Proceedings of CONF-SEML 2026 Symposium: Learning and Decision Making in Multi Agent Software Systems
DOI: 10.54254/2755-2721/2026.BA32663

Comparative Study of LSTM, Transformer, and Mixture of Experts for RUL Prediction with Regime-Aware Optimization Research

Sisi Ma

School of Data Science and Intelligent Media, Communication University of China, Bejing, China
202216103008@mails.cuc.edu.cn

Abstract. Remaining Useful Life (RUL) prediction is crucial for predictive maintenance in complex engineering systems. In recent years, deep learning methods have become the dominant approach for RUL prediction due to their ability to capture complex temporal dependencies. Long Short-Term Memory (LSTM) networks, originally designed for sequence modeling, have been widely applied in time-series prediction tasks. The Transformer architecture, known for its powerful attention mechanism, has achieved remarkable success in various sequential data analysis domains. However, these methods typically assume a single global degradation pattern, which may limit their performance under varying operating conditions. To address this issue, this paper presents a two-fold investigation: first, a comparative performance analysis of three prominent architectures—LSTM, Transformer, and Mixture of Experts (MoE). Second, we focus on the optimization of the MoE framework by proposing a Regime-Aware MoE (RA-MoE). This model integrates regime identification techniques (K-Means, HMM, and VAE) to optimize the gating mechanism. Experimental results show that while LSTM remains the most robust performer among the candidate architectures, the proposed RA-MoE significantly enhances the performance of the standard MoE architecture, demonstrating the effectiveness of regime-aware optimization in complex scenarios.

Keywords: Remaining Useful Life (RUL), Deep Learning, Mixture of Experts (MoE), Regime Identification, CMAPSS Dataset, LSTM, Transformer

1. Introduction

With the development of modern industrial equipment towards intelligence and complexity, Prognostics and Health Management (PHM) has attracted widespread attention. Recent systematic reviews have provided a comprehensive roadmap for current deep learning-based Remaining Useful Life (RUL) research [1]. As a key technology of PHM, Predictive Maintenance (PdM) reduces costs through proactive maintenance. As the core of PdM, RUL prediction aims to estimate the safe operation time of equipment based on historical and current data, which is crucial for optimizing maintenance allocation and preventing accidents.

DOI: 10.54254/2755-2721/2026.BA32663

RUL prediction technologies have continuously evolved, with early studies focusing on physics-based and statistical degradation models. With the explosion of sensor data, deep learning methods such as LSTM and Transformer have become the mainstream, excelling at capturing long-term dependencies. Meanwhile, Mixture of Experts and regime-aware models have been introduced to handle multi-regime operating conditions, and frameworks integrating Time-Series K-Means with bidirectional LSTMs have demonstrated significantly better performance than monolithic models [2].

Although these models have demonstrated promising performance, most of them still assume relatively stationary degradation patterns. In real industrial systems, however, equipment often operates under significantly varying regimes, which can degrade the robustness of RUL predictions if ignored. This is particularly evident in aero-engines where sensor distributions drift across takeoff, cruise, and landing phases; thus, explicit regime-awareness is essential to capture time-varying degradation laws [3]. This paper therefore focuses on regime-aware Mixture of Experts (RA-MoE) models that explicitly integrate regime identification modules (e.g., K-Means, HMM, VAE) with MoE to adaptively capture different operating conditions on the CMAPSS dataset.

The main contributions of this paper include a comprehensive evaluation of various deep learning architectures for RUL prediction on the NASA CMAPSS dataset, the construction and validation of RA-MoE models based on three regime identification methods, and a systematic ablation study analyzing the impact of expert count on prediction accuracy to identify the optimal expert configuration.

2. Methodology

2.1. Problem definition

The objective of Remaining Useful Life prediction is to forecast the number of operating cycles remaining before equipment failure occurs based on historical operational data. Given a sample containing multi-sensor time-series data $X=\{x_1,x_2,...,x_T\}$, where $x_t \in R^D$ represents the D-dimensional sensor measurement at time step t, the RUL prediction task can be formalized as learning a mapping function $f:X\rightarrow y$, where $y \in R^+$ denotes the predicted remaining useful life. This paper adopts the piecewise linear degradation assumption, specially, when the cumulative operating time exceeds a predefined threshold (set to 125 cycles in this paper), the RUL is capped at this threshold. This assumption follows the standard evaluation protocol of the NASA CMAPSS dataset, aiming to more reasonably assess prediction behavior during the terminal degradation phase. Furthermore, identifying the specific change point at which degradation transitions from a healthy state to a faulty state—rather than assuming degradation starts from the first cycle—has been shown to reduce noise in the training process [4].

2.2. Model architectures

$$\hat{y}_t=\max(0,P-t) \tag{1}$$

where P denotes the total expected operating cycles from new to failure, and t represents the current cumulative operating time.

Long Short-Term Memory (LSTM) addresses the gradient vanishing problem of conventional recurrent neural networks through introducing gating mechanisms. The core computational formulas are:

DOI: 10.54254/2755-2721/2026.BA32663

$$f_t=\sigma(W_f\cdot[h_{t-1},x_t]+b_f) \tag{2}$$

$$i_t=\sigma(W_i\cdot[h_{t-1},x_t]+b_i) \tag{3}$$

$$\tilde{c}_t=\tanh(W_c\cdot[h_{t-1},x_t]+b_c) \tag{4}$$

$$c_t=f_t\odot c_{t-1}+i_t\odot\tilde{c}_t \tag{5}$$

$$o_t=\sigma(W_o\cdot[h_{t-1},x_t]+b_o) \tag{6}$$

$$h_t=o_t\odot\tanh(c_t) \tag{7}$$

where σ denotes the sigmoid function, $\odot$ represents element-wise multiplication, f_t, i_t, and o_t are the forget gate, input gate, and output gate respectively, c_t is the cell state, and h_t is the hidden state. The final RUL prediction is obtained by passing the last hidden state through a fully connected layer.

The Transformer model employs self-attention mechanisms to capture long-range dependencies within sequences. The computation of queries, keys, and values is:

$$\text{Attention}(Q,K,V)=\text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{8}$$

where d_k is the dimension of key vectors. Multi-head attention executes h attention functions in parallel, concatenates the results, and obtains the final output through a linear transformation.

The Mixture of Experts (MoE) consists of several expert networks and a gating network. For a given input x, the gating network computes the expert weight distribution over the ensemble:

$$\hat{y}=\sum_{i=1}^{N} g_i\cdot E_i(x) \tag{9}$$

where N is the total number of experts, and g_i is the activation weight of the i-th expert.

2.3. Regime-aware mixture of experts

For multi-regime data, this paper proposes three strategies for integrating regime identification with MoE.

K-Means + MoE: Initially, K-Means clustering is applied to historical sensor data to obtain K regime centers $\{\mu_1,\ldots,\mu_K\}$. For the final timestep feature x_T of an input sequence, its Euclidean distances to all regime centers are computed and normalized into a probability distribution:

$$p_k=\frac{\exp(-\|x_T-\mu_k\|^2/\tau)}{\sum_{j=1}^{K}\exp(-\|x_T-\mu_j\|^2/\tau)} \tag{10}$$

DOI: 10.54254/2755-2721/2026.BA32663

This regime probability distribution serves as additional input to the gating network, guiding the model to select experts matching the current operational condition.

Hidden Markov Model + MoE: HMM models latent states of time-series data by learning a transition matrix A and observation probability matrix B. Given an input sequence X, the forward algorithm computes the probability of being in different latent states at each timestep $\alpha_t=P(s_t|x_{1:t})$. The final latent state probability distribution α_T is injected into the gating network, enabling the model to perceive the equipment's current regime and its historical evolution.

2.4. Loss function

$$L_{total}=L_{RUL}+\lambda_1 L_{regime}+\lambda_2 L_{diversity} \tag{11}$$

The total loss function consists of three components: RUL prediction loss (Huber loss), regime prediction loss (cross-entropy), and diversity loss (to prevent gate collapse). The diversity loss is computed as the negative entropy of gate weights:

$$L_{diversity}=-\sum_{i=1}^{N} g_i \log g_i \tag{12}$$

3. Experiments

3.1. Dataset

We conduct experiments on the NASA CMAPSS (Commercial Modular Aero-Propulsion System Simulation) dataset, which is widely used for RUL prediction research . The dataset simulates a large commercial aircraft turbofan engine with 21 sensor channels measuring various physical parameters such as temperature, pressure, and rotational speed. The dataset consists of four independent subsets with varying complexity:

·FD001: Single operating regime, 100 training engines, 100 test engines. This subset represents the simplest scenario with consistent operational conditions.

·FD002: Multiple operating regimes, 260 training engines, 259 test engines. This subset exhibits more complex degradation patterns due to varying operational conditions.

·FD003: Single operating regime with additional fan degradation fault mode, 100 training engines, 100 test engines. The presence of a distinct fault mode increases prediction complexity.

·FD004: Multiple operating regimes with multiple fault modes, 249 training engines, 248 test engines. This is the most challenging subset, featuring both operational regime variations and multiple failure mechanisms.

Data Preprocessing: Following standard practices in RUL prediction , we apply piecewise linear degradation assumption where the RUL is capped at 125 cycles. This approach addresses the issue of equal degradation rates in early operating phases while emphasizing accurate predictions near failure. All sensor readings are normalized using z-score standardization to zero mean and unit variance, computed on the training set and applied consistently to validation and test sets. While we employ manual normalization, emerging Automated Machine Learning (AutoML) frameworks now offer end-to-end pipelines that can optimize the entire workflow from preprocessing to model ensemble [5].

Feature Selection: From the original 21 sensor channels, we select 14 key sensors that exhibit significant degradation trends. These include sensors measuring total temperature, pressure ratios,

DOI: 10.54254/2755-2721/2026.BA32663

physical core speed, and other critical engine parameters. Aggregated feature importance techniques, which synthesize insights from multiple algorithms such as Random Forest and LASSO, ensure the selection of the most robust degradation signals and simultaneously reduce model complexity [6].The input sequence length is set to 50 time steps, providing a balanced trade-off between historical information and computational efficiency.

Dataset Split: For each subset, we use 80% of the training data for model training and 20% for validation. The test set is used for final performance evaluation.

3.2. Experimental setup

All deep learning models are implemented in PyTorch, with an initial learning rate of 1×10^{-4} and weight decay of 1×10^{-5} . The learning rate follows a cosine annealing schedule with a minimum learning rate of 1×10^{-6} . Models are trained for 50 epochs with early stopping based on validation RMSE. The batch size is set to 256. Gradient clipping with a maximum norm of 1.0 is applied to prevent gradient explosion.

For the MoE variants, we employ 4 expert networks with a hidden dimension of 128. The regime encoder produces a probability distribution over 4 latent regimes. The loss function combines Huber loss for RUL prediction, cross-entropy loss for regime classification, and diversity loss to prevent gate collapse. The loss weights are set to $\lambda_1=0.1$ for regime loss and $\lambda_2=0.05$ for diversity loss.

All experiments are repeated 3 times with different random seeds, and the mean performance with standard deviation is reported.

3.3. Evaluation metrics

We evaluate model performance using three standard metrics for RUL prediction tasks:

1. RMSE (Root Mean Square Error): Measures the standard deviation of prediction errors. A lower RMSE indicates better performance. RMSE is sensitive to large errors and penalizes severe mispredictions more heavily.

2. MAE (Mean Absolute Error): Represents the average magnitude of prediction errors without considering direction. MAE is more robust to outliers compared to RMSE.

3. NASA Score: A weighted scoring function used in NASA's PHM challenges. It exponentially penalizes underestimation (predicting RUL lower than actual) more severely than overestimation, as unexpected failures are more costly than delayed maintenance.

The NASA Score is asymmetric and defined as:

$$\text{Score}=\sum_{i=1}^{N} s_i, \quad s_i=\begin{cases} e^{-\frac{d_i}{13}}-1, & d_i<0 \\ e^{\frac{d_i}{10}}-1, & d_i\geq 0 \end{cases} \tag{13}$$

where d_i is the deviation between predicted and actual values.

4. Results and discussion

4.1. Model comparison

Table 1 and Figure 1 present the performance comparison of all models across the four CMAPSS subsets. While Table 1 provides precise numerical values, the heatmap in Figure 1 visually

DOI: 10.54254/2755-2721/2026.BA32663

highlights the performance density, clearly showing LSTM's consistent superiority (indicated by the darkest green areas) compared to other architectures.

Table 1. Performance comparison of all models on the CMAPSS dataset

Subset	Model	RMSE	MAE	NASA Score
FD001	Linear	49.32	38.01	6852832.00
FD001	Transformer	46.41	34.92	3143916.33
FD001	LSTM	6.21	4.05	1961.12
FD001	MoE	19.99	14.54	28959.40
FD001	MoE + K-Means	19.71	14.25	24931.69
FD001	MoE + HMM	19.76	14.34	26370.85
FD001	MoE + VAE	19.69	14.13	25488.35
FD002	Linear	41.38	34.01	1842334.70
FD002	Transformer	20.18	14.60	151205.78
FD002	LSTM	12.41	8.73	22607.93
FD002	MoE	22.11	15.62	142765.35
FD002	MoE + K-Means	21.94	15.61	139549.50
FD002	MoE + HMM	21.96	15.61	138874.58
FD002	MoE + VAE	22.06	15.64	145925.50
FD003	Linear	56.55	44.25	10073586.75
FD003	Transformer	9.61	5.58	5912.13
FD003	LSTM	5.93	3.56	2531.07
FD003	MoE	17.50	10.95	32678.51
FD003	MoE + K-Means	17.45	10.89	29266.70
FD003	MoE + HMM	17.35	11.07	28544.69
FD003	MoE + VAE	17.33	11.23	32101.64
FD004	Linear	48.51	38.49	10445893.75
FD004	Transformer	55.18	42.38	17935898.56
FD004	LSTM	13.39	8.27	623486.28
FD004	MoE	21.32	13.19	858956.91
FD004	MoE + K-Means	21.02	13.16	678949.94
FD004	MoE + HMM	20.96	13.20	666772.31
FD004	MoE + VAE	21.07	13.17	653907.38

From Table 1, we observe several key findings :

·LSTM achieves the best overall performance, particularly on single-regime datasets (FD001, FD003) with RMSE of 6.21 and 5.93 respectively. This aligns with recent comparative studies on CMAPSS, which identify LSTM as a robust baseline for stable scenarios [7]. This observation demonstrates that LSTM's sequential modeling capability is well-suited for capturing degradation patterns under stable operating conditions. The forget gate mechanism effectively learns the temporal dependencies in equipment degradation.

DOI: 10.54254/2755-2721/2026.BA32663

·Transformer shows competitive performance on FD003 but struggles on multi-regime datasets like FD004. This suggests that while self-attention can capture long-range dependencies, it may be less effective when dealing with regime shifts that introduce non-stationary patterns into the data .

·The simple Linear degradation model provides a reasonable baseline, outperforming the Transformer on FD004. Research confirms that well-engineered linear frameworks, utilizing distance-to-boundary features, can provide high interpretability and efficiency suitable for real-time industrial deployment [8]. This highlights that for some datasets, complex models may overfit to noise rather than learn meaningful degradation patterns.

·MoE baseline underperforms LSTM on most subsets. This indicates that without proper regime-aware gating, the MoE architecture may not effectively leverage its expert networks for RUL prediction tasks .

·Regime identification modules improve MoE performance on multi-regime datasets, as shown in Figure 2. The integration of K-Means, HMM, or VAE encoders enables the model to better understand the current operational regime and route to appropriate experts .

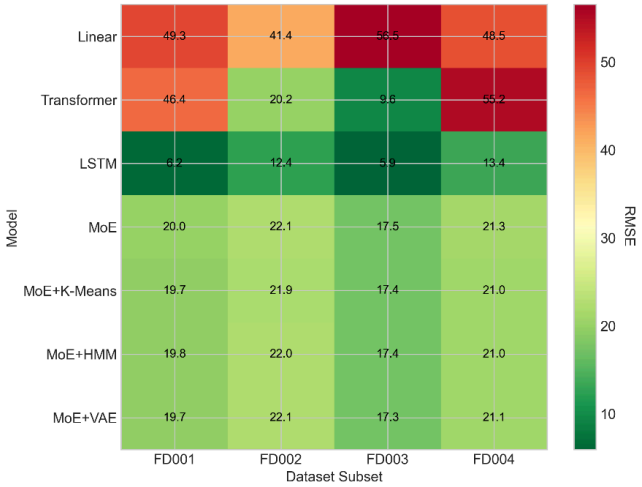

Figure 1. RMSE heatmap: model vs dataset

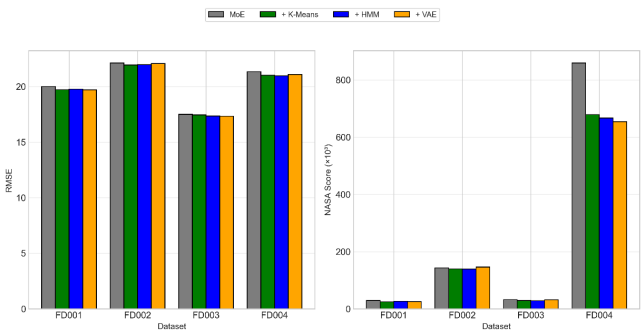

Figure 2. Performance comparison: MoE vs regime-aware MoE

4.2. Impact of regime identification on moe

To analyze the effect of regime identification, we focus on the FD004 multi-regime dataset where regime-aware MoE shows the most significant improvements.

Table 2 shows the impact of the number of experts in the HMM-MoE model:

DOI: 10.54254/2755-2721/2026.BA32663

Table 2. Ablation study on the number of experts (HMM-MoE on FD004)

Model	Experts	RMSE	MAE	NASA Score
MoE	4	22.13	13.86	849897.94
HMM-MoE	2	21.70	13.91	688671.56
HMM-MoE	3	21.70	13.67	747612.25
HMM-MoE	4	21.46	13.79	646949.19
HMM-MoE	6	22.23	13.70	891127.50
HMM-MoE	8	22.04	13.51	873475.81

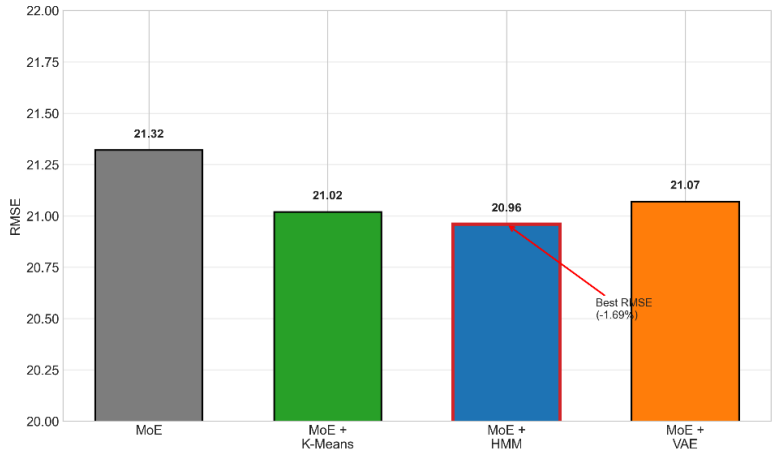

Figure 3. RMSE comparison: regime encoders on FD004 dataset

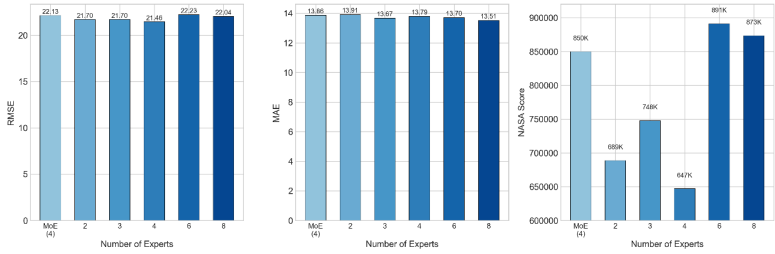

Figure 4. Ablation study: impact of number of experts (HMM-MoE on FD004)

Key findings from the ablation study:

·All three regime encoders improve baseline MoE on the multi-regime FD004 dataset, as illustrated in Figure. 3. This confirms that incorporating regime information helps the model better understand the underlying operational states and route inputs to the most suitable experts.

·The HMM-based regime encoder achieves the best RMSE (20.96), demonstrating that the sequential modeling capability of HMM is well-suited for capturing temporal regime transitions in equipment operation. Meanwhile, VAE yields the best NASA Score (653907.38), indicating that its probabilistic latent representation can better capture the uncertainty in RUL prediction.

·The optimal number of experts for HMM-MoE is 4, achieving 3.03% RMSE improvement and 23.88% NASA Score improvement over the baseline MoE, as illustrated in Figure. 4. Increasing the number of experts beyond 4 does not provide additional benefits and may even degrade performance due to increased model complexity and training instability.

DOI: 10.54254/2755-2721/2026.BA32663

·The improvement from regime identification is more pronounced on multi-regime datasets (FD002, FD004) compared to single-regime datasets (FD001, FD003), validating the hypothesis that regime-aware mechanisms are particularly valuable when operational conditions vary significantly.

5. Conclusion

This paper presents a comprehensive comparative study of deep learning models for RUL prediction. Our experimental results on the CMAPSS dataset demonstrate:

·LSTM achieves the best overall performance among all baseline models, especially on single-regime datasets (FD001, FD003);

·While MoE models underperform LSTM, the integration of regime identification modules (K-Means, HMM, VAE) provides measurable improvements, particularly on multi-regime datasets (FD002, FD004);

·Among the three regime encoders, HMM achieves the best RMSE reduction (3.03% on FD004), while VAE yields the best reduction in the NASA Score (23.88% on FD004);

·The optimal number of experts for HMM-MoE is 4, which provides the best balance between model complexity and performance.

These findings suggest that for RUL prediction tasks, LSTM should be the primary choice due to its superior performance. However, MoE with regime identification can be considered as an alternative approach, especially when interpretability of different operational regimes is important.

Future work will explore more advanced regime representation learning methods and investigate hybrid approaches that combine the strengths of LSTM and regime-aware MoE.

References

[1] Wu, F., Wu, Q., Tan, Y., & Xu, X. (2024). Remaining useful life prediction based on deep learning: A survey. Sensors, 24(11), 3454. https: //doi.org/10.3390/s24113454

[2] Seo, J. (2025). Fault-type-aware remaining useful life prediction of aircraft engines using an integrated deep learning framework. International Journal of Prognostics and Health Management, 16(2).

[3] Wang, Y., & Zhao, Y. (2022). Attention-based dual-channel deep neural network for aero-engine RUL prediction under time-varying operating conditions. Proceedings of the 2022 IEEE 5th International Conference on Electronics Technology (ICET), 988-993. https: //doi.org/10.1109/ICET55642.2022.9944445

[4] Rath, S., Saha, D., Chatterjee, S., & Chakraborty, A. K. (2025). Remaining useful life prediction of turbofan engine in varied operational conditions considering change point: A novel deep learning approach with optimum features. Mathematics, 13(1), 130. https: //doi.org/10.3390/math13010130

[5] Richmond, D., Ayodeji, O., & Standley, T. (2023). Automated machine learning for remaining useful life predictions (arXiv: 2306.12215). arXiv. https: //doi.org/10.48550/arXiv.2306.12215

[6] Alomari, Y., Andoga, M., & Baptista, M. L. (2024). Advancing aircraft engine RUL predictions: An interpretable integrated approach of feature engineering and aggregated feature importance. Scientific Reports, 13, 40315.

[7] Guilherme, D. N. V. B. (2024). Remaining useful life prediction on the NASA CMAPSS dataset comparing LSTM and transformer models [Master's dissertation, Instituto Politécnico do Porto]. ReCIPP. https: //hdl.handle.net/10400.22/25413

[8] Yildirim, M., & Guler, N. (2025). Linear methods for predictive maintenance: The case of NASA C-MAPSS datasets. Applied Sciences, 15(18), 9945. https: //doi.org/10.3390/app15189945