processes

MDPI

Article

Intelligent Micro-Kick Detection Using a Multi-Head Self-Attention Network

Dezhi Zhang [1], Weifeng Sun [2,*], Yongshou Dai [2], Dongyue Wang [2], Yanliang Guo [2] and Chentao Gong [2]

[1] College of Control Science and Engineering, China University of Petroleum (East China), Qingdao 266580, China; zhang_dezhi@163.com
[2] College of Oceanography and Space Informatics, China University of Petroleum (East China), Qingdao 266580, China
* Correspondence: sunwf@upc.edu.cn

Abstract: Accurate micro-kick detection is crucial for blowout accident preventions. The more drilling parameters that change due to kicks, the more accurate the warning results become. However, when the micro-kick occurs, there is a significant time lag between these parameter changes. Dominant kick detection methods based on long short-term memory (LSTM) forget early parameter trends when dealing with long time series. To improve the recognition accuracy of micro-kicks and avoid potential blowout accidents by memorizing the early or long-term trends in drilling parameters, an intelligent micro-kick detection method based on a multi-head self-attention network is proposed. First, a novel multi-head structure is designed to separate various types of features due to different monitoring parameter changes at different speeds or trends. Second, a self-attention mechanism is employed to focus on parameter changes in separated monitoring data sequences. Then, a feed-forward network with parallel computation capability is utilized to analyze long-range correlations, thus avoiding the loss of early or long-term trend information. Finally, an artificial neural network is used to establish nonlinear relationship models between the trend features of each monitoring parameter and kick accidents. The experiment results demonstrate that the recognition accuracy of the proposed micro-kick detection method is 7.9% higher than that of the LSTM-based method.

Keywords: micro-kick detection; asynchronous changes; multi-head self-attention network; trend feature extraction

check for updates

Academic Editor: Zheng Sun

Received: 3 January 2025
Revised: 26 January 2025
Accepted: 5 February 2025
Published: 8 February 2025

Citation: Zhang, D.; Sun, W.; Dai, Y.; Wang, D.; Guo, Y.; Gong, C. Intelligent Micro-Kick Detection Using a Multi-Head Self-Attention Network. *Processes* **2025**, *13*, 465. https://doi.org/10.3390/pr13020465

Copyright: © 2025 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

The kick is a common drilling risk that increases non-productive time and drilling costs. Moreover, it can trigger a blowout, causing significant harm to human lives, property, and the natural environment. The Kaixian blowout accident in Chongqing, China (2003), and the Deepwater Horizon oil spill in the Gulf of Mexico, USA (2010), are two typical cases, which caused substantial damage to society, ecology, and economy. Therefore, it is essential to be constantly alert to the dangers of kicks during drilling.

When the pore pressure exceeds the bottom hole pressure, oil, gas, and water from the formation begin to enter the wellbore, leading to a kick accident. Drilling parameters, such as weight on bit (WOB), torque, downhole pressure, standpipe pressure (SPP), rate of penetration (ROP), differential flow out (DFO), drilling fluid density, and total active pit volume (TAPV), are correlated with pore pressure, bottom-hole pressure, and drilling fluid properties. Therefore, these parameters can be used simultaneously for accurate kick detection [1]. Kick detection methods can be primarily categorized into model-based

methods and data-based methods. Model-based methods utilize physical knowledge or statistical techniques to construct mathematical models. Therefore, they are highly interpretable and easy to understand. In contrast, data-based methods learn knowledge from data. Therefore, they are computationally efficient and self-learning.

Model-based methods include those based on physical models and comprehensive indicator models. In physical model-based methods, the model of each drilling parameter is built according to physical laws. Known physical models include the active pit volume models, drilling fluid flow-related parameter models, annular pressure models, and annular temperature models. For example, Blue [2] excludes the influence of rock chip volume on mud volume in the wellbore and establishes a mud gain model. Yalamarty [3] establishes a pit volume model for different drilling conditions, such as drilling, pump off, and tripping. Lafond [4] improves the prediction accuracy of the drilling fluid flow model using an isothermal pump model and a filtering method. Wang [5] presents an annulus flow rate inversion method based on the difference between downhole dual measurement points' pressures, which can predict the maximum allowable fluctuation upper limit. Ye [6] uses the invasion rate of overflow fluid, drilling fluid density, and formation pressure coefficient to calculate the overflow formation energy, which can reflect the strength of kick. Dow [7] designs a mud logging system using surface backpressure to implement a differential flow calculation in the managed pressure drilling system. He [8] proposes an actual transient annular pressure model based on well hydraulics analysis and uses this model to determine the kick rate and the position of the kick. Zhang [9] establishes an annulus temperature model based on the first law of thermodynamics, and determines the location where well kick occurred according to the temperature distribution of annulus fluid. These physical model-based methods are effective in conventional drilling processes. However, in complex situations, some model assumptions do not hold, leading to false alarms. In contrast, more factors are considered in the comprehensive indicator model, resulting in more accurate kick recognition. For example, Tang [10] defines two kick indicators: the drilling parameter group (d-exponent) and the flow parameter group. The drilling parameters involved include ROP, WOB, rotary speed, bit size, flow out rate, flow in rate, and SPP. Building upon Tang's research, Mao [11] introduces a new indicator, mud gain, and realizes the multi-indicator fusion. Zhu [12] considers more indicators, including human, environmental, technological, and managerial aspects, and achieves dynamic risk assessment for the entire drilling cycle. However, in comprehensive indicator models, the setting of some model parameters relies on subjective experience and knowledge. The rationality of these settings still requires improvement.

Data-driven methods include two main approaches: traditional machine learning and deep learning. Data-driven methods based on measured data do not require assumptions and can self-organize the relationship between monitoring parameters and kick accidents. Therefore, data-driven methods are applicable to multi-parameter monitoring tasks. Traditional machine learning algorithms that are commonly used for kick detection include artificial neural networks (ANNs) and Bayesian networks (BNs). Artificial neural networks are suitable for describing the nonlinear relationship between drilling parameters and kick accidents. Bayesian networks can describe and explain the causal relationships between kick-related factors and kick events. For example, Muojeke [13] explores the simple structure and interpretability of ANN and establishes a kick detection model based on an ANN-binary classifier. Duan [14] establishes eight different ANN models for kick monitoring in different drilling conditions, such as back reaming, circulation, and connection. Nhat [15] proposes a data-driven Bayesian network for early kick detection when knowledge about the drilling process is not adequate. Chen [16] builds a BN model incorporating the bowtie approach commonly used for risk management to enhance its interpretabil-

ity. Meng [17] considers the stochastic uncertainties and fuzzy uncertainties of drilling parameters and proposes a dynamic Bayesian network based on fuzzy probability with the real-time probability of the blowout calculated. Zhang [18] considers more accident-causing factors such as equipment failures, abnormal processes, and operator errors and presents a more comprehensive dynamic Bayesian network to achieve more accurate early kick detection. In addition, support vector machines (SVMs) and decision trees (DTs) have also been used for kick detection [19,20]. However, there have been fewer research and applications in recent years due to their reliance on the experience of experts and poor performance when handling large amounts of data. To overcome the limitations of individual models, some ensemble learning algorithms are employed to combine multiple weak models into a strong model with more accurate kick recognitions [21–23]. However, if the weak model is not chosen appropriately, the combination of multiple models cannot eliminate their common shortcomings, while increasing the difficulty of interpretation.

Deep learning algorithms have feature extraction capabilities that traditional machine learning algorithms lack. The extracted high-level features related to kick accidents can effectively avoid false alarms caused by noisy data, parameter outliers, and missing values. Deep learning algorithms include supervised learning algorithms and unsupervised learning algorithms. The dominant supervised learning algorithms in the field of kick detection are the recurrent neural network (RNN) and its variants, such as long short-term memory (LSTM) and gate recurrent unit (GRU). These algorithms can extract the trend features of the drilling data sequences. For example, Fjetland [24] constructed an LSTM-based model for kick recognition using simulation data generated by the OpenLab drilling simulator. Osarogiagbon [25] built an LSTM model using the measured d-exponent and standpipe pressure, validating the effectiveness of the LSTM-based early kick detection method. Yin [26] constructed an LSTM-based model using a multi-source dataset, including surface mud logging data, riser acoustic data, and downhole measurement data. It was demonstrated that LSTM outperforms DT, k-nearest neighbor (KNN), and SVM in several model evaluation metrics. Duan [27] stacked a one-dimensional convolutional network on top of an LSTM network to enhance the local feature capturing capability of the intelligent kick detection model and reduce the false alarm rate. Compared with LSTM networks, GRU networks with simpler structures are less prone to overfitting with small sample sizes. Zhang [28] proposed a cascaded GRU-based model for hierarchical early kick detection, which achieves accurate alarms for different risk levels of kick accidents. Zhang [29] proposed a cascaded GRU network with adaptive monitoring parameters to improve the recognition accuracy of the kick detection method when some monitoring parameters are not obtainable. Autoencoders are the commonly used unsupervised deep learning methods for kick detection, which can handle unlabeled drilling data [30]. Zhu [31] used a bidirectional LSTM network as the encoder and decoder of the autoencoder to improve its capability of handling drilling data sequences, achieving higher kick recognition accuracy compared with traditional autoencoders.

The LSTM and GRU forget historical information due to the gradient vanishing when dealing with excessively long time series. Therefore, it is difficult to extract early or long-term trend features of long sequences. To overcome these problems, the Transformer is proposed to capture long-term dependencies and learn the intrinsic relationships within the sequence. The Transformer and its variants are widely used in various fields. In the field of natural language processing, the Transformer performs well in tasks such as language translation, sentiment analysis, text generation, and question-answering systems due to its ability to extract long-term dependencies between words [32]. In computer vision research, the Vision Transformer is different from the traditional convolutional neural network's limited receptive fields and can extract global image features. Therefore, it is widely used

in image recognition, object detection, and image classification tasks [33]. ESPNet-ST and Conformer are applied in the field of speech recognition and speech synthesis [34,35]. Additionally, the Transformer is applied in areas such as financial market forecasting, market demand prediction, mechanical failure detection, and recommendation system development [36]. In the field of oil and gas development, Wang [37] is the first to apply the Transformer to the prediction of real-time drilling data.

The response speeds of various monitoring parameters differ and are related to the influx velocity of the kick accident. The influx velocity is small when a micro-kick occurs. The small influx velocity results in a slow change in the monitoring parameters, which increases the gap between the response times of these monitoring parameters, as well as the uncertainty of the start time and duration of parameter changes. The dominant deep learning-based kick detection methods process data sequences using a recursive update mechanism, which relies on two assumptions: the time intervals between key information are consistent, and the correlations between neighboring moments are stronger. In the kick detection task, the key information refers to the change within the data sequence of each parameter. Therefore, these methods face two challenges when monitoring micro-kick accidents: the difficulty of flexibly capturing changes within parameter data sequences and the loss of information about early or long-term parameter trends. This leads to low accuracy of micro-kick detection. To improve the recognition accuracy of micro-kicks and avoid potential blowout accidents by memorizing the early or long-term trends in drilling parameters, an intelligent micro-kick detection method based on a multi-head self-attention network is proposed in this paper. First, the cosine similarity between the monitoring parameter values at each moment is calculated to capture the changes within parameter data sequences. The difference in values between adjacent moments increases when the parameters change. Second, higher attention weights are assigned to these values using the attention mechanism. Third, a weighted sum of the parameter values at all moments is calculated to obtain the overall trend features of the parameter data sequence. Finally, a nonlinear relationship model is established between the trend features of all monitoring parameters and kick accidents. Additionally, a new multi-head structure and a new position encoding are redesigned to adapt to upward or downward trends of different parameters with different speeds of change. The multi-head structure can individually extract different types of trend features and fuse these features. The position encoding can increase the magnitudes of changes in parameter data sequences. The original Transformer model can only process one-dimensional data. Therefore, a two-dimensional classifier is designed to analyze both temporal correlations and correlations between parameters in drilling data.

When handling asynchronous and multivariate time-series data in drilling operations, traditional threshold-based systems use fixed thresholds and cannot adapt to the changeable drilling data, whereas MSA learns directly from the measured drilling data to automatically capture complex temporal and spatial dependencies based on the self-trends of parameters, improving the adaptability of the detection method. Unlike LSTM's sequential computation, which causes the weights of earlier information to gradually decrease with repeated updates and transmissions, the MSA model addresses this limitation through parallel processing data at all positions in the data sequence. This approach enables each position to directly access information from any other position, effectively remembering asynchronous information. Unlike other attention-based models, which rely on external parameters to guide attention allocation, the MSA model focuses more on internal data relationships of each parameter by directly analyzing the self-trends of parameters. Compared with hybrid models with complex structures and numerous hyperparameters, MSA's simpler design reduces the risk of overfitting, lowers data requirements, and delivers superior performance in scenarios with limited kick accident data. In summary, the multi-head struc-

ture of the MSA model effectively addresses the temporal irregularities of asynchronous data and the coupled relationships within multivariate data. The self-attention mechanism of the MSA model enables parallel computations of all elements in the data sequence. Additionally, it can leverage historical data to adjust the distribution of attention based on the inherent trends of the parameters.

2. Asynchronous Change of Monitoring Parameters

The asynchronous change in monitoring parameters refers to the fact that different parameters change at different moments after a kick accident occurs and the durations of those changes are different.

2.1. Reasons for Asynchronous Change

There are two main reasons why the monitoring parameter changes are not synchronized: the different signal propagation speeds and the different conditional relationships between parameters and kick accidents. The durations of the parameter change are related to the influx volume and annular pressure differences.

The propagation velocity of pressure waves is faster than the flow velocity of drilling fluid. As a result, engineering parameters such as ROP, WOB, SPP, and torque change earlier than the mud pit volume and drilling fluid performance parameters collected by mud logging. A decreased annular pressure difference is a necessary but insufficient condition for the kick occurrence. The engineering parameters can reflect the changes in bottom hole pressure and pore pressure, indicating a decrease in annular pressure difference. A negative annular pressure difference and an increased pit volume are necessary and sufficient conditions for a kick accident. A decreased DFO means a negative annular pressure difference, while TAPV and mud performance parameters reflect changes in the drilling fluid properties. Therefore, the engineering parameters change no later than DFO and TAPV. When a micro-kick accident occurs, the influx volume is small, while the annular pressure difference decreases slowly, resulting in longer durations of the parameter change. Conversely, during a sudden kick event, the parameters change rapidly.

The moments of changes of six monitoring parameters for seven kick cases are statistically analyzed. The statistical results are shown in Table 1. The moment of kick occurrence is considered as the moment zero (00:00:00).

Table 1. Moments of parameter changes in kick cases.

Cases	ROP	WOB	Torque	SPP	DFO	TAPV
Case 1	00:00:00	00:00:13	00:00:20	00:00:17	00:01:55	00:03:08
Case 2	00:00:00	00:00:18	00:00:00	00:00:00	00:01:10	00:02:04
Case 3	00:00:00	00:00:09	00:00:00	00:00:10	00:00:03	00:00:17
Case 4	00:00:08	00:00:00	00:00:26	00:00:01	00:01:17	00:02:09
Case 5	00:00:00	00:00:03	00:00:12	00:00:00	00:01:12	00:01:56
Case 6	00:00:14	00:00:00	00:00:00	00:00:14	00:03:15	00:03:40
Case 7	00:00:00	00:00:00	00:00:22	00:00:00	00:01:27	00:02:14

To avoid the randomness of the statistical results, the trimmed mean of the moments of parameter changes is used to evaluate the response time of parameters. The average moments of parameter changes are as follows: ROP (3 s), WOB (6 s), SPP (6 s), torque (11 s), DFO (88 s), and TAPV (133 s). Therefore, the moments of change are different for different monitoring parameters. The highest delay for TAPV was 220 s in the drilling data collected. To encompass the range of changes in all monitoring parameters, the sequence length is set to 240, including the drilling data within a 4 min time window.

2.2. Capturing Changes in Monitoring Parameters

Changes in monitoring parameters are captured by selecting the appropriate length of the data sequence. A short sequence fails to capture the long-term gradual trend features, and a longer sequence ignores early transient trend features. The divergence of moving average (DMA) value is employed to evaluate the rate of change in parameter data sequence with different lengths [38]. The DMA value is the difference between the moving average values of a long sliding window (MA_l) and a short sliding window (MA_s), where the length of the short window is half that of the long window. Figure 1 shows the calculation of the DMA value.

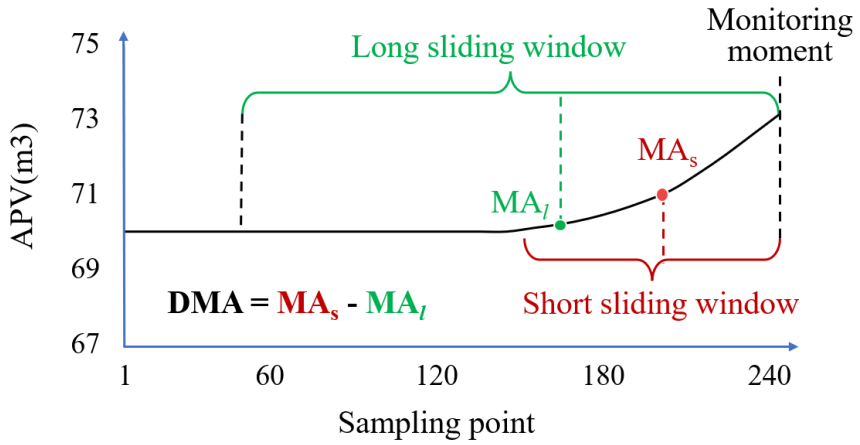


Figure 1. Calculation of DMA value.

Table 2 lists the DMA values of parameter data sequences with different lengths at five monitoring moments. At different monitoring moments, the optimal length of the parameter data sequence is different. Additionally, at the same monitoring moment, different monitoring parameters have different optimal lengths of the data sequence.

Table 2. DMA values of parameter data sequences with different lengths.

Monitoring Moment	Sequence Length	ROP	Torque	WOB	SPP	DFO	TAPV
00:01:00	120	0.167	0.022	−0.070	−0.071	0.005	−0.003
	180	0.118	0.042	−0.009	−0.052	0.004	0.000
	240	0.091	0.041	−0.008	−0.044	0.003	0.004
00:02:00	120	0.026	0.031	−0.040	−0.072	0.023	0.003
	180	0.154	0.048	−0.068	−0.142	0.021	−0.002
	240	0.191	0.052	−0.076	−0.107	0.018	−0.003
00:03:00	120	0.011	0.004	−0.003	−0.019	0.029	0.000
	180	0.043	0.027	−0.018	−0.051	0.035	−0.006
	240	0.112	0.047	−0.077	−0.116	0.041	0.002
00:04:00	120	0.020	0.012	0.023	0.034	0.034	0.008
	180	0.021	0.001	0.005	0.015	0.050	0.007
	240	0.036	0.027	−0.010	−0.036	0.056	0.001
00:05:00	120	−0.026	−0.001	−0.009	−0.017	0.022	0.011
	180	0.001	0.008	0.009	0.010	0.040	0.006
	240	0.014	0.014	0.018	0.018	0.059	0.013

The time periods with the highest rate of change in each parameter are as follows: ROP (0 min to 2 min), torque (0 min to 2 min), SPP (30 s to 2 min), WOB (1 min to 3 min), DFO (1 min to 5 min), and TAPV (1 min to 5 min).

Due to the kick recognition result at the current moment being influenced by the trends of the monitoring parameters a long time ago, longer data sequences can lead to the problem of long-term dependencies. In dominant deep learning methods for kick detection (such as LSTM and GRU networks), the problem of long-term dependencies refers to the vanishing gradient problem. These networks fail to memorize information from a long time ago due to the vanishing gradient problem. Furthermore, the correlation of data from adjacent time points becomes stronger in these networks due to their cyclic structure. These methods are unable to establish long-range correlations between asynchronous monitoring parameters, particularly when the response times of these parameters are significantly disparate.

Figure 2 shows the attention distribution of the LSTM-based kick detection model on parameter data sequences. Changes in ROP and SPP are overlooked due to their earlier occurrence when the influx volume is small. The response time difference between ROP, SPP, and TAPV is significant. Due to reliable kick detection depending on the trends of drilling parameters rather than their real-time values, more attention should be paid to changes in asynchronous monitoring parameters to improve the accuracy of micro-kick detection. Therefore, selecting appropriate data sequence lengths and capturing changes in parameter data sequences can improve the accuracy of micro-kick detection.

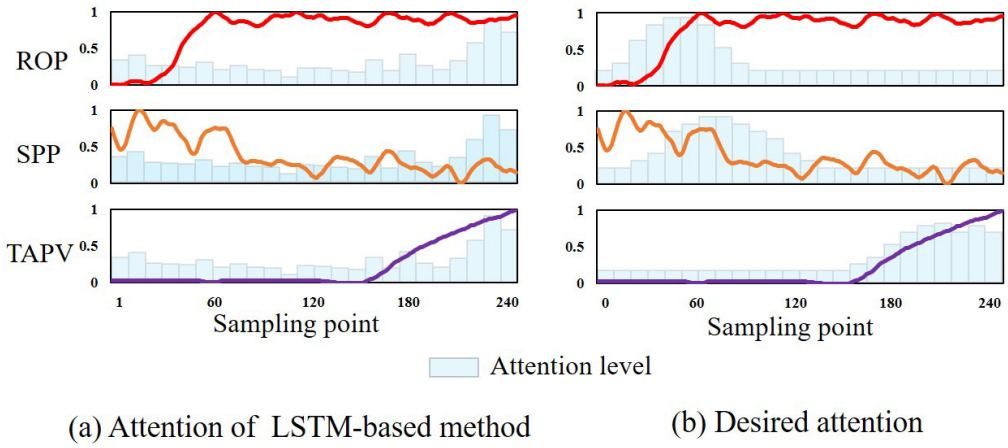


Figure 2. Attention distribution on drilling parameter sequences.

3. Proposed Micro-Kick Detection Method

The intelligent micro-kick detection method based on the multi-head self-attention (MSA) network combines the multi-head self-attention mechanism and the fully connected neural network. The asynchronous and uncertain trend features of monitoring parameters are extracted by the multi-head self-attention mechanism, and then a relationship model between the trend features and kick accidents is established using fully connected neural networks. The proposed method improves the attention of the intelligent kick detection model to changes in parameter values, thus accurately capturing the changes in asynchronous parameters at different moments when the micro-kick occurs.

Figure 3 shows a flowchart of the proposed micro-kick detection method. The steps of the method are as follows: First, samples are created using a window slicing method and data preprocessing methods. Second, sequential information is added to the samples by an improved positional encoding method. Third, the linear features of each monitoring parameter in the samples are extracted by an enhanced multi-head attention mechanism. Subsequently, the linear features are transformed into nonlinear features using a feed-

forward network. Finally, the probability of the samples belonging to the two categories is calculated using a proposed two-dimensional classifier according to the nonlinear features.

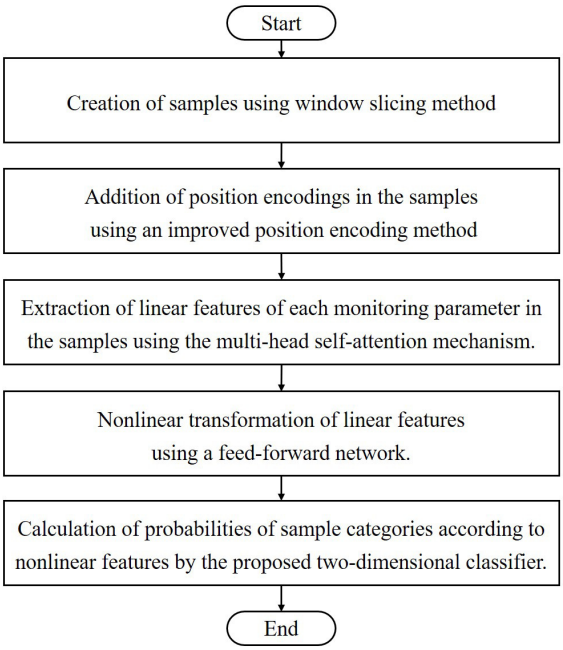


Figure 3. Flowchart of the proposed intelligent micro-kick detection method.

Figure 4 shows the structure of the MSA-based kick detection model. The model consists of four main components: the positional encoding module, the multi-head self-attention module, the feed-forward network module, and the classifier. The positional encoding module is used to add sequential information to the input sequence. The multi-head self-attention module consists of three self-attention heads and a Concat layer. Each self-attention head is used to extract a type of trend features of monitoring parameters. Different types of linear trend features are captured using multiple self-attention heads. Then outputs of all self-attention heads are fused into a new sequence through the Concat layer. The feed-forward network module includes a ReLU layer with a nonlinear activation function and a linear layer without an activation function. The linear trend features are transformed into nonlinear features using the nonlinear activation function in the ReLU layer, and the feature dimension is kept constant by the linear layer. The classifier consists of two fully connected (FC) layers. The trend features of the monitoring parameters in the temporal dimension are quantified by the first FC layer. A nonlinear relationship model between the quantified trend features and the kick accident is established by the second FC layer.

3.1. Positional Encoding Module

To add order information for the input sequence, the Transformer model adds the "positional encodings". The positional encodings decrease as the relative distance between two elements increases [39]. However, the downward trend introduced by the positional encodings interferes with trend determinations of monitoring parameters. To address this problem, the new positional encodings are designed according to the direction and rate of change of monitoring parameters. It reduces the impact of positional encoding on parameter trends and adds order information.

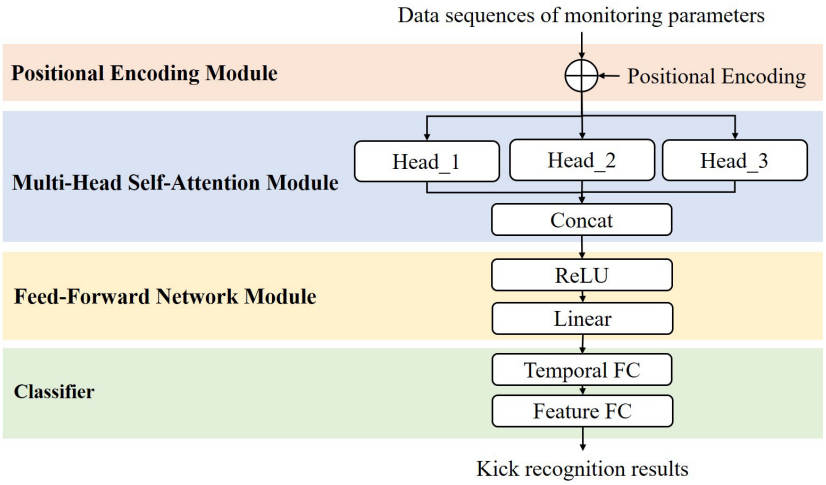


Figure 4. Structure of the MSA-based model.

The formula for positional encoding in the Transformer model is

$$\begin{cases} PE(t, 2i) = sin(t/k^{2i/d}) \\ PE(t, 2i+1) = cos(t/k^{2i/d}), \end{cases} \tag{1}$$

where PE refers to positional encoding, t is the position of positional encoding, d represents the dimension of the input sequence, $i = 0, 1, \ldots, d/2 - 1$ is the dimension of positional encoding, and k is a constant. The size of the position encoding for monitoring parameters is determined by dimension i and constant k.

3.1.1. Determining Dimension i

The smaller the value of i, the larger the value of the position encoding, resulting in a greater impact on the trend of monitoring parameters. When $i = 0$, the position encoding is like the high-frequency noise. To reduce the influence of position encoding on the trend of the parameter, the value of i for different parameters is set according to the direction and rate of the change of the parameter. For example, the i for WOB and SPP with a rapidly decreasing trend is set to 1. The i for ROP and torque with a rapid increasing trend is set to 2, and the i for DFO and TAPV with a slower increasing trend is set to 3.

3.1.2. Setting Constant k

The setting of k follows the principle that the positional encoding decreases as the relative distance between two elements increases. The maximum of relative distance between two elements is the length of the data sequence.

According to the difference formula for cosine $cos(\alpha - \beta) = cos\alpha cos\beta + sin\alpha sin\beta$, the dot product of the positional encodings at two moments is calculated as

$$PE_t^T \cdot PE_{t+\triangle t} = \sum_{i=0}^{2} cos(\triangle t/k^{2i/6}), \tag{2}$$

where PE_t represents the positional encoding of the t-th element in the data sequence, and k is a constant. When the value of $\triangle t/k^{2i/6}$ remains within the range of $[0, \pi]$, the dot product of two positional encodings, $PE_t^T \cdot PE_{t+\triangle t}$, decreases as $\triangle t$ increases. When the value of $\triangle t$ is the maximum l, $l/k^{2i/6} = \pi$. The numerical ranges of k when $i = 0, 1, 2, 3$ are listed in Table 3.

Table 3. Numerical ranges of k for different i's.

i	k
0	$(-\infty, +\infty)$
1	$[(l/\pi)^3, +\infty)$
2	$[(l/\pi)^{3/2}, +\infty)$
3	$[l/\pi, +\infty)$

Therefore, the value of k should be greater than $(l/\pi)^3$. For example, if the size of a sliding time window is 240, the value of k should be greater than 446,524. For ease of calculation, $k = 5 \times 10^5$.

3.1.3. Position Encodings of Monitoring Parameters

The impacts of the traditional position encoding and the proposed position encoding on the drilling data are compared in Figure 5. The traditional position encoding introduces high-frequency noise to ROP and torque and alters the downward trends of WOB and SPP. In contrast, the proposed position encoding preserves the individual trends of six monitoring parameters and adds positional information.

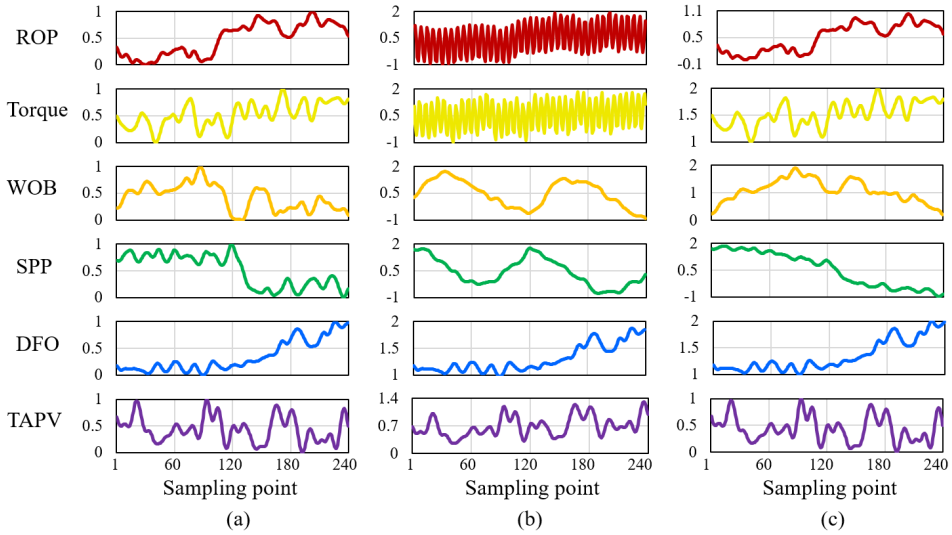


Figure 5. Curves of drilling data with (**a**) no position encoding, (**b**) traditional position encoding, and (**c**) proposed position encoding.

Finally, the positional encodings of the six monitoring parameters are

$$x_t + PE_t = \begin{bmatrix} x_t^{ROP} + sin[t/(5\times10^5)^{2/3}] \\ x_t^{Torque} + cos[t/(5\times10^5)^{2/3}] \\ x_t^{WOB} + sin[t/(5\times10^5)^{1/3}] \\ x_t^{SPP} + cos[t/(5\times10^5)^{1/3}] \\ x_t^{DFO} + sin[t/(5\times10^5)] \\ x_t^{TAPV} + cos[t/(5\times10^5)] \end{bmatrix}, \quad (3)$$

where x_t and PE_t, respectively, represent the values and positional encodings of the monitoring parameters at moment t.

3.2. Multi-Head Self-Attention Module

The multi-head self-attention mechanism is first proposed in the Transformer model for natural language processing. It captures the positional information of key elements in

the data sequence according to their correlations. The loss of historical information in long data sequences is avoided by parallel computations.

In the proposed multi-head self-attention module, to adapt to parameter trends with different directions or rates of change, a new multi-head structure is designed. This enables an attention head to focus on extracting a type of trend feature. This module uses the self-attention mechanism to adjust the model's attention distribution by modifying the attention weights at different times, thus focusing attention on the trend changes of parameters.The attention weight is a trainable weight that is calculated according to the similarity between the current moment and other moments. Therefore, the attention weights are related to the trends of the parameter data sequence and the historical drilling data.

3.2.1. Self-Attention Mechanism

The self-attention mechanism is a specific type of attention mechanism. In an attention mechanism, attention weights are calculated according to the relationship between the query and key vectors, which are used to compute a weighted sum of all elements in the value vector. In the self-attention mechanism, the query, key, and value vectors are the same vector. Therefore, the self-attention-based model can focus more on the internal correlations within the input data and reduce reliance on external information.

The formula of the attention mechanism is

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V}, \tag{4}$$

where $\mathbf{Q}$ represents the query matrix consisting of l query vectors, $\mathbf{K}$ is the key matrix consisting of l key vectors, and $\mathbf{V}$ refers to the value matrix consisting of l value vectors. d_k denotes the dimensionality of the query or key vectors.

In the self-attention mechanism, the matrices $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ are calculated by multiplying the input matrix $\mathbf{X}$ by three different trainable weight matrices, denoted as $\mathbf{W}_q$, $\mathbf{W}_k$, and $\mathbf{W}_v$, respectively. The formula of the self-attention mechanism is

$$Attention(\mathbf{X}) = softmax[\frac{\mathbf{W}_q\mathbf{X}(\mathbf{W}_k\mathbf{X})^T}{\sqrt{d_k}}]\mathbf{W}_v\mathbf{X}, \tag{5}$$

where $\mathbf{X}$ represents the input matrix, and $\mathbf{W}_q$, $\mathbf{W}_k$, and $\mathbf{W}_v$ denote the query, key, and value weight matrices, respectively. d_k is the dimensionality of the query or key vectors.

3.2.2. Multi-Head Structure

There are different types of trends of data sequences of monitoring parameters: sudden increases in ROP and torque, rapid decreases in WOB and SPP, and gradual rises in DFO and TAPV [40]. Therefore, a new multi-head structure is designed to extract upward and downward trends and distinguish between rapid and gradual changes. By employing the multi-head structure, the negative impact of asynchronous parameter changes on the recognition accuracy is effectively mitigated.

Figure 6 shows the drilling data process of the multi-head self-attention module. The multi-head self-attention module consists of three self-attention heads and a Concat layer. A feature vector of two monitoring parameters with similar trend changes is obtained by each self-attention head. The three feature vectors are concatenated into a vector using the Concat layer to provide a comprehensive feature representation for subsequent analysis processes.

Referring to the formula for the self-attention mechanism in Equation (8), the calculation formula for a self-attention head is

$$\mathbf{H}_j = \mathbf{V}_j^T softmax(\frac{\mathbf{K}_j^T \mathbf{Q}_j}{\sqrt{d_k}}), \tag{6}$$

where $\mathbf{H}_j$ represents the output matrix of the j-th self-attention head; $\mathbf{Q}_j$, $\mathbf{K}_j$, and $\mathbf{V}_j$ denote the query, key, and value matrices of the j-th self-attention head, respectively; and d_k is the dimensionality of the query or key vectors.

The output matrices of the three self-attention heads are concatenated by the Concat layer as

$$\mathbf{F}_{concat} = [\mathbf{H}_1, \mathbf{H}_2, \mathbf{H}_3], \tag{7}$$

where $\mathbf{F}_{concat}$ is the concatenated matrix, and $\mathbf{H}_1$, $\mathbf{H}_2$, and $\mathbf{H}_3$ represent the output matrices of the three self-attention heads.

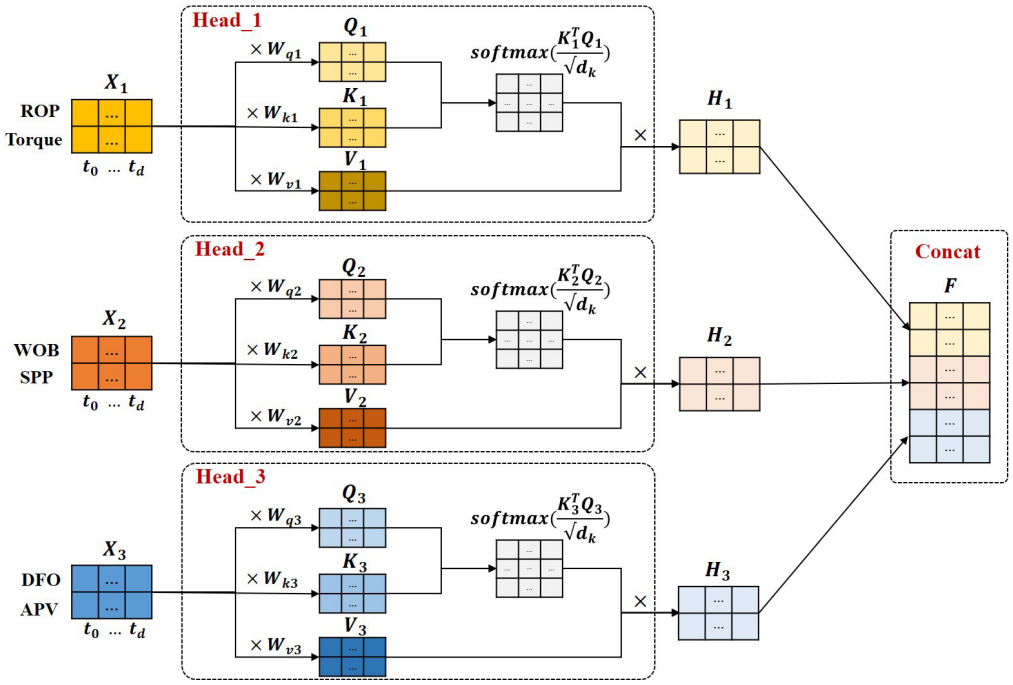


Figure 6. Data processing in the multi-head self-attention module.

3.3. Feed-Forward Network Module

The feed-forward network module consists of a ReLU layer and a linear layer, and it is used to model the nonlinear relationships between monitoring parameter features. The ReLU layer extracts nonlinear features of monitoring parameter trends, and the linear layer models the relationships among these features.

3.3.1. ReLU Layer

The trend features of each monitoring parameter extracted by the multi-head self-attention module are inputted into the ReLU layer. After the computation of the nonlinear neuron nodes, the associated features are obtained by

$$\mathbf{F}_{ReLU} = ReLU(\mathbf{W}_{ReLU}\mathbf{F}_{concat} + \mathbf{b}_{ReLU}), \tag{8}$$

where $\mathbf{F}_{concat}$ represents the output matrix of the multi-head self-attention module, $\mathbf{F}_{ReLU}$ is the output matrix of the ReLU layer, and $\mathbf{W}_{ReLU}$ and $\mathbf{b}_{ReLU}$ denote the weights and biases of the ReLU layer, respectively.

3.3.2. Linear Layer

The associated features obtained from the ReLU layer are fed into the linear layer. These features are fused into new features through linear transformation. The number of new features is the same as the number of monitoring parameters. The new feature matrix is

$$\mathbf{F}_{linear} = ReLU(\mathbf{W}_{linear}\mathbf{F}_{ReLU} + \mathbf{b}_{linear}), \tag{9}$$

where $\mathbf{F}_{ReLU}$ and $\mathbf{F}_{linear}$ represent the output matrices of the ReLU layer and the linear layer, respectively. $\mathbf{W}_{linear}$ and $\mathbf{b}_{linear}$ denote the weights and biases of the linear layer, respectively.

3.4. Two-Dimensional Classier

Since extracted features of the monitoring parameters are two-dimensional data, the two-dimensional classifier is designed to establish a nonlinear relationship model between extracted monitoring parameter features and kick accidents. The two-dimensional classifier consists of two fully connected layers. The first layer is a temporal FC layer, which is used to extract nonlinear features of parameter trends. The second layer is a feature FC layer, which is employed to achieve the feature classification. Additionally, it uses the softmax algorithm to convert the output into a probability distribution.

3.4.1. Temporal FC Layer

The temporal FC layer with the ReLU nonlinear activation function can efficiently process long data sequences, as the ReLU function randomly omits some neurons. The output matrix of the temporal FC layer is

$$\mathbf{F}_{time}^{T} = ReLU(\mathbf{W}_{time}\mathbf{F}_{linear}^{T} + \mathbf{b}_{time}), \tag{10}$$

where $\mathbf{F}_{linear}^{T}$ and $\mathbf{F}_{time}^{T}$ represent the transpose matrices of the output matrices of the linear layer and the temporal FC layer, respectively. $\mathbf{W}_{time}$ and $\mathbf{b}_{time}$ represent the weights and biases of the temporal FC layer, respectively.

3.4.2. Feature FC Layer

The feature FC layer establishes a nonlinear relationship model between the extracted features and kick accidents. The model's outputs are transformed into a probability distribution using the softmax nonlinear activation function for feature classification. The outputs of the model are

$$\mathbf{Y} = softmax(\mathbf{W}_{output}\mathbf{F}_{time} + \mathbf{b}_{output}), \tag{11}$$

where $\mathbf{F}_{time}$ and $\mathbf{Y}$ represent the output matrices of the temporal FC layer and the feature FC layer, respectively, and $\mathbf{W}_{output}$ and $\mathbf{b}_{output}$ denote the weights and biases of the feature FC layer, respectively. Finally, the probabilities of the sample belonging to the two categories are denoted as y_1 and y_2. If y_1 is greater than y_2, the sample is classified as a "kick" sample; otherwise, it is classified as a "non-kick" sample. For example, if the size of the input matrix is $6 * l$, Figure 7 shows the input–output matrix sizes of each layer in the proposed model.

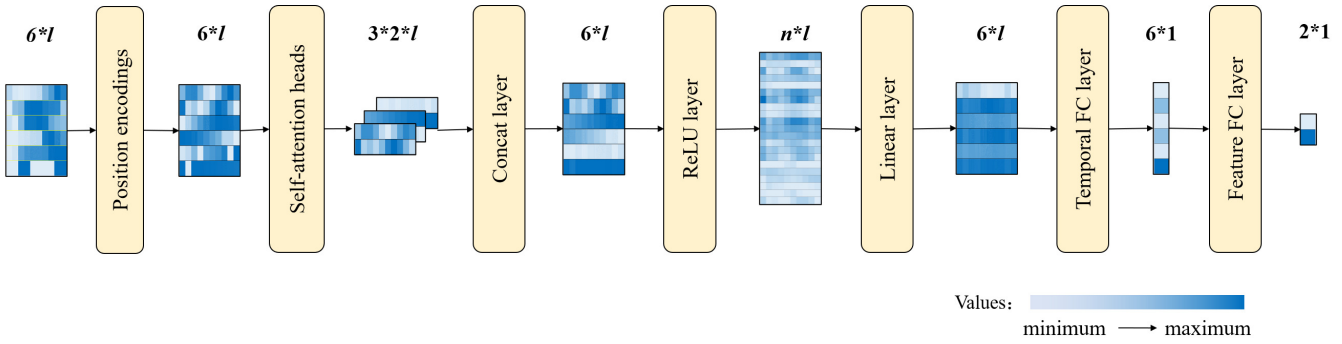


Figure 7. Sizes of the input–output matrices in the proposed model.

4. Experiment

The proposed micro-kick detection method based on a multi-head self-attention network is a supervised learning approach that requires training the model using labeled samples. The experimental process consists of three steps: dataset creation, model building, and model evaluation. The dataset creation involves data augmentation, data preprocessing, data labeling, and data splitting. First, a sliding time window is employed to obtain data sequences from drilling data to produce samples. Second, the outliers and noise in the samples are removed using data preprocessing techniques to improve the efficiency of model training and convergence. Third, the samples are labeled according to expert knowledge and field records and categorized as "kick" and "non-kick" classes. Finally, the dataset creation is accomplished by assigning the labeled samples to the training, validation, and testing datasets. A grid search technique is employed to determine the network parameters to complete the model-building process. After model building, multiple evaluation metrics are used to assess the model's performance.

4.1. Dataset Creation

The experimental data are collected from 22 wells in an oilfield in China. The sampling frequency is 1 Hz. The measured data are collected in real time using the comprehensive logging unit. The data type is floating-point, and the number is rounded to two decimal places. The units of each monitoring parameter are ROP (m/h), WOB (kN), torque (kN∗m), SPP (MPa), DFO (L/s), and TAPV (m^3). The kick samples are obtained through data augmentation, data preprocessing, and data labeling. Then the dataset is generated by performing data splitting.

1. Data augmentation: To obtain more samples from drilling data, a time series data augmentation technique called window slicing is employed to create kick samples [41]. Additionally, to meet the real-time requirements for kick warning in the field, samples are obtained using time windows ranging from 1 to 6 min, based on the recommendations of drilling experts. Subsequently, the recognition accuracies of the MSA-based and LSTM-based kick detection models are tested using samples of different sizes, as shown in Figure 8. The MSA-based and LSTM-based methods have the highest recognition accuracy when the time window length is set to 4 min. A total of 17,745 samples are obtained using a 4 min sliding window.
2. Data preprocessing: Data preprocessing includes the identification and removal of outliers, denoising, and normalization. First, since the outliers in the measured data hinder model training and convergence, these outliers are identified using the 3-sigma method and are removed [42]. The 3-sigma is based on the normal distribution.

In a normal distribution, the mean is μ and the standard deviation is σ. They are calculated as

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n}, \tag{12}$$

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n-1}}, \tag{13}$$

where x denotes the parameter value, n is the number of parameter values in the data sequence. Based on the 3-sigma principle, approximately 88.9% of the data lie within the range $[\mu - 3\sigma, \mu + 3\sigma]$. Therefore, any data point x such that $x < \mu - 3\sigma$ or $x > \mu + 3\sigma$ is considered an outlier. When there is a large amount of noise in drilling data, $H\infty$ Control can be effectively used to suppress these interferences [43]. The singular value decomposition algorithm can be used to calibrate the sensors of monitoring parameters, which can improve the data quality [44]. Second, since the fluctuations in the parameter curves due to noise hinder the accurate determination of the monitoring parameter trends, a delay-free denoising method based on the heat function for time series data is employed to smooth the parameter curves [45]. The heat function for time series data is

$$x_i = x_i + 0.25 * (x_{i+1} + x_{i-1} - 2 * x_i), \tag{14}$$

where x_i represents the value of the i-th element in a parameter data sequence. Each element in the original data sequence and its two neighboring elements are inputted into the heat equation for the calculation to obtain a new data sequence. Then, the previous step is repeated with continuous iteration of computation. The optimal number of iterations is 20, which is determined by trial-and-error experiments. After 20 iterations of computation, the noise reduction is finished. Compared with denoising methods based on wavelet transforms or the moving average filters, the heat function causes less delay and produces a smoother sequence. Figure 9 shows the total active pit volume curves before and after smoothing. Specifically, when the moving average filter is applied for denoising, a delay in the parameter trend is observed. Similarly, wavelet transform denoising leaves some noise in the data sequence unremoved. In contrast, the heat function effectively eliminates the instantaneous fluctuations in the data sequence while preserving its fundamental trend.

Since drilling parameters have different numerical ranges in different oilfields or at different well depths, the min–max normalization method is used to eliminate the units and dimensions of the drilling parameters [46]. The formula of min–max normalization is

$$\overline{x_i} = (x_i - x_{min})/(x_{max} - x_{min}), \tag{15}$$

where x_i represents the value of the i-th element in a parameter data sequence, $\overline{x_i}$ is the normalized value of the i-th element in parameter data sequence, and x_{max} and x_{min} denote the maximum and minimum values of all elements in parameter data sequence, respectively.

3. Data labeling: In the kick detection model, the sample labels must be numeric data. Therefore, the "kick" or "non-kick" samples are labeled by numerical labels of "01" or "10" using one-hot encoding. Then all samples are labeled according to drilling logs and expert knowledge. Among 17,745 samples, 8014 samples are labeled as "kick", and 9731 samples are labeled as "non-kick".
4. Data splitting: To evaluate the generalization ability of the proposed method, 1/5 labeled samples are randomly selected as the test samples according to the common 8:2 ratio. The remaining labeled samples are divided randomly into a training dataset

and a validation dataset according to the common 8:2 ratio. The training dataset is used for model training, the validation dataset is used to optimize the hyperparameters of the model, and the testing dataset is used to evaluate the performance of the model. In total, there are 11,357 training samples, 2839 validation samples, and 3549 test samples.

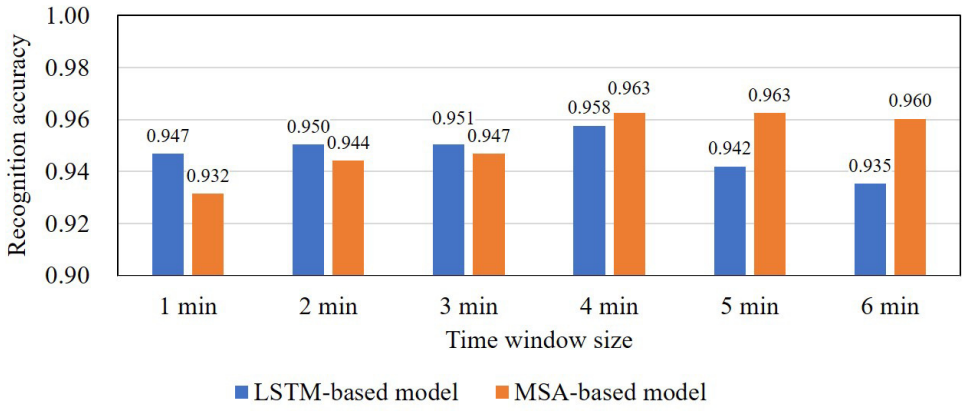


Figure 8. Recognition accuracies of models for time windows of different sizes.

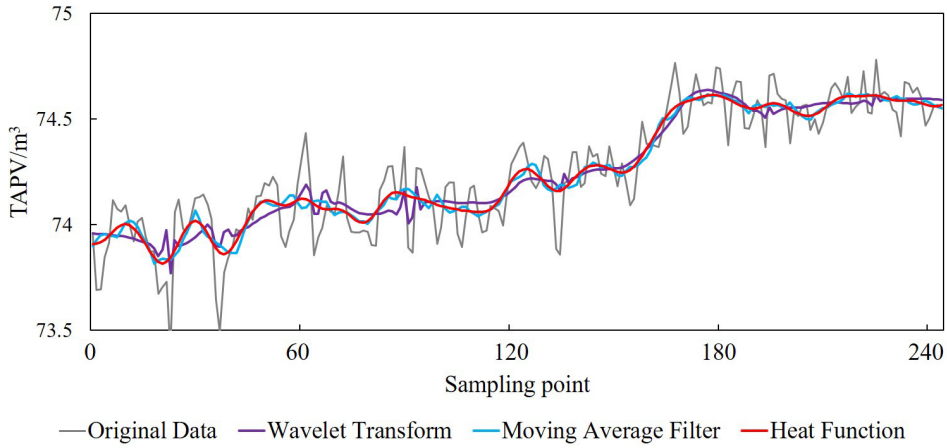


Figure 9. TAPV curves before and after denoising.

4.2. Model Building

The MSA-based and LSTM-based methods are employed to establish the relationship models between drilling parameters and kick accidents to compare the recognition accuracies of the two methods. Both models utilize the artificial neural network, which consists of an input layer, hidden layers, and an output layer, with multiple neural units in each layer. Therefore, the number of neurons in the model requires to be determined during model building.

In the MSA-based model, the number of neural units in each layer, except for the ReLU layer, is determined based on the size of the input or output matrix. Since ReLU is a nonlinear function, the ReLU layer can be called the nonlinear layer. To determine the number of neural units in the nonlinear layer of two models, a grid search approach is employed to try different values and select the number that leads to the highest accuracy in feature classification. The candidate values for the number of neural units are 20, 40, 60, 80, and 100 [47]. Table 4 lists the recognition accuracies of models with different numbers of neural units. The numbers of neural units in each model are listed in Table 5.

Table 4. Recognition accuracies of models with different numbers of neural units.

Model	Number of Neural Units in the Hidden Layer				
	20	**40**	**60**	**80**	**100**
LSTM-based	0.946	0.949	0.955	0.959	0.937
MSA-based	0.964	0.960	0.961	0.963	0.959

Table 5. Number of neural units in MSA-based and LSTM-based models.

Model	Input Layer	Hidden Layers					Output Layer
		Concat	**Nonlinear**	**Linear**	**Time FC**	**Feature FC**	
LSTM-based	6	/	80	/	/	2	2
MSA-based	6	6	20	6	240	2	2

4.3. Model Evaluation

To comprehensively compare the performances of the MSA-based model and the LSTM-based model, five commonly used evaluation metrics are employed, including Accuracy, Precision, Recall, F1-score, and AUC (area under the curve) value. Larger values of these metrics indicate better performance of the models. Additionally, they are derived metrics calculated based on the confusion matrix. A confusion matrix includes true positive (TP), true negative (TN), false positive (FP), and false negative (FN), as shown in Table 6.

Table 6. Confusion matrix for kick recognition.

Actual Labels	Predicted Labels	
	Kick	**Non-Kick**
Kick	TP	FN
Non-kick	FP	TN

The calculated formulas of Accuracy, Precision, Recall, and F1-score are

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN}, \tag{16}$$

$$recall = \frac{TP}{TP + FN}, \tag{17}$$

$$precision = \frac{TP}{TP + FP}, \tag{18}$$

$$F1\text{-}score = \frac{2TP}{2TP + FN + FP}. \tag{19}$$

The AUC value represents the area under the ROC curve. The ROC curve is plotted with False Positive Rate (x-axis) against Recall (y-axis). The False Positive Rate is calculated by

$$FPR = \frac{FP}{TN + FP}. \tag{20}$$

The computational complexity of the proposed model is evaluated using Big-O notation. The computational complexity of the multi-head self-attention module is $O(n^2 * d)$, where n is the length of the data sequence and d denotes the number of input monitoring parameters. The computational complexity of a feed-forward network is $O(n * d * df)$, where df is the dimension of the nonlinear mapping space. Due to df being a multiple of d, the computational complexity of the feed-forward network can be simplified to $O(n * d^2)$.

In a two-dimensional classifier, the computational complexities of two fully connected layers are $O(n*d*1)$ and $O(n*1)$. The computational complexity of the two-dimensional classifier is $O(n*d)$ due to an evaluation of the upper limit. In the proposed model, the computational complexities of the three modules are $O(n^2*d)$, $O(n*d^2)$, and $O(n*d)$. Due to the sequence length n being much larger than the number of parameters d, the computational complexity of the multi-head self-attention network is $O(n^2*d)$.

5. Results and Discussion

To validate the improvement in accuracy of the kick detection model achieved by the designed multi-head structure or self-attention mechanism, ten models are designed for experiments with measured data, including a LSTM-based model, a multi-head LSTM-based (mLSTM-based) model, a GRU-based model, a multi-head GRU-based (mGRU-based) model, a CNN-based model, a multi-head CNN-based (mCNN-based) model, an adaptive boosting (AdaBoost) model, a gradient boosting decision tree (GBDT) model, a self-attention-based model, and an MSA-based model. The structure of the AdaBoost model refers to the research in the article [48]. The GBDT model is constructed using the method described in the article [49]. Furthermore, to validate that the proposed method can adaptively pay more attention to trends of monitoring parameters, simulation data are created to mimic the changes in response times of the parameters. A proposed model is trained using the whole data sequences, and an LSTM-based model is trained using the data sequences during parameter changes.

5.1. Experimental Results of Measured Data

The ten models are trained using the training dataset, and the length of the parameter data sequences in the training samples is 240. The training steps for the models are as follows:

1. Forward Propagation: Each training sample is inputted into the models. The probabilities of the samples belonging to the "kick" and "non-kick" categories are calculated.
2. Backward Propagation: The model parameters are updated according to the loss function of the model's outputs. The cross-entropy loss function is commonly used in classification tasks to measure the discrepancy between the model's outputs and the actual labels. The computation formula is

$$H(\mathbf{E}, \mathbf{Y}) = -(e_1 log y_1 + e_2 log y_2 + e_1 log y_2 + e_2 log y_1), \tag{21}$$

where $\mathbf{E}$ represents the expected probability matrix, e_1 is the expected probability for kick, e_2 denotes the expected probability for non-kick, $\mathbf{Y}$ represents the predicted probability matrix, y_1 is the predicted probability for kick, and y_2 denotes the predicted probability for non-kick. The network parameters (weights and biases) in the multi-head self-attention module, feed-forward network module, and classifier are updated using gradient descent as

$$p' = p - \frac{\partial H}{\partial p} \tag{22}$$

where p' represents the updated network parameters, p is the network parameters before updating, and H denotes the cross-entropy loss function.

3. Iterative Training: An iteration of training is completed when all the training samples have gone through forward and backward propagation. Then multiple epochs of iterations are performed on the entire training dataset to gradually optimize the network parameters.

4. Early Stopping: After an iteration of training, the average error of model outputs is calculated on the validation dataset using the loss function. If the error is greater than the error calculated in the previous iteration, the model training is stopped.

The test samples are inputted into the trained models in a randomized order. Through testing, the time for the trained micro-kick detection model to process six parameter data sequences each with a length of 240 is 2.13 ms, which can meet the real-time monitoring requirements at the well site. Evaluation metrics are calculated according to the predicted results of the models. Table 7 presents the five model evaluation metrics for the ten models.

Figure 10 shows the five model evaluation metrics for the ten models on the testing dataset. Among them, the MSA-based model has the highest accuracy. The performance of the GRU-based method is comparable to that of the LSTM-based method, coinciding with findings from other fields. The GRU architecture offers advantages due to its simpler structure, which enhances computational efficiency and reduces the risk of overfitting. The CNN-based model exhibits higher recognition accuracy than both LSTM-based and GRU-based models. The reason is that the CNN-based model can analyze longer data sequences due to the data compression capabilities of the one-dimensional convolutional kernel. However, the CNN-based model does not pay more attention to early or long-term changes in parameters, as it assigns equal attentional weight to the drilling data at each moment in time. Consequently, the recognition accuracy of the CNN-based model is lower than that of the MSA-based method. The AdaBoost and GBDT models are overfitting due to their stronger fitting capabilities. Therefore, they achieve an accuracy of 99% on the training set, but their performance on the test set was inferior to those of other models. The self-attention-based models have higher recall (i.e., sensitivity) compared with the LSTM-based, GRU-based, and CNN-based models, indicating that the self-attention mechanism effectively reduces the false-negative rate of the intelligent kick detection method. Moreover, the multi-head models have higher precision (i.e., specificity) than models without the multi-head structure. This means that the multi-head model has a lower false-positive rate. The comprehensive performance of the models is assessed by two evaluation metrics, F1-score and AUC. Therefore, the MSA-based model has the best monitoring performance.

Table 7. Model evaluation metrics for the ten models.

Model	Accuracy	Recall	Precision	F1-Score	AUC
AdaBoost	0.810	0.815	0.902	0.857	0.842
GBDT	0.793	0.808	0.885	0.844	0.800
LSTM-based	0.886	0.939	0.903	0.920	0.849
mLSTM-based	0.903	0.917	0.945	0.930	0.894
GRU-based	0.887	0.885	0.949	0.916	0.889
mGRU-based	0.909	0.902	0.965	0.933	0.893
CNN-based	0.910	0.891	0.967	0.927	0.871
mCNN-based	0.921	0.917	0.977	0.946	0.883
Attention-based	0.920	0.999	0.911	0.950	0.870
MSA-based	0.943	0.998	0.927	0.961	0.905

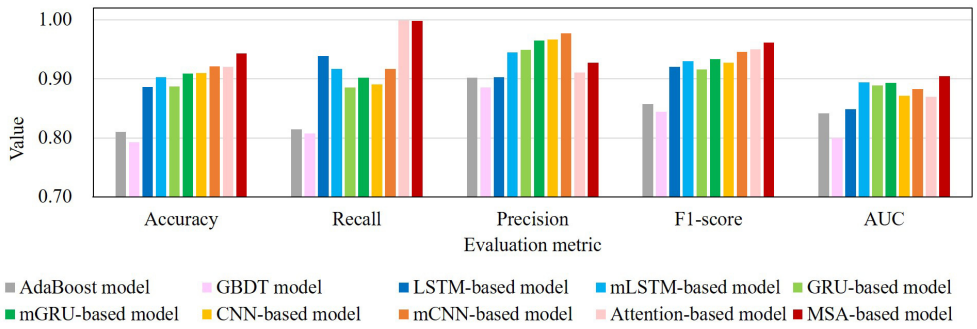


Figure 10. Evaluation metrics for the ten models on the testing dataset.

Figure 11 shows the curves and attention weights of monitoring parameters in a micro-kick case. The monitoring parameters with rapid trends have higher weights. For example, the attention weights of the rapidly changing WOB, ROP, and SPP are higher than those of torque, DFO, and TAPV.

5.2. Experimental Results of Simulation Data

To verify that the proposed method can pay more attention to the trends of monitoring parameters, an LSTM network is trained using the data sequences during parameter changes. To capture the trends of monitoring parameters, different sizes of sliding time windows are employed to obtain parameter data sequences at different moments. The multiple LSTM networks are trained using these different data sequences. The LSTM network with the highest classification accuracy is chosen as a control group.

To remove the impact of the number of training samples on the performance of the model, an equal number of training samples is first obtained by window slicing using the largest time window (4 min). Subsequently, a secondary window slicing for these samples is performed using sliding time windows of different sizes. After the data preprocessing and data labeling, a total of 30 datasets are created using 10 different window sizes and 3 combinations of parameters. Each dataset is divided into training and testing datasets in an 8:2 ratio. Finally, LSTM-based models are trained on the training datasets, and the testing dataset is fed into the trained model to calculate the accuracy of the model. Table 8 lists the recognition accuracies of the models on each dataset.

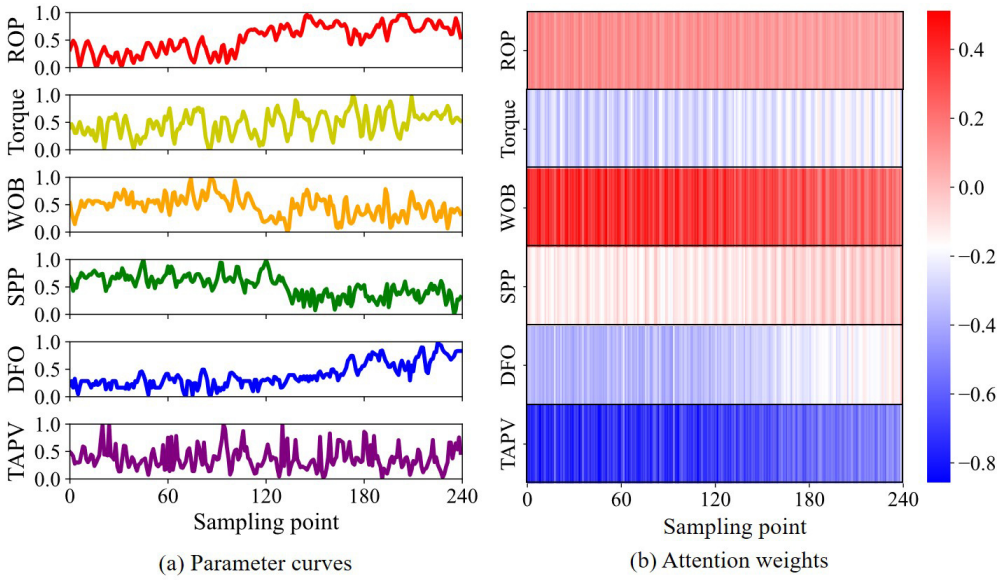


Figure 11. Curves and attention weights of monitoring parameters.

Table 8. Accuracies of LSTM-based models on diverse data sequences.

Data Sequences	Drilling Parameters		
	ROP and Torque	**WOB and SPP**	**DFO and TAPV**
00:00:00–00:01:00	0.693	0.691	0.778
00:00:00–00:02:00	0.698	0.704	0.825
00:00:00–00:03:00	0.702	0.629	0.861
00:00:00–00:04:00	0.779	0.691	0.873
00:01:00–00:02:00	0.698	0.703	0.754
00:01:00–00:03:00	0.705	0.661	0.828
00:01:00–00:04:00	0.859	0.758	0.827
00:02:00–00:03:00	0.694	0.694	0.767
00:02:00–00:04:00	0.706	0.703	0.755
00:03:00–00:04:00	0.695	0.698	0.697

The experimental results show that changes in drilling parameters such as ROP, Torque, WOB, and SPP occurred within the previous 3 min. Changes in DFO and TAPV occurred within the previous 4 min. The LSTM-based model with six monitoring parameters is trained on these selected short data sequences.

Two simulation testing datasets are created to test the impact of changes in the response time of parameters on the recognition results of the intelligent kick detection models. In one dataset, the response times of DFO and APV are advanced by 30 s to simulate sudden kick accidents, while in the other dataset, these response times are delayed by 30 s to simulate micro-kick accidents. LSTM-based or MSA-based models are trained on real training datasets, and the accuracies of these models are computed on simulation testing datasets. Figure 12 shows the evaluation metrics of the LSTM-based and MSA-based models trained using the long data sequences and the LSTM-based model trained using the selected short data sequences.

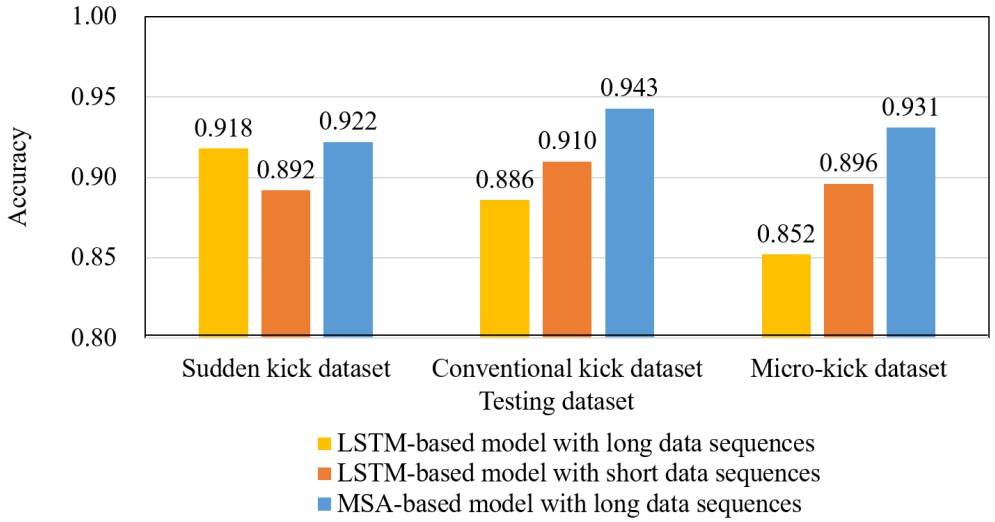


Figure 12. Accuracies of three trained models on three testing datasets.

The experimental results show that the accuracy is improved by using the selected short data sequences to train the LSTM-based model when the parameter response times remain constant. Additionally, the LSTM-based model trained using the long data sequences is most affected by changes in the parameter response times. There is a negative correlation between the model's accuracy and the difference between the response times of asynchronous monitoring parameters. Moreover, the accuracy of the MSA-based model is higher than those of both LSTM-based models. This indicates that the MSA-based model

can automatically and accurately capture the trend of the asynchronous parameters even if the response times of parameters have changed. Therefore, early or long-term signs of micro-kick accidents can be accurately identified using the MSA-based model, leading to more accurate micro-kick detection.

6. Conclusions

The deep learning-based intelligent kick detection method is employed to monitor kick accidents by comprehensively detecting the trends in multiple monitoring parameters. However, when a micro-kick occurs, the differences in response times between different monitoring parameters increase, leading to a long-term dependency problem in the deep learning-based method. These methods cannot remember and capture the trends of monitoring parameters from a long time ago. In the proposed intelligent micro-kick detection method, the similarities between the parameter values at each moment and those at other moments are computed in parallel using the self-attention mechanism. A small similarity indicates that the parameter is changing at this moment, and then a bigger attention weight is assigned to the value at this moment. Additionally, a designed multi-head structure is employed to analyze the different types of parameter trends, such as rapidly increasing trends, rapidly decreasing trends, and gradually increasing trends. Finally, the nonlinear relationship between the trend features of monitoring parameters and kick accidents is modeled in both time and feature dimensions.

The effectiveness of the multi-head structure and self-attention mechanism in improving the accuracy of kick detection methods is validated by using the measured drilling data. Compared with the LSTM-based method, the accuracy of the MSA-based method is improved by 5.7%. The accuracy of micro-kick monitoring is improved by 7.9%. Based on the experimental results, the following conclusions are drawn:

1. The recognition accuracy of the dominant LSTM-based kick detection models is correlated with the response time difference between drilling parameters. The larger the time difference, the lower the accuracy.
2. The self-attention mechanism facilitates the timely extraction of early trend features from multiple asynchronous monitoring parameters, thereby reducing the false-negative rate of the intelligent kick detection method.
3. The fast-increasing trend features of ROP and torque, fast-decreasing trend features of WOB and SPP, and slow-increasing trend features of DFO and APV are accurately extracted using the multi-head structure, thereby reducing the false-positive rate of the intelligent kick detection method.

The generalization ability of the proposed model is affected by the diversity of the training data. When the training data lack sufficient diversity, the model may overfit and thus fail to be applied to drilling data with different data distributions. Although the drilling data are collected from wells in different blocks of the same oilfield to enhance data diversity, it still cannot fully represent the diversity of the real-world environments. Since the measured drilling data are not publicly available and difficult to collect, more kick data under different environments need to be simulated using data generation technologies. Additionally, data transfer techniques should be employed to reduce the distribution discrepancies between drilling data collected from different wells to improve the applicability of the proposed micro-kick detection model. When the proposed intelligent micro-kick detection model is applied to a new dataset, this method becomes more sensitive and categorizes ambiguous samples as kick samples. Since there is an inverse relationship between specificity and sensitivity, false alarms may arise when applying the proposed model to new wells. In the samples that make false alarms of the model, some parameters have the same trend as when a kick occurs. Since the model can monitor kicks and update the alarm status

in real time, false alarms usually last for a very short time. Therefore, it is recommended that interventions be taken only after the model gives continuous alarms. To reduce false alarms when applying the model to new wells, further exploration of techniques such as transfer learning or domain adaptation is needed to enable the proposed model to adapt to diverse drilling data. The generalization ability of the kick detection model still requires further improvement using advanced techniques such as data augmentation, transfer learning, and meta-learning.

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

Funding: This research was funded by the National Natural Science Foundation of China (41974144, 42274159, and 42472378).

Data Availability Statement: Restrictions apply to the availability of these data. Data were obtained from Sinopec Shengli Oilfield and are available from the Weifeng Sun with the permission of Sinopec Shengli Oilfield.

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

Abbreviations

The following abbreviations are used in this manuscript:

AdaBoost	adaptive boosting
ANN	artificial neural network
AUC	area under the curve
BN	Bayesian network
CNN	convolutional neural network
DFO	differential flow out
DMA	divergence of moving average
DT	decision tree
FC	fully connected
FN	false negative
FP	false positive
GBDT	gradient boosting decision tree
GRU	gate recurrent unit
KNN	K-nearest neighbor
LSTM	long short-term memory
MSA	multi-head self-attention
RNN	recurrent neural network
ROP	rate of penetration
SPP	Standpipe pressure
SVM	support vector machine
TAPV	total active pit volume
TN	true negative
TP	true positive
WOB	weight on bit

References

1. Sule, I.O.; Khan F.; Butt S. Experimental investigation of gas kick effects on dynamic drilling parameters. *J. Pet. Explor. Prod. Technol.* **2019**, *9*, 605–616. [CrossRef]
2. Blue, D.; Blakey, T.; Rowe, M. Advanced mud logging: Key to safe and efficient well delivery. In Proceedings of the Offshore Technology Conference, Houston, TX, USA, 6–9 May 2019; p. D032S058R003. [CrossRef]

3. Yalamarty, S.S.; Singh, K.; Kamyab, M.; Cheatham, C.; Crkvenjakov, V.; Flurry, K. Early detection of well control kick events by applying data analytics on real time drilling data. In Proceedings of the IADC/SPE International Drilling Conference and Exhibition, Galveston, TX, USA, 8–10 March 2022; p. D021S016R002. [CrossRef]
4. Lafond, A.; Leblay, F.; Roguin, G.; Ringer, M. Automated Influx and Loss Detection System Based on Advanced Mud Flow Modeling. In Proceedings of the SPE Annual Technical Conference and Exhibition, Calgary, AB, Canada, 30 September–2 October 2019; p. D012S058R001. [CrossRef]
5. Wang, B.; Li, J.; Zhang, G.; Li, Y.; Huang, H.; Zhan, J.; Yang, H. A novel early gas kick monitoring method using the difference between downhole dual measurement points pressure and a genetic algorithm-based model. *Geoenergy Sci. Eng.* **2023**, *231*, 212371.. [CrossRef]
6. Ye, Y.; Fan, H.; Liu, Y.; Tao, Z.; Diao, H.; Zhou, F. A New Method of Kick Evaluation Based on Energy Analysis. In Proceedings of the SPE/IADC Middle East Drilling Technology Conference and Exhibition, Abu Dhabi, United Arab Emirates, 23–25 May 2023; p. D011S006R004. [CrossRef]
7. Dow, B.; Ferrando, P.; Abolins, N.I.; Leonard, T.; Abuelaish, A.; Gallinal Cuenca, N.; Hansen, J.; Rojas Rodriguez, F. Advancing Influx Detection Toward Automated Well Control. In Proceedings of the IADC/SPE International Drilling Conference and Exhibition, Galveston, TX, USA, 8–10 March 2022; p. D021S016R003. [CrossRef]
8. He, M.; Chen, X.; Xu, M.; Chen, H. Inversion-based model for quantitative interpretation by a dual-measurement points in managed pressure drilling. *Process Saf. Environ. Prot.* **2022**, *165*, 969–976. [CrossRef]
9. Zhang, Z.; Xiong, Y.; Mao, L.; Lu, J.; Wang, M.; Peng, G. Transient Temperature Prediction Models of Wellbore and Formation in Well-Kick Condition during Circulation Stage. *J. Pet. Sci. Eng.* **2019**, *175*, 266–279. [CrossRef]
10. Tang, H.; Zhang, S.; Zhang, F.; Venugopal, S. Time Series Data Analysis for Automatic Flow Influx Detection during Drilling. *J. Pet. Sci. Eng.* **2019**, *172*, 1103–1111. [CrossRef]
11. Mao, Y.; Zhang, P. An Automated Kick Alarm System Based on Statistical Analysis of Real-Time Drilling Data. In Proceedings of the Abu Dhabi International Petroleum Exhibition and Conference, Abu Dhabi, United Arab Emirates, 11–14 November 2019; p. D021S032R004. [CrossRef]
12. Zhu, J.; Chen, G.; Yin, Z.; Khan, F.; Meng, X. An Integrated Methodology for Dynamic Risk Evaluation of Deepwater Blowouts. *J. Loss Prev. Process Ind.* **2022**, *74*, 104647. [CrossRef]
13. Muojeke, S.; Venkatesan, R.; Khan, F. Supervised Data-Driven Approach to Early Kick Detection during Drilling Operation. *J. Pet. Sci. Eng.* **2020**, *192*, 107324. [CrossRef]
14. Duan, S.; Song, X.; Cui, Y.; Xu, Z.; Liu, W.; Fu, J.; Zhu, Z.; Li, D. Intelligent Kick Warning Based on Drilling Activity Classification. *Geoenergy Sci. Eng.* **2023**, *222*, 211408. [CrossRef]
15. Nhat, D. M.; Venkatesan, R.; Khan, F. Data-Driven Bayesian Network Model for Early Kick Detection in Industrial Drilling Process. *Process Saf. Environ. Prot.* **2020**, *138*, 130–138. [CrossRef]
16. Chen, K.; Wei, X.; Li, H.; Lin, H.; Khan, F. Operational Risk Analysis of Blowout Scenario in Offshore Drilling Operation. *Process Saf. Environ. Prot.* **2021**, *149*, 422–431. [CrossRef]
17. Meng, X.; Zhu, J.; Chen, G.; Shi, J.; Li, T.; Song, G. Dynamic and Quantitative Risk Assessment under Uncertainty during Deepwater Managed Pressure Drilling. *J. Clean. Prod.* **2022**, *334*, 130249. [CrossRef]
18. Zhang, W.; Meng, X.; Zhang, W.; Zhu, J.; Chen, G. Dynamic Risk Assessment of Deepwater Drilling Using Data-Based and Probabilistic Approach. *Ocean Eng.* **2023**, *268*, 113414. [CrossRef]
19. Alouhali, R.; Aljubran, M.; Gharbi, S.; Al-yami, A. Drilling Through Data: Automated Kick Detection Using Data Mining. In Proceedings of the SPE International Heavy Oil Conference and Exhibition, Kuwait City, Kuwait, 10–12 December 2018; p. D012S018R001. [CrossRef]
20. Balaji, K.; Rabiei, M.; Suicmez, V.; Canbaz, C.H.; Agharzeyva, Z.; Tek, S.; Bulut, U.; Temizel, C. Status of Data-Driven Methods and Their Applications in Oil and Gas Industry. In Proceedings of the SPE Europec Featured at 80th EAGE Conference and Exhibition, Copenhagen, Denmark, 11–14 June 2018; p. D031S005R007. [CrossRef]
21. Liang, H.; Han, H.; Ni, P.; Jiang, Y. Overflow Warning and Remote Monitoring Technology Based on Improved Random Forest. *Neural Comput. Appl.* **2021**, *33*, 4027–4040. [CrossRef]
22. Chi, P.; Qingfeng, L.; Jianhong, F.; Yun, Y.; Xiaomin, Z.; Yu, S.; Zhaoyang, X.; Chengxu, Z.; Pengcheng, W. An Intelligent Model for Early Kick Detection Based on Cost-Sensitive Learning. *Process Saf. Environ. Prot.* **2023**, *169*, 398–417. [CrossRef]
23. Wang, Z.; Chen, G.; Zhang, R.; Zhou, W.; Hu, Y.; Zhao, X.; Wang, P. Early Monitoring of Gas Kick in Deepwater Drilling Based on Ensemble Learning Method: A Case Study at South China Sea. *Process Saf. Environ. Prot.* **2023**, *169*, 504–514. [CrossRef]
24. Fjetland, A.K.; Zhou, J.; Abeyrathna, D.; Gravdal, J.E. Kick Detection and Influx Size Estimation during Offshore Drilling Operations Using Deep Learning. In Proceedings of the 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA), Xi'an, China, 19–21 June 2019; pp. 2321–2326. [CrossRef]
25. Osarogiagbon, A.; Muojeke, S.; Venkatesan, R.; Khan, F.; Gillard, P. A New Methodology for Kick Detection during Petroleum Drilling Using Long Short-Term Memory Recurrent Neural Network. *Process Saf. Environ. Prot.* **2020**, *142*, 126–137. [CrossRef]

26. Yin, Q.; Yang, J.; Tyagi, M.; Zhou, X.; Hou, X.; Wang, N.; Tong, G.; Cao, B. Machine Learning for Deepwater Drilling: Gas-Kick-Alarm Classification Using Pilot-Scale Rig Data with Combined Surface-Riser-Downhole Monitoring. *SPE J.* **2021**, *26*, 1773–1799. [CrossRef]
27. Duan, M.; Wei, Q.; Yang, G.; He, X.; Xue, Q.; Feng, Y. Research on Drilling Overflow Warning Method Based on Logging Data and LSTM. In Proceedings of the 2023 8th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA), Chengdu, China, 26–28 April 2023; pp. 106–112. [CrossRef]
28. Zhang, D.; Sun, W.; Dai, Y.; Liu, K.; Li, W.; Wang, C. A Hierarchical Early Kick Detection Method Using a Cascaded GRU Network. *Geoenergy Sci. Eng.* **2023**, *222*, 211390. [CrossRef]
29. Zhang, D.; Sun, W.; Dai, Y.; Bu, S.; Feng, J. Intelligent Kick Detection Method Using Cascaded GRU Network with Adaptive Monitoring Parameters. In *International Field Exploration and Development Conference*; Springer Nature: Singapore, 2023; pp. 215–224.
30. Zhang, Z.; Lai, X.; Du, S.; Yu, W.; Wu, M. Early Warning of Loss and Kick for Drilling Process Based on Sparse Autoencoder with Multivariate Time Series. *IEEE Trans. Ind. Inf.* **2023**, *19*, 11019–11029. [CrossRef]
31. Zhu, Z.; Zhou, D.; Yang, D.; Song, X.; Zhou, M.; Zhang, C.; Duan, S.; Zhu, L. Early Gas Kick Warning Based on Temporal Autoencoder. *Energies* **2023**, *16*, 4606. [CrossRef]
32. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762.
33. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2021**, arXiv.2010.11929.
34. Inaguma, H.; Kiyono, S.; Duh, K.; Karita, S.; Soplin, N.E.Y.; Hayashi, T.; Watanabe, S. ESPnet-ST: All-in-One Speech Translation Toolkit. *arXiv* **2020**, arXiv.2004.10234.
35. Gulati, A.; Qin, J.; Chiu, C.-C.; Parmar, N.; Zhang, Y.; Yu, J.; Han, W.; Wang, S.; Zhang, Z.; Wu, Y.; Pang, R. Conformer: Convolution-Augmented Transformer for Speech Recognition. *arXiv* **2020**, arXiv.2005.08100.
36. Lin, T.; Wang, Y.; Liu, X.; Qiu, X. A Survey of Transformers. *AI Open* **2022**, *3*, 111–132. [CrossRef]
37. Wang, J.; Sajeev, S.K.; Ozbayoglu, E.; Baldino, S.; Liu, Y.; Jing, H. Reducing NPT Using a Novel Approach to Real-Time Drilling Data Analysis. In Proceedings of the SPE Annual Technical Conference and Exhibition, San Antonio, TX, USA, 16–18 October 2023; p. D021S012R005.. [CrossRef]
38. Zhang, C.; Weng, N.; Chang, J.; Zhou, A. Detecting Abnormal Trend Evolution over Multiple Data Streams. In *Advances in Data and Web Management*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 285–296
39. Kazemnejad, A.; Padhi, I.; Ramamurthy, K. N.; Das, P.; Reddy, S. The Impact of Positional Encoding on Length Generalization in Transformers. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 24892–24928.
40. Yin, Q.; Yang, J.; Tyagi, M.; Zhou, X.; Wang, N.; Tong, G.; Xie, R.; Liu, H.; Cao, B. Downhole Quantitative Evaluation of Gas Kick during Deepwater Drilling with Deep Learning Using Pilot-Scale Rig Data. *J. Pet. Sci. Eng.* **2022**, *208*, 109136. [CrossRef]
41. Guennec, A. L.; Malinowski, S.; Tavenard, R. Data Augmentation for Time Series Classification Using Convolutional Neural Networks. In Proceedings of the ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data, Riva Del Garda, Italy, 19–23 September 2016.
42. Du, H.; Zhao, S.; Zhang, D. Robust Local Outlier Detection. In Proceedings of the 2015 IEEE International Conference on Data Mining Workshop (ICDMW), Atlantic City, NJ, USA, 14–17 November 2015; pp. 116–123. [CrossRef]
43. Zhao, J.; Wang, Z.; Lv, Y.; Na, J.; Liu, C.; Zhao, Z. Data-Driven Learning for $H\infty$ Control of Adaptive Cruise Control Systems. *IEEE Trans. Veh. Technol.* **2024**, *73*, 18348–18362. [CrossRef]
44. Su, H.; Qi, W.; Hu, Y.; Sandoval, J.; Zhang, L.; Schmirander, Y.; Chen, G.; Aliverti, A.; Knoll, A.; Ferrigno, G.; et al. Towards model-free tool dynamic identification and calibration using multi-layer neural network. *Sensors* **2019**, *19*, 3636. [CrossRef] [PubMed]
45. Zhang, D.; Sun, W.; Dai, Y.; Bu, S.; Feng, J.; Huang, W. Intelligent Kick Detection Using a Parameter Adaptive Neural Network. *Geoenergy Sci. Eng.* **2024**, *234*, 212694. [CrossRef]
46. Borkin, D.; Némethová, A.; Michaľčonok, G.; Maiorov, K. Impact of Data Normalization on Classification Model Accuracy. *Res. Pap. Fac. Mater. Sci. Technol. Slovak Univ. Technol.* **2019**, *27*, 79–84. [CrossRef]
47. Yin, Q.; Yang, J.; Tyagi, M.; Zhou, X.; Hou, X.; Cao, B. Field Data Analysis and Risk Assessment of Gas Kick during Industrial Deepwater Drilling Process Based on Supervised Learning Algorithm. *Process Saf. Environ. Prot.* **2021**, *146*, 312–328. [CrossRef]

48. Kang, Q.; Li, K.; Fu, J.; Liu, Y. Hybrid LBM and machine learning algorithms for permeability prediction of porous media: A comparative study. *Comput. Geotech.* **2024**, *168*, 106163. [CrossRef]
49. Li, K.; Horton, R.; He, H. Application of machine learning algorithms to model soil thermal diffusivity. *Int. Commun. Heat Mass Transf.* **2023**, *149*, 107092. [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.