# A Compact CNN-Transformer Model for Robust Fault Diagnosis in Large-Scale Chemical Processes

Pratyush Kumar Pal[1], Sumona Ray[2] , Narottam Behera[3], Somnath Chowdhury[4], Abhiram Hens[5], and Sandip Kumar Lahiri[6*]

[1,2,3,4,5,6]*National Institute of Technology Durgapur, Mahatma Gandhi Avenue, Durgapur 713209, West Bengal, India*
*pkp.21ch1502@phd.nitdgp.ac.in*
*sr.19ch1501@phd.nitdgp.ac.in*
*nb.20ch1504@phd.nitdgp.ac.in*
*sc.19ch1103@phd.nitdgp.ac.in*
*ahens.che@nitdgp.ac.in*
**Corresponding author: sklahiri.che@nitdgp.ac.in*

**ABSTRACT**

Ensuring the safe and stable operation of large-scale chemical processes requires accurate fault detection and diagnosis under nonlinear dynamics and strong variable interactions. This study investigates deep learning-based fault diagnosis for the Tennessee Eastman Process (TEP), focusing on whether performance improvements beyond near-saturation Long Short-Term Memory (LSTM) baselines can be achieved. A standardised TEP dataset with 52 measured and manipulated variables is used, excluding the non-detectable fault cases (IDV 3, 9, and 15), resulting in a 19-class classification problem.

A hybrid Convolutional Neural Network–Transformer (CNN-Transformer) architecture is proposed in which one-dimensional convolutional layers capture local cross-variable correlations, while a Transformer encoder models long-range temporal dependencies through self-attention. To ensure fair comparison, both the proposed model and a strong LSTM baseline are trained and evaluated under identical preprocessing, optimization, and evaluation protocols.

The CNN-Transformer achieves an overall classification accuracy of 99.92%, marginally outperforming the LSTM baseline (99.86%). Although the numerical improvement is slight, the proposed model consistently yields higher macro-averaged F1-scores and reduced fault-wise misclassification, indicating enhanced robustness in challenging fault scenarios.

The key contribution of this work is demonstrating that combining convolutional feature extraction with attention-based temporal modelling provides consistent class-level robustness beyond near-saturated recurrent architectures, while maintaining a compact structure suitable for practical deployment.



https://doi.org/10.36001/IJPHM.2026.v17i1.4744

## 1. INTRODUCTION

Fault detection and diagnosis (FDD) is crucial for ensuring safe, stable, and efficient operation of modern chemical processes. (Agarwal et al., 2021; Chadha and Schwung, 2017; Labbaf-Khaniki et al., 2024; Pozdnyakov et al., 2024; Yin et al., 2014, 2012). As modern processes become more complex and interconnected, fault propagation may lead to safety hazards or unplanned process shutdowns. However, accurate fault diagnosis is challenging due to nonlinear behaviour, strong variable interactions, noise, and similar fault patterns. (Pal et al., 2025).

The Tennessee Eastman Process (TEP) (Downs and Vogel, 1993; Chen et al., 2022) is commonly used as a benchmark for evaluating FDD methods in complex industrial systems. Initially introduced as a realistic plant-wide control and monitoring challenge, the TEP (schematic diagram shown in Figure 1) represents a large-scale chemical process with nonlinear reactions, recycle streams, and closed-loop control structures. ( Downs and Vogel, 1993; Reinartz et al., 2021; Yin et al., 2012).

Early data-driven FDD approaches for the TEP mainly relied on multivariate statistical techniques like principal component analysis (PCA), dynamic PCA (DPCA), Support Vector Machine (SVM) (Onel et al., 2019; Reinartz et al., 2021; Yin et al., 2012) and related linear extensions. Although these methods are computationally efficient and interpretable, their effectiveness is fundamentally limited by

linearity assumptions and their limited ability to capture complex temporal dependencies. As a result, they often show reduced detection accuracy and high misclassification rates for nonlinear, dynamic, or slowly evolving faults. (Yin et al., 2012).

To address these limitations, deep learning-based methods have gained increasing attention in process monitoring and fault diagnosis. Neural network architectures such as autoencoders, convolutional neural networks (CNNs), and recurrent neural networks (RNNs) (Chen et al., 2022; Verma et al., 2022) have shown significant improvements over classical statistical approaches by learning nonlinear representations directly from process data. Numerous studies have reported strong performance of deep learning models on the TEP benchmark, with classification accuracies nearing saturation levels (Agarwal et al., 2022; Chen et al., 2023; Verma et al., 2022; Zhang et al., 2022).

Despite extensive research on deep learning-based fault diagnosis for the TEP benchmark, each deep learning architecture exhibits specific limitations when applied to complex industrial process data. Convolutional Neural Networks (CNNs) are effective at extracting local spatial patterns from multivariate sensor signals through convolutional filters. However, their receptive field is inherently limited by kernel size and network depth, which can restrict their ability to capture long-range temporal dependencies present in slowly evolving industrial faults. (Chadha and Schwung, 2017; Lomov et al., 2021).

Recurrent architectures such as Long Short-Term Memory (LSTM) networks address temporal modelling by maintaining gated memory states that propagate information across time steps. While this structure enables modelling of sequential dependencies, LSTM networks process data sequentially, which limits parallelisation and may reduce their ability to capture global temporal relationships in long time-series sequences efficiently (Agarwal et al., 2022; Verma et al., 2022).

More recently, Transformer architectures have been introduced for time-series modelling, using self-attention mechanisms that directly learn relationships across all time steps in a sequence. This attention-based formulation allows efficient modelling of long-range dependencies and parallel sequence processing. (Vaswani et al., 2017; Zhang et al., 2022). However, when applied directly to raw multivariate process measurements, Transformers may underutilize localised sensor interactions that arise from tightly coupled physical process variables in chemical systems.
These complementary characteristics suggest that hybrid architectures combining convolutional feature extraction with attention-based temporal modelling may provide a more balanced representation of both local sensor interactions and global temporal dynamics.

Consequently, several recent studies have explored hybrid architectures combining convolutional feature extraction with sequential or attention-based temporal modelling (Chen et al., 2022). Although CNN-LSTM hybrid architectures have demonstrated strong performance for industrial time-series analysis, several limitations remain. First, LSTM-based temporal modelling processes sequences sequentially, which restricts parallelisation and may increase computational cost when handling long multivariate sequences typical of industrial process data. Second, many CNN-LSTM frameworks rely on deep stacked architectures that significantly increase model size and training complexity, limiting their suitability for real-time industrial monitoring applications.

Recent research has also explored Transformer-based architectures for long-range temporal modelling via self-attention mechanisms. While Transformers effectively capture global temporal relationships, they may underutilize local correlations among tightly coupled process variables when applied directly to raw multivariate sensor data. Furthermore, many Transformer-based frameworks introduce large attention stacks or complex architectures, thereby increasing computational overhead.

These limitations highlight the need for compact hybrid architectures that can jointly capture local sensor interactions and long-range temporal dependencies while maintaining manageable model complexity. However, many existing hybrid approaches rely on complex stacked architectures or require large model sizes, which may limit their deployment in real-time monitoring systems. These limitations motivate the investigation of compact hybrid architectures that can jointly model local sensor correlations and global temporal dependencies while maintaining efficient model complexity.

Despite the strong performance of existing deep learning approaches for the Tennessee Eastman benchmark, several research gaps remain. Many previous studies have focused on increasing classification accuracy by introducing deeper architectures or complex hybrid models, but relatively few works have investigated whether compact hybrid architectures can improve diagnostic robustness beyond near-saturated recurrent baselines. In particular, it remains unclear whether combining lightweight convolutional feature extraction with Transformer-based temporal modelling can enhance class-level robustness while maintaining a compact model structure suitable for industrial deployment.

Motivated by this gap, the present study proposes a hybrid Convolutional Neural Network-Transformer (CNN-Transformer) framework for fault diagnosis in the Tennessee Eastman Process. The objective is to examine whether combining convolutional feature extraction with attention-based temporal modelling can improve diagnostic robustness

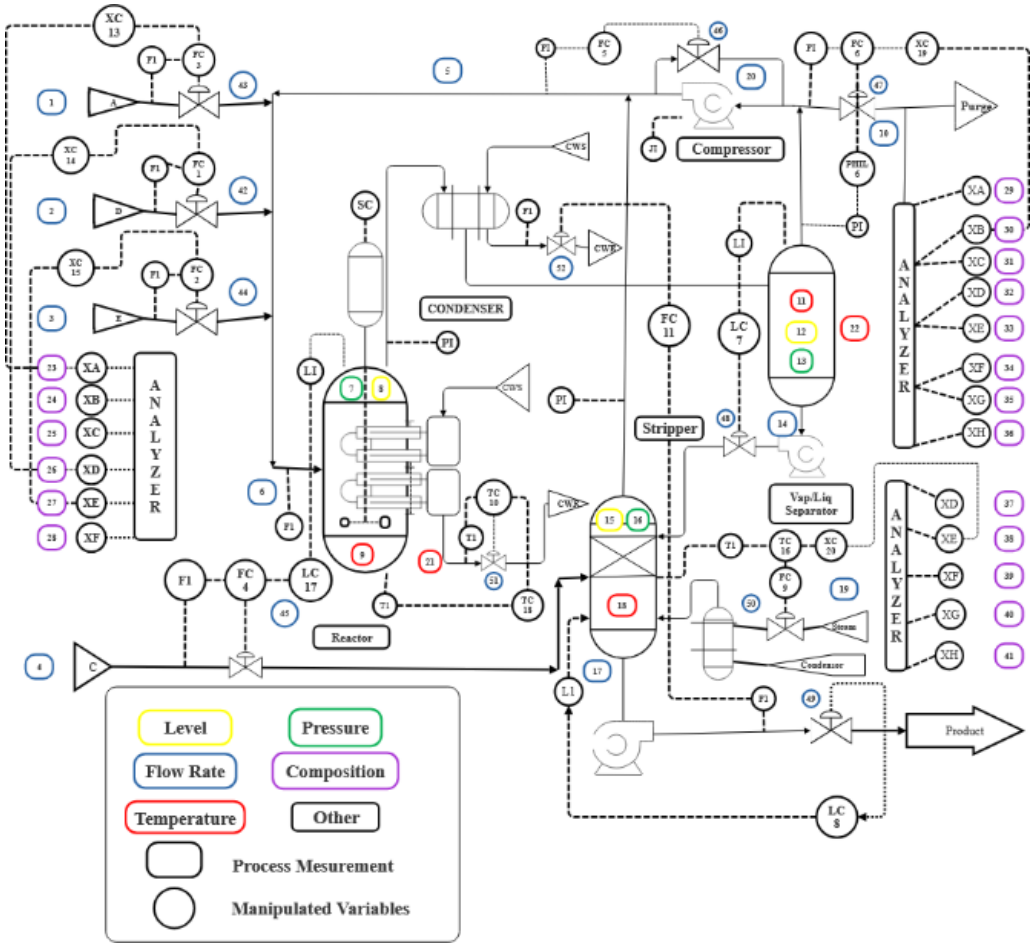


Figure 1. Schematic of the TEP flow chart

beyond near-saturated recurrent architectures while maintaining a compact and computationally efficient model structure.

The main contributions of this study can be summarised as follows:

1. A compact CNN-Transformer architecture that integrates one-dimensional convolutional feature extraction with Transformer-based self-attention to jointly capture local cross-variable correlations and global temporal dependencies in multivariate process data.

2. A controlled experimental comparison against a strong stacked LSTM baseline using identical preprocessing, training settings, and evaluation metrics, enabling a fair architectural comparison independent of experimental configuration.

3. A comprehensive macro-averaged and fault-wise evaluation on the Tennessee Eastman benchmark demonstrating improved class-level robustness for challenging fault scenarios involving gradual dynamics, noise disturbances, and overlapping fault signatures.

The remainder of this paper is organised as follows. Section 2 introduces the main theories behind the LSTM, CNN, and CNN-Transformer models. Section 3 explains the proposed method in detail. Section 4 describes the Tennessee Eastman Process and shows how the proposed approach is applied to its dataset. Section 5 presents the results and discusses the study's implications. Lastly, Section 6 concludes the paper.

## 2. DEEP LEARNING MODELS FOR FAULT DIAGNOSIS IN INDUSTRIAL PROCESSES

Deep learning has become an important approach for fault detection and diagnosis in complex industrial processes, as it can automatically learn nonlinear representations from multivariate process data. Architectures such as Long Short-Term Memory (LSTM) networks, Convolutional Neural Networks (CNNs), and Transformer-based models have been widely explored for fault diagnosis in benchmark systems like the Tennessee Eastman Process (Agarwal et al., 2020; Yin et al., 2014; Zhang et al., 2022).

LSTM networks, a subset of recurrent neural networks, are specifically engineered to address vanishing and exploding

gradient challenges through gated memory cells, facilitating effective modelling of long-range temporal dependencies in sequential data (Agarwal et al., 2022; Verma et al., 2022).

Convolutional Neural Networks have also garnered attention for process monitoring due to their robust feature extraction capabilities. When applied using one-dimensional convolutions, CNNs effectively capture local cross-variable correlations among process variables, such as coordinated changes in temperature, pressure, and flow rate. (Chadha and Schwung, 2017; Lomov et al., 2021). However, CNNs face limitations in modelling long-term temporal dependencies due to their fixed receptive fields, which constrain their ability to represent slowly evolving fault dynamics.

Transformer architectures, initially developed for natural language processing, have recently been applied to industrial time-series analysis. By utilising self-attention mechanisms, Transformers explicitly model global temporal relationships and enable parallel sequence processing, thereby enhancing robustness and fault separability in TEP fault diagnosis. (Labbaf-Khaniki et al., 2024; Zhang et al., 2022). Nonetheless, Transformers applied directly to raw process data may miss local sensor interactions, and complex attention designs can increase computational costs. Overall, existing studies suggest that while LSTM, CNN, and Transformer models each offer unique benefits, none individually address all challenges of fault diagnosis in complex chemical processes, prompting the exploration of hybrid modelling strategies.

Several previous studies have explored hybrid deep learning architectures that combine convolutional feature extraction with sequential temporal modelling. For example, CNN-LSTM frameworks have been proposed for process fault diagnosis, in which convolutional layers extract spatial correlations among process variables, while recurrent networks model temporal dependencies in the process dynamics. (Chen et al., 2022).

While such hybrid CNN-LSTM models have demonstrated strong performance on industrial datasets, recurrent architectures process sequences sequentially, which may limit their ability to capture global temporal relationships efficiently. In contrast, the proposed approach replaces recurrent modelling with Transformer-based self-attention, enabling direct global temporal interactions across the sequence while supporting parallel sequence processing.

This architectural difference provides an alternative strategy for modelling complex temporal dynamics in industrial process data while maintaining a relatively compact model structure.

From a practical perspective, these architectures differ in terms of computational complexity, scalability, and interpretability. LSTM networks model sequential dependencies effectively but require step-by-step processing of time-series data, which can limit scalability for long sequences. CNN-based models are computationally efficient due to parallel convolution operations and relatively low parameter counts, but their fixed receptive fields restrict their ability to capture long-range temporal dependencies. Transformer architectures overcome this limitation by using self-attention mechanisms that directly model global temporal relationships and enable parallel sequence processing. However, Transformer models often involve higher computational overhead due to attention operations, particularly for long sequences.

In terms of interpretability, convolutional filters can reveal local feature patterns, while attention mechanisms in Transformers can highlight important temporal relationships in the data. Nevertheless, deep learning models generally remain less interpretable than classical statistical methods, which motivates ongoing research into explainable AI techniques for industrial process monitoring.

Motivated by these observations, this study investigates whether a compact CNN-Transformer architecture can provide improved class-level robustness for fault diagnosis in the Tennessee Eastman Process.

## 3. PROPOSED CNN-TRANSFORMER FAULT DIAGNOSIS METHODOLOGY

This section presents the proposed hybrid Convolutional Neural Network–Transformer (CNN-Transformer) framework for fault diagnosis in the Tennessee Eastman Process (TEP). The methodology is designed to address the limitations of recurrent architectures by integrating local feature extraction and global temporal modelling within a unified deep learning framework.

### 3.1 Motivation and Design Rationale

Long Short-Term Memory (LSTM) networks, illustrated in Figure 2, are widely used for modelling sequential dependencies in industrial time-series data. The stacked LSTM baseline used in this study consists of three recurrent layers that process the multivariate time series and generate a hidden representation, which is subsequently mapped to fault classes through a fully connected output layer.

However, the proposed framework in this work replaces recurrent modelling with a Transformer encoder to capture long-range dependencies via self-attention, while maintaining parallel sequence processing to model temporal

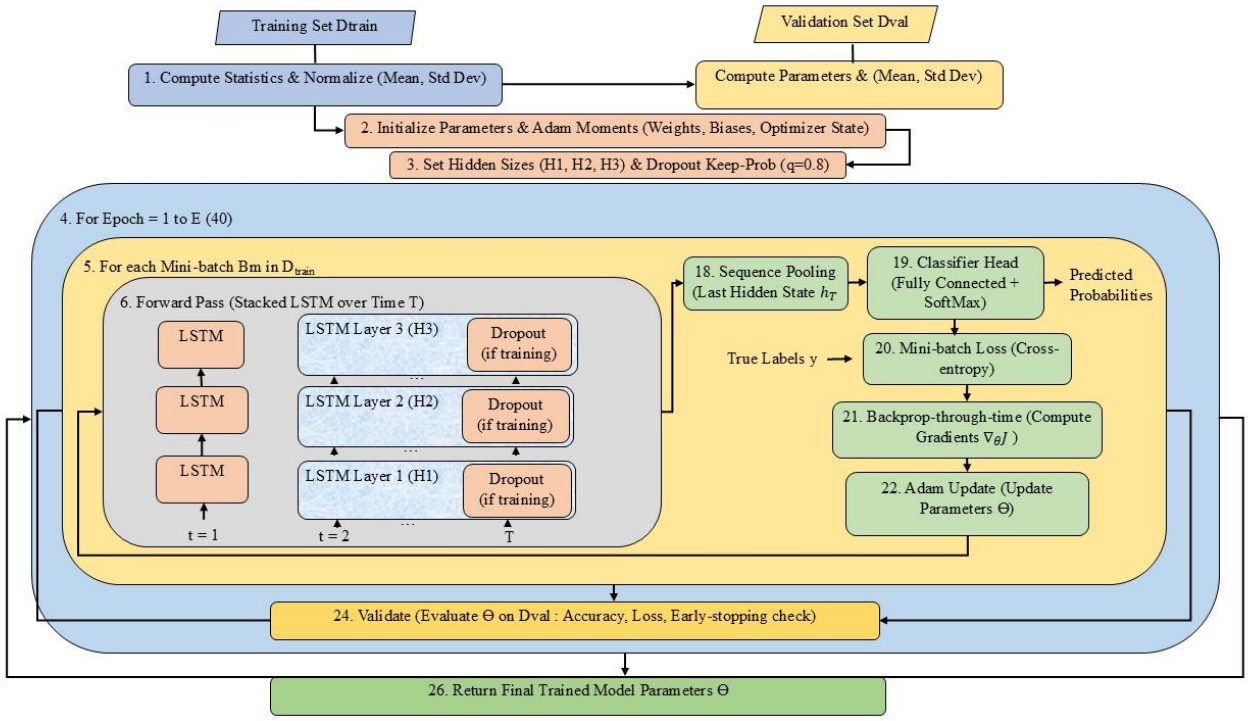


Figure 2. Architecture of the stacked LSTM baseline model used for comparison. The model consists of three sequential LSTM layers followed by a fully connected layer and a softmax classifier for multi-class fault diagnosis.

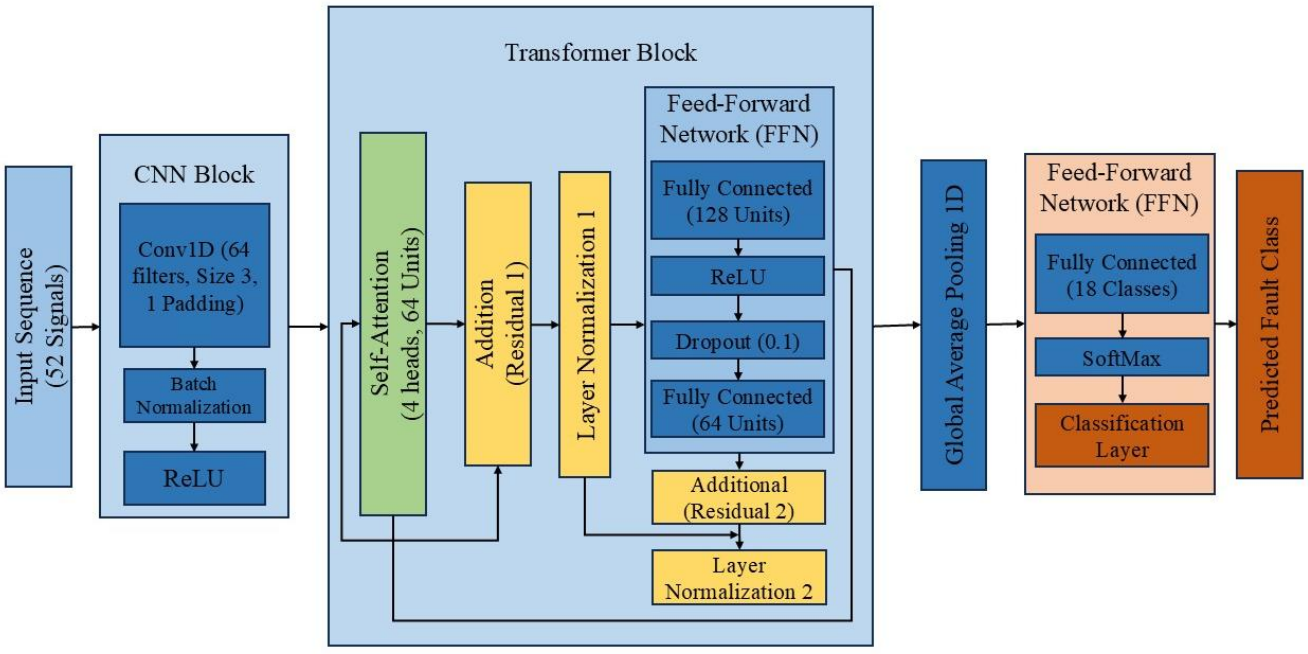


Figure 3. Architecture of the proposed CNN-Transformer model consisting of a 1D convolutional feature extractor followed by two Transformer encoder blocks and a softmax classification layer

dependencies in industrial process data. Recent studies have reported near-saturation accuracy on the TEP benchmark using such architectures. (Agarwal et al., 2022; Verma et al., 2022). Transformer architectures address these limitations by leveraging self-attention mechanisms that explicitly model relationships across all time steps in a sequence and enable parallel processing. (Zhang et al., 2022). However, when applied directly to raw multivariate process data, Transformers may underutilize local correlations among sensor variables that are important in chemical processes. Motivated by these complementary strengths and weaknesses, the proposed CNN-Transformer architecture combines convolutional feature extraction with Transformer-based temporal modelling to achieve more effective fault representation. Figure 3 illustrates the structure of the proposed CNN-Transformer architecture, highlighting the sequential flow from convolutional feature extraction to Transformer-based temporal modelling and final classification.

### 3.2 Overall Architecture

The proposed CNN-Transformer framework consists of three main stages: convolutional feature extraction, Transformer-based temporal modelling, and a classification head. Unlike recurrent architectures, the proposed model does not include LSTM layers. Instead, temporal dependencies are learned via the Transformer encoder's self-attention mechanism.

The architecture is intentionally designed to remain compact compared with deeper hybrid architectures reported in recent studies. (Chen et al., 2023, 2022; Pozdnyakov et al., 2024). The proposed framework uses a single convolutional feature-extraction stage followed by two Transformer encoder blocks, rather than deeper, stacked attention architectures commonly used in large-scale sequence models. This design provides sufficient modelling capacity for capturing temporal dependencies in industrial process signals while maintaining a relatively compact model structure. This design reduces the number of trainable parameters while preserving the ability to capture both local process-variable correlations and global temporal dependencies.

To evaluate architectural compactness, the number of trainable parameters was computed for both models using MATLAB's layer parameter inspection. The stacked LSTM baseline has 43,788 trainable parameters, while the proposed CNN-Transformer architecture has 45,010 trainable parameters. Although the CNN-Transformer integrates convolutional feature extraction and attention-based temporal modelling, the increase in model size is only about 2.8%. This small difference indicates that the proposed architecture maintains a comparable parameter footprint while providing improved class-level robustness and better handling of challenging fault scenarios.

For a multivariate time-series sequence $X = \{x_1, x_2, \dots, x_T\}$, a one-dimensional convolution computes feature representations as

$$h_T = \sigma\left(\sum_{k=0}^{K-1} W_k x_{t-k} + b\right)$$

Where $x_t \in R^d$ is a multivariate input vector at time step $t$, $W_k$ is the convolution weight matrix corresponding to the $k^{th}$ temporal lag, $K$ is the convolution kernel size (temporal receptive field), $b$ is the bias term, $\sigma(\cdot)$ nonlinear activation function (e.g., ReLU), the resulting feature vector $h_t$ represents the extracted local temporal features at the time step $t$.

The first stage of the proposed framework employs one-dimensional convolutional layers applied along the temporal dimension of the multivariate input sequences. These convolutional layers act as local feature extractors, learning short-range cross-variable correlations among process measurements and filtering high-frequency noise in the raw measurements.

The convolutional feature maps are projected into a fixed-dimensional embedding space and passed to a Transformer module. The Transformer module consists of two encoder blocks; each composed of a multi-head self-attention layer followed by a position-wise feed-forward network. Residual connections and layer normalization are applied after each sub-layer to stabilise training and improve gradient propagation. (Chadha and Schwung, 2017; Lomov et al., 2021; Vaswani et al., 2017; Verma et al., 2022).

Recent studies have demonstrated that Transformer-based models can enhance diagnostic robustness and reduce misclassification rates in complex industrial systems, including applications to the TEP benchmark. (Labbaf-Khaniki et al., 2024; Vaswani et al., 2017; Zhang et al., 2022). In the proposed framework, the Transformer encoder complements the convolutional feature extractor by modelling global temporal relationships among the learned features.

All trainable parameters are optimised using stochastic gradient-based methods, and regularisation techniques such as dropout and weight decay is applied where appropriate to improve generalization. These design choices follow best practices in deep learning-based fault diagnosis and are consistent with prior work on the TEP dataset. (Agarwal et al., 2022; Zhang et al., 2022).

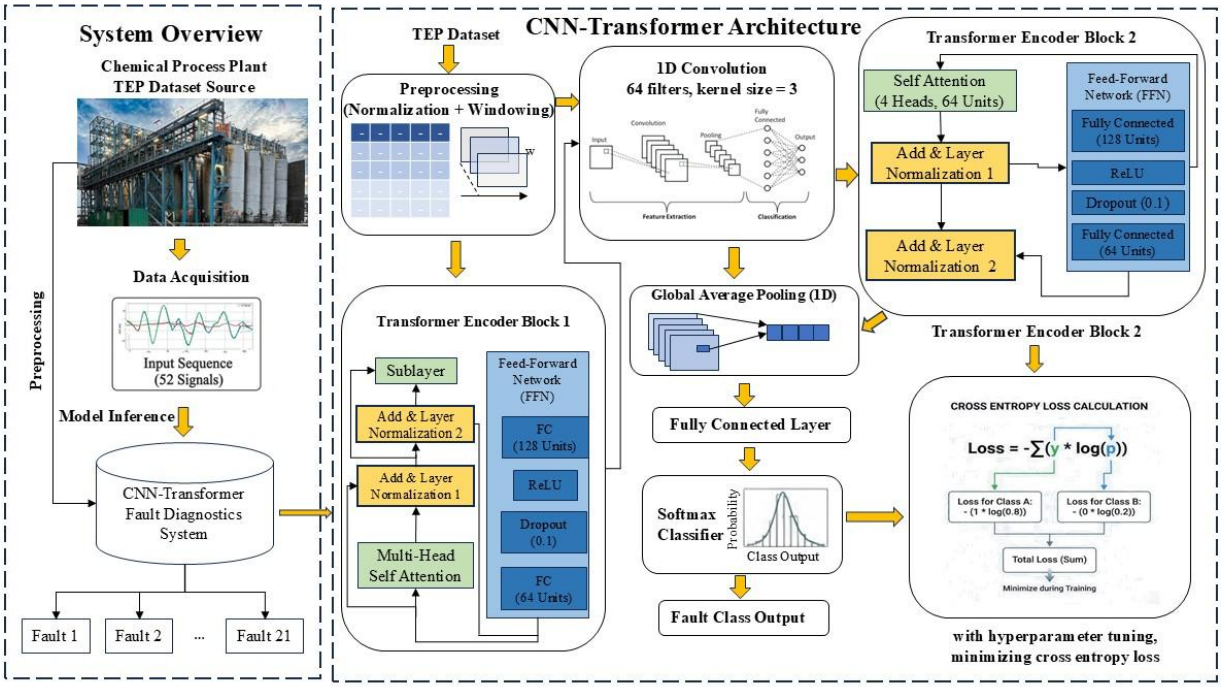


Figure 4. Graphical workflow of the proposed CNN-Transformer fault diagnosis framework integrating convolutional feature extraction and attention-based temporal modelling

Figure 4 illustrates the complete workflow of the proposed CNN-Transformer model. The architecture uses one-dimensional convolution for temporal feature extraction, followed by two Transformer encoder blocks composed of multi-head self-attention and position-wise feed-forward networks. Global average pooling is applied before the final fully connected classification layer.

1D Convolution → Transformer Encoder Block 1 → Transformer Encoder Block 2 → Global Average Pooling → Fully Connected Classification Layer.

### 3.3 Training and Evaluation Protocol

For both architectures, categorical cross-entropy was adopted as the training objective and minimised using the Adam optimiser, with identical learning rate scheduling, mini-batch size, class encoding, and early stopping settings. The performance was assessed on the unseen test set using accuracy, precision, recall, and macro-averaged F1-score computed from the confusion matrix, ensuring a balanced evaluation across all fault classes. The model was selected based on the validation loss to prevent overfitting.

The LSTM architecture is implemented only as a baseline model for comparative evaluation. The CNN-Transformer model consists of a one-dimensional convolutional feature extractor followed by two Transformer encoder blocks and a fully connected classification layer. To assess robustness, each model was trained across multiple random initialisations, and the mean performance values were reported. Comparative fairness was maintained by keeping all experimental factors, other than the model architecture, constant throughout the study.

Model performance was quantified on a held-out test set using the following metrics computed from the confusion matrix. (Agarwal et al., 2021; Han et al., 2020; Verma et al., 2022) shown in Table 1. The proposed architecture is a CNN-Transformer model and does not include any recurrent layers such as LSTM. The LSTM network is used only as a baseline model for comparative evaluation.

## 4. EXPERIMENTAL VALIDATION

### 4.1 Tennessee Eastman Process (TEP) Case Study

The standard benchmark, the Tennessee Eastman Process (TEP), is used for evaluating fault detection and diagnosis methods in chemical processes. (Downs and Vogel, 1993; Reinartz et al., 2021; Yin et al., 2012). It represents a nonlinear multivariable plant with strong coupling between the process units and control loops.

The process consists of a compressor, condenser, reactor, separator, and stripper. It provides 52 variables, including 11 manipulated variables (XMV) and 41 measured variables (XMEAS), covering flows, temperatures, pressures, levels, and compositions throughout the plant. These variables

| Metric | Formula | Description / Interpretation |
|---|---|---|
| **Accuracy** | $\left(\frac{TP+TN}{TP+TN+FP+FN}\right)$ | Indicates the model's overall correctness by reporting the proportion of samples correctly classified. |
| **Precision** | $\left(\frac{TP}{TP+FP}\right)$ | Represents the proportion of predicted positive instances that are correctly identified as positive, reflecting the reliability of optimistic predictions. |
| **Recall (Sensitivity)** | $\left(\frac{TP}{TP+FN}\right)$ | Fraction of actual positives correctly identified (measures ability to capture positives). |
| **F1-Score** | $\left(2\frac{Precision \times Recall}{Precision+Recall}\right)$ | Quantifies the harmonic mean of precision and recall, offering a balanced evaluation that simultaneously considers false positives and false negatives. |

Table 1. Four standard classification performance metrics and their equations in mathematical form

Legend: TP: True Positives, TN: True Negatives, FP: False Positives, FN: False Negatives

reflect the realistic plant-wide dynamics. Table S2 shows key measured (XMEAS) and manipulated (XMV) variables in the Tennessee Eastman Process.

In the standard Tennessee Eastman benchmark, 21 predefined fault scenarios are provided. However, faults IDV 3, 9, and 15 are widely recognised as incipient disturbances with statistical signatures very similar to normal operating conditions, making them extremely difficult to detect reliably using data-driven methods. Following established benchmarking practices reported in prior studies (Reinartz et al., 2021; Yin et al., 2012)These three faults were excluded from the classification task. Consequently, the dataset used in this study consists of 18 detectable fault categories, including the normal operating condition, resulting in a 19-class fault diagnosis problem. This formulation ensures consistency with commonly used TEP evaluation protocols and enables meaningful comparison with previously reported deep learning-based fault-diagnosis studies. Table S1 shows a summary of fault scenarios in the Tennessee Eastman Process.

Each simulation run contained 500 samples recorded at a 3-minute sampling interval, corresponding to approximately 25h of operation. Faults were introduced after sample index 160, which corresponds to approximately 8h of regular operation. This structure allows the models to learn both healthy and faulty process dynamics within each run.

To construct time-series inputs for the deep learning models, a sliding-window strategy was applied to the multivariate process sequences. Using the full simulation run as the base sequence ensures that the temporal evolution of the process dynamics is preserved before window segmentation. Industrial faults often develop gradually, and important diagnostic information may appear across extended time horizons rather than in isolated measurements. By first maintaining the complete run sequence and then extracting sliding windows, the model can learn both early fault signatures and long-term temporal dependencies while maintaining continuity of the process dynamics. Each input sample consists of a look-back window of 50 consecutive time steps, corresponding to approximately 150 minutes of process operation at a 3-minute sampling interval. The window is moved along the sequence with a stride of 1 to generate overlapping temporal segments while preserving process continuity. Each resulting input sample therefore has a dimension of 50 × 52, representing 50 time steps of the 52 measured and manipulated process variables.

Because the Tennessee Eastman benchmark generates a comparable number of simulations runs for each operating condition, the dataset remains approximately balanced across classes. Each fault scenario contributes a similar number of time-series segments after sliding-window generation, ensuring that no single class dominates the training dataset. This balanced distribution is advantageous for multi-class fault diagnosis because it reduces bias toward frequently occurring faults and allows macro-averaged metrics to reflect true class-level performance.

To prevent label leakage, dataset partitioning was performed at the simulation-run level before window generation. Entire simulation runs were assigned to either the training, validation, or test set, ensuring that windows extracted from a particular run were not distributed across different subsets. In addition, windows that overlapped the fault injection boundary were labelled according to the dominant class within the window. This protocol prevents information leakage between training and evaluation data and ensures that

the reported results reflect genuine generalization performance.

### 4.2 Data Preparation

All simulations were processed using a unified preprocessing pipeline.

Fault IDs {3, 9, 15}, which are widely recognised as incipient disturbances with weak statistical signatures, were removed before the analysis. All variables were normalised using z-score scaling, with the standard deviation and mean computed from the training set only. At the simulation-run level, 80% of the simulation runs were assigned to the training set, 10% to the validation set, and the remaining 10% to the test set.

To clarify the scale of the dataset used in the experiments, the processed data matrices used for training and testing are summarised as follows. The fault-free dataset consists of 250,000 × 55 samples for training and 480,000 × 55 samples for testing, while the faulty dataset consists of 5,000,000 × 55 samples for training and 9,600,000 × 55 samples for testing. Each row corresponds to a time-step observation containing the multivariate process measurements and the associated class label. This large-scale dataset provides extensive coverage of both normal operating conditions and multiple fault scenarios, enabling robust training and reliable evaluation of the deep learning models.

All simulation runs were kept within a single subset to preserve temporal continuity and prevent data leakage. Each run was treated as a 500-sample sequence with a stride of 1. Samples recorded before index 160 were labelled as regular operation, whereas samples from index 160 onward were labelled according to the active fault. This labelling follows the known fault injection point defined in the TEP simulator. Both the LSTM and CNN-Transformer models were trained on the same normalised and consistently labelled dataset. The architecture used in this study follows a hybrid convolution–attention design and does not include any recurrent layers. The model pipeline can be summarised as:

1D Convolutional Feature Extraction → Transformer Encoder Block 1 → Transformer Encoder Block 2 → Fully Connected Layer → Softmax Classification

The convolutional stage captures local cross-variable interactions in the multivariate process signals, while the Transformer encoder models long-range temporal dependencies using multi-head self-attention. No LSTM layers are used in the proposed architecture.

### 4.3 Model Training Setup

All experiments were conducted using MATLAB R2023a. The baseline model is a stacked LSTM network with three recurrent layers of 52, 40, and 25 hidden units, respectively, followed by a fully connected layer and a softmax classifier. A 20% dropout rate was applied to mitigate overfitting.

The proposed CNN-Transformer architecture comprises a one-dimensional convolutional layer with 64 filters and a kernel size of 3, followed by two Transformer encoder blocks composed of multi-head self-attention and position-wise feed-forward networks. The convolutional layer extracts local correlations among process variables, while the Transformer encoder captures long-range temporal relationships across the time-series sequence.

The LSTM and CNN-Transformer were trained for 30 epochs each, with early stopping based on validation performance. In addition to the architectural differences, all training settings were kept identical to ensure a fair comparison.

The model's performance was evaluated on the held-out test set using Accuracy, Precision, Recall, and Macro-averaged F1-score across all 19 classes. Macro-averaging was used to avoid biases toward easily detectable faults. Confusion matrices were also analyzed to assess the fault-wise misclassifications and class separability.

Both models were trained using the Adam optimiser with an initial learning rate of 0.001. The mini-batch size was set to 64, and categorical cross-entropy was used as the loss function for multi-class classification. A dropout rate of 0.2 was applied in the LSTM baseline to reduce overfitting, while the Transformer encoder utilized internal attention dropout as implemented in the MATLAB deep learning framework. Early stopping was applied based on validation loss to prevent overfitting, and the model that achieved the lowest validation loss was selected for final evaluation. All experiments were implemented in MATLAB R2023a using the Deep Learning Toolbox and executed on a workstation equipped with GPU acceleration. The same training configuration was used for both models to ensure a fair architectural comparison.

The Tennessee Eastman Process dataset used in this study is publicly available and widely used as a benchmark for fault detection and diagnosis research. The dataset generation procedure and simulation configuration follow the standard formulation originally introduced by Downs and Vogel (1993) and later extended in several benchmark studies. Because the dataset, preprocessing pipeline, and model training settings are fully described in this work, the experimental setup can be reproduced by other researchers

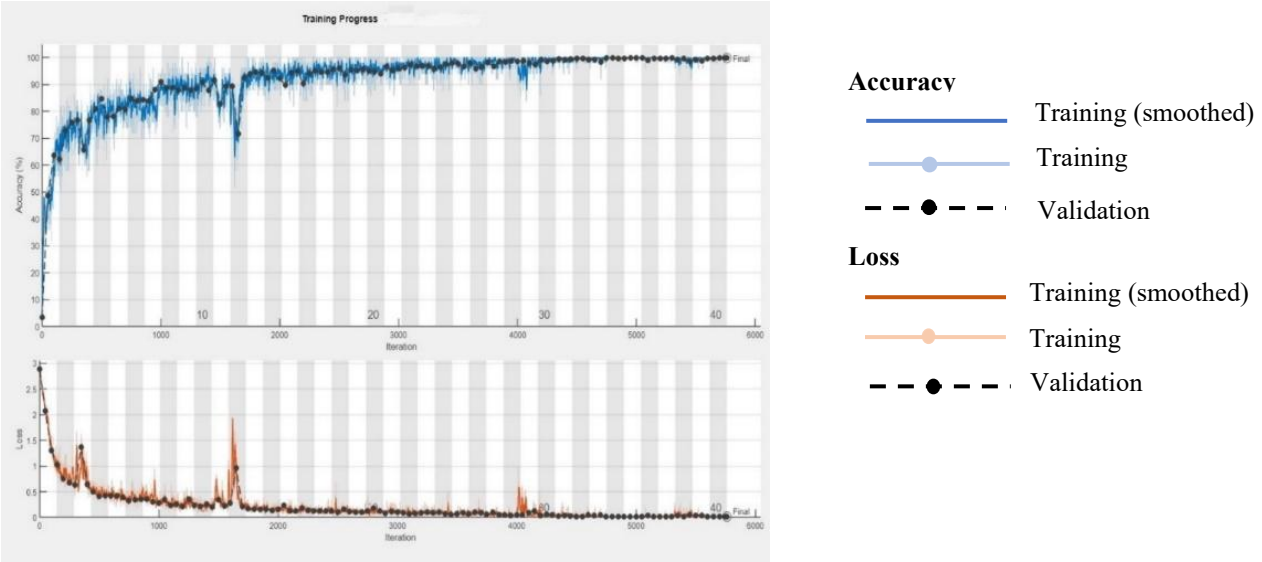


Figure 5. Training and validation accuracy and loss curves for the LSTM baseline model during training

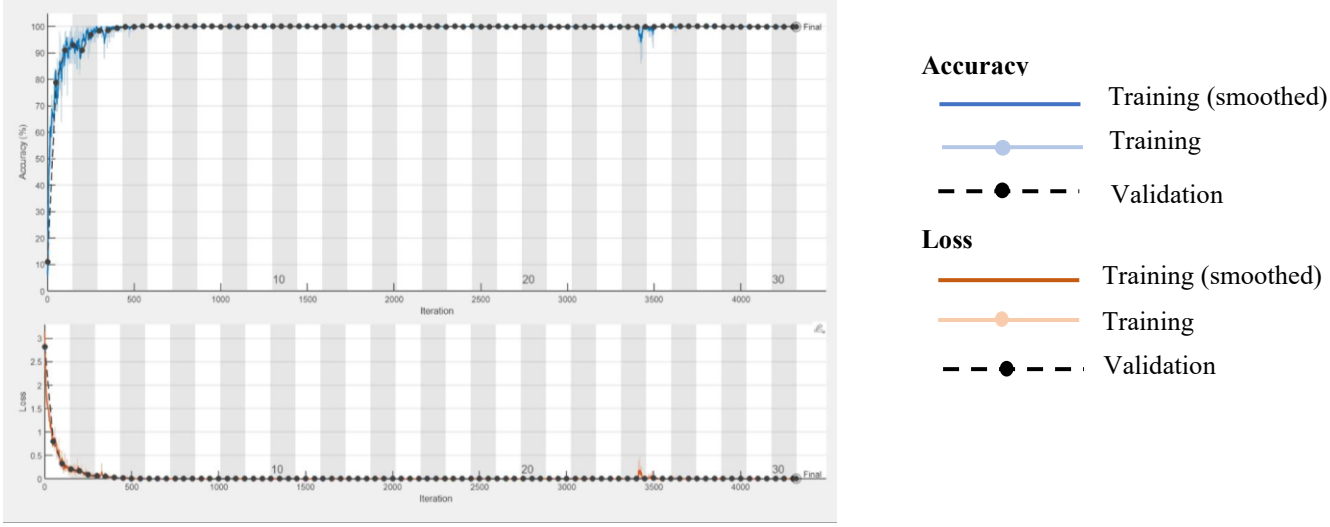


Figure 6. Training and validation accuracy and loss curves for the proposed CNN-Transformer model during training

using the same benchmark data and implementation framework.

## 5. EXPERIMENTAL RESULTS AND DISCUSSION

### 5.1 Model Convergence and Training Behaviour

This section presents the experimental results obtained using the baseline LSTM model and the proposed CNN-Transformer framework on the Tennessee Eastman Process (TEP) dataset. Figures 5 and 6 illustrate the training and validation behaviours of the LSTM and CNN-Transformer models in terms of classification accuracy and loss over epochs. Both architectures exhibited rapid and stable convergence, indicating effective optimization and good generalization.

For both models, learning progressed quickly during the initial training phase. The accuracy increases sharply, and the loss decreases substantially within the first 10-12 epochs, showing that the dominant fault patterns are captured early without unstable initialisation. As training proceeds, performance improvements become more gradual, reflecting the refinement of decision boundaries rather than the relearning of coarse features.

Both models exhibit smooth convergence during training, with the validation loss closely tracking the training loss, indicating effective regularisation and minimal overfitting.

The CNN-Transformer demonstrates comparable training stability to the LSTM model despite its use of self-attention mechanisms, confirming that the proposed architecture can be trained reliably under standard optimization settings.

The strong generalization performance observed on the unseen test set further confirms that the CNN-Transformer does not achieve improved accuracy through overfitting, but rather through enhanced feature representation and temporal modelling.

The smooth convergence behaviour and close alignment between training and validation curves in Figure 5 and Figure 6 indicate stable optimization dynamics and confirm that both models generalise well without significant overfitting.

### 5.2 Overall Fault Classification Performance

Table 2 summarizes the overall fault diagnosis performance of the LSTM and CNN-Transformer models on the unseen test dataset. Both models achieve very high classification accuracy, reflecting the maturity of deep learning approaches for the TEP benchmark.

The LSTM model achieves an overall accuracy of 99.86%, consistent with previously reported near-saturation results for well-tuned recurrent architectures on the TEP dataset (Agarwal et al., 2022; Verma et al., 2022). The proposed CNN-Transformer model achieves a slightly higher accuracy of **99.92%**, corresponding to an absolute improvement of **0.06%**.

While the absolute accuracy improvement is small, performance gains at this level are increasingly difficult to achieve, given the near-saturation performance already achieved by modern deep learning models on the TEP benchmark. Therefore, even modest improvements often reflect improved handling of the most challenging fault scenarios rather than simple scaling of model capacity. The LSTM model's overall macro-average accuracy is 0.9986, whereas the CNN-Transformer achieves 0.9992. The corresponding macro-averaged F1-scores are also 0.9986 and 0.9992, respectively.

### 5.3 Macro-Averaged Performance Metrics and Statistical Robustness

To provide a balanced evaluation across all fault categories, macro-averaged precision, recall, and F1-score are also reported. These metrics assign equal importance to each class and are particularly relevant for fault-diagnosis problems in which specific faults exhibit subtle or overlapping characteristics.

The CNN-Transformer model consistently achieves higher macro-averaged F1-scores than the LSTM baseline, indicating improved class-wise balance. This suggests that the CNN-Transformer reduces bias toward easily detectable faults and improves discrimination for more challenging fault classes. Similar observations have been reported in recent studies emphasising the importance of macro-level evaluation for multi-fault industrial diagnostics. (Chen et al., 2023; Zhang et al., 2021, 2022).

To assess reliability, both the LSTM and CNN-Transformer models were trained and tested with five different random initialisations. Across these runs, the standard deviations of accuracy, precision, recall, and F1-score stayed below 0.03% for both models. This shows that performance is stable and not sensitive to random initialisation or mini-batch sampling.

Table 3 reports the mean and standard deviation of sample-level performance metrics across the five runs. The low variance across all metrics confirms that the results are reproducible. The slightly higher mean values achieved by the CNN-Transformer indicate a consistent, though slight, performance advantage.

The CNN-Transformer performs better on more challenging faults involving noise, reaction-kinetics drift, and unknown disturbances, while matching the LSTM on simpler cases. Overall, these results suggest that the CNN-Transformer offers a modest yet consistent robustness benefit without sacrificing stability, making it well-suited for high-performance fault-diagnosis tasks.

### 5.4 Visualization of Fault Classification

Figures 7 and Figure 8 present the confusion matrices for the LSTM and CNN-Transformer models. Both models achieve strong class separation with minimal cross-class misclassification. However, the CNN-Transformer further reduces several minor errors observed in the LSTM results, particularly for faults exhibiting overlapping sensor patterns.

For the LSTM, two minor off-diagonal errors appear. A few samples from IDV 13 are predicted as IDV 18, and a few samples from IDV 18 appear under IDV 8. These faults exhibit similar sensor response patterns, which explains the observed misclassification.

The CNN-Transformer further reduces these errors. It shows only two minor mistakes: a single sample from IDV 12 was misclassified as IDV 5, and six samples from IDV 13 were misclassified as IDV 16. The convolutional layer helps capture local variations in temperature, pressure, and flow sensor readings before the Transformer encoder models the global temporal relationships in the time-series data. This

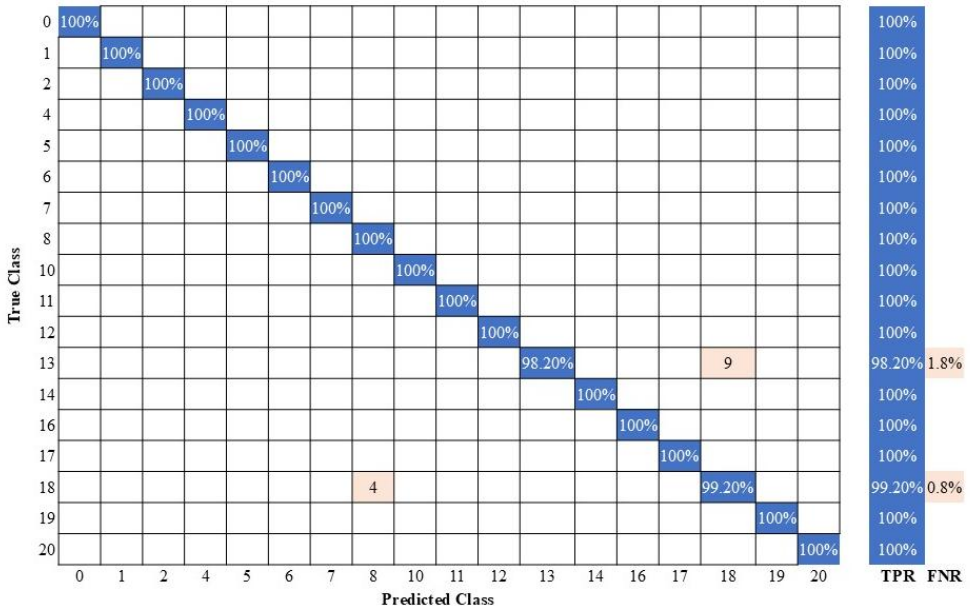


Figure 7. Confusion matrix for the LSTM baseline model showing fault-wise classification performance across the 19 operating conditions

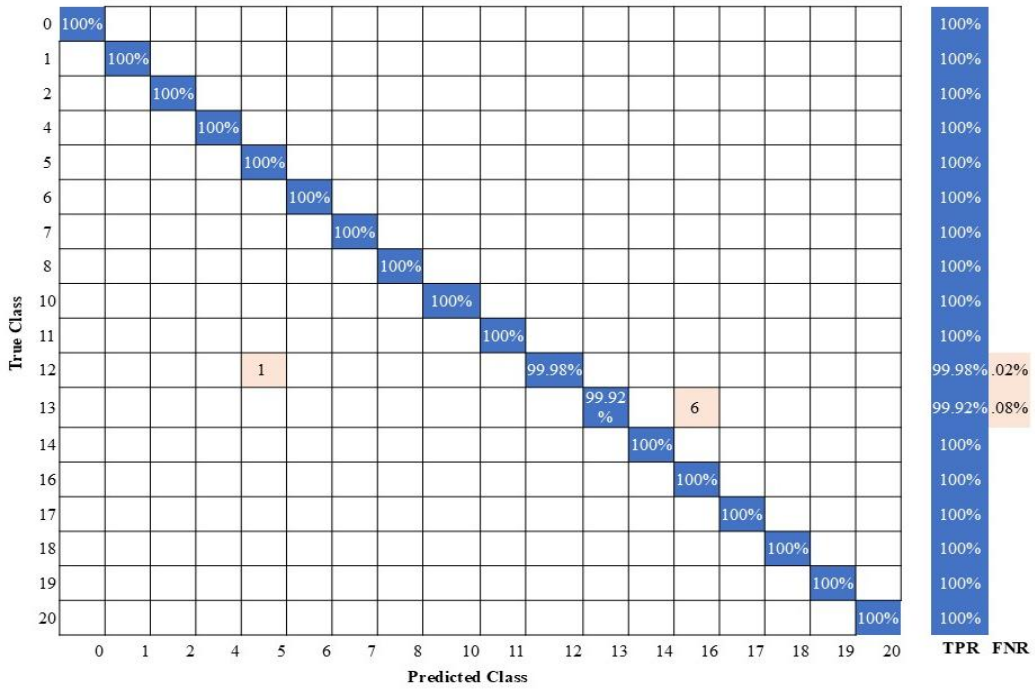


Figure 8. Confusion matrix for the proposed CNN-Transformer model showing improved class separation and reduced misclassification for challenging fault scenarios

Table 2. Fault-wise classification performance of LSTM and CNN-Transformer models on the TEP dataset

| Fault ID | Fault Description (Brief) | LSTM Accuracy | CNN-Transformer Accuracy | LSTM Precision | CNN-Transformer Precision | LSTM Recall | CNN-Transformer Recall | LSTM F1-Score | CNN-Transformer F1-Score |
|---|---|---|---|---|---|---|---|---|---|
| 0 | Normal | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 1 | A/C Feed Ratio | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 2 | B Composition | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 4 | Reactor Cooling | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 5 | Condenser Cooling | 1.000 | 0.9999 | 0.998 | 1 | 1 | 1 | 1.000 | 0.999 |
| 6 | Feed Loss | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 7 | Header Pressure Loss | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 8 | Feed Composition Noise | 0.9996 | 1 | 1 | 1 | 1 | 1 | 0.9964 | 1 |
| 10 | C Feed Temperature Noise | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 11 | Reactor CW Temperature Noise | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 12 | Condenser CW Temperature Noise | 1.000 | 0.9999 | 1 | 1 | 1 | 0.998 | 1.000 | 0.999 |
| 13 | Reaction Kinetics Drift | 0.9999 | 0.9993 | 1 | 0.994 | 0.9824 | 0.988 | 0.9909 | 0.994 |
| 14 | CW Valve Stiction | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 16 | Unknown Disturbance 1 | 1.000 | 0.9993 | 0.9881 | 1 | 1 | 1 | 1.000 | 0.994 |
| 17 | Unknown Disturbance 2 | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 18 | Unknown Disturbance 3 | 0.9986 | 1 | 1 | 1 | 0.9923 | 1 | 0.9871 | 1 |
| 19 | Unknown Disturbance 4 | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |
| 20 | Unknown Disturbance 5 | 1.000 | 1 | 1 | 1 | 1 | 1 | 1.000 | 1 |

Table 3. Mean ± standard deviation of sample-level performance metrics across five independent runs

| Metric | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| LSTM | 0.9987 ± 0.0002 | 0.9985 ± 0.0003 | 0.9989 ± 0.0003 | 0.9991 ± 0.0003 |
| CNN-Transformer | 0.9993 ± 0.0001 | 0.9994 ± 0.0002 | 0.9991 ± 0.0002 | 0.9992 ± 0.0002 |

improves class separation, consistent with observations in (Han et al., 2020; Verma et al., 2022).

These results indicate that most remaining misclassifications occur between faults with similar dynamic behaviour rather than from systematic model bias.

The confusion matrices, therefore, provide visual confirmation that the CNN-Transformer improves class separability for difficult fault categories while maintaining near-perfect classification for easily detectable faults.

The fault scenarios that benefit most from the proposed CNN-Transformer framework include reaction-kinetics drift (IDV 13), random disturbances, and unknown composite faults. These faults exhibit gradual or distributed effects across multiple units, such as the reactor, separator, and condenser, rather than abrupt step changes in a single variable.

### 5.5 Comparative Positioning Within Literature

Table 4 and Table 5 compare the proposed approach with previously reported fault diagnosis methods for the Tennessee Eastman benchmark. Although direct numerical comparison is difficult due to differences in dataset partitions and experimental protocols, the results demonstrate that the proposed CNN-Transformer architecture achieves competitive or superior performance relative to classical statistical methods and recent deep learning models. (Chen et al., 2023; Chiang et al., 2001; Han et al., 2020; Heo and Lee, 2019, 2018; Yin et al., 2012).

## 6 Discussion of Performance Improvement

The improvement achieved by the CNN-Transformer architecture should be interpreted in the context of near-saturation performance on the TEP benchmark, where many modern deep learning models already achieve accuracies above 98% (Chen et al., 2022).

Although the numerical improvement from 99.86% to 99.92% may appear modest, such gains are non-trivial in the context of the TEP benchmark, where diagnostic accuracy has already approached saturation. At this stage, improvements typically reflect better discrimination of the most challenging fault scenarios rather than general performance scaling.

The results suggest that combining convolutional feature extraction with Transformer-based self-attention improves the model's ability to capture both local sensor interactions and long-range temporal dependencies. This combination provides a consistent robustness advantage while maintaining a relatively compact architecture suitable for practical industrial monitoring applications**.**

The proposed CNN-Transformer achieves improved diagnostic robustness with only a marginal increase in model complexity compared with the stacked LSTM baseline (45,010 vs 43,788 parameters). The model complexity comparison is shown in Table 6 for both architectures.

Although the CNN-Transformer architecture contains additional functional blocks compared with the stacked LSTM baseline, the overall model complexity remains comparable. The stacked LSTM model contains 43,788 trainable parameters, whereas the proposed CNN-Transformer contains 45,010 trainable parameters, an increase of only about 2.8%. This small difference indicates that the additional convolutional and attention components do not substantially increase the model's overall parameter footprint.

It is also important to note that architectural depth alone does not necessarily imply higher computational complexity. The Transformer encoder enables parallel processing of sequence elements through self-attention, whereas LSTM networks process time steps sequentially. As a result, Transformer-based architectures can achieve efficient training and inference despite having multiple internal blocks.

Furthermore, the proposed framework intentionally uses only a single convolutional layer and two Transformer encoder blocks, which are significantly smaller than those in many recent deep learning architectures used for industrial time-series modelling. This design maintains a compact model structure while still allowing the network to capture both local sensor correlations and long-range temporal dependencies.

Table 4. Accuracy-based comparison with literature

| Method | Model Type | Metric | Reported Performance | Source |
|---|---|---|---|---|
| PCA | Linear Statistical | Detection Rate | 61.77% ($T^2$), 74.72% (Q) | (Yin et al., 2012) |
| DPCA + SVM | Linear + Classifier | Detection Rate | 72.35% | (Agarwal et al., 2021, 2020) |
| ICA-SC | Independent Component | Avg Accuracy | ≈90% | (Zhang et al., 2021) |
| DSAE | Deep Autoencoder | Accuracy | 91.55–93.23% | (Agarwal et al., 2021, 2020) |
| DDSAE (lag1) | Dynamic Supervised AE | Accuracy | 93.96–95.85% | (Agarwal et al., 2021, 2020) |
| DDSAE (lag2) | Dynamic Supervised AE | Accuracy | 93.50–96.43% | (Agarwal et al., 2021, 2020) |
| DLSAE Diagnosis | Supervised AE | Accuracy | ≈88.41% | (Agarwal et al., 2021, 2020) |
| **Proposed LSTM** | Recurrent DL | **Accuracy** | **99.86%** | This work |
| **Proposed CNN-Transformer** | Hybrid DL | **Accuracy** | **99.92%** | This work |

Table 5. F1-Score / Detection-metric-based comparison with literature

| Method | Model Type | Metric | Reported Performance | Source |
|---|---|---|---|---|
| CNN (Chadha & Schwung) | CNN DL | F1-score (per fault) | 50–100% | (Chadha and Schwung, 2017) |
| Temporal CNN1D2D (GAN) | CNN + GAN | TPR / FPR | Severe faults: 100% TPR, 0% FPR (Not Specified Correctly) | (Lomov et al., 2021) |
| Nonlinear SVM | Kernel ML | Avg F1 | ≈78% | (Onel et al., 2019) |
| Autoencoder–Attention–LSTM | Hybrid DL | F1-score | ≈76–90% | (Li and Shao, 2021) |
| Twin Transformer (all faults) | Transformer + Attention | Avg F1 | 94% | (Labbaf-Khaniki et al., 2024.) |
| Twin Transformer (no incipient faults) | Transformer + Attention | Avg F1 | 97% | (Labbaf-Khaniki et al., 2024.) |
| **Proposed LSTM** | Recurrent DL | **Macro F1** | **>0.998** | This work |
| **Proposed CNN-Transformer** | Hybrid DL | **Macro F1** | **>0.999** | This work |

These observations support the statement that the proposed CNN-Transformer architecture improves diagnostic robustness while maintaining a lightweight architecture that remains feasible for practical industrial monitoring applications.

Furthermore, when benchmark performance approaches saturation levels, improvements in overall accuracy alone may not fully reflect the practical benefit of a model. In such cases, improvements in class-level robustness and reduction of specific fault misclassifications become more meaningful indicators of model effectiveness. The confusion matrix analysis shows that the CNN-Transformer reduces several minor misclassification cases observed in the LSTM model, particularly for faults with overlapping sensor signatures. This indicates that the proposed architecture provides improved discrimination for difficult fault scenarios rather than merely increasing overall accuracy.

In addition to the parameter comparison, the compactness of the proposed architecture can also be interpreted in terms of structural depth. The CNN-Transformer uses only a single convolutional layer and two Transformer encoder blocks, which is significantly smaller than many recent deep learning architectures for industrial time-series modelling that employ deeper convolutional stacks or multiple attention layers. Because the parameter count differs by only about 2.8% from the stacked LSTM baseline, the proposed model maintains a lightweight parameter footprint.

Table 6: Model Complexity Comparison

| Model | Layers | Trainable Parameters |
|---|---|---|
| LSTM baseline | 3 LSTM + FC | **43,788** |
| CNN-Transformer | CNN + 2 Transformer blocks + FC | **45,010** |

Furthermore, Transformer-based architectures enable parallel processing of sequence elements, reducing inference latency compared with sequential recurrent networks such as LSTMs. These characteristics support the claim that the proposed architecture remains computationally feasible for real-time monitoring environments.

## 6. CONCLUSION

This work demonstrates that integrating convolutional feature extraction with attention-based temporal modelling provides a consistent robustness advantage over near-saturated recurrent architectures for fault diagnosis in complex chemical processes.

A hybrid CNN-Transformer architecture was proposed that integrates convolutional feature extraction with Transformer-based self-attention to jointly model local sensor correlations and long-range temporal dependencies. Experimental results demonstrate that the proposed approach achieves 99.92% classification accuracy, marginally outperforming a strong LSTM baseline (99.86%). Although the absolute accuracy improvement is slight, the CNN-Transformer consistently exhibits higher macro-averaged F1-Scores and reduced fault-wise misclassification, indicating improved robustness and class-wise balance.

The findings indicate that self-attention-based temporal modelling, when combined with convolutional feature extraction, provides a systematic advantage in distinguishing challenging fault scenarios that exhibit subtle or delayed dynamic signatures. Importantly, this improvement is achieved without introducing excessive architectural complexity, supporting the feasibility of the proposed framework for practical deployment in industrial monitoring applications.

The results highlight that even modest improvements in fault classification accuracy at this stage of benchmark maturity reflect better handling of difficult fault scenarios rather than simple performance scaling.

Although the proposed CNN-Transformer architecture demonstrates strong performance on the Tennessee Eastman benchmark, several limitations should be noted. First, the evaluation was conducted using an offline benchmark dataset, and the model's behaviour under real-time industrial deployment conditions was not investigated. Second, the study focuses on a single benchmark process; therefore, the architecture's generalization capability across different industrial processes and operating regimes requires further validation. In addition, while the model maintains a compact parameter size, attention-based architectures may introduce additional computational overhead compared with simpler recurrent models when applied to long time-series sequences.

Future work will focus on extending the proposed approach to online fault diagnosis, investigating lightweight attention mechanisms to further reduce computational overhead, and evaluating generalization performance across different operating modes and industrial processes. These directions

will help advance the practical applicability of attention-based deep learning models for reliable fault diagnosis in complex chemical systems.

Future work will also explore interpretability techniques such as attention visualization and gradient-based saliency analysis to identify which sensor variables and temporal segments contribute most strongly to fault predictions, thereby improving transparency and trust for industrial process engineers.

## REFERENCES


Agarwal, P., Gonzalez, J.I.M., Elkamel, A., Budman, H., 2022. Hierarchical Deep LSTM for Fault Detection and Diagnosis for a Chemical Process. Processes 10. https://doi.org/10.3390/pr10122557

Agarwal, P., Tamer, M., Budman, H., 2021, Explainability: relevance based dynamic deep learning algorithm for fault detection and diagnosis in chemical processes. Comput. Chem. Eng. 154, 107467. https://doi.org/10.1016/j.compchemeng.2021.107467

Agarwal, P., Tamer, M., Budman, H., 2020. Assessing observability using supervised autoencoders with application to Tennessee eastman process, in: IFAC-PapersOnLine. Elsevier B.V., pp. 206–211. https://doi.org/10.1016/j.ifacol.2020.12.122

Chadha, G.S., Schwung, A., 2017. Comparison of deep neural network architectures for fault detection in Tennessee Eastman process, in: 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA). Presented at the 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, Limassol, pp. 1–8. https://doi.org/10.1109/ETFA.2017.8247619

Chen, B., Zhang, H., He, F., Zhang, C., Chen, Y., Liao, H., Zheng, S., 2023. Modified Q-$\sigma$ Rule for Incipient Fault Detection in Industrial Processes on Analysis of Intermittent Process Variation. SSRN Electronic Journal null, null. https://doi.org/10.2139/ssrn.4375935

Chen, H., Cen, J., Yang, Z., Si, W., Cheng, H., 2022. Fault Diagnosis of the Dynamic Chemical Process Based on the Optimized CNN-LSTM Network. ACS Omega 7, 34389–34400. https://doi.org/10.1021/acsomega.2c04017

Chiang, L.H., Russell, E.L., Braatz, R.D., 2001. Fault Detection and Diagnosis in Industrial Systems, Advanced Textbooks in Control and Signal Processing. Springer London, London. https://doi.org/10.1007/978-1-4471-0347-9

Downs, J.J., Vogel, E.F., 1993 A plant-wide industrial process control problem Comput. Chem. Eng. 17 (3), 245–255 https://doi.org/10.1016/0098-1354(93)80018-I

Han, Y., Ding, N., Geng, Z., Wang, Z., Chu, C., 2020. An optimized long short-term memory network based fault diagnosis model for chemical processes. Journal of Process Control 92, 161–168. https://doi.org/10.1016/j.jprocont.2020.06.005

Heo, S., Lee, J.H., 2019. Statistical process monitoring of the Tennessee Eastman process using parallel autoassociative neural networks and a large dataset. Processes 7. https://doi.org/10.3390/pr7070411

Heo, S., Lee, J.H., 2018. Fault detection and classification using artificial neural networks. IFAC-PapersOnLine 51, 470–475. https://doi.org/10.1016/j.ifacol.2018.09.380

Labbaf-Khaniki, M.A., Manthouri, M., Ajami, H., 2024 Twin Transformer using Gated Dynamic Learnable Attention mechanism for Fault Detection and Diagnosis in the Tennessee Eastman Process. https://doi.org/10.48550/arXiv.2403.10842

Li, M., Shao, Y., 2021. Deep Compression of Neural Networks for Fault Detection on Tennessee Eastman Chemical Processes.

Lomov, I., Lyubimov, M., Makarov, I., Zhukov, L.E., 2021. Fault detection in Tennessee Eastman process with temporal deep learning models. Journal of Industrial Information Integration 23, 100216. https://doi.org/10.1016/j.jii.2021.100216

Onel, M., Kieslich, C.A., Pistikopoulos, E.N., 2019. A nonlinear support vector machine-based feature selection approach for fault detection and diagnosis: Application to the Tennessee Eastman process. AIChE Journal 65, 992–1005. https://doi.org/10.1002/aic.16497

Pal, P.K., Hens, A., Behera, N., Lahiri, S.K., 2025. Digital twins: Transforming the chemical process industry—A review. Can J Chem Eng 103, 3611–3636. https://doi.org/10.1002/cjce.25611

Pozdnyakov, V., Kovalenko, A., Makarov, I., Drobyshevskiy, M., Lukyanov, K., 2024. Adversarial Attacks and Defenses in Fault Detection and Diagnosis: A Comprehensive Benchmark on the Tennessee Eastman Process. IEEE Open J. Ind. Electron. Soc. 5, 428–440. https://doi.org/10.1109/OJIES.2024.3401396

Reinartz, C., Kulahci, M., Ravn, O., 2021. An extended Tennessee Eastman simulation dataset for fault-detection and decision support systems. Computers and Chemical Engineering 149. https://doi.org/10.1016/j.compchemeng.2021.107281

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017 Attention is All you Need.

Verma, R., Yerolla, R., Besta, C.S., 2022. Deep Learning-based Fault Detection in the Tennessee Eastman Process, in: Proceedings of the 2nd International Conference on Artificial Intelligence and Smart Energy, ICAIS 2022. Institute of Electrical and Electronics Engineers Inc., pp. 228–233. https://doi.org/10.1109/ICAIS53314.2022.9743021

Yin, S., Ding, S.X., Haghani, A., Hao, H., Zhang, P., 2012. A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process. Journal of Process Control 22, 1567–1581. https://doi.org/10.1016/j.jprocont.2012.06.009

Yin, S., Gao, X., Karimi, H.R., Zhu, X., 2014. Study on Support Vector Machine-Based Fault Detection in Tennessee Eastman Process. Abstract and Applied Analysis 2014, 1–8. https://doi.org/10.1155/2014/836895

Zhang, C., Zheng, X., Li, Y., 2021. A novel fault detection and diagnosis scheme based on independent component analysis-statistical characteristics: Application on the tennessee eastman benchmark process. Journal of Chemical Engineering of Japan 54, 304–312. https://doi.org/10.1252/jcej.20we045

Zhang, L., Song, Z., Zhang, Q., Peng, Z., 2022. Generalized transformer in fault diagnosis of Tennessee Eastman process. Neural Comput & Applic 34, 8575–8585. https://doi.org/10.1007/s00521-021-06711-2

## BIOGRAPHIES

**A. Pratyush** is currently a Sr. Technical Officer at CSIR-CMERI, Durgapur, India. He did his M.Sc. in Computer Science and B.Sc. in Computer Application, and is Currently Pursuing a PhD.

**B. Sumona** completed her M.Tech in Microelectronics and VLSI and is currently pursuing a PhD.

**C. Narottam** is a metallurgical engineering professional with over 24 years of experience in steel industry Operations, R&D, quality assurance, and major project execution. He holds Bachelor's and Master's degrees in Metallurgical Engineering from NIT Warangal and a Master's degree from IIT (BHU) Varanasi, and is currently pursuing a PhD.

**D. Somnath** is currently working as a Technical Services Engineer at Scientific Design LLC in Dubai, United Arab Emirates. He did his M.Tech in Chemical Engineering and is currently pursuing a PhD

**E. Abhiram** is currently a faculty member in the Department of Chemical Engineering of the National Institute of Technology Durgapur, India. He did his B.Tech. and M.Tech. in Chemical Engineering before completing a PhD in Engineering Science. His research area includes process modelling and simulation using computational fluid dynamics and molecular dynamics. Recently, he began working in AI/ML-based process optimization**.**

**F. Sandip** is currently a faculty member in the Department of Chemical Engineering of the National Institute of Technology Durgapur, India. He did his M.Tech in Chemical Engineering and PhD from NIT Durgapur**.** He was awarded the title of Innovation Ambassador by the Ministry of Education, Government of India.

**Supplementary Information**

This supplement provides additional reproducibility and robustness analyzes supporting the main manuscript.

**Table S1. Summary of fault scenarios in the Tennessee Eastman Process** (Faults 3, 9 and 15 are typically excluded from quantitative benchmarking because of their weak statistical signatures ( Downs and Vogel, 1993; Agarwal et al., 2021; Chiang et al., 2001; Reinartz et al., 2021).

| Fault ID | Process variable or unit affected | Fault type | Typical description | Detectability |
|---|---|---|---|---|
| 1 | A/C feed ratio, B composition constant (stream 4) | Step change | A change in the reactant feed ratio causes a composition imbalance | High |
| 2 | B composition, A/C ratio constant (stream 4) | Step change | Shift in B feed composition | High |
| 3 | D feed temperature (stream 2) | Step change | Slight change in feed temperature | Low (incipient) |
| 4 | Reactor cooling water inlet temperature | Step change | Cooling disturbance affecting reactor temperature control | High |
| 5 | Condenser cooling water inlet temperature | Step change | Cooling disturbance in the condenser loop | High |
| 6 | A feed loss (stream 1) | Step change | Reduction or loss of A feed stream | High |
| 7 | C header pressure loss (stream 4) | Step change | Reduced availability of C feed gas | High |
| 8 | A/B/C feed composition (stream 4) | Random variation | Fluctuating feed composition | Moderate |
| 9 | D feed temperature (stream 2) | Random variation | Small random thermal disturbance | Low (incipient) |
| 10 | C feed temperature (stream 4) | Random variation | Temperature fluctuations in the C feed | Moderate |
| 11 | Reactor cooling water inlet temperature | Random variation | Cooling water temperature fluctuation | Moderate |
| 12 | Condenser cooling water inlet temperature | Random variation | Variation in condensing temperature | Moderate |
| 13 | Reaction kinetics | Slow drift | Gradual catalyst deactivation or reaction-rate change | High |
| 14 | Reactor cooling water valve | Sticking fault | Actuator stiction affecting heat removal | Moderate |
| 15 | Condenser cooling water valve | Sticking fault | Valve stiction in the condensing loop | Low (incipient) |
| 16 | Unknown disturbance 1 | Unknown | Unspecified composite fault | Moderate |
| 17 | Unknown disturbance 2 | Unknown | Unspecified process disturbance | Moderate |

| | | | | |
|---|---|---|---|---|
| 18 | Unknown disturbance 3 | Unknown | Unspecified process disturbance | Moderate |
| 19 | Unknown disturbance 4 | Unknown | Unspecified process disturbance | Moderate |
| 20 | Unknown disturbance 5 | Unknown | Unspecified process disturbance | Moderate |
| 21 | Valve or instrument bias | Bias fault | Sensor drift or bias in measurement | High |

**Table S2. Key measured (XMEAS) and manipulated (XMV) variables in the Tennessee Eastman Process** ( Variable designations follow the original notation by (Downs and Vogel, 1993; Agarwal et al., 2021; Reinartz et al., 2021).

| **Variable ID** | **Description** | **Variable ID** | **Description** |
|---|---|---|---|
| XMEAS 1 | A feed (flow) | XMEAS 22 | Stripper steam flow |
| XMEAS 2 | D feed (flow) | XMEAS 23 | Stripper level |
| XMEAS 3 | E feed (flow) | XMEAS 24 | Stripper pressure |
| XMEAS 4 | A and C feed (flow) | XMEAS 25 | Stripper temperature |
| XMEAS 5 | Recycle flow | XMEAS 26 | Stripper steam temperature |
| XMEAS 6 | Reactor feed rate | XMEAS 27 | Condenser cooling water outlet temperature |
| XMEAS 7 | Reactor pressure | XMEAS 28 | Condenser level |
| XMEAS 8 | Reactor level | XMEAS 29 | Separator temperature |
| XMEAS 9 | Reactor temperature | XMEAS 30 | Separator level |
| XMEAS 10 | Purge rate | XMEAS 31 | Compressor work |
| XMEAS 11 | Product flow | XMEAS 32 | Product cooling water flow |
| XMEAS 12 | Separator pressure | XMEAS 33 | Product temperature |
| XMEAS 13 | Separator level | XMEAS 34 | Recycle temperature |
| XMEAS 14 | Separator temperature | XMEAS 35 | Cooling water inlet temperature |
| XMEAS 15 | Stripper pressure | XMEAS 36 | Cooling water outlet temperature |
| XMEAS 16 | Stripper level | XMEAS 37 | Process gas molecular weight |
| XMEAS 17 | Stripper temperature | XMEAS 38 | Purge gas composition (A) |
| XMEAS 18 | Compressor work | XMEAS 39 | Purge gas composition (B) |
| XMEAS 19 | Reactor cooling water outlet temperature | XMEAS 40 | Product composition (G) |
| XMEAS 20 | Condenser cooling water outlet temperature | XMEAS 41 | Product composition (H) |

| | | | |
|---|---|---|---|
| XMV 1 | D feed flow | XMV 7 | Purge rate |
| XMV 2 | E feed flow | XMV 8 | Compressor recycle valve |
| XMV 3 | A feed flow | XMV 9 | Condenser cooling water valve |
| XMV 4 | A and C feed flow | XMV 10 | Separator cooling water valve |
| XMV 5 | Recycle valve | XMV 11 | Stripper steam valve |
| XMV 6 | Reactor cooling water valve | | |