNN, with PSO integration. High predictive accuracy was achieved, with R^2 values exceeding 0.94.	All models demonstrated strong performance; however, addressing complex nonlinear relationships or dynamic changes may require further adjustments or more sophisticated models.
Martínez and Rocha (2023)	LSTM, GRU (Cho et al., 2014)	The LSTM and GRU models were developed and compared, with adjustments made to architecture and input/output parameters.	The study is constrained by challenges such as handling zero values, removing outliers, mitigating noise in the dataset, and optimizing hyperparameters more effectively.
Xu and Leung (2024)	RUF	A novel framework, Recurrent Update Forecasting (RUF), is proposed to enhance accuracy and training efficiency by leveraging the full time-series structure.	The RUF framework faces limitations in extrapolation accuracy, handling arbitrary output sizes, and inefficient cache usage due to discarding intermediate predictions, which increases prediction time.

continued refinement is necessary.

In application, a single ML model may exhibit drawbacks such as low convergence, outlier influence, and limited predictive capability etc. Its applicability varies significantly depending on different data sets and conditions (Li et al., 2022). Whereas single model is insufficient for addressing complex problems, combination structures have emerged as a research trend in time series forecasting (Kong et al., 2023). Researchers have found that combining different models and algorithms can substantially enhance model accuracy, stability, and generalization capability in time-series prediction problems (Iwana and Uchida, 2021; Wen et al., 2023; Tu et al., 2024). The relevant literature on using hybrid methods to forecast oil well production is summarized in Table 2.

While the predictions from the aforementioned studies are quite accurate, they still face some unavoidable challenges. Firstly, the hybrid model's performance is largely determined by the effectiveness of its constituent units, and their combined performance. If the base units are selected inadequately or exhibits deficiencies, the efficacy of the hybrid model may fall short of expectations. Secondly, hybrid models involve multiple individuals. Each potentially requiring tuning of its specific hyperparameters, which significantly increases the complexity of hyperparameter tuning and the consumption of calculation resources. Researcher must allocate considerable amount of time to find the optimal hyperparameter configurations, as well as to train and fine-tune the models. Consequently, while hybrid models can significantly improve predictive accuracy, it is essential to minimize model complexity and calculation resource consumption when employing them. In practical applications, accurately forecasting single-well production is challenging because its fluctuate are more unpredictable and drastic compared to block or reservoir production (Zhang et al., 2023). The implementation of measures such as water injection, enhanced oil recovery during development process causes significant fluctuations in well production at different stages, indicating varied patterns of change (Prasetyo et al., 2022).

The attention mechanism enhances information extraction by focusing on the most relevant data, thereby effectively improving the accuracy of prediction results. Considering the attention mechanism's ability to strengthen the impact of important features on predictive outcomes, its practical combination with other deep learning models has achieved promising forecasting outcomes (Vaswani et al., 2023). In this paper, a new temporal forecasting framework known as "Informer" is introduced. Based on the Transformer architecture, the Informer model integrates the self-attention mechanism, which addresses various issues encountered by conventional time-series approaches, such as time lag, accumulation of errors during training on long sequences, and slow training speeds. This study employs dynamic data, including oil, gas, and water production rates, to develop multi-feature models for predicting single-well production, while addressing the nonlinearity of production data by designing and implementing a Dynamic Floating Window (DFW) technique. Finally, the Informer model's generalization ability is assessed through various performance metrics, and the predictive results of the Informer model are compared with reference models under equal optimization conditions.

The other sections of this paper are organized as follows. The Methodology section introduces readers to the fundamental principles of the Informer model and other models (CNN, LSTM, GRU, CNN-GRU, GRU-LSTM), which are crucial for understanding the work in this paper. Next, a concise dataset overview is provided, followed by a detailed analysis and meticulous data processing. Finally, after training the processed dataset using multiple models and analyzing the research findings, the "Results and discussion" and the "Conclusion" sections are presented.

2. Methodology

2.1. Informer neural network

At present, three principal types of feature processors are employed in the field of natural language processing, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), and transformers. The Transformer framework discards traditional CNN and RNN neural networks, instead comprising an entirely Attention-based feed-forward neural network framework. A multitude of empirical studies have demonstrated the efficacy of Transformer in time series prediction tasks (Lin et al., 2021).

Based on the Transformer architecture, the Informer model offers enhanced capabilities for long sequence time-series forecasting (LSTF) (Zhou et al., 2021). Fig. 1 shows the architecture of Informer.

Compared to the traditional Transformer model, Informer has three distinctive features as follows.

Feature 1. Probsparse attention

The Informer model employs the Probsparse self-attention mechanism, which reduces the time complexity and memory usage of the original Transformer model from $O(L^2)$ to O(L log L). This mechanism enhances the model's scalability in handling long sequence inputs, enabling it to effectively capture long-term dependencies within the sequences. The sparse self-attention mechanism is a distinct adaptation

Table 2
Summary of literature on hybrid prediction methods.

Literature	Methods	Contributions	Insights/Limitations
Zhen et al. (2022)	TCN (Lea et al., 2017), AM	A TCN-attention model combining temporal convolution and attention mechanisms is proposed in the study for improved oil production prediction.	The TCN model alone doesn't outperform RNN and LSTM in well production prediction, so it needs to be combined with other methods for better performance.
Li et al. (2022)	Bi-GRU/SSA	A novel framework using Bi-GRU and SSA is proposed to enhance oil production prediction, with superior accuracy and robustness demonstrated over traditional methods and unidirectional RNNs.	The model's accuracy and generalization can be improved, and more research on multi-well differences is needed to address temporal-spatial production prediction.
Chang et al. (2023)	GRU-AM/ Physics Constraint	A novel physics-constrained sequence learning method with attention for multi-level production prediction is proposed, showing significant advantages over ten baselines and validating key model components through ablation studies.	The proposed method relies on known future engineering parameters as physical constraints; if unavailable, assumptions must be made, focusing on predicting potential production under predefined development plans.
Zhou et al. (2023)	CNN-Bi-GRU-AM	A CNN-Bi-GRU-AM method is proposed for predicting shale oil production, with spatial-temporal features, irregular trends, and intrinsic information leveraged to enhance prediction accuracy, providing advanced tools for shale oil forecasting.	Multiple extracted integrated models could prove pivotal for the accelerated discovery and development of production prediction. Additionally, the importance of variables in influencing predictions is significant.
Zhang et al. (2023)	CNN-LSTM	The study introduces a Bayesian-optimized CNN-LSTM model incorporating key factors, with prediction accuracy exceeding 90 %. The model is validated in actual reservoirs and provides guidance for waterflooding and other reservoir applications.	The study demonstrates that complex models effective in other tasks may not be suitable for well production prediction.
Chen et al. (2024)	CNN-GRU	The study presents a novel CNN-GRU model for accurate oil production forecasting, enhancing EOR performance, and demonstrated to outperform other deep learning and hybrid methods.	The study's limitations include the longer runtime of the CNN-GRU and the focus on a single variable, necessitating future testing with multivariate data and different oil fields.

of the self-attention mechanism, and its calculation process can be described as follows:

$$ProbSparseAttention(Q,K,V) = softmax\left(\frac{Q_{reduce}K^T}{\sqrt{d}}\right)V \quad (1)$$

where Q represents the query matrix, K represents the key matrix, V represents the value matrix, Q_{reduce} is a sparse matrix of the same size as q, which includes only the Top-u queries based on the sparsity measure $M(q_i, k)$ and d represents the dimension of the query, key, and value matrix.

The above equation includes only the Top-u queries within the sparse metric $M(q_i, K)$, where the formula for $M(q_i, K)$ is displayed in the Eq. (2).

$$M(q_i,K) = ln\sum_{j}^{L_K} e^{\frac{q_i k_j^T}{\sqrt{d}}} - \frac{1}{L_K}\sum_{j}^{L_K}\frac{q_i k_j^T}{\sqrt{d}} \quad (2)$$

In this formula, the first term represents the LSE (Log-Sum-Exp) of q_i across all keys, while the subsequent term is their arithmetic mean.

Feature 2. Self-attention distilling

Informer decreases the scale of dimensionality and the number of network parameters by employing self-attention distillation, which is a method of extracting the dominant attention. This approach allows for the efficient handling of long input sequences. In fact, minimize the length of the input sequence in each layer, the "distillation" operation in the Informer is akin to downsampling. This approach involves the insertion of one-dimensional convolutional layers and subsequent max pooling between encoder layers, effectively halving the input sequence length for each layer.

The process of distillation from the j^{th} layer to the $(j+1)^{th}$ layer is expressed as:

$$X_{j+1}^t = MaxPool\left(ELU\left(Convld\left(\left[X_j^t\right]_{AB}\right)\right)\right) \quad (3)$$

Where the notation $[\cdot]_{AB}$ represents the attention block, which includes multi-head Probsparse self-attention and basic operations, whereas Convld() represents a one-dimensional convolution performed along the time dimension(kernel width=3),with ELU() used as the activation function.

Feature 3. Generative decoder

Informer's generative decoder enables predicting the whole long time series with only one forward operation instead of step-by-step, thus preventing the spread of cumulative errors during inference. This approach drastically improves the prediction efficiency and accuracy of long-sequence predictions, overcoming the limitations of traditional Transformer models. As shown in the structural diagram of Informer, the decoder adopts encoder mechanism, which is composed of two identical multi-head probability coefficient self-attention layers stacked sequentially. The preceding segment of this section corresponds to the output of the encoder, while the other segment corresponds to the embedded decoder input, which is masked with zeros.

We will take the following vector as input to the decoder:

$$X_{de} = Concat(X_{token}, X_0) \in R^{(L_{token}+ L_y) \times d_{model}} \quad (4)$$

Where X_{token} represents the start token, and X_0 serves as a placeholder for the target sequence (scalar set to 0). The Probsparse self-attention method applies masked multi-head attention by setting masked dot products to -∞ to ensure each position avoids attending to subsequent positions, effectively preventing autoregression.

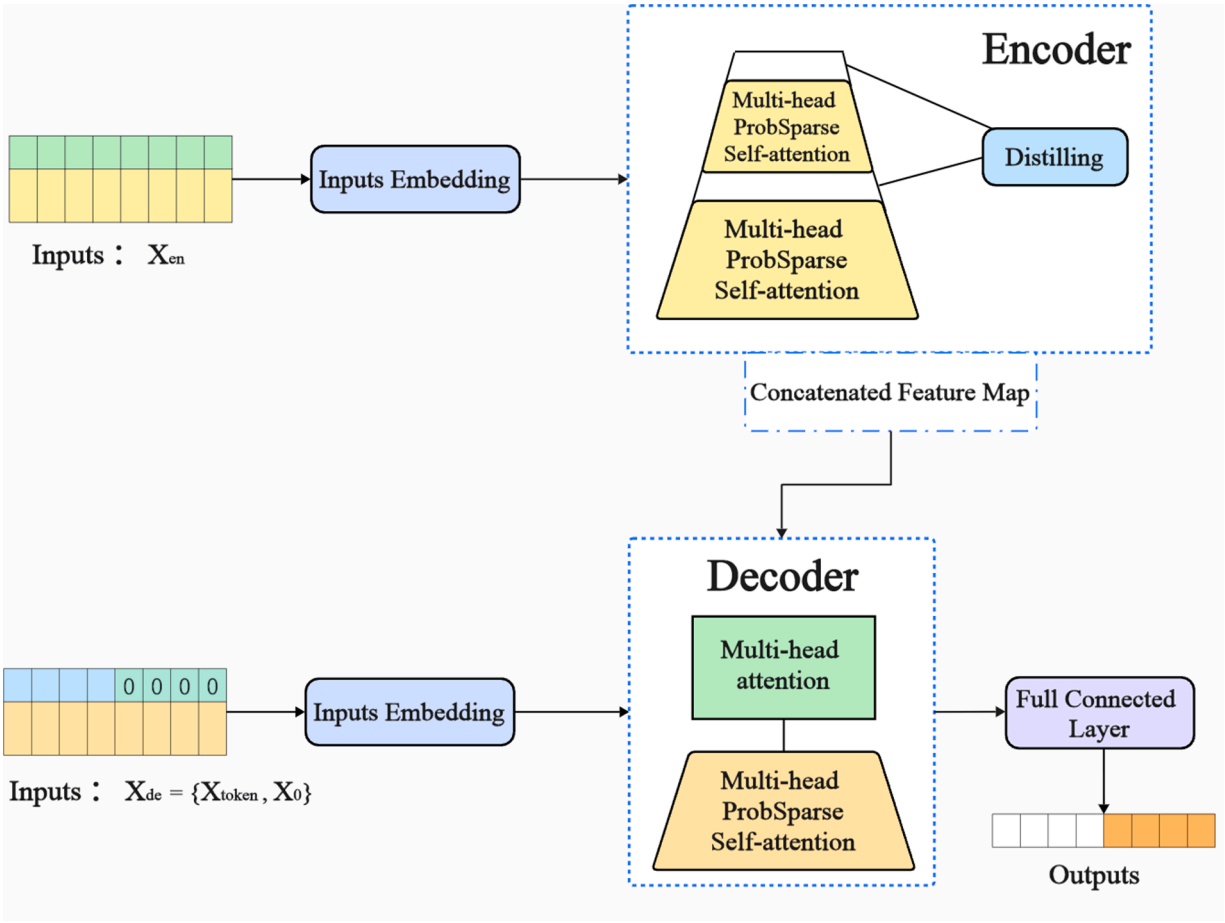

Fig. 1. Internal structure of Informer.

2.2. Reference models

For performance evaluation, this study compares the introduced model with distinct reference models, such as commonly used deep learning models and hybrid models. The following provides a concise summary of the reference models used for comparison.

CNNs (Convolutional Neural Networks) can automatically extract useful features from time series data while preserving most information, thereby enhancing generalization ability and training efficiency. LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit), both derived from RNNs, are designed to capture long-term dependencies and short-term memory. LSTM employs input, output, and forget gates, whereas GRU, with its update and reset gates, provides a simpler and computationally more efficient alternative. Hybrid models such as CNN-GRU and GRU-LSTM leverage the strengths of these architectures: CNN-GRU extracts spatiotemporal features by feeding CNN-derived features into a GRU network, while GRU-LSTM integrates the computational efficiency of GRU with the memory capability of LSTM in a multi-layer structure, balancing convergence speed and predictive accuracy for improved time series forecasting.

2.3. Hyperparameter optimization algorithm

The determination of the model structure parameters is a crucial step in model training, and numerous hyperparameters greatly impact the model's predictive performance. The basic strategies for determining hyperparameters are random search and grid search, optimization algorithms, etc. Nevertheless, these methods may require numerous experiments to discover optimal hyperparameter combinations, which is computationally inefficient.

The Bayesian optimization algorithm searches for points in unknown regions that are likely to minimize the loss function based on current data. It iteratively builds Gaussian process models and selects the most informative points for exploration, thereby rapidly finding the optimal hyperparameter combination in the search space. However, when handling complex time series models, the Bayesian optimization algorithm necessitates multiple iterations of model training, potentially leading to substantial resource and time consumption. The Hyperband algorithm is based on stratified sampling and the early stopping principle to find near-optimal result with a relatively small budget through dynamic resource allocation. To address the problem of inefficient computation in complex deep learning models with the Bayesian optimization algorithms, this study employs the BOHB algorithm for hyperparameter optimization of the models. The workflow of BOHB algorithm is shown in Fig. 2.

This algorithm combines the advantages of Bayesian optimization and the hyperband algorithm by using the resource allocation strategy from hyperband to evaluate different hyperparameter combinations, and then update the surrogate model in Bayesian global optimization. Based on the optimization results, it employs an early stopping mechanism to reduce computational costs and avoid multiple iterations during the experimental process. The BOHB algorithm can efficiently find optimal hyperparameter configurations while exhibiting robustness and scalability, providing a novel solution for advanced hyperparameter optimization techniques (Falkner et al., 2018).

2.4. Evaluation criteria

Many machine learning models exhibit instability, which means that slight changes to parameters or conducting multiple training iterations may lead to significant variations in their performance. This indicates that single-model evaluations may be unreliable, and may either underestimate or overestimate the model's true potential. Therefore, it is common practice to deploy multiple evaluation metrics. Different metrics can present varying perspectives of results and enhance transparency in research.

Traditionally, the performance of regression models such as those used for forecasting time series data is evaluated using mean squared error (MSE), mean absolute error (MAE), coefficient of determination (R^2), and root mean squared error (RMSE). Attributable to the pronounced fluctuations in the production data of single wells, MSE and RMSE are highly sensitive to points with significant fluctuations, potentially misrepresenting model performance. The above evaluation metrics are limited to comparing the predictive performance of different models on the same dataset. Therefore, mean absolute percentage error (MAPE) and symmetric mean absolute percentage error (SMAPE) were selected for their interpretability and comparability across different scales, while mean directional accuracy (MDA) was included to evaluate both error magnitude and directional accuracy, offering a more comprehensive assessment. In this study, the commonly adopted metrics such as R^2, MAE, MAPE, SMAPE, and MDA are used. These are represented as follows.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{5}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{6}$$

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \tag{7}$$

$$SMAPE = \frac{100\%}{n}\sum_{i=1}^{n}\frac{|\hat{y}_i - y_i|}{(|\hat{y}_i| + |y_i|)/2} \tag{8}$$

$$MDA = \frac{1}{N}\sum_{t} 1(sign(y_t - y_{t-1}) == sign(\hat{y}_t - \hat{y}_{t-1})) \tag{9}$$

In these equations, y_i represents the known target value, $\hat{y}_i$ represents the predicted value, and $\bar{y}$ represents the average of the target value. y_t and $\hat{y}t$ represent the actual value and the predicted value at time t, respectively;$1(\cdot)$ is the indicator function, which returns 1 when the condition inside the parentheses is true and 0 otherwise.

When the coefficient of determination (R^2) within a high range, lower values of MAE, MAPE, and SMAPE reflect better model performance in time-series forecasting. MDA measures the consistency between the predicted and actual trends at each time step. The resulting metric ranges from 0 to 1, where a higher value signifies greater accuracy in the model's ability to predict directional trends.

3. Case studies

3.1. The entire workflow of the case studies

The case study process is depicted in Fig. 3. The following provides a detailed description of each step in the flowchart. Firstly, a comprehensive dataset was collected and preprocessed for deep learning algorithms, including standardization of data and the creation of time-series datasets using sliding windows. Secondly, this study implemented various reference models and the Informer model and compared their prediction performance. Finally, the prediction performance of the aforementioned models was analyzed, with a discussion on the practical significance of hybrid methods and the Informer model.

3.2. Dataset description

The production dataset used in this study is sourced from the Volve oil field on the Norwegian Continental Shelf, spanning approximately eight years of historical data. This dataset contains a large amount of monthly production data of single wells, which provides the potential for oil prediction by deep learning models. The contribution of wells 15/9-F-12H and 15/9-F-14H to crude oil production at the Volve oilfield is

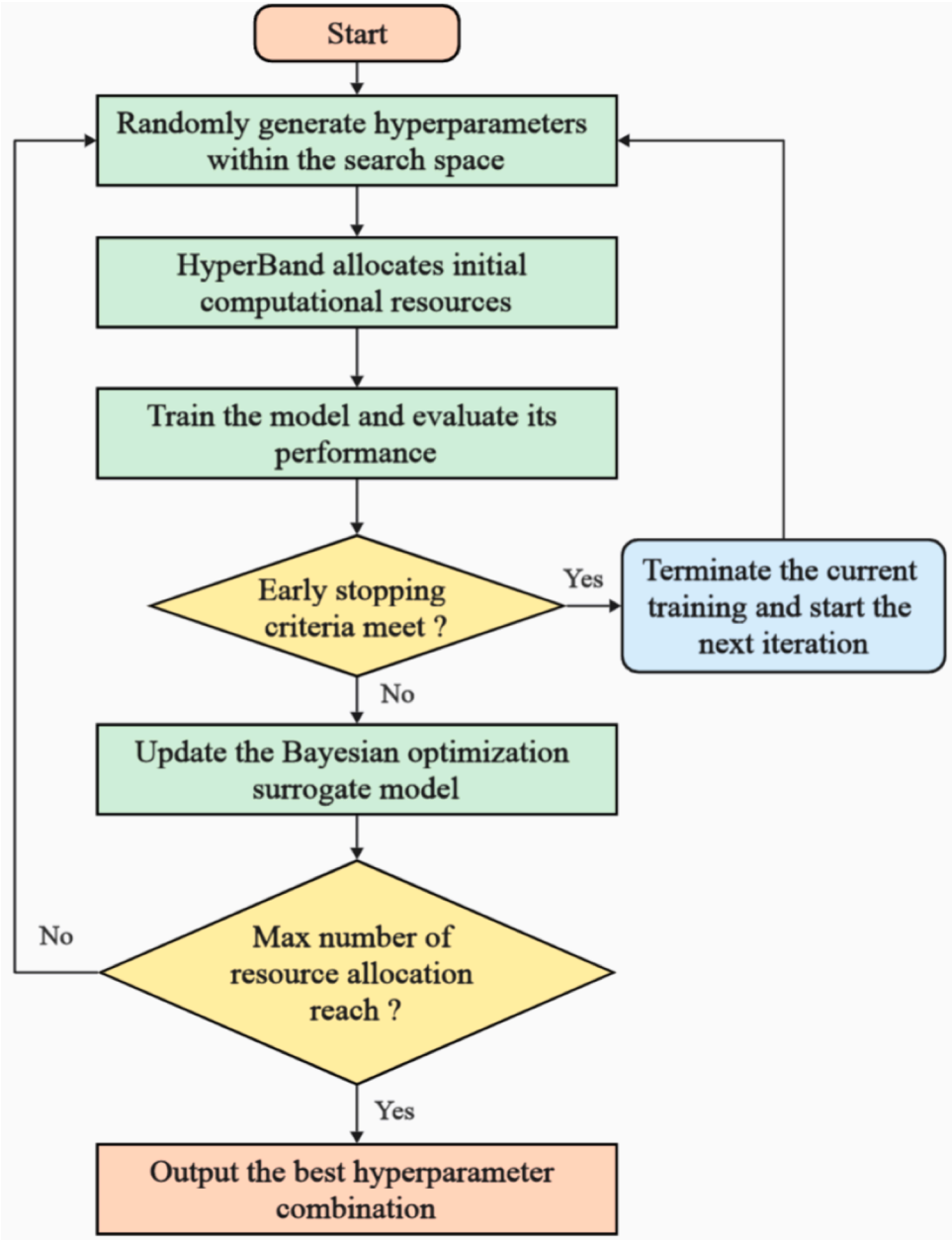

Fig. 2. The workflow of BOHB for optimizing hyperparameters.

nearly 85 %,with peak production rates of 5561 m^3/d for well 15/9-F-12H and 4885 m^3/d for well 15/9-F-14H (Paullada et al., 2021). For practical considerations, the data of longest production well (NO15/9-F-12H) will serve for training and assessing the model in this research.

In addition, it is necessary to extract the relevant and sufficient feature from raw data to optimize model performance. Accordingly, columns containing a significant number of null values are excluded from this study.

To further describe the internal variation and magnitude of fluctuations within the data, this study employs the coefficient of variation (the ratio of the standard deviation to the mean, dimensionless) and the standard deviation to quantitatively measure the dispersion of the data. Larger values represent more dispersion, while smaller values represent less dispersion. Following preliminary screening and processing (removing zero values), the coefficients of variation and standard deviations for each parameter were calculated and are shown in Table 3. In this study, the selection of inputs is based on reservoir and production engineering expertise. To ensure the effectiveness of the model, features with more comprehensive data are prioritized for input.

Furthermore, the oil production profile of well NO15/9–F–12H after screening is shown in Fig. 4.

The following conclusions can be derived from Fig. 5. The ACF plot suggests that the time series is non-stationary and may exhibit long-term dependency. The first lag in the PACF plot shows a prominent peak followed by a sharp decline, which implies that the data points in the NO15/9–F–12H production data are primarily influenced by their preceding data point. Furthermore, the majority of lagged PACF values fall within the blue confidence interval, indicating that time series forecasting models might achieve decent predictive results.

3.3. *Data preprocessing*

Oil reservoir production data typically include missing values and exhibit non-stationary, which affects the accuracy and reliability of model predictions. Consequently, pre-processing of the raw production

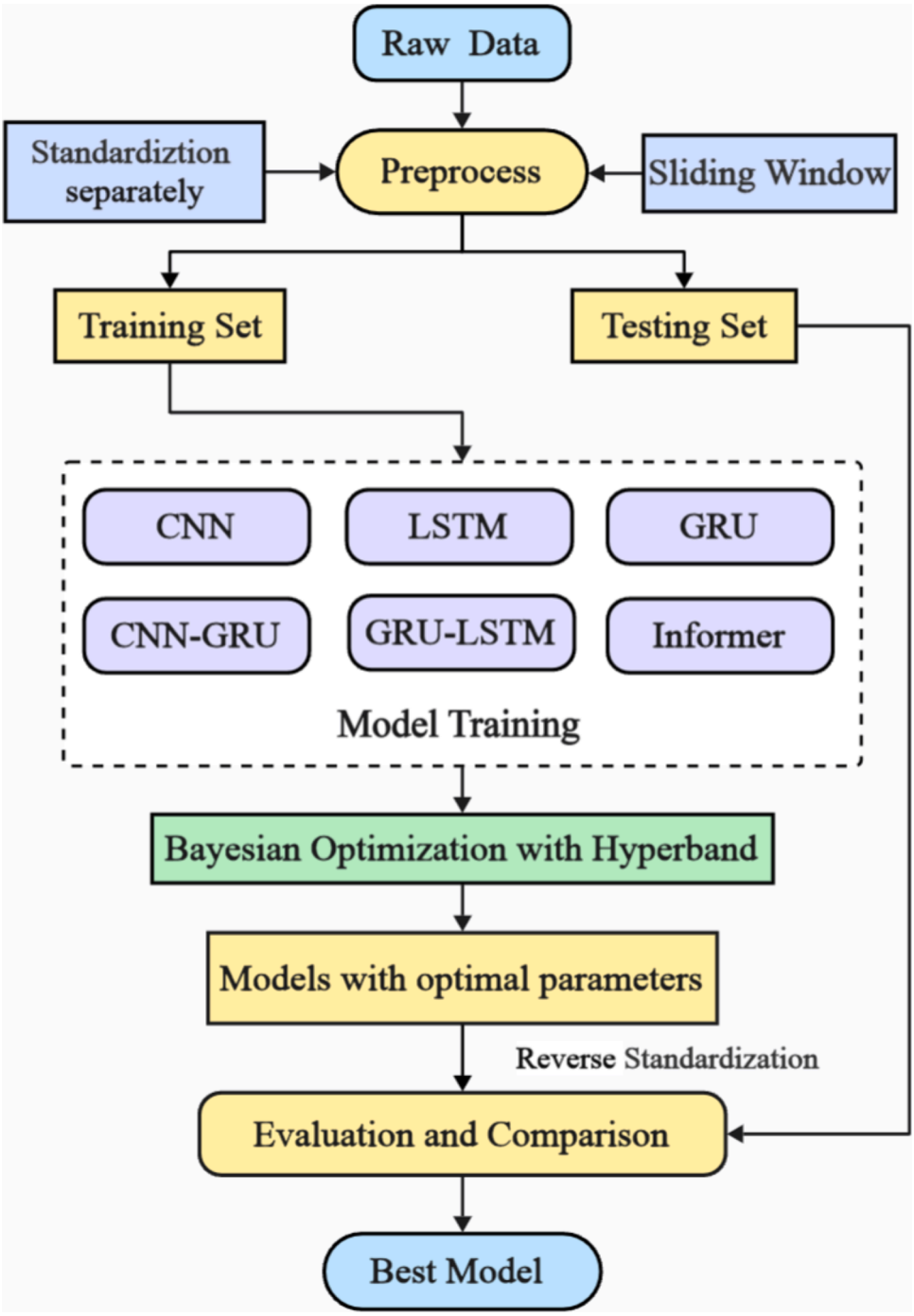

Fig. 3. Overall workflow of case study.

data is necessary before training the model. A small portion of missing data was imputed based on industry knowledge and interpolation algorithms.

3.3.1. *Standardization*

From the above data analysis, it can be concluded that the coefficient of variation and standard deviation of both the feature and label columns are substantial, which indicates poor stability and significant fluctuations in the data. Additionally, there are significant differences in data scales among various columns. Due to algorithms potentially prioritizing features with larger numerical values, those features with larger values or broader ranges will greatly impact the model's training and predictions, thereby diminishing the relative impact of other

Table 3
Coefficient of variation (Cov) and standard deviation (Std) of input and output parameters of the well NO15/9-F-12H.

Parameters	Cov	Std
Average Differential Pressure of Tubing	0.86	70.18
Average Annular Pressure	0.28	5.21
Average Choke Size Percentage	0.34	25.51
Average Wellhead Pressure	0.41	19.04
Average Wellhead Temperature	0.08	7.13
Differential Pressure Choke Size	1.08	17.36
Water Volume from Well	1.01	2.27×10^5
Gas Volume from Well	0.75	1.89×10^3
Oil Volume from Well	1.05	1.61×10^3

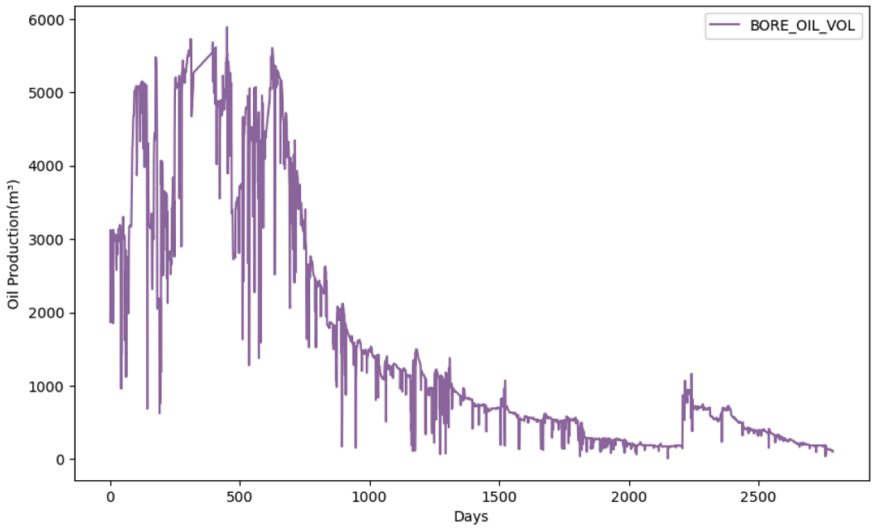

Fig. 4. Oil production of the well NO15/9–F–12H.

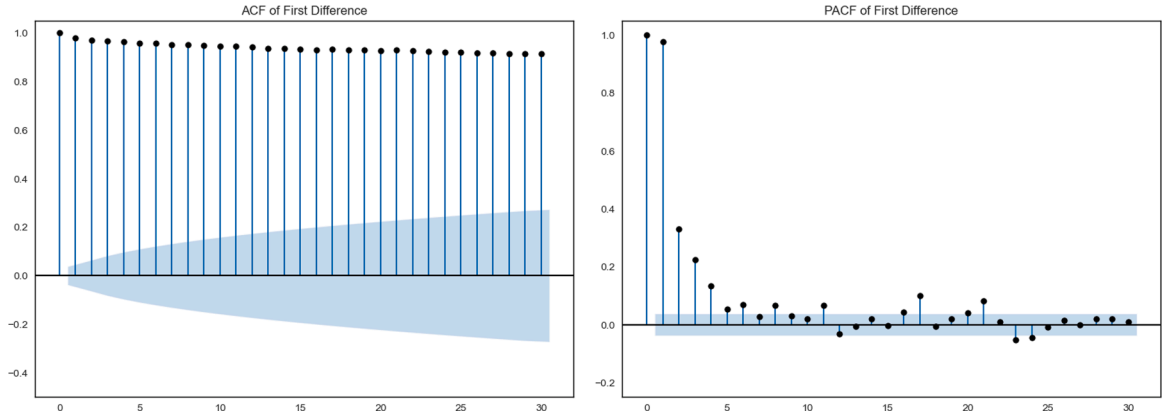

Fig. 5. PACF and ACF analysis plots.

features on the final prediction results.

Therefore, the dataset will be standardization to transform the distribution with a mean of 0 and a standard deviation of 1, which can reduce the influence of data scale and variability on the algorithm in this paper. The standardization of data can not only accelerate the convergence speed of the model (gradient descent solution speed), but also prevent the model gradient explosion problem caused by the excessive gradient value of some data in the back propagation process, and improve the efficiency and accuracy of the algorithm.

$$X_{i,new} = \frac{X_i - \mu}{\sigma} \tag{10}$$

where $X_{i,\,new}$ represents the standardized value of the actual sample data X_i, where μ is the mean of the sample data and σ is the standard deviation (Std) of the sample data.

It is worth noting that time series data differs from many other kinds of data in that the order of the data points is important. To prevent leakage of training data into the test set, the original dataset should be split first, and then the training and test sets should be standardized separately. Finally, during prediction result validation, they should be denormalized separately. In this study, the data was separated into training and testing sets using a 17:3 split ratio. This implies that the initial 85 % of the data is allocated for model training, while the remaining 15 % will serve as blind examples to evaluate the model's predictive capabilities. It is important to recognize that the division between the training set and the prediction set is determined by practical considerations related to the actual data and relies on the researcher's understanding of the data and research experience.

3.3.2. *Sliding window configuration*

For predicting time series data, it's often necessary to establish a model that relies on previous data points for future predictions. A sliding window operation is performed on the original dataset to fully exploit the available data in this study (Davtyan et al., 2020; Huang and Deng, 2021).

The sliding time window algorithm is a commonly method for handling time series data. This approach slides the time series data according to a fixed window size, with slides a fixed step size each time to generate a series of subsequence samples to produce data and labels for training or testing time series. By continuously sliding the time window, this algorithm can perform continuous processing and analysis of time series, thereby eliminating the noise interference of a single data point, capturing the local characteristics and trends of the time series data, and improving the forecasting precision.

Oil well production activities typically fluctuate on a weekly, monthly, or yearly cycle, with output often influenced by factors such as operational conditions of production facilities, climate changes, and work rates. Therefore, considering the temporal characteristics of the data and the practical context of the prediction task, the case study was initially configured to utilize six time steps to predict the target variable (oil production) for the subsequent time step. This allows forms a time-series window that spans one week, which is a common cycle in real-life

and production scenarios (e.g., variations between weekdays and weekends). In other words, such a setup is not only theoretically reasonable but also aligned with practical application needs. The specific implementation process is shown in Fig. 6.

The shape of the input is (batch size, window size, features), while the shape of the tensor output is transformed to (batch size, window size, target) by the sliding window mechanism. In this context, batch size represents the number of data entries input into the model at one time; window size represents the length of each input sequence; features represent the number of features included at each time step; target represents the number of variables the model needs to predict.

This method not only allows the model to focus more on trend changes rather than single data fluctuations, but also helps reduce computational complexity and data volume, which is particularly important for real-time predictions and model updates. Many researchers have introduced the time-series sliding window technique in forecasting tasks, where the size of the sliding window is set to determine the historical time steps fed into the model at each iteration. In addition, researchers often treat the sliding window size as a hyperparameter and use various optimization algorithms to tune it in order to find the optimal window size.

Chen et al. (2024) set the Time lag to range from 1 to 3; Zhen et al. (2022) set the Windows length to range from 4 to 35; Liu et al. (2020) set the Lag to range from 1 to 10; Yan et al. (2024) based on the monthly single-well production dataset, the window size is set to range from 12 to 24. The above researchers analyzed the datasets and leveraged their experience to set different ranges of window lengths, aiming to better align with the production patterns of oil wells and enhance the model's predictive performance under varying conditions. In the subsequent model optimization process, we consider defining a range of window sizes, allowing the optimization algorithm to automatically adjust the sliding window length. This approach can gradually improve model performance through automated parameter tuning, adapt to the optimal temporal dependency structure of the data, and make the prediction results more reliable.

4. Results and discussion

In this section, this study commences by training all models discussed in Section 2 with a unified dataset, applying the optimal hyperparameter combinations by Bayesian optimization with hyperband algorithm to configure the models. Following this, the Informer model is evaluated for its effectiveness and compared to the performance of other reference models. A reliable assessment of prediction performance should depend on test data rather than train data. Most deep learning models perform well on the training set, while the error accumulation is basically generated on the test set. Therefore, this section primarily demonstrates the test set predicted outcomes to evaluate the generalization capabilities of the models.

4.1. Model evaluation and comparison

In practical applications, four parameters are considered for optimization using the Bayesian optimization and hyperband algorithm: time step (the number of previous time steps, with the prediction step size to 1), the number of epochs, batch size, and learning rate, while manually adjusting parameters such as the number of layers and neurons in the neural network. We initially set the model hyperparameters as follows: time step of 6, epochs of 500, batch size of 24, and learning rate of 1e-4. The model was trained on the raw data that had only been standardized, and we obtained the loss value changes with respect to epochs for different models, as shown in Fig. 7.

By analyzing the loss change plots, we can evaluate the model's training process, observe the trend of the loss values, and determine whether the model has converged, as well as identify if overfitting or underfitting occurs during training. This helps optimize the model's hyperparameter settings and improve its performance.

As shown in Fig. 7, the CNN and CNN-GRU models experienced significant fluctuations in the early of training. This is typically because the structures of these models may be more sensitive when capturing local features of the data, leading to unstable weight updates across epochs on a global scale. Such fluctuations do not necessarily indicate poor model performance; they may reflect a higher model complexity, requiring more training epochs to learn the complex relationships within the data. This also indirectly suggests that the number of training epochs is likely an important hyperparameter for the aforementioned models. For models such as LSTM, GRU, GRU-LSTM, and Informer, the loss curves tend to stabilize after approximately 100 epochs. This indicates that these models have largely captured the primary trends and patterns in the data by this point, and further increases in the number of epochs may not result in significant performance improvements. The phenomenon indicates that the learning process of these models is relatively stable, and they are able to reach a convergent state within a relatively small number of epochs.

The Informer model is specifically designed to capture long-term dependencies and global information, featuring a self-attention mechanism and efficient temporal modeling capabilities. The model has a faster convergence speed and can achieve lower losses in fewer training

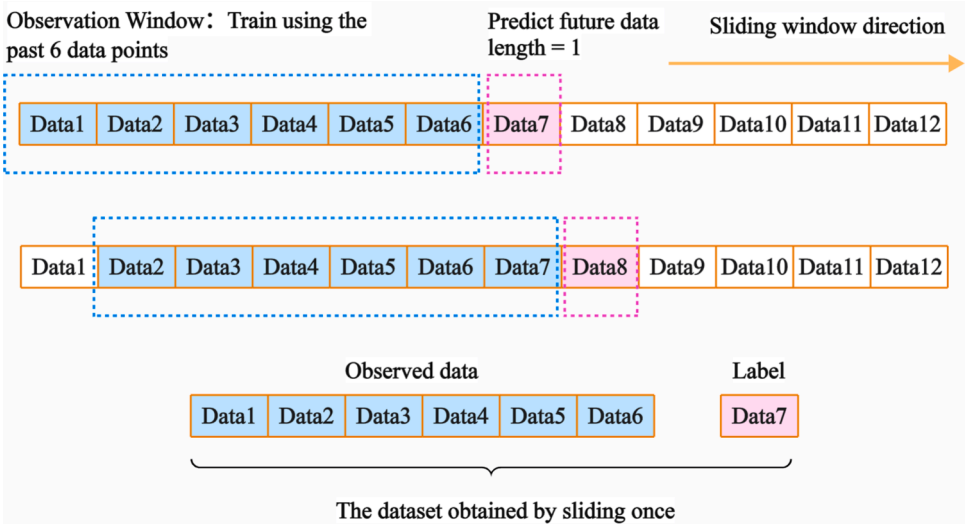

Fig. 6. The workflow of Sliding window splitter.

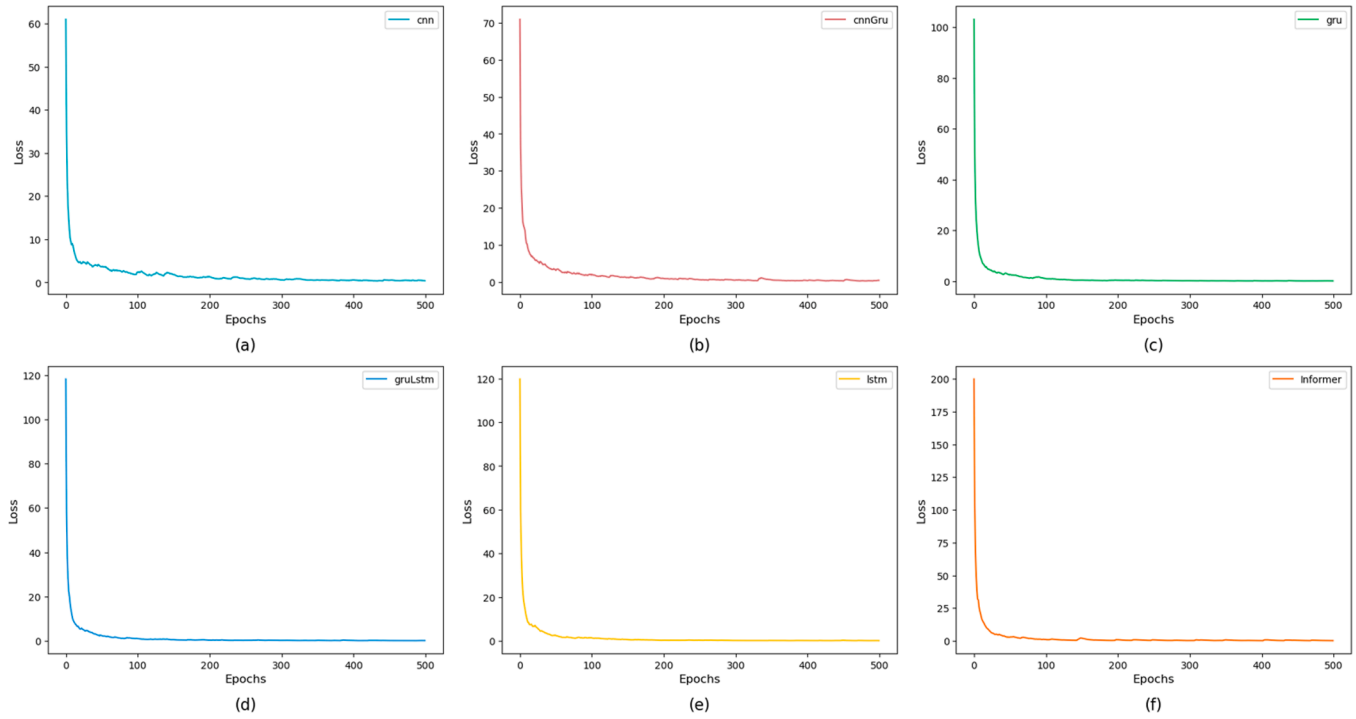

Fig. 7. Training loss plots for different models.

cycles. After 150 training epochs, the loss value has stabilized, indicating that the model has converged, and further increasing the number of epochs will not lead to significant improvements in model performance. Although the training curve becomes relatively stable after 100 epochs, in some cases, the optimal performance of the Informer model may be achieved at a higher number of epochs. Therefore, the optimal number of epochs for the Informer model may be after 150, which can be attributed to its reliance on long-term trends in the data. Finally, through the analysis of the above Loss plots, we set the optimized range for the number of training epochs between 20 and 300. A reasonable optimization range ensures computational efficiency, avoids overtraining, and allows the optimization algorithm to identify the optimal number of epochs that balance model complexity and training performance within a reasonable time frame.

It is worth noting that the loss plots of the aforementioned models are based on the initial configuration. When optimization algorithms are applied to adjust the hyperparameters, the model's performance may vary due to changes in other hyperparameters. Therefore, the plots above can only serve as a reference for preliminary analysis and should not be considered as the final results. Based on the researchers' experience, literature review, dataset analysis, and experimental validation, the optimization ranges for the four hyperparameters have been established, as presented in Table 4.

The performance of each model on the test set is shown in the Figs. 8, 9 and the Table 5.

According to Table 5, these six models demonstrated strong predictive performance on the test set, with R^2 values consistently exceeding 0.93, a maximum MAE of 22.48, and the highest MAPE at 11.7 %. In academic contexts, MAPE is generally deemed a robust predictive model when it is below 10 %. When MAPE ranges between 10 % and 20 %, the accuracy of the predictions can still be further optimized.

Fig. 10 provides a more intuitive visualization of the performance disparities among the models on the test set.

A high MDA value typically signifies that the model excels in capturing trend changes within the data, offering reliable predictive insights for researchers and developers. As oil well production data follows the law of declining production, the dataset demonstrates a pronounced downward pattern, which facilitates deep learning models in effectively capturing these trends. The MDA metric indicates that the aforementioned time-series models not only provide accurate numerical predictions but also demonstrate high accuracy in trend identification, which is crucial for decision-making in reservoir development.

On the other hand, because of the obvious trend in the oil well production data, the model may tend to rely on this trend, leading to a high MDA metric value, which makes it difficult to fully reflect the differences in predictive performance between the models. Therefore, it is necessary to combine other metrics for a comprehensive evaluation of the model's predictive accuracy. The best-performing model for each metric is marked with an asterisk ('★') in Fig. 10, and the results show that the Informer model outperforms all other models in every metric.

In comparison to other methods, superior performance across various metrics is demonstrated by the Informer model. Moreover, as shown in Table 6, the training time is considerably shorter than other models with the same number of epochs. Followed by GRU-LSTM, CNN-GRU, and GRU, while the CNN model exhibits larger prediction error.

From Table 7, observation reveals that the Informer model's residual mean is nearly zero and its residual standard deviation is minimal. This indicates that the model predicts without systematic bias, while also demonstrating high prediction accuracy.

In addition, Fig. 11 shows the distribution of residuals from various models, which can reveal the presence of systematic biases or specific trends in the model predictions. Ideally, the residuals in the figure should be randomly distributed around the center line without any identifiable patterns. If the residual plot shows a pattern, such as orderly fluctuations or a structured distribution, it suggests that the model may failed to capture certain influential variables affecting predictions or

Table 4
Optimization range of hyperparameters.

Hyperparameters	Min	Max
Time step	2	8
Epochs	20	300
Batch size	12	36
Learning rate	1e-5	1e-3

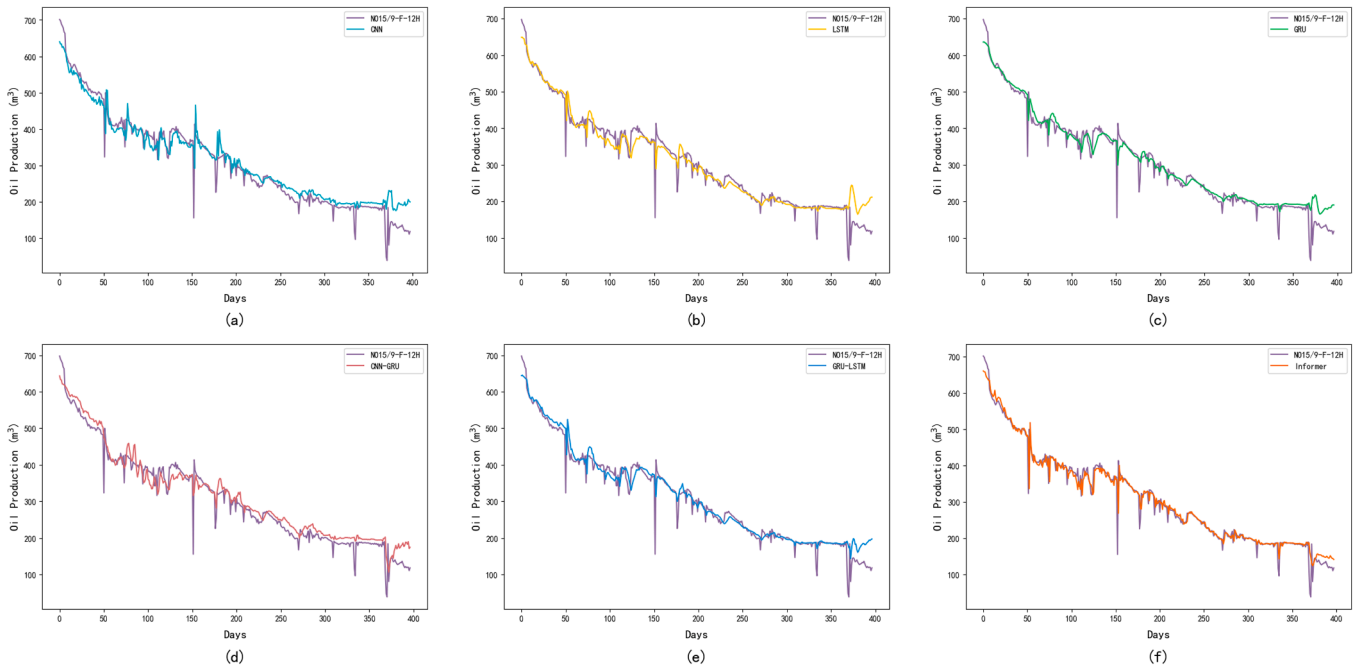

Fig. 8. Comparison between NO15/9-F-12H production data and prediction of the six models.

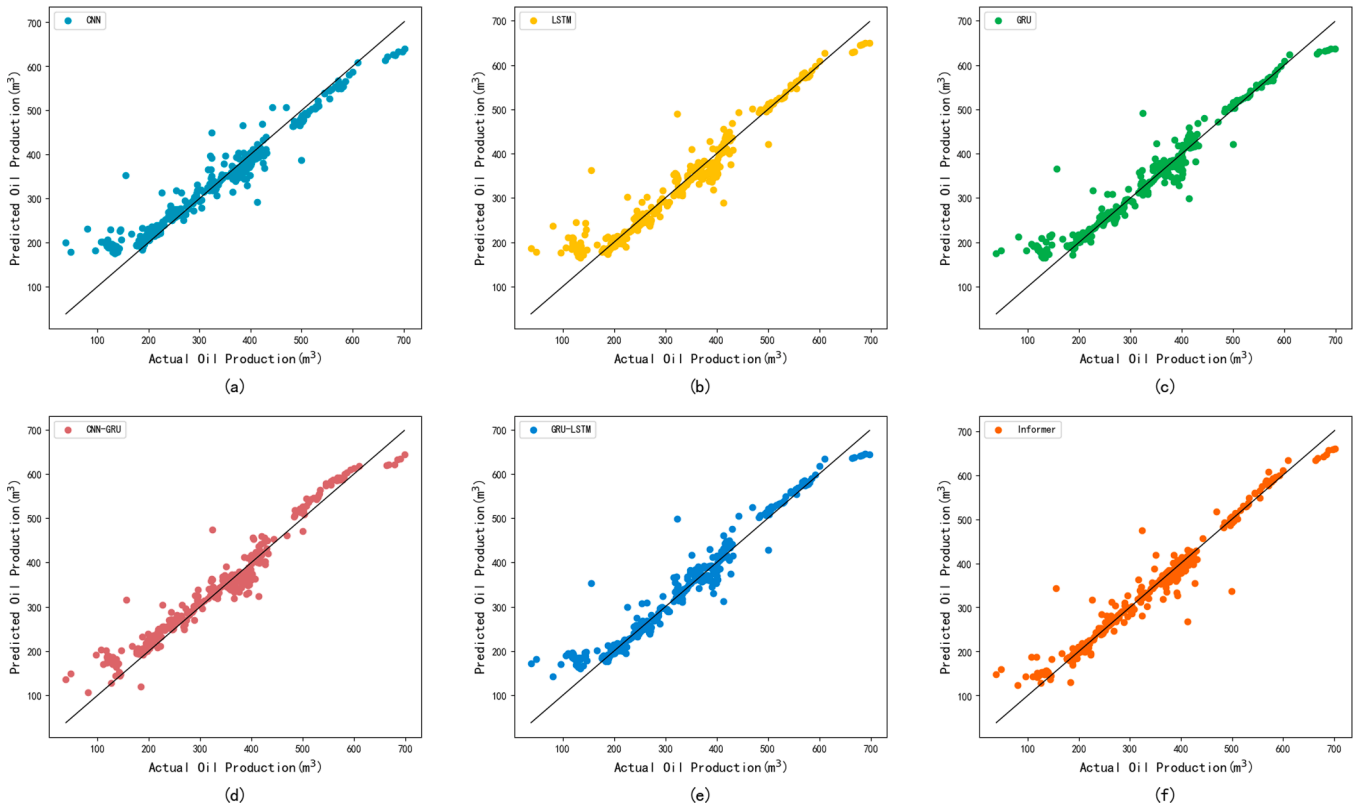

Fig. 9. Cross plot of the actual and predicted oil production of the six models.

exhibits underlying issues.

The research findings indicate that when displaying residuals over time, there is no apparent trend observed across all six models, demonstrating a random distribution. This means that the above models have no systematic biases during the prediction process, effectively captures the temporal characteristics and patterns of the data. Furthermore, the prediction errors are random rather than being caused by any uncaptured pattern or trend. Such characteristic further strengthens confidence in the models' integrity and robustness presented in this paper. The optimal hyperparameter combination of the models obtained using the BOHB algorithm is shown in Table 8.

Fig. 12 quantitatively analyzes the impact of hyperparameters on predictive performance.

The selection of the above parameters directly affects the model's

Table 5
Performance metrics for evaluating prediction results using the test sets of well NO15/9-F-12H.

Models	R^2 test	MAE	MAPE	SMAPE	MDA
CNN	0.933	22.48	11.7 %	9.2 %	99.492 %
LSTM	0.939	18.67	9.95 %	7.81 %	99.496 %
GRU	0.946	17.19	9.38 %	7.32 %	99.496 %
CNN-GRU	0.949	19.67	9.53 %	7.29 %	99.497 %
GRU-LSTM	0.952	16.64	8.67 %	6.82 %	99.574 %
Informer	0.974	12.4	6.05 %	4.98 %	99.623 %

learning and fitting ability of the data, subsequently impacting the precision of the prediction results. From Fig. 12, it can be identified which parameter adjustments significantly affect the model's performance improvement, allowing for the prioritization of adjustments and optimizations. For instance, this study can prioritize adjusting those parameters that exhibit the highest importance in the figure to achieve the maximum performance improvement.

In this case, it can be observed that the learning rate is the most critical hyperparameter, which directly affects the stability and convergence speed of the models. The time step determines the length of each input sequence, directly impacting the model's capacity to learn from historical data and accurately forecast future data. RNN models, as opposed to CNN models, focus more on capturing the temporal dependencies within sequential data, thus the configuration of the time step has a more profound impact. The number of epochs determines how often the entire training dataset is utilized by the model, whereas the batch size specifies the number of training samples used for each weight update. For complex structured mixture models, epochs are required to optimize model performance, while the batch size primarily influences training stability and time. Therefore, its impact may be slightly less significant compared to the number of epochs.

In this study, the single GRU model achieved an R^2 of 0.946 on the test set, demonstrating better performance when integrated into combination models with LSTM or CNN architecture. This indicates that compared to the single model, hybrid models demonstrate superior predictive performance in time series data. However, the final prediction outcomes of the hybrid model are heavily determined by the performance of the base component chosen. Therefore, further research and exploration are necessary to determine how to select models with better

Table 6
The training time across different epochs.

Models	$Epochs_{20}$	$Epochs_{50}$	$Epochs_{100}$	$Epochs_{200}$
CNN	5 m	12 m41 s	25 m11 s	48 m49 s
LSTM	5 m2 s	12 m39 s	25 m3 s	49 m46 s
GRU	5 m1 s	12 m20 s	24 m32 s	49 m1 s
CNN-GRU	5 m8 s	12 m44 s	25 m51 s	52 m9 s
GRU-LSTM	5 m16 s	12 m48 s	25 m20 s	54 m56 s
Informer	2 m10 s	5 m32 s	10 m56 s	28 m51 s

Table 7
Mean and standard deviation of residuals between predicted results from different models and actual values.

Models	Residual Mean	Residual Std
CNN	−7.61	33.12
LSTM	−2.33	33.07
GRU	−7.41	29.09
CNN-GRU	−10.58	26.55
GRU-LSTM	−6.83	27.77
Informer	−0.72	21.27

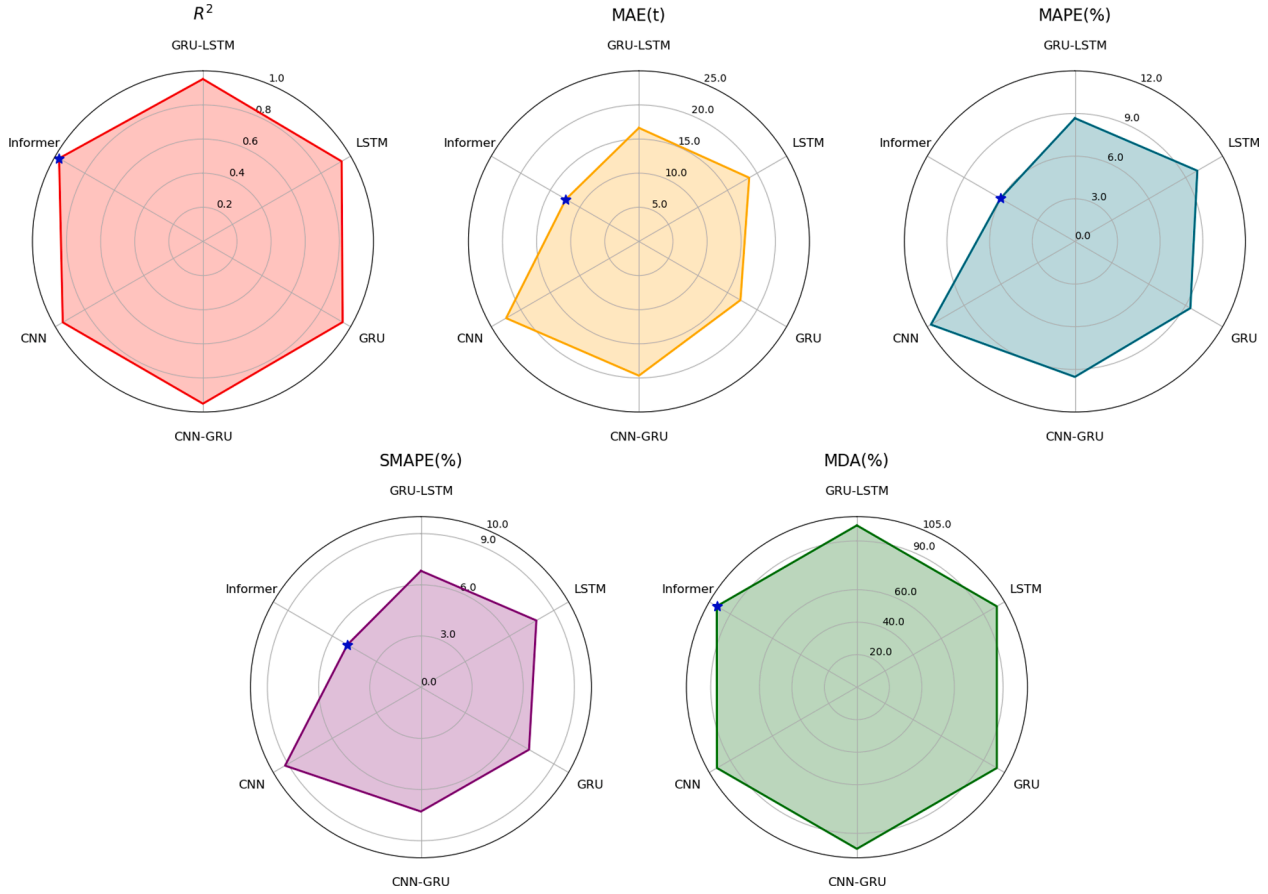

Fig. 10. The comparison of model performance on the test set.

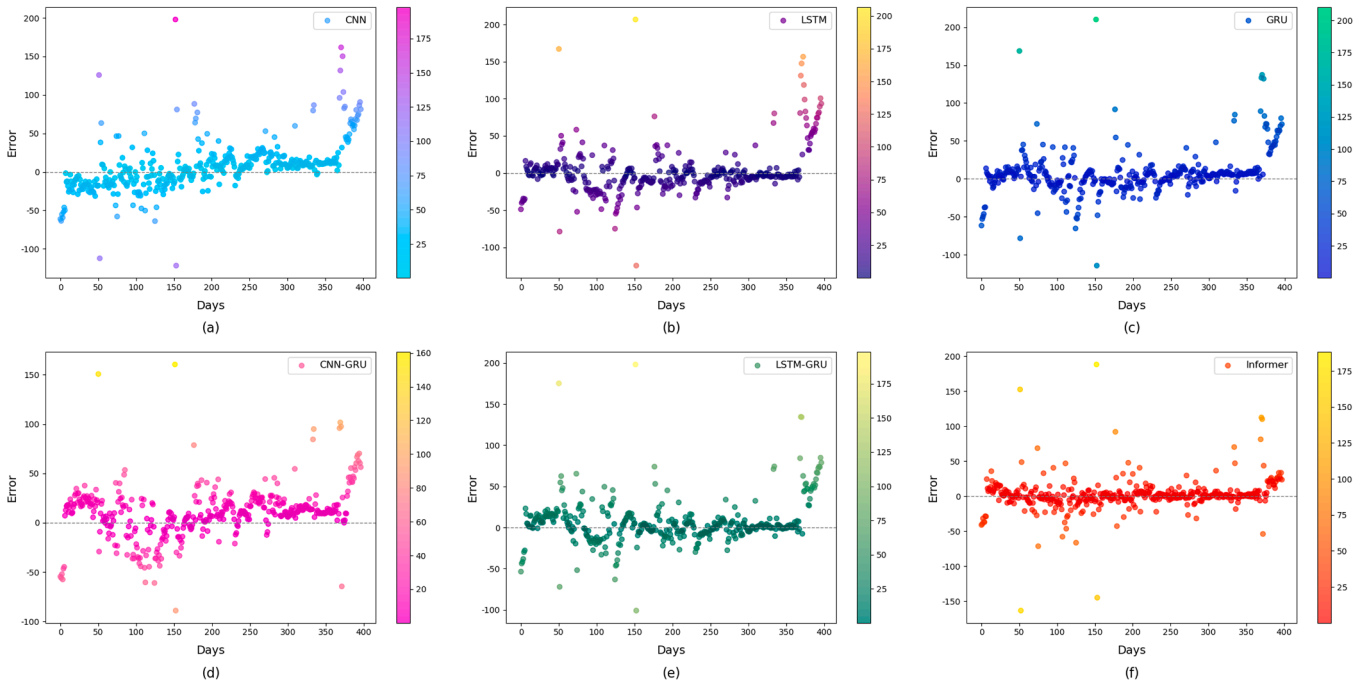

Fig. 11. Distribution of residuals.

Table 8
The best hyperparameters by BOHB.

Models	Time step	Epochs	Batch size	Learning rate
CNN	6	200	24	1e-4
LSTM	8	75	17	3.7e-5
GRU	6	76	25	3.3e-5
CNN-GRU	8	128	17	1e-5
GRU-LSTM	6	146	16	1.3e-5
Informer	6	177	24	1.4e-5

predictive performance that are suitable for the dataset as foundational components of hybrid models. Additionally, while the hybrid model demonstrated a certain degree of improvement in accuracy, but the performance improvement of the GRU model in practical applications is limited, and it also increases the training time.

4.2. Results analysis

The primary challenge in forecasting production from single wells lies in accurately predicting variations while trying to maintain the

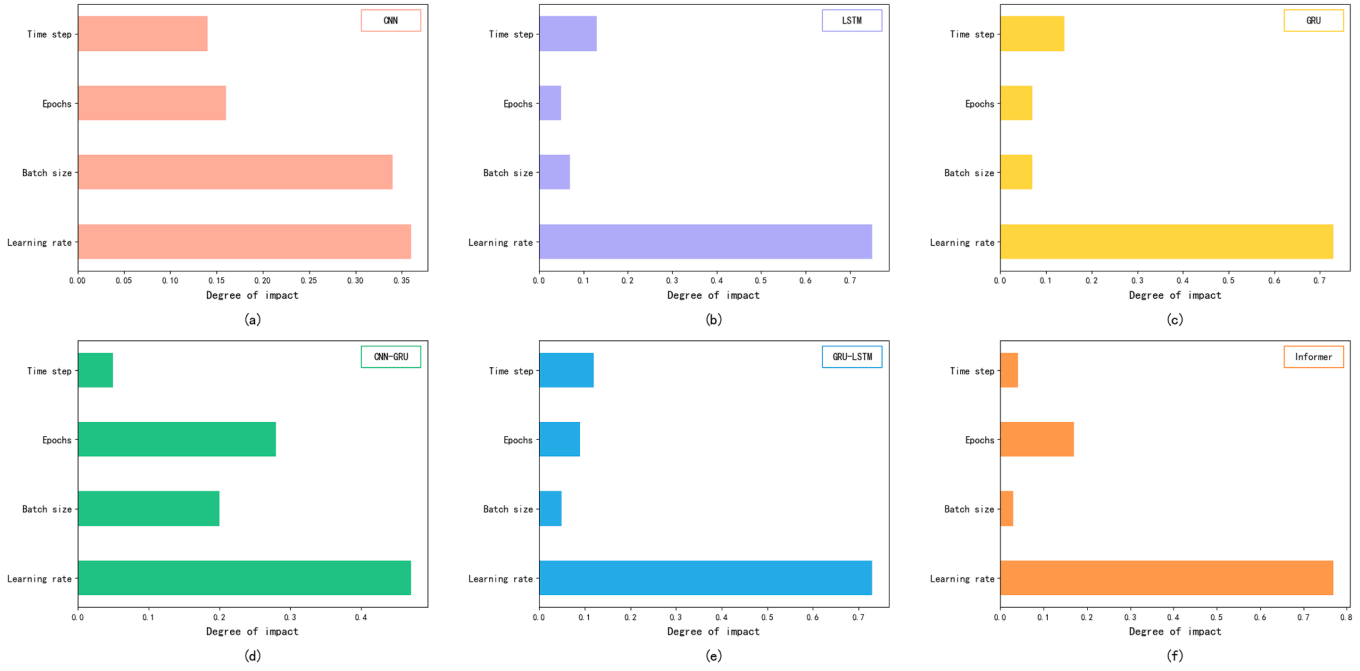

Fig. 12. Relative impact of hyperparameters on model predictive performance.

inherent fluctuations in the original data. The production data from single wells cannot be simply smoothed or directly processed for fluctuation values because fluctuations in production may result from adjustments in development plans and various external factors. This study aims to predict real-world well production patterns rather than conforming to a theoretical smooth decline curve. Compared to datasets exhibiting strong periodicity and regularity such as electricity and wind speed data, single-well production data primarily demonstrate declining trend and pronounced fluctuation. This implies that model performing well on other datasets may not necessarily be suitable for single well production datasets.

We attempted to use differencing transformation during the data preprocessing stage to make the original data stationary, thereby facilitating model predictions. However, experimental results show that for the specific characteristics of oil well production data, differencing did not yield the expected performance improvement, and the time series models exhibited relatively unstable performance on differenced data. The main reason may be related to the inherent characteristics of the data. While differencing can render the original data stationary, it simultaneously eliminates the inherent long-term trends within the dataset, which may impair the model's capacity to capture long-term dependencies and nonlinear characteristics. Directly using the original data or applying alternative preprocessing techniques may be more effective. Based on these considerations, we believe that differencing has limited improvement in this study and therefore has not been widely applied in the models. Such findings also suggest that different machine learning models will naturally exhibit distinct performance outcomes depending on the problem type, dataset characteristics, and feature selection. Even with the same model on the same dataset, slight modifications to parameters such as learning rate and the number of network layers can lead to substantially different prediction outcomes. Generally, it is necessary to conduct further research and validation using diverse datasets and various evaluation methodologies to comprehensively evaluate performance and practical applicability of models. Consequently, obtaining a substantial, high-quality, and multifaceted dataset of well production conducting sound feature engineering, efficient data processing appropriate model selection, and hyperparameter optimization are crucial for achieving accurate predictions.

The simplified structure of the GRU is more adept at capturing short-term dependencies in time series data, allowing it to more effectively utilize the characteristics of the data for modeling. Consequently, the GRU model performs significantly superior performance on the test set compared to the LSTM model, which is more proficient in handling long-term dependencies. The CNN model typically lacks memorization capabilities, which presents challenges to the effective capture of temporal dependencies in time series data. The structures and training methods of LSTM and GRU allow them to more precisely capture dynamic changes and trends in data, leading to better prediction over the CNN model in time series forecasting. CNN are particularly adept at capturing local correlations and features within data, and it also demonstrate high efficiency in processing datasets. The CNN-GRU model employs CNN to extract features, then feeds the feature sequence into GRU for prediction, which can enhance prediction accuracy to a certain extent. However, the primary advantage of CNN lies in handling data with spatial structure, while it is relatively weaker in capturing temporal dependencies for time series forecasting. Consequently, the performance of a CNN-GRU model in time series prediction may be inferior to the GRU-LSTM model.

From Fig. 8, it is evident that reference models such as LSTM, GRU, and GRU-LSTM exhibit a phase shift towards the right relative to the actual values in the interval of 120~140 Days. This suggests that these models may exhibit lag during this specific range. When production data shows declining or rising trends within the range of 100 to 150 days, it typically takes several time steps for the reference models to learn the dynamics in the data and make predictions. Furthermore, Fig. 8 indicate that the predictions of most reference models show significant discrepancies from the actual values in the 350 to 400 days interval, indicating higher prediction errors in the later stages.

When the time series data shows a sharp decline during the interval from 360 to 400 days, models frequently face challenges in promptly capturing the true changes in the data, instead tending to adhere to the previous trend of the time series. While the aforementioned models perform adequately on the test set, they exhibit issues such as lagging predictions in local intervals and higher errors in forecasting later periods. These issues undermine the reliability of the model predictions, necessitating analysis into the root causes. Firstly, from the data analysis of case studies, it is apparent that the standard deviation of production data from NO15/9-F-12H is large and non-stationary. While generally exhibiting a decreasing trend, there are significant fluctuations in certain ranges, which may hinder the model's ability to accurately capture these fluctuations, resulting in larger prediction errors within those ranges. Secondly, RNN-based models such as LSTM, GRU, and GRU-LSTM, when employed in a sequential manner, suffer from slower processing speeds and diminished effectiveness as the time series length increases in time series forecasting. Meanwhile, due to these models predicting the next feature based on previous features, they tend to employ the actual value from the previous time step as the prediction for the next. When the impact of historical values within a specific range lacks direct influence or when trends change frequently in a random manner, predictive outcomes may exhibit lagging effects. In contrast to RNN structures, where each time step's output relies on the current input and the previous time step's state, CNN extract features layer by layer from input data and are less sensitive to temporal sequence information. Consequently, CNN-based models generally exhibit less pronounced lag in testing data compared to models based on RNN structures. Adjusting the settings of the sliding window or inputting more enriched time-series features can be considered to mitigate the problem of lagging in time-series forecasting. Finally, discrepancies in data distribution and range between the training and test sets can potentially undermine the model's performance on the test set. In such cases, the prediction errors at each time step may be influenced by the errors from previous time steps, these errors can accumulate and propagate over time, resulting in higher prediction errors in the later stages.

According to the experimental results, the Informer model effectively mitigates lag problems in time series prediction while also promptly captures sharp fluctuations in historical production data, which demonstrates significant advantages in handling non-stationary long time series. Consequently, it can be observed that indiscriminately blending various models may not necessarily lead to improved predictive efficacy. Combining real-world data with model implementation principles is essential to adjust and optimize the model's structure, which can significantly improve its predictive performance.

5. Conclusion

In this work, a novel model based on the Transformer framework, called Informer, is successfully proposed for oil well forecasting with favorable outcomes. The Informer model was utilized for predicting oil production on a public dataset, and the results were compared against common time series forecasting models and hybrid models. From the research conducted, a range of primary conclusions can be established.

- The Informer model has been rigorously validated on the production dataset, and the results indicate that this model outperforms conventional time series forecasting approaches (CNN, LSTM, GRU) and the hybrid approaches (CNN-GRU, GRU-LSTM) on the same dataset, while also exhibits the most rapid training speed. Especially, the model can predict points of severe fluctuations in production data, and the common lagging issue in time series forecasting has also been alleviated.
- Compared to single models, the performance of hybrid models has shown a certain improvement. Experimental results demonstrated the feasibility of enhancing model prediction capabilities by

combining diverse units. However, while hybrid methods can enhance model performance to a limited extent, the inherent deficiencies of the base components persist also within the hybrid model.

- The selection of models as components in a hybrid model also requires careful consideration because the specific roles of each component within the hybrid model may be ambiguous, leading to potential random biases in the final prediction outcomes. Moreover, different models exhibit varying sensitivities to hyperparameters. It is essential to analyze the characteristics of each model and prioritize the optimization of high-impact hyperparameters to effectively enhance the model's predictive performance.
- The BOHB algorithm is suitable for deep learning tasks. By combining Bayesian optimization with the hyperband algorithm to efficiently find the best hyperparameter configuration with fewer experiments, significantly improving the efficiency of hyperparameter optimization.

In this study, the proposed model is capable of predicting oil well production based solely on historical production data. Based on production forecasting outcomes, engineers can evaluate the effectiveness of past reservoir development and refine the plans for reservoir development to ensure maximum economic gains.

Additionally, the introduced model exhibits the following shortcomings that need to be optimized and addressed through various methods in the future.

1) The introduced Informer model exhibits higher complexity and potential loss of critical information during the distillation process, which requires further optimization.

   To address this issue, this study considers employing various metaheuristic algorithms to optimize the hyperparameters of the Informer model, such as the number of layers in the encoder and decoder, and the number of hidden units. These algorithms are particularly suitable for dynamic environments or time-varying data distributions, effectively balancing model complexity and generalization ability. Additionally, modal decomposition methods such as EMD, VMD, and EEMD can be employed to eliminate the original noise from data and extract important information.
2) This study utilized data exclusively from Volve, which shows a distinct decreasing trend after initial processing, and Informer is particularly effective in handling data that exhibits pronounced patterns. Informer's performance may not necessarily outperform other models when encountering significant distribution drift in data. Therefore, to further validate the effectiveness and practical application of the Informer model in forecasting oil well production, which will be necessary to involve testing and evaluating the model with time-series data from various oil fields in future research.

CRediT authorship contribution statement

Wu Deng: Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis, Data curation. **Xiankang Xin:** Writing – review & editing, Supervision, Methodology. **Ruixuan Song:** Writing – review & editing, Validation, Supervision. **Xinzhou Yang:** Supervision, Methodology. **Weifeng Wang:** Supervision, Methodology. **Gaoming Yu:** Writing – review & editing, Supervision, Methodology.

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.

Acknowledgements

This research was funded by the National Natural Science Foundation of China, grant number 52104020.

Data availability

Dataset related to this paper is available and can be found at https://www.equinor.com/en/what-we-do/digitalisation-in-our-dna/volve-field-data-village-download.html. This open-source online dataset is provided by Equinor for research, study, and development purposes.

Equinor has granted all academic institutions, students, and researcher permission to use the dataset under the Equinor Open Data License, without requiring further written permission.

References

Al-qaness, M.A.A., Ewees, A.A., Abualigah, L., AlRassas, A.M., Thanh, H.V., Abd Elaziz, M., 2022. Evaluating the applications of dendritic neuron model with metaheuristic optimization algorithms for crude-oil-production forecasting. Entropy 24 (11). https://doi.org/10.3390/e24111674. Article 11.

Anderson, O.D., Box, G.E.P., Jenkins, G.M., 1978. Time series analysis: forecasting and control. Statistician 27 (3/4), 265. https://doi.org/10.2307/2988198.

Arps, J.J., 1945. Analysis of decline curves. Trans. AIME 160 (01), 228–247. https://doi.org/10.2118/945228-G.

Cao, C., Jia, P., Cheng, L., Jin, Q., Qi, S., 2022. A review on application of data-driven models in hydrocarbon production forecast. J. Petrol. Sci. Eng. 212, 110296. https://doi.org/10.1016/j.petrol.2022.110296.

Cao, Y., Liu, S., Cao, X., Liu, X., Hu, H., Zhang, T., Yu, L., 2022. EMD-based multi-algorithm combination model of variable weights for oil well production forecast. Energy Rep. 8, 13389–13398. https://doi.org/10.1016/j.egyr.2022.09.140.

Chang, J., Zhang, D., Li, Y., Lv, W., Xiao, Y., 2023. Physics-constrained sequence learning with attention mechanism for multi-horizon production forecasting. Geoenergy Sci. Eng. 231, 212388. https://doi.org/10.1016/j.geoen.2023.212388.

Chen, G., Tian, H., Xiao, T., Xu, T., Lei, H., 2024. Time series forecasting of oil production in enhanced oil recovery system based on a novel CNN-GRU neural network. Geoenergy Sci. Eng. 233, 212528. https://doi.org/10.1016/j.geoen.2023.212528.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder–Decoder for statistical machine translation. In: Moschitti, A., Pang, B., Daelemans, W. (Eds.), Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, pp. 1724–1734. https://doi.org/10.3115/v1/D14-1179.

Davtyan, A., Rodin, A., Muchnik, I., Romashkin, A., 2020. Oil production forecast models based on sliding window regression. J. Petrol. Sci. Eng. 195, 107916. https://doi.org/10.1016/j.petrol.2020.107916.

Falkner, S., Klein, A., & Hutter, F. (2018). *BOHB: robust and efficient hyperparameter optimization at scale* (arXiv:1807.01774). arXiv. https://doi.org/10.48550/arXiv.1807.01774.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural. Comput. 9 (8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.

Huang, Y., Deng, Y., 2021. A new crude oil price forecasting model based on variational mode decomposition. Knowl. Based. Syst. 213, 106669. https://doi.org/10.1016/j.knosys.2020.106669.

Iwana, B.K., Uchida, S., 2021. An empirical survey of data augmentation for time series classification with neural networks. PLoS One 16 (7), e0254841. https://doi.org/10.1371/journal.pone.0254841.

Jongkittinarukorn, K., Last, N., Escobar, F.H., Srisuriyachai, F., 2021. A straight-line DCA for a gas reservoir. J. Petrol. Sci. Eng. 201, 108452. https://doi.org/10.1016/j.petrol.2021.108452.

Kazemi, H., Merrill Jr., L.S., Porterfield, K.L., Zeman, P.R, 1976. Numerical simulation of water-oil flow in naturally fractured reservoirs. Soc. Petrol. Eng. J. 16 (06), 317–326. https://doi.org/10.2118/5719-PA.

Khanamiri, H.H., 2010. A non-iterative method of decline curve analysis. J. Petrol. Sci. Eng. 73 (1), 59–66. https://doi.org/10.1016/j.petrol.2010.05.007.

Kong, X., Liu, Y., Xue, L., Li, G., Zhu, D., 2023. A hybrid oil production prediction model based on artificial intelligence technology. Energies. (Basel) 16 (3). https://doi.org/10.3390/en16031027. Article 3.

Lea, C., Flynn, M.D., Vidal, R., Reiter, A., Hager, G.D., 2017. Temporal convolutional networks for action segmentation and detection. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1003–1012. https://doi.org/10.1109/CVPR.2017.113.

Lecun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2324. https://doi.org/10.1109/5.726791. Proceedings of the IEEE.

Li, W., Wang, L., Dong, Z., Wang, R., Qu, B., 2022. Reservoir production prediction with optimized artificial neural network and time series approaches. J. Petrol. Sci. Eng. 215, 110586. https://doi.org/10.1016/j.petrol.2022.110586.

Li, X., Ma, X., Xiao, F., Wang, F., Zhang, S., 2020. Application of gated recurrent unit (GRU) neural network for smart batch production prediction. Energies (Basel) 13 (22). https://doi.org/10.3390/en13226121. Article 22.

Li, X., Ma, X., Xiao, F., Xiao, C., Wang, F., Zhang, S., 2022. Time-series production forecasting method based on the integration of bidirectional gated recurrent unit (bi-GRU) network and sparrow search algorithm (SSA). J. Petrol. Sci. Eng. 208, 109309. https://doi.org/10.1016/j.petrol.2021.109309.

Lin, T., Wang, Y., Liu, X., & Qiu, X. (2021). *A survey of transformers* (arXiv:2106.04554). arXiv. https://doi.org/10.48550/arXiv.2106.04554.

Liu, W., Liu, W.D., Gu, J., 2020. Forecasting oil production using ensemble empirical model decomposition based long short-term memory neural network. J. Petrol. Sci. Eng. 189, 107013. https://doi.org/10.1016/j.petrol.2020.107013.

Ma, X., Hou, M., Zhan, J., Zhong, R., 2023. Enhancing production prediction in Shale gas reservoirs using a hybrid gated recurrent unit and multilayer perceptron (GRU-MLP) model. Appl. Sci. 13 (17). https://doi.org/10.3390/app13179827. Article 17.

Martínez, V., Rocha, A., 2023. The Golem: A general data-driven model for oil & gas forecasting based on recurrent neural networks. IEEE Access. 11, 41105–41132. https://doi.org/10.1109/ACCESS.2023.3269748.

Mohd Razak, S., Cornelio, J., Cho, Y., Liu, H.-H., Vaidya, R., Jafarpour, B., 2022. Transfer learning with recurrent neural networks for long-term production forecasting in unconventional reservoirs. SPE J. 27 (04), 2425–2442. https://doi.org/10.2118/209594-PA.

Negash, B.M., Yaw, A.D., 2020. Artificial neural network based production forecasting for a hydrocarbon reservoir under water injection. Petrol. Expl. Dev. 47 (2), 383–392. https://doi.org/10.1016/S1876-3804(20)60055-6.

Ng, C.S.W., Jahanbani Ghahfarokhi, A., Nait Amar, M., 2022. Well production forecast in volve field: application of rigorous machine learning techniques and metaheuristic algorithm. J. Petrol. Sci. Eng. 208, 109468. https://doi.org/10.1016/j.petrol.2021.109468.

Ng, C.S.W., Nait Amar, M., Jahanbani Ghahfarokhi, A., Imsland, L.S., 2023. A survey on the application of machine learning and metaheuristic algorithms for intelligent proxy modeling in reservoir simulation. Comput. Chem. Eng. 170, 108107. https://doi.org/10.1016/j.compchemeng.2022.108107.

Ning, Y., Kazemi, H., Tahmasebi, P., 2022. A comparative machine learning study for time series oil production forecasting: ARIMA, LSTM, and prophet. Comput. Geosci. 164, 105126. https://doi.org/10.1016/j.cageo.2022.105126.

Niu, W., Lu, J., Zhang, X., Sun, Y., Zhang, J., Cao, X., Li, Q., Wu, B., 2023. Time series modeling for production prediction of shale gas wells. Geoenergy Sci. Eng. 231, 212406. https://doi.org/10.1016/j.geoen.2023.212406.

Nwaobi, Ukadike, Anandarajah, Gabrial, 2018. Parameter determination for a numerical approach to undeveloped shale gas production estimation: the UK Bowland shale region application. J. Nat. Gas. Sci. Eng. 58, 80–91. https://doi.org/10.1016/j.jngse.2018.07.024.

Pan, S.J., Yang, Q., 2010. A survey on transfer learning. IEEE Trans. Knowl. Data Eng. 22 (10), 1345–1359. https://doi.org/10.1109/TKDE.2009.191. IEEE Transactions on Knowledge and Data Engineering.

Paullada, A., Raji, I.D., Bender, E.M., Denton, E., Hanna, A., 2021. Data and its (dis) contents: A survey of dataset development and use in machine learning research. Patterns 2 (11), 100336. https://doi.org/10.1016/j.patter.2021.100336.

Prasetyo, J.N., Setiawan, N.A., Adji, T.B., 2022. Forecasting oil production flowrate based on an improved backpropagation high-order neural network with empirical mode decomposition. Processes 10 (6). https://doi.org/10.3390/pr10061137. Article 6.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning representations by back-propagating errors. Nature 323 (6088), 533–536. https://doi.org/10.1038/323533a0.

Schmidhuber, J., 2015. Deep learning in neural networks: an overview. Neural Netw. 61, 85–117. https://doi.org/10.1016/j.neunet.2014.09.003.

Sheikhoushaghi, A., Gharaei, N.Y., Nikoofard, A., 2022. Application of rough neural network to forecast oil production rate of an oil field in a comparative study. J. Petrol. Sci. Eng. 209, 109935. https://doi.org/10.1016/j.petrol.2021.109935.

Song, X., Liu, Y., Xue, L., Wang, J., Zhang, J., Wang, J., Jiang, L., Cheng, Z., 2020. Time-series well performance prediction based on Long short-Term memory (LSTM) neural network model. J. Petrol. Sci. Eng. 186, 106682. https://doi.org/10.1016/j.petrol.2019.106682.

Sun, J., Ma, X., & Kazi, M. (2018). Comparison of decline curve analysis DCA with recursive neural networks RNN for production forecast of multiple wells. *Day 4 Wed, April 25, 2018*, D041S012R009. https://doi.org/10.2118/190104-MS.

Tadjer, A., Hong, A., Bratvold, R.B., 2021. Machine learning based decline curve analysis for short-term oil production forecast. Energy Expl. Exploit. 39 (5), 1747–1769. https://doi.org/10.1177/01445987211011784.

Taylor, S., & Letham, B. (2017). *Forecasting at scale*. https://doi.org/10.7287/peerj.preprints.3190v2.

Tu, B., Bai, K., Zhan, C., Zhang, W., 2024. Real-time prediction of ROP based on GRU-Informer. Sci. Rep. 14 (1), 2133. https://doi.org/10.1038/s41598-024-52261-7.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., & Polosukhin, I. (2023). *Attention is all you need* (arXiv:1706.03762). arXiv. https://doi.org/10.48550/arXiv.1706.03762.

Wang, Lei, Wang, Shihao, Zhang, Ronglei, 2017. Review of multi-scale and multi-physical simulation technologies for shale and tight gas reservoirs. J. Nat. Gas. Sci. Eng. 37, 560–578. https://doi.org/10.1016/j.jngse.2016.11.051.

Wen, S., Wei, B., You, J., He, Y., Xin, J., Varfolomeev, M.A., 2023. Forecasting oil production in unconventional reservoirs using long short term memory network coupled support vector regression method: A case study. Petroleum 9 (4), 647–657. https://doi.org/10.1016/j.petlm.2023.05.004.

Williams, R.J., Zipser, D., 1989. A learning algorithm for continually running fully recurrent neural networks. Neural Comput. 1 (2), 270–280. https://doi.org/10.1162/neco.1989.1.2.270. Neural Computation.

Xu, Z., Leung, J.Y., 2024. A novel formulation of RNN-based neural network with real-time updating – an application for dynamic hydraulic fractured shale gas production forecasting. Geoenergy Sci. Eng. 233, 212491. https://doi.org/10.1016/j.geoen.2023.212491.

Yan, H., Liu, M., Yang, B., Yang, Y., Ni, H., Wang, H., 2024. Short-term forecasting approach of single well production based on multi-intelligent agent hybrid model. PLoS One 19 (4), e0301349. https://doi.org/10.1371/journal.pone.0301349.

Zhang, L., Dou, H., Zhang, K., Huang, R., Lin, X., Wu, S., Zhang, R., Zhang, C., Zheng, S., 2023. CNN-LSTM model optimized by bayesian optimization for predicting single-well production in water flooding reservoir. Geofluids 2023, 1–16. https://doi.org/10.1155/2023/5467956.

Zhen, Y., Fang, J., Zhao, X., Ge, J., Xiao, Y., 2022. Temporal convolution network based on attention mechanism for well production prediction. J. Petrol. Sci. Eng. 218, 111043. https://doi.org/10.1016/j.petrol.2022.111043.

Zhou, G., Guo, Z., Sun, S., Jin, Q., 2023. A CNN-BiGRU-AM neural network for AI applications in shale oil production prediction. Appl. Energy 344, 121249. https://doi.org/10.1016/j.apenergy.2023.121249.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). *Informer: beyond efficient transformer for long sequence time-series forecasting* (arXiv: 2012.07436). arXiv. https://doi.org/10.48550/arXiv.2012.07436.