Large Language Models as Abstract Feature Enhancers for Time-series-based Fault Diagnosis

Takanobu Minami[1,2], Dai-Yan Ji[1], and Lee Jay[1]

[1]*Center for Industrial Artificial Intelligence, Department of Mechanical Engineering, University of Maryland, College Park, MD, USA*

minamitu@umd.edu , jidn@umd.edu , leejay@umd.edu

[2]*Komatsu Ltd., Tokyo, Japan*

ABSTRACT

In recent years, Deep Learning has shown remarkable success in fault diagnosis for Prognostics and Health Management, automatically extracting features from complex sensor data. However, its application to real-world industrial systems is often hampered by fundamental limitations such as the scarcity of comprehensive fault data and the difficulty of creating accurate physics-based models for complex systems. This lack of sufficient data and explicit domain knowledge makes it challenging for conventional models to distinguish different fault modes that produce highly similar sensor patterns. To compensate for this scarcity, this paper proposes a novel and versatile framework that leverages the vast, pre-trained knowledge of a Large Language Model (LLM) to enrich features extracted from limited sensor data. The framework connects an arbitrary time-series feature extractor to a frozen-weight LLM via a trainable adapter layer, using the LLM as an efficient feature enhancer. We demonstrate its effectiveness and versatility on a challenging rock drill fault diagnosis task, which suffers from both the aforementioned data ambiguity and significant domain shift. Experimental results show that our proposed method outperforms the baseline models, achieving the highest performance with an Accuracy of 0.811 and a Macro F1-Score of 0.793. Notably, the classification accuracy for fault classes that were conventionally difficult to identify improved significantly, indicating that the utilization of abstract knowledge from LLMs is highly effective for building more robust and accurate fault diagnosis systems.

1. INTRODUCTION

In modern manufacturing and social infrastructure, enhancing equipment reliability and reducing operational costs are critical management challenges. Prognostics and Health Management (PHM), a core technology for preventing failures by monitoring and predicting equipment status based on sensor data, is gaining increasing importance. In particular, fault diagnosis, the fundamental task of detecting signs of abnormality and identifying their causes, constitutes the cornerstone of PHM. In recent years, numerous studies have reported that Deep Learning (DL) models can achieve high diagnostic performance by automatically extracting features from complex time-series sensor data.

However, the practical application of such data-driven methods to real-world industrial environments still faces fundamental challenges. Industrial systems are often characterized by limited data availability, especially for various fault conditions, and their physical complexity makes it difficult to construct accurate physics-based models. Therefore, the ability to learn robust and highly discriminative feature representations from scarce numerical data is paramount. This becomes particularly critical when trying to distinguish between different fault modes that exhibit highly similar waveform patterns, a task where conventional DL models often struggle, as they can only form an understanding based on the patterns found within the provided training data.

To address this fundamental challenge, this study proposes a novel framework that integrates the extensive knowledge of pre-trained Large Language Models (LLMs) into time-series fault diagnosis models. Here, "Large Language Model (LLM)" specifically refers to general-purpose text-based LLMs (e.g., GPT-4), which we leverage for knowledge transfer and reasoning support. We distinguish these from pre-trained time-series foundation models (e.g., TimesNet, MOMENT), which are complementary but not the focus of this study. Our central hypothesis is that by leveraging the abstract conceptual understanding and contextual reasoning capabilities that LLMs acquire from vast textual data, we can enhance the feature representations extracted from sensor

data. This approach aims to move beyond mere numerical pattern matching and capture more essential, robust features, thereby improving diagnostic performance even with limited data. Importantly, the LLM component is treated as a drop-in module: the interface consumes task prompts and returns abstract features that are then fused in our similarity-based pipeline, without relying on model-specific internals. While broader benchmarking across multiple LLMs and datasets is valuable, our primary goal here is to introduce and validate the methodology; accordingly, we scope experiments to a representative LLM and dataset split to foreground clarity and reproducibility.

Our contributions are therefore threefold: First, we design a versatile and modular integration framework that utilizes a frozen-weight LLM as a computationally efficient feature enhancer. Second, to demonstrate its effectiveness and versatility, we apply the framework to a challenging rock drill fault diagnosis task characterized by domain shift, and show that it can be flexibly integrated with the established Domain-Adversarial Neural Network (DANN) adaptation technique. Third, through extensive experiments, we demonstrate that our method not only surpasses baseline models but also the DANN-applied model, with a particularly significant improvement for fault classes that were conventionally difficult to distinguish. This result substantiates that the enhanced features provided by the LLM are effective and robust even under domain shift conditions.

The remainder of this paper is organized as follows. Section 2 provides an overview of related work. Section 3 details the proposed LLM-integrated framework. Section 4 describes the experimental setup for validating our method, and Section 5 presents and discusses the experimental results. Finally, Section 6 concludes the paper and outlines future work.

2. RELATED WORK

Recent advances in Transformer-based time-series modeling, including Time Series Transformer (TST), Informer, PatchTST, and TS-BERT, have reported strong performance on temporal representation learning and PHM-related tasks. While a systematic head-to-head benchmarking against these models is beyond the scope of this paper, our goal here is to isolate and validate the contribution of the frozen, text-based LLM as a plug-in feature enhancer within a similarity-based PHM pipeline. Extending the evaluation to include such Transformer-based time-series baselines is a natural direction for future work and is facilitated by our model-agnostic interface.

This research is situated at the intersection of two major research fields: deep learning-based PHM, and the emerging frontier of applying LLMs to time-series analysis. In this section, we review prior work in these areas to clarify the academic positioning and contribution of our study.

First, deep learning has been firmly established as a powerful tool in PHM, particularly for fault diagnosis using time-series sensor data. While signal processing and classical machine learning methods were traditionally dominant, in recent years, CNNs and Recurrent Neural Networks (RNNs) have been widely adopted due to their ability to automatically learn hierarchical features directly from data (Chang & Han, 2024; Zhang et al., 2019; Zhao et al., 2019). Among these, specialized architectures like the Depth-wise CNN are noted for their efficiency in processing multi-channel sensor data. However, the efficacy of these purely data-driven approaches is often constrained in real-world settings by the scarcity of comprehensive fault data and the difficulty of formulating accurate physics-based models for complex systems. This lack of data and explicit domain knowledge makes it challenging for models to learn highly discriminative features, especially for faults with similar signatures. Furthermore, this challenge is often compounded by "domain shift" from unit-to-unit variations, which also degrades generalization. While domain adaptation techniques like DANN have been developed to address the latter issue, the fundamental challenge of enriching feature representations from limited data remains a key research objective.

Recent years have seen two parallel lines of progress: (i) general-purpose text-based LLMs, which excel in natural language understanding and reasoning; and (ii) pre-trained foundation models for time series, such as TimesNet or MOMENT. While our framework primarily exploits the first category (text-based LLMs) for knowledge transfer and reasoning support, we view time-series foundation models as complementary and discuss their integration as future work (Nie et al.,2023; Gruver et al., 2023).

In contrast to this prior work, our study proposes a different paradigm for leveraging LLMs. The novelty of our approach lies not in fine-tuning the LLM as a direct predictor, but in using it as a computationally efficient frozen feature enhancer. Furthermore, we have designed a modular and versatile framework that connects an arbitrary time-series feature extractor to the frozen LLM via a trainable adapter layer. This high degree of versatility allows our framework not only to be compatible with various time-series models but also to be flexibly combined with existing techniques like DANN to address specific problems, such as the aforementioned domain shift. In this paper, we demonstrate that this LLM-integrated framework can provide a robust and high-precision solution for challenging PHM tasks.

3. PROPOSED FRAMEWORK

In this section, we propose a versatile framework, which is the core of our research, for integrating pre-trained LLMs with time-series analysis models.

3.1. Overview and Motivation

Conventional time-series analysis models learn patterns solely from numerical data, such as sensor readings. In contrast, LLMs, trained on vast amounts of text data, possess sophisticated capabilities for conceptual understanding and contextual reasoning. Our research is founded on the hypothesis that by leveraging this abstract "knowledge" from LLMs in time-series analysis, we can acquire more advanced and robust feature representations that are unattainable from numerical data alone. The proposed framework aims to connect an arbitrary time-series model with an LLM, enabling the latter to function as a powerful feature enhancer.

3.2. Framework Architecture

The overall architecture of the proposed framework is illustrated in Figure 1. It is composed of four functionally distinct primary modules: (1) a Time-Series Feature Extractor, (2) Adapter layers, (3) a Frozen Pre-trained LLM, and (4) a Task-Specific Head.

First, the Time-Series Feature Extractor is responsible for extracting an initial feature vector from the raw input time-series data for use in subsequent tasks. The primary characteristic of this module is its flexibility; depending on the nature of the task and data characteristics, an optimal model can be selected from a variety of time-series architectures, such as CNNs or Transformers.

Next, the Adapter layers act as a bridge to fill the domain gap between the numerical feature representations generated by the time-series feature extractor and the high-dimensional embedding space that LLMs operate on. Specifically, it is tasked with projecting the feature vector into an embedding space suitable for processing by the LLM. Potential implementations include linear transformations using a fully connected layer or techniques like patching, which divides the data into local structures.

The Frozen Pre-trained LLM forms the core of the framework. It refines the features received from the adapter layer into more sophisticated, context-aware representations by leveraging its extensive pre-trained knowledge. A key design principle is that the LLM's parameters are kept frozen and are not updated during training. This approach avoids the computationally expensive process of fine-tuning and allows the LLM to be used as a lightweight yet powerful feature enhancer.

Finally, the Task-Specific Head takes the features enhanced by the LLM as input and performs the final prediction to achieve the task objective, such as classification or regression. This module is also highly flexible, and its architecture can be designed according to the specific problem to be solved. Common implementations include a Multi-Layer Perceptron (MLP) or Support Vector Machine (SVM).

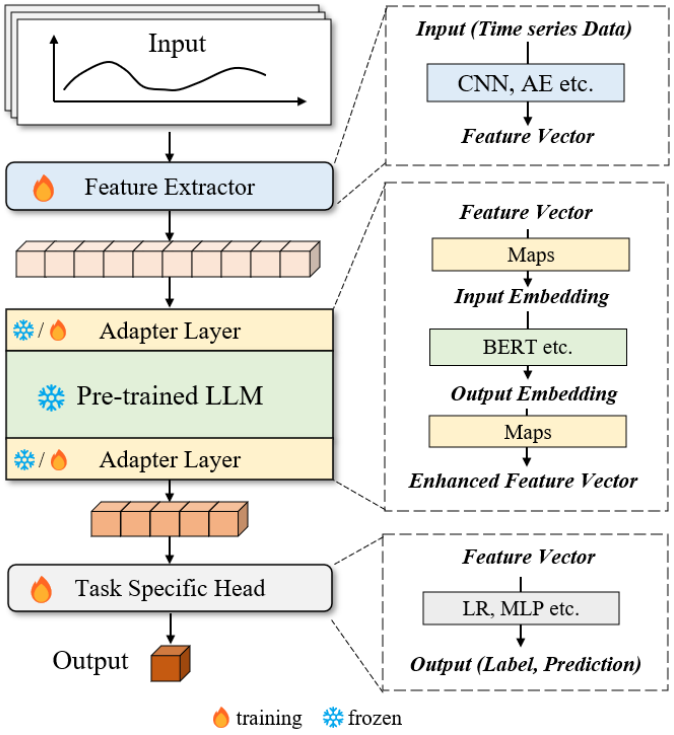


Figure 1. Conceptual Diagram of the Proposed Framework. This figure illustrates the overall architecture, comprising four main modules: a Time-Series Feature Extractor, an Adapter, a Frozen Pre-trained LLM, and a Task-Specific Head, designed for modular and flexible integration.

By adopting this modular design, where each component has a clear and independent role, the proposed framework is expected to demonstrate high adaptability and effectiveness for a wide range of time-series analysis problems without being overly dependent on a specific time-series model or a limited set of tasks.

4. EMPIRICAL STUDY: ROCK DRILL FAULT DIAGNOSIS

To validate the effectiveness of the LLM-integrated framework proposed in the previous section, this chapter presents a concrete case study on a rock drill fault diagnosis task. This study aims to evaluate the performance and robustness of the proposed method, particularly under conditions where domain shift, caused by unit-to-unit variations in equipment, exists

4.1. Dataset and Preprocessing

This study utilizes the rock drill dataset released for a PHM Data Challenge (Jakobsson et al., 2022; PHM Society, 2022). The dataset is composed of time-series data from eight distinct individuals, each exhibiting different operational characteristics, across 11 fault modes. Each data sample consists of 3-channel pressure sensor signals and exhibits

heterogeneity with varying sequence lengths. Because the individuals included in the training and evaluation data are intentionally separated, this task is defined as a domain shift problem arising from unit-to-unit variations.

Table 1 shows the data partitioning method and statistics for each individual used in this study. Although the original challenge setting used IDs 3 for validation and IDs 7–8 for testing, in this study we use only seven individuals (IDs 1–7). We exclude ID 8 to avoid having two distinct test domains, which would complicate interpretation, and to keep the evaluation protocol simple and focused on demonstrating the proposed methodology. To assess the final generalization performance on an unseen individual, Individual 7 is completely held out as the test set. Individual 3 is used as the validation set for hyperparameter tuning and early stopping. The remaining five individuals (1, 2, 4, 5, and 6) are used for model training.

As a preprocessing step for the input data, zero-padding is applied to handle samples with varying sequence lengths uniformly. All time-series samples were standardized to a length of 748, which is the maximum sequence length observed in the entire dataset.

Table 1. Dataset Overview and Split

Note: ID 8, which was part of the original challenge split, is excluded here to simplify evaluation and interpretation.

Individual	Data Use	No. of Sample	Sample Length	
			Min.	Max.
1	Train	7,331	617	748
2	Train	7,887	603	729
3	Validation	7,887	594	715
4	Train	7,617	585	710
5	Train	7,997	579	705
6	Train	3,313	571	692
7	Test	7,955	560	681

4.2. Compared and Proposed Models

To comprehensively evaluate the effectiveness of our proposed method, we designed and implemented three types of models: a baseline model, a domain adaptation model, and our proposed LLM-integrated model.

4.2.1. Compared Methods

Depth-wise CNN (Baseline): We adopt a Depth-wise CNN, which has reported high performance in prior work on the PHM Data Challenge, as the baseline model (Oh et al., 2023). This model is characterized by applying independent convolutional layers to each sensor channel upstream of the feature extractor, a technique known as depth-wise separable convolution (Chollet, 2017).

CNN+DANN: This model integrates the DANN, a domain adaptation technique, with the aforementioned CNN model (Ganin et al., 2016). It is included as a comparative model given that this task involves a domain shift problem between individuals. In this method, the feature extractor is trained adversarially with a domain classifier to learn domain-invariant feature representations.

4.2.2. Proposed Method (CNN+DANN+LLM)

Based on the versatile framework described in Section 3, we constructed the architecture shown in Figure 2. The details of each component in the proposed model are shown in Table 2.

The architecture of the proposed model is constructed by integrating multiple modules. First, for the time-series feature extractor, we employ a Depth-wise CNN, similar to the baseline, due to its ability to efficiently process 3-channel sensor signals. Next, to further enhance the features extracted by the CNN, we selected the pre-trained DistilBERT (distilbert-base-uncased) for the LLM component of our framework, considering its lightweight nature and affinity for classification tasks (Sanh et al., 2019). To leverage its extensive knowledge while curbing computational costs, the weights of DistilBERT are frozen during training, allowing it to function as a feature enhancer.

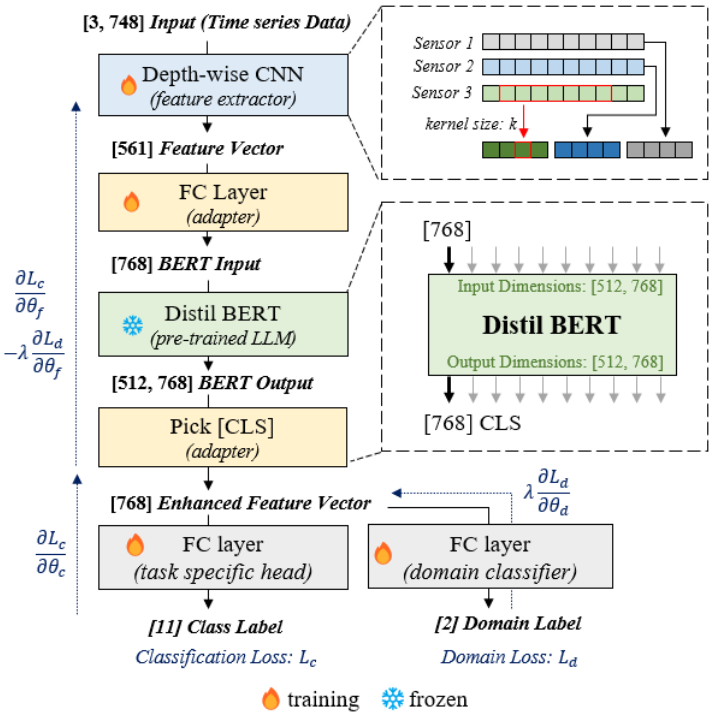


Figure 2. Proposed LLM integrated Model for Rockdrill Fault Diagnosis

Table 2. Details of Proposed Model Components

Component	Detail	Dimensions
Feature Extractor	Applies individual CNN branches to each sensor (3ch). Outputs from the 3 branches are concatenated and flattened.	in: [3, 748] out: [561]
Adapter (in)	A fully-connected layer that transforms CNN features to the LLM's embedding dimension.	in: [561] out: [768]
LLM Embedding	Uses a pre-trained DistilBERT (distilbert-base-uncased). Weights are frozen.	in: [1, 768] out: [512, 768]
Adapter (out)	Extracts the embedding vector of the [CLS] token.	in: [512, 768] out: [768]
Label Predictor	Fault class classifier (MLP).	in: [768] out: [11]
Domain Classifier	Domain classifier (MLP) via a Gradient Reversal Layer (GRL).	in: [768] out: [2]

To connect these modules, the 561-dimensional feature vector output by the Depth-wise CNN is transformed by a trainable adapter layer (a fully-connected layer) into the 768-dimensional standard input for DistilBERT. Receiving this vector as input, DistilBERT outputs a context-aware embedding vector. In this study, we utilize the 768-dimensional vector corresponding to the [CLS] token from this output, which holds aggregated sequence-level information, as the LLM-enhanced feature for subsequent tasks. Finally, to address the domain shift problem arising from unit-to-unit variations, we apply the DANN technique to these enhanced features, promoting the learning of domain-invariant representations.

The final training objective function is defined using the classification loss L_c and the domain loss L_d as follows, where the hyperparameter λ balances the two losses:

$$L_{Total} = L_c - \lambda * L_d \quad (1)$$

4.3. Experimental Setup

4.3.1. Evaluation Protocol and Metrics

To rigorously evaluate the model's generalization performance, we conduct five independent training-testing runs, each using one of the five individuals (IDs 1, 2, 4, 5, 6) as the training source. In all runs, ID 3 is fixed as the validation set, which is used for early stopping, and ID 7 is consistently held out as the unseen test set, on which accuracy and macro F1-score are reported. To account for variations due to stochastic elements (e.g., weight initialization, data shuffling), we conduct three independent trials for each run, changing only the random seed. The final scores are reported as the mean and standard deviation (SD) over all 15 runs (5 training IDs × 3 trials). Additionally, a Confusion Matrix is used for a detailed class-by-class performance analysis.

4.3.2. Implementation Details and Hyperparameters

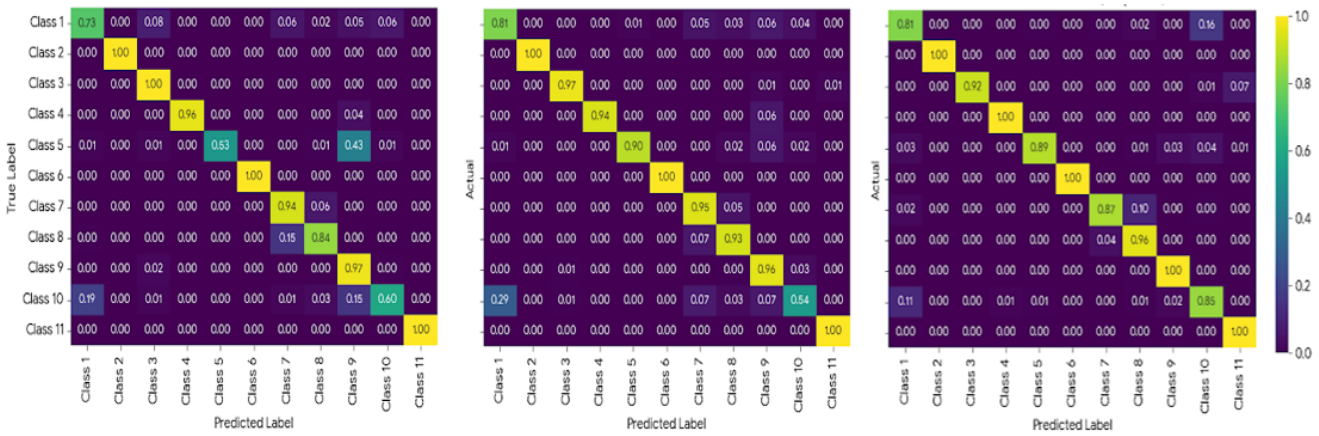
The experiments were conducted on a machine equipped with an NVIDIA GeForce RTX 3090 GPU, with an average training time of approximately 6.5 minutes per run. Key hyperparameters are shown in Table 3

Table 3. Key Hyperparameters

Hyper Parameters	Value
Learning Rate	0.0005
Batch Size	32
Max. No. of Epochs	100
Patience (Early Stopping)	30
Optimizer	Adam
Class Prediction Loss	Cross-entropy
Domain Prediction Loss	Cross-entropy
DANN Weight: λ	5

Architectural specifics: The depth-wise CNN uses three 1-D convolutional blocks per channel with kernel sizes [7, 5, 3], strides [1, 1, 1], and output channels [64, 64, 64], followed by BatchNorm + ReLU after each block and a final global average pooling over time. The three channel features are concatenated to form a 561-dimensional vector that feeds the adapter. The adapter (in) is a fully connected layer 561→768 with ReLU and dropout p=0.1. The LLM component is DistilBERT (distilbert-base-uncased) with all weights frozen; we do not tokenize the time series—the 768-D adapter output is provided as a single pseudo-token (one 768-D token injected at the embedding layer), and we take the [CLS] representation as the LLM-enhanced feature. The adapter (out) keeps the dimensionality at 768 for the downstream heads (fault label predictor and domain classifier). Random seeds for the three trials are {1, 2, 3} to vary initialization and shuffling.

Training procedure: Each mini-batch from a training individual is passed through the CNN–adapter–LLM stack. The resulting representation is used by two heads: the fault classifier (cross-entropy loss) and the domain classifier (cross-entropy loss via a GRL, λ=5). The combined loss (Eq. (1)) is used to update the CNN, adapter, and classifiers, while the LLM remains frozen. Early stopping is based on validation performance (ID 3, patience = 30), and final test results are reported on ID 7. Each configuration is repeated with three random seeds, and mean ± SD over 15 runs (five training-ID runs × three seeds) are reported.



(a) CNN (b) CNN+DANN (c) CNN+DANN+LLM

Figure 3. Comparison of Normalized Confusion Matrices

5. RESULTS AND DISCUSSION

5.1. Overall Performance Evaluation

Table 4 presents the overall performance comparison of the three models evaluated in this study, using two metrics: Accuracy and Macro F1-Score. Each score represents the mean and SD from a total of 15 runs, derived from three independent trials across five runs with different training individuals (IDs 1, 2, 4, 5, 6). In every run, ID 3 serves as the validation set and ID 7 as the final test set. Thus, Table 4 summarizes the test performance obtained under these five different training configurations.

As shown in Table 4, the baseline CNN model recorded an Accuracy of 0.778 and a Macro F1-Score of 0.751. In comparison, the CNN + DANN model, which applies the domain adaptation technique DANN, improved the Accuracy to 0.792 and the Macro F1-Score to 0.772. This result suggests that DANN contributed to mitigating the domain shift between individuals and improving generalization performance

Our proposed LLM-integrated model (CNN+DANN+LLM) further surpassed the performance of the DANN-applied model, achieving the highest performance among all compared methods with an Accuracy of 0.811 and a Macro F1-Score of 0.793. Furthermore, the SD of the proposed method was comparable to the other models, and its Macro F1-Score SD was the lowest, indicating that the integration of the LLM may contribute not only to performance improvement but also to the stabilization of training. Beyond LLMs, recent time-series Transformers such as TST, Informer, PatchTST, and TS-BERT represent strong alternative baselines. While a full comparison is beyond this paper's scope, our modular design allows straightforward substitution or integration with such models in future work.

It is important to note that the proposed framework introduces an adapter layer to connect CNN features with the LLM. To ensure a fair comparison, we retained the adapter even in the CNN+DANN baseline when the LLM component was removed (by applying a skip connection over the LLM). This design guarantees that the number of trainable parameters remains essentially identical across models, isolating the contribution of the LLM itself. The consistent improvement observed, particularly for fault classes that are conventionally difficult to distinguish, therefore cannot be attributed merely to parameter growth but instead highlights the added value of LLM-projected representations.

Table 4. Performance Comparison

Model	Accuracy		F1 score	
	Mean	± SD	Mean	± SD
CNN	0.778	±0.131	0.751	±0.165
CNN+DANN	0.792	±0.130	0.772	±0.162
CNN+DANN+LLM	**0.811**	**±0.120**	**0.793**	**±0.147**

5.2. Analysis of Class-wise Classification Performance

Next, to analyze the factors contributing to the overall performance improvement in detail, we visualize how each model classified the 11 fault classes using normalized confusion matrices for the test data (Figure 3). The diagonal elements of the matrices correspond to the recall for each class, where a higher value signifies better classification performance for that class.

Figure 3 clearly illustrates the improvement in classification performance as the model is enhanced.

(a) CNN (Baseline): In the baseline model, values are dispersed outside the diagonal, indicating that

misclassifications occurred between multiple classes. In particular, there was a noticeable tendency for 'Class 5' to be misclassified as 'Class 9', 'Class 8' as 'Class 7', and 'Class 10' as both 'Class 1' and 'Class 9'.

(b) CNN+DANN: In the central model with DANN, the misclassification from 'Class 5' to 'Class 9', which was prominent in the baseline, is significantly improved. The misclassifications of 'Class 8' as 'Class 7' and 'Class 10' as 'Class 9' are also reduced. This aligns with the overall performance improvement confirmed in Section 5.1 and is likely a result of DANN acquiring domain-invariant features. However, confusion in classification, such as 'Class 10' being misclassified as 'Class 1', is still observed.

(c) CNN+DANN+ LLM (Proposed): In the proposed method on the right, the diagonal values are the highest overall, and misclassifications are substantially suppressed. What is particularly noteworthy is the significant improvement in the classification accuracy for 'Class 10', which was difficult for the other models to distinguish. For this class, by integrating the LLM, a mapping to a higher-order, more abstract feature space was performed. We infer that this endowed the model with the ability to capture subtle feature differences, much like how a human would make judgments from context. This analysis corroborates the quantitative performance evaluation presented in Section 5.1, demonstrating that the integration of LLM knowledge helps solve problems that were particularly challenging for conventional methods and contributes to the realization of a more robust and high-precision classifier.

6. CONCLUSION AND FUTURE WORK

6.1. Conclusion

This research proposed a novel, versatile, and modular framework for integrating the knowledge of LLMs to enhance the performance of time-series analysis in PHM. The framework utilizes an LLM as a computationally efficient feature enhancer by connecting an arbitrary time-series feature extractor to a frozen-weight LLM via an adapter layer. We demonstrated its effectiveness on a rock drill fault diagnosis task that includes a domain shift problem arising from unit-to-unit variations. In the experiments, a model integrating DANN was constructed to improve robustness against this domain shift.

Experimental results showed that the proposed LLM-integrated model (CNN+DANN+LLM) achieved the highest performance on both Accuracy and Macro F1-Score metrics compared to the baseline CNN and the DANN-applied models. A detailed analysis of the confusion matrices confirmed a significant improvement in the classification accuracy of specific fault classes that were difficult for conventional methods to distinguish, most notably Class 10. This result suggests that projecting the extensive pre-trained knowledge of an LLM into the feature space contributes to the identification of complex and subtle patterns that cannot be captured by sensor data alone.

6.2. Limitations and Future Work

While this research opens up many possibilities for future development, it also has several limitations.

First, the validation in this paper is limited to a single dataset. Therefore, it is necessary to apply this framework to diverse PHM datasets to further evaluate its versatility.

Second, there is room for architectural improvements to fully leverage the LLM's capabilities. In this study, we input the time-series features as a single vector to the LLM, an approach that may not fully utilize the self-attention mechanism's ability to capture relationships within a sequence. A promising future approach involves dividing the time-series features into multiple patches and inputting them to the LLM as a sequence with corresponding positional encodings. This would allow the self-attention mechanism to model temporal and contextual dependencies between features, which is expected to yield richer feature representations.

Third, although we completely froze the LLM's weights to curb computational costs, introducing limited fine-tuning is another promising option. Depending on the dataset size and task characteristics, unfreezing only the final few layers of the LLM or adapting its output layer to serve as the task-specific head could further specialize its representational power for the task, potentially leading to additional performance gains.

Finally, using larger models than the DistilBERT used in this study, or LLMs pre-trained on specific industrial domains, could also contribute to future performance improvements. However, the aforementioned architectural enhancements, the introduction of fine-tuning, and the adoption of larger models all present a trade-off with increased computational cost. Consequently, designing efficient integration methods and adapter layers that balance performance and cost remains a critical research challenge for practical application.

Beyond these points, we see two practical extensions and one evaluation plan. First, incorporating domain-specific LLMs trained on industrial corpora may further improve generalization, especially for fault classes that are semantically or operationally similar. Second, while this study addresses single-label classification, the architecture readily extends to multi-label settings by replacing the softmax head with independent sigmoid outputs; the upstream CNN–adapter–LLM stack remains unchanged. To more fully situate our method within the broader state of the art, we also plan to benchmark against Transformer-based time-series baselines (e.g., TST, Informer, PatchTST, TS-BERT) and domain-specific LLMs; our model-agnostic interface makes such substitutions straightforward.

We hope this study serves as a solid step toward leveraging the potential of LLMs in time-series analysis, particularly in industrial applications.

REFERENCES

Chang, Z., & Han, T. (2024). Prognostics and health management of photovoltaic systems based on deep learning: A state-of-the-art review and future perspectives. *Renewable and Sustainable Energy Reviews*, 205, 114861.

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. *In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., ... & Lempitsky, V. (2016). Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59), 1-35.

Gruver, N., Finzi, M., Qiu, S., & Wilson, A. G. (2023). Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 19622-19635.

Jakobsson, E., Frisk, E., Krysander, M., & Pettersson, R. (2022, October). A dataset for fault classification in rock drills, a fast-oscillating hydraulic system. *In Annual conference of the phm society* (Vol. 14, No. 1).

Nie, Y., Nguyen, N. H., Sinthong, P., & Kalagnanam, J. (2022). A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint* arXiv:2211.14730.

Oh, H. J., Yoo, J., Lee, S., Chae, M., Park, J., & Youn, B. D. (2023). A hybrid approach combining data-driven and signal-processing-based methods for fault diagnosis of a hydraulic rock drill. *International Journal of Prognostics and Health Management*, 14(1).

PHM Society. (2022). *2022 PHM Conference Data Challenge*. Retrieved June 19, 2025, from https://data.phmsociety.org/2022-phm-conference-data-challenge/

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint* arXiv:1910.01108.

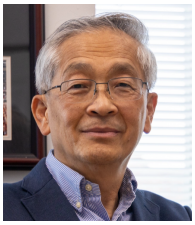
Zhang, L., Lin, J., Liu, B., Zhang, Z., Yan, X., & Wei, M. (2019). A review on deep learning applications in prognostics and health management. *IEEE Access*, 7, 162415-162438.

Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., & Gao, R. X. (2019). Deep learning and its applications to machine health monitoring. *Mechanical Systems and Signal Processing*, 115, 213-237.

BIOGRAPHIES

Takanobu Minami received his B.S. and M.S. degrees in mechanical engineering from Kyoto University in 2008 and in 2011, respectively. Currently, he is pursuing his Ph.D. degree in mechanical engineering with the University of Maryland, College Park, MD, USA, and is employed as an engineer in Komatsu Ltd. His research interests include machine learning, deep learning, prognostics and health management, and industrial AI.

Dai-Yan Ji received his B.S. degree in Electronic Engineering and M.S. degree in Communications Engineering from Feng Chia University, Taiwan, in 2009 and 2012, respectively. He is currently pursuing a Ph.D. degree in Mechanical Engineering with the University of Maryland, College Park, MD, USA. His research interests include machine learning, signal processing, industrial artificial intelligence, and prognostics, and health management.



Jay Lee is Clark Distinguished Professor and Director of the Industrial AI Center in the Mechanical Engineering Dept. of the Univ. of Maryland College Park. His research is focused on intelligent analytics of complex systems including highly-connected industrial systems including energy, manufacturing, healthcare/medical, etc. He has been working with medical school in Traumatic Brain Injury (TBI) using multi-dimension data for predictive assessment of patient in ICU with funding from NIH and NSF. Previously, he served as an Ohio Eminent Scholar, L.W. Scott Alter Chair and Univ. Distinguished Professor at Univ. of Cincinnati. He was Founding Director of National Science Foundation (NSF) Industry/University Cooperative Research Center (I/UCRC) on Intelligent Maintenance Systems during 2001-2019. IMS Center pioneered industrial Ai-augmented prognostics technologies for highly-connected industrial systems and has developed research memberships with over 100 global company since 2000 and was selected as the most economically impactful I/UCRC in the NSF Economic Impact Study Report in 2012. He is also the Founding Director of Industrial AI Center. He mentored his students and developed a number of start-up companies including Predictronics through NSF iCorp in 2013 and has won 1st Place for PHM Society Data Challenges competition 5 times He was on leave from UC to serve as Vice Chairman and Board Member for Foxconn Technology Group (ranked 26th in Global Fortune 500) during 2019-2021 to lead the development of Foxconn Wisconsin Science Park (~$1B investment) in Mt. Pleasant, WI. In addition, he advised Foxconn business units to successfully receive five WEF Lighthouse Factory Awards since 2019. He is a member of

Global Future Council on Advanced Manufacturing and Production of the World Economics Council (WEF), a member of Board of Governors of the Manufacturing Executive Leadership Council of National Association of Manufacturers (NAM), Board of Trustees of MTConnect, as well as a senior advisor to McKinsey. Previously, he served as Director for Product Development and Manufacturing at United Technologies Research Center (now Raytheon Technologies Research Center) as well as Program Director for a number of programs at NSF. He was selected as 30 Visionaries in Smart Manufacturing in by SME in Jan. 2016 and 20 most influential professors in Smart Manufacturing in June 2020, SME Eli Whitney Productivity Award and SME/NAMRC S.M. Wu Research Implementation Award in 2022