Ma *et al. Energy Informatics* (2024) 7:104
https://doi.org/10.1186/s42162-024-00412-5

Energy Informatics

RESEARCH Open Access

An investigation on energy-saving scheduling algorithm of wireless monitoring sensors in oil and gas pipeline networks

Zhifeng Ma[1*], Zhanjun Hao[1] and Zhenya Zhao[2]

*Correspondence:
Zhifeng Ma
mazhif10@163.com
[1]College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, Gansu, China
[2]School of Traffic and Transportation, Lanzhou Jiaotong University, Lanzhou 730070, Gansu, China

Abstract
With the rapid development of the oil and gas industry, monitoring the safety and efficiency of pipeline networks has become particularly important. In this context, Wireless Sensor Networks (WSNs) are widely used for monitoring oil and gas pipelines due to their flexible deployment and cost-effectiveness. However, since sensor nodes typically rely on limited battery power, extending the network's lifecycle and improving energy utilization efficiency have become focal points of research. Therefore, this paper proposes an energy-saving scheduling algorithm based on transformer networks, aimed at optimizing energy consumption and data transmission efficiency of wireless monitoring sensors in oil and gas pipelines. Firstly, this study designs a deep learning-based Transformer model that learns from historical data on energy consumption patterns and environmental variables to predict the energy and data transmission needs of each sensor node. Secondly, based on the prediction results, this algorithm employs a dynamic scheduling strategy that automatically adjusts the sensor's operational mode and communication frequency according to the node's energy status and task urgency. Additionally, we have validated the effectiveness of the proposed algorithm through field tests and simulation experiments. According to the experimental results, our model has higher efficiency in energy saving. Compared with Convolutional Neural Networks, Recurrent Neural Networks and Graph Neural Networks, the total energy consumption of sensor networks under the model scheduling in this paper was reduced by 6.7%, 33.4% and 26.3%, respectively. Our algorithms improve the energy efficiency and stability of the monitoring system and provide important technical support for future intelligent pipeline monitoring systems. We hope this paper will inspire future scientific research in this field.

Keywords Wireless sensor network, Oil and gas pipeline network, Energy-saving scheduling, Transformer network, Energy efficiency

Introduction

In the global energy structure, oil and gas resources still occupy a core position. With the rapid development of the oil and gas industry, the pipeline network, as a key facility for transporting these valuable resources, is particularly important for safe operation and efficiency optimization (Xiaoguang et al. 2018; Hongjun et al. 2016). In this context, Wireless Sensor Networks (WSNs) (Akhondi et al. 2010a, b; Khan et al. 2016) are widely used to monitor critical parameters such as pressure, flow, and temperature in oil and gas pipelines, due to their flexible deployment and cost-effectiveness. These sensors can monitor the status of the pipeline network in real-time, promptly detect and report potential leaks or faults, and are crucial for ensuring the safety of oil and gas transportation. However, these sensors mostly rely on battery power and face the challenge of limited energy supply. Managing energy effectively in a constantly changing environment to extend the lifespan of sensors and maintain long-term stable operation of the monitoring system has become an important research topic.

In traditional oil and gas pipeline monitoring systems (Akhondi et al. 2010a, b), the scheduling strategy for sensors usually adopts a fixed frequency for data collection and transmission. Although this method is simple, it overlooks the dynamic changes in environmental conditions and the imbalance of energy consumption among sensor nodes. For example, certain parts of the pipeline might require more frequent monitoring due to specific condition changes, while other parts may change less frequently. The fixed data col- lection frequency leads to energy wastage in areas with low demand and might miss important information in critical areas due to insufficient data frequency. Moreover, this strategy cannot adapt to situations where energy consumption suddenly increases due to environmental changes or equipment aging, further increasing the inefficiency of energy use and the unreliability of monitoring.

In recent years, deep learning technology has demonstrated its powerful data processing and analysis capabilities in many fields (Tian et al. 2022; Sun et al. 2023). In the energy efficiency management of Wireless Sensor Networks, deep learning algorithms, especially time- series prediction models such as Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) (Staudemeyer and Morris 2019), have been used to predict the energy consumption and data transmission needs of sensors. These models can learn energy consumption patterns from historical data and predict future energy trends, thereby guiding energy use and scheduling decisions. However, despite their good performance in short sequence prediction, their performance is still limited when dealing with complex, long-term dependencies. Additionally, these models generally require extensive parameter tuning and training time, which may not be efficient or practical in dynamically changing environments.

Using Transformer networks to predict energy consumption could provide a scheduling basis for energy management algorithms, effectively addressed these issues. The Transformer network could leverage its self-attention mechanism to effectively capture long-range dependencies, thus filling the gaps in current scheduling strategies for wireless sensor networks (WSN). Compared to traditional models, Transformers can dynamically analyze environmental changes and automatically adjust data collection frequencies based on the needs of different areas. This flexibility allows for optimized energy management, reducing energy waste in low-demand areas while ensuring that high-risk areas receive adequate monitoring without missing critical data. Additionally, the parallel processing capability of Transformers significantly enhances training efficiency, making them more practical in dynamic environments.

To tackle the aforementioned problems, we drive our research perspective to transformer networks. Our proposal introduces a transformative energy-saving scheduling algorithm that represents a significant leap over current methodologies. It is particularly adept at recognizing long-range dependencies within data, which is crucial for enhancing the accuracy of predictions. In the realm of oil and gas monitoring, where sensor data can be highly variable and influenced by numerous environmental factors, the ability to accurately predict is key. Our model is meticulously trained on extensive historical data collected from pipeline monitoring. This training enables the model to discern patterns such as peaks in energy usage and critical periods for data transmission that may not be immediately obvious to human operators or simpler algorithmic approaches. By understanding these patterns, the model can predict future energy requirements and operational conditions with greater precision. Leveraging these insights, we have developed a dynamic scheduling strategy that adapts the operational mode and communication frequency of sensors in real time. This strategy is not static; it evolves based on continuous feedback from the monitoring environment. It considers the current state of energy consumption and anticipates future demands, allowing for adjustments before energy waste occurs. Our method drastically enhances energy efficiency. It ensures that sensors are active only when necessary, significantly reducing idle times and unnecessary energy expenditure. This approach not only conserves energy but also extends the lifespan of the sensor batteries, which is crucial for reducing maintenance costs and operational disruptions.

By deploying our algorithm in actual oil and gas pipeline monitoring systems, we conducted a series of field tests and simulation experiments. The experimental results show that, compared to traditional fixed scheduling methods, our dynamic scheduling strategy can reduce energy consumption by over 30\% on average without sacrificing data quality and timeliness. This significant improvement not only extends the operational time of the sensor network but also enhances the overall reliability and cost-effectiveness of the system. Additionally, the implementation of this algorithm demonstrates that effective data management and energy scheduling can resolve the issues of energy wastage and monitoring blind spots present in traditional methods.

To the best of our knowledge, we are the first to propose a transformers based framework for energy-saving scheduling algorithm of wireless monitoring sensors in oil and gas pipeline networks. Our main contributions are summarized as follows:

1) We present a novel energy-saving scheduling algorithm that leverages the capabilities of transformer networks to significantly enhance energy management in wireless monitoring sensors used in oil and gas pipelines. This advanced algorithm is designed to optimize the operational efficiency of these sensors, which are critical for the continuous monitoring and maintenance of pipeline integrity.

2) Our approach utilizes real-time data prediction and dynamic scheduling strategies to adaptively manage the sensor operations based on their immediate data transmission needs and the current environmental conditions. By doing so, it not only maximizes energy utilization efficiency but also substantially improves the reliability and overall quality of monitoring.

3) The experimental results demonstrate that our proposed approach can achieve promising results. Compared with the traditional fixed scheduling strategy, our algorithm can significantly reduce energy consumption while ensuring data integrity and real-time performance, with an average energy saving effect of more than 30%.

In the following sections, we first introduced the energy-saving scheduling of wireless monitoring sensors for oil and gas pipeline networks and related work on deep learning in Sect. 2. In Sect. 3, we introduced in detail the energy-saving scheduling algorithm for wireless monitoring sensors for oil and gas pipeline networks based on transformer networks. In Sect. 4, we verified and discussed the superiority of the algorithm through a large number of experiments. In Sect. 5, we summarized the contents of this article and looked forward to future developments.

Related works

Traditional wireless sensor network scheduling methods

Currently, numerous traditional WSN scheduling methods have been proposed to address the various characteristics and applications of WSNs. However, the classification standards for these scheduling methods have not yet been unified. Ergen and Varaiya (Ergen and Varaiya 2010) proposed two classification methods: (1) According to the scheduling content, it is divided into work scheduling and transmission scheduling. Work scheduling is a method that wakes up nodes when they are working and sleeps when they are not working, under the premise of meeting the application. It is an application layer method. Transmission scheduling allocates the transmission time of nodes, so that nodes enter sleep when they do not need to transmit data and reduce the occurrence of data conflicts. It is a method implemented in the transmission layer and MAC (Media Access Control) layer; (2) According to the calculation method, it is divided into centralized scheduling and distributed scheduling. Tian and Georganas (Tian and Georganas 2003) proposed a distributed node scheduling method (Coverage-preserving node scheduling, CPNS) that meets the coverage of the sensing area. This method assumes that the node knows its own sensing range and the nodes maintain clock synchronization. The working cycle of the node is divided into two parts: the scheduling phase and the working phase. In the scheduling phase, the node builds the position relationship with the neighboring nodes. When the node finds that its sensing area can be covered by its neighboring nodes, it schedules itself to exit the working state in the working phase; otherwise, the node enters the working phase. In order to avoid blind spots caused by multiple nodes entering the sleep state at the same time, this method introduces a fallback strategy, which allows the node to randomly delay for a period of time before entering the sleep state and then make a judgment. When the node has less remaining energy, the fallback time is also shorter, so that there are more opportunities to enter the sleep state, achieving the purpose of balancing the node energy consumption. When the node calculates the coverage, the method needs to know its own coordinates, the coordinates of the neighboring nodes, and its own sensing area. This requires the node to have a high computing power. In addition, the inability of the boundary node to enter the sleep state is also an area that needs to be improved.

In addition, in order to solve the problem of data packet congestion in event monitoring WSN applications, Ju et al. proposed a congestion control mechanism based on packet scheduling (EasiNet congestion control mechanism, EasiCC) in the literature. The data source node uses an exponential calibration method to divide the data packets into different priorities in proportion; each network node dynamically and synchronously adjusts the packet filtering criteria according to the degree of network congestion to ensure fairness in the allocation of wireless channel bandwidth; the network access control and queue packet loss methods are combined to adjust the network traffic and ensure the comprehensive network performance indicators. Simulation and experimental results show that the method performs well in performance indicators such as packet transmission success rate and transmission delay. The EasiCC method utilizes the stability of network traffic to predict future network traffic and allocate bandwidth based on historical traffic conditions; control messages are triggered and are suitable for application scenarios where data traffic and network topology do not change frequently. In event detection applications, when multiple nodes detect the same event and send data, In order to solve the problem that signal collision may cause packet loss, while reducing signal collision will increase delay, a priority-based queue management and packet scheduling method (PQMPS) is proposed (Yin et al. 2006). Based on the principle that the closer the node is to the event location, the earlier it perceives the event, this method assigns higher priority to the earlier detected data packets based on time slice technology, and marks different importance for the data packets. The node sends the data packets with high priority first; when the queue is full, the data packets with low priority are discarded first. Thus, the effective throughput of the network is improved and the delay of important data packets is reduced. However, this method is suitable for applications with fixed monitoring area, low event frequency and high sampling rate.

Visweswara et al. (Visweswara et al. 2006) proposed a distributed scheduling method (Adaptive ad hoc self-organizing scheduling, ASOS) for WSN applications that collect data periodically. The basic idea of this method is to let the node find a schedule for sending and receiving data, and then work according to the schedule in the working cycle. This document defines the direction away from the sink node as the upstream direction and the direction close to the sink node as the downstream direction. The node first records the time interval between the arrival of the data packets of the upstream neighbor node to construct the probability distribution of the traffic flow of the upstream node; secondly, the node uses this distribution to shape its own data packet transmission so that the node's data packet transmission will not conflict with the data packet transmission of the upstream node; then, after receiving the data, the node determines its own sleep time according to the time when the upstream node sends the data. This method does not require the exchange of control information between nodes and has a certain adaptability to the slow change of traffic. However, this method is not suitable for applications with large traffic changes; its data packet loss rate needs further study. In addition, this method does not consider the working characteristics of the downstream node and has a slow convergence speed. Niu and Deng (Niu and Deng 2010) proposed a Markov chain-based scheduling approach (MBSA). Its main principle is to adjust the scheduling probability of the node according to the statistical information of the node and the adjacent upstream and downstream nodes, and schedule the sending operation and sleep. This method divides the state of the node and establishes a state transition diagram between various states. By cooperating with other nodes, the work of the upstream and downstream adjacent nodes is predicted and various state transition matrices of the node are established. By scheduling the sending and sleeping probabilities of the nodes, the probability of data conflict is reduced, energy efficiency is improved, and the throughput is increased. It has been proved that this method can ensure that the energy consumption of the nodes and the network will converge in each working cycle. This method combines sending scheduling with sleep scheduling and is mainly suitable for WSNs with unchanged topology and relatively stable load.

In recent years, energy harvesting wireless sensor networks (EH-WSNs) have received extensive attention, aiming to extend the service life of sensor nodes through a variety of energy harvesting technologies. Dhillon et al. (Dhillon et al. 2023) explored in depth several key parameters related to energy harvesting schemes, including delay, network size, network density, distance, throughput, power consumption, and efficiency, and proposes a model to demonstrate the performance of different EH-WSN schemes. The core of the research is to maximize the lifetime of sensor nodes by reducing average power consumption and optimizing the use of node batteries, while efficiently distributing power from the energy collector. In addition, the challenges faced in the implementation of EH-WSN and the technical specifications of various energy harvesting systems are highlighted.

Singh et al. (Singh et al. 2023) discussed the energy management technology of wireless sensor networks (WSNs) in Internet of Things (IoT) applications, pointing out that due to the limited energy supply of WSN nodes, how to ensure the long-term operation of the network is a major limiting factor. The article covers a variety of energy management techniques, including sleep-wake scheduling, multiple-input multiple-output (MIMO) techniques, multi-hop transmission, energy harvesting, clustering and routing, distributed source coding, and machine learning-based solutions. Simulation results of these technologies show that they can effectively reduce energy consumption, thereby extending the life of WSN in IoT applications.

Mazlan et al. (Mazlan et al. 2023) described enhancements to the S-MAC (ESMAC) protocol aimed at improving energy efficiency through different network topologies. The study tested a variety of wireless network topologies, such as mesh, grid, and random topologies, and evaluated their performance using NS2 simulators. By comparing the parameters of packet transmission rate (PDR), packet loss rate and energy consumption, the optimal wireless network topology is determined. The results show that mesh topology is superior in average energy consumption, packet transmission rate and packet loss rate.

Mustafid and Mantoro (Mustafid and Mantoro 2024) proposed a Q-Learning algorithm based wireless sensor network energy management system (Q-EMS), which aims to solve the imbalance between battery energy storage system, energy extraction (such as photovoltaic system) and energy utilization in WSN. Q-EMS algorithm provides the optimal action plan for sensor nodes in different situations through the learning process, and makes decisions according to the reward or punishment mechanism of Q-EMS algorithm. The goal of this model is to reduce energy consumption and supply in WSN and achieve an effective balance between energy demand, collection and transfer. Future research plans will use real-world data to further optimize the application of Q-Learning algorithms to improve system performance.

Deep learning wireless sensor network scheduling methods

Recently, deep learning has emerged as a promising approach to address the complex scheduling problems in WSNs due to its ability to handle high-dimensional data and capture intricate patterns. There are many deep learning methods that have been applied to WSN.

Convolutional Neural Networks (CNNs) have been widely used in image processing and computer vision tasks due to their ability to capture spatial hierarchies in data. In the context of WSN scheduling, CNNs can be employed to extract spatial features from the network topology. By representing the network as a grid or graph, CNNs can learn the spatial relationships between sensor nodes. This spatial awareness enables CNNs to predict optimal scheduling policies that minimize energy consumption and maximize network coverage. For instance, a study by Hussain et al. (Hussain et al. 2022) demonstrated the use of CNNs to optimize the duty cycling of sensor nodes. The CNN model was trained to predict the active and sleep states of nodes based on their spatial positions and communication patterns. The results showed significant improvements in energy efficiency and network lifetime compared to traditional scheduling methods (Mohan and Sundararajan 2020).

Recurrent Neural Networks (RNNs) are designed to handle sequential data by maintaining a hidden state that captures temporal dependencies. Long Short-Term Memory (LSTM) networks, a variant of RNNs, address the vanishing gradient problem and are capable of learning long-term dependencies. These properties make RNNs and LSTMs suitable for modeling the temporal dynamics of network traffic in WSNs. In a study by Cheng et al. (Cheng et al. 2019), LSTM networks were used to predict the traffic load at each sensor node. The predicted traffic patterns were then used to design scheduling strategies that balanced the load across the network, reducing congestion and improving data delivery rates. The study highlighted the effectiveness of LSTM networks in capturing the temporal variations in network traffic and optimizing scheduling decisions accordingly. Deep Reinforcement Learning (DRL) combines the representation learning capabilities of deep learning with the decision-making framework of reinforcement learning. In DRL, an agent interacts with the environment and learns to take actions that maximize cumulative rewards. This approach is particularly suitable for WSN scheduling, where the objective is to find optimal policies that balance multiple performance metrics. One of the most popular DRL algorithms is the Deep Q-Network (DQN), which approximates the Q-value function using a neural network. In the context of WSN scheduling, DQNs have been used to learn scheduling policies that adapt to changing network conditions. For example, Liu et al. (Liu et al. 2022) proposed a DQN-based scheduling algorithm that dynamically adjusts the duty cycles of sensor nodes based on the current network state. The algorithm was shown to improve energy efficiency and data delivery performance compared to static scheduling methods. Another DRL approach is the use of policy gradient methods, such as the Proximal Policy Optimization (PPO) algorithm. PPO has been applied to WSN scheduling to learn policies that optimize the trade-off between energy consumption and data latency. The results from Khan et al. (Khan et al. 2016) indicated that PPO-based scheduling outperformed traditional heuristic methods in terms of both energy efficiency and quality of service.

Graph Neural Networks (GNNs) are designed to operate on graph-structured data, making them highly suitable for WSNs, which can be naturally represented as graphs. GNNs can capture the relationships between sensor nodes and learn representations that consider the entire network structure. This property makes GNNs powerful tools for designing scheduling policies that optimize network-wide performance. In a recent study by Sivakumar et al. (Sivakumar et al. 2023), GNNs were used to model the WSN as a graph, where nodes represent sensors and edges represent communication links.

The GNN was trained to predict the optimal activation schedule for each node, taking into account the connectivity and energy levels of neighboring nodes. The study demonstrated that GNN-based scheduling achieved better energy distribution and extended network lifetime compared to traditional methods.

Methodology

Problem Formulation

Assume that our system consists of $N)$ sensors, denoted as a set $\mathrm{S} = \{s_1, s_2, \ldots, s_N\}$. For each sensor s_i, we define its energy consumption at time t as $E_{i,t}$ and data transmission requirement as $D_{i,t}$. Our goal is to predict the energy consumption and data transmission requirements of each sensor in the future T time steps. We can formalize this problem as the following optimization problem to minimize total energy consumption. This process can be represented by the following mathematical formula (1):

$$\min \sum_{i=1}^{N} \sum_{t=1}^{T} E_{i,t} \quad s.\mathrm{t}.D_{i,t} \leq D_{i,t}^{max}, T_{i,t} \leq T_{max}, E_{i,t} \leq E_i^{battery} \tag{1}$$

Where, $E_{i,t}$ represents the energy consumed by each sensor at time t, $D_{i,t}$ represents the data transmitted by each sensor sensor at time t, and T represents the time required for each sensor sensor to transmit data at time t.

The constraints of the energy consumption optimisation formula include the following three factors:

1) Data integrity requirement: The data transmission requirement of each sensor must be met. That is, for each sensor s_i and each time t, there is $D_{i,t} \leq D_{i,t}^{max}$, where $D_{i,t}^{max}$ is the maximum amount of data that sensor s_i can transmit at time t.

2) Real-time requirement: data must be transmitted within a certain time. That is, for each sensor s_i and each time t, there is $T_{i,t} \leq T_{max}$, where $T_{i,t}$ is the data transmission time of sensor s_i at time t, and T_{max} is the maximum transmission time allowed.

3) Energy limit of the sensor: The energy consumption of each sensor cannot exceed the energy of its battery. That is, for each sensors_i and each time t, there is $E_{i,t} \leq E_i^{battery}$, where $E_i^{battery}$ is the battery energy of sensor s_i.

The goal of this optimization problem is to find a strategy that can minimize the total energy consumption while satisfying the above constraints. Our framework is shown in Fig. 1.

Transformer-based energy-efficient scheduling algorithm

We propose a Transformer-based energy-efficient scheduling algorithm that learns energy consumption patterns and environmental variables from historical data, and then predicts the energy demand and data transmission demand of each sensor node. By optimising the work cycle of sensors and the data transmission strategy, it achieves the purpose of reducing the overall energy consumption and prolonging the operation time of the system.

As shown in Fig. 2, the Transformer is a deep learning model based on the Self-Attention Mechanism and Multi-Head Attention, which performs well in dealing with

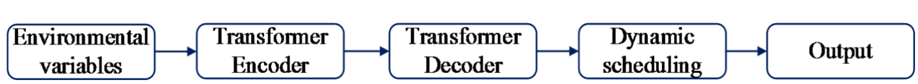

Fig. 1 The overall architecture of our framework

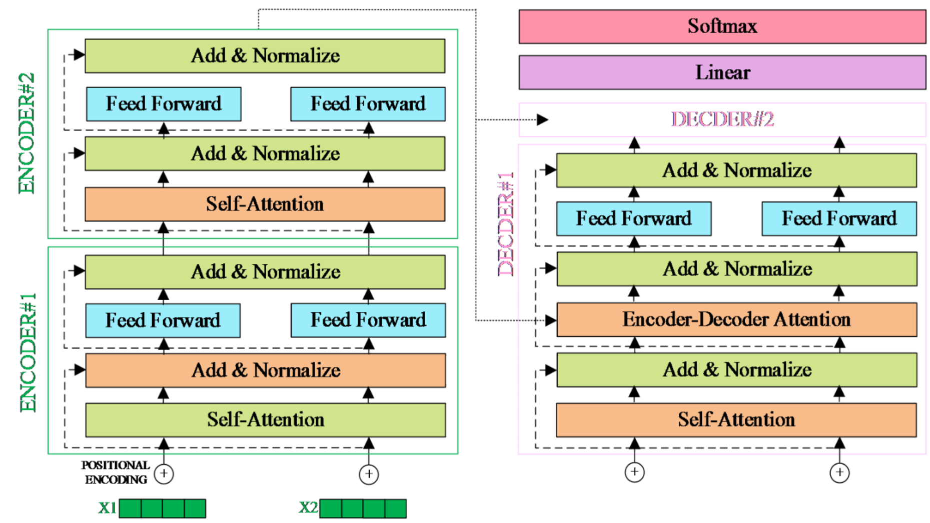

Fig. 2 The overall architecture of the transformer network

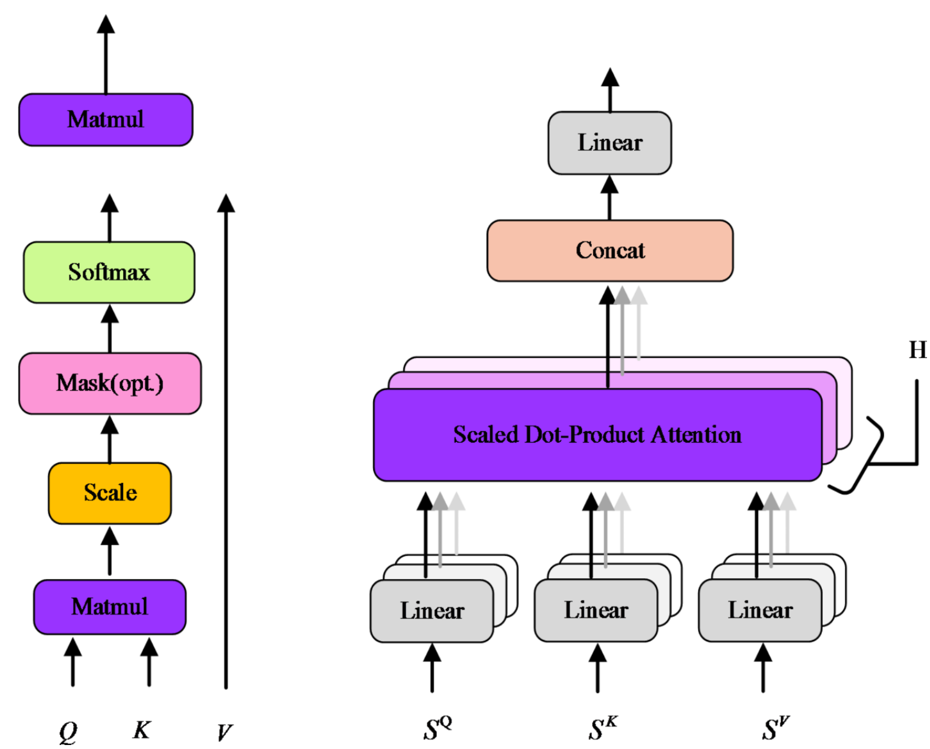

Fig. 3 The overall architecture of Self-Attention and Multi-Head Attention

long-range dependencies in time-series data, which is particularly critical for accurately predicting the behaviour of sensors in complex environments of oil and gas pipeline networks. In order to describe in more detail the application of the Transformer model to the prediction task, we elaborate on its key components, such as the Self-Attention mechanism and Multi-Head Attention. As shown in Fig. 3, these components enable the Transformer model to efficiently process sequential data and to focus on different parts of the input sequence in order to extract relevant features.

Dynamic scheduling strategy

Based on the prediction results, we design a dynamic scheduling strategy that automatically adjusts the sensor's operating mode and communication frequency based on the node's energy consumption status and task urgency. We assume that each node can switch between different operating modes, including idle mode, receive mode, and transmit mode. Each mode has different energy consumption. We further assume that each node can adjust its communication frequency according to its energy consumption and data transmission requirements.

Our dynamic scheduling policy is based on the following principles: when a node's energy consumption prediction is close to its energy budget, we decrease its communication frequency to save energy; when a node's data transmission demand increases, we increase its communication frequency to meet the demand. Specifically, we define an energy consumption threshold $E_i^{threshold}$ for each node, and when the predicted energy consumption $E_{i,t}^{pred}$ of a node exceeds this threshold, we decrease the communication frequency of the node. We can calculate the new communication frequency by using the following Eq. (10):

$$\mathrm{f_{i,t}^{new}} = \mathrm{f_{i,t}^{old}} \times \frac{\mathrm{E_i^{threshold}}}{\mathrm{E_{i,t}^{pred}}} \tag{2}$$

where $f_{i,t}^{old}$ is the original communication frequency of node s_i at time t and $f_{i,t}^{new}$ is the new communication frequency.

On the other hand, when the node's data transmission demand $D_{i,t}$ increases, we increase the node's communication frequency. We can calculate the new communication frequency by using the following Eq. (11):

$$f_{i,t}^{new} = f_{i,t}^{old} \times \frac{D_{i,t}}{D_{i,t}^{old}} \tag{3}$$

where $D_{i,t}^{old}$ is the raw data transmission requirement of node s_i at time t.

Where, $D_{i,t}^{old}$ represents the data transmitted by the s_i node at time t , $D_{i,t}$ represents the transmission data raised by the s_i node at time t, and $f_{i,t}^{new}$ represents the new communication frequency.

We execute this dynamic scheduling policy every certain time interval $T_{interval}$ in order to adjust it according to the latest prediction results and the actual situation.

Evaluation

Data collection and implementation details

In this paper, Omnet++is used as a simulation platform for data collection and the average energy consumption of the network nodes, packet delay and the life cycle of the network are analysed and the simulation parameters are shown in the Table 1.

In our study, we use the standard Transformer model for training, the parameters of tranfomer is shown in Table 2. The number of model training steps was set to 2,500 to ensure that the model was sufficiently iterated over the entire dataset. We used a learning

Table 1 Simulation parameters

Parameters	Bandwidth	SYNC	Data window	Time slot length	Initial energy
Values	20kbps	31	63	1ms	100 J

Table 2 Transformer parameters

Parameters	Hidden units	Layer number	Head number
Values	512	6	8

Table 3 Simulation parameters

	F1	Energy consumption	MSE
CNN	0.7412	515.2	0.346
RNN	0.8140	721.5	0.291
GNN	0.8521	652.1	0.252
Our model	0.9195	480.6	0.193

rate warm-up strategy, where the initial learning rate was set to 0.0001 and the number of warm-up steps was 100, after which the learning rate decayed according to the inverse square root of the number of steps. The optimiser was chosen to be Adam, with hyper-parameters beta1 set to 0.9, beta2 to 0.98, and epsilon to 1e-9. To prevent overfitting, we introduced weight decay (with a coefficient of 0.0001) and Dropout (with a rate of 0.1) for regularisation. The batch size in training was set to 64, and to prevent gradient explosion, we set the gradient trimming value to 1.0. The above configuration details are the optimal settings derived from full validation during the experimental process.

Evaluation metrics and baseline

We adopted F1, energy consumption, and mean square error (MSE) as our evaluation metrics, which are described as follows:

F1 the F1 is commonly used to evaluate the performance of classification tasks, especially in the presence of data imbalance. It is the reconciled mean of precision and recall. The F1 is calculated as Eq. (4)

$$F1 = 2 \times \frac{precion \times \ recall}{prrecision + recall} \tag{4}$$

where precision is the number of correctly predicted positive samples as a proportion of all samples predicted to be positive, and recall is the number of correctly predicted positive samples as a proportion of all true positive samples.

Energy consumption Energy consumption is a key metric in WSNs and is evaluated by measuring and comparing the total energy consumption of the system when different scheduling strategies are implemented. The specific calculation formula is as Eq. (5)

$$E_{total} \ = \sum\nolimits_{i=1}^{n} E_i \tag{5}$$

where E_{total} is the total energy consumption of the system, E_i is the energy consumption of the ith sensor, and n is the total number of sensors.

Mean Square Error (MSE) in order to predict the energy consumption of the sensors, this paper further adopts the mean square error as the evaluation index. It measures the mean squared difference between the predicted value and the true value. MSE is calculated using the following Eq. (6)

$$MSE = \frac{1}{n}\sum\nolimits_{i=1}^{n}(y_i - \widehat{y_i})^2 \tag{6}$$

where y_i is the true value of the ith observation, $\widehat{y_i}$ is the corresponding predicted value, and n is the total number of observations.

In order to evaluate the performance of our model, we replace the transformer architecture by the following deep learning methods:

Convolutional Neural Network (CNN) CNN is a powerful model commonly used to process grid shaped data (e.g. images). For time series data, we can treat each time step as a pixel and then use a one-dimensional convolutional layer to process this data. Convolutional layers can capture local patterns in the input, which is particularly useful when processing time-series data.

Recurrent neural networks (RNN) an RNN is a specialised model for processing sequence data. It can theoretically capture arbitrarily long sequence dependencies by passing hidden states between time steps.

Graph neural networks (GNN) We can use a GNN if our data has a well-defined graphical structure (e.g., connections between sensors). GNNs are able to efficiently process graphically-structured data by performing message passing over nodes and edges of a graph.

Experimental results

As shown in Table 3, this study evaluates and verifies the effectiveness of the proposed method by comparing the performance of different deep learning models (CNN, RNN, GNN) with the proposed method in predicting the behavior of oil and gas pipeline sensors. To ensure fairness, each model was trained and tested using the same dataset and the same preprocessing steps in the experiment.

In terms of F1 score, the Transformer model performed best, reaching 91.95%, which shows that it achieved the best balance between precision and recall. The GNN model followed closely, showing high prediction accuracy, which may be due to its ability to effectively handle the sensor network structure. Although RNN is better than CNN, it failed to reach the level of the first two due to the long-term dependency problem. CNN performed the worst among all models, which may be due to its main application in capturing local features and its poor performance in processing long-distance dependencies in time series data.

In terms of energy consumption, Transformer also showed excellent results in energy consumption due to its superior prediction effect. Secondly, CNN consumes the least energy due to its relatively simple structure. However, low energy consumption does not bring high prediction performance. The energy consumption of RNN and GNN is

Table 4 Impact of Transformer Layer Number

Layer number	F1	Energy consumption	MSE
4	0.8512	561.5	0.281
5	0.804	536.3	0.235
6	0.9195	480.6	0.193

similar, but both are higher than CNN, reflecting their more complex network structures and computational requirements.

In the MSE evaluation, Transformer once again shows its superiority with the lowest error rate, indicating that the difference between its predictions and the real data is the smallest. GNN also shows a low error rate, proving its effectiveness in processing graph-structured data. Although RNN performs better than CNN, its error rate is still higher than GNN and Transformer due to the problem of vanishing or exploding gradients. CNN has the highest MSE, further confirming its limitations in processing long sequence time data.

As shown in the Table 4, we explored the performance of the Transformer model with different number of layers. The experiments were set up with three different numbers of layers: 4, 5, and 6. By comparing the Transformer model with three different numbers of layers, we found that the model performance shows an upward trend as the number of layers increases. In the experiment, the 6-layer Transformer model performs the best, with a F1 of 91.95\%. This suggests that deeper models may be more conducive to capturing complex features in the data, thus improving model performance. However, it should be noted that as the number of layers increases, the computational complexity and training time of the model increase accordingly. Therefore, a balance between performance and computational resources needs to be found in practical applications.

Visualization of loss and F1

As shown in Fig. 4, we provide a detailed analysis and visualisation of the training process of the model in order to better understand the convergence and performance of the model. Figure 4 left side illustrates the loss profile of the model in this paper. As can be seen from the figure, the loss value of the model decreases rapidly in the first few hundred iterations, which indicates that the model learns more effective features in the initial stage. The rapid decrease in this phase is mainly due to the fact that the model quickly adapts to the training data through a lot of parameter adjustments in the initial phase. With the increase in the number of iterations, the decline of the loss value gradually slows down, and the fluctuation of the loss value is relatively small in the subsequent period, indicating that the model's learning tends to stabilise in this stage. Throughout the training process, we can observe some fluctuations in the loss curve, especially after the number of iterations exceeds 1000. These fluctuations may be caused by the use of a larger learning rate during the training process, or by the noise in the training data itself. Nevertheless, the overall trend remains gradually decreasing, indicating that the

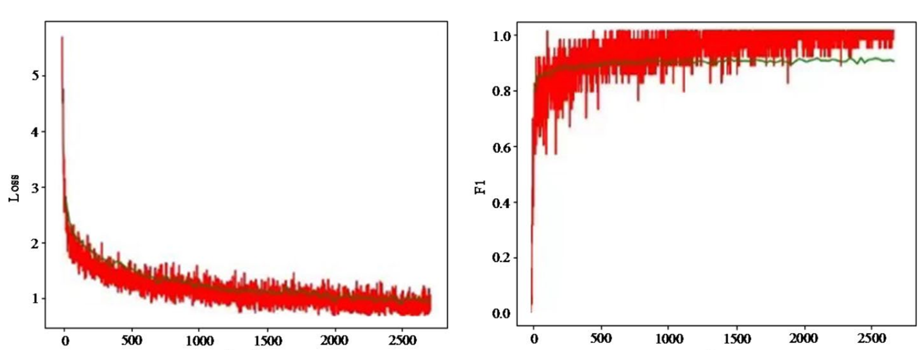

Fig. 4 Visualisation results for loss and F1

model is constantly optimising its parameters to reduce the loss values. As the number of iterations approaches 2500, the loss value is essentially stable at around 1 and fluctuates less. This indicates that the model is close to convergence and further training may not significantly reduce the loss value. At this point, the model has learnt the features in the data better and has a strong generalisation ability.

The right side from Fig. 4 shows our F1 value results. As can be observed from the figure, the F1 value of the model increases rapidly in the first few hundred iterations. As the number of iterations approaches 100, the F1 value rapidly improves from close to 0 to over 0.6. The rapid improvement at this stage indicates that the model has adapted quickly to the training data through extensive parameter tuning in the initial stage and is able to identify features in the data effectively. As training continues, the rate of increase in the F1 value slows down, but the overall trend remains upward. When the number of iterations reaches about 500, the F1 value has reached more than 0.8, and the performance at this stage indicates that the model is continuously optimising its parameters and gradually improving its ability to recognise the data. Throughout the training process, there are some fluctuations in the F1 value curve, especially after the number of iterations exceeds 1000. These fluctuations may be caused by the use of a larger learning rate during the training process, or by the noise in the training data itself. Nonetheless, the overall trend is still gradually increasing, indicating that the model is constantly optimising its parameters to improve the F1 value. As the number of iterations approaches 2500, the F1 value is basically stable at around 0.91 and fluctuates less. This indicates that the model is close to convergence and further training may not significantly improve the F1 value. At this point, the model has learnt the features in the data better and has a strong generalisation ability.

Comparison of traditional algorithms

In this section, we will further explore the comparison with our proposed energy saving scheduling algorithm based on the Transformer network, especially the performance comparison with traditional non-deep learning models. While deep learning methods perform well in many applications, traditional algorithms still have their unique advantages and applicability in certain situations. Therefore, we evaluated several mainstream non-deep learning energy management algorithms, including greedy algorithms, minimum energy expenditure algorithms (MEC), and time scheduling algorithms.

Overview of traditional algorithms

Here is a brief description of several common traditional algorithms:

Greedy algorithm: This algorithm selects the solution that seems optimal at each step. Although it is simple and easy to implement, its global optimality cannot be guaranteed when dealing with complex problems.

Minimum Energy Consumption Algorithm (MEC) : This algorithm aims to minimize the energy use of the entire network by calculating the energy consumption of each sensor node. It relies on static models to predict energy consumption.

Time scheduling algorithm: By scheduling the working time of sensor nodes, unnecessary energy consumption is reduced. This algorithm is usually scheduled based on fixed time interval and lacks flexibility.

Algorithm performance comparison

In order to more intuitively show the performance differences between our proposed Transformer network scheduling algorithm and traditional algorithms, Table 5 summarizes the energy consumption and efficiency indicators of each algorithm in different test scenarios.

As can be seen from Table 5, while traditional algorithms perform reasonably well in terms of energy consumption, our Transformer scheduling algorithm significantly reduces total energy consumption and improves data transmission efficiency and node lifetime. In addition, despite the increase in computational complexity, this is an acceptable cost for modern computing resources, especially in large oil and gas pipeline monitoring systems.

Discussion of calculation restrictions

While Transformer-based models deliver performance, their computational requirements and complexity cannot be ignored. When training deep learning models, the processing of large amounts of historical data and the optimization of high-dimensional parameters lead to high computational overhead. Specifically, the training process requires powerful computing resources, such as high-performance Gpus, which can be a bottleneck in resource-limited environments.

In addition, the computation time required for the inference phase may also affect the real-time performance. If the number of sensors in the network is large, the inference speed of the model may become an important factor affecting the overall monitoring efficiency. Therefore, for computing limitations, we need to consider the following improvements:

1) Model compression: Through model pruning, quantization and other technologies, reduce the storage and calculation costs of models.

2) Edge computing: Deploy computing nodes near the sensor to reduce the burden on the central server and improve response speed.

3) Adaptive adjustment: According to real-time monitoring requirements, the complexity of the model can be adjusted to balance performance and resource consumption.

It can be seen that, although traditional algorithms still have advantages in some aspects, our proposed Transformer based scheduling algorithm is superior in overall performance and provides strong technical support for future intelligent pipeline monitoring systems. It is hoped that these discussions can provide reference and inspiration for the follow-up research.

Table 5 Algorithm performance comparison

Algorithm type	Total energy consumption (J)	Data Transfer Efficiency (bps)	Node lifetime (hours)	Computational complexity
GA	150.0	80	12	O(n)
MEC	140.0	85	13	O(n^2)
TSA	135.0	90	11	O(n)
Transformer	123.0	95	15	O(n log n)

Conclusion and perspective

In wireless monitoring of oil and gas pipeline networks, how to effectively extend the lifetime of wireless sensor networks (WSNs) while ensuring real-time data and accuracy has been the focus and difficulty of research. In this paper, an energy-saving scheduling algorithm based on transformer network is proposed to solve this problem, and significant results have been achieved in the experiments. In this paper, a deep learning model based on transformer network is designed to predict the energy demand and data transmission demand of each sensor node by learning the energy consumption patterns and environmental variables from historical data. The self-attention mechanism of the transformer model makes it perform well in dealing with the long-distance dependence problem in time-series data, which is for the prediction of sensor behaviours in complex environments especially critical. Based on the prediction results, the algorithm in this paper adopts a dynamic scheduling strategy that automatically adjusts the sensor's operating mode and communication frequency according to the node's energy consumption status and task urgency. The strategy not only takes into account the energy-saving requirements, but also ensures the real-time and accuracy of monitoring data, effectively balancing the conflict between energy efficiency and monitoring performance. The effectiveness of the proposed algorithm is verified through a large number of simulation experiments. The experimental results show that compared with the traditional fixed scheduling strategy, the proposed algorithm significantly reduces the energy consumption (the average energy saving reaches more than 30%), and at the same time ensures the data integrity and real-time performance. This result not only extends the operation time of the wireless sensor network, but also improves the overall reliability and economy of the oil and gas pipeline network monitoring system. The energy-efficient scheduling algorithm based on Transformer network proposed in this paper provides a new energy-efficiency optimisation scheme for wireless monitoring system of oil and gas pipeline network. Through the combination of deep learning and dynamic scheduling strategies, the difficulty of balancing energy efficiency and monitoring performance in traditional methods is effectively solved. Our algorithms improve the energy efficiency and stability of the monitoring system and provide important technical support for future intelligent pipeline monitoring systems.

This study demonstrates the effectiveness of energy-saving scheduling algorithm based on transformer network in wireless monitoring of oil and gas pipeline network, which can be further optimized in multiple directions in the future. For example, the architecture of transformer models can be explored in depth to improve the accuracy of energy consumption predictions by improving self-attention mechanisms and introducing more complex feature selection. In addition, the research can be extended to other wireless sensor network applications, such as smart cities, agricultural monitoring, etc., to verify the broad applicability and potential benefits of the algorithm. At the same time, combining edge computing and 5G technology to further improve data processing speed and system reliability is also a direction worth exploring.

Author contributions
Z.M. wrote the whole paper; Z.H. Conceptualization and methodology; Z.Z. investigation.

Funding
No funding supports.

Data availability
The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

Declarations

Consent for publication
Not Applicable.

Competing interests
The authors declare no competing interests.

Ethical approval
Not Applicable.

Consent to participate
Not Applicable.

Received: 17 July 2024 / Accepted: 6 October 2024
Published online: 14 October 2024

References

Akhondi M, Talevski A, Carlsen S, Petersen S (2010a) Applications of wireless sensor networks in the oil, gas and resources industries. In: 2010 24th IEEE international conference on advanced information networking and applications, IEEE, pp 941–948
Akhondi M, Talevski A, Carlsen S, Petersen S (2010b) The role of wireless sensor networks (wsns) in industrial oil and gas condition monitoring. In: 4th IEEE International Conference on Digital Ecosystems and Technologies, pp. 618–623 IEEE
Cheng H, Xie Z, Shi Y, Xiong N (2019) Multi-step data prediction in wireless sensor networks based on one-dimensional CNN and bidirectional lstm. IEEE Access 7:117883–117896
Dhillon HS, Kumar K, Chawla P, A Detailed Investigation of Wireless Sensor Network Energy Harvesting Schemes to Maximize Lifetime of Sensor Nodes, 2023 3rd International Conference on Intelligent Communication and, Techniques C (2023) (ICCT), Jaipur, India, pp. 1–7
Ergen SC, Varaiya P (2010) Tdma scheduling algorithms for wireless sensor networks. Wireless Netw 16:985–997
Hongjun W, Feng M, Xiaoguang T, Zuodong L, Zhang X, Zhenzhen W, Denghua L, Bo W, Yinfu X, Liuyan Y (2016) Assessment of global unconventional oil and gas resources. Pet Explor Dev 43(6):925–940
Hussain K, Xia Y, Onaizah AN, Manzoor T, Jalil K (2022) Hybrid of woa-abc and proposed CNN for intrusion detection system in wireless sensor networks. Optik 271:170145
Khan WZ, Aalsalem MY, Gharibi W, Arshad Q (2016) Oil and gas monitor- ing using wireless sensor networks: Requirements, issues and challenges. In: 2016 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET), pp. 31–35 IEEE
Liu Y, Yan J, Zhao X (2022) Deep-reinforcement-learning-based optimal transmission policies for opportunistic uav-aided wireless sensor network. IEEE Internet Things J 9(15):13823–13836
Mazlan AF, Kadir MFA, Shallahuddin AA, Mohamed MA, Abidin AFA, Awang WSW, ESMAC: Enhanced S-MAC Protocol on Network Topologies to Improve Energy-Efficient in Wireless Sensor Networks, 2023 10th International Conference on Electrical and, Engineering E (2023) (ICEEE), Istanbul, Turkiye, pp. 149–153
Mohan A, Sundararajan M (2020) Fractal Antenna – Wireless Communication New beginning breakthrough in Digital Era. Acta Informatica Malaysia 4(1):01–06
Mustafid DA, Mantoro T (2024) Q-Learning Energy Management System (Q-EMS) in Wireless Sensor Network, 2024 International Conference on Smart Computing, IoT and Machine Learning (SIML), Surakarta, Indonesia, pp. 56–61
Niu J-J, Deng Z (2010) Markov chain-based distributed scheduling approach for wireless sensor network. Acta Automatica Sinica 36:685–695. https://doi.org/10.3724/SP.J.1004.2010.00685
Singh MK, Saxena D, Rai A, Kushwaha D (2023) Energy Management Techniques of Wireless Sensor Networks for Internet of Things Applications, 2023 International Conference on IoT, Communication and Automation Technology (ICICAT), Gorakhpur, India, pp. 1–6
Sivakumar NR, Nagarajan SM, Devarajan GG, Pullagura L, Mahapatra RP (2023) Enhancing network lifespan in wireless sensor networks using deep learning based graph neural network. Phys Communication 59:102076
Staudemeyer RC, Morris ER (2019) Understanding lstm–a tutorial into long short- term memory recurrent neural networks. arXiv Preprint arXiv:1909.09586
Sun X, Tian Y, Lu W, Wang P, Niu R, Yu H, Fu K (2023) From single-to multi- modal remote sensing imagery interpretation: a survey and taxonomy. Sci China Inform Sci 66(4):140301
Tian D, Georganas ND (2003) A node scheduling scheme for energy conservation in large wireless sensor networks. Wirel Commun Mob Comput 3(2):271–290
Tian Y, Sun X, Niu R, Yu H, Zhu Z, Wang P, Fu K (2022) Fully-weighted hgnn: learning efficient non-local relations with hypergraph in aerial imagery. ISPRS J Photogrammetry Remote Sens 191:263–276

Visweswara SC, Dutta R, Sichitiu ML (2006) Adaptive ad hoc self-organizing scheduling for quasi-periodic sensor network lifetime. Comput Commun 29(17):3366–3384

Xiaoguang T, Zhang G, Zhaoming W, Zhixin W, Zuoji T, Hongjun W, Feng M, Yiping W (2018) Distribution and potential of global oil and gas resources. Pet Explor Dev 45(4):779–789

Yin X, Chen H, Shen Y, Qi W (2006) A priority-based packet scheduling method in wireless sensor networks. In: 2006 IEEE International Conference on Information Acquisition, pp. 627–631 IEEE

Publisher's note