Evaluation of Multi-Modal Learning for Predicting Coolant Pump Failures in Heavy Duty Vehicles

Yuantao Fan, M. Amine Atoui, Sławomir Nowaczyk, and Thorsteinn Rögnvaldsson

Center for Applied Intelligent Systems Research (CAISR), Halmstad University, Sweden
{*yuantao.fan, amine.atoui, slawomir.nowaczyk, thorsteinn.rognvaldsson*}*@hh.se*

ABSTRACT

Coolant Pump failures in heavy-duty vehicles can cause severe collateral damage if they are not detected and resolved in time; the engine will overheat quickly, rendering the vehicle inoperable. Nowadays, a vast amount of heterogeneous sensor data from different sources is being collected in the automotive industry. Such multi-modal data include onboard signals reflecting the overall usage of the vehicle, multi-dimensional histograms that capture the relation between physical quantities, and categorical variables that encode the physical configuration of the vehicle. This work evaluates several multi-modal learning approaches leveraging this diverse data to build a prognosis and health management system for coolant pumps in commercial heavy-duty vehicles. Four auto-encoder architectures are examined to extract features from 2D histograms. These trained models are anticipated to capture key characteristics of the healthy system operation and yield large reconstruction errors when applied on faulty, or near end-of-life samples. Such learned representations are then combined with expert-engineered features. Both early and intermediate fusion are evaluated on a real-world coolant pump replacement dataset. Results indicate that the combination of diverse features was the most effective approach, thereby motivating further research on multi-modal methods.

1. INTRODUCTION

Unplanned downtime due to component failure is very costly for the operation of commercial heavy-duty vehicles. The development of prognosis and health management systems focuses on predicting the remaining useful life or upcoming failure of the equipment and could help improve the current reactive and preventive maintenance paradigm to a predictive one, thus enhancing operational efficiency and reducing downtime. With the advancement of electronic computing units (ECUs), communication, and sensor technologies, a huge amount of heterogeneous data is being collected from the vehicles and utilized as the input that fuels data-driven prognostic methods.

Features being collected from onboard ECUs include scalar variables that reflect the overall usage of the vehicle (e.g. mileage and fuel consumption), snap-shots of single- and multi-dimensional histograms that capture the density function of the signals, and categorical variables of the physical configuration of the vehicle. The presence of data from diverse modalities sparks an interest in investigating methods for multi-modal learning, aiming to combine these features and enhance overall performance.

In the field of machine learning, it is a common practice to combine available features for training machine learning models, regardless of their modalities. This practice typically involves incorporating feature selection or feature learning methods into the overall process. However, it is crucial to acknowledge that this approach carries the risk of losing information or not fully exploiting all the available data. Our study focuses on evaluating and comparing different ways that combine multi-modal features for developing prognosis models. These approaches are based on self-supervised learning models, such as auto-encoders. Several multi-modal learning methods using intermediate fusion against several early fusion approaches, as well as methods without feature learning, i.e., using raw sensor readings or expert-engineered features as input, are compared. Two experiments were conducted to investigate the different approaches, and the evaluation focused on two specific tasks: failures and remaining useful life (RUL) prediction of the coolant pump.

The first experiment was conducted on simulated data. Features learned and extracted by self-supervised learning with auto-encoders are found capable of detecting faults. Therefore, the reconstruction error, i.e. the residual, of the testing samples can serve as indicators for anomalies and therefore shall be included for RUL prediction as well.

The second experiment focused on real-world data and ex-

amined the performance of the multi-modal learning-based intermediate-fusion approach. This approach involves learning representations separately for each modality and at different stages. Our findings indicate that this approach outperforms early fusion and two other conventional data-driven approaches in predicting upcoming failure and predicting the RUL for the coolant pump. The best results for both tasks were achieved by combining the features extracted by the convolutional auto-encoders with the expert-suggested features.

The remainder of the paper is organized as follows: a literature review is described in 2. Section 3 explains the problem under investigation. Section 4 introduces the fundamental principles behind the evaluated multi-modal learning approaches. The outcomes of the experiments are discussed in Section 5. Finally, Section 6 concludes the paper and discusses future research directions.

2. LITERATURE REVIEW

The fusion of multi-modal data can be performed in different stages, namely early, intermediate and late fusion (Ramachandram & Taylor, 2017). Multi-modal representation learning frameworks play an important role in early and intermediate multi-modal fusion. These frameworks can generate various types of representations. The most popular ones are the joint, coordinated, and encoder-decoder model-based representation (Guo, Wang, & Wang, 2019). Learning and generating joint representation with Restricted Boltzmann Machines (RBMs), and deep Boltzmann Machines (DBMs) for sensor data is quite popular. Audio and visual information is fused using Convolutional Neural Network for manipulator failure detection in (Inceoglu et al., 2021). A study (Pang, Zhu, & Ngo, 2015) uses joint representation generated by Deep Boltzmann machine (DBM) based on user-generated visual, auditory, and textual data for affective analysis and retrieval.

Moreover, in prognosis and predictive maintenance applications, most of the methods create joint representations from multi-modal data. The work in (Huang et al., 2023) creates a joint representation that fuses two different modalities with two DBMs and uses a feed-forward network to predict failures, on turbofan engines. (H. Li et al., 2021) combine CNN and denoising auto-encoders for fusing different modalities using the joint representation for diagnosis purposes in a gear device. An approach in (Chen et al., 2023) fuses denoised sensor data and simulated data using RBMs and DBMs for RUL prediction in an ensembled fashion. A hybrid deep neural network is used in (Al-Dulaimi et al., 2019), concatenating features learned from CNNs and LSTMs to predict RUL for turbofan engines. In (C. Li et al., 2015) Gaussian-Bernoulli deep Boltzmann machine (GDBM) is used to fuse multi-modal homologous features, e.g., time, frequency, wavelet features, generated from vibration measurements, for gearbox state diagnosis.

However, very few works explore the use of 2D histograms of sensor readings, which estimate the density function of two numerical variables, and expert knowledge in a multi-modal learning setting for predicting failures and predicting remaining useful life.

3. PROBLEM STATEMENT

The coolant pump repair dataset contains 127 cases of coolant pump replacement in heavy-duty vehicles. Due to the high cost associated with unplanned road stops, the OEM has implemented a preventive maintenance strategy. Thus, representing the journey until the replacement of the coolant pump is stored. Sensor readings, K multivariate and multi-modal time series features $\mathbf{x}$, of two modalities are available: i) histogram parameter $\mathbf{x}^h, \mathbf{x}^h \in \mathrm{R}^{K_h}$, of coolant temperature versus oil temperature, which reflects the temperature condition of a truck, where K_h corresponds to the number of bins (or cells) of a 2D histogram; ii) a vector of K_s scalar variables $\mathbf{x}^s, \mathbf{x}^s \in \mathrm{R}^{K_s}$ reflecting the overall usage of the vehicle, i.e., total operating hours, mileage, fuel consumption, vehicle age, and time since last coolant system repair.

Let us denote data of the multivariate time series $\mathbf{x}$ of each trajectory/vehicle v by $X = \{\ x^i_{v,t} \mid t = 1, 2, ..., T(v),\ i = 1, 2, ..., K\}$, where $x^i_{v,t}$ is the value of the i^{th} feature $\mathbf{x}$ given a vehicle/trajectory v at time t, and $T(v)$ denote the end-of-life of trajectory v. Note that an observation of the histogram feature subset at time t is denoted with x^h_t, and one of the scalar variables is denoted with x^s_t. Each vehicle exhibits the coolant pump replacement at the end of its trajectory. In addition, consider a set of healthy and fault-free samples $X^H : \{x^i_{v,t} |\ t + \tau_\Delta \leq T(v)\}$ with RUL larger than $\tau_\Delta = 360$ days. Several other features are considered in the following based on suggestions from the domain expert, including mileage per operating hour; fuel consumed per kilometer traveled, and per operating hour. Since the coolant systems cannot be run to failure during normal operations, a preventive maintenance strategy is implemented.

3.1. Failure Prediction

Coolant systems are subject to diagnostic tests in the workshops, and those deemed inadequate are replaced. Thus, it is necessary to make the assumption that the time of coolant pump replacement signifies the termination of its operational life, as there is no subsequent data generated by the unit. However, it is important to note that there is no definitive confirmation of its failure.

Consider a classification model $f_c(\cdot)$ is trained to predict whether any failure will occur in the given time horizon using learned features θ, i.e., f_c: $\theta \rightarrow Y_c$, where Y_c denotes the

target. θ are obtained from a self-supervised model, e.g. an auto-encoder, $g(\cdot)$, trained only using the data that are fault-free, i.e. g: $x \rightarrow \theta$. A prediction horizon τ_{ph} of 90 days is selected. The binary classification target, given the time t and vehicle v, can be configured as follows:

$$y_{c_{v,t}} = \begin{cases} 1 & \text{if } T(v) - \tau_{ph} \leqslant t \leqslant T(v) \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

3.2. Remaining Useful Life Prediction

The RUL prediction provides time-to-failure information for each coolant pump unit. Practically, RUL can be capped at 180 days, indicating no imminent need for repair. In this paper, the following target sequence is used:

$$y_{r_{v,t}} = \begin{cases} T(v) - t & \text{if } T(v) - \tau_{max} \leqslant t \leqslant T(v) \\ \tau_{max} & \text{otherwise,} \end{cases} \quad (2)$$

where τ_{max} is set to 180. Regression models $f_r(\cdot)$ are trained to cast prediction on RUL, i.e. f_r: $\theta \rightarrow Y_r$, where Y_r denotes the RUL.

4. METHOD

An investigation and comparison of intermediate and early fusion approaches for prognosis are conducted. Auto-encoders with different architectures are trained with samples X^H that are considered fault-free, or far from a failure in time, i.e. having an RUL larger than 360 days.

The introduced approach, in our context, uses multi-modal learning via intermediate fusion for the coolant pump prognosis tasks. The joint representation $\Theta^{\{h,s\}}$ is the result of the combination of the scalar variables x with features $\{\delta, \theta\}$ learned using an auto-encoder $g^h(\cdot)$. The fused features are then used to train a classification model f_c: $\Theta \rightarrow Y_c$) and a RUL predictor model f_r: $\Theta \rightarrow Y_r$.

4.1. Self-Supervised Learning using Auto-encoders

Learning useful representations that can capture key characteristics of the underlying signal is crucial for carrying out Machine Learning tasks. Auto-encoders are used to learn latent features θ (with the reconstruction error δ) in a self-supervised learning setting.

It is expected that greater reconstruction errors δ are yielded while applying the trained auto-encoder $g(\cdot)$ on faulty samples. A near-end-of-life coolant pump exhibits different behavior from the population on which the auto-encoder was trained. The auto-encoder is trained to find latent features, i.e., a code θ in between the encoder $E_\varphi(\cdot)$ and the decoder $D_\phi(\cdot)$. It aims to minimize the reconstruction error, e.g., the mean squared error between the input and the reconstructed samples: $\frac{1}{N}\sum_{u=1}^{N} ||D_\phi(E_\varphi(x_{v,t}^k)) - x_{v,t}^k||_2^2$, where k is the set of selected features of $x_{v,t}$, and N is the total number of dimensions.

In this work, a comparative evaluation is conducted to assess the performance of four types of encoder-decoder architectures, i.e., fully connected networks (FC); fully connected networks with sparse constraints (FCS) on encoder representations; fully connected deep auto-encoders with more hidden layers (FCD), and convolutional auto-encoders (CNN).

4.2. Multi-Modal Fusion Framework

The multi-modal learning approach, essentially based on intermediate fusion, learns features at different stages and creates a joint representation $\Theta^{\{h,s\}}$ for the prognosis tasks. Figure. 1 provides a visual depiction of the intermediate fusion method, wherein data comprising multiple modalities, such as histograms and scalar variables, are utilized as input. This approach involves learning and extracting features in a modality-specific manner. Latent representation θ and reconstruction error δ are extracted for histograms ($\mathbf{x}^h$) using trained auto-encoders applied to healthy data. For scalar variables, a small set of engineered features $\mathbf{x}^a$ are derived from the raw scalar features $\mathbf{x}^s$ based on expert knowledge. The samples, given an instant t, of these extracted features $\{\delta, \theta, \mathbf{x}^a, \mathbf{x}^s\}$ were concatenated into a joint representation $\Theta^{\{h,s\}}$ as the input for training supervised machine learning model, i.e., f_c or f_r for the prognosis task.

Additionally, an alternative approach based on early fusion (depicted in Figure 2) has been considered. In this case, both modalities are concatenated, and a compressed representation is learned using an auto-encoder $g(\cdot)$ trained in a similar manner as described above. The extracted compressed representation $\{\delta, \theta\}$ is comprised of the reconstruction error and the latent features for performing the prognosis tasks.

4.3. Prognosis Modeling and Evaluation Metric

The prognosis modeling includes two tasks, i.e., training a classifier f_c for forecasting whether a failure will occur in a given time frame τ_{ph}, and a regressor f_r for predicting the remaining useful life. For both tasks, an auto-encoder $g(\cdot)$ was trained first with the healthy samples in the training population. Then, learned compressed representation $\{\delta, \theta\}$ was extracted for both training and testing samples. Prognostic models were constructed utilizing the acquired representations from the training population, enabling the prediction to be applied to the testing population for evaluation purposes.

The evaluation metric for the failure prediction classification task is the area under the curve (AUC). The metric for evaluating the RUL regression task is the mean absolute percentage error (MAPE) since it weights errors near end-of-life higher.

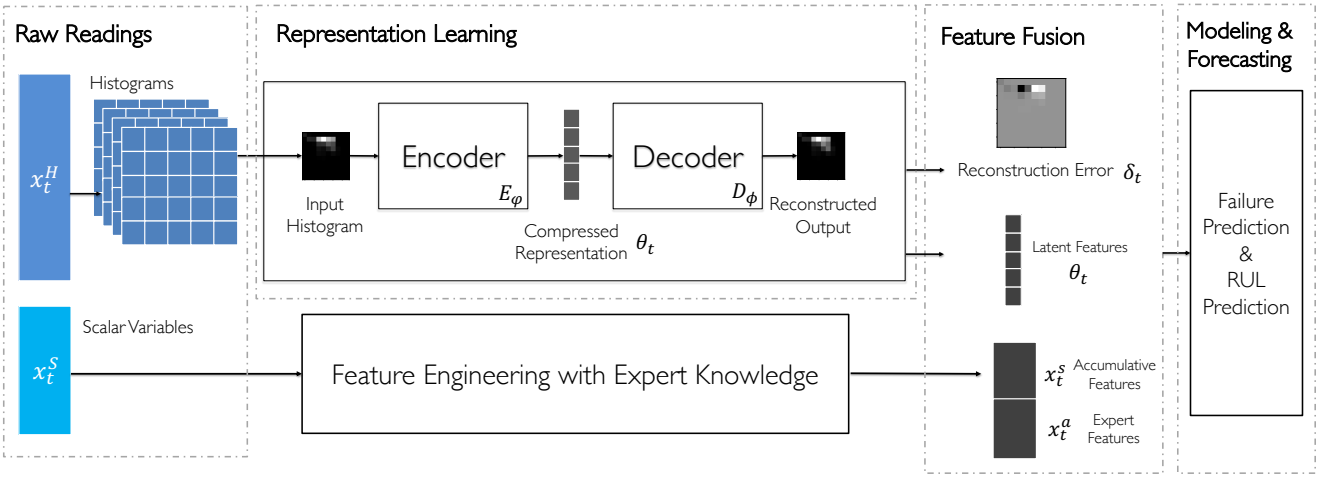

Figure 1. Multi-Modal Learning with Intermediate Fusion

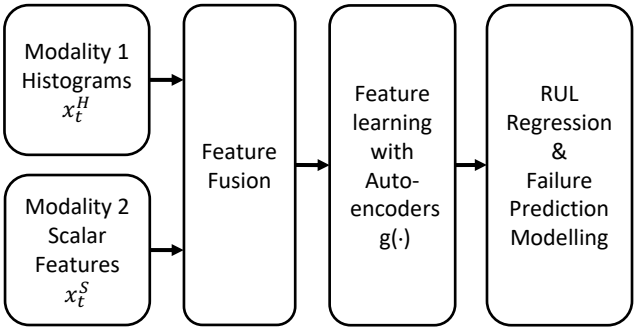

Figure 2. Multi-Modal Learning with Early Fusion

5. Experiment Results

5.1. Experimental Setting

Auto-encoder architectures: For the experiments conducted in this paper, four different types of auto-encoders are implemented: i) the auto-encoder with a fully connected network (FC) employs a single fully-connected neural layer for the encoder as well as for the decoder, obtaining K^2 numbers of ReLU units in the encoder and the same amount of sigmoid units in the decoder; ii) the auto-encoder with a sparsity constraint (FCS) has the same architecture w.r.t. FC, but employed an $L1$ regularization with a factor of 10^{-4}; iii) the deep auto-encoder with fully connected layers (FCD) has four dense layers in the encoder, in which (512, 256, 128, 64) ReLU units were employed in each hidden layer, and four dense layers in the decoder, in which (64, 128, 256) ReLU units in the first three layers and 512 sigmoid units in the last layer: iv) the convolutional auto-encoder (CNN) is comprised of a stack of 2D convolutional layers with ReLu units (except for the last layer in the decoder, sigmiod units are employed) and 2D max pooling layers.

The encoded latent features in all four types of auto-encoders were configured to have a dimensionality of 32. Training of the networks was conducted using an ADAM optimizer, with mean squared error serving as the loss function for FC, FCS, FCD, and CNN architectures.

Classification and Regression Models: The classification methods evaluated in this study include: i) random forest classifier (RF) with 100 estimators using Gini impurity; ii) ridge classifier with an alpha of 1; iii) K-NearestNeighbors (kNN) classifier with the number of neighbors equal 15; iv) linear discriminant analysis (LDA) with an SVD solver; v) multi-layer perceptron (MLP) classifier with two hidden layers, each containing 100 ReLU units, trained using an ADAM optimizer.

The regression methods evaluated in this study include: i) random forest regressor with 100 estimators; ii) ridge regression with L2 regularization and an alpha of 1; iii) K-NearestNeighbors Regressor with a k equal to 15; iv) Linear Regression; v) multi-layer perceptron regressor with two hidden layers, each containing 100 ReLU units, optimized with ADAM. For the classification and regression models, the scikit-learn library (Pedregosa et al., 2011) was employed. The experiments were conducted using 4-fold cross-validation vehicle-wise, i.e., data from the same vehicle would never appear in the training and the testing population together.

5.2. Detecting Simulated Faults

Regrettably, the ground truth pertaining to the condition of the coolant pump in the real-world dataset remains inaccessible. Solely the information regarding replacements is available, with an anticipated presence of inaccuracies within a subset of the replacements. There exists very little reliable information regarding the degree of wear in individual coolant pumps and the associated fault modes.

Hence, an experiment is conducted wherein faults are deliberately injected into the data to emulate the behavior of a compromised coolant pump. Intuitively, such a pump exhibits reduced cooling efficiency, consequently leading to an extended duration of elevated temperatures within the system.

The primary objective of the experiment involving injected faults is to verify and illustrate whether the trained auto-encoder and the reconstruction error δ can be used to successfully detect symptoms of faults injected in the 2D histogram of coolant versus oil temperature. More precisely, if the coolant pump is not working properly, the coolant and the oil temperature would be higher than usual. Hence, we have injected faults by redistributing the mass of the histogram towards the lower right corner, which corresponds to high coolant and oil temperatures.

A set of new histograms were randomly drawn from the histogram features in X^H. In Figure 3, a 2D histogram of oil versus coolant temperature without any fault injected is shown in the upper left corner. The rest are histograms with different degrees of faults injected. The degree of the fault injected is governed by a pair of meta parameters $(\Delta N, \Delta D)$, in which ΔN governs the ratio of masses that are redistributed towards the lower right corner of the histogram, while ΔD governs the degree of the diffusion of the redistributed mass. The histogram array illustrated in figure 3 shows an increasing ΔN towards the right side of the array and an increasing ΔD towards the bottom.

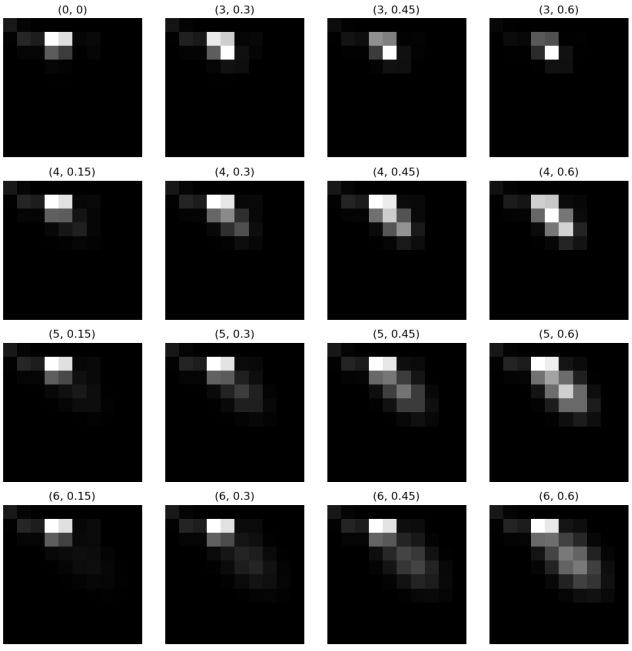

Figure 3. Simulated faults of different degrees in the 2D histogram (coolant versus oil temperature)

For the experiment, an auto-encoder $g^h(\cdot)$ was trained on 300 healthy histograms and was applied to a balanced testing set. Half of the latter set, histograms, were injected with faults of different degrees, yielding a reconstruction error δ for every testing histogram. These outputs were used in the binary classification task, i.e., distinguishing histograms with faults from healthy ones. The area under the receiver operating characteristic curve (AUC) was computed for a few experiments with increasing values of $(\Delta N, \Delta D)$. As is shown in Figure 4, the reconstruction error δ for all four auto-encoders with different settings is capable of detecting the fault injected. This validates the theoretical foundations and demonstrates that histograms exhibiting the patterns expected from weak coolant pumps provide useful information.

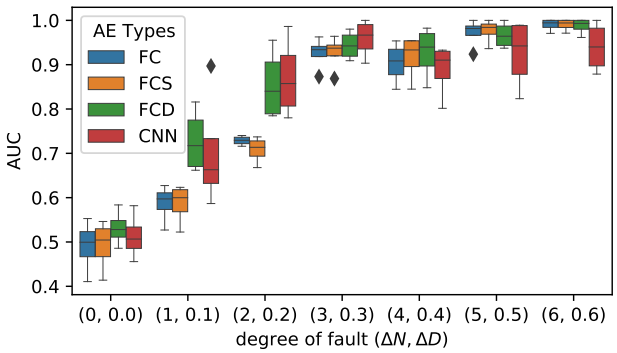

Figure 4. Area under the ROC curve (AUC) of using reconstruction error δ for detecting simulated faults in 2D histograms

5.3. Prognostic tasks on the coolant pump dataset

Table 1 shows performance evaluation, comparing different multi-modal fusion methods in predicting coolant pump failures. It is shown that classification methods with the top 50 features (from all features available $\{x^s, x^a, x^h\}$) selected by using ANOVA F-value perform on par with the expert-suggested features. The AUC of early fusion (EF) methods, i.e. auto-encoders with FC, FCS, FCD, and CNN, is worse compared to the rest of the evaluated methods. Multi-modal learning with intermediate fusion (IF) methods outperforms other approaches; MLP with features extracted from FC and CNN auto-encoder achieves the overall best performance.

Table 2 shows the comparison, in terms of MAPE, for RUL prediction. The best performance was, again, achieved with intermediate fusion using features learned from convolutional auto-encoders.

6. Conclusion and Future Work

In this paper, several muti-modal fusion approaches to predict failures and RUL of coolant pumps in commercial heavy-duty vehicles were introduced and evaluated The findings of this study highlight that the intermediate fusion of the learned features, comprising the latent features alongside the reconstruc-

Table 1. Performance Comparison (AUC) for predicting coolant pump failures in a time frame of 90 days

AUC	RF	Ridge	kNN	LDA	MLP
EF-FC	0.66 ± 0.03	0.6 ± 0.01	0.58 ± 0.02	0.61 ± 0.02	0.61 ± 0.02
EF-FCS	0.61 ± 0.03	0.58 ± 0.03	0.58 ± 0.02	0.58 ± 0.01	0.62 ± 0.05
EF-FCD	0.55 ± 0.02	0.55 ± 0.03	0.59 ± 0.06	0.59 ± 0.05	0.6 ± 0.03
EF-CNN	0.52 ± 0.03	0.52 ± 0.01	0.52 ± 0.01	0.54 ± 0.02	0.54 ± 0.03
IF-FC	0.66 ± 0.06	0.66 ± 0.04	0.63 ± 0.04	0.66 ± 0.05	**0.72 ± 0.04**
IF-FCS	**0.68 ± 0.08**	**0.67 ± 0.04**	0.66 ± 0.02	**0.68 ± 0.01**	0.71 ± 0.03
IF-FCD	0.62 ± 0.07	**0.67 ± 0.05**	**0.67 ± 0.05**	0.7 ± 0.04	0.69 ± 0.02
IF-CNN	0.63 ± 0.1	**0.67 ± 0.05**	0.58 ± 0.05	**0.68 ± 0.03**	**0.72 ± 0.02**
x (top 50)	0.64 ± 0.09	0.62 ± 0.06	0.63 ± 0.04	0.67 ± 0.07	0.68 ± 0.04
Expert x^a	0.64 ± 0.08	0.62 ± 0.06	0.63 ± 0.04	0.67 ± 0.07	0.68 ± 0.06

Table 2. Performance Comparison (MAPE) for predicting RUL for coolant pumps

MAPE	RF	Ridge	kNN	LDA	MLP
EF-FC	0.66 ± 0.02	1.07 ± 0.28	0.74 ± 0.04	1.08 ± 0.29	1.14 ± 0.26
EF-FCS	0.66 ± 0.02	1.01 ± 0.24	0.71 ± 0.04	1.02 ± 0.25	1.42 ± 0.56
EF-FCD	0.69 ± 0.02	0.85 ± 0.11	0.71 ± 0.02	0.87 ± 0.12	0.75 ± 0.06
EF-CNN	0.77 ± 0.04	0.94 ± 0.18	0.84 ± 0.04	1.1 ± 0.28	0.83 ± 0.02
IF-FC	0.57 ± 0.07	1.07 ± 0.26	0.65 ± 0.06	1.15 ± 0.31	1.05 ± 0.37
IF-FCS	**0.56 ± 0.06**	1.0 ± 0.27	0.62 ± 0.04	1.07 ± 0.31	1.31 ± 0.32
IF-FCD	**0.56 ± 0.04**	0.72 ± 0.08	**0.61 ± 0.06**	0.76 ± 0.09	0.7 ± 0.12
IF-CNN	0.58 ± 0.04	0.67 ± 0.06	0.7 ± 0.03	0.83 ± 0.12	**0.55 ± 0.02**
x (top 50)	0.63 ± 0.07	**0.64 ± 0.07**	0.64 ± 0.06	**0.65 ± 0.06**	0.73 ± 0.11
Expert x^a	0.63 ± 0.08	**0.64 ± 0.07**	0.64 ± 0.06	**0.65 ± 0.06**	0.71 ± 0.05

tion error, derived from the convolutional auto-encoder in combination with expert features, yields the best performance for both failure and Remaining Useful Life (RUL) prediction. Furthermore, the results demonstrate the efficiency of self-supervised feature learning using auto-encoders in generating valuable features to recognize faults in multi-modal data Important future work, however, includes improving the performance by filtering out premature- or mis-replacement samples in the training population.

REFERENCES

Al-Dulaimi, A., Zabihi, S., Asif, A., & Mohammadi, A. (2019). A multimodal and hybrid deep neural network model for remaining useful life estimation. *Computers in industry*, *108*, 186–196.

Chen, J., Li, D., Huang, R., Chen, Z., & Li, W. (2023). Aero-engine remaining useful life prediction method with self-adaptive multimodal data fusion and cluster-ensemble transfer regression. *Reliability Engineering & System Safety*, *234*, 109151.

Guo, W., Wang, J., & Wang, S. (2019). Deep multimodal representation learning: A survey. *IEEE Access*, *7*, 63373–63394.

Huang, Y., Tao, J., Sun, G., Wu, T., Yu, L., & Zhao, X. (2023). A novel digital twin approach based on deep multimodal information fusion for aero-engine fault diagnosis. *Energy*, *270*, 126894.

Inceoglu, A., Aksoy, E. E., Ak, A. C., & Sariel, S. (2021). Fino-net: A deep multimodal sensor fusion framework for manipulation failure detection. In *2021 ieee/rsj international conference on intelligent robots and systems (iros)* (pp. 6841–6847).

Li, C., Sanchez, R.-V., Zurita, G., Cerrada, M., Cabrera, D., & Vásquez, R. E. (2015). Multimodal deep support vector classification with homologous features and its application to gearbox fault diagnosis. *Neurocomputing*, *168*, 119–127.

Li, H., Huang, J., Huang, J., Chai, S., Zhao, L., & Xia, Y. (2021). Deep multimodal learning and fusion based intelligent fault diagnosis approach. *Journal of Beijing Institute of Technology*, *30*(2), 172–185.

Pang, L., Zhu, S., & Ngo, C.-W. (2015). Deep multimodal learning for affective analysis and retrieval. *IEEE Transactions on Multimedia*, *17*(11), 2008–2020.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Ramachandram, D., & Taylor, G. W. (2017). Deep multimodal learning: A survey on recent advances and trends. *IEEE signal processing magazine*, *34*(6), 96–108.