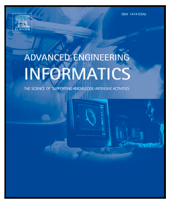
Advanced Engineering Informatics 73 (2026) 104592

Contents lists available at ScienceDirect

Advanced Engineering Informatics

journal homepage: www.elsevier.com/locate/aei



Full length article

A novel multi-task multi-view approach with custom multi-label loss for fault detection in complex industrial apparatus

Riccardo Rosati [a],*, Lucia Pepa [a], Luca Romeo [b]

[a] Department of Information Engineering, Marche Polytechnic University, Ancona, Italy
[b] Department of Economics and Law, University of Macerata, Macerata, Italy

ARTICLE INFO

Keywords:
Fault detection
Multi-task learning
Multi-view learning
Decision support systems

ABSTRACT

Machine Learning (ML) plays a crucial role in Industry 4.0, enabling predictive fault detection (FD) by analyzing vast amounts of log data. However, current ML approaches often rely on single-task learning, neglecting the diverse nature of log data and the prediction of interrelated faults. Moreover, multi-view learning (MVL) and multi-task learning (MTL) are usually applied disjointly without applying joint learning across tasks and views. To address these gaps, we a novel approach which leverages multi-task and multi-view learning frameworks, augmented by a multi-label Cross Entropy loss (MTMVL-CE). MTMVL-CE improves generalization performance between and within different fault types, enabling the classification of multiple faults in complex industrial machines. Indeed, MTMVL-CE optimizes classification performance by learning across numerous faults simultaneously, achieving an accurate representation of heterogeneous log data, robust fault classification, and feasible generalization over time. We tested our approach through extensive experiments on an real use case involving FD in a complex banknote recirculator device inside Automated Teller Machines. Our results demonstrate MTMVL-CE's superior performance compared to MTL and MVL competitors in capturing fault interdependencies and providing accurate, reliable predictions.

1. Introduction

In the era of Industry 4.0, the Internet of Things (IoT), big data analytics, Artificial Intelligence (AI), and cloud computing allow to create interconnected and intelligent production environments for manufacturing and industrial processes. This paradigm shift aims to enhance operational efficiency, flexibility, and productivity by integrating physical machines with digital systems through seamless data exchange and real-time monitoring. One critical challenge is ensuring the continuous and reliable operation of machinery, particularly as modern production systems become more complex and interconnected. In this context, fault detection (FD) plays a pivotal role as a key strategy for identifying and classifying the system malfunctions or abnormalities, thereby supporting effective fault diagnosis and system reliability. The increasing amount of data collected from IoT-enabled sensors and devices forms the backbone of this transformation [1]. Machine Learning (ML) approaches capitalize on this wealth of information, analyzing it to uncover hidden patterns, detect anomalies, and pinpoint fault causes [2,3].

However, a significant issue in many industrial environments is the inability to extract real-time data directly from sensors. In such situations, the available data often consist solely of logs that the machines generate. This limitation can be challenging for FD methods, which rely heavily on continuous sensor data [4,5]. ML approaches are well-suited to process these log data, extracting meaningful patterns and identifying discriminative features that can predict different faults. A crucial role is ensured by the efficient training of the model across different machines [6] and the integration of the ML models in a Decision Support System (DSS) that represents a fault diagnostic/prognostic platform [7], driving the shift from Industry 4.0 toward Industry 5.0 paradigm. However, two significant gaps in the current state-of-the-art lie in the lack of robust methodological approaches to address the challenges posed by the heterogeneity of log data and the occurrence of multiple simultaneous faults within industrial apparatuses [8,9].

Existing ML solutions for FD often focus on implementing a single-task learning (STL) approach and assume homogeneous data sources, typically processing a single input [10,11]. They fail to model the diverse nature of log data originating from different sources within the machinery. At the same time, they overlook the possibility of predicting multiple faults across these sources, which may exhibit interdependencies. In this context, we proposed Multi-Task Learning (MTL) and Multi-View Learning (MVL) to classify multiple faults represented by heterogeneous feature sets. This paper introduces an innovative

* Correspondence to: Faculty of Engineering, Via Brecce Bianche 12, 60131 Ancona, Italy.
E-mail address: r.rosati@staff.univpm.it (R. Rosati).

https://doi.org/10.1016/j.aei.2026.104592
Received 4 November 2025; Received in revised form 5 February 2026; Accepted 9 March 2026
Available online 11 March 2026

approach, MTMVL-CE, leveraging MTL and MVL learning frameworks, augmented by a multi-label Cross Entropy (CE) loss, to enhance the classification accuracy of multiple fault types within complex industrial devices. MTMVL-CE stands out by optimizing a loss function that simultaneously considers multiple faults, thereby maximizing classification performance across various fault types. The proposed approach achieved three main contributions: (i) accurate representation of heterogeneous log data, (ii) robust classification performance across multiple faults, and (iii) feasible generalization over time. To evaluate the performance of our proposed method, we conduct extensive experiments on a demanding Industry 4.0 use case, namely the FD task in Automated Teller Machines (ATMs). This use case poses significant challenges for maintenance personnel in identifying and addressing the exact location of the fault within the machine [12].

The paper is summarized as follows. Related Work section reviews MVL, MLC, and MTL strategies for FD approaches. Method section presents our MTMVL-CE approach. Experiments section details its application to a real ATM dataset, with findings analyzed in Results section. Finally, Conclusions section summarize key insights and implications.

2. Related work

In this section, we categorize the state-of-the-art work for FD into two threads: MVL approaches for incorporating different input views and ML/MTL approaches for classifying multiple faults.

2.1. Multi-view learning strategies for incorporating different input views in a FD scenario

Multi-View Learning (MVL) is a ML paradigm designed to leverage multiple heterogeneous data representations, referred to as views, to enhance learning performance. These views may correspond to different feature sets, sensor modalities, or data sources, each capturing complementary aspects of the underlying structure of the data. By integrating information from multiple perspectives, MVL aims to identify common discriminative patterns and improve generalization, leading to more robust and accurate models [13]. MVL methods typically involve techniques like canonical correlation analysis (CCA) and deep neural networks (DNNs) [14]. Recent advances in MVL have focused on developing general theoretical frameworks for adaptive view fusion [15], robustness to incomplete views [16], and semi-supervised learning scenarios [17]. These approaches primarily address problem settings that differ from our target scenario such as semi-supervised classification and clustering. In contrast, our work targets a supervised industrial fault detection scenario and proposes a unified framework that jointly integrates MVL and MTL, together with a custom ML loss, to explicitly model fault interdependencies from heterogeneous log-based data sources. Considering the multiple natures of machines data, the use of MVL in FD frameworks has recently shown significant advancement.

[18,19] have employed CCA or topic modeling to map extracted data from different sources into a correlated feature space. The extracted view-invariant and category-discriminative features have been used as input for a standard ML classifier. In this context, a hybrid ensemble approach [20] has also been utilized to integrate diverse features. The standard limitation of these approaches is that they separate the feature extraction and classification steps. This solution is not always feasible in terms of convergence rate and generalization performance. [21] has leveraged multi-view graph networks to fuse data from different sources (vibration, current, etc.) for fault diagnosis in wind turbines. They have used interactive shared fusion to enhance feature learning. One of the main limitations of this approach is the need for partial a priori knowledge to model the relationships between different data sources effectively. Furthermore, as the size of the graph increases, scalability becomes a significant concern. Adaptive co-attention mechanisms have been introduced in [22] to enhance FD in rotating machinery by integrating multiple views, such as vibration and acoustic signals. However, the dynamic adjustment of attention weights across multiple views requires additional computations, which can slow down the overall training and validation process. [23] have integrated multi-view and multi-level features using a DNN to improve FD accuracy. They have combined data from different levels of the machinery to enhance feature representation.

Unlike these methods, our MTMVL-CE strategy has been designed to effectively model both the heterogeneity of log data sources and the various faults that may occur in the machinery. The features learned are view-invariant and can adapt across different types of faults. This adaptability is crucial, particularly when heterogeneity exists not only in the feature space but also in the multi-task output space. By addressing both aspects, our approach ensures robust and accurate fault classification even in complex and varied conditions.

2.2. Multi-label and multi-task approaches for classifying multiple faults in a FD scenario

Multi-Task Learning (MTL) involves training a model on multiple related tasks simultaneously. It can be particularly effective when tasks are related or have overlapping domains, as it allows the model to generalize better by learning a shared representation that captures the underlying structure of all tasks. Multi-label classification (MLC) is a case of MTL, where the input features are the same across all tasks/labels, and the model predicts multiple labels simultaneously. Most recent multi-label research has primarily focused on theoretical aspects such as causal and Markov-boundary feature selection, as well as streaming feature selection in multi-label settings [24–26], rather than on domain-specific fault diagnosis applications.

MTL and MLC approaches have been increasingly applied to improve the generalization performance of STL models to predict multiple faults in challenging domains such as aerospace manufacturing [27] and marine machinery [28]. Although MTL [29,30] and MLC [31,32] have been widely utilized to address domain adaptation challenges in complex FD scenarios, our approach aims to develop a unified framework that not only addresses the heterogeneity of data sources, but also effectively captures the interdependencies among multiple faults. Traditional classification methods, such as K-Nearest Neighbors [33] and classification trees [34], have been extended for multi-fault classification, while deep learning models integrating MLC strategies have demonstrated effectiveness across various domains [35,36]. Our work in this paper goes beyond their approach by additionally incorporating the MTL and MVL frameworks to classify multiple faults represented by heterogeneous log data sources. Moreover, our approach is enhanced by a multi-label Cross Entropy loss to enhance the classification accuracy. Our MTMVL-CE has the advantage of maximizing the classification performance between and within fault types at the same time.

The works most similar to ours have been presented in [37–39]. In [37], the authors have proposed a multi-input MTL CNN for fault diagnosis using bearing vibration signals. Similarly, [38] have proposed an MTL approach to elaborate data from multiple sensors and jointly learn fault classification and sensor validation tasks. [39] have employed a multitask CNN with information fusion to diagnose and localize bearing faults. The model integrates information from various data sources to enhance fault diagnosis and localization accuracy. Unlike our approach, their methods do not incorporate a multi-label loss function, which can significantly enhance the joint learning across multiple tasks. Additionally, our framework is more generalized because each view is not confined to a specific output subset but can potentially be discriminative for all types of faults, providing a broader and more flexible diagnostic capability. Another key difference lies in the signal used: unlike other methods, which typically rely on sensor-based data such as vibration, electric, and acoustic signals, our approach is based entirely on log data. Predicting faults using log data is more challenging due to the unstructured nature of logs and their potentially lower discriminative power.

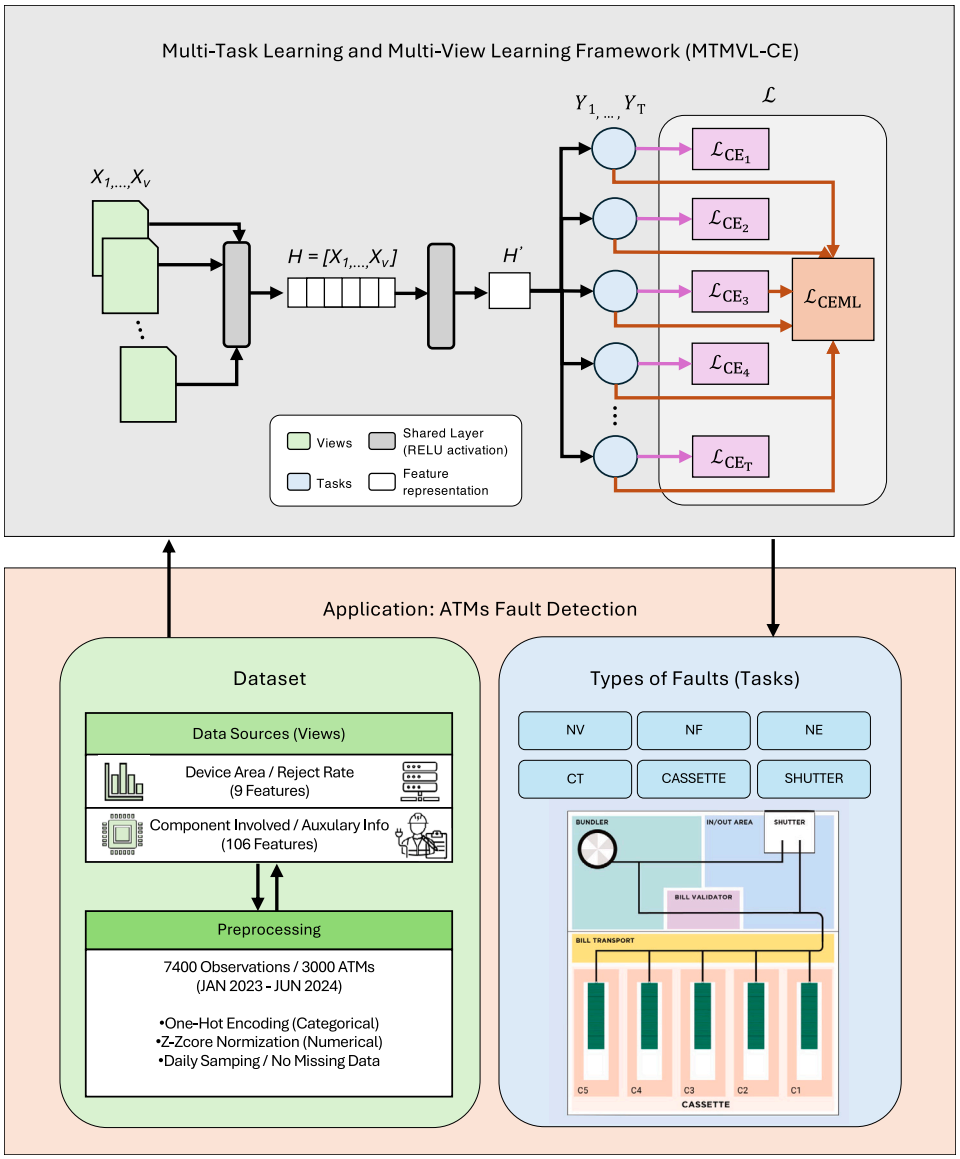


Fig. 1. Detailed architectural diagram of the proposed Multi-Task Multi-View Learning with multi-label Cross Entropy (MTMVL-CE) framework. The upper section illustrates the methodological architecture, which is general and applicable to various multi-source, multi-task problems. Directional black arrows indicate the data flow from input views to task outputs. The shared layer block performs cross-view feature fusion (Eq. (3)), projecting concatenated features into a unified latent space (H'). The training objective combines task-specific objectives via standard Cross-Entropy ($\mathcal{L}_{CE_t}$) and cross-task dependencies via the custom Multi-Label Loss ($\mathcal{L}_{CEML}$), as annotated in the loss aggregation block $\mathcal{L}$. The lower section presents the application to ATM fault detection, where the two views correspond to device-level indicators and component-related auxiliary information, collected from 7400 observations across 3000 ATMs (Jan 2023–Jun 2024).

3. Methodology

This section describes the proposed MTMVL-CE framework, designed to jointly model multiple tasks and heterogeneous data sources (see Fig. 1). While the present study applies this method to fault detection from ATMs log data, the framework defines a general methodological architecture that can be applied to a wide range of multi-source, multi-task classification problems beyond the specific domain considered in this work, such as other industrial or operational domains characterized by heterogeneous signals and multiple correlated fault types.

3.1. Notation

To define our MTMVL-CE, we use the following notation:

- we let N the number of training points
- we let V and T the number of views and tasks respectively;
- we let $X_v \in \mathbb{R}^{N \times D_v}$ the input features from the vth view where D_v is the dimensionality of the vth view;
- we let $\mathcal{Y}_t = \{0, 1\}$ the output space from the tth task;
- we let $\mathcal{Y} \in \mathbb{R}^{N \times T}$ the overall output space;
- we let $y_{t,j}$ the class label of the jth training sample in the tth task;
- we let $\hat{p}_{t,j}$ and $\hat{y}_{t,j}$ the predicted probability and output of the jth training sample in the tth task.

We set the output space to be binary. Nonetheless, our approach is general and can be extended to any multi-classification task. In addition, in our formulation, we fix the number of samples for each view and task. Although this study focuses on log-based data, the proposed methodology is broadly applicable and can be adapted to other data modalities such as images, audio signals, or time-series sensor data. The proposed MTMVL-CE naturally supports such extensions, as it readily generalizes to different number of views (V) and tasks (T) by

taking into account the distinct nature of input signals (i.e., different dimensions for each view) and fault types. To confirm this point, we have provided the general formulation of the proposed MTMVL-CE algorithm.

3.2. Our multi-view multi-task learning approach

To formalize the MVL strategy, we define a general equation combining features from multiple views and process them through shared and task-specific layers. Each view X_v is processed through a shared layer to produce a transformed feature representation H_v:

$$H_v = \text{ReLU}(W_v^V X_v + b_v^V) \tag{1}$$

where W_v^V and b_v^V are view specific weights and biases respectively. The transformed features from all views are concatenated to form a combined feature representation H:

$$H = \text{concat}(H_1, H_2, \dots, H_V) \tag{2}$$

where V is the total number of views.

The combined feature representation H is processed through an additional shared layer and then fed into task-specific layers to produce the final predictions for each task:

$$H' = \text{ReLU}(W^S H + b^S) \tag{3}$$

where W^S and b^S are the weights and biases of the combined shared layer. It is worth noting that the number of shared layers and the task specific layers are hyperparameters to be tuned during the validation stage. In the MVL scenario, it often makes sense to process one source of information in the context of another. This process is known as context-based processing or conditioning, where the computation carried out by a single model (single task) is conditioned by each view [40]. Our methodology aims to generalize this step by avoiding any conditioning mechanism, as the classification of a single fault may be influenced by multiple views rather than a single view.

The objective in our MTMVL-CE is to learn a set of functions $\mathbf{f} = \{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_T\}$ which map the combined feature representation $H' \rightarrow \mathcal{Y}$, such that the losses within and between all tasks are minimized. Our formulation minimizes the penalized empirical loss:

$$\min_{\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_T} \sum_{t=1}^{T} \mathcal{L}(\mathbf{f}_t(H'), \mathcal{Y}_t) + \lambda \Omega(\mathbf{f}_1, \dots, \mathbf{f}_T) \tag{4}$$

where $\mathcal{L}$ represents a loss function suitable for binary classification (e.g., cross-entropy loss) and Ω serves as a regularization term that promotes shared learning across related tasks by capturing underlying similarities. Our method combines two loss functions that maximize the classification performance between and within fault types. The standard cross-entropy (CE) loss is typically used to optimize classification performance within each task or fault type. It focuses on independently maximizing the CE of the correct class labels for each task. This procedure can be particularly effective in scenarios where the tasks are relatively isolated or where there is no need to capture interactions between tasks.

CE is computed for each tth task as follows:

$$\mathcal{L}_{CE_t} = -\frac{1}{N} \sum_{j=1}^{N} c_{t,j} y_{t,j} \log(\hat{p}_{t,j}) \tag{5}$$

where $\hat{p}_{t,j} = \mathbf{f}_t^\top H'_j$ and H'_j is the combined feature representation for the jth sample across all views. Here, $c_{t,j}$ represents the class weight for the class of the jth sample in the tth task. Class weights are set according to the prior probability of the training set. This adjustment allows the model to handle class imbalance by giving more importance to under-represented classes. In contrast, the multi-label CE loss is designed to maximize classification performance across multiple tasks. This loss is beneficial when the tasks are related or potential interactions between different fault types should be captured. By averaging over tasks the CE, this loss formulation ensures that the performance on other tasks influences the error gradient received by each task. This procedure encourages the model to find an effective feature representation H' across all tasks. Multi-label CE is computed for each jth sample as follows:

$$\mathcal{L}_{CEML_j} = -\frac{1}{T} \sum_{t=1}^{T} y_{t,j} \log(\hat{p}_{t,j}) \tag{6}$$

It is worth noting that $\mathcal{L}_{CE_t}$ focuses on each task t individually averaging the single task loss across all training samples. In contrast $\mathcal{L}_{CEML_j}$ aggregates the contributions of the predicted probabilities across all tasks for each sample, averaging the multiple task loss over tasks to promote more robust generalization. Furthermore, $\mathcal{L}_{CEML}$ can be seen as a proxy for the Hamming loss metric. We combine the following losses as follows:

$$\mathcal{L} = \alpha \sum_{t=1}^{T} \mathcal{L}_{CE_t} + (1-\alpha) \sum_{j=1}^{N} \mathcal{L}_{CEML_j} \tag{7}$$

where α is a hyperparameter that regulates a trade-off between $\mathcal{L}_{CE}$ and $\mathcal{L}_{CEML}$. This composite loss function incorporates $\mathcal{L}_{CE}$, which is designed to optimize performance within individual tasks independently, and $\mathcal{L}_{CEML}$, which aims to enhance performance across multiple tasks collectively. For the regularization term, we employ a quadratic regularizer akin to ridge regression, which both encourages linear relationships among tasks and penalizes the Euclidean norm of each model [41]. This regularizer is mathematically represented as:

$$\Omega(\mathbf{f}) = \beta_0 \|\mathbf{f} R\|_F^2 + \beta_1 \|\mathbf{f}\|_F^2 \tag{8}$$

where R encodes linear relationships between tasks. $\| \cdot \| F$ denotes the Frobenius norm (i.e., the Euclidean norm of the vector formed by the matrix elements). The hyper-parameter $\beta 1$ regulates the magnitude of the model parameters, while β_0 promotes structural relationships between the models. In our problem, we relax the set of task-specific function $\mathbf{f}$ to be linear:

$$\mathbf{f}_t = (W_t, b_t) \tag{9}$$

where W_t and b_t are the weights and biases of the task-specific layer for the tth task. Thus Eq. (8) can be formulated as follows:

$$\Omega(W) = \beta_0 \|W R\|_F^2 + \beta_1 \|W\|_F^2 \tag{10}$$

The matrix R encodes linear relationships computed based on task correlation. To ensure a rigorous quantification of both positive and negative dependencies, we compute ρ as the Pearson correlation matrix derived from the binary target vectors of the training set considering historical data occurrences of faulty events. ρ is the correlation matrix and $\rho_{s,t}$ denotes the correlation coefficient between training occurrences of task (fault) s and t. The R matrix encodes a complete graph structure as follows:

$$R = \mathbb{I}_T - \frac{\mathbb{1}_T - \rho}{T} \tag{11}$$

where $\mathbb{I}$ and $\mathbb{1}$ are the identity matrix and the matrix with all ones, respectively. By incorporating R into the regularization term $\Omega(W)$, we explicitly enforce consistency among the learned parameters W. Specifically, the regularization penalizes the model when the weight vectors of correlated tasks diverge, thereby acting as an inductive bias that transfers knowledge between related faults based on their statistical dependencies observed in the training set. In our industrial setting, fault interdependency patterns and the resulting task-relatedness matrix ρ used to construct R were observed to be statistically stable over time. Hence, R is computed offline from historical correlation statistics and kept fixed during training and inference to ensure stability and low computational overhead. Nevertheless, the framework can support periodic updates of R through lightweight fine-tuning when new annotations become available, and can be adapted to substantially different industrial domains via transfer learning by recomputing R and fine-tuning the shared and task-specific layers.

For each task, we compute the individual prediction as follows:

$$\hat{y}_t = \text{Softmax}(W_t H' + b_t) \quad \forall t \in \{1, \dots, T\} \tag{12}$$

Algorithm 1 outlines the overall procedure of the proposed MTMVL-CE. Since we implement a standard grid search procedure, our Pseudocode does not include the validation stage.

Algorithm 1 Pseudocode MTMVL-CE

function TRAIN(Training data)
  Initialize view-specific weights W_v^V and biases b_v^V
  Initialize shared layer weights W^S and bias b^S
  Initialize task-specific weights W_t and biases b_t for each t
  for each epoch **do**
    for each batch in Training data **do**
      for each view $v = 1$ to V **do**
        $H_v \rightarrow \text{ReLU}(W_v^V X_v + b_v^V)$ ▷ (1)
      end for
      $H \rightarrow \text{concat}(H_1, H_2, \dots, H_V)$ ▷ (2)
      $H' \rightarrow \text{ReLU}(W^S H + b^S)$ ▷ (3)
      for each task $t = 1$ to T **do**
        Compute CE loss $\rightarrow \mathcal{L}_{CE_t}$ ▷ (5)
      end for
      for each sample $j = 1$ to N **do**
        Compute ML CE loss $\rightarrow \mathcal{L}_{CEML_j}$ ▷ (6)
      end for
      Aggregate losses $\rightarrow \mathcal{L}$ ▷ (7)
      Compute regularization $\rightarrow \Omega$ ▷ (10)
      Penalized empirical loss $\rightarrow \mathcal{L} + \lambda\Omega$ ▷ (4)
      Update weights via backpropagation
    end for
  end for
  return (W^V, b^V), (W^S, b^S), (W, b)
end function
function TEST(Testing data, (W^V, b^V), (W^S, b^S), (W, b))
  for each sample in Testing data **do**
    for each view $i = 1$ to V **do**
      $H_i \rightarrow \text{ReLU}(W_v^V X_v + b_v^V)$
    end for
    $H \rightarrow \text{concat}(H_1, H_2, \dots, H_V)$
    $H' \rightarrow \text{ReLU}(W^S H + b^S)$
    for each task $t = 1$ to T **do**
      $\hat{y}_t \rightarrow \text{Softmax}(W_t H' + b_t)$ ▷ (12)
    end for
  end for
  return Predictions $\{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_T\}$
end function

4. Experiments

This section presents the dataset used for the experiments, the experimental procedure and the comparisons.

4.1. Dataset

The selected case study centers on the banknote recirculator, a complex component within ATMs responsible for dispensing and depositing banknotes. Troubleshooting this device presents significant challenges and is often expensive, necessitating skilled technicians to interpret error codes and diagnostics, especially when the faults do not interrupt operations. The data, extracted from the ATM platform, were sampled continuously over time and included fault annotations made by technicians. Specifically, technicians highlighted target areas requiring inspection. These areas indicate potential failure points, assisting technicians in detecting the locations requiring maintenance intervention. Thus, in our FD task, failure annotation refers to identifying the specific area within the device where the failure occurs (see Table 1): CASSETTE: Drawers, CT: Bill transport, NE: Bundler, NF: In/Out area, NV: Note Validator, SHUTTER. These represent the six key fault categories of our problem.

Table 1
Specific Target Areas of banknote recirculator.

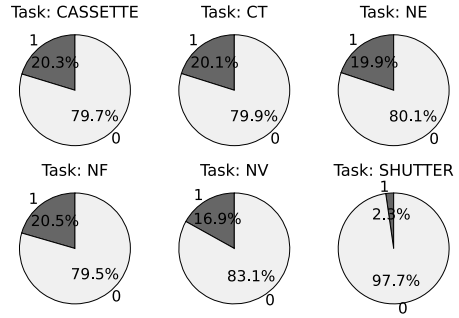
Id area	Area name	Description
CASSETTE	Drawers	Drawers, contain the banknotes
CT	Bill transport	Bill transport system
NE	Bundler	Component that contains the banknotes dispensed from the drawers or coming from the In/Out area
NF	In/Out area	Component that contains an area, accessible to the user, where banknotes presented or to be deposited are placed
NV	Note Validator	Banknote validation device
SHUTTER	Shutter	Shutter that allows the user to access the In/Out area

The dataset comprised 7400 observations collected from 3000 ATMs between January 1, 2023, and June 30, 2024. Raw ATM logs were structured in a SQLite database maintained by the service provider and derived from device-level operational records. The dataset, aggregated daily with minimal formatting issues, enabled feature construction without extensive raw-text parsing. Given that ATMs typically experience infrequent faults, the dataset shows an uneven distribution of failures across different devices. The number of recorded faults per ATM varies from one to ten, reflecting real-world operational conditions, where some ATMs experience frequent issues while others remain operational for extended periods without faults.

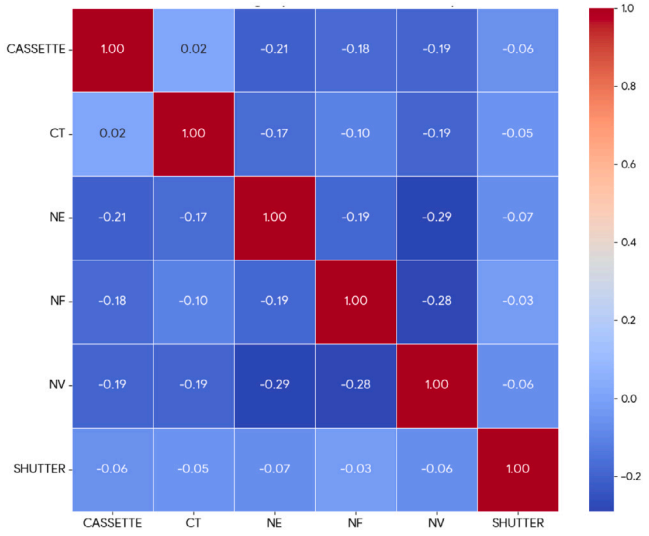
The log operational data serves as the feature set for our model, derived from two distinct sources or views: device area and reject rate and component involved and auxiliary information. The former view provides detailed information on the operational performance of the machine, including the history of errors and maintenance actions. This view results in a total of nine features. The latter offers supplementary data about the machine and its specific components. This view results in a total of one hundred-six features. The device area/reject rate view included reject rates (numerical) and error logs (categorical), while the component involved/auxiliary info view combined pre-incident errors (categorical) and operational counters (numerical). Features were selected with domain experts to capture anomalies in the banknote recirculator. Categorical features were encoded using one-hot encoding, while numerical variables were normalized using z-score. Data was consistently sampled daily, with no missing views or temporal misalignment. As a consequence, our experiments did not encounter degraded or incomplete input views.

These views provide a comprehensive understanding of the machine's operational state, enabling the precise identification of potential issues. Both views were deliberately treated uniformly to reflect their complementary nature, capturing device anomalies and providing contextual signals related to banknote flow. This design choice was also inspired by how expert technicians analyze logs holistically without prioritizing one source over another. Our MVL strategy is specifically tailored to leverage these data sources' heterogeneity, integrating diverse information types for a more robust analysis. Concurrently, our MTL strategy is designed to predict multiple fault areas simultaneously, optimizing the diagnostic process by addressing several potential issues in one cohesive model.

Together, these strategies enhance the system's ability to diagnose and prioritize interventions effectively, ensuring a thorough and efficient response to the complexities of the ATM's operational dynamics. Certain failure annotations (areas) like NF, CASSETTE, and CT show much higher incident rates than others like SHUTTER (see Fig. 2(a)). Our MTMVL-CE incorporates class weights in the CE loss function ($\mathcal{L}_{CE}$). This procedure helps balance the learning process within each task, ensuring the model does not neglect the less frequent classes.



(a) Class distribution for each fault task, highlighting the data imbalance.



(b) Fault-category correlation heatmap displaying the Pearson correlation coefficients between fault occurrences.

Fig. 2. Dataset analysis from January 2023–June 2024 interval.

Beyond the class distribution, analyzing the statistical relationships between fault categories provides critical insights for diagnosis. Fig. 2(b) presents the Pearson correlation heatmap of the fault occurrences, revealing the inherent statistical structure governing the interplay between different failure modes. We observe distinct patterns of dependency among the fault categories (e.g., the relationship between NE and NV or CASSETTE and CT). These structural constraints are explicitly encoded in the regularization matrix R (Eq. (11)) of our MTMVL-CE framework.

4.2. Experimental procedure

To thoroughly assess the effectiveness and generalizability of our MTMVL-CE approach, we implemented a temporal hold-out evaluation strategy. This method is essential for testing the model's ability to maintain reliable classifications across time, ensuring generalization to unseen faulty events. The training phase utilized data collected from January 1 to November 30, 2023, while testing was conducted on data spanning January 1 to June 30, 2024. Hyperparameter tuning was performed using validation data from December 2023, with a grid-search procedure employed to maximize subset accuracy. Table 2 describes the hyperparameters range for the proposed MTMVL-CE approach and state-of-the-art comparisons. Due to the limited size of our dataset and the inherently low frequency of daily-level faults and predictors, we deliberately adopted simpler architectures with linear blocks to mitigate overfitting risks. More complex models, such as transformers

Table 2
Hyperparameters range for the proposed MTMVL-CE approach and state-of-the-art comparisons. Hyperparameter tuning was conducted by validating the model on observations from December 2023 and optimizing the subset accuracy using a grid-search procedure.

Method	Hyperparameters	Range
MTMVL-CE	learning rate	$(0.0001, 0.001, 01)$
	batch size	$(16, 32, \dots, 256)$
	n. of epochs	$(10, 20, \dots, 50)$
	α	$(0, 0.1, \dots, 1)$
	β_0	$(0, 0.0001, 0.001, \dots, .1)$
	β_1	$(0, 0.0001, 0.001, 0.01)$
	n. of task specific layers	$(1, 2, \dots, 5)$
	n. of shared layers	$(1, 2, \dots, 5)$
LR	ρ_2	$(0.0001 - 0.01)$
XGBoost	learning rate	$(0.0001, 0.001, 01)$
	n. of estimators	$(10, 20, \dots, 200)$
	max depth	$(5, 10, \dots, 100)$
MLMV-DNN	learning rate	$(0.0001, 0.001, 01)$
	batch size	$(16, 32, \dots, 256)$
	n. of epochs	$(10, 20, \dots, 50)$
	n. of hidden layers	$(1, 2, \dots, 5)$
	n. of shared layers	$(1, 2, \dots, 5)$
ML-DNN	learning rate	$(0.0001, 0.001, 01)$
	batch size	$(16, 32, \dots, 256)$
	n. of epochs	$(10, 20, \dots, 50)$
	n. of hidden layers	$(1, 2, \dots, 5)$
	n. of shared layers	$(1, 2, \dots, 5)$
HEA-MV	combining strategy	$(soft, hard)$

or GNNs, require larger datasets to generalize effectively. This choice is further supported by our prior work in data-rich industry domains [12], where we successfully applied CNN- and transformer-based approaches. As further evidence the validation process consistently selected shallow architectures (one shared and one task-specific layer), indicating that simpler models provide better generalization and reduced overfitting in this real-world FD setting. Moreover, we adopted a linear trade-off between L_{CE} and L_{CEML} to ensure a balance between task-specific and cross-task learning contributions. This design is particularly suitable for our setting, where the number of tasks and view dimensions is moderate and the dataset originates from a real industrial environment with limited sample size. In such scenarios, overly complex adaptive or nonlinear combinations may increase overfitting risk without guaranteeing improved generalization.

We performed various experiments to evaluate our approach comprehensively. These experiments included (i) testing each month separately to assess performance for individual periods and (ii) testing cumulatively on each month to analyze how the model performs as more data is incrementally added over time.

For evaluating the performance of the proposed approach, the following multi-label metrics [42] were used: *balanced accuracy (BA)*: the average of recall obtained on each class for each task; *macro- balanced accuracy (macro-BA)*: Averages the balanced accuracy across all tasks; *micro- balanced accuracy (micro-BA)*: the accuracy considering all true positives, false positives, true negatives, and false negatives across all tasks; *hamming Loss (HL)*: Measures the fraction of wrong labels to the total number of labels; *ranking Loss (RL)*: the average fraction of label pairs that are incorrectly ordered and *Subset Accuracy (SA)*: the fraction of samples whose predicted multi-labels match exactly the ground-truth labels.

All the experiments are reproducible and were performed using an NVIDIA GPU RTX 2080 Ti. The code for our MTMVL-CE approach is available at the following link: https://github.com/rosati1392/MTMVL-CE

Table 3
Experimental results of the proposed MTMVL-CE approach and state-of-the-art comparisons. The evaluated metrics are macro-balanced accuracy (macro-BA), micro-balanced accuracy (micro-BA), Hamming Loss (HL), Ranking Loss (RL) and Subset Accuracy (SA). The method achieving the highest SA is highlighted in bold.

Method	Test set	macro-BA↑	micro-BA↑	HL↓	RL↓	SA↑
MTMVL-CE	January	0.774	0.784	0.115	0.134	**0.601**
MLMV-DNN		0.706	0.743	0.120	0.171	0.502
ML-DNN		0.740	0.759	0.119	0.155	0.536
HEA-MV		0.762	0.780	0.112	/	0.589
XGBoost		0.762	0.778	0.117	0.128	0.576
LR		0.760	0.775	0.115	0.127	0.580
MTMVL-CE	February	0.804	0.813	0.095	0.112	**0.627**
MLMV-DNN		0.728	0.765	0.105	0.167	0.530
ML-DNN		0.770	0.791	0.104	0.140	0.584
HEA-MV		0.794	0.806	0.095	/	0.622
XGBoost		0.790	0.801	0.099	0.110	0.601
LR		0.788	0.806	0.097	0.109	0.620
MTMVL-CE	March	0.799	0.790	0.111	0.117	**0.613**
MLMV-DNN		0.688	0.742	0.121	0.193	0.501
ML-DNN		0.732	0.764	0.118	0.162	0.558
HEA-MV		0.778	0.780	0.111	/	0.590
XGBoost		0.778	0.781	0.114	0.118	0.578
LR		0.770	0.772	0.116	0.119	0.578
MTMVL-CE	April	0.782	0.799	0.108	0.122	0.595
MLMV-DNN		0.705	0.756	0.115	0.160	0.530
ML-DNN		0.743	0.768	0.121	0.173	0.546
HEA-MV		0.780	0.793	0.105	/	0.597
XGBoost		0.782	0.797	0.104	0.104	**0.609**
LR		0.759	0.774	0.115	0.106	0.560
MTMVL-CE	May	0.768	0.773	0.123	0.136	**0.545**
MLMV-DNN		0.679	0.720	0.135	0.196	0.435
ML-DNN		0.737	0.748	0.134	0.192	0.484
HEA-MV		0.768	0.768	0.120	/	0.527
XGBoost		0.766	0.768	0.123	0.121	0.522
LR		0.759	0.760	0.126	0.120	0.515
MTMVL-CE	June	0.774	0.789	0.110	0.122	**0.577**
MLMV-DNN		0.693	0.742	0.123	0.176	0.487
ML-DNN		0.741	0.763	0.122	0.174	0.520
HEA-MV		0.755	0.778	0.112	/	0.558
XGBoost		0.764	0.783	0.113	0.110	0.548
LR		0.756	0.771	0.117	0.120	0.546
MTMVL-CE	January-February	0.788	0.797	0.106	0.124	**0.613**
MLMV-DNN		0.716	0.753	0.113	0.170	0.515
ML-DNN		0.754	0.774	0.112	0.148	0.558
HEA-MV		0.776	0.792	0.104	/	0.604
XGBoost		0.775	0.788	0.108	0.119	0.587
LR		0.773	0.789	0.106	0.119	0.599
MTMVL-CE	January-March	0.791	0.795	0.107	0.122	**0.613**
MLMV-DNN		0.711	0.750	0.115	0.176	0.511
ML-DNN		0.750	0.771	0.114	0.152	0.558
HEA-MV		0.777	0.789	0.106	/	0.600
XGBoost		0.776	0.786	0.110	0.119	0.585
LR		0.773	0.784	0.109	0.119	0.593
MTMVL-CE	January-April	0.790	0.796	0.107	0.122	**0.609**
MLMV-DNN		0.711	0.751	0.115	0.173	0.515
ML-DNN		0.750	0.770	0.115	0.157	0.555
HEA-MV		0.779	0.789	0.106	/	0.599
XGBoost		0.779	0.789	0.109	0.116	0.590
LR		0.772	0.782	0.110	0.116	0.586
MTMVL-CE	January-May	0.786	0.791	0.111	0.125	**0.596**
MLMV-DNN		0.705	0.745	0.119	0.177	0.499
ML-DNN		0.747	0.766	0.119	0.164	0.541
HEA-MV		0.777	0.785	0.109	/	0.585
XGBoost		0.776	0.784	0.111	0.117	0.577
LR		0.769	0.777	0.113	0.117	0.572
MTMVL-CE	January-June	0.785	0.791	0.111	0.124	**0.593**
MLMV-DNN		0.704	0.744	0.120	0.177	0.498
ML-DNN		0.747	0.765	0.119	0.166	0.538
HEA-MV		0.775	0.784	0.109	/	0.581
XGBoost		0.776	0.784	0.112	0.116	0.572
LR		0.768	0.776	0.114	0.117	0.568

(*continued on next page*)

Table 3 (*continued*).

4.3. Experimental comparisons

The XGBoost method [43] was employed as a classification model due to its high generalization performance and low risk of overfitting, outperforming other data mining techniques commonly used for structured data in the FD task [44,45]. Additionally, XGBoost surpassed other tested supervised classifiers, including Random Forest, Decision Tree, and K-Nearest Neighbors. The Logistic Regression model (LR) was chosen as the baseline because our MTMVL-CE approach represents a natural extension of standard supervised single-task learning methods, namely LR. All baseline models, including XGBoost, Logistic Regression, Random Forest, Decision Tree, and K-NN, were implemented using a binary relevance strategy, where independent binary classifiers are trained for each fault class. We further compare our approach with state-of-the-art ML-DNN classifiers [35,36] widely used for detecting multiple faults across various domains. In this comparison (ML-DNN), we consider a single view of the input data by combining the different views. Furthermore, we considered an additional comparison (MLMV-DNN), where we employed a fusion strategy to integrate multi-view [23,37–39] using DNN shared layers and multi-task branches to integrate multiple faults. Similar to [20], we decided to compare our MTMVL-CE approach with a hybrid ensemble approach (HEA-MV) that combines different views of data by ensembling various classifiers. For this comparison we used a simplified version of the hybrid ensemble in [20] combining XGBoost and LR via hard voting and omitting the stacking meta-learner to align with the non-ensemble nature of our approach and reduce deployment overhead. As further experiments, we conducted an ablation study on our MTMVL-CE approach by excluding the proposed multi-view (MV) strategy and the multi-label CE loss.

5. Results

This section presents the experimental results of various experiments, including experimental comparisons and ablation study. Table 3 summarizes the test results for each month individually to evaluate performance for specific periods. It also includes cumulative monthly testing results to examine how the model's performance evolves as more data is incrementally added. Table 3 includes macro-BA, micro-BA, HL, RL, and SA. Further, BA metrics for each class are reported in the supplementary material (see Technical Appendix A).

MTMVL-CE consistently achieved the highest SA across all test periods except April, indicating its superior ability to predict the complete set of faults correctly. The method also showed robust performance in metrics like macro-BA and micro-BA, demonstrating its balanced and comprehensive approach to handling multi-label classification. In cumulative testing, MTMVL-CE continued to demonstrate strong performance. This robustness indicates that MTMVL-CE effectively adapts to changes in the data distribution over time and maintains high predictive accuracy as the test set grows.

To verify the superiority of the proposed approach with that of other competitors, we used statistical tests to compare the SA of different experiments. Since the SA metric deviates from normality according to the Anderson–Darling test ($p < .01$), we used non-parametric one-sided Wilcoxon signed-rank tests ($\alpha = 0.05$). SA of MTMVL-CE is significantly greater than MLMV-DNN ($W = 66; p < 0.01$), ML-DNN ($W = 66; p < 0.01$), HEA-MV ($W = 65; p < 0.01$), XGBoost ($W = 65; p < 0.01$) and

Table 4
Experimental results of the ablation study: our proposed MTMVL-CE without the multi-label cross-entropy loss (ML-CE loss) and without the MV strategy. The evaluated metrics are macro-balanced accuracy (macro-BA), micro-balanced accuracy (micro-BA), Hamming Loss (HL), Ranking Loss (RL), and Subset Accuracy (SA). The method achieving the highest SA is highlighted in bold.

Method	Test set	macro-BA↑	micro-BA↑	HL↓	RL↓	SA↑
MTMVL-CE	January	0.774	0.784	0.115	0.134	**0.601**
-ML-CE LOSS		0.759	0.766	0.145	0.183	0.526
-MV		0.770	0.784	0.119	0.143	0.577
MTMVL-CE	February	0.804	0.813	0.095	0.112	**0.627**
-ML-CE LOSS		0.786	0.795	0.127	0.157	0.551
-MV		0.801	0.812	0.103	0.130	0.611
MTMVL-CE	March	0.799	0.790	0.111	0.117	**0.613**
-ML-CE LOSS		0.766	0.767	0.140	0.174	0.546
-MV		0.792	0.783	0.118	0.143	0.584
MTMVL-CE	April	0.782	0.799	0.108	0.122	**0.595**
-ML-CE LOSS		0.740	0.765	0.148	0.174	0.506
-MV		0.768	0.782	0.121	0.136	0.556
MTMVL-CE	May	0.768	0.773	0.123	0.136	**0.545**
-ML-CE LOSS		0.743	0.748	0.164	0.210	0.455
-MV		0.764	0.767	0.132	0.165	0.515
MTMVL-CE	June	0.774	0.789	0.110	0.122	**0.577**
-ML-CE LOSS		0.744	0.760	0.153	0.192	0.489
-MV		0.778	0.784	0.119	0.150	0.556
MTMVL-CE	January-	0.788	0.797	0.106	0.124	**0.613**
-ML-CE LOSS	February	0.771	0.779	0.137	0.171	0.537
-MV		0.784	0.797	0.112	0.137	0.593
MTMVL-CE	January-	0.791	0.795	0.107	0.122	**0.613**
-ML-CE LOSS	March	0.771	0.776	0.138	0.172	0.540
-MV		0.786	0.793	0.113	0.139	0.591
MTMVL-CE	January-	0.790	0.796	0.107	0.122	**0.609**
-ML-CE LOSS	April	0.765	0.773	0.140	0.172	0.532
-MV		0.784	0.790	0.115	0.138	0.583
MTMVL-CE	January-	0.785	0.791	0.111	0.125	**0.596**
-ML-CE LOSS	May	0.760	0.767	0.145	0.180	0.517
-MV		0.781	0.786	0.118	0.144	0.570
MTMVL-CE	January-	0.785	0.791	0.111	0.124	**0.593**
-ML-CE LOSS	June	0.760	0.767	0.146	0.182	0.513
-MV		0.781	0.785	0.118	0.145	0.567

LR ($W = 66; p < 0.01$). The Friedman test results indicate a statistically significant difference in performance ranking among the six methods ($F = 51.07; p < 0.001$). Furthermore the mean rankings (MTMVL-CE: 1.18, MLMV-DNN: 6.00, ML-DNN: 5.00, HEA-MV: 2.00, XGBoost: 3.23, LR: 3.59) highlights the superiority of the MTMVL-CE.

5.1. Ablation study

The ablation study in Table 4 provides a detailed comparison of the performance of our MTMVL-CE model against its variants where either the multi-label CE loss (ML-CE LOSS) or the multi-view component (MV) is removed. Table 4 includes macro-BA, micro-BA, HL, RL, and SA. Further, BA metrics for each class are reported in the Supplementary material (see Technical Appendix B).

Removing the ML-CE loss significantly reduces performance across all metrics, highlighting its critical role in classification. Likewise, excluding the MV component degrades performance, reinforcing the importance of leveraging multiple data views. Indeed, the SA of MTMVL-CE is significantly greater than the variants without ML-CE loss ($W = 66; p < 0.01$) and MV ($W = 66; p < 0.01$) components. According to the Friedman test, a significant difference in the performance rankings of the three methods across the test sets was found ($F = 22; p < 0.001$). Furthermore, the mean rankings (MTMVL-CE: 1.0, -ML-CE LOSS: 3, -MV: 2) indicate that the full MTMVL-CE model consistently ranks highest, demonstrating the best performance in terms of Subset Accuracy.

To further illustrate the synergistic impact, we analyze the specific case of CASSETTE and CT (Bill Transport). Although the heatmap (Fig. 2(b)) shows a negligible direct correlation, indicating these faults rarely occur strictly simultaneously, they are mechanically distinct components of the same subsystem. The ablation study in Table B.6 shows that removing the Multi-Label Loss leads to a performance drop (e.g., CASSETTE BA drops from .742 to .715). This suggests that even in the absence of strong positive correlation, the MTMVL-CE framework is beneficial. The custom loss (Eq. (7)) aggregates the error gradients across all tasks. By doing so, it forces the shared layers to learn features that are highly discriminative: the model learns to distinguish strictly between a CASSETTE log pattern and a CT log pattern, preventing confusion between these functionally related but statistically distinct events.

5.2. Analysis on time complexity

The time complexity of MTMVL-CE originated from the MVL and MTL steps. Each view X_v is processed through a shared layer with a time complexity of $O(N \cdot \sum_{v=1}^{V} D_v \cdot D')$ where D' is the dimension of the shared layer. The shared layer acts as a bottleneck to fuse multi-view data, and its time complexity grows linearly with the number of views, samples, dimensions of each view, and dimension of the shared layer. The computation complexity of the proposed MTL is composed by Eq. (5) $O(N^2 \cdot T)$ and Eq. (6) $O(N \cdot T^2)$. $\mathcal{L}_{CE}$ is more computationally intensive with many samples since it computes each task's loss independently over the dataset. On the other hand, the computational complexity of $\mathcal{L}_{MLCE}$ scales quadratically with the number of tasks due to the aggregation of task contributions per sample.

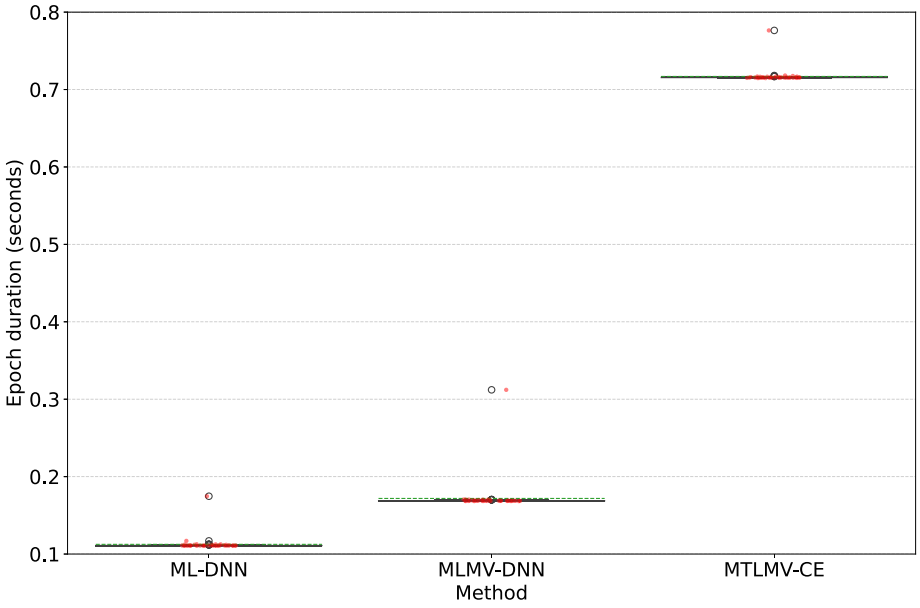


Fig. 3. Computation time in terms of epoch duration for ML-DNN, MLMV-DNN, and the proposed MTMVL-CE methods.

5.3. Computation time

While ML-DNN and MLMV-DNN exhibit slightly reduced computation times, attributed to their simpler architectures and the lack of ML-CE loss, the proposed MTMVL-CE achieves a remarkable balance. MTMVL-CE maintains computational efficiency with epoch durations consistently below one second (see Fig. 3). Considering that an epoch involves both forward and backward passes over the current dataset, the inference latency—requiring only a forward pass—is orders of magnitude lower (milliseconds), fully satisfying real-time constraints. Moreover, given that this inference time is negligible compared to the daily log aggregation interval, the proposed method is well-suited for deployment. This result highlights MTMVL-CE's ability to offer a significant trade-off between computational cost and enhanced learning capability.

Furthermore, the proposed gradient-based architecture inherently supports continuous learning paradigms. In a large-scale industrial deployment, the model does not need to be retrained from scratch on the entire historical archive as the dataset grows. Instead, it can be updated incrementally by fine-tuning the weights on new incoming samples (e.g., daily logs). This capability ensures that the training computational cost remains low and constant over time, regardless of the total accumulated data volume, allowing the system to adapt dynamically to concept drift and new fault patterns described in the temporal analysis.

5.4. Sensitivity analysis

The sensitivity analysis on α and β_0 hyperparameters is conducted on the January test set. Specifically α regulates a trade-off between $\mathcal{L}_{CE}$ and $\mathcal{L}_{CEML}$ and β_0 promotes structural relationships between the models. Values of α and β_0 are selected respectively from $\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$ and $\{0, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$. Fig. 4 shows the results in terms of SA. Our MTMVL-CE achieved its best performance with $0.4 \leq \alpha \leq 0.6$ and $\beta_0 \leq 10^{-3}$.

This result highlights that moderate structural sharing ($\beta_0 \leq 10^{-3}$) is advantageous, allowing the model to place less emphasis on enforcing strict relationships between tasks. The optimal value of β_0 achieves a balance between task independence and shared learning, reflecting the inherent nature of task relationships within the data. Furthermore, an effective trade-off between the two loss components, $\mathcal{L}_{CE}$ and $\mathcal{L}_{CEML}$, underscores the effectiveness of the proposed multi-label loss in fostering robust generalization across multiple labels.

5.5. Human-centric decision support system

The proposed MTMVL-CE approach advances ML-driven FD by capturing dependencies between faults and integrating heterogeneous data sources. This capability makes it highly generalizable across various industrial apparatuses, where faults are often interrelated, and data originates from heterogeneous logs. As a result, its applicability extends beyond the tested use case to a wide range of critical applications. In manufacturing production lines, failures in robotic arms can propagate to conveyor belts and CNC machines; in aerospace systems, anomalies in hydraulic, electronic, and structural components are interrelated. Similarly, railway systems, electric vehicles, or medical equipment generate multi-source operational logs where understanding fault interactions is crucial for accurate diagnostics. MTMVL-CE's ability to synthesize insights from log-based data makes it effective in environments where real-time sensor readings are unavailable.

The integration of MTMVL-CE within a DSS enables technicians and operators at all experience levels to perform diagnostics with greater precision and reduced latency, eliminating uncertainty in decision-making and accelerating maintenance actions. Differently from rule-based schemes that require extensive domain expertise, this model-driven DSS decreases training overhead and expedites knowledge transfer. The system can deliver real-time FD without imposing burdensome hardware requirements, making it an ideal AI-enhanced solution for large-scale industrial deployments. By simultaneously modeling multiple fault types and incorporating heterogeneous machine data, MTMVL-CE generates interpretable predictions that can be used to guide technician interventions. MTMVL-CE's ability to identify interrelated faults reduces diagnostic ambiguity and narrows the search space for maintenance actions, even in the absence of real-time sensor data. By integrating these capabilities, the DSS places human expertise at the core of the FD loop, driving the shift from Industry 4.0 toward Industry 5.0. Rather than simply automating routine analyses, it reinforces human–machine collaboration, thereby fostering a truly human-centric production paradigm that optimizes operational resilience and supports sustainable, next-generation maintenance practices. In real-world

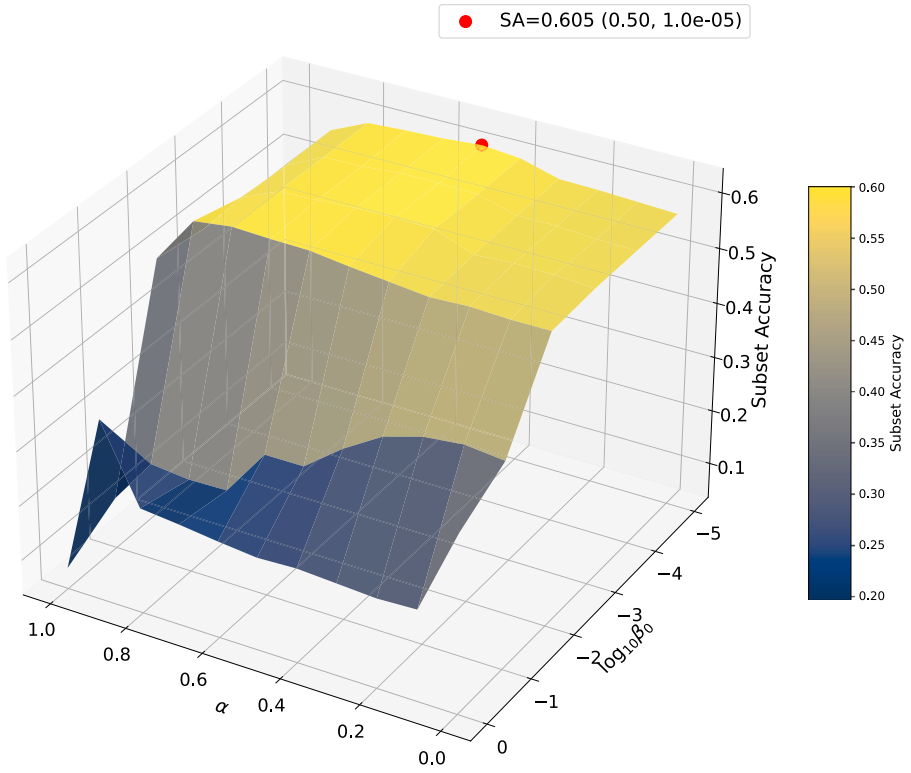


Fig. 4. Sensitivity analysis on Subset Accuracy for different values of α and β_0: test set January. The maximum Subset Accuracy is marked with a red dot.

deployments, MTMVL-CE is designed for continuous learning rather than full retraining on expanding historical data. Faults are annotated daily, and distribution shifts are typically limited, as confirmed by the extracted results. The model can therefore be efficiently fine-tuned on newly collected samples, keeping computational overhead low. More substantial changes in fault relationships may arise when the operational context itself evolves or when the framework is deployed on different industrial domain (e.g., devices with distinct fault mechanisms). In such cases, MTMVL-CE can be readily adapted through a transfer learning strategy, where R is recomputed for the new domain and the shared and task-specific layers are fine-tuned accordingly.

6. Conclusions

In this paper, we proposed MTMVL-CE, a unified framework for fault detection that jointly leverages Multi-Task and Multi-View Learning augmented by a custom multi-label Cross Entropy loss. Experimental results on 7400 real-world ATM log observations demonstrate that MTMVL-CE significantly outperforms state-of-the-art competitors, achieving superior Subset Accuracy in temporal hold-out validation. Statistical tests (Friedman and Wilcoxon, $p < 0.01$) confirm the method's robustness, particularly in its ability to generalize over time compared to standard baselines. The ablation study further verified that both the multi-view fusion and the composite loss function are essential for capturing the structural interdependencies between different fault types. An interesting future research direction is the integration of feature-level and view-level explainability mechanisms to quantify the contribution of different log sources, thereby further enhancing human-centered decision support within the DSS pipeline. Despite these promising results, we acknowledge certain limitations. First, the current implementation utilizes linear shared layers to ensure generalization. This architectural choice was necessitated by the limited size of the available dataset and the scarcity of specific fault classes; however, it may restrict the model's capacity to learn complex non-linear feature representations compared to deeper architectures like Transformers or Graph Neural Networks on a different industry domain. Second, the framework currently operates exclusively on log-based data. While this ensures broad applicability without additional hardware, it does not leverage real-time physical sensor data (e.g., vibration signals), which could further enhance diagnostic granularity in other scenarios.

Future works will focus on scaling the approach to larger datasets to explore more complex architectures and extending the multi-view framework to incorporate heterogeneous sensor fusion for real-time monitoring. Moreover, although the proposed multi-label formulation effectively models fault interdependencies, future work could explore integrating imbalance-aware single-label losses (e.g., Focal Loss) within the MTMVL-CE framework to further improve robustness under severe class imbalance.

Ethical statement

The dataset will be released after signing a Non-Disclosure Agreement (NDA) and approval by the data owners. The authors ensure the reproducibility of the results by providing code, pseudocode, and a detailed dataset description.

The proposed approach adheres to the principle of Human Agency and Oversight (Ethical Guidelines for Trustworthy AI [46]), by ensuring that human operators remain in control of maintenance decisions, using ML-driven fault detection as an assistive tool. Additionally, under the EU AI Act, the system falls within the low-risk category, as it enhances human decision-making rather than replacing it, promoting safety, and responsible AI deployment in industrial fault detection.

Table A.5
Experimental results of the proposed MTMVL-CE approach and state-of-the-art comparisons. The evaluated metrics are balanced accuracy for each task (1: CASSETTE, 2:CT, 3:NE, 4:NF, 5:NV, 6:SHUTTER), macro balanced accuracy (macro-BA), micro balanced accuracy (micro-BA), Hamming Loss (HL), Ranking Loss (RL) and Subset Accuracy. The method achieving the highest SA is highlighted in bold.

Method	Test set	1↑	2↑	3↑	4↑	5↑	6↑	macro-BA↑	micro-BA↑	HL↓	RL↓	SA↑
MTMVL-CE	January	0.716	0.734	0.855	0.803	0.777	0.756	0.774	0.784	0.115	0.134	**0.601**
MLMV-DNN		0.699	0.695	0.856	0.779	0.708	0.500	0.706	0.743	0.120	0.171	0.502
ML-DNN		0.693	0.702	0.855	0.790	0.751	0.651	0.740	0.759	0.119	0.155	0.536
HEA-MV		0.715	0.718	0.851	0.804	0.786	0.698	0.762	0.780	0.112	*NA*	0.589
XGBoost		0.716	0.702	0.853	0.809	0.777	0.718	0.762	0.778	0.117	0.128	0.576
LR		0.712	0.709	0.849	0.789	0.784	0.717	0.760	0.775	0.115	0.127	0.580
MTMVL-CE	February	0.760	0.763	0.892	0.816	0.817	0.775	0.804	0.813	0.095	0.112	**0.627**
MLMV-DNN		0.723	0.739	0.892	0.782	0.731	0.500	0.728	0.765	0.105	0.167	0.530
ML-DNN		0.729	0.735	0.891	0.801	0.810	0.652	0.770	0.791	0.104	0.140	0.584
HEA-MV		0.743	0.751	0.892	0.806	0.823	0.747	0.794	0.806	0.095	*NA*	0.622
XGBoost		0.733	0.762	0.888	0.798	0.812	0.748	0.790	0.801	0.099	0.110	0.601
LR		0.736	0.741	0.891	0.804	0.840	0.720	0.788	0.806	0.097	0.109	0.620
MTMVL-CE	March	0.696	0.759	0.883	0.801	0.737	0.916	0.799	0.790	0.111	0.117	**0.613**
MLMV-DNN		0.631	0.737	0.881	0.783	0.597	0.500	0.688	0.742	0.121	0.193	0.501
ML-DNN		0.635	0.742	0.885	0.793	0.719	0.620	0.732	0.764	0.118	0.162	0.558
HEA-MV		0.665	0.760	0.883	0.803	0.724	0.832	0.778	0.780	0.111	*NA*	0.590
XGBoost		0.674	0.754	0.878	0.795	0.734	0.833	0.778	0.781	0.114	0.118	0.578
LR		0.652	0.752	0.883	0.801	0.702	0.832	0.770	0.772	0.116	0.119	0.578
MTMVL-CE	April	0.769	0.753	0.901	0.794	0.726	0.748	0.782	0.799	0.108	0.122	0.595
MLMV-DNN		0.703	0.735	0.898	0.776	0.620	0.500	0.705	0.756	0.115	0.160	0.530
ML-DNN		0.703	0.728	0.901	0.797	0.693	0.638	0.743	0.768	0.121	0.173	0.546
HEA-MV		0.729	0.754	0.901	0.802	0.745	0.749	0.780	0.793	0.105	/	0.597
XGBoost		0.756	0.754	0.893	0.800	0.743	0.749	0.782	0.797	0.104	0.104	**0.609**
LR		0.702	0.729	0.901	0.793	0.712	0.718	0.759	0.774	0.115	0.106	0.560
MTMVL-CE	May	0.759	0.684	0.886	0.791	0.720	0.767	0.768	0.773	0.123	0.136	**0.545**
MLMV-DNN		0.701	0.665	0.886	0.776	0.547	0.500	0.679	0.720	0.135	0.196	0.435
ML-DNN		0.703	0.679	0.885	0.786	0.692	0.675	0.737	0.748	0.134	0.192	0.484
HEA-MV		0.738	0.676	0.886	0.795	0.742	0.769	0.768	0.768	0.120	/	0.527
XGBoost		0.725	0.689	0.885	0.792	0.739	0.768	0.766	0.768	0.123	0.121	0.522
LR		0.733	0.671	0.886	0.787	0.708	0.769	0.759	0.760	0.126	0.120	0.515
MTMVL-CE	June	0.738	0.783	0.892	0.810	0.674	0.745	0.774	0.789	0.110	0.122	**0.577**
MLMV-DNN		0.689	0.748	0.895	0.773	0.557	0.500	0.693	0.742	0.123	0.176	0.487
ML-DNN		0.692	0.754	0.891	0.790	0.671	0.650	0.741	0.763	0.122	0.174	0.520
HEA-MV		0.708	0.770	0.885	0.814	0.688	0.662	0.755	0.778	0.112	/	0.558
XGBoost		0.713	0.766	0.887	0.819	0.693	0.703	0.764	0.783	0.113	0.110	0.548
LR		0.698	0.759	0.888	0.798	0.686	0.704	0.756	0.771	0.117	0.120	0.546
MTMVL-CE	January-February	0.738	0.747	0.872	0.809	0.795	0.764	0.788	0.797	0.106	0.124	**0.613**
MLMV-DNN		0.711	0.714	0.872	0.780	0.719	0.500	0.716	0.753	0.113	0.170	0.515
ML-DNN		0.711	0.716	0.871	0.795	0.778	0.651	0.754	0.774	0.112	0.148	0.558
HEA-MV		0.728	0.732	0.870	0.805	0.803	0.718	0.776	0.792	0.104	/	0.604
XGBoost		0.724	0.728	0.868	0.804	0.793	0.731	0.775	0.788	0.108	0.119	0.587
LR		0.724	0.722	0.868	0.796	0.810	0.718	0.773	0.789	0.106	0.119	0.599
MTMVL-CE	January-March	0.726	0.751	0.876	0.807	0.786	0.797	0.791	0.795	0.107	0.122	**0.613**
MLMV-DNN		0.688	0.722	0.876	0.781	0.697	0.500	0.711	0.750	0.115	0.176	0.511
ML-DNN		0.689	0.725	0.876	0.795	0.768	0.645	0.750	0.771	0.114	0.152	0.558
HEA-MV		0.710	0.742	0.874	0.804	0.789	0.743	0.777	0.789	0.106	/	0.600
XGBoost		0.710	0.737	0.872	0.802	0.784	0.753	0.776	0.786	0.110	0.119	0.585
LR		0.703	0.733	0.873	0.797	0.791	0.743	0.773	0.784	0.109	0.119	0.593
MTMVL-CE	January-April	0.738	0.753	0.882	0.804	0.779	0.786	0.790	0.796	0.107	0.122	**0.609**
MLMV-DNN		0.692	0.726	0.881	0.780	0.687	0.500	0.711	0.751	0.115	0.173	0.515
ML-DNN		0.693	0.727	0.882	0.795	0.758	0.643	0.750	0.770	0.115	0.157	0.555
HEA-MV		0.716	0.746	0.881	0.804	0.785	0.745	0.779	0.789	0.106	/	0.599
XGBoost		0.722	0.742	0.877	0.801	0.780	0.752	0.779	0.789	0.109	0.116	0.590
LR		0.703	0.732	0.880	0.796	0.781	0.737	0.772	0.782	0.110	0.116	0.586
MTMVL-CE	January-May	0.743	0.737	0.883	0.802	0.770	0.783	0.786	0.791	0.111	0.125	**0.596**
MLMV-DNN		0.694	0.711	0.882	0.779	0.664	0.500	0.705	0.745	0.119	0.177	0.499
ML-DNN		0.696	0.716	0.883	0.793	0.747	0.648	0.747	0.766	0.119	0.164	0.541
HEA-MV		0.721	0.729	0.882	0.802	0.778	0.749	0.777	0.785	0.109	/	0.585
XGBoost		0.723	0.729	0.879	0.800	0.774	0.755	0.776	0.784	0.111	0.117	0.577
LR		0.710	0.718	0.881	0.795	0.769	0.742	0.769	0.777	0.113	0.117	0.572
MTMVL-CE	January-June	0.742	0.745	0.884	0.803	0.759	0.778	0.785	0.791	0.111	0.124	**0.593**
MLMV-DNN		0.693	0.718	0.884	0.778	0.651	0.500	0.704	0.744	0.120	0.177	0.498
ML-DNN		0.696	0.723	0.884	0.793	0.739	0.648	0.747	0.765	0.119	0.166	0.538
HEA-MV		0.719	0.737	0.883	0.804	0.769	0.738	0.775	0.784	0.109	/	0.581
XGBoost		0.721	0.737	0.880	0.803	0.765	0.748	0.776	0.784	0.112	0.116	0.572
LR		0.708	0.726	0.882	0.795	0.760	0.738	0.768	0.776	0.114	0.117	0.568

Table B.6
Experimental results of the ablation study: our proposed MTMVL-CE without the introduction of multi-label cross-entropy loss (ML-CE loss) and the MV strategy. The evaluated metrics are balanced accuracy for each task (1: CASSETTE, 2:CT, 3:NE, 4:NF, 5:NV, 6:SHUTTER), macro balanced accuracy (macro-BA), micro balanced accuracy (micro-BA), Hamming Loss (HL), Ranking Loss (RL) and Subset Accuracy. The method achieving the highest SA is highlighted in bold.

Method	Test set	1↑	2↑	3↑	4↑	5↑	6↑	macro-BA↑	micro-BA↑	HL↓	RL↓	SA↑
MTMVL-CE	January	0.716	0.734	0.855	0.803	0.777	0.756	0.774	0.784	0.115	0.134	**0.601**
-ML-CE LOSS		0.708	0.715	0.855	0.784	0.775	0.715	0.759	0.766	0.145	0.183	0.526
-MV		0.717	0.723	0.857	0.808	0.785	0.729	0.770	0.784	0.119	0.143	0.577
MTMVL-CE	February	0.760	0.763	0.892	0.816	0.817	0.775	0.804	0.813	0.095	0.112	**0.627**
-ML-CE LOSS		0.759	0.736	0.891	0.794	0.821	0.716	0.786	0.795	0.127	0.157	0.551
-MV		0.748	0.747	0.896	0.813	0.842	0.763	0.801	0.812	0.103	0.130	0.611
MTMVL-CE	March	0.696	0.759	0.883	0.801	0.737	0.916	0.799	0.790	0.111	0.117	**0.613**
-ML-CE LOSS		0.657	0.753	0.883	0.780	0.738	0.786	0.766	0.767	0.140	0.174	0.546
-MV		0.661	0.760	0.882	0.805	0.735	0.910	0.792	0.783	0.118	0.143	0.584
MTMVL-CE	April	0.769	0.753	0.901	0.794	0.726	0.748	0.782	0.799	0.108	0.122	**0.595**
-ML-CE LOSS		0.708	0.735	0.900	0.802	0.705	0.590	0.740	0.765	0.148	0.174	0.506
-MV		0.716	0.740	0.903	0.802	0.715	0.733	0.768	0.782	0.121	0.136	0.556
MTMVL-CE	May	0.759	0.684	0.886	0.791	0.720	0.767	0.768	0.773	0.123	0.136	**0.545**
-ML-CE LOSS		0.733	0.683	0.885	0.756	0.745	0.654	0.743	0.748	0.164	0.210	0.455
-MV		0.751	0.675	0.885	0.797	0.724	0.753	0.764	0.767	0.132	0.165	0.515
MTMVL-CE	June	0.738	0.783	0.892	0.810	0.674	0.745	0.774	0.789	0.110	0.122	**0.577**
-ML-CE LOSS		0.708	0.763	0.888	0.772	0.705	0.629	0.744	0.760	0.153	0.192	0.489
-MV		0.710	0.780	0.887	0.815	0.704	0.771	0.778	0.784	0.119	0.150	0.556
MTMVL-CE	January-	0.738	0.747	0.872	0.809	0.795	0.764	0.788	0.797	0.106	0.124	**0.613**
-ML-CE LOSS	February	0.733	0.724	0.871	0.789	0.796	0.716	0.771	0.779	0.137	0.171	0.537
-MV		0.732	0.733	0.875	0.810	0.811	0.743	0.784	0.797	0.112	0.137	0.593
MTMVL-CE	January-	0.726	0.751	0.876	0.807	0.786	0.797	0.791	0.795	0.107	0.122	**0.613**
-ML-CE LOSS	March	0.711	0.734	0.875	0.786	0.786	0.732	0.771	0.776	0.138	0.172	0.540
-MV		0.712	0.743	0.877	0.809	0.798	0.780	0.786	0.793	0.113	0.139	0.591
MTMVL-CE	January-	0.738	0.753	0.882	0.804	0.779	0.786	0.790	0.796	0.107	0.122	**0.609**
-ML-CE LOSS	April	0.711	0.735	0.881	0.790	0.776	0.700	0.765	0.773	0.140	0.172	0.532
-MV		0.713	0.743	0.884	0.808	0.788	0.770	0.784	0.790	0.115	0.138	0.583
MTMVL-CE	January-	0.743	0.737	0.883	0.802	0.770	0.783	0.786	0.791	0.111	0.125	**0.596**
-ML-CE LOSS	May	0.716	0.723	0.882	0.783	0.771	0.692	0.761	0.768	0.145	0.180	0.517
-MV		0.722	0.727	0.884	0.806	0.778	0.767	0.781	0.786	0.118	0.144	0.570
MTMVL-CE	January-	0.742	0.745	0.884	0.803	0.759	0.778	0.785	0.791	0.111	0.124	**0.593**
-ML-CE LOSS	June	0.715	0.730	0.883	0.781	0.764	0.684	0.760	0.767	0.146	0.182	0.513
-MV		0.720	0.737	0.884	0.807	0.770	0.767	0.781	0.785	0.118	0.145	0.567

CRediT authorship contribution statement

Riccardo Rosati: Writing – review & editing, Writing – original draft, Methodology, Data curation, Conceptualization. **Lucia Pepa:** Writing – review & editing, Software. **Luca Romeo:** Writing – original draft, Supervision, Software, Project administration, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Riccardo Rosati reports financial support was provided by Ministero dello Sviluppo Economico (Italia). Other authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the research agreement between Università Politecnica delle Marche and Sigma S.p.A for the project "AI.EM - Artificial Intelligence Evolutive Machine" funded by Ministero dello Sviluppo Economico (Italia) - Fondo per la Crescita Sostenibile - Accordi per l'innovazione di cui al D.M. 31 December 2021.

Appendix A. Results

Table A.5 shows the experimental results of the proposed MTMVL-CE approach and state-of-the-art comparisons, reporting the BA metrics for each class.

Appendix B. Ablation study

Table B.6 shows the experimental results of the ablation study: our proposed MTMVL-CE without introducing multi-label cross-entropy loss (ML-CE loss) and the MV strategy. BA metrics are reported for each class.

Data availability

Data will be made available on request.

References

[1] Y. Liu, W. Yu, W. Rahayu, T. Dillon, An evaluative study on IoT ecosystem for smart predictive maintenance (IoT-SPM) in manufacturing: Multiview requirements and data quality, IEEE Internet Things J. 10 (13) (2023) 11160–11184, http://dx.doi.org/10.1109/JIOT.2023.3246100.

[2] S. Gawde, S. Patil, S. Kumar, P. Kamat, K. Kotecha, A. Abraham, Multi-fault diagnosis of industrial rotating machines using data-driven approach: A review of two decades of research, Eng. Appl. Artif. Intell. 123 (2023) 106139.

[3] J. Aldrini, I. Chihi, L. Sidhom, Fault diagnosis and self-healing for smart manufacturing: a review, J. Intell. Manuf. 35 (6) (2024) 2441–2473.

[4] C. Gutschi, N. Furian, J. Suschnigg, D. Neubacher, S. Voessner, Log-based predictive maintenance in discrete parts manufacturing, Procedia CIRP 79 (2019) 528–533, http://dx.doi.org/10.1016/j.procir.2019.02.098.

[5] R. Rosati, L. Romeo, G. Cecchini, F. Tonetto, P. Viti, A. Mancini, E. Frontoni, From knowledge-based to big data analytic model: a novel IoT and machine learning based decision support system for predictive maintenance in industry 4.0, J. Intell. Manuf. 34 (1) (2023) 107–121.

[6] Y. Chen, Z. Chen, S. Guo, Y. Zhao, Z. Liu, P. Wu, C. Yang, Z. Li, H. Yu, Efficient training of large-scale industrial fault diagnostic models through federated opportunistic block dropout, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, (13) 2023, pp. 15485–15493.

[7] Q.-T. Nguyen, T.N. Tran, C. Heuchenne, K.P. Tran, Decision support systems for anomaly detection with the applications in smart manufacturing: a survey and perspective, in: Machine Learning and Probabilistic Graphical Models for Decision Support Systems, CRC Press, 2022, pp. 34–61.
[8] M. Achouch, M. Dimitrova, K. Ziane, S. Sattarpanah Karganroudi, R. Dhouib, H. Ibrahim, M. Adda, On predictive maintenance in industry 4.0: Overview, models, and challenges, Appl. Sci. 12 (16) (2022) 8081.
[9] P. Nunes, J. Santos, E. Rocha, Challenges in predictive maintenance–a review, CIRP J. Manuf. Sci. Technol. 40 (2023) 53–67.
[10] Q. Li, Y. Wang, J. Dong, C. Zhang, K. Peng, Multi-node knowledge graph assisted distributed fault detection for large-scale industrial processes based on graph attention network and bidirectional LSTMs, Neural Netw. 173 (2024) 106210, http://dx.doi.org/10.1016/j.neunet.2024.106210.
[11] H.N. Noura, Z. Allal, O. Salman, K. Chahine, Explainable artificial intelligence of tree-based algorithms for fault detection and diagnosis in grid-connected photovoltaic systems, Eng. Appl. Artif. Intell. 139 (2025) 109503.
[12] V.M. Vargas, R. Rosati, C. Hervás-Martínez, A. Mancini, L. Romeo, P.A. Gutiérrez, A hybrid feature learning approach based on convolutional kernels for ATM fault prediction using event-log data, Eng. Appl. Artif. Intell. 123 (2023) 106463.
[13] S. Sun, A survey of multi-view machine learning, Neural Comput. Appl. 23 (2013) 2031–2038.
[14] X. Yan, S. Hu, Y. Mao, Y. Ye, H. Yu, Deep multi-view learning methods: A review, Neurocomputing 448 (2021) 106–129, http://dx.doi.org/10.1016/j.neucom.2021.03.090.
[15] B. Jiang, J. Xiang, X. Wu, W. He, L. Hong, W. Sheng, Robust adaptive-weighting multi-view classification, in: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, 2021, pp. 3117–3121.
[16] B. Jiang, C. Zhang, X. Liang, P. Zhou, J. Yang, X. Wu, J. Guan, W. Ding, W. Sheng, Collaborative similarity fusion and consistency recovery for incomplete multi-view clustering, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 39, (17) 2025, pp. 17617–17625.
[17] B. Jiang, C. Zhang, Y. Zhong, Y. Liu, Y. Zhang, X. Wu, W. Sheng, Adaptive collaborative fusion for multi-view semi-supervised classification, Inf. Fusion 96 (2023) 37–50.
[18] Z. Tong, W. Li, B. Zhang, H. Gao, X. Zhu, E. Zio, Bearing fault diagnosis based on discriminant analysis using multi-view learning, Mathematics 10 (3889) (2022) http://dx.doi.org/10.3390/math10203889.
[19] G. Jiang, C. Jia, S. Nie, X. Wu, Q. He, P. Xie, Multiview enhanced fault diagnosis for wind turbine gearbox bearings with fusion of vibration and current signals, Measurement 196 (2022) 111159, http://dx.doi.org/10.1016/j.measurement.2022.111159.
[20] S.V.V.S.N. Pichika, G. Meganaa, S. Geetha Rajasekharan, A. Malapati, Multi-component fault classification of a wind turbine gearbox using integrated condition monitoring and hybrid ensemble method approach, Appl. Acoust. 195 (2022) 108814, http://dx.doi.org/10.1016/j.apacoust.2022.108814.
[21] L. Gao, Y. Shen, L. Zhang, J. Zhang, Y. Wu, Z. Yang, MVGNet: Multi-view graph network with interactive shared fusion for fault diagnosis of wind turbines, Appl. Soft Comput. J. (2023) http://dx.doi.org/10.1016/j.asoc.2023.109849.
[22] X. Liu, J. Wang, S. Meng, X. Qiu, G. Zhao, Multi-view rotating machinery fault diagnosis with adaptive co-attention fusion network, Eng. Appl. Artif. Intell. 122 (2023) 106138, http://dx.doi.org/10.1016/j.engappai.2023.106138.
[23] N. Lu, Z. Cui, H. Hu, T. Yin, Multi-view and multi-level network for fault diagnosis accommodating feature transferability, Expert Syst. Appl. 213 (2023) 119057, http://dx.doi.org/10.1016/j.eswa.2022.119057.
[24] X. Wu, B. Jiang, K. Yu, H. Chen, C. Miao, Multi-label causal feature selection, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, (04) 2020, pp. 6430–6437.
[25] X. Wu, B. Jiang, Y. Zhong, H. Chen, Multi-target Markov boundary discovery: Theory, algorithm, and application, IEEE Trans. Pattern Anal. Mach. Intell. 45 (4) (2022) 4964–4980.
[26] X. Wu, B. Jiang, X. Wang, T. Ban, H. Chen, Feature selection in the data stream based on incremental markov boundary learning, IEEE Trans. Neural Netw. Learn. Syst. 34 (10) (2023) 6740–6754.
[27] M. Shcherbakov, C. Sai, A hybrid deep learning framework for intelligent predictive maintenance of cyber-physical systems, ACM Trans. Cyber-Physical Syst. (TCPS) 6 (2) (2022) 1–22.
[28] Y. Tan, J. Zhang, H. Tian, D. Jiang, L. Guo, G. Wang, Y. Lin, Multi-label classification for simultaneous fault diagnosis of marine machinery: A comparative study, Ocean Eng. 239 (2021) 109723, http://dx.doi.org/10.1016/j.oceaneng.2021.109723.
[29] X. Cao, B. Chen, N. Zeng, A deep domain adaption model with multi-task networks for planetary gearbox fault diagnosis, Neurocomputing 409 (2020) 173–190, http://dx.doi.org/10.1016/j.neucom.2020.05.064.
[30] Z. Xie, J. Chen, Y. Feng, K. Zhang, Z. Zhou, End to end multi-task learning with attention for multi-objective fault diagnosis under small sample, J. Manuf. Syst. 62 (2022) 301–316, http://dx.doi.org/10.1016/j.jmsy.2021.11.002.
[31] C. Yu, Y. Ning, Y. Qin, W. Su, X. Zhao, Multi-label fault diagnosis of rolling bearing based on meta-learning, Neural Comput. Appl. 33 (2021) 5393–5407, http://dx.doi.org/10.1007/s00521-020-05345-0.
[32] Z. Wang, Q. Zhang, L. Tang, T. Shi, J. Xuan, Transfer reinforcement learning method with multi-label learning for compound fault recognition, Adv. Eng. Inform. 55 (2023) 101818, http://dx.doi.org/10.1016/j.aei.2022.101818.
[33] X. Ma, Y. Hu, M. Wang, F. Li, Y. Wang, Degradation state partition and compound fault diagnosis of rolling bearing based on personalized multilabel learning, IEEE Trans. Instrum. Meas. 70 (2021) 3520711, http://dx.doi.org/10.1109/TIM.2021.3091504.
[34] M. Juez-Gil, J.J. Saucedo-Dorantes, Á. Arnaiz-González, C. López-Nozal, C. García-Osorio, D. Lowe, Early and extremely early multi-label fault diagnosis in induction motors, ISA Trans. 106 (2020) 367–381, http://dx.doi.org/10.1016/j.isatra.2020.07.002.
[35] J. Shen, S. Li, F. Jia, H. Zuo, J. Ma, A deep multi-label learning framework for the intelligent fault diagnosis of machines, IEEE Access 8 (2020) 113557–113567, http://dx.doi.org/10.1109/ACCESS.2020.3002826.
[36] P. Liang, C. Deng, J. Wu, Z. Yang, J. Zhu, Z. Zhang, Compound fault diagnosis of gearboxes via multi-label convolutional neural network and wavelet transform, Comput. Ind. 113 (2019) 103132, http://dx.doi.org/10.1016/j.compind.2019.103132.
[37] Y. Wang, M. Yang, Y. Li, Z. Xu, J. Wang, X. Fang, A multi-input and multi-task convolutional neural network for fault diagnosis based on bearing vibration signal, IEEE Sensors J. 21 (9) (2021) 10946–10956.
[38] J. Cui, P. Xie, X. Wang, J. Wang, Q. He, G. Jiang, M2FN: An end-to-end multi-task and multi-sensor fusion network for intelligent fault diagnosis, Measurement 204 (2022) 112085, http://dx.doi.org/10.1016/j.measurement.2022.112085.
[39] S. Guo, B. Zhang, T. Yang, D. Lyu, W. Gao, Multitask convolutional neural network with information fusion for bearing fault diagnosis and localization, IEEE Trans. Ind. Electron. 67 (9) (2020) 8005–8015, http://dx.doi.org/10.1109/TIE.2019.2942548.
[40] V. Dumoulin, E. Perez, N. Schucher, F. Strub, H.d. Vries, A. Courville, Y. Bengio, Feature-wise transformations, Distill 3 (7) (2018) e11.
[41] T. Evgeniou, C.A. Micchelli, M. Pontil, J. Shawe-Taylor, Learning multiple tasks with kernel methods., J. Mach. Learn. Res. 6 (4) (2005).
[42] J. Díez, O. Luaces, J.J. del Coz, A. Bahamonde, Optimizing different loss functions in multilabel classifications, Prog. Artif. Intell. 3 (2015) 107–118.
[43] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, Xgboost: extreme gradient boosting, 2015, pp. 1–4, R package version 0.4-2.
[44] Y. Xiao, Y. Huo, J. Cai, Y. Gong, W. Liang, et al., ERF-XGB: An edge-IoT-based explainable model for predictive maintenance, IEEE Trans. Consum. Electron. (2024).
[45] M. Paolanti, L. Romeo, A. Felicetti, A. Mancini, E. Frontoni, J. Loncarski, Machine learning approach for predictive maintenance in industry 4.0, in: 2018 14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications, MESA, IEEE, 2018, pp. 1–6.
[46] H.-L.E.G. on Artificial Intelligence, Ethics guidelines for trustworthy AI, 2019, https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai.