

An unsupervised intelligent warning model for drilling kick risk based on multi-temporal feature coupling

De-Tao Zhou^a, Zhao-Peng Zhu^{b,d,*}, Tao Pan^c, Xian-Zhi Song^{c,d}, Shi-Jie Xiao^e,
Gen-Sheng Li^{a,c}, Cheng-Kai Zhang^c, Chen-Zhan Zhou^c, Zi-Yue Zhang^c

^a College of Artificial Intelligence, China University of Petroleum (Beijing), Beijing, 102249, China

^b College of Mechanical and Transportation Engineering, China University of Petroleum (Beijing), Beijing, 102249, China

^c College of Petroleum Engineering, China University of Petroleum (Beijing), Beijing, 102249, China

^d Research Center for Intelligent Drilling & Completion Technology and Equipment, China University of Petroleum (Beijing), Beijing, 102249, China

^e Engineering Technology Research Institute, CNPC Great Wall Drilling Company, Panjin, 124010, Liaoning, China

ARTICLE INFO

Article history:

Received 20 February 2025

Received in revised form

5 June 2025

Accepted 19 June 2025

Available online 9 July 2025

Edited by Jia-Jia Fei

Keywords:

Kick warning

Graph autoencoder

Field engineer operations

False alarms

Weighted dynamic time warping

ABSTRACT

As oil and gas exploration continues to progress into deeper and unconventional reservoirs, the likelihood of kick risk increases, making kick warning a critical factor in ensuring drilling safety and efficiency. Due to the scarcity of kick samples, traditional supervised models perform poorly, and significant fluctuations in field data lead to high false alarm rates. This study proposes an unsupervised graph autoencoder (GAE)-based kick warning method, which effectively reduces false alarms by eliminating the influence of field engineer operations and incorporating real-time model updates. The method utilizes the GAE model to process time-series data during drilling, accurately identifying kick risk while overcoming challenges related to small sample sizes and missing features. To further reduce false alarms, the weighted dynamic time warping (WDTW) algorithm is introduced to identify fluctuations in logging data caused by field engineer operations during drilling, with real-time updates applied to prevent normal conditions from being misclassified as kick risk. Experimental results show that the GAE-based kick warning method achieves an accuracy of 92.7% and significantly reduces the false alarm rate. The GAE model continues to operate effectively even under conditions of missing features and issues kick warnings 4 min earlier than field engineers, demonstrating its high sensitivity and robustness. After integrating the WDTW algorithm and real-time updates, the false alarm rate is reduced from 17.3% to 5.6%, further improving the accuracy of kick warnings. The proposed method provides an efficient and reliable approach for kick warning in drilling operations, offering strong practical value and technical support for the intelligent management of future drilling operations.

© 2025 The Authors. Publishing services by Elsevier B.V. on behalf of KeAi Communications Co. Ltd. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

1. Introduction

As the exploration and development of oil and gas resources gradually shift toward deeper and unconventional reservoirs fields, drilling faces challenges such as high temperature and high pressure, narrow safe density windows, and other issues. These challenges have significantly increased the kick risk and other drilling hazards, which can easily trigger severe blowout accidents (Vandenbussche et al., 2012). Therefore, early warning of kick risk

is extremely important, as they can effectively reduce downhole accidents and are critical for ensuring safe and efficient drilling operations (Griffin, 1967; Han et al., 2017). Currently, kick warning methods mainly include traditional early warning methods and intelligent early warning methods. Traditional methods suffer from poor timeliness and insufficient accuracy, while intelligent diagnostic methods, despite their broad application prospects, are still rarely applied in actual field operations, primarily due to the need for improved generalization capabilities. This situation highlights the practical significance of conducting in-depth research on intelligent early warning technologies for drilling kick.

Traditional early warning methods rely on kick warning by setting thresholds for single parameters. Liu (2006) developed a novel drilling fluid level monitoring and automatic grouting device

* Corresponding author.

E-mail address: zhuzp@cup.edu.cn (Z.-P. Zhu).

Peer review under the responsibility of China University of Petroleum (Beijing).

that utilizes ultrasonic waves to monitor changes in the fluid level of drilling mud tanks. Jiang et al. (2013) combined Coriolis mass flow technology with mud logging processes to establish an early kick detection (EKD) system. By comparing the flow rate changes at the inlet and outlet, they achieved fast and accurate kick warning and early warning. Wang et al. (2017) proposed an early kick warning method based on changes in drilling fluid resistivity (related to drilling fluid density and chloride ion content). This method demonstrated high sensitivity and real-time performance. Schafer et al. (1992) introduced an innovative rolling float flowmeter for precise monitoring of pipeline flow, showing excellent performance. Ablard et al. (2012) employed Coriolis flowmeters, which operate under low to medium pressure conditions and effectively eliminate temperature effects, allowing accurate measurement of oil, gas, and water mass flow, temperature variations, and density in pipelines. In recent years, with the significant advantages of artificial intelligence (AI) and big data technologies in solving complex problems involving multiple parameters and non-linear relationships, these technologies have achieved mature applications and rapid promotion in the energy sector. Research on kick early warning based on AI is currently in a stage of rapid development (Li et al., 2022). Researchers primarily use supervised and unsupervised learning algorithms for kick early warning, achieving certain results. Liang and Wang (2019) and Liang et al. (2020, 2019) predicted kick based on the correlation between kick incidents and casing pressure variation trends, utilizing clustering algorithms and neural network algorithms. Yin et al. (2021, 2022) defined five risk levels using three key indicators: flow rate difference, mud pit volume increment, and duration. They compared the performance of three algorithms: Long short-term memory neural networks (LSTM), recurrent neural networks (RNN), and sparse autoencoder-support vector machines. Song et al. (2021, 2022) conducted a comparative analysis of the kick and loss early warning performance of random forest, support vector machines, fully connected neural networks, and LSTM. They proposed a formation pore pressure prediction model based on the sequential characteristics of sedimentary sequences and drilling-logging-measurement multivariate data, combined with LSTM and BP neural networks, and successfully achieved accurate kick risk prediction. Zhang et al. (2024) classified kick warning parameters into dominant and auxiliary ones, and used a parameter adaptive neural network, which improved the accuracy by 12.8%. Chen et al. (2024) proposed a convolutional neural network (CNN)-based kick warning method for deepwater drilling, showing that this method can identify kick more quickly and accurately, with good model robustness. Zhu et al. (2023) utilized an unsupervised time-series intelligent model for kick warning, achieving an accuracy of 95%. Duan et al. (2023) first performed drilling condition classification and developed an intelligent kick early warning method based on random forest and artificial neural networks. Zhang and Samuel (2024) proposed a CNN-LSTM hybrid model for real-time early kick and loss prediction (EKLP), demonstrating its effectiveness in enhancing detection accuracy and operational safety during drilling operations. Sha et al. (2024) developed a real-time kick detection system utilizing artificial intelligence and real-time drilling data, achieving a 90% accuracy in early kick prediction during field applications. Qiao et al. (2024) developed a hybrid deep learning model combining convolutional neural networks (CNNs), gated recurrent units (GRUs), and an attention mechanism, with an accuracy of 98.64%.

In the early stages of a kick event, integrated field logging data typically exhibit complex interdependent changes and nonlinear fluctuations in time series. These features reflect the interactions of multiple factors and reveal the key dynamic patterns during the kick process. However, existing intelligent warning methods

typically rely on non-sequential point data modeling or fail to fully account for the interdependencies within multi-dimensional time-series data, which limits their ability to capture the underlying temporal dynamics effectively. Moreover, field warning often faces the issue of missing data features. For example, due to sensor malfunctions or data transmission delays, the outflow rate characteristics may be temporarily absent, which significantly impacts the warning accuracy of traditional methods and may even lead to model failure. Kick risk often suffer from a limited number of effective data samples, making it a typical small-sample event. To address these issues, the unsupervised learning-based graph autoencoder offers an innovative solution for kick warning. It not only captures the dynamic features of time-series data but also integrates multi-dimensional information through graph-based modeling. Additionally, even when some features are missing, the model can compensate for the missing data by leveraging the relationships between neighboring data nodes, ensuring its robustness. By constructing a graph structure that incorporates logging data nodes and their relationships, the graph autoencoder effectively captures the dynamic patterns of time series and integrates the interdependencies between different nodes into a unified graph representation. In comparison to other methods, kick warning using the graph autoencoder can comprehensively address both the dynamic nature of time-series data and the challenge of local feature loss, enabling a more accurate identification of the inherent correlations within multi-dimensional data. However, the graph autoencoder method still presents certain false alarm issues in practical applications. To address this, this study utilizes the weighted dynamic time warping (WDTW) method to identify pump on-off operations during the drilling process. Based on the identified operational states, the false alarm rate is further reduced. At the same time, false alarm samples are fully leveraged to enable real-time model updates through ensemble learning, thereby enhancing the practical value of the model. This approach comprehensively improves the accuracy and applicability of kick warning, providing more reliable technical support for intelligent kick warning during the drilling process.

2. Methodology

This section aims to highlight the details of the framework, including the description of the data preprocessing, the method of graph neural networks, the method of real-time update, field engineer operations, and evaluation metrics.

2.1. Data preprocessing

The data used in this study were collected from 12 wells in western China oilfields. To ensure the reliability and generalization of the experimental results, we provide a detailed statistical summary of the dataset. The dataset contains a total of 94,320 samples, of which 2635 are labeled as kick and 91,685 as normal, resulting in a class distribution ratio of approximately 1:34.8 (kick: normal). This high imbalance reflects real-world operational conditions and presents challenges typical in anomaly detection tasks. The data were acquired under diverse geological and operational environments across different wells, supporting the robustness and generalization of the proposed model. During the collection and storage of integrated logging data, outliers may occur, which significantly deviate from the normal data distribution and range of variation. To detect these outliers, a boxplot method is employed. This method identifies data points that fall outside the interquartile range (IQR) as outliers, ensuring that any abnormal values are effectively flagged for further processing (Tukey, 1977). For handling missing values, spline interpolation is

applied. This method fits a smooth curve through the known data points, and uses this curve to estimate the missing values, providing a reliable approach to fill in gaps in the data. To further enhance the quality of the data, exponential weighted moving average (EWMA) is used for data smoothing and noise reduction. This method assigns exponentially decreasing weights to past observations, with more recent data points receiving higher weights. As a result, the data is smoothed, reducing random fluctuations and improving the accuracy of subsequent analyses. The calculation formula is shown in Eq. (1).

$$S_t = a \cdot x_t + (1 - a) \cdot S_{t-1} \quad (1)$$

where S_t is the EWMA value at the current time point, x_t is the actual observed value at the current time point, S_{t-1} is the EWMA value at the previous time point, and a is the smoothing coefficient, which ranges between 0 and 1. The larger the value of a , the greater the weight of recent data, and the smaller the value of a , the smaller the weight of recent data.

After analyzing the comprehensive engineering expertise and the Spearman correlation coefficient feature selection method, this study ultimately selected the following 8 parameters as the input features for the construction of the intelligent model: hook load, weight on bit (WOB), torque, standpipe pressure, total pit volume, outlet density, outlet flow rate, and inlet flow rate. The Spearman correlation method was applied to evaluate the monotonic relationships between each candidate parameter and kick occurrence. Parameters with relatively strong correlations were retained. Meanwhile, domain knowledge from experienced drilling engineers was used to ensure the selected features reflected key physical mechanisms related to kick events, such as pressure changes. The final selection balances data-driven relevance and practical interpretability, ensuring robust and meaningful model inputs.

To eliminate the dimensional differences between different parameters and reduce their impact on the model's performance, thereby providing a more robust and reliable data foundation for the subsequent construction and analysis of intelligent models, this study uses the Max-Min method to normalize the logging data. In real-time applications, since future data are unavailable, the normalization process in Eq. (2) is implemented using fixed normalization parameters. Specifically, $x_{\max}$ and $x_{\min}$ are pre-calculated from the training dataset and remain constant during online monitoring. This approach ensures stable and consistent normalization across different operational conditions and avoids potential instability caused by short-term fluctuations in real-time data. The calculation formula is shown in Eq. (2).

$$x_{\text{new}} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (2)$$

where x_{new} represents the value after data normalization, x represents the original value of the data, $x_{\max}$ represents the maximum value of all sample data under the selected feature, and $x_{\min}$ represents the minimum value of all sample data under the selected feature.

2.2. Graph neural network-based kick warning method

A graph structure is a mathematical representation widely used to describe complex relational data, consisting of nodes and edges to model the relationships between data points. A graph can be represented as $G = (V, E)$, where $V = \{v_1, v_2, \dots, v_N\}$ is the set of nodes and $E \subseteq V \times V$ is the set of edges. The graph can be designed as a directed or undirected graph, as well as a weighted or un-weighted graph, depending on the application scenario. Each node

$v_i \in V$ typically has an associated feature vector $x_i \in R^F$, where F represents the dimensionality of the features. The structural relationships of the entire graph can be represented by an adjacency matrix $\mathbf{A} \in R^{N \times N}$, where $\mathbf{A}_{ij} = 1$ indicates that nodes v_i and v_j are connected by an edge, otherwise $\mathbf{A}_{ij} = 0$.

Graph neural networks (GNN) are a deep learning framework designed for graph-structured data. The concept of GNN was firstly proposed by Scarselli et al. (2008) in 2008. The core idea is to iteratively propagate messages, enabling each node to aggregate information from its local neighborhood, thus learning higher-order feature representations of the nodes. The basic process of GNN includes:

- 1) Neighbor feature aggregation: collecting information from neighboring nodes.
- 2) Feature update: fusing the aggregated neighbor information with the node's own features.
- 3) Multi-layer propagation: through successive aggregation and update operations, the feature representation of a node gradually captures relationships at further distances.

Similar to traditional convolutional neural networks (CNN) applying local convolutions on regular grids (such as images), GNN performs feature learning on irregular graph-structured data. However, the neighborhoods in graphs are dynamic and irregular. Variants of GNN mainly depend on how the neighborhood features are aggregated. Classic models, such as graph convolutional networks (GCN), aggregate information from neighboring nodes using spectral methods, while graph attention networks (GAT) introduce attention mechanisms to assign different weights to neighboring nodes, allowing for more flexible modeling of the heterogeneous importance between nodes. Compared to traditional methods, GNN has inherent advantages in handling time series and feature missingness. Its feature aggregation mechanism allows for compensation using neighborhood information in the case of missing features, thus maintaining the robustness of the model.

2.2.1. Graph convolutional network (GCN)

Graph convolutional network is a deep learning model based on graph-structured data that can effectively capture the complex relationships between node features and their neighboring nodes (Chen et al., 2020). GCN performs graph convolution operations to aggregate the node's own features with those of its neighboring nodes. It also combines the adjacency matrix and the normalized degree matrix to normalize the features, alleviating the impact of uneven node degree distribution on feature updates. Its core mathematical expression is shown in Eq. (3). This study employs it as one of the supervised base learners to capture spatial correlations in the constructed logging data graph. By leveraging GCN's ability to extract topological information, it supports the classification of kick risk in complex operating environments. A standard two-layer GCN is shown in Fig. 1.

$$\mathbf{H}^{(l+1)} = \sigma\left(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(l)} \mathbf{W}^{(l)}\right) \quad (3)$$

where, $\mathbf{H}^{(l)}$ represents the feature matrix of the nodes at the l -th layer, $\mathbf{W}^{(l)}$ represents the weight matrix at the l -th layer, $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ represents the adjacency matrix plus the self-connection matrix, $\tilde{\mathbf{D}}$ represents the degree matrix plus the diagonal matrix of the self-connection matrix, and σ represents the activation function. By stacking convolution operations layer by layer, GCN is able to embed the node features into a lower-dimensional representation space, while also aggregating information from neighborhoods at

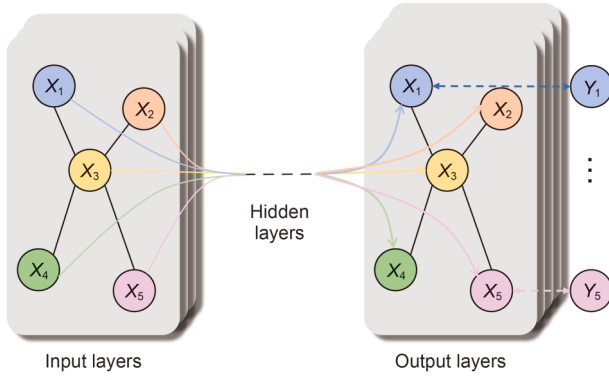


Fig. 1. The standard architecture of two-layer GCN.

further distances, capturing multi-order structural relationships in the graph.

2.2.2. Graph attention network (GAT)

Graph attention network (GAT) is an improved graph neural network model that introduces an attention mechanism to dynamically adjust the aggregation weights of neighborhood information, thereby more flexibly capturing local and global relationships within the graph structure (Velickovic et al., 2017). The core idea of GAT is to assign different attention weights to each node and its neighboring nodes, highlighting the importance of key nodes in feature aggregation. The GAT contributes to improved adaptability by assigning dynamic importance to logging parameters under varying conditions. Its attention-based aggregation helps distinguish subtle feature patterns associated with kick events. The coefficients (Fig. 2) calculated by the attention mechanism are shown in Eq. (4).

$$\alpha_{ij} = \frac{\exp\left(\text{LeakyReLU}\left(\vec{a}^T \left[\mathbf{W}\vec{h}_i \parallel \mathbf{W}\vec{h}_j\right]\right)\right)}{\sum_{k \in N_i} \exp\left(\text{LeakyReLU}\left(\vec{a}^T \left[\mathbf{W}\vec{h}_i \parallel \mathbf{W}\vec{h}_k\right]\right)\right)} \quad (4)$$

where $\vec{h}_i$ represents the node feature vector, $\mathbf{W}$ represents the weight matrix, $\parallel$ denotes the vector concatenation operation, $\vec{a}^T$ denotes the parameter of a single-layer feedforward neural network, $\text{LeakyReLU}(\cdot)$ represents the rectified linear unit activation function with negative parts included, k means the neighbors

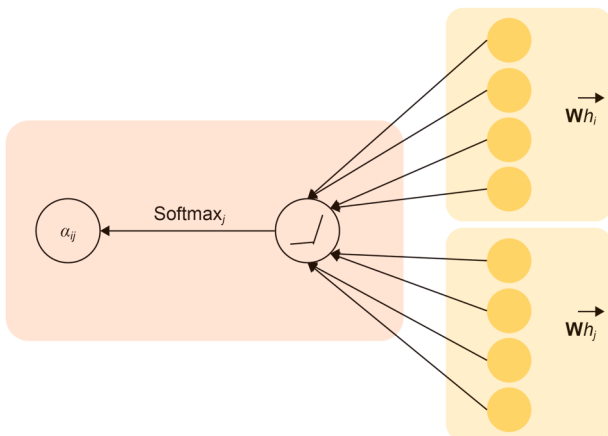


Fig. 2. Attention mechanism diagram.

of node i , and α_{ij} denotes the attention coefficient between node i and node j .

The attention coefficients are used to compute the linear combination with their corresponding features, which serve as the final output features for each node. These features are then concatenated to obtain the output feature representation, as shown in Eq. (5). The aggregation process of the multi-head graph attention layer is shown in Fig. 3.

$$\vec{h}_i' = \parallel_{k=1}^K \sigma \left(\sum_{j \in N_i} \alpha_{ij}^k \mathbf{W}^k \vec{h}_j \right) \quad (5)$$

where α_{ij}^k is the normalized attention coefficient calculated by the k -th attention mechanism. K means the number of multi-head attention, and $\vec{h}_i'$ represents the finally learned node representations.

2.2.3. Graph autoencoder (GAE)

Graph autoencoder (GAE) is a graph embedding model derived from neural networks (Rennard et al., 2021). In this study, a GCN encoder and an inner product decoder are used to obtain optimal representations through graph data reconstruction. The GCN encoder aggregates the features of neighboring nodes to generate optimal node representations, and its process is expressed as shown in Eq. (6). The overall framework of the model is shown in Fig. 4, which includes two methods: path graph construction and K-Nearest Neighbor (KNN) graph construction. It is adopted as the core unsupervised model in this study to address challenges such as data imbalance, missing labels, and evolving feature dynamics in real-time drilling operations. Using the sliding window technique, the time series is divided into segments with a window length of 60 and a step size of 1. In the path graph construction, each time point is treated as a node, and edges are connected sequentially in time order, forming a chain-like graph structure. This approach is well-suited for capturing the sequential and dynamic evolution characteristics of time series. In contrast, for KNN graph construction, nodes are connected based on similarity using the cosine similarity algorithm, with K set to 9. This generates a global, non-sequential graph structure designed to capture complex interaction patterns driven by similarity.

$$\mathbf{Z} = \text{GCN}(\mathbf{X}, \mathbf{A}) \quad (6)$$

Given the node feature matrix $\mathbf{X} \in \mathbb{R}^{N \times P}$ and adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$, they are fed into the GCN function to output $\mathbf{Z} \in \mathbb{R}^{N \times F}$, where $z_i \in \mathbb{R}^F$ represents the node embedding vector and F represents the dimensionality of the node embeddings. The calculation process of GCN is shown in Eq. (7).

$$\text{GCN}(\mathbf{X}, \mathbf{A}) = \tilde{\mathbf{A}} \text{ReLU}(\tilde{\mathbf{A}} \mathbf{X} \mathbf{W}_0) \mathbf{W}_1 \quad (7)$$

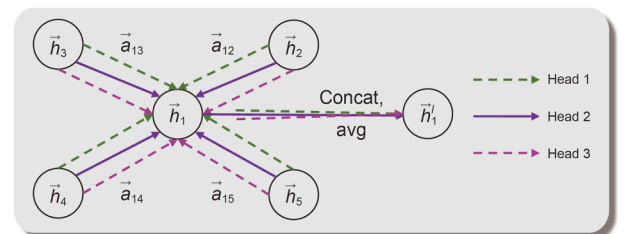


Fig. 3. Aggregation process of the multi-head graph attention layer ($K = 3$) (Li et al., 2022).

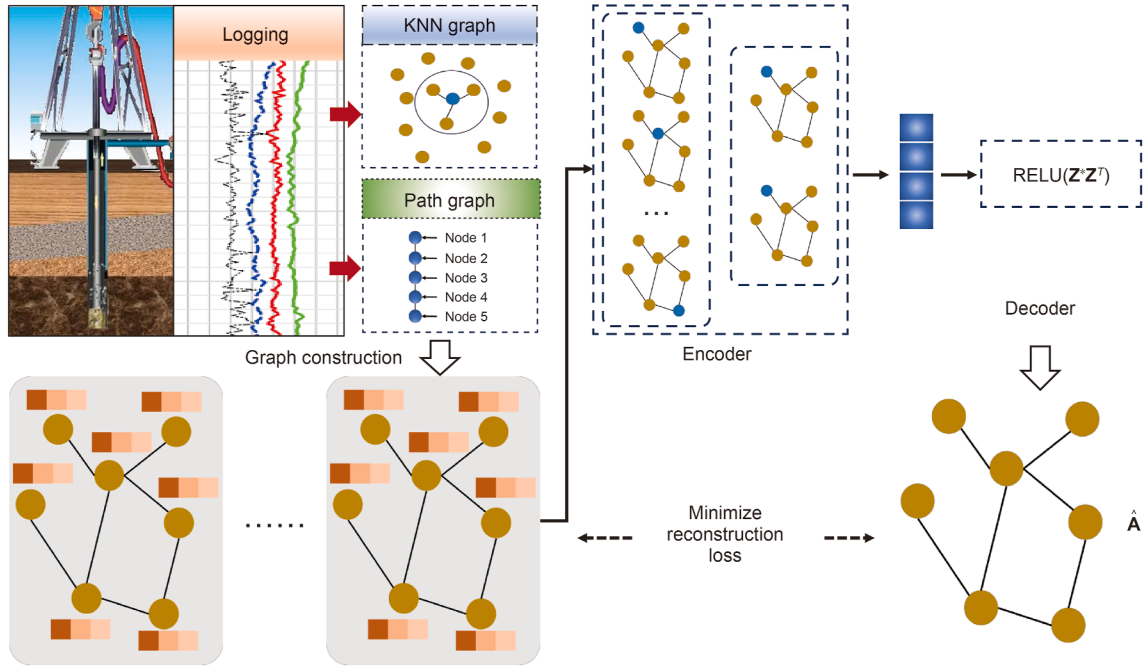


Fig. 4. GAE structure diagram.

where $\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$, $\mathbf{D}$ is the degree matrix of the nodes, $\mathbf{W}_0$ and $\mathbf{W}_1$ are the parameters to be learned.

The decoder consists of two parts: adjacency matrix decoding and node feature decoding. Accordingly, the GAE loss function combines the reconstruction loss of both the adjacency matrix and the node feature matrix. A cross-entropy loss measures how well the reconstructed adjacency matrix preserves the graph structure, while root mean square error (RMSE) evaluates the similarity between the original and reconstructed node features. Smaller differences indicate that the latent representation $\mathbf{Z}$ retains both structural and attribute information effectively (Zhao et al., 2023). Since the GAE is trained only on risk-free samples in an unsupervised manner, it focuses on learning the normal patterns of the drilling process. Deviations in reconstruction errors during inference are used to identify potential kick risks, making the method inherently suitable for handling rare-event scenarios without requiring a balanced dataset.

After training the model with risk-free data, it is used for kick warning by analyzing the features and patterns in the logging data through the difference between the reconstructed data and the original data. When the reconstruction error exceeds a preset threshold (0.5), it is determined that a risk is present. The reconstruction error for GAE is computed using Eqs. (8)–(10).

$$L_1 = -\frac{1}{N} \sum_{i=1}^N y_i \log(1 - y_i) + \hat{y}_i \log(1 - \hat{y}_i) \quad (8)$$

$$L_2 = \text{RMSE}(\mathbf{X}, \hat{\mathbf{X}}) \quad (9)$$

$$L = L_1 + L_2 \quad (10)$$

where y represents a specific element (0 or 1) in the adjacency matrix $\mathbf{A}$; $\hat{y}$ represents the corresponding value (between 0 and 1) in the reconstructed matrix $\hat{\mathbf{A}}$; $\mathbf{X}$ is the original node feature matrix; $\hat{\mathbf{X}}$ is the reconstructed node feature matrix; RMSE represents the calculation of the root mean square error; L_1 is the loss

function for the reconstructed adjacency matrix; L_2 is the loss function for the reconstructed node features; and L is the total loss function.

As GAE is trained using only risk-free data, its output reflects deviations from normal operational patterns. However, an abnormal reconstruction error alone does not directly indicate a kick event. To improve interpretability and reduce false positives, this study further examines the trends of key parameters—specifically, outlet flow rate and total pit volume—around the abnormal time point. In particular, during the calculation of the L_2 reconstruction error, the model extracts the element-wise reconstruction differences for each parameter. If the reconstruction errors for both outlet flow rate and total pit volume are positive, it implies that the GAE underestimates the actual values, indicating an upward trend in these parameters. Such simultaneous rising trends are consistent with typical kick dynamics (e.g., influx of formation fluid), and thus the anomaly is confirmed as a potential kick risk. This approach leverages not only the magnitude but also the sign of the reconstruction error to interpret latent representation behaviors, allowing GAE to function as both an anomaly detector and a physically interpretable kick warning model.

2.3. Real-time update based on stacking ensemble learning

To further improve the accuracy and generalization performance of kick warning, this study designs and constructs a Stacking-based ensemble learning model. GCN, GAT, and GAE are selected as base learners, and the ensemble learning strategy is used to fully leverage the characteristics and advantages of each model. Additionally, a dynamic updating mechanism is introduced to continuously optimize the model's performance, as shown in Fig. 5.

First, multidimensional temporal logging parameters are used as input and transformed into graph-structured data through the path graph construction method. This graph data is then fed into base learners (GCN, GAT, GAE) for feature extraction and processing to identify the key features of the logging data.

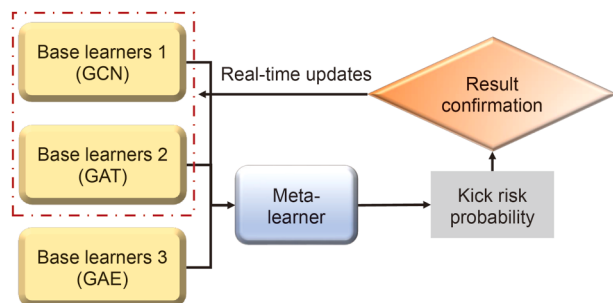


Fig. 5. Real-time dynamic model update.

The features extracted by the base learners are passed to the meta-learner. In this study, extreme gradient boosting (XGBoost) is selected as the meta-learner, which integrates and analyzes these features to ultimately output a comprehensive kick risk probability. XGBoost, as an ensemble learning algorithm based on gradient boosting, exhibits highly efficient feature fusion capabilities and excellent generalization performance, making it particularly effective in handling high-dimensional and heterogeneous data (Chen and Guestrin, 2016). Furthermore, XGBoost's built-in regularization mechanism effectively prevents model overfitting, enabling it to maintain stable warning performance even under complex operating conditions. These advantages make it an ideal choice as the meta-learner for integrating the results of multiple base learners in kick warning. During the model training process, the Stacking model is first trained offline using the available labeled data to initialize the parameters of the base learners and the meta-learner (Naimi and Balzer, 2018). Subsequently, the weights of the base learners are dynamically adjusted during training, optimizing the parameter configuration of the Stacking framework based on their contributions to overall performance (Guo et al., 2017). This hierarchical learning and optimization mechanism fully leverages the characteristics of each base learner, combined with the meta-learner's comprehensive evaluation capabilities, to achieve high-precision kick risk prediction. With this design, the model can not only effectively adapt to the learning of complex temporal features but also maintain stable warning performance under different operating conditions.

To adapt to the dynamic changes in drilling conditions, this study designs a real-time data-based dynamic update mechanism that continuously optimizes model performance by incorporating real-time confirmation and updates of false alarm samples. After a kick risk alarm is issued, field engineers confirm the alarm results and label false alarm samples in real-time, adding them to the dataset for retraining the supervised base learners, GCN and GAT. Normal samples are not included in this real-time update process. This selective update strategy focuses the model's attention on addressing recent misjudgments and avoids introducing redundant or potentially noisy information. Through this mechanism, the model can correct its tendency to misjudge similar false alarm samples, thereby significantly reducing the false alarm rate. The unsupervised GAE, which does not require labeled data, maintains unchanged parameters during the real-time update process, providing stable global feature support for the model. As false alarm samples are gradually incorporated, GCN and GAT continuously learn more operating condition characteristics, effectively improving the model's accuracy and transferability, achieving

adaptive optimization under complex drilling environments. This dynamic update strategy significantly enhances the robustness and real-time performance of the model, ensuring the reliability of kick warning.

2.4. Field engineer operations

In drilling operations, to monitor kick risk, it is usually necessary to observe changes in the outlet flow rate of drilling fluid and the volume of the drilling fluid pit. However, these parameters are not only affected by kick but also exhibit significant fluctuations due to pump-on and pump-off operations. These fluctuations are often misinterpreted by monitoring systems as kick, leading to false alarms and affecting the accuracy of the judgment.

Dynamic time warping (DTW) is a method for finding the minimum path by providing a nonlinear alignment between two time series. In actual industrial processes, data measured by sensors are inevitably subject to interference or delays. Therefore, using DTW distance instead of the commonly used Euclidean distance often results in more accurate fault warning. Weighted dynamic time warping (WDTW) and derivative dynamic time warping (DDTW) are variant algorithms of DTW, each optimized and improved for addressing different issues in time series matching. In previous studies, Zhou et al. (2024) used DDTW for condition identification. This method, by matching the first derivatives of time series, is better able to capture the similarity of changing trends in drilling conditions. This study adapts WDTW as a preprocessing module to distinguish engineer-induced operational changes from actual kick events. However, DDTW's sensitivity to noise and outliers limits its applicability in high-noise, complex condition datasets. Additionally, DDTW has limited capability to balance global and local alignments. To address the aforementioned issues, this study introduces a new algorithm—WDTW. WDTW incorporates a penalty factor into the distance metric, where the larger the time gap between two points, the greater the penalty; conversely, the smaller the time gap, the smaller the penalty. This weighting scheme allows WDTW to align short-term trends more accurately. By analyzing the relative variation trends between pump strokes and outlet flow rates, WDTW can effectively identify pump-on and pump-off conditions, thereby improving the reliability of operational interpretation. In our proposed framework, kick detection and engineer operation identification are carried out simultaneously. When WDTW identifies that a pump-on or pump-off operation is occurring and the kick detection module simultaneously raises an alert, the system treats this as a likely false alarm. In such cases, the warning is suppressed to avoid misclassification caused by operational noise. This strategy enables the model to distinguish between genuine kick patterns and normal operation-induced fluctuations, improving both the precision and robustness of the overall warning system.

For the two sequences $A(a_1, a_2, \dots, a_i, \dots, a_m)$ and $B(b_1, b_2, \dots, b_j, \dots, b_n)$, representing pump strokes and outlet flow rates, WDTW finds the minimum warping path, which is expressed as Eq. (11).

$$E_p(A, B) = \sqrt[p]{\gamma^*(i, j)} \quad (11)$$

where $\gamma^*(i, j)$ is the accumulated warping path between sample points i and j , and their corresponding relationship can be recursively obtained through Eq. (12).

$$\gamma^*(i, j) = |w_{|i-j|} (a_i - b_j)|^p + \min \{ \gamma^*(i-1, j-1), \gamma^*(i-1, j_q) \gamma^*(i, j-1) \} \quad (12)$$

where $w_{|i-j|}$ represents the weight of the phase difference between sample points in A and B . The greater the distance between the sample points, the larger the value of $w_{|i-j|}$. When $w_{|i-j|}$ is considered as a constant, the penalty between all sample points is the same, and in this case, the WDTW distance is no different from the DTW distance. However, when $w_{|i-j|}$ is set to be a very large value, the penalty for even closely spaced sample points becomes significant, and in this case, the WDTW distance is equivalent to the Euclidean distance.

By setting an appropriate threshold K , if the value of $|i-j|$ exceeds this threshold K , a larger weight $w_{|i-j|}$ is assigned to prevent matching between sample points with excessive phase differences. If the value of $|i-j|$ is smaller, the assigned weight $w_{|i-j|}$ will not affect the matching between sample points with similar amplitudes. The definition of the WDTW distance is given in Eq. (13).

$$\text{WDTW}(A, B) = \|w_{|i-j|} (a_i - b_j)\|_p \quad (13)$$

2.5. Evaluation metrics

The intelligent warning of kick risk studied in this study essentially belongs to a binary classification problem. Therefore, it is necessary to select relevant evaluation metrics for classification models. Accuracy, Recall, Precision, false alarm rate (FAR), and missed alarm rate (MAR) are adopted to evaluate the performance of the intelligent kick warning model. These evaluation metrics are calculated based on the confusion matrix, as shown in Table 1, and their definitions are provided in Eqs. (14)–(18).

TP represents the number of samples where the model correctly classifies the true positive class as positive; FN represents the number of samples where the model incorrectly classifies the true positive class as negative; FP represents the number of samples where the model incorrectly classifies the true negative class as positive; and TN represents the number of samples where the model correctly classifies the true negative class as negative.

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP} \quad (14)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (15)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (16)$$

$$\text{FAR} = \frac{FP}{TN + FP} \quad (17)$$

$$\text{MAR} = \frac{FN}{TP + FN} \quad (18)$$

Table 1
Confusion matrix.

True result	Forecast result	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

3. Results and discussion

This section analyzes and discusses the proposed kick warning method through a series of experiments. It focuses on comparing the impact of different graph construction methods, model types, and time window size on warning performance, while further optimizing the FAR by incorporating pump on-off condition recognition. The experiments comprehensively evaluate the robustness and sensitivity of the method, providing strong support for its practical application in drilling operations.

3.1. Graph construction method analysis

In the experiment on kick warning based on the GAE model, we compared two graph construction methods: path graph and KNN graph. By comparing the experimental results, the performance of different graph construction methods was comprehensively evaluated using multiple metrics, including accuracy, recall, precision, FAR, and MAR. The experiments revealed that the overall performance of the path graph is superior to that of the KNN graph, especially demonstrating significant advantages in terms of FAR and MAR. The results are shown in Table 2.

To determine the optimal value of K for the KNN graph, we conducted comparative experiments with $K = 7, 9$, and 11 . The results show that when $K = 9$, the model achieves the best overall performance, with an accuracy of 84.9%, a recall of 83.4%, and a precision of 72.4%, while maintaining a lower FAR of 28.5% and a lower MAR of 16.6%. In contrast, $K = 7$ and 11 resulted in lower accuracy and higher error rates. Therefore, $K = 9$ was selected as the adjacency parameter, balancing both detection effectiveness and model robustness.

The accuracy of the path graph reached 92.3%, which is 7.4% higher than the 84.9% achieved by KNN graph construction, indicating higher reliability in overall classification performance. In terms of recall and precision, the path graph achieved 95.7% and 82.2%, respectively, compared to 83.4% and 72.4% for the KNN graph, showing improvements of 12.3% and 9.8%, respectively. This demonstrates the significant advantage of the path graph in capturing true kick events and reducing the number of false alarms. Additionally, the FAR of the path graph is 17.8%, significantly lower than the 28.5% of the KNN graph, indicating fewer misclassifications in non-kick conditions, making it more suitable for practical warning needs. Meanwhile, the MAR of the path graph is only 4.3%, significantly lower than the 16.6% of the KNN graph, reflecting its strong ability to reduce missed warnings of critical kick risk.

The path graph construction method is directly based on the time-series characteristics of logging data, treating each time step as a node and establishing edges through the natural sequence of time steps. This construction method effectively captures the dynamic variation patterns and temporal correlations of time-series data, enabling the GAE model to uncover potential association patterns along the temporal dimension. This characteristic is particularly important for the warning of kick risk, as kick is typically characterized by a series of dynamic feature changes. In contrast, the KNN graph construction method establishes edges based on feature similarity between nodes, but it has significant

Table 2
Experimental results of path graph and KNN graph construction.

Index	Accuracy, %	Recall, %	Precision, %	FAR, %	MAR, %
KNN graph ($K = 7$)	75.3	72.9	68.5	34.8	27.1
KNN graph ($K = 9$)	84.9	83.4	72.4	28.5	16.6
KNN graph ($K = 11$)	80.1	82.5	70.2	29.0	17.5
Path graph	92.3	95.7	82.2	17.8	4.3

limitations when applied to time-series kick warning of logging data. Firstly, KNN graph construction ignores the temporal characteristics of logging data, connecting nodes purely based on static feature similarity, and is therefore unable to capture the dynamic evolution patterns of kick risk. Secondly, it may introduce erroneous connections between nodes that are feature-similar but not actually related, which can reduce the model's accuracy and significantly increase the FAR.

3.2. Model comparison and optimization

After selecting the path graph construction method in Section 3.1, this section further compares the performance of four models: multilayer perceptron-based autoencoder (MLP-AE), GCN, GAT, and GAE. For the supervised models (GCN and GAT), a SMOTE-Tomek hybrid sampling strategy was applied to balance the training data and mitigate the impact of class imbalance. The experimental results show that the GAE model, based on unsupervised learning, achieved the best performance across all evaluation metrics.

A multilayer perceptron-based autoencoder (MLP-AE) model was first established for kick warning. The network structure of MLP-AE consists of an encoder and a decoder. The encoder contains two hidden layers with 64 and 32 neurons, respectively, compressing the input data into a 32-dimensional latent feature space. The decoder reconstructs the latent features output by the encoder and also includes two fully connected layers with 32 and 64 neurons, respectively. All hidden layers use the ReLU activation function to introduce nonlinearity, while the optimizer is Adam with a learning rate of 0.001. The experimental results, shown in Fig. 6, reveal that the MLP-AE achieved an accuracy of 80.9%, a precision of 75.4%, a recall of 78.6%, FAR of 29.2%, and MAR of 21.4%. The underperformance of this model is mainly attributed to its inability to effectively leverage the temporal characteristics of logging data, as it relies solely on static global features for warning, overlooking the critical dynamic changes associated with kick risk. Additionally, the MLP-AE model is highly sensitive to short-term noise and operational fluctuations in logging data, resulting in a high false alarm rate and difficulty reducing the miss rate. Although the MLP-AE model is simple to implement and

Table 3
Optimal value selection for model parameters.

Model	Parameter	Parameter range	Optimal value
GCN	Number of layers	1, 2, 3, 4	3
	Learning rate	0.001, 0.003, 0.01	0.003
	Activation function	ReLU, Tanh	ReLU
GAT	Number of layers	1, 2, 3, 4	2
	Learning rate	0.001, 0.003, 0.01	0.01
	Number of attention heads	2, 3, 5	3
GAE	Activation function	ReLU, Tanh	ReLU
	Number of layers	1, 2, 3, 4	2
	Learning rate	0.001, 0.003, 0.01	0.003
	Activation function	ReLU, Tanh	Tanh

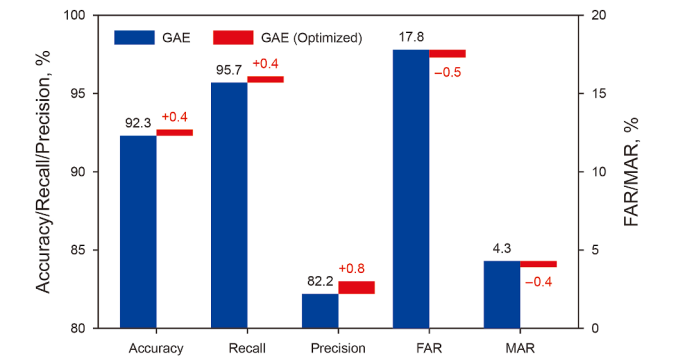


Fig. 7. Comparison of GAE results before and after optimization.

computationally efficient, its lack of capability in capturing complex temporal dynamics severely limits its applicability in practical kick warning tasks.

To further improve model performance, key parameters of GCN, GAT, and GAE were optimized using the Bayesian optimization method. The optimized parameters include the number of encoding layers, learning rate, and others, as shown in Table 3. By applying Bayesian optimization to fine-tune the key parameters of the GAE model, a slight improvement over the original performance was achieved, as illustrated in Fig. 7.

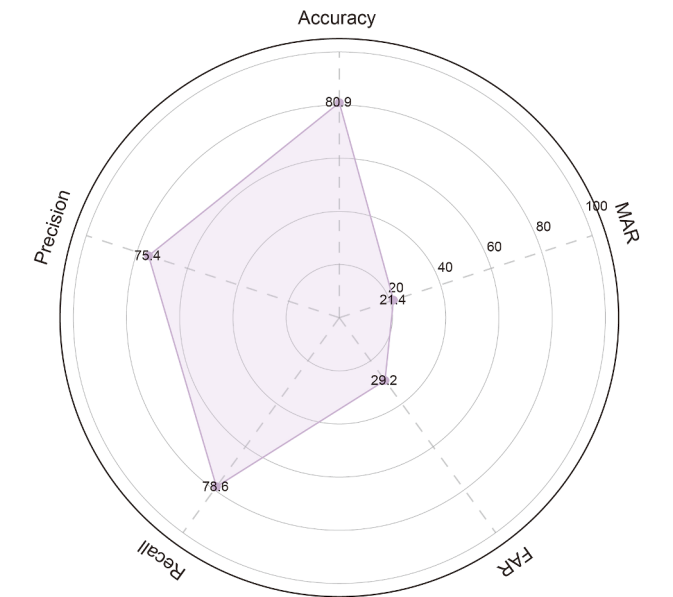


Fig. 6. MLP-AE experimental results.

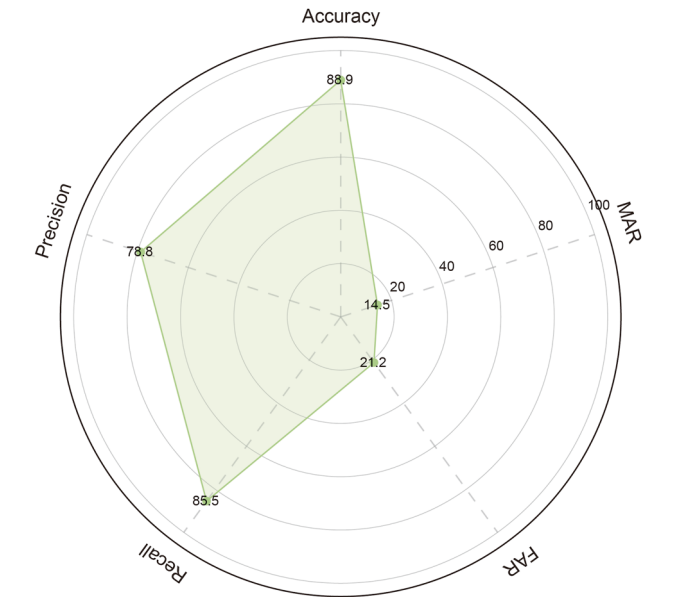


Fig. 8. GCN experimental results.

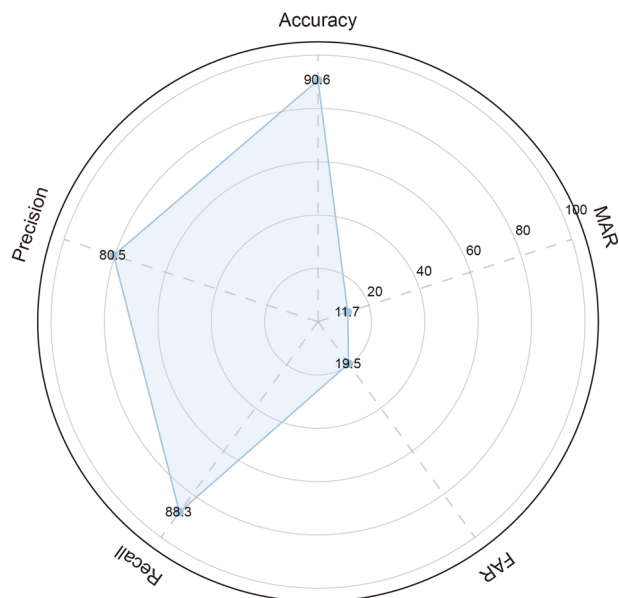


Fig. 9. GAT experimental results.

The model results based on the path graph construction method are shown in Figs. 8–10. GAE outperforms all other models across all evaluation metrics, with an accuracy of 92.7%, which is 3.8% higher than GCN (88.9%) and 2.1% higher than GAT (90.6%), demonstrating its superiority in overall classification performance. In terms of recall, GAE achieves 96.1%, an improvement of 10.6% over GCN (85.5%) and 7.8% over GAT (88.3%), reflecting its stronger ability to capture true kick events. Meanwhile, GAE's precision is 83.0%, which is 4.2% higher than GCN (78.8%) and 2.5% higher than GAT (80.5%), indicating its better performance in reducing false alarms. Additionally, for FAR and MAR, GAE reaches 17.3% and 3.9%, respectively, which are significantly lower than GCN (21.2%, 14.5%) and GAT (19.5%, 11.7%), highlighting GAE's comprehensive advantage in reducing misclassifications in non-kick states and missed warning of critical kick risk.

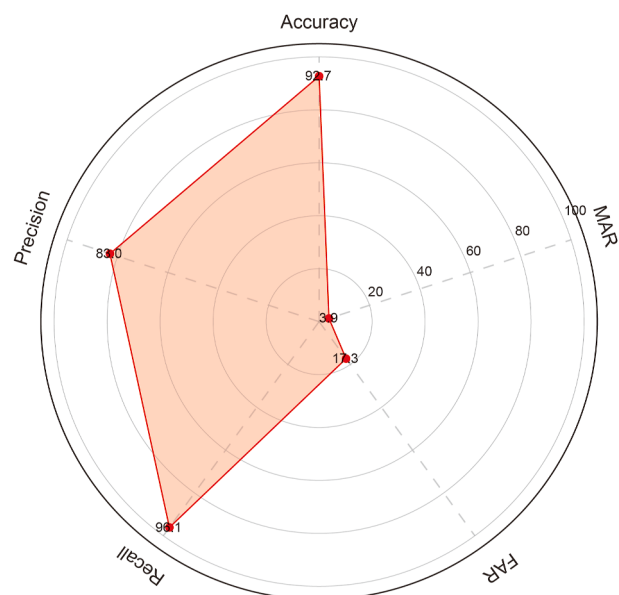


Fig. 10. GAE experimental results.

The unsupervised architecture of the GAE model allows for training through the reconstruction of the adjacency matrix without the need for large amounts of labeled data. This is of significant importance in kick warning, as field logging data often suffer from issues such as insufficient samples or difficulties in labeling. Compared to GCN and GAT, which rely on supervised learning and have a strong dependency on large amounts of labeled data, GAE extracts information directly from the graph's structure and features through unsupervised learning, overcoming the small sample problem and enhancing the model's generalization ability in small sample scenarios. The encoder of the GAE model generates low-dimensional node embeddings that preserve the graph's structural information while compressing the complexity of high-dimensional features. This feature not only reduces computational overhead but also strengthens the model's ability to represent multi-dimensional features. In contrast, although GCN and GAT can also generate embeddings, their optimization targets are more dependent on task labels, making the models more susceptible to limitations imposed by the quality of the labels.

3.3. Time window size optimization

In this section, different time windows (40, 50, 60, 70, 80) were selected to construct path graph structured data, and the performance of kick warning was tested based on the GAE model. The experimental results show that when the time window is set to 60, all evaluation metrics achieve optimal values, as shown in Fig. 11. Specifically, with a time window of 60, the model exhibits the best performance, achieving an accuracy of 92.7%, a recall of 96.1%, a precision of 83.0%, and FAR and MAR of 17.3% and 3.9%, respectively.

In contrast, smaller time windows, such as 40 and 50, while capable of capturing short-term dynamic changes, are more susceptible to short-term fluctuations, increasing the likelihood of misclassifying normal conditions as kick risk, resulting in higher false alarm rates. On the other hand, larger time windows, such as 70 and 80, due to their longer time spans, may weaken the early characteristics of kick risk, leading to an increase in missed alarm rates and a decrease in recall. A time window of 60 achieves the best balance between capturing short-term dynamic features and long-term trend characteristics. Under this window size, the critical features of kick risk in the logging data (e.g., a drop in stand-pipe pressure, an increase in outlet flow, and a rise in total pit volume) can be fully captured. Compared to smaller time windows, it effectively reduces the interference caused by data fluctuations, while compared to larger time windows, it avoids the signal weakening and warning delays caused by an excessively long time span.

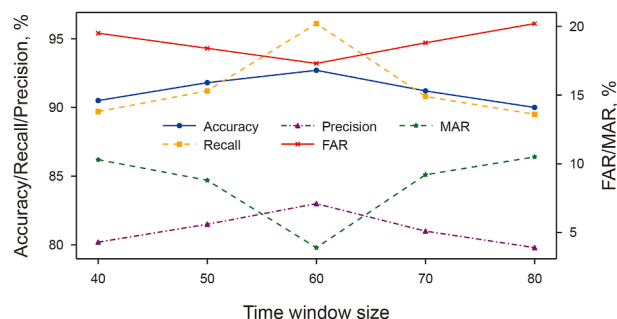


Fig. 11. Comparison of results for different time window sizes.

Therefore, the selection of a time window of 60 not only significantly enhances the sensitivity and robustness of the model but also ensures the timely warning of kick risk, effectively suppressing the occurrence of false alarms.

3.4. Reducing false alarms by combining field engineer operations

In Section 3.2, although the GAE model demonstrated superior performance in kick warning tasks, there was still a certain percentage of false alarms (17.3%). The primary cause of these false alarms is the short-term sharp fluctuations in logging curves triggered by operational changes during non-kick states in the drilling process (e.g., pump-on and pump-off operations). As shown in Fig. 12, after pump-off, the total pit volume shows an upward trend, and during pump-on, the outlet flow rate exhibits a significant increase. These changes resemble the characteristics of kick risk, leading to misclassification by the model as kick risk. And the identification result of “1” indicates that the intelligent model has determined a kick risk has occurred, while “0” indicates that no kick risk is identified. To further reduce the FAR, this section uses the WDTW method based on pump strokes and outlet flow rate data to identify pump-on and pump-off operations. Alerts related to pump-on and pump-off operations are marked as false alarms and filtered out, effectively reducing the FAR. To investigate the impact of the threshold parameter K in the WDTW algorithm for pump-on and pump-off operations identification, we conducted experiments using three values: $K = 5, 10$, and 15 . The corresponding accuracy results were 81.3%, 88.6%, and 84.2%, respectively. Among these, the best performance was observed at $K = 10$, indicating that this value strikes a favorable balance between local sensitivity and general trend capture. A lower value such as $K = 5$ tends to restrict the model's flexibility in matching sequences with minor phase shifts, while a higher value like $K = 15$ may lead to overly smoothed matching results that fail to reflect important temporal patterns. Therefore, based on empirical analysis, $K = 10$ was selected as the optimal threshold for this application, as it enables the WDTW algorithm to effectively model temporal dynamics while maintaining strong overall predictive performance.

At the same time, a real-time updating method based on ensemble learning is introduced to dynamically label and update false alarm samples. Confirmed false alarm samples are immediately added to the training set of the supervised base learners (GCN and GAT) to further optimize the model parameters. Through continuous iteration, the model effectively learns the characteristics of false alarm samples, thereby improving its ability to recognize similar operating conditions and reducing false alarms. On the other hand, the unsupervised learner GAE, which does not rely on labeled data, keeps its parameters unchanged and continues to provide stable global feature support for the ensemble model. Through this dynamic updating mechanism, the model's false alarm rate is significantly reduced, while its ability to generalize and adapt to diverse drilling conditions is further enhanced.

After introducing the WDTW pump on-off recognition and real-time updating module, the experimental results show that this method significantly reduces false alarms, with the false alarm rate dropping from 17.3% to 5.6%, a decrease of 11.7%, effectively minimizing misclassifications in non-kick states. At the same time, the precision increased from 83.0% to 86.0%, an improvement of 3.0%, further demonstrating the effectiveness of this method in suppressing false alarms. Accuracy also improved from 92.7% to 93.4%, reflecting an optimization in overall warning performance. Although the recall rate slightly decreased from 96.1% to 95.8%, and MAR slightly increased from 3.9% to 4.2%, these changes were relatively minor and remained at a high level.

Based on the experimental results, it can be concluded that the WDTW pump on-off recognition and real-time updating method significantly reduces the model's FAR while positively impacting the overall model performance, providing higher reliability and practical applicability for kick warning tasks. In terms of practical deployment, the proposed real-time update strategy was designed with computational efficiency in mind to suit the resource-constrained environments commonly found in oilfield operations. The dynamic update process only retrain the supervised components (GCN and GAT) incrementally using a limited number of newly confirmed false alarm samples, which significantly reduces the computational load. The unsupervised

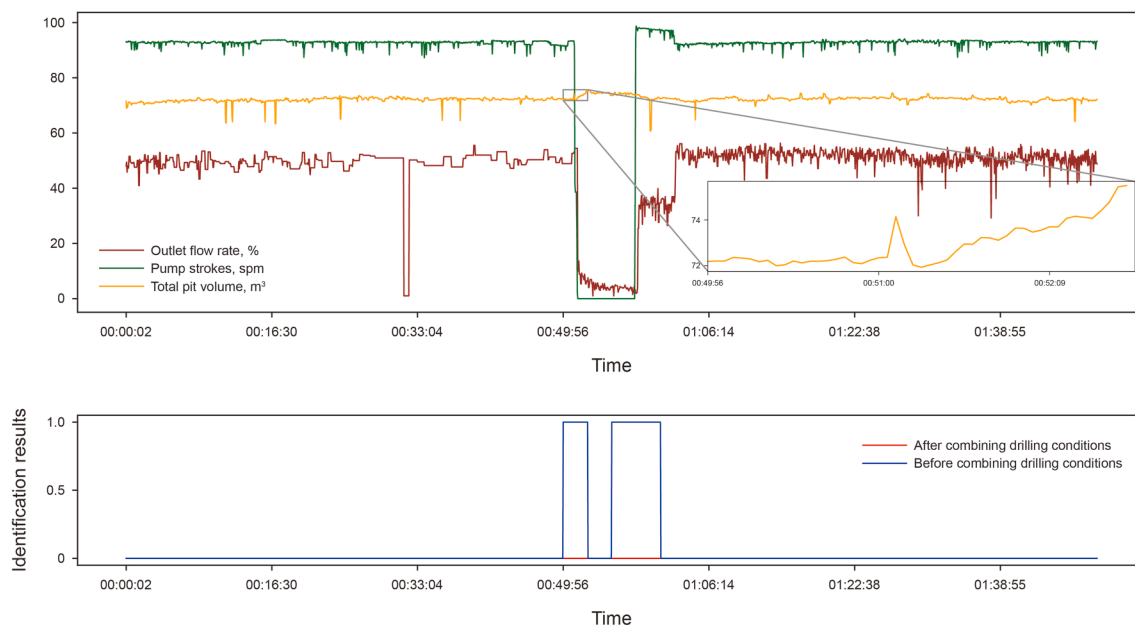


Fig. 12. Pump-on and pump-off conditions causing false alarms in kick warning.

GAE remains fixed during real-time updates, serving as a stable global feature extractor without introducing additional training overhead. Furthermore, the entire system is compatible with standard edge computing units typically available at drilling sites. In our testing, the model update cycle can be completed in under 30 s using a mid-range GPU, and even on a CPU-only setup, the latency remains within acceptable limits for field operations. These features make the model feasible for real-time deployment, providing both adaptability and robustness without requiring high-end hardware.

4. Case analysis

This section selects two kick cases to validate the accuracy of the proposed method. Real-time monitoring data from drilling operations in western China oilfields were chosen, including relevant logging parameters such as outlet flow rate and standpipe pressure. By applying the unsupervised GAE model and the method for reducing false alarms proposed in this study, the effectiveness of kick event warning was evaluated. The model's reliance on trend information rather than absolute values enables

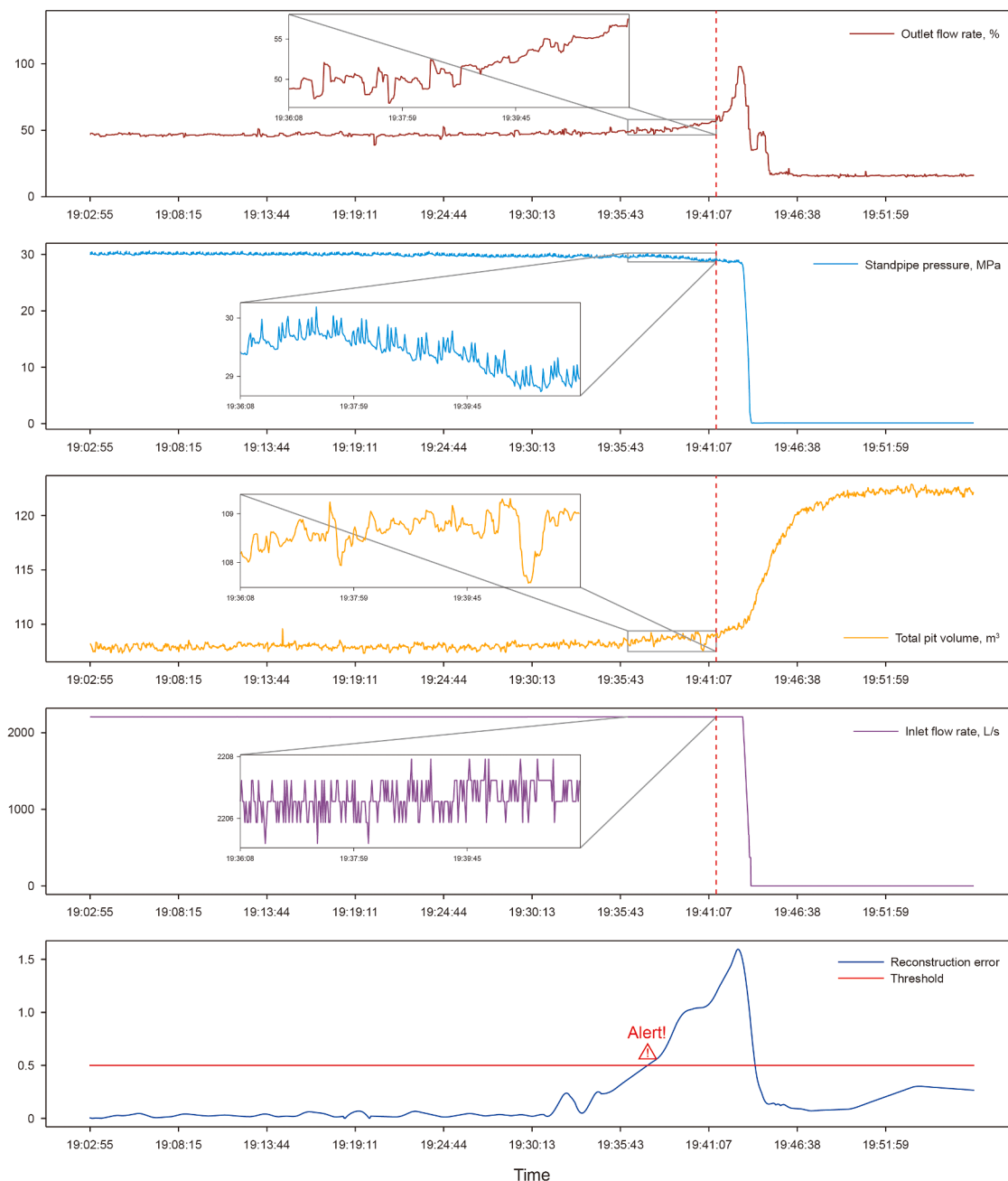


Fig. 13. Real-time logging data and kick risk warning results for Well 1 (the red dashed line represents the time when the field engineer issued the kick alarm).

it to adapt to percentage-based or lower-precision signals, such as those from orifice-plate flowmeters commonly used in field operations.

The logging data and kick warning results for Well 1 are shown in Fig. 13. From the logging curve, it can be seen that starting at 19:37, the standpipe pressure begins to decrease, accompanied by an increase in outlet flow rate and total pit volume. The field engineer confirmed the occurrence of kick risk at 19:41:37. The kick warning method proposed in this study issued an alert at 19:37:22, approximately 4.25 min earlier than the field engineer's confirmation. The GAE model, by considering the temporal nature of the data, is able to capture early feature changes associated with kick risk. Its high sensitivity allows for timely warning when the kick risk first emerges. This temporal sensitivity enables the GAE model

to provide sufficient time for the field engineer to take effective countermeasures.

The logging data and kick warning results for Well 2 are shown in Fig. 14. During the time period indicated in the figure, the outlet flow rate was negative, while the inlet flow rate exceeded 30 L/s, indicating that the outlet flow parameter measurement had failed. During data processing, the missing outlet flow data was filled with a value of 0, but this did not affect the timeliness of the early kick warning method. The on-site engineers confirmed the occurrence of a kick risk at 8:41:30, while the proposed kick warning method issued an alert at 8:37:07, approximately 4.38 min earlier. Compared to Well 1, where the warning was issued 4.24 min earlier than the engineers, the GAE model maintained a similar or even slightly better early warning performance



Fig. 14. Real-time logging data and kick risk warning results for Well 2 (the red dashed line represents the time when the field engineer issued the kick alarm).

in Well 2 despite the presence of missing features. This demonstrates the strong robustness of the model in maintaining timely detection. To further validate the effectiveness of GAE under missing feature conditions, the MLP-AE model was tested on Well 2. The MLP-AE issued a kick warning 1.53 min later than the field engineers, indicating a significant delay and highlighting its lower resilience to missing key input parameters. The superior performance of GAE can be attributed to its graph-based structure, which enables multi-feature interaction and aggregation. Even when certain critical features are missing, GAE can integrate correlated information from other available parameters (such as inlet flow rate, pit volume, and pressure) through neighborhood feature propagation, preserving the overall dynamic patterns necessary for accurate kick detection. This multi-feature compensation mechanism is a key advantage of using graph-based models over traditional fully connected architectures when dealing with incomplete or noisy real-world data.

5. Conclusion

This study proposes a kick warning method based on an unsupervised GAE and effectively reduces false alarms by incorporating pump-on and pump-off condition recognition. The results indicate that the proposed method accurately identifies kick risk while successfully minimizing false alarms caused by operational changes. The specific conclusions are as follows:

1. Among different graph construction methods, the path graph construction method demonstrated the best performance. The path graph effectively captures the temporal characteristics in logging data, ensuring early warning of kick risk and a low false alarm rate. Experimental results show that the accuracy of the path graph is 92.7%, highlighting its superiority in overall kick warning performance.
2. The kick warning method based on the unsupervised GAE achieved the best performance among all models. By leveraging temporal information, GAE overcomes the small sample problem and can operate effectively even in the presence of feature missing, demonstrating its strong robustness and sensitivity. The unsupervised learning nature of the GAE model enables it to function effectively in scenarios where a large amount of labeled data is unavailable, significantly improving its practical applicability. GAE achieved an accuracy of 92.7%, a recall of 96.1%, and a precision of 83.0% across all test cases, outperforming both GCN and GAT significantly.
3. The integration of the WDTW method effectively reduced false alarms caused by short-term fluctuations in pump strokes, outlet flow rates, and other operational changes induced by field engineer activities. These fluctuations, often mistaken for kick risk, were mitigated by the model. The real-time update mechanism dynamically labels false alarm samples and incorporates them into training, further improving the model's performance, reducing the false alarm rate by 11.7% (from 17.3% to 5.6%) and increasing precision by 3.0%. Moreover, the GAE model proved reliable even with missing data, issuing early warnings 4.25 and 4.38 min earlier than field engineers in two case studies, demonstrating its sensitivity and adaptability.

Although the kick warning method proposed in this study has shown promising results, there is still room for improvement. Future work will explore the integration of self-supervised learning, reinforcement learning, and more advanced graph neural network structures to enhance model adaptability and performance. We will also incorporate additional logging parameters to improve the model's ability to handle changing operating

conditions and reduce false alarms. To address the computational complexity of GAE and WDTW, more efficient algorithms and optimization strategies will be investigated for real-time warning applications. Furthermore, efforts will be made to promote the application of this technology in drilling sites, aiming to improve the safety and efficiency of drilling operations.

Declaration of competing interest

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

Acknowledgments

The authors express their appreciation for Youth Foundation of National Natural Science Foundation of China (No. 52204020) and Distinguished Young Foundation of National Natural Science Foundation of China (No. 52125401).

Nomenclature

GNN	graph neural networks
CNN	convolutional neural networks
GCN	graph convolutional networks
GAT	graph attention networks
GAE	graph autoencoder
DTW	dynamic time warping
WDTW	weighted dynamic time warping
DDTW	derivative dynamic time warping
FAR	false alarm rate
MAR	missed alarm rate
EWMA	exponential weighted moving average
KNN	K-nearest neighbor
WOB	weight on bit
RMSE	root mean square error
S_t	EWMA value at the current time point
$\mathbf{H}^{(l)}$	the feature matrix of the nodes at the l -th layer
$\mathbf{A}$	the adjacency matrix
$\mathbf{D}$	the degree matrix
$\tilde{\mathbf{D}}$	the degree matrix plus the diagonal matrix of the self-connection matrix
σ	the activation function
$\vec{h}_i$	the node feature vector
$\mathbf{W}$	the weight matrix
$\parallel$	the vector concatenation operation
α_{ij}^k	the normalized attention coefficient
L_1	the loss function for the reconstructed adjacency matrix
L_2	the loss function for the reconstructed node features
L	the total loss function
$w_{ i-j }$	the weight of the phase difference between sample points in A and B

References

- Ablard, P., Bell, C., Cook, D., et al., 2012. The expanding role of mud logging. *Oilfield Rev.* 24 (1), 24–41. https://scholar.google.com/scholar_lookup?title=The+20Expanding+20Role+20of+20Mud+20Logging&author=P./20Ablard&author=C./20Bell&author=D./20Cook&publication_year=2012&journal=Oilfield+20Review+20Spring&volume=24&pages=
- Chen, G., Wang, X., Ji, P., et al., 2024. A kick monitoring method for deepwater open-circuit drilling based on convolutional neural network. *Geoenergy Sci. Eng.* 234, 212656. <https://doi.org/10.1016/j.jgeoen.2024.212656>.

- Chen, M., Wei, Z., Huang, Z., et al., 2020. Simple and Deep Graph Convolutional Networks. International Conference on Machine Learning, 119. PMLR, Vienna, Austria. <https://doi.org/10.48550/arXiv.2007.02133>.
- Chen, T., Guestrin, C., 2016. Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. <https://doi.org/10.1145/2939672.2939785>.
- Duan, S., Song, X., Cui, Y., et al., 2023. Intelligent kick warning based on drilling activity classification. *Geoenergy Sci. Eng.* 222, 211408. <https://doi.org/10.1016/j.geoen.2022.211408>.
- Griffin, P., 1967. Early Kick Detection Holds Kill Pressure Lower. SPE Mechanical Engineering Aspects of Drilling and Production Symposium, Fort Worth, Texas, USA. <https://doi.org/10.2118/1755-MS>.
- Guo, L., Wang, S., Cao, Z., 2017. An Ensemble Classifier Based on Stacked Generalization for Predicting Membrane Protein Types. International Congress on Image and Signal Processing, Biomedical Engineering and Informatics (CISP-BMEI). <https://doi.org/10.1109/cisp-bmei.2017.8302278>.
- Han, J., Yang, H., Zhang, J., 2017. Accurate discovery of overflow research and field application in the northwest industrial area. *Logging Engineering* 28 (3), 69–74. <https://doi.org/10.3969/j.issn.1672-9803.2017.03.015> (in Chinese).
- Jiang, Z.Q., Wang, Y.S., Mao, M., et al., 2013. Early kick detection (EKD) system based on coriolis mass flowmeter. *Journal of Oil and Gas* 35 (4), 158–160. <https://doi.org/10.3969/j.issn.1000-9752.2013.04.036> (in Chinese).
- Li, G., Song, X., Tian, S., Zhu, Z., 2022a. Intelligent drilling and completion: a review. *Engineering* 18, 33–48. <https://doi.org/10.1016/j.eng.2022.07.014>.
- Li, T., Zhou, Z., Li, S., et al., 2022b. The emerging graph neural networks for intelligent fault diagnostics and prognostics: a guideline and a benchmark study. *Mech. Syst. Signal Process.* 168, 108653. <https://doi.org/10.1016/j.ymssp.2021.108653>.
- Liang, H., Liu, G., Gao, J., Khan, M.J., 2020. Overflow remote warning using improved fuzzy c-means clustering in IoT monitoring system based on multi-access edge computing. *Neural Comput. Appl.* 32 (19), 15399–15410. <https://doi.org/10.1007/s00521-019-04540-y>.
- Liang, H., Zou, J., Liang, W., 2019. An early intelligent diagnosis model for drilling overflow based on GA-BP algorithm. *Clust. Comput.* 22, 10649–10668. <https://doi.org/10.1007/s10586-017-1152-5>.
- Liang, H.B., Wang, Z., 2019. Application of an intelligent early-warning method based on DBSCAN clustering for drilling overflow accident. *Clust. Comput.* 22 (Suppl. 5), 12599–12608. <https://doi.org/10.1007/s10586-017-1687-5>.
- Liu, S.J., 2006. Development of device for drilling fluid level detection and automatic grout system. *China Petroleum Machinery* 34 (2), 29–30. <https://doi.org/10.3969/j.issn.1001-4578.2006.02.010> (in Chinese).
- Naimi, A.I., Balzer, L.B., 2018. Stacked generalization: an introduction to super learning. *Eur. J. Epidemiol.* 33, 459–464. <https://doi.org/10.1007/s10654-018-0390-z>.
- Qiao, Y., Tu, X., Zhou, L., Guo, X., 2024. Hybrid convolutional and gated recurrent unit network with attention for drilling kick prediction. *SPE J.* 1–17. <https://doi.org/10.2118/223604-PA>.
- Rennard, V., Nikolentzos, G., Vazirgiannis, M., 2021. Graph auto-encoders for Learning Edge Representations. Proceedings of the Ninth International Conference on Complex Networks and their Applications Complex Networks 2020. https://doi.org/10.1007/978-3-030-65351-4_10.
- Scarselli, F., Gori, M., Tsoi, A.C., et al., 2008. The graph neural network model. *IEEE Trans. Neural Network.* 20 (1), 61–80. <https://doi.org/10.1109/TNN.2008.2005605>.
- Schafer, D.M., Loeppke, G.E., Glowka, D.A., Scott, D.D., Wright, E.K., 1992. An Evaluation of Flowmeters for the Detection of Kicks and Lost Circulation During Drilling. SPE/IADC Drilling Conference and Exhibition, New Orleans, Louisiana, USA. <https://doi.org/10.2118/23935-MS>.
- Sha, Q., Ding, Y., Cui, M., et al., 2024. Automatic Kick Detection Using Artificial Intelligence. International Petroleum Technology Conference. <https://doi.org/10.2523/IPTC-23603-EA>.
- Song, X., Duan, S., Pei, Z., Zhu, Z., 2021. Research on Kick Detection Model Based on Machine Learning. International Conference on Offshore Mechanics and Arctic Engineering, Virtual. <https://doi.org/10.1115/OMAE2021-62785>.
- Song, X.Z., Yao, X.Z., Li, G.S., et al., 2022. A novel method to calculate formation pressure based on the LSTM-BP neural network. *Petroleum Science Bulletin* 7 (1), 12–23. <https://doi.org/10.3969/j.issn.2096-1693.2022.01.002> (in Chinese).
- Tukey, J.W., 1977. *Exploratory Data Analysis*, vol. 2. Addison-Wesley Pub. Co, Reading, Mass. https://archive.org/details/exploratorydataa0000tukey_7616/mode/2up.
- Vandenbussche, V., Bergsli, A., Brandt, H., et al., 2012. Well-Specific Blowout Risk Assessment. International Conference on Health, Safety and Environment in Oil and Gas Exploration and Production. <https://doi.org/10.2118/157319-MS>.
- Velickovic, P., Cucurull, G., Casanova, A., et al., 2017. Graph attention networks. *Stat* 1050 (20), 10–48550. <https://doi.org/10.48550/arXiv.1710.10903>.
- Wang, J.B., Sun, B.J., Li, H., et al., 2017. Early gas kick detection based on the LWD resistivity in deepwater drilling. *Journal of China University of Petroleum (Edition of Natural Science)* 41 (6), 94–100. <https://doi.org/10.3969/j.issn.1673-5005.2017.06.011> (in Chinese).
- Yin, Q., Yang, J., Tyagi, M., et al., 2021. Field data analysis and risk assessment of gas kick during industrial deepwater drilling process based on supervised learning algorithm. *Process Saf. Environ. Prot.* 146, 312–328. <https://doi.org/10.1016/j.psep.2020.08.012>.
- Yin, Q., Yang, J., Tyagi, M., et al., 2022. Downhole quantitative evaluation of gas kick during deepwater drilling with deep learning using pilot-scale rig data. *J. Petrol. Sci. Eng.* 208, 109136. <https://doi.org/10.1016/j.petrol.2021.109136>.
- Zhang, D., Sun, W., Dai, Y., et al., 2024. Intelligent kick detection using a parameter adaptive neural network. *Geoenergy Sci. Eng.* 234, 212694. <https://doi.org/10.1016/j.geoen.2024.212694>.
- Zhang, S., Samuel, R., 2024. A Machine Learning Model for Real-Time Early Kick and Loss Prediction EKLP Application. The Abu Dhabi International Petroleum Exhibition and Conference. <https://doi.org/10.2118/221948-MS>.
- Zhao, D., Du, P., Liu, T., et al., 2023. Spatio-temporal distribution prediction model of urban theft by fusing graph autoencoder and GRU. *Journal of Geo-Information Science* 25 (7), 1448–1463. <https://doi.org/10.12082/dqxkx.2023.220584> (in Chinese).
- Zhou, D., Zhu, Z., Li, G., et al., 2024. Combining Drilling Condition Analysis with Unsupervised Time Series Models for Kick Monitoring. International Conference on Offshore Mechanics and Arctic Engineering, Singapore EXPO, Singapore. <https://doi.org/10.1115/OMAE2024-128411>.
- Zhu, Z., Zhou, D., Yang, D., et al., 2023. Early gas kick warning based on temporal autoencoder. *Energies* 16 (12), 4606. <https://doi.org/10.3390/en16124606>.