energies

MDPI

Article

Research on Oil Well Production Prediction Based on GRU-KAN Model Optimized by PSO

Bo Qiu, Jian Zhang *, Yun Yang, Guangyuan Qin, Zhongyi Zhou and Cunrui Ying

School of Computer Science and Software Engineering, Southwest Petroleum University, Chengdu 610500, China;
* Correspondence: zhangjian@swpu.edu.cn

Abstract: Accurately predicting oil well production volume is of great significance in oilfield production. To overcome the shortcomings in the current study of oil well production prediction, we propose a hybrid model (GRU-KAN) with the gated recurrent unit (GRU) and Kolmogorov–Arnold network (KAN). The GRU-KAN model utilizes GRU to extract temporal features and KAN to capture complex nonlinear relationships. First, the MissForest algorithm is employed to handle anomalous data, improving data quality. The Pearson correlation coefficient is used to select the most significant features. These selected features are used as input to the GRU-KAN model to establish the oil well production prediction model. Then, the Particle Swarm Optimization (PSO) algorithm is used to enhance the predictive performance. Finally, the model is evaluated on the test set. The validity of the model was verified on two oil wells and the results on well F14 show that the proposed GRU-KAN model achieves a Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and Coefficient of Determination (R^2) values of 11.90, 9.18, 6.0% and 0.95, respectively. Compared to popular single and hybrid models, the GRU-KAN model achieves higher production-prediction accuracy and higher computational efficiency. The model can be applied to the formulation of oilfield-development plans, which is of great theoretical and practical significance to the advancement of oilfield technology levels.

Keywords: oil well production prediction; GRU-KAN model; Particle Swarm Optimization algorithm; MissForest

check for updates

Citation: Qiu, B.; Zhang, J.; Yang, Y.; Qin, G.; Zhou, Z.; Ying, C. Research on Oil Well Production Prediction Based on GRU-KAN Model Optimized by PSO. *Energies* **2024**, *17*, 5502. https://doi.org/10.3390/en17215502

Academic Editors: Marcin Blachnik and Grzegorz Dudek

Received: 29 September 2024
Revised: 26 October 2024
Accepted: 29 October 2024
Published: 4 November 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

In the process of oil field development, the production capacity of a single well or oil field is crucial for the economical and efficient development of that oil field. The production capacity of an oil field is determined by the output of its internal wells. Therefore, accurately predicting well production has always been a key focus of research in the oil industry [1]. By predicting oil well production, researchers can develop more rational extraction plans, ensure scientific production, optimize production processes, and have significant implications for the coordinated development of the oil and gas industry and investment decision-making for enterprises [2].

Traditional oil well production-prediction methods are mainly divided into two kinds: mathematical analysis and numerical simulation. Mathematical analysis is a statistical analysis method mainly based on the historical law of reservoir development, such as the method of Decline Curve Analysis and Water Drive Decline Curve, and these techniques have been widely used in oilfield production [3–5]. However, this curve fitting method is based on many assumptions and experiences, while oil well production data often exhibit highly nonlinear characteristics, which these statistical methods struggle to handle. As a result, it typically adapts poorly and achieves low calculation accuracies [6]. Numerical simulation methods are currently the most commonly used methods for predicting oil production [7–9]. This method is based on the understanding of the real flow process of the underground porous medium, utilizing physics-driven data analysis and takes into

account more factors, thereby predicting the results more objectively. Nonetheless, the computational process is limited by the development of a geologic reservoir model, which requires a laborious and time-consuming process that includes developing the geologic model, numerical modeling and performing history fitting. Each model must be of high quality to accurately predict oil well production.

Machine learning, as the mainstream direction of artificial intelligence, especially the rapid progress of deep learning, has brought new technologies and methods to guide oilfield production. In recent years, more and more petroleum enterprises have been building smart oilfields through machine learning methods to improve quality and efficiency. It has been applied in a number of oil-related works and has produced good results [10], such as seismic interpretation [11,12], pipeline leakage [13,14] and drilling-optimization analysis [15,16]. In oil well production prediction, many traditional machine learning methods (such as support vector machine, artificial neural network and random forest) have been widely applied [17–19], but most have focused on regression methods and paid little attention to the time series characteristics of production data. Production data are a typical time series structure, influenced by many internal and external factors. Consequently, many researchers utilize time series methods to predict future oil well production. Temporal Convolutional Networks (TCNs) [20], Recurrent Neural Networks (RNNs) [21], specifically Long Short-Term Memory (LSTM) [22] and Gated Recurrent Unit (GRU) [23] have emerged as prominent focal points in production prediction. Zhang et al. [24] proposed using TCN to predict the production of water flooding oilfield wells, achieving high prediction accuracy. Indrajeet Kumar et al. [25] integrated the LSTM algorithm with a genetic algorithm (GA) to predict oil well production, demonstrating that the GA-LSTM model surpasses other production-prediction models; Raghad et al. [26] proposed a GRU model for petroleum production prediction, which features a low-complexity architecture and the capability to track long-interval time series data, demonstrating excellent performance.

Although machine learning methods have made significant progress in oil well production prediction, there is still room for improvement. The single RNNs show low accuracy in time series prediction and struggle to fully capture input data characteristics. To enhance prediction accuracy, researchers have combined RNNs with other methods to build hybrid models for time series. Convolutional neural networks (CNNs) [27] are one of the most representative network structures in deep learning, excelling in feature extraction and information mining compared to other methods. Zhang Lei et al. [28] introduced the CNN-LSTM method to predict oil well production, with the model's efficacy validated in actual reservoirs. Chen et al. [29] introduced the CNN-GRU method to predict oil well production and indicated that the proposed CNN-GRU model outperforms other prediction approaches. Recently, The Kolmogorov–Arnold Network (KAN) has presented a novel approach for time series prediction. Xu et al. [30] applied KAN in time series analysis, proving KAN's ability in capturing complex temporal dependencies. Based on this, this study proposes a GRU-KAN hybrid model to establish an oil well production-prediction model, where GRU excels at capturing long-term and short-term dependencies in time series and KAN can accurately capture the nonlinear relationships in oil well production affected by multiple complex factors through its flexible nonlinear feature-extraction capabilities. Compared with other models, this combination not only achieves better prediction accuracy but also reduces training time and improves computational efficiency, demonstrating potential applications in the oil industry. The contributions of this study are as follows:

(1) We propose the GRU-KAN model to address challenges in oil well production prediction and build a prediction model for each well using GRU-KAN.
(2) To evaluate the effectiveness of the GRU-KAN model, we introduced popular single and hybrid machine learning models for comparison, including LSTM, GRU, TCN, Bidirectional LSTM (BiLSTM) [31], Bidirectional GRU (BiGRU) [23], CNN-LSTM, and CNN-GRU. Evaluation results indicate that GRU-KAN outperforms the other methods in predicting oil well production accuracy and generalization ability.

(3) We also compared the effects of different data imputation methods and hyperparameter optimization methods on the prediction results, and the results show that using the MissForest and PSO algorithms outperforms traditional methods.

The remainder of this paper is organized as follows: In Section 2, we describe in detail the core algorithms used in this study, including GRU and KAN. We also explain how the Particle Swarm Optimization (PSO) algorithm determines the model's hyperparameters. Additionally, we introduce the MissForest algorithm, which is employed to handle outlier data. In Section 3, we provide a comprehensive overview of the experiments. First, we discuss the data-processing steps, which cover data cleaning, feature parameter analysis, data normalization, the design of the time window size and dataset splitting. Next, we describe the model training and prediction process, using various evaluation metrics to assess model performance. This section also compares the prediction results of single models and hybrid models, as well as the impact of different data-imputation methods and different hyperparameter optimization methods on model performance. In the end, we discuss the model's generalization ability. In Section 4, we summarize the contributions of this study and discuss potential future research directions.

2. Materials and Methods

2.1. Gated Recurrent Unit (GRU)

Gated Recurrent Unit (GRU) is a special type of recurrent neural network (RNN) specifically designed for processing and predicting sequence data. Traditional RNNs are prone to gradient vanishing or exploding when processing long-sequence data, resulting in the loss of long-range dependency information. GRU effectively solves this problem by introducing gating mechanisms. The core components of GRU are hidden states and two gating structures: update gate and reset gate. The reset gate determines how much past information should be retained when updating the hidden state. The update gate determines how much current information should be retained when updating the hidden state. Hidden states are used to store information for each time step in a sequence. Through these gate structures, GRU can selectively retain and discard information when processing long-time sequences, thereby capturing long-range dependencies. Compared with the Long Short Term Memory (LSTM) network, GRU combines the forget gate and input gate in LSTM into one update gate and eliminates independent cell states, with hidden states serving as both cell states. The structure is more simplified, making GRU's training more efficient than LSTM. The network structure is shown in Figure 1.

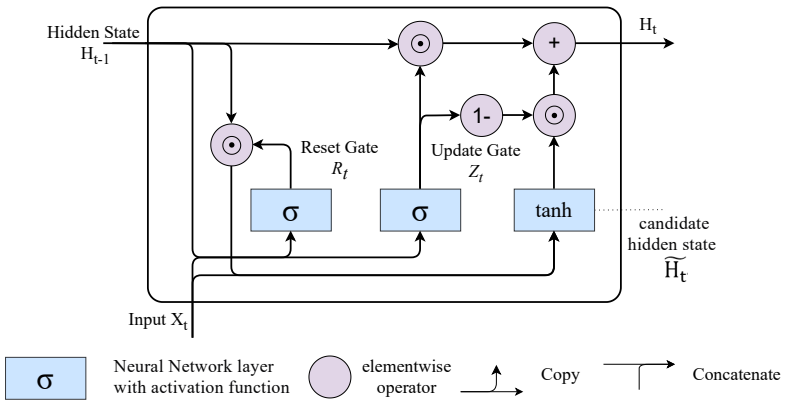


Figure 1. GRU network structure.

In Figure 1, t represents the time step; R_t represents the reset gate; Z_t represents the update gate; σ represents the activation function; tanh represents the hyperbolic tangent function; $\odot$ denotes element-wise multiplication; H_{t-1} and H_t represent the hidden layer

states at the previous and current moments, respectively; X_t represents the input at the current moment; $\tilde{H}_t$ represents the candidate hidden state at the current moment.

2.2. Kolmogorov–Arnold Network (KAN)

The Kolmogorov–Arnold Network (KAN) is a novel network architecture proposed by a team at MIT, inspired by the Kolmogorov–Arnold Theorem [32], as shown in Figure 2. In recent years, KAN's application in machine learning has garnered widespread attention. Its primary advantages lie in model flexibility and interpretability. Unlike traditional multilayer perceptrons (MLP), KAN does not have fixed activation functions but achieves this through learnable single-variable functions. This characteristic makes KAN excel in data fitting and complex function learning. KAN has advantages over MLP and CNN in terms of model performance, computational efficiency and interpretability.

Kolmogorov's theorem states that any continuous multivariable function can be represented as a superposition of a finite number of continuous single-variable functions. Specifically, for an n-dimensional continuous function $f(x_1, x_2, ..., x_n)$, there exists a set of continuous single-variable functions g_i and h_{ij} such that:

$$f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{2n+1} g_i \left(\sum_{j=1}^{n} h_{ij}(x_j) \right) \tag{1}$$

where n is the number or dimension of the input variables. g_i are the outer functions, which are continuous functions of a single variable. These functions operate on the sum of the outputs of the inner functions. $h_{ij}(x_i)$ are the inner functions, which are also continuous functions of a single variable. Each h_{ij} processes one input variable x_j and contributes to the sum for each outer function. Here, i ranges from 1 to $2n + 1$ and j ranges from 1 to n.

This theory indicates that complex multivariable functions can be represented through combinations of simpler single-variable functions, offering a new perspective for neural network design [33].

It can be seen from Figure 2 that the KAN architecture is composed of multiple layers, each containing several nodes and edges. Each node receives input signals from the previous layer and applies nonlinear transformations to these signals through learnable single-variable functions on the edges. In KAN, each weight parameter is represented as a single-variable function, the specific form of which can be adjusted during the training process. This design allows each layer's output to depend not only on the input signals but also on optimized parameters that adapt to different data patterns, thereby enhancing the model's expressive power.

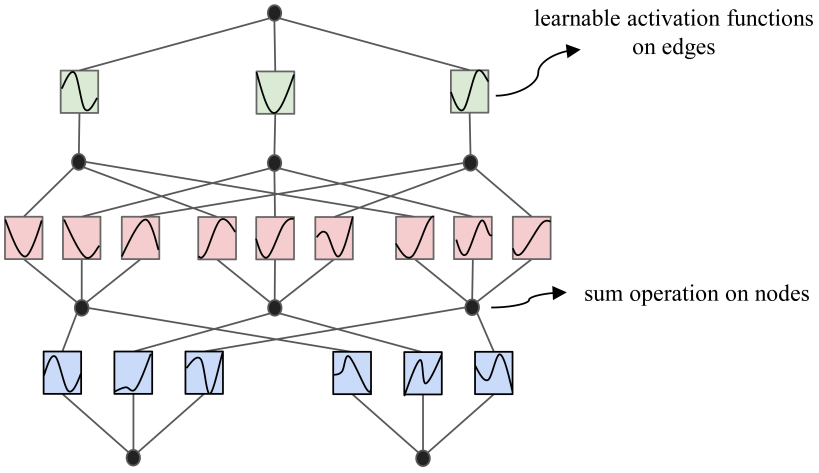


Figure 2. Kolmogorov–Arnold network structure.

2.3. GRU-KAN

The GRU-KAN proposed in this article mainly consists of four parts: input, GRU, KAN and output. The input layer receives oil well production data and processes them into data suitable for GRU input format. GRU consists of update gates, reset gates and candidate hidden state, which are used to process time series information in oil well-generated data and extract temporal features. The data processed by the GRU module are transmitted to the KAN layer. The KAN layer consists of multiple layers, each containing multiple nodes and edges and learns the feature through spline functions. Finally, the output layer outputs the final prediction result. The GRU-KAN combines the advantages of the GRU network in handling temporal data with the capabilities of the Kolmogorov–Arnold Network (KAN) in processing complex and nonlinear data, achieving modeling of complex time dependencies. This hybrid method takes full advantage of GRU's ability to process sequential data and KAN's strength in handling complex, nonlinear data. It is especially effective for tasks involving temporal analysis and intricate feature interactions. The structure of the GRU-KAN model is shown in Figure 3.

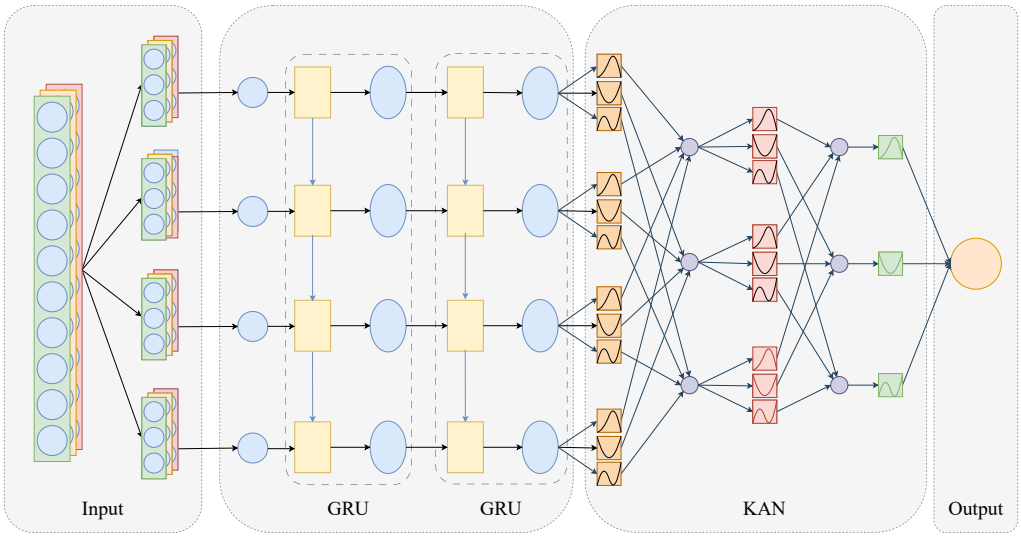


Figure 3. GRU-KAN network structure.

2.4. Particle Swarm Optimization Algorithm

Hyperparameter optimization algorithm in oil well production prediction can effectively improve model performance, save the time of parameter tuning and enhance the generalization ability of the model [34]. Due to the complex structure of the GRU-KAN model, it is important to set the hyperparameters in the model. The Particle Swarm Optimization (PSO) algorithm is a collective intelligence-driven optimization method, drawing inspiration from the cooperative behaviors observed in animal societies, like the coordinated movements of bird flocks and fish schools. The optimization problem is solved by simulating the foraging behavior of these groups [35]. The steps of PSO hyperparameter optimization are as follows:

(1) Initialize the parameters of the particle swarm, including the number of particles, the number of iterations, the initial value of the weight coefficient and the range of the search space for the learning factor. In this study, the number of particles is set to 20, the number of iterations is set to 100, the initial weight coefficient is set to 0.9, the learning factor is set to 2 and the range of the search space is determined based on the characteristics of the hyperparameters of each model.

(2) Import an oil well production-prediction model and define the fitness function for the particles. In this study, the root mean square error serves as the criterion for evaluating fitness.

(3) Compute the fitness values of the particles, rank them based on fitness and identify both the individual best value (pbest) and the global best value (gbest), recording them as their historical best positions.
(4) Update the velocity and position of the particles, including both the ordinary and optimal particles.
(5) Check if the particles meet the iteration-termination conditions. If the iteration limit is exceeded, substitute the optimal parameters into the model for training and prediction. If not, return to step (3) and continue until the termination criteria are fulfilled.

The optimization process of the entire network is shown in Figure 4.

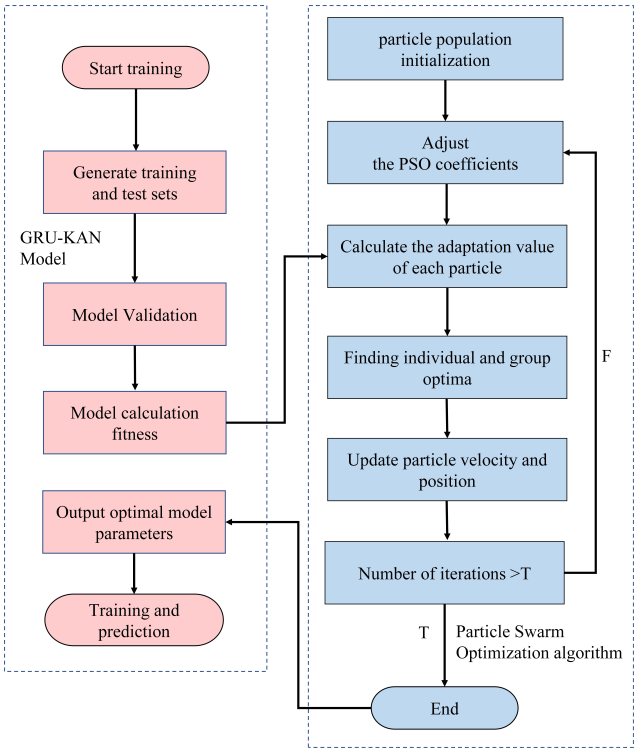


Figure 4. GRU-KAN network process based on PSO algorithm.

2.5. MissForest Algorithm

MissForest is a missing value imputation technique based on random forests [36]. It averages a series of uncut classification or regression trees to conduct deep learning-based research on predicting dynamic indicators of oilfield development and implement data interpolation. After extensive comparison, it was found that the MissForest method fully demonstrates excellent performance on various datasets and with different degrees of missing data. It is not only computationally efficient but also particularly suitable for handling missing values in high-dimensional datasets. Using MissForest for data imputation significantly reduces prediction errors. Compared with other regression imputation methods, it is easy to operate and can achieve good results even under default settings. In addition, compared with traditional data-imputation methods such as using 0, mean, median or mode imputation, MissForest has significantly improved accuracy, indicating its superiority in handling missing data.

3. Experiments

3.1. Data Processing

The experimental data were obtained from the Volve oilfield. The oilfield is located in the central North Sea and is mainly composed of sandstone formations. The oilfield maintains formation pressure through water injection to support production. Drilling in the oilfield began in 2006 and due to the long period of production, a substantial volume

of historical production data has been amassed at the oilfield. There are a total of seven wells in the oilfield, including five production wells (F1, F11, F12, F14, F15) and two water injection wells (F4, F5). We use the data from five of the production wells to create the dataset. Well F14 and F12 in this oilfield have long development time and obvious changing trends. These wells were selected for the oil-production prediction. The remaining three oil wells (F1, F11, F15) are used for model-generalization analysis to verify the model's versatility on new wells. The oil production of the five wells is shown in Figure 5.

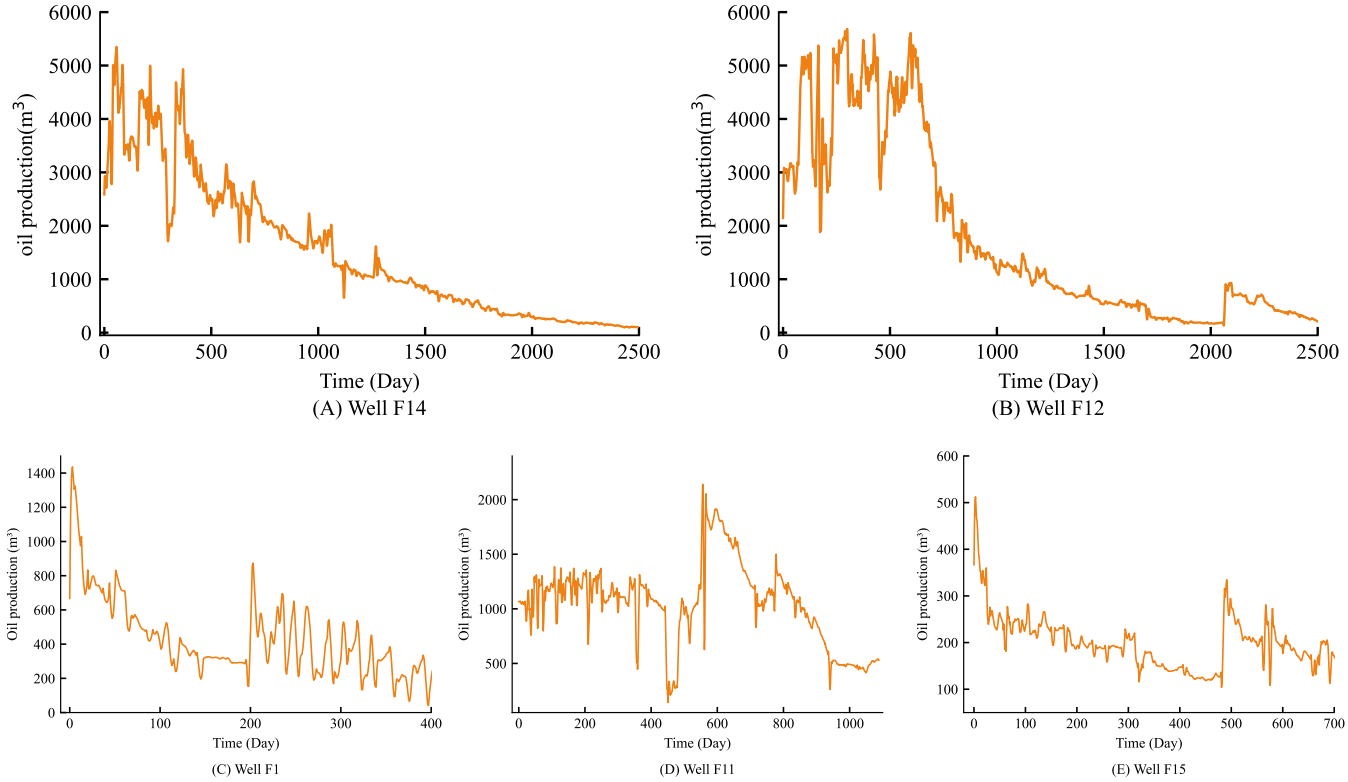


Figure 5. Oil production of five oil wells.

3.1.1. Data Cleaning

When important features contain outliers but cannot be discarded, data cleaning becomes a crucial step in building a high-quality dataset and appropriate data-filling logic is directly related to the model's predictive accuracy. For a few continuously missing features, a random forest model can be constructed between the input features and the features to be filled. Complete data are used as the training set for training and data with missing values are used for model prediction, with the predicted values serving as the filling values for the missing features. This study applies the MissForest algorithm to handle data missingness. MissForest is a specialized version designed specifically for dealing with missing data and compared to the random forest, it exhibits stronger robustness in dealing with noise and outliers, thereby improving the accuracy of the imputed data. For large sections of continuously missing data, we choose to delete the relevant rows or columns to handle them.

3.1.2. Feature Parameter Analysis

The Pearson correlation coefficient method is used to assess the correlation between features, select key features related to production and eliminate redundant ones, improving model performance while reducing complexity. The Pearson correlation coefficient measures the strength and direction of a linear relationship between two variables, ranging

from −1 to 1. A value close to ±1 indicates a strong linear relationship, while a value near 0 suggests a weak linear correlation. The formula is shown in Equation (2).

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \tag{2}$$

The correlation between the input features and the oil well production is shown in Figure 6. The correlation between wellhead pressure and choke size pressure differential is 0.99; the choke size pressure differential is selected for removal to reduce the complexity of model training.

After removing the parameters with low correlation to oil production, wellhead pressure, wellhead temperature, downhole pressure, tubing pressure differential, choke size percentage, annular pressure, gas production, water production and water cut were selected as the input features. The oil production was used as the label to construct the dataset.

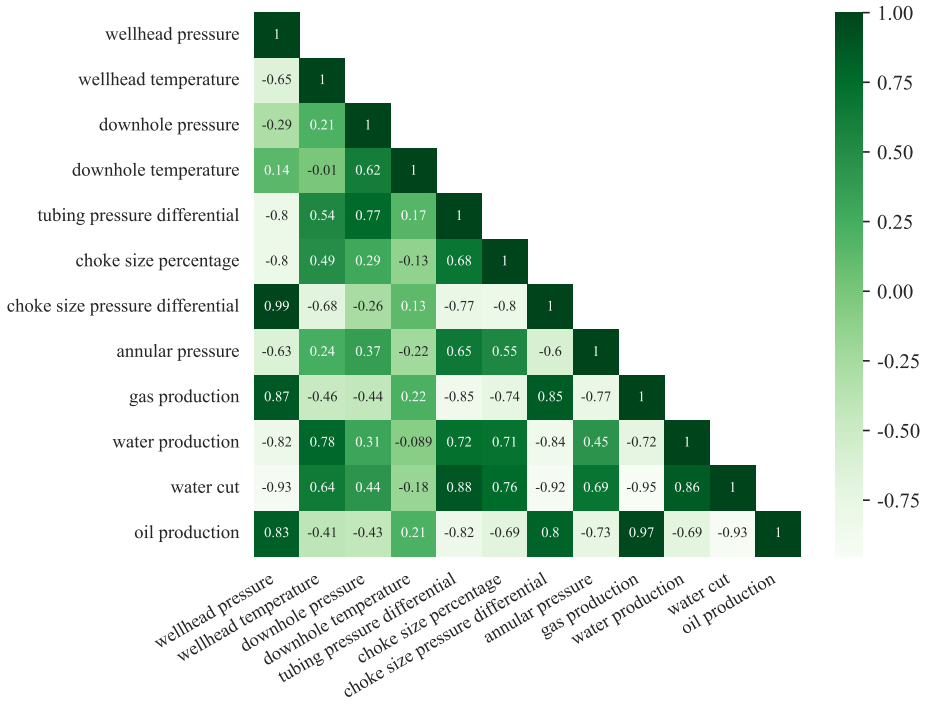


Figure 6. Well F14 input characteristics and production correlation.

3.1.3. Data Normalization

Data normalization is a crucial preprocessing step that significantly enhances the performance of oilfield production prediction. The attributes in oil well production data have varying units and their magnitudes can differ significantly. Attributes with larger magnitudes often dominate and impact the model's training convergence speed. To mitigate this issue, it is essential to normalize all attributes in the oil well production data. In this study, Z-Score standardization is used as the normalization method. It is a normalization technique that scales data to a standard scale. This method normalizes data by subtracting the mean and dividing it by the standard deviation, as shown in Equation (3).

$$Z = \frac{(X - \mu)}{\sigma} \tag{3}$$

where Z is the normalized value, X is the original data, μ is the mean of the data and σ is the standard deviation of the data.

3.1.4. Design of Data Window Size

Production data are a typical time series structure. To account for the temporal correlations in oil well production data, the input data format required for the model is constructed using a sliding window. If the size of the sliding window is set to n, the first sample input data consists of the data from the first to the nth time step, which is processed by the model and used to predict the $n+1$th data. The second sample input data slide the window back by one time step and consist of the data from the second to the $n+1$th time step, used to predict the $n+2$th data. Eventually, the entire time series is traversed in turn, forming an input format of (samples, sliding window, features).

3.1.5. Dataset Splitting

After preprocessing, 20% of the oil well production data are randomly selected as the test set and the remaining data are used as the training set.

3.2. Model Training and Prediction

3.2.1. Evaluation Metrics

The Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and the value of the coefficient of determination R^2 are selected as the evaluation indexes for the experiments. The closer the value of R^2 is to 1, the better the model fits the data and can account for more of the data variability. The smaller the MAE, RMSE and MAPE, the smaller the error between the model's predicted value and the actual observed value and the more accurate the model's prediction. The calculation formula is as follows.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{4}$$

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{5}$$

$$\mathrm{MAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{6}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{7}$$

where y_i is the actual value of oil production; $\hat{y}_i$ is the predicted value of oil production; and n is the total amount of oil production data in the test set.

3.2.2. Experimental Process

The experimental process of GRU-KAN for the production prediction of oil wells is as follows:

(1) Preprocess the oil well production data, including cleaning abnormal data and feature selection, and normalize the data.
(2) Reshape the oil well production data into a format suitable for model input. The data are then split into training and test sets in chronological order randomly with a ratio of 8:2.
(3) Construct the GRU-KAN model and perform iterative training on the model using the training set, with all hyperparameters optimized using the PSO method.
(4) Evaluate the model's performance using the test set and determine whether it meets the accuracy requirements. Save the model if the accuracy requirements are met; otherwise, adjust the model's hyperparameters using PSO and continue iterative training.
(5) After the iterations are complete, the model with the best performance is selected as the production prediction model for the oil well.

3.2.3. Comparison Between Single Model Predictions

First, we selected five single models for comparison, TCN, LSTM, GRU, BiLSTM and BiGRU. These models are popular for time series forecasting tasks and have been widely applied in the field of oil well production prediction. All model hyperparameters were optimized by PSO. The relevant hyperparameters of the models involved in this study are shown in Table 1. The hyperparameter combination with the lowest RMSE is selected as the final one for oil well production prediction. The hyperparameter settings are shown in Table 2 and the model comparison results are shown in Figures 7 and 8 and Table 3.

Table 1. Hyperparameter ranges for the models.

Training-Related				LSTMs/GRUs		CNN/TCN		KAN
Epoch	Window size	Learning rate	Dropout	Number layers	Hidden size	Kernel size	Channels	Grid size
500	3–30	0.001–0.1	0–0.5	2–10	32–512	2–5	4–1024	16–64

Table 2. Single models hyperparameter settings.

Well	Model	Epoch	Window Size	Learning Rate	Number Layers	Hidden Size	Kernel Size	Channels	Dropout
F14	TCN	500	15	0.004	/	/	3	[300, 600, 300]	0.20
	LSTM	500	5	0.001	2	150	/	/	0.28
	GRU	500	14	0.005	2	117	/	/	0.30
	BiLSTM	500	11	0.002	3	87	/	/	0.14
	BiGRU	500	15	0.002	3	184	/	/	0.26
F12	TCN	500	5	0.004	/	/	4	[196, 128, 196]	0.13
	LSTM	500	5	0.003	3	85	/	/	0.12
	GRU	500	20	0.004	2	204	/	/	0.10
	BiLSTM	500	30	0.001	3	129	/	/	0.22
	BiGRU	500	20	0.004	2	253	/	/	0.24

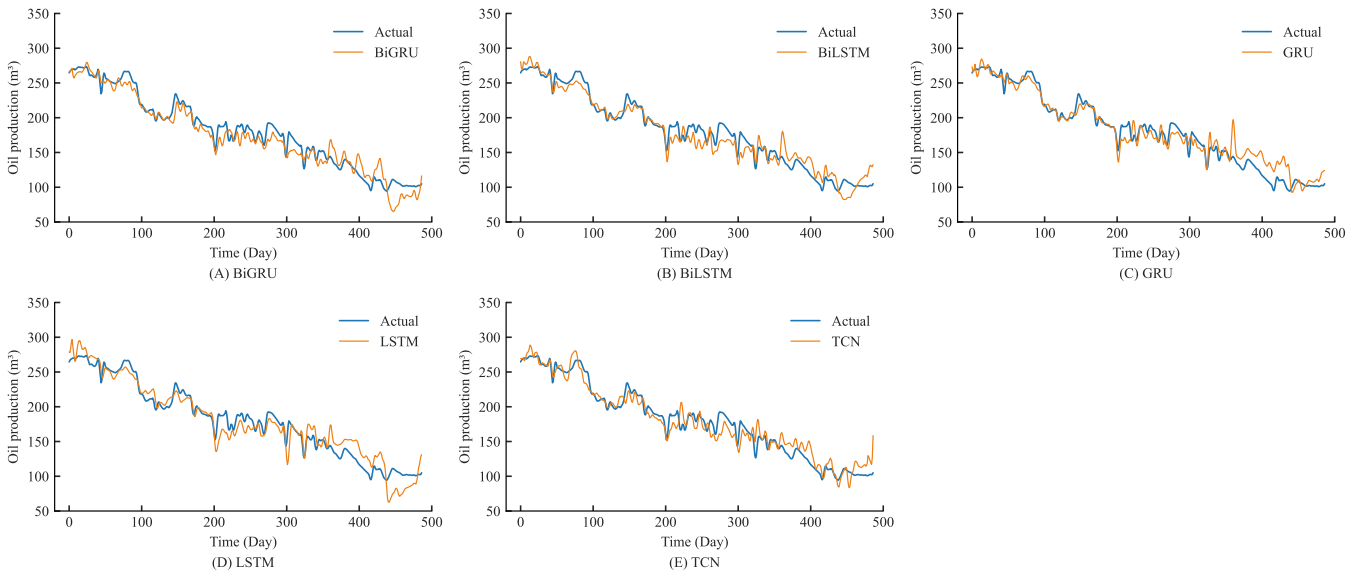


Figure 7. Comparison of actual and predicted values on Well F14 using the (**A**) BiGRU, (**B**) BiLSTM, (**C**) GRU, (**D**) LSTM and (**E**) TCN models.

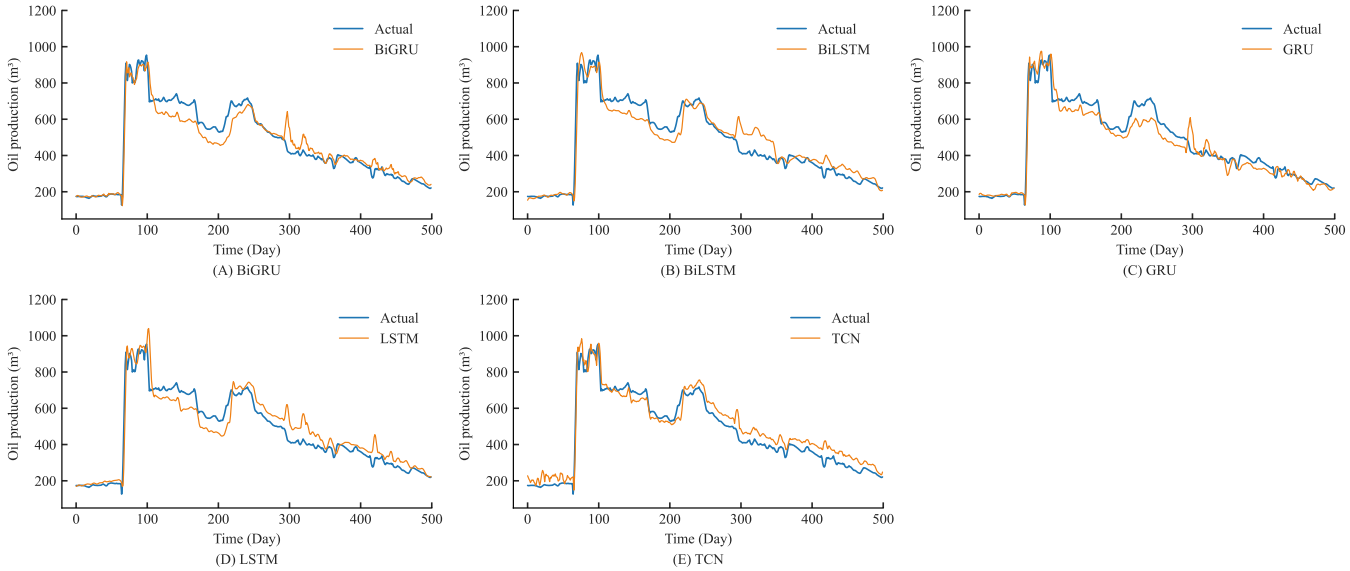


Figure 8. Comparison of actual and predicted values on Well F12 using the (**A**) BiGRU, (**B**) BiLSTM, (**C**) GRU, (**D**) LSTM and (**E**) TCN models.

Table 3. Prediction results of five single models.

Well	Model	RMSE	MAE	MAPE/%	R^2	Train Time/s
F14	TCN	14.98	11.46	7.5	0.92	22.80
	LSTM	16.40	12.98	8.9	0.90	5.77
	GRU	14.66	11.06	7.8	0.92	4.36
	BiLSTM	15.56	12.27	8.3	0.91	11.79
	BiGRU	14.49	11.45	7.7	0.92	7.50
F12	TCN	55.19	38.15	13.25	0.91	17.64
	LSTM	60.24	47.47	13.29	0.89	6.23
	GRU	55.20	40.20	11.67	0.91	5.74
	BiLSTM	57.11	41.56	10.98	0.91	15.32
	BiGRU	56.82	40.49	11.21	0.91	11.15

Figures 7 and 8 show the comparison of actual production and predicted production by different models for wells F12 and F14. The x-axis represents time, and the y-axis represents the corresponding daily oil well production. Each subfigure presents the predicted daily oil well production values (orange line) by a specific model alongside the actual production values (blue line). The closer the curves, the more accurate the model's prediction. By comparing the performance of five models in the task of oil well production prediction, we can see the strengths and weaknesses of each model.

TCN captures long-term dependencies through causal and dilated convolutions; Table 3 shows that TCN demonstrates excellent predictive performance with RMSE, MAE, MAPE and R^2 values of 14.98, 11.46, 7.5%, 0.92 on well F14 and 55.19, 38.15, 13.25%, 0.91 on well F12, respectively. Figures 7E and 8E show that the prediction curve fits the actual values very well. But due to the complex convolutional structure, the training time is quite long, requiring 22.80 s on well F14 and 17.64 s on well F12 per epoch, which results in poor performance in terms of training time.

LSTM, as a classic recurrent neural network, performs well in time series prediction. Table 3 shows that the optimized model has an RMSE, MAE, MAPE and R^2 values of 16.40, 12.98, 8.9%, 0.90 on well F14 and 60.24, 47.47, 13.29%, 0.89 on well F12, slightly lower in accuracy than TCN. On the other hand, LSTM's training speed is much faster than TCN, with each epoch taking only 5.77 s on well F14 and 6.23 s on F12, making it more advantageous for oilfield-development applications.

GRU is a variant of LSTM that simplifies the gating mechanism to achieve faster training speed while maintaining nearly the same prediction accuracy. The optimized GRU model has a prediction accuracy similar to TCN as shown in Figures 7C and 8C, with an RMSE, MAE, MAPE and R^2 values of 14.66, 11.06, 7.8%, 0.92 on well F14 and 55.20, 41.56, 10.98%, 0.91 on well F12 as shown in Table 3. Furthermore, its training time is significantly reduced to just 4.36 s and 5.72 s per epoch, significantly less time than TCN. This balance between accuracy and efficiency makes GRU highly promising for oilfield-development applications.

BiLSTM enhances data understanding by processing both forward and backward time sequences simultaneously. Figures 7B and 8B show that the prediction curve of oil production generated by the BiLSTM model and the actual value curve fit well. Table 3 shows that the optimized model has an RMSE, MAE, MAPE and R^2 values of 15.56, 12.27, 8.3%, 0.91 on well F14 and 57.11, 41.56, 10.98%, 0.91 on well F12, similar to GRU. The performance improvement is not substantial. Moreover, due to its bidirectional structure, it consumes twice the resources and has a longer training time of 11.79 s and 15.32 s per epoch, performing poorly in terms of training time compared to LSTM and GRU.

BiGRU is similar to BiLSTM but with a simpler structure. The optimized model achieves an RMSE, MAE, MAPE and R^2 values of 14.49, 11.45, 7.7%, 0.92 on well F14 and 56.82, 40.49, 11.21%, 0.91 on well F12 as shown in Table 3, slightly better than GRU. Nonetheless, compared to GRU, BiGRU consumes twice the computational resources without achieving corresponding improvements. This bidirectional mechanism shows significant advantages in certain tasks, such as natural language processing and speech recognition, because these tasks typically require the combination of both preceding and following context to understand the overall structure of the data. However, in oil well production prediction, we usually can only use historical data for prediction, as future data are unknown. Therefore, the backward propagation in BiLSTM and BiGRU cannot provide additional information when future information is unavailable. This bidirectional structure does not bring significant performance improvements.

3.2.4. Comparison Between Hybrid Model Prediction

After comparing five single models, we constructed the GRU-KAN model based on the GRU network. To validate the effectiveness of GRU-KAN, we compared it with the CNN-LSTM and CNN-GRU models. All model hyperparameters were optimized using PSO and the final hyperparameter values are provided in Table 4. The model's comparison results are provided in Figures 9 and 10 and Table 5, where the blue line shows the actual oil well production, the orange line represents the GRU-KAN predictions, the red line indicates the CNN-GRU predictions and the light purple line shows the CNN-LSTM predictions.

Table 4. Hybrid model hyperparameter settings.

Well	Model	Epoch	Window Size	Learning Rate	Number Layers	Hidden Size	Kernel Size	Channels	Grid Size	Dropout
F14	CNN-LSTM	500	15	0.003	2	137	2	8	/	0.31
	CNN-GRU	500	15	0.001	4	125	2	8	/	0.11
	GRU-KAN	500	5	0.002	2	316	/	/	30	0.10
F12	CNN-LSTM	500	15	0.003	3	214	2	8	/	0.13
	CNN-GRU	500	15	0.001	2	192	2	8	/	0.11
	GRU-KAN	500	12	0.006	3	144	/	/	32	0.21

The CNN-LSTM model is a hybrid architecture that integrates CNN and LSTM, frequently utilized to address intricate factors and long-term dependencies in time series data, demonstrating exceptional efficacy in predicting oil well production [1,28,37]. The CNN-LSTM model initially extracts localized characteristics from time series data through 1D convolutional layers to discern short-term trends. It then utilizes pooling layers to diminish feature dimensionality and extract essential information. The results from the

convolutional layers are transmitted to the LSTM layer, which analyzes long-term dependencies in the time series and encapsulates the global temporal characteristics of the data. A full connected layer ultimately converts the LSTM output into the target space, producing the final prediction results.

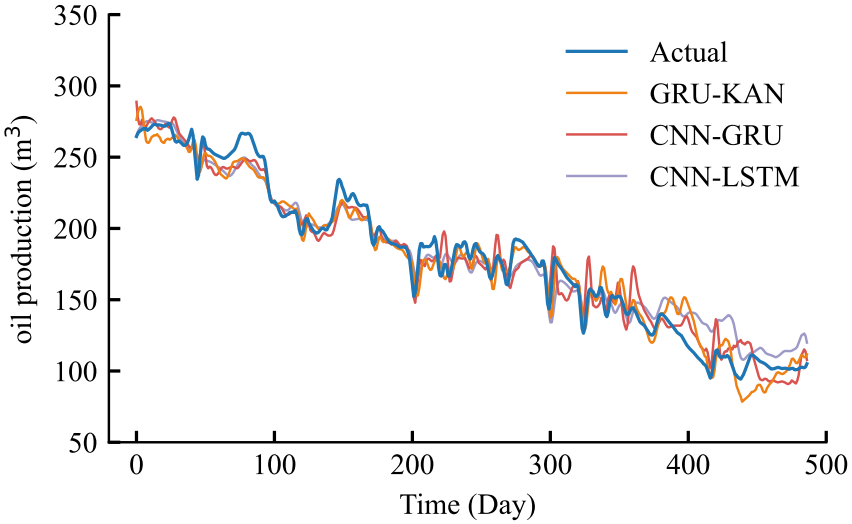


Figure 9. Comparison of actual and predicted values on well F14 using the GRU-KAN, CNN-GRU and CNN-LSTM models.

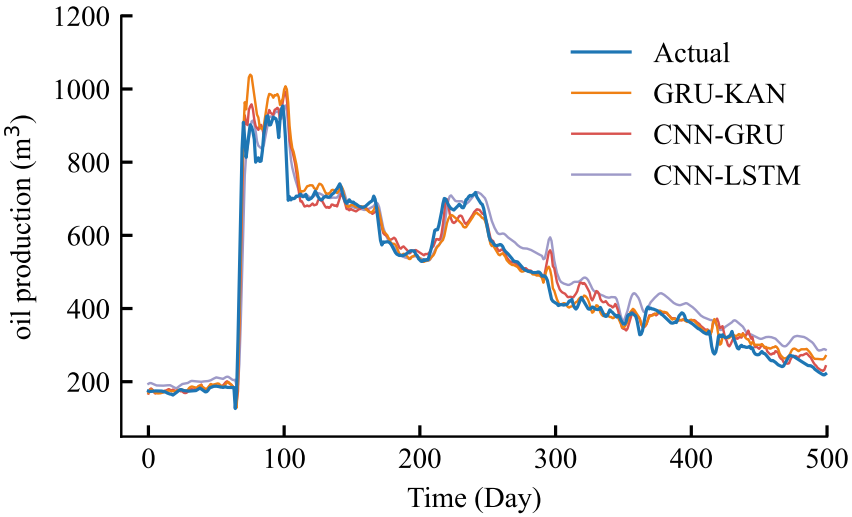


Figure 10. Comparison of actual and predicted values on well F12 using the GRU-KAN, CNN-GRU and CNN-LSTM models.

Table 5. Prediction results of hybrid models.

Well	Model	RMSE	MAE	MAPE/%	R^2	Train Time/s
F14	GRU-KAN	11.90	9.18	6.0	0.95	10.07
	CNN-GRU	12.61	9.45	6.2	0.94	21.49
	CNN-LSTM	13.40	10.50	7.2	0.93	27.74
F12	GRU-KAN	45.69	36.26	11.02	0.94	9.97
	CNN-GRU	50.08	41.61	13.74	0.93	20.99
	CNN-LSTM	52.05	44.12	15.69	0.93	25.31

From the comparison between Tables 3 and 5, it can be observed that the CNN-LSTM model outperforms all single models, achieving an RMSE, MAE, MAPE and R^2 values of 13.40, 10.50, 7.2%, 0.93 on well F14 and 52.05, 44.12, 15.69%, 0.93 on well F12. However, the combination with CNN makes the model structure more complex, resulting in longer training times and higher computational resource consumption.

The CNN-GRU is structured similarly to the CNN-LSTM model, utilizing CNN to extract local features from the data and GRU to capture temporal features. It has also achieved excellent performance in oil well production prediction. Table 5 shows that the CNN-GRU obtains an RMSE, MAE, MAPE and R^2 values of 12.61, 9.45, 6.2%, 0.94 on well F14 and 50.08, 41.61, 13.74%, 0.93 on well F12, which is slightly better than CNN-LSTM. Similarly to CNN-LSTM, the use of CNN adds considerable complexity to the model, which leads to longer training times and a greater need for computational resources.

GRU-KAN is a hybrid model that combines GRU and KAN, utilizing GRU to process chronic dependencies in the data and KAN to manage complicated relationships. From Table 5 and Figure 9, it can be seen that the GRU-KAN model achieves an RMSE, MAE, MAPE and R^2 values of 11.90, 9.18, 6.0%, 0.95 on well F14 and 45.69, 36.26, 11.02%, 0.94 on well F12, slightly higher than CNN-LSTM and CNN-GRU models. The prediction curve fits the actual values best. Furthermore, the training time is 10.07 s and 9.97 s, which is longer than LSTM and GRU but significantly shorter than CNN-LSTM and CNN-GRU. Compared to CNN, KAN significantly reduces the number of parameters by using learnable single-variable functions. The training time remains within an acceptable range while achieving a notable increase in accuracy. Therefore, the GRU-KAN model demonstrates strong application potential in oil well production prediction.

3.2.5. Comparison of Data-Imputation Methods

Using well F12 and F14 for data-imputation comparison, Table 6 presents the GRU-KAN model-prediction results after imputation with MissForest, mean imputation and the filled data. The four evaluation metrics indicate that the input features filled by MissForest have the most minor model-prediction error and the highest prediction accuracy.

Table 6. The performance of different missing value-filling methods in oil production prediction.

Well	Method	RMSE	MAE	MAPE/%	R^2
F14	MissForest	11.90	9.18	6.0	0.95
	Mean	14.07	12.68	7.8	0.93
F12	MissForest	45.69	36.26	11.02	0.94
	Mean	53.76	40.72	15.85	0.92

3.2.6. Comparison of Model Performance Before and After PSO Optimization

The setting of model hyperparameters has a significant impact on model performance. Choosing the right combination of hyperparameters is crucial to ensuring the model's efficiency, fast convergence and improved prediction accuracy. In the past, we manually adjusted parameters for tuning, which was time-consuming, labor-intensive and subjective and not always accurate. By using hyperparameter-optimization algorithms such as PSO, we can automate the tuning process, reducing human intervention. PSO's strong global search capability increases the likelihood of finding a global optimal solution, thereby improving model accuracy. We compared the GRU-KAN model on wells F12 and F14. The results in Table 7 show that PSO hyperparameter optimization significantly improved the model's predictive performance, outperforming manual tuning in all metrics with lower error and higher accuracy.

Table 7. Comparison of GRU-KAN model performance before and after PSO optimization on wells F14 and F12.

Well	Method	RMSE	MAE	MAPE/%	R^2
F14	PSO	11.90	9.18	6.0	0.95
	Manual	16.49	13.19	8.3	0.91
F12	PSO	45.69	36.26	11.02	0.94
	Manual	59.26	41.44	17.63	0.90

3.2.7. Model Generalization Analysis

To validate the generalization ability of the model on new wells, we used the model built based on well F14 and randomly selected data segments from wells F1, F11 and F15 for prediction, with the results shown in Figure 11 and Table 8. It is clear that the GRU-KAN model achieves the highest accuracy compared to the CNN-LSTM and CNN-GRU model in predicting the production of wells F1, F11 and F15. Among the three wells, the prediction for well F11 is the best, well F1 has the worst performance and well F15's prediction results are in between. The predicted production of well F1 shows significant fluctuations compared to the actual production, where the MAPE value is 89.44% and R^2 value is 0.55. The poor performance of well F1 is due to the smaller amount of data and large fluctuations, which led to greater prediction errors. The latter part of the data from well F15 also shows significant fluctuations, resulting in a lower R^2 value of 0.62. Well F11, on the other hand, has a large amount of stable data, leading to better prediction accuracy.

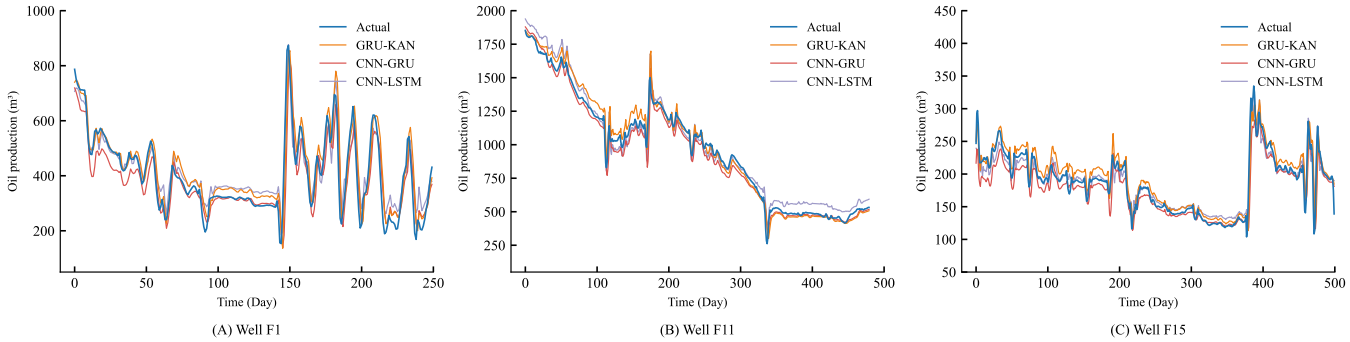


Figure 11. Comparison of actual and predicted values on well F1, F11 and F15 using the GRU-KAN, CNN-GRU and CNN-LSTM models.

Table 8. Model-generalization analysis metrics of hybrid models.

Well	Model	RMSE	MAE	MAPE/%	R^2
F1	GRU-KAN	112.94	64.39	89.44	0.55
	CNN-GRU	115.54	66.56	94.25	0.53
	CNN-LSTM	121.38	79.91	103.64	0.51
F11	GRU-KAN	146.26	87.27	11.20	0.86
	CNN-GRU	154.48	92.78	12.32	0.84
	CNN-LSTM	168.47	99.29	12.73	0.82
F15	GRU-KAN	28.59	16.76	9.70	0.62
	CNN-GRU	28.92	17.36	10.68	0.60
	CNN-LSTM	30.84	18.04	11.04	0.58

Given the poor performance of the model trained on single-well data in the generalization analysis on well F1 and F15, we considered combining the data from multiple wells for mixed training. The data from wells F14, F12 and F11 were merged for training and then the data from wells F1 and F15 were used for generalization analysis. The results of the GRU-KAN model trained with the merged data on wells F1 and F15 are shown in Table 9.

By comparing the performance of the GRU-KAN model on different wells in Tables 8 and 9, it can be seen that compared to the model built using single-well data, the model trained with combined data shows an improvement in generalization ability.

Table 9. Model-generalization analysis metrics of GRU-KAN model trained with well merged data.

Well	Model	RMSE	MAE	MAPE/%	R^2
F1	GRU-KAN	102.41	58.66	81.81	0.67
F15	GRU-KAN	25.54	15.01	8.59	0.72

4. Conclusions

Accurate prediction of oil well production is of great significance for the development and management of oilfields, especially in optimizing production efficiency and reducing operating costs. This study proposes an oil well production-prediction model based on the GRU-KAN model and validates it using real oilfield data. The prediction results are accurate. The MissForest algorithm was also applied for data imputation, which improved the data quality. Specifically, on the test datasets of wells F14 and F12, the GRU-KAN model achieved excellent results: RMSE of 11.90, MAE of 9.18, MAPE of 6.0% and R^2 of 0.95 for well F14; and RMSE of 45.69, MAE of 36.26, MAPE of 11.02% and R^2 of 0.94 for well F12. The model's generalization was also validated using data from wells F1, F11 and F15. In comparison, the performance of CNN-LSTM and CNN-GRU models on the same datasets was noticeably inferior to that of the GRU-KAN model, highlighting the latter's significant advantage in improving prediction accuracy. Furthermore, the computation time of the GRU-KAN model was nearly half that of CNN-LSTM and CNN-GRU, significantly improving computational efficiency. These results indicate that the GRU-KAN model not only enhances the accuracy of oil well production predictions but also offers higher computational efficiency, making it suitable for real-time predictions in large-scale production environments.

The broader impact of this research lies in its practical potential for oilfield production optimization. The model provides a more efficient and accurate solution for oil well production prediction and also offers insights into other oil-production-related prediction tasks. In the future, this model could be applied to construct prediction models for other production indicators, enabling multi-index prediction and supporting comprehensive optimization of oilfield production.

Despite the significant achievements of this study, certain limitations still exist. First, the data used in this study primarily come from a specific oilfield and have not been widely validated under different geological conditions or types of oilfields. Additionally, the model is constructed based on single-well data and its generalization ability needs to be improved. Future research could consider integrating data from multiple wells to further enhance the model's robustness and applicability.

The practical significance of this study lies in providing a more reliable prediction tool for oilfield production management, especially when dealing with complex nonlinear data. The GRU-KAN model demonstrates unique advantages in such scenarios. Future research can further optimize this model and explore its application in other production areas (such as hydraulic fracturing and well pressure monitoring) to address more challenges in real-world production.

Author Contributions: Conceptualization, B.Q.; methodology, B.Q.; software, B.Q. and C.Y.; validation, B.Q. and Z.Z.; formal analysis, B.Q. and Z.Z.; investigation, Z.Z.; resources, G.Q.; data curation, B.Q.; writing—original draft preparation, B.Q. and Z.Z.; writing—review and editing, J.Z. and Y.Y.; visualization, B.Q. and Z.Z.; supervision, J.Z. and Y.Y.; project administration, J.Z. and Y.Y.; funding acquisition, J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Sichuan Science and Technology Innovation Talent Fund (2022JDRC0009) and Open Research Project of State Key Laboratory of Hydrocarbon Reservoir Geology and Development Engineering (PLN 2022-33).

Data Availability Statement: Dataset available on request from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Pan, S.; Yang, B.; Wang, S.; Guo, Z.; Wang, L.; Liu, J.; Wu, S. Oil well production prediction based on CNN-LSTM model with self-attention mechanism. *Energy* **2023**, *284*, 128701. [CrossRef]
2. Kuang, L.; Liu, H.; Ren, Y.; Luo, K.; Shi, M.; Su, J.; Li, X. Application and development trend of artificial intelligence in petroleum exploration and development. *Pet. Explor. Dev.* **2021**, *48*, 1–14. [CrossRef]
3. Ma, X.; Liu, Z. Predicting the oil production using the novel multivariate nonlinear model based on Arps decline model and kernel method. *Neural Comput. Appl.* **2018**, *29*, 579–591. [CrossRef]
4. Yehia, T.; Abdelhafiz, M.M.; Hegazy, G.M.; Elnekhaily, S.A.; Mahmoud, O. A comprehensive review of deterministic decline curve analysis for oil and gas reservoirs. *Geoenergy Sci. Eng.* **2023**, *226*, 211775. [CrossRef]
5. Zhao, H. Study and Application on Production Decline Law of Water Drive Characteristic Curves. *IOP Conf. Ser. Earth Environ. Sci.* **2020**, *558*, 022026. [CrossRef]
6. Ojedapo, B.; Ikiensikimama, S.S.; Wachikwu-Elechi, V.U. Petroleum Production Forecasting Using Machine Learning Algorithms. In Proceedings of the SPE Nigeria Annual International Conference and Exhibition, Lagos, Nigeria, 3 August 2022; Volume Day 3 Wed. [CrossRef]
7. Liang, Y.; Cheng, Y.; Han, Z.; Yan, C. Numerical Simulation Analysis on Production Evolution Laws in Shale Reservoirs Considering a Horizontal Well Interwell Interference Effect. *Energy Fuels* **2024**, *38*, 4076–4090. [CrossRef]
8. Alotaibi, B.; Schechter, D.; Wattenbarger, R.A. Production Forecast, Analysis and Simulation of Eagle Ford Shale Oil Wells. In Proceedings of the SPE Middle East Unconventional Resources Conference and Exhibition, Muscat, Oman, 28 January 2015; Volume Day 3 Wed. [CrossRef]
9. Finol, A.; Ali, S.F. Numerical Simulation of Oil Production With Simultaneous Ground Subsidence. *Soc. Pet. Eng. J.* **1975**, *15*, 411–424. [CrossRef]
10. Rahmanifard, H.; Plaksina, T. Application of artificial intelligence techniques in the petroleum industry: A review. *Artif. Intell. Rev.* **2019**, *52*, 2295–2318. [CrossRef]
11. AlBinHassan, N.M.; Wang, Y. Porosity prediction using the group method of data handling. *Geophysics* **2011**, *76*, O15–O22. [CrossRef]
12. Wei, X.L.; Zhang, C.X.; Kim, S.W.; Jing, K.L.; Wang, Y.J.; Xu, S.; Xie, Z.Z. Seismic fault detection using convolutional neural networks with focal loss. *Comput. Geosci.* **2022**, *158*, 104968. [CrossRef]
13. Xiao, R.; Hu, Q.; Li, J. A model-based health indicator for leak detection in gas pipeline systems. *Measurement* **2021**, *171*, 108843. [CrossRef]
14. Hu, X.; Zhang, H.; Ma, D.; Wang, R. A tnGAN-Based Leak Detection Method for Pipeline Network Considering Incomplete Sensor Data. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–10. [CrossRef]
15. Boukredera, F.S.; Youcefi, M.R.; Hadjadj, A.; Ezenkwu, C.P.; Vaziri, V.; Aphale, S.S. Enhancing the drilling efficiency through the application of machine learning and optimization algorithm. *Eng. Appl. Artif. Intell.* **2023**, *126*, 107035. [CrossRef]
16. Song, J.; Wang, J.; Li, B.; Gan, L.; Zhang, F.; Wang, X.; Wu, Q. Real-Time Drilling Parameter Optimization Model Based on the Constrained Bayesian Method. *Energies* **2022**, *15*, 8030. [CrossRef]
17. Rong, W.Y.; Rui, Z.; An, X.S.; Jing, Y.P. Prediction of oil production based on SVM optimised multi-objective particle swarm optimisation. *Int. J. Intell. Syst. Technol. Appl.* **2018**, *17*, 415–426. [CrossRef]
18. Negash, B.M.; Yaw, A.D. Artificial neural network based production forecasting for a hydrocarbon reservoir under water injection. *Pet. Explor. Dev.* **2020**, *47*, 383–392. [CrossRef]
19. Tong, S.; Wang, F.; Gao, H.; Zhu, W. A machine learning-based method for analyzing factors influencing production capacity and production forecasting in fractured tight oil reservoirs. *Int. J. Hydrogen Energy* **2024**, *70*, 136–145. [CrossRef]
20. Bai, S.; Kolter, J.Z.; Koltun, V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv* **2018**, arXiv:1803.01271.
21. Elman, J.L. Finding structure in time. *Cogn. Sci.* **1990**, *14*, 179–211. [CrossRef]
22. Hochreiter, S. *Long Short-Term Memory*; Neural Computation MIT-Press: Cambridge, MA, USA, 1997.
23. Cho, K. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078.
24. Zhang, L.; Dou, H.; Wang, T.; Wang, H.; Peng, Y.; Zhang, J.; Liu, Z.; Mi, L.; Jiang, L. A production prediction method of single well in water flooding oilfield based on integrated temporal convolutional network model. *Pet. Explor. Dev.* **2022**, *49*, 1150–1160. [CrossRef]
25. Kumar, I.; Tripathi, B.K.; Singh, A. Attention-based LSTM network-assisted time series forecasting models for petroleum production. *Eng. Appl. Artif. Intell.* **2023**, *123*, 106440. [CrossRef]

26. Al-Shabandar, R.; Jaddoa, A.; Liatsis, P.; Hussain, A.J. A deep gated recurrent neural network for petroleum production forecasting. *Mach. Learn. Appl.* **2021**, *3*, 100013. [CrossRef]
27. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
28. Zhang, L.; Dou, H.; Zhang, K.; Huang, R.; Lin, X.; Wu, S.; Zhang, R.; Zhang, C.; Zheng, S. CNN-LSTM Model Optimized by Bayesian Optimization for Predicting Single-Well Production in Water Flooding Reservoir. *Geofluids* **2023**, *2023*, 5467956. [CrossRef]
29. Chen, G.; Tian, H.; Xiao, T.; Xu, T.; Lei, H. Time series forecasting of oil production in Enhanced Oil Recovery system based on a novel CNN-GRU neural network. *Geoenergy Sci. Eng.* **2024**, *233*, 212528. [CrossRef]
30. Vaca-Rubio, C.J.; Blanco, L.; Pereira, R.; Caus, M. Kolmogorov–Arnold Networks (KANs) for Time Series Analysis. *arXiv* **2024**, arXiv:2405.08790.
31. Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.* **2005**, *18*, 602–610. [CrossRef]
32. Liu, Z.; Wang, Y.; Vaidya, S.; Ruehle, F.; Halverson, J.; Soljačić, M.; Hou, T.Y.; Tegmark, M. KAN: Kolmogorov–Arnold Networks. *arXiv* **2024**, arXiv:2404.19756.
33. Kolmogorov, A. *On the Representation of Continuous Functions of Several Variables by Superpositions of Continuous Functions of a Smaller Number of Variables*; American Mathematical Society: Providence, RI, USA, 1961.
34. Davis, B.W.; Vass, A. Efficiency of CH2F2 laser lines at 184 and 214 micron operating with cascade transitions. In Proceedings of the 1987 Twelth International Conference on Infrared and Millimeter Waves, Orlando, FL, USA, 14–18 December 1987; pp. 370–371. [CrossRef]
35. Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948. [CrossRef]
36. Stekhoven, D.J.; Bühlmann, P. MissForest—non-parametric missing value imputation for mixed-type data. *Bioinformatics* **2011**, *28*, 112–118. [CrossRef]
37. Song, F.; Ding, H.; Wang, Y.; Zhang, S.; Yu, J. A Well Production Prediction Method of Tight Reservoirs Based on a Hybrid Neural Network. *Energies* **2023**, *16*, 2904. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.