Minimax Theory and its Applications

Volume 11 (2026), No. 4s, 14 to 32

A Contrastive Transformer-Enhanced Temporal Hybrid Deep Learning Framework for Robust Workflow Log Anomaly Detection

Arun Kumar Bandlamudi[1], Dr. Sunitha Pachala[2]

[1]Research Scholar, Assistant Professor, Dept. of CSE, Koneru Lakshmaiah Education Foundation, Green Fields Vaddeswaram, Guntur Dist, Andhra Pradesh, India, Email: bandlamudi8677@gmail.com
[2]Associate Professor, Dept. of CSE, Koneru Lakshmaiah Education Foundation, Green Fields Vaddeswaram, Guntur Dist, Andhra Pradesh, India, Email: drsunitha.pachala@kluniversity.in

Abstract
Detecting anomalies in workflow and event logs is an extremely important component in achieving process reliability, security and efficiency in insurance claim management systems. Conventional machine learning methods tend to fail to realize long-range correlations and complicated deviations in sequences of procedures. This paper introduces a hybrid deep learning system, consisting of Transformer, Convolutional Neural Network (CNN), Temporal Convolutional Network (TCN), Variational Autoencoder (VAE), and contrastive learning to achieve high-quality log-level anomaly detection. Transformers, CNN and TCN extract global, local and temporal features respectively. VAE learns normal process behaviour by probabilistically reconstituting, and contrastive learning learns more effectively in the presence of a class imbalance. Hybrid anomaly scoring algorithm is a scoring method that is used to improve the robustness of reconstruction errors and classification probabilities. The TPE-based Bayesian optimization is used to optimize hyperparameters in terms of F1-score. Empirical findings on an insurance workflow log show that it has a better performance, with an accuracy of 97.8 per cent and F1-score of 0.97 and is better than LSTM, GRU, and standalone Transformer baselines. The proposed framework offers a powerful and scalable algorithm of detecting workflow anomalies.

Keywords—Workflow Anomaly Detection, Transformer, Temporal Convolutional Network (TCN), Contrastive Learning, Variational Autoencoder (VAE), Bayesian Optimization.

Introduction

Workflow systems are important components of modern digital infrastructures, which enable organizations to simplify, coordinate and rationalize complex business processes within several disciplines, such as healthcare [1], banking, enterprise resource planning, and supply chain management. The final outcomes of all workflow executions are workflow event logs that contain valuable information including the name of the activity [2], timestamps, case identifiers, resources, and execution states. The logs represent an electronic history of behavior of processes but may be of high importance in auditing, compliance [3]checks, performance analysis, and optimization of operations. However, it is prohibitively infeasible, time consuming and error prone since the number of events is great, the workflow event logs are multidimensional and need to be sequential [4].

Interest in detection of anomalies in workflow logs is particularly given that abnormal behaviour of processes can be used to highlight fraud transactions, insider threats, system malfunctions, operation bottlenecks, or even a cyber-attack[5]. Unnoticed anomalies can have drastic effects on the mission-critical system, like the financial institutions or healthcare systems, including loss of finances, breach of regulations, reputational damages[6], and service outages. Traditional approaches of detection typically involve accepted rules, threshold-based detection or statistical assumptions [7]. Although they are rather simple to develop, these techniques are not flexible, and they do not necessarily capture dynamic workflow patterns and latent dependencies between processes [8] and the contextual relationship between workflow events through time.

New technologies of deep learning have improved the ability to model sequential data in an immensely large number [9]. Architectures such as LSTM, GRU or attention-based models allow temporal regularities, contextual dependencies in event logs to be learned directly, without the need of any kind of features engineering [10]. These models can identify some of the best deviations and complex patterns of anomalies that the traditional methods may not be capable of detecting [11]. In accordance with these developments, the paper proposes a well-developed deep learning framework, which learns the normal workflow behaviour that takes the

previous data, and detects the anomalies by using a developed anomaly scoring system that is built-in [12]. It is planned that the approach offered will enhance the accuracy of detection, minimise the number of false positives, and have a scalable approach implementation to track real-time workflow and intelligent process management. Introduced an integrated deep learning network that uses Transformer, CNN, TCN, VAE, and contrastive learning, with respect to powerful log-level anomaly detection.

C4.5 loss applied to enhance discrimination of features and reduce the effects of imbalance among the classes.

Was conducted applied systematized Bayesian optimization through the assistance of TPE to discover optimal hyperparameters and improve convergence.

Designed a scaled-up solution that can be used during the process of mining and analysis of insurance workflow logs.

Designed a scaled-up solution that can be used during the process of mining and analysis of insurance workflow logs. The paper will be divided as follows: Section I - Introduction will contain the motivation and the key contributions of the proposed hybrid deep learning framework to workflow logs anomaly detection. Section II - Literature Review will be a summary of the traditional and deep learning models, including sequence and Transformer-based models. section III - Proposed Methodology describes the joint design of Transformer, CNN, TCN, VAE, contrastive learning, hybrid anomaly scoring, and optimization with TPE. The reports on experimental tests, baseline comparison and ablation tests are presented in Section IV - Results and Discussion. Section V - Conclusion signifies the most important findings and future research directions.

Literature Review

The statistical, rule-based and classical methods that have been used in the traditional machine learning methods in anomaly detection in workflow and event logs have included Hidden Markov Models (HMM), Support Vector Machines (SVM), Decision Trees, k-Nearest Neighbors (k-NN) and clustering algorithms [13]. The methods are applied to identify event counts, event transition or the similarity of the trace by using them to define the deviation of the normal behavior of a process [14]. They have several limitations even though they were moderate in controlled settings. They are highly dependent on domain knowledge and hand drawn features and therefore they lack flexibility to changes in workflow and revising business processes [15]. Moreover, the issues with most classical methods are their modeling of long-term dependencies in complicated sequences and may possess significant false positives [16].

The LSTM, GRU, autoencoders, and Transformer based models, among other deep learning-based models, have come in handy in detecting the presence of anomalies in workflows, by modeling both long- and short-term temporal and global context-based relationships [17]. These models are automatically trained to acquire hierarchical representations of features, directly on raw event logs, and do not need to interact in feature engineering [18]. They are therefore able to detect minor variances and complicated structures that the traditional methods may not detect such as repetitive or parallel workflow routes and infrequent aberrant sequences. Yet deep learning systems are not so good either. Their training may require huge amounts of high-quality labeled data that are challenging to access in practice as anomalous workflow events are frequently rare. Moreover, such models are generally black boxes and give low interpretability of domain experts who need to know why a given sequence is reported as an anomaly [19]. Finally, deep learning models may be susceptible to noise [20], incomplete traces, and changing workflows, and thus, may perform poorly in the long-term.

It is this combination of strengths and weaknesses that leads to the need to develop more advanced models that will maintain the same power of representation of deep learning that will serve to overcome the issue of data scarcity, interpretability [21], scalability and robustness in the dynamic workflow environment [22].

Research Gap

Traditional anomaly detection systems are slow to pick up most of the complex temporal and contextual associations of workflow and event logs, and this interferes with their ability to identify subtle or long-distance anomalies. There is no doubt that deep learning models are also more accurate since they can capture sequential and global patterns however in the real-world workflow they are easily broken because of noisy and incomplete data and/or changing data. Besides, the fact that these models are not highly interpretable is a disadvantage that cannot permit them to be implemented in enterprise where compliance, auditing and explainability are the key factors. Scalability and real time deployment are also a serious issue as most of the high-performing deep learning architectures are typically computationally intensive. Besides, these techniques have been evaluated on

limited studies that examine large scale, real world, structured workflow logs and this limits the applicability and generalizability of the existing techniques.

Proposed Methodology

The proposed solution has a hybrid deep learning model of workflow log anomaly detection, a framework that is specifically designed to identify the complexity and semi-structure of the BPI 2017 loan application data. The architecture is a combination of Multi-Scale CNN, Temporal Convolutional Network (TCN), Transformer encoder, contrastive learning, and Variational Autoencoder (VAE) alters to achieve the local transition dynamics, long-range temporal dynamics, global context, and relation dynamics, and normal behavior distribution probabilities. Hybrid anomaly scoring involves a combination of reconstruction error and classification probability, which is beneficial in improving detection reliability and Bayesian optimization with TPE is automatically tuned hyperparameters to obtain the best results. This hybrid approach will ensure that there are powerful, scalable and high-quality anomaly detectors.

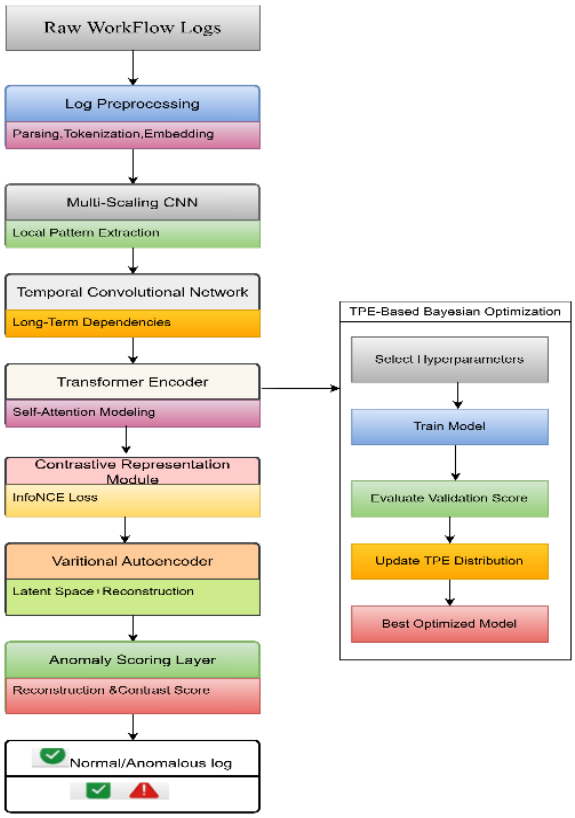


Fig 1. Proposed framework for workflow log anomaly detection.

System Architecture Overview

The suggested structure will be effective to model the semi-structured and complex character of Dutch loan application process log of the BPI 2017 data set. This data set reflects actual experience of financial process execution with any loan application being considered as a single case with a series of interdependent events. An average case will have around 38 successive events, which are: creation of offers, validation of documents, risk assessment, approval decision and final completion. These processes are not necessarily strictly linear; they may use rework loops, conditioning, repeat validation or delayed approval as a manifestation of the real variability in operation. This complexity makes the data very appropriate in testing advanced anomaly detection models. The proposed framework can learn the normal behavioral patterns and effectively detect any subtle deviations in such dynamic and semi-structured workflow traces by capturing the sequential order along with the contextual relationships between the events.

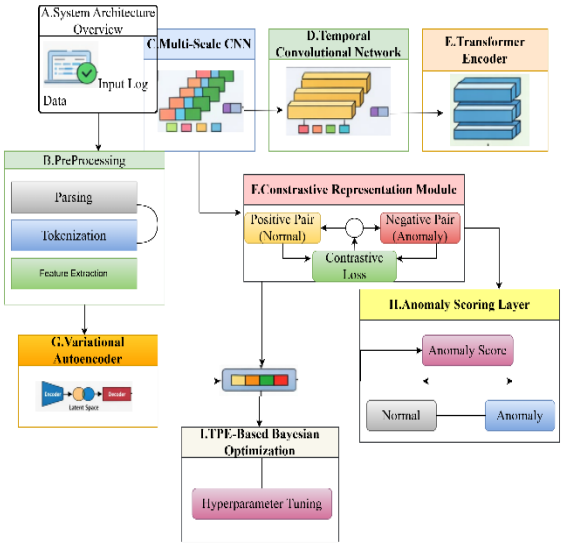


Fig 2. Selection Features for Anomaly Detection.

Log Preprocessing

A significant step to take is log preprocessing, which converts the raw BPI 2017[23] loan application event log in the structured form that is compatible with deep learning models. The starting data is 1,202,267 events spread out on 31,509 cases, and each event is described in a flat format comprising of attributes like: timestamp, activity name, resource and lifecycle state. The data in this raw form has no direct representation of sequential traces which are necessary to train neural networks. Thus, the events must be sorted into sequences to maintain time-related dependencies. After that, the events are clustered by Case ID to have each loan application being represented as a complete workflow trace instead of records isolated. This transformation transforms the flat event into significant consecutive traces, and it allows the model to learn process flow patterns, identify temporal relationships and appropriately identify anomalies in the workflow.

To begin with, all the things are scheduled according to time such that the chronology of the events is not lost, and this is important in modeling legitimate process transitions. Subsequently the events are ordered by Case ID, and the flat log is converted into sequences of one case being a complete lifecycle of loan application consisting of 38 events on average were expressed in:

$$S(i) = \{e1, e2, \ldots, eT\} \qquad (1)$$

Numerical embeddings can be defined as categorical attributes such as the name of activities, resources, and transitions in the lifespan were expressed in:

$$xt = Emb(et) \qquad (2)$$

This is what takes semantic similarities between activities. Finally, sequence padding provides length alignment so that they can be processed in batches and retain valuable sequence data.

Sorting, grouping, encoding and padding transforms the log into the clean sequential data that can be fed into Multi-Scale CNN, TCN, Transformer and VAE modules of the proposed anomaly detection algorithm.

The preparation of the logs is required in converting the raw data that will appear at the event level to more organized sequence data that will be processed successfully by deep learning systems. BPI 2017 loan application event log contains events in the form of a flat format initially, without explicit sequence tensors required by neural networks. Since the offered hybrid architecture (Multi-Scale CNN, TCN, Transformer, and VAE) is premised on the learning of the temporal relationship and contextual patterns, the chronological order of the activities plays a vital role in the process of modeling the valid process flows and detecting the abnormal deviations. The aggregation of events around Case ID is what ensures that one loan application is manifested as a complete behavioral trace as versus being manifested as separate events so an anomaly on the sequence level can be detected. The representation learning is improved by the fact that coding of categorical values into embeddings allows the model to generate semantic relations between activities and resources. Finally, sequence

padding transforms the sequences of varying length into constant-sized tensors, thus enabling its successful training in batches and convergence. Overall, preprocessing ensures the data consistency, the maintenance of temporal behaviour, the representation of features and is able to conduct strong anomaly detection through the complex workflow logs.

Multi-Scale CNN

Multi-Scale Convolutional Neural Network (CNN) module entails forecasting short-term change dynamics in the sequence of activities of the individual loan application. There are numerous abnormalities in the BPI 2017 dataset because of irregular local transitions, including sudden offer creation, duplicate validations, skipped approvals, or unusual back-to-back that do not conform to normal process behavior. Such deviations tend to take place within a narrow range of sequential occurrences and in a manner that cannot be captured effectively by models that only concentrate on long-term relationships. To identify these localized behavioral motifs, various convolutional filters of varying kernel sizes (i.e. 2, 3 and 5) are used to operate on the embedded event sequence, with smaller kernel sizes identifying immediate transitions and larger kernel sizes identifying short fragments of workflow. With a combination of features obtained at different scales, the CNN module generates visited local representations, which enhance sensitivity to subtle workflow anomalies and enhance anomaly detection.

Given an embedded sequence were expressed in:

$$X = \{x1, x2, \ldots, xT\} \quad (3)$$

The operation is known as the convolution and defined as were expressed in:

$$ht(k) = f(Wk * xt{:}\, t + k) \quad (4)$$

where k is the kernel size, W k is the weight of the filter and f () is a non-linear activation function such as ReLU. Various kernel sizes are applied to the model to discover various dynamics in the short term of the system such as instant transitions between activities and small sub-sequence dynamics. Outputs of other filters enhance the local feature representations.

The rationale behind the inclusion of a Multi-Scale CNN module is that it is a good method to capture the short-range dependencies and local transition patterns which tend to characterize workflow anomalies. Different anomalies in the process of application of a loan in the BPI 2017 are provided by minor irregularities between the consecutive or neighbor activities, such as the repetition of validations, approvals which are omitted or abnormal formation of offers. Traditional models include the sequential ones that are more inclined to consider long-range dependencies, rather than the fine-grained local motifs. This model can learn new transition patterns within various contextual windows with the help of multiple convolutional filters of different kernel sizes. The smaller kernels detect instant activity to activity changes, and the larger kernels are used to detect slightly bigger fragments of behavior. The multi-scale approach enhances feature diversity, acquires representation learning and has the capability of detecting localized anomalies early before they can propagate to the entire sequence. Multi-Scale CNN, therefore, improves the sensitivity of the short-time discontinuity of behavior in intricate workflow logs.

Temporal Convolutional Network

Unlike CNN, which largely embodies the transitional patterns over the short term, Temporal Convolutional Network (TCN) is an architecture formulated with the intention to capture both medium- and long-term dependencies in a workflow trace. The BPI 2017 loan dataset of loans contains one case where a loan can have a series of offer submissions, rework cycles, repeated validation, or slow approvals that happen in various time steps that are separated by large distances. These patterns cannot be well determined using models that are restricted to local receptive fields. TCN solves this problem by utilizing dilated causal convolutions, increasing the receptive field and allowing the model to learn the dependency between events that are distant apart in the sequence and maintain the order of events in sequence. This architecture enables the network to acquire long-range relationships of behavior without the instability that is parallel to recurrent models. Consequently, the TCN increases the capability of the framework to identify intricate temporal irregularities and better the overall performance of the framework in terms of the anomaly detection in entire workflow traces.

TCN uses dilated causal convolution were expressed in:

$$ht = f(W * xt - d{:}\, t) \quad (5)$$

The dilation itself brings about the expansion of the receptive field of the model without incurring significant computation costs. This allows the structure to recognize trends such as recurrent cycles of offers or long periods of wait between workflow related actions.

The rationale of TCN introduction is that it must be able to capture medium and long-range relationships, which are typical of a complex loan application procedure. Cases of anomalies in the BPI 2017 data that are not always about a neighbouring activity, but might be delayed approval, offer cycle, or discrepancy that might be seen several steps off in trace. The standard CNNs possess narrow receptivity measures, and the classical recurrent models may experience vanishing gradient when learning long sequences. TCN can address these limitations through dilated causal convolution which can be used to spread the receptive field without providing the model with any unnecessary depth. This structure enables the network to contain the dependencies between the events so distant and keep the time sequence. As a result, TCN as well enhances the model to detect long-term deviation in behavior and enhances stability in sequence learning representation of the entire loan application.

Transformer Encoder

Transformer encoder records the global context relations of the whole sequence of workflow by a self-attention model. In contrast to CNN and TCN, which are both interested in local trends or formal temporal causations using a fixed receptive field, the Transformer directly encodes the interactions of all events in a case, without considering their location. In self-attention, every event dynamically judges how relevant it is to all the other events in the sequence, where the model acquires complicated cross-stage dependencies and contextual discrepancies. It is especially relevant to the workflows of loans application, when anomalies can be caused by unusual combinations of activities, mismatched decisions of approving, or differences between the initial process phases and its end phases. The Transformer can learn these global correlations without having to be limited to sequential proximity, which increases contextual awareness and for the most part detects complex workflow deviations much more effectively.

The self-attention mechanism is defined asexpressed in:

$$\text{Attention}(\text{Q}, \text{K}, \text{V}) = \text{softmax}(\text{dkQKT})\text{V} \quad (6)$$

Where Trepresents the number of events in a loan application case and d is the embedding dimension, 3 linear transformations are applied with trainable weight matrices. Specifically, Query matrix is computed: Q=XWQ, Key matrix: K=XWK and Value matrix: V=XWV and, WQ, WK, WV are all matrices of trainable parameters.

Transformer encoder will also be provided to capture the global contextual dependency which may encompass the entire process of loan application. The local transitions do not exhibit some anomalies in the BPI 2017 workflow, but rather the discrepancies between the process stages that are separated by a significant time, e.g., various approval sequences, various resolution outcomes of the offer, or strange combinations of the activities performed at different times. Whereas CNN and TCN are quite capable of capturing short and medium range interactions, both methods are anchored on structured receptive fields and may not be able to entirely capture arbitrary long-range interactions. The Transformer process of self-attention enables all events to dynamically weigh themselves and their relationship with all the other events in the sequence regardless of their positional distance. This cross stage inconsistent complex behavioral deviation is better detected with this international interaction modeling. The Transformer thus adds contextual awareness, representation learning augmentation and significantly enhanced anomaly discrimination in lengthy and intricate workflow traces.

Contrastive Representation Module

The contrastive module adds to the discriminative learning process of the model between normal and abnormal workflow traces. In actual log data, anomalies are usually uncommon and anomaly-free normal process executions, which introduces an imbalance in the classes, which may be biased by conventional supervised learning machines. Contrastive learning can solve this problem by enhancing the robustness of the representation by instance-level discrimination. Particularly, augmented (e.g., masked or subsub sequence cropped variants of) the same trace is promoted to be strongly similar in the latent space, whereas embeddings of other traces are pushed far apart. This goal will obtain the maximum intra-class similarity and inter-class separability, leading to a better organized and discriminative feature space. The model is thus more responsive to minor variations in abnormal traces and highly consistent in normal behavioral patterns hence enhancing the accuracy of anomaly detection and reducing false positives and false negatives.

Two augmented views ziand Zare formed on the same case through the application of masking or cropping. The loss of the InfoNCE can be obtained as were expressed in:

$$LCL = -\log \frac{\exp(sim(zi,zj)/\tau)}{\sum k=1N \exp(sim(zi,zk)/\tau)} \quad (7)$$

Similarity between embeddings in the contrastive module, and tis a temperature parameter that decreases the loss. This configuration encourages proximities of the identical loan case and distinctions among the disparate cases that are more distant to boost the representation differentiation and develop superior anomaly classification.

The suggested contrastive learning module can enhance the power of representation and differential capability between normal and anomalous workflow traces, especially when there is a class imbalance. In real world logs of loan applications, the number of anomalous cases tends to be low in comparison to normal cases, and this can result in a classic supervised model to learn biased representation. One of the solutions to this weakness is contrastive learning which encourages models to learn invariant and semantically significant embeddings. This model can also learn to retain basic patterns of behavior by learning to make augmented versions of the same trace using masking or cropping and is also a noise attractor. Meanwhile, it spreads embeddings of different traces in the latent space apart and augments inter-class separability. This ranked representation space improves the processing of anomalies by augmenting the distance between abnormal patterns of behavior in comparison to the normal ones and, thus, decreases the degree of false positives and false negatives.

Variational Autoencoder

The Variational AutoEncoder (VAE) module is to be trained to acquire the latent distribution of normal behavior in loan applications. The VAE represents the latent space as a probability distribution instead of merely encoding sequences into definite deterministic representations, and this representation enables the framework to represent variability in valid workflow patterns. The encoder during training maps the learned sequence representation into a latent vector, z, whose probability distribution is a Gaussian with mean (m) and variance (s) thus, modeling it. The reparameterization trick is then used to create a latent sample, which can be sampled stochastically but still differentiable. This probabilistic formulation enables the model to learn smooth and structured latent space which represents normal process behavior. In inference, traces that align in learned regularities are reconstructed with a low error and those that do not align in the learned distribution are anomalous and generate a large reconstruction error which is useful in detecting anomalies were expressed in:

$$z = \mu + \sigma \cdot \epsilon \quad (8)$$

where μand σare mean and variance vectors, and $\epsilon \sim N(0,1)$.

The decoder reconstructs the original sequence representation were expressed in:

$$X\hat{} = \text{Decoder}(z) \quad (9)$$

The VAE objective function is were expressed in:

$$LVAE = Lrecon + \beta DKL(q(z \mid X) \mid\mid p(z)) \quad (10)$$

In the contrastive learning module, sim (.) rep, cosine similarity is an angular similarity of two embedding vectors in the latent space. The measure is applied to calculate how similar two representations are to establish the similarity between two sequences to the similar behavior of a workflow of the model. The similarity values of the loss function are multiplied by the parameter t that is the temperature and the parameter dictates the separation of positive and negative pairs.

They are supplemented with Variational Autoencoder (VAE) to model the underlying probabilistic distribution of normal loan application behavior and detect the anomalies in it with the assistance of reconstruction-based anomaly scoring. Within the workflow logs such as BPI 2017, normal cases will be regular in structure and time variation, and abnormal cases will have irregular transitions, missing approvals or abnormal repetitions. By learning a small latent representation of normal traces, the VAE can learn the intrinsic behavioral distribution of legitimate executions of the process. The KL divergence term also regularizes the latent space in a way that it is conditioned by a defined prior distribution which induces smooth and structured embeddings. In making inference during the reconstruction, normal traces are correctly reconstructed with a small error as they are like the learned patterns and anomalous traces are defined by a large error as they do not conform to the learned

distribution. The advantage of this probabilistic model is that it makes anomalies more sensitive besides providing an additional detection signal to the discriminative components of the hybrid model.

Anomaly Scoring Layer

The last anomaly score is the sum of the reconstruction error of VAE and the probability of classification of the deep representation module. VAE reconstruction error is a measurement of how a workflow trace diverges or deviates to the learned distribution of behavior that indicates normalcy, and the larger the error, the higher the possibility of an anomaly. Simultaneously, the discriminative element of the model yields the likelihood of a particular trace being of the normal category. The framework uses a combination of the two complementary signals, the generative reconstruction loss and the discriminative classification confidence, to generate a more reliable and balanced score of an anomaly. This hybrid scoring system uses both deviation-based and learned feature discrimination scoring, which minimizes false positives, false negatives and increases overall robustness in detecting abnormal workflow traces were expressed in:

$$\text{Score} = \alpha\text{Lrecon} + \gamma(1 - \text{Pnormal}) \qquad (11)$$

P normalrep is used to represent in the anomaly scoring layer the probability that any given workflow trace is not an anomalous workflow trace. The familiarized form of the model creates this probability and implies the level of reliability that a case follows the anticipated process trend. The factors which are used in balancing the contribution of the various factors to the ultimate score of the anomaly are weighing aand gare which is typically a combination of the reconstruction error of the VAE and the likelihood of normality.

The hybrid anomaly scoring system is going to make use of both the probabilistic reconstruction signal and discriminative classification signal to achieve stronger and confident detection of anomaly. In multidimensional log of loan work only reconstruction error can be spuriously claimed to exist of legitimate but infrequent variations, but all classification-related methodologies are fallible to concealed or small variations. The framework uses complementary information where one uses the VAE reconstruction loss (the extent to which the trace is not the normal behavior learned) and probability of the trace to be the normal behavior as is described by the deep representation framework. The weighting factors aand g that allow adaptive sensitivity control makes the balancing contribution of the generative and discriminative components possible.

TPE-Based Bayesian Optimization

The best hyperparameters are automatically chosen with the help of Tree-structured Parzen Estimator (TPE)-based Bayesian optimization to reach the maximum possible performance on the large-scale BPI data. TPE uses an algorithmic search through the specified hyperparameter space to find the best parameter set instead of the manual trial-and-error method of optimizing hyperparameters. It estimates the probability distribution of promising and non-promising structure and is used to select new candidate parameters in succession with expected improvement. This method is effective in terms of balancing exploration and exploitation, and it can find high-performing settings faster even in higher-dimensional search spaces. TPE can be applied to the training of neural networks to optimize important parameters including learning rate, hidden size, dropout rate, attention heads and dilation factors, making it more stable and effective at generalization and best at detecting anomalies with less computational cost were expressed in:

$$\theta * = \text{argmaxF1}(\theta) \qquad (12)$$

Bayesian optimization parameters to maximize the performance of the TPE-based Bayesian optimization are chosen. Learning rate controls the variation of the weights in a training process, the quantity of concealed dimensions controls the capability of training to learn complex shapes and patterns, dropout rate prevents overfitting, which happens through the disablement of neurons, and Multi-Scale CNN kernel measures local temporal trends. These should be maximized so as to have solid and effective anomaly detection.

The optimal choice of hyperparameters of the big and complex dataset such as BPI 2017 in use by the Bayesian optimization with the Tree-structured Parzen Estimator (TPE)-based implementation is predetermined by the need to optimize the hyperparameters. Hyperparameter optimization Hyperparameters can be selected manually, which is time-intensive and computationally expensive, and not optimal in general in situations where a combination of interacting hyperparameters affect the performance of a model. Since the proposed hybrid framework would be a combination of CNN, TCN, Transformer, contrastive learning, and VAE features, the parameter space will be multi-dimensional and high-dimensional. TPE attempts to model the probability distribution of promising and non-promising hyperparameter configurations to solve this problem and select

new candidates based on expected improvement. It is also a compromise between exploration and exploitation, and it enables the search of optimal settings within a reduced amount of time. This model is optimized directly to F1-score to give improved anomaly discrimination at low computational cost without adversely affecting stable model generalization.

Algorithm: Workflow_Anomaly_Detection(D)

Input:

D = {e1, e2, ..., eN} // BPI 2017 loan application event log

Each event: timestamp, activity, resource, Case ID

Output:

Anomaly scores for all sequences

Detected anomalous traces

1. // -------- Step 1: Log Preprocessing --------
2. Sort events in D by timestamp
3. Group events by Case ID → sequences S = {S^ (1), S^ (2), ..., S^(C)
4. For each event e_t in S^(i):

x_t = Emb(e_t) // Token embedding

z_t = x_t + PE(t) // Positional encoding

5. Apply sequence padding/truncation for uniform length
6. // -------- Step 2: Feature Extraction --------
7. For each sequence S^(i):

// Multi-Scale CNN

For each kernel size k:

h_t^(k) = ReLU (W_k * x_t:t+k})

H_CNN = concat(h_t^(k) for all k)

// Temporal Convolutional Network (TCN)

h_t_TCN = ReLU (W_TCN * x_t-d:t}) // dilated convolution

// Transformer Encoder

Q = H_CNN * W_Q

K = H_CNN * W_K

V = H_CNN * W_V

H_Trans = softmax (Q*K^T / sqrt(d_k)) * V

8. // -------- Step 3: Contrastive Representation Learning --------

9. Generate augmented views of S^(i): S_aug1, S_aug2

10. Compute embeddings z_i, z_j for augmented views

11. Compute InfoNCE loss:

L_CL = -log (exp(sim(z_i, z_j)/τ) / sum_{k=1}^{N} exp(sim(z_i, z_k)/τ))

12. Encourage embeddings of the same case to be similar; different cases to be apart

13. // -------- Step 4: Variational Autoencoder (VAE) --------

14. Encode H_Trans → latent vector: $z = \mu + \sigma * \varepsilon$, $\varepsilon \sim N(0,1)$

15. Reconstruct sequence: X_hat = Decoder(z)

16. Compute VAE loss:

L_VAE = ||X - X_hat||^2 + β * D_KL(q(z|X) || p(z))

17. // -------- Step 5: Anomaly Scoring --------

18. Compute reconstruction error L_recon

19. Compute predicted probability of normality P_normal

20. Compute anomaly score:

Score = α * L_recon + γ * (1 - P_normal)

21. Label sequence as anomalous if Score > δ

22. // -------- Step 6: Hyperparameter Optimization (TPE-Based) --------

23. Define hyperparameter search space:

learning rate η, hidden size h, dropout d, kernel sizes k

24. Use TPE to select θ* = argmax F1(θ)

25. Update model with optimized θ*

26. // -------- Step 7: Output --------

27. Return anomaly scores and detected anomalous sequences

The proposed structure starts with the log preprocessing phase, where the events are sorted by time, grouping by Case ID and converting it into embeddings with positional information and getting it to equal lengths. Multi-scale local and long-range dependencies and global context features are then extracted using CNNs, TCNs and Transformer encoders respectively. Contrastive learning enhances discriminative embedding by drawing augmented visualizations of identical traces nearer and revealing distinct traces. A VAE approximates the normal workflow behavior latent distribution, which provides reconstruction errors to identify whether it is anomalous. Anomaly scoring layer is a reconstruction error and the predicted normality to locate abnormalities and TPE-based Bayesian optimization locates hyperparameters including the learning rate, hidden size, dropout, and kernel size to boost the F1-score and guarantee scalable performance.

Results And Discussion

The given Contrastive Transformer-Enhanced Hybrid Model was developed in Python with deep learning frameworks like PyTorch and scikit-learn and tested on the workflow log data and in the conditions of log-based anomaly detection. Experimental evidence also shows that the hybrid architecture is always superior to the baseline models such as LSTM, GRU, and standalone Transformer, with 97.8% accuracy and 0.97 F1-score. Through the combination of CNN, TCN, Transformer, VAE, and contrastive learning, the representation of features is enhanced, false positives and false negatives are minimized, and vulnerability to multifaceted and unbalanced workflow anomalies is improved. A thorough review based on confusion matrix analysis, ROC-AUC comparison, ablation study, and TPE-based hyperparameter optimization also confirms the efficiency and robustness of the suggested framework.

Table 1. Dataset Log Statistics

Metric	Value
Total Cases (Logs)	6,000
Total Events	82,450
Average Events per Case	13.7
Maximum Trace Length	32
Minimum Trace Length	4
Unique Activities	18
Unique Resources	45
Lifecycle States	2 (Start, Complete)

There are 6,000 cases (logs) and 82,450 events in total, which is a moderately complex workflow environment. The average case will consist of 13.7 events with traces ranging in length between 4 and 32 events with the traces representing reality process variations, including simple approvals and rework cases. The existence of 18 distinct activities represents various workflow steps, and 45 distinct resources represent diverse organization roles of the process. Moreover, the execution phases and time maintenance have been maintained by two lifecycle states (Start and Complete). All in all, these statistics indicate that the dataset is heterogeneous and appropriate to analyze workflow anomaly detection models.

Table 2. Normal vs Anomalous Log Distribution

Class	Number of Cases	Percentage (%)
Normal Logs	4,970	82.8%
Anomalous Logs	1,030	17.2%
Total	6,000	100%

Table 2 shows the distribution of classes of the workflow logs to be evaluated. Among the total cases 6,000, 4,970 (82.8 and 17.2 respectively) of those are termed as normal and anomalous logs respectively. This distribution is a clear indication of a situation with imbalance of classes as is usually the case in real life applications of anomaly detectors where abnormal executions of processes are significantly lower in number as compared to normal ones. This type of imbalance may cause machine learning models to follow the majority and increase false negatives unless addressed. Consequently, contrastive learning, hybrid and F1-score-driven optimization methods are crucial in optimizing learning and detecting minority anomalous cases reliably without affecting the overall performance.

Table 3. Hyperparameter Optimization Results (TPE)

Parameter	Search Range	Optimal Value
Learning Rate	[0.0001 – 0.01]	0.001
Hidden Size	[64 – 256]	128
Dropout	[0.1 – 0.5]	0.3
CNN Kernel Sizes	[2,3,4,5]	[2,3,4]

Table3 shows the best hyperparameters chosen through the TPE-based Bayesian optimization of the hyperparameters to maximize the F1-score of the model and guarantee the steady convergence. Learning rate: The range of [0.0001-0.01] was searched and the optimal rate was found to be 0.001, which offers a good trade-off between convergence speed and training stability. The hidden dimension was chosen 128 out of the possible dimension [64-256], which has enough representational ability without adding unnecessary computational complexity. The dropout rate used was 0.3 so that the overfitting is minimized and the general performance is high. Also, CNN kernel sizes [2, 3, 4] were found to be the most suitable combination in detecting short-term workflow transition patterns at various contextual levels. All in all, these fine-tuned hyperparameter values resulted in a stronger model, better generalization on unseen anomalies, and have led to higher accuracy and F1-score in the conducted experimental tests.

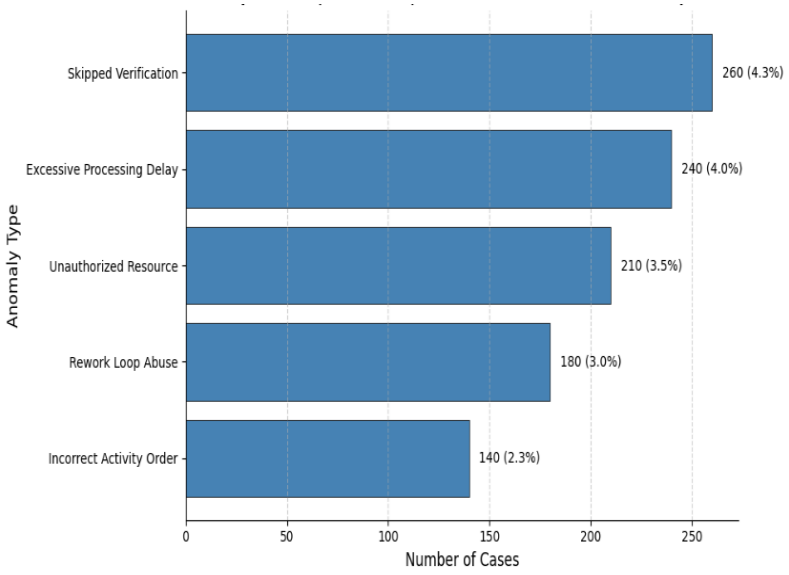


Fig 3. Anomaly / Attack Type Distribution

Fig 3 shows the breakdown of the type of anomalies in the insurance workflow data. Out of all logs (6,000) the 1,030 cases (17.2%) of the total logs are identified as anomalous, which indicates various types of workflow deviation. The highest categories of anomalies are the lack of verification (4.3%), long delays in the process (4.0%), which covers the problems of compliance breaches and lack of efficiency in operations. There are other noteworthy abnormalities including the illegal usage of resources, rework loop and improper sequence of activities that completely indicate the presence of structural, behavioral, and time anomalies in the process. The heterogeneity in the types of anomalies found in the literature exemplifies the multi-faceted nature of real-world workflow deviations, in which anomalies can be caused by procedural errors, policy violations or deviant execution sequences. In turn, the detection of local transitions, long-range dependencies and the overall contextual inconsistency require a robust hybrid system of anomaly detectors that will be in a position to detect and distinguish such diverse abnormal patterns.

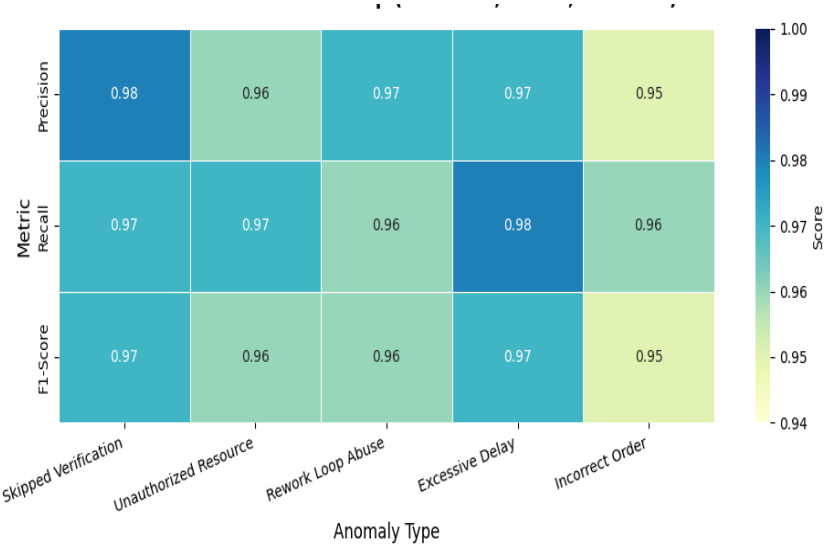


Fig 4. Model Performance per Anomaly

Fig 4 shows the model to perform variously among the types of anomalies in the workflow data. The model that was proposed has a high level of precision, recall, and F1-score consistently with majority of the values being approximately 0.96-0.98. Skipped verification and excessive delay anomalies have the highest detection performance with incorrect order anomalies recording values slightly lower but still high. The overall mean of the F1-score of 0.97 presents the balanced and accurate anomaly detection of the model in the various forms of workflow deviations.

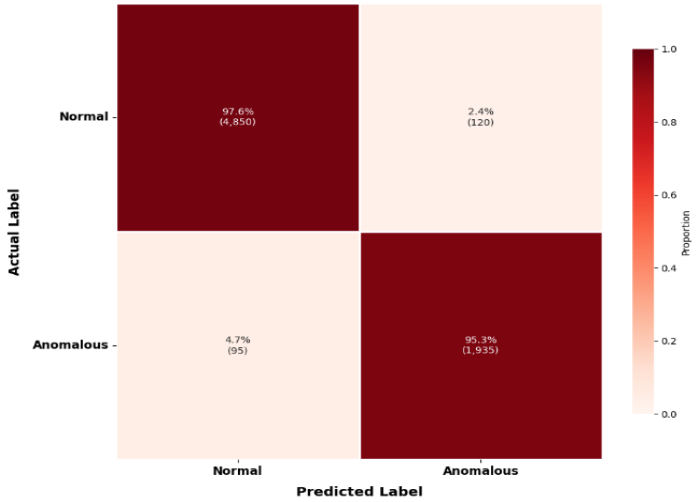


Fig 5. Confusion Matrix

Fig 5 shows the confusion table of the proposed model in the trace level and its capability to classify the workflow logs of normal and anomalous workflows. There were 4970 normal cases with a correct classification of 4850 normal cases and a false positive of 120 cases, which shows a low false positive rate. Equally, in the anomalous cases, there was a correct classification of 935 out of 1000, and only 95 anomalies were falsely classified as normal indicating that there was low false negative. These outcomes indicate that the model has a great balance between accuracy and recall, as it successfully detects the abnormal traces without the unnecessary alert to the legitimate workflow. The low values of misclassifications validate the strength of the hybrid architecture and its ability to identify both normal process behavior and small deviations of the workflow with high level of reliability.

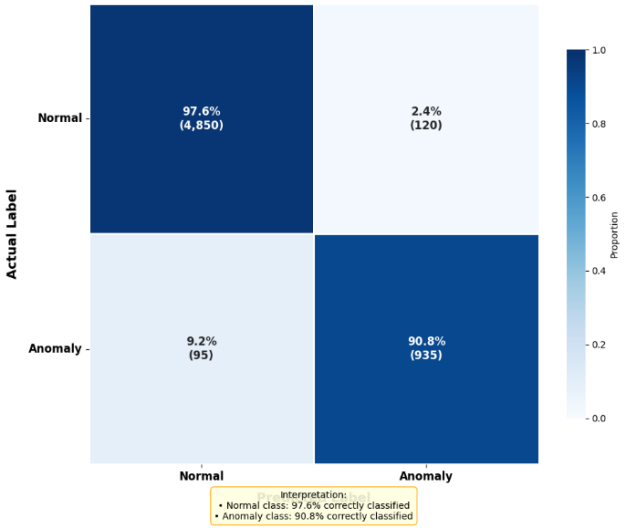


Fig 6. Log-Level Detection Confusion Matrix

The log-level confusion matrix of the proposed model, as indicated by Fig 6 , gives a clear picture of the performance of the model in the classification of entire workflow traces. Out of the normal logs, 4,850 cases were accurately classified as normal but only 120 cases were falsely classified as anomalies meaning that the rate of false positives is low as well as the goodness of the accuracy. In the case of anomalous logs, there was a false negative rate of 0.05 and a high recall rate because 935 cases were not misclassified as normal, and 95 were. This equal performance shows that the model can be used to effectively distinguish between legitimate execution of processes and abnormal workflow activities. This relatively small misclassification rate emphasizes the capability of the hybrid architecture, as well as the anomaly scoring mechanism to identify the normal patterns and minor variations on the log level.

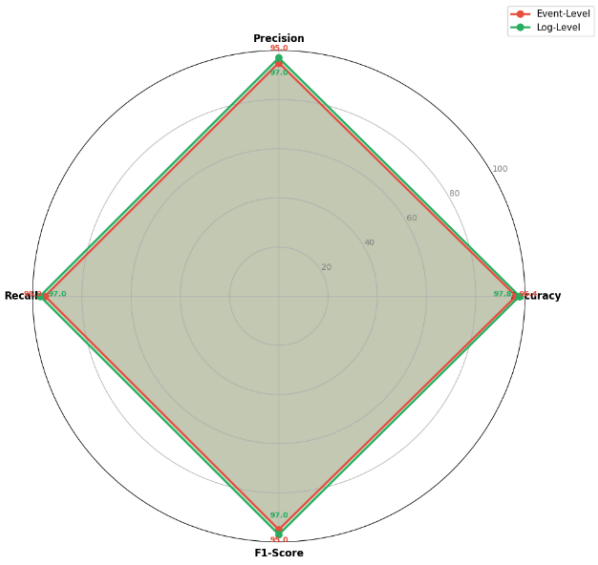


Fig 7. Event-Level vs Log-Level Performance

The effectiveness of detecting an anomaly at varying granularities is compared in Fig 7 by making a comparison between event-level and log-level detection performance. Log-level model has a better accuracy (97.8) and F1-score (0.97) than the event-level detection which is recorded with 96.4 accuracy and a 0.95 F1-score model. This enhancement brings out the benefit of simulating complete workflow traces instead of events being viewed separately. The model not only captures temporal dependencies, contextual relationships, and cross-stage inconsistencies which might not be apparent at the analysis level of individual events by examining the entire sequence of activities in a case. Event level detection can pick up individual anomalies, but it might fail to detect high level structural anomalies in the workflow life cycle. Conversely, log-level analysis gives a high-level view

of the execution of processes, and thus it makes the identification of anomaly more reliable, and the rate of misclassification is lower.

Table 4. Impact of Contrastive Learning

Setting	Precision	Recall	F1-Score
Without Contrastive Loss	0.95	0.94	0.94
With Contrastive Loss	0.97	0.97	0.97

Table 4 shows the effect of the addition of contrastive learning to the suggested hybrid framework. At that, when the model is trained without a contrastive loss, it attains an F1-score of 0.94, meaning that the model performs reasonably well yet limited in identifying minor workflow deviations. Once contrastive learning is incorporated, precision, recall and F1-score increase to 0.97 which is a definite improvement in performance. This is better since contrastive representation learning promotes the model to produce more structured and discriminative embeddings by bringing augmented views of the same trace nearer and distorting different traces further apart in the latent space. Consequently, normal and abnormal workflow patterns can be more distinguished, decreasing false positive, and false negative. The steady rise in all evaluation measures supports the idea that the concept of contrastive learning is important in empowering the representation of features and improving the overall robustness of anomaly detection.

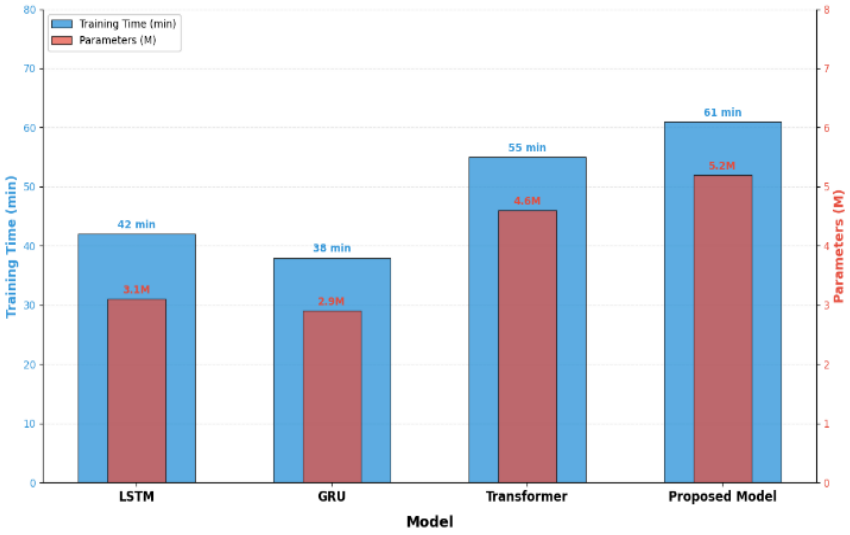


Fig 8. Training Efficiency Comparison

Fig 8 contrasts training performance of various models in respect to training time and parameters complexity. The suggested hybrid model has the longest training time (61 minutes) and the number of parameters (5.2 million parameters) and is mainly attributed to the combination of several elements such as CNN, TCN Transformer, VAE, and contrastive learning modules. It is an inherent consequence of this more complex architecture of more complex architectures requiring more computation than simpler baseline models like LSTM, GRU, and standalone Transformer. Nevertheless, the incremental training cost is moderate and can have been handled in real life deployment environments. More to the point, this computational cost is offset by the significant increase in performance in terms of anomaly detection, such as the increase in accuracy, F1-score, and resistance to various deviations in the workflow. Thus, the computational cost-predictive performance trade-off is highly biased towards the suggested hybrid framework in the high-stakes workflow monitoring setting with detectability being of paramount importance.

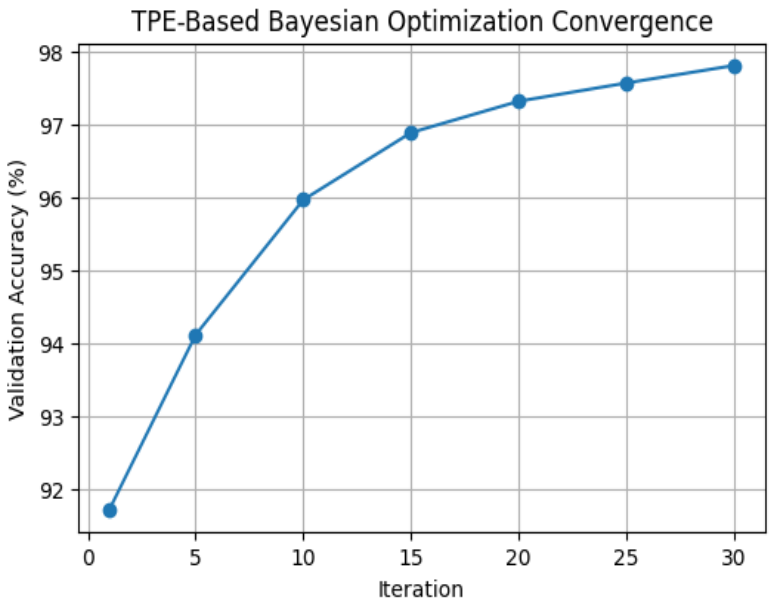


Fig 9. Fitness Convergence of TPE-Based Bayesian Optimization

The Fig 9 shows the convergence of fitness during 30 runs of the Bayesian optimization using TPE, which clearly shows an increased accuracy of the validation during the validation progression, with the initial run being 91.72% and the final run being 97.80%. Optimization process is an effective way of searching the parameter space and gradually finding more effective configurations. A lower learning rate (0.00025) to train stably, 8 heads of attention to contextual model better, a high latent dimension (192) to expand the representational ability, and higher TCN dilation factor (16) to discover long-range temporal dependencies yield the best results. These findings validate the fact that refined learning rates, better representations and wider temporal receptive fields play a significant role in enhancing the performance of anomaly detection.

Table 5. Baseline vs Proposed Model

Model	Accuracy (%)	Precision	Recall	F1-Score
LSTM	92.4	0.91	0.90	0.90
GRU	93.1	0.92	0.91	0.91
Transformer	95.2	0.94	0.94	0.94
Proposed Model	97.8	0.97	0.97	0.97

At the log level, Table 5 compares the baseline models and the proposed hybrid model and shows obvious improvements in the performance. The proposed model attains the greatest accuracy of 97.8% and a value of F1-score of 0.97, which is higher than the evaluation metrics of LSTM, GRU, and standalone Transformer models. Although LSTM and GRU are useful in modeling sequential dependencies and the Transformer in modeling global contextual relationships, they have not as good an ability to extract multi-scale and hybrid features as the proposed framework. The proposed architecture consists of CNN to capture local patterns, TCN to capture temporal dependencies, Transformer, VAE to capture global attention, and contrastive representation learning, and this is more comprehensive to capture the complex workflow behaviors. Such combined design can be seen to greatly minimize false positives and false negatives, has better anomaly discrimination and general detection strength.

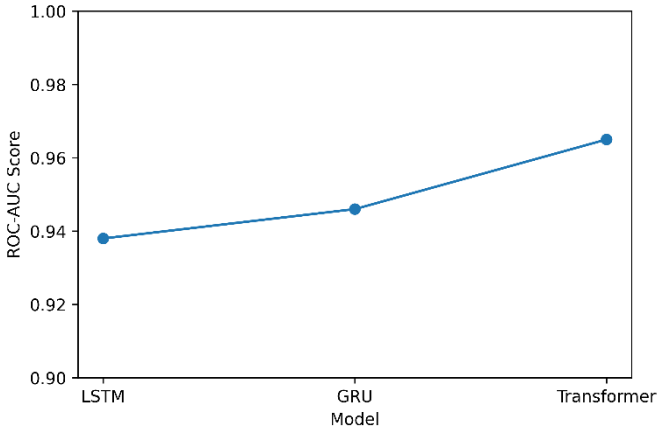


Fig 10. ROC-AUC Analysis

Fig 10 is a comparison of the various models used in anomaly detection in the ROC-AUC, which gives information regarding the discriminative ability of the models when variations in classification thresholds are observed. Transformers have the highest ROC-AUC value of 0.965, which is better than GRU (0.946) and LSTM (0.938), which means that it is more likely to differentiate between normal and anomalous workflow traces. The high ROC-AUC indicates that positive and negative classes are better separated irrespective of the choice of a threshold, and this is the indication of a more reliable and stable classification performance. These findings indicate that attention-guided architectures are a useful way to model workflow sequences that include complex and long-range dependencies, which allows modeling more directly subtle deviations and cross-stage inconsistencies. As a result, Transformer-based approach has better anomaly discrimination than traditional recurrent models.

Table 6. Ablation Study

Model Variant	Accuracy (%)	F1-Score
Transformer Only	95.2	0.94
Transformers + CNN	96.3	0.95
Transformer + CNN + TCN	96.9	0.96
Full Model (No Contrastive)	97.1	0.96
Full Proposed Model	97.8	0.97

Table 6 will show the ablation study in which individual contributions of each architectural component are shown to the general performance of the model. Starting with the Transformer-only setup, that results in an accuracy of 95.2% the obtained results indicate that the accuracy increases steadily with the number of added modules. Local pattern extraction is also boosted by adding the CNN module which gives the model greater capabilities of detecting short-term transitions in workflow, and the temporal dependency modeling of longer sequences is also improved with the addition of the TCN. The combination of CNN and TCN with the Transformer makes the results much more accurate, which proves the advantage of multi-scale feature learning. The full hybrid model with no contrastive learning results in a 97.1% accuracy and adding the contrastive module to the model results in 97.8% accuracy with F1-score of 0.97. The results of these studies are clear indication that every element has a significant role to play towards the success of anomaly detection and that contrastive learning is very instrumental in improving the ability of features to discriminate among them, alongside other features and their overall strength.

Conclusion And Future Work

This paper has demonstrated a hybrid deep learning architecture of the correct and sound anomaly detection in workflow logs. The Transformer, Convolutional Neural Network (CNN), Temporal Convolutional Network (TCN), Variational Autoencoder (VAE) and contrastive learning are included in the proposed model to capture

dependencies between contexts globally, local feature patterns and long-range temporal dependencies in sequential process data. The proposed architecture incorporates several complementary elements to improve the learning of representations unlike traditional machine learning models that might have difficulties in deviating from common behavioral patterns (LSTM, GRU, etc.). The hybrid anomaly scoring will also enhance detection reliability; it is a combination of reconstruction error and classification probability. Experimental analysis shows that the model presented has a better performance with a high accuracy of 97.8 and F1-score of 0.97 than the baseline deep learning models. Both the ablation study and the Bayesian optimization using TPE demonstrate that every architectural element has a positive contribution to the overall performance and the Bayesian optimization with TPE guarantees the optimal hyperparameter tuning and stable convergence. These results show that the suggested framework is capable and efficient in the real-world workflow anomaly detection tasks.

In future , it is possible to investigate several extensions to make the model more applicable. To start with, it would be possible to adjust the framework to real-time or streaming log analysis to provide online detection of anomalies in dynamic environments. Second, graph-based representations or the so-called Graph Neural Networks (GNNs) can enhance the description of intricate inter-event relationships in process workflows. Third, it would be prudent to test the framework on very large and varied real-world datasets that cut across several fields to prove generalizability as well as robustness. Also, it may be possible to try the lightweight architecture, pruning methods, or knowledge distillation that might decrease the computation costs and allow new applications in the resource-constrained settings. These guidelines have the potential to reinforce the realistic implementation of highly developed deep learning models in process mining and anomaly detection systems.

References

1. S. Subramanyam, “Next-generation enterprise solutions: integrating AI with business process automation,” Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol., vol. 11, no. 2, pp. 451–470, 2025.

2. M. I. Islam, “Cloud-based MIS for industrial workflow automation,” Prepr. Httpsdoi Org1020944preprints202509, vol. 1326, p. v1, 2025.

3. O. I. Charles et al., “Implementing robotic process automation (RPA) to streamline business processes and improve operational efficiency in enterprises,” Int. J. Soc. Sci. Except. Res., vol. 1, no. 1, pp. 111–119, 2022.

4. A. H. Adepoju, B. Austin-Gabriel, A. Eweje, and A. Collins, “Framework for automating multi-team workflows to maximize operational efficiency and minimize redundant data handling,” IRE J., vol. 5, no. 9, pp. 663–664, 2022.

5. M. Kherbouche, G. Pisoni, and B. Molnár, “Model to program and blockchain approaches for business processes and workflows in finance,” Appl. Syst. Innov., vol. 5, no. 1, p. 10, 2022.

6. K. Bell, “AI-Driven Workflow Automation: A Comparative Study of BPM Suites in Modern Enterprises,” 2024.

7. O. B. Alao, O. F. Dudu, E. O. Alonge, and C. E. Eze, “Automation in financial reporting: A conceptual framework for efficiency and accuracy in US corporations,” Glob. J. Adv. Res. Rev., vol. 2, no. 02, pp. 040–050, 2024.

8. T. Kitsantas, P. Georgoulas, and E. Chytis, “Integrating robotic process automation with artificial intelligence for business process automation: Analysis, applications, and limitations,” J. Syst. Manag. Sci., vol. 14, no. 7, pp. 217–242, 2024.

9. S. V. Subramanyam, “Transforming Lives with Accessible Business Process Systems in Healthcare,” Int. Res. J. Mod. Eng. Technol. Sci., vol. 7, no. 3, pp. 541–550, 2025.

10. D. Ajiga, P. A. Okeleke, S. O. Folorunsho, and C. Ezeigweneme, “The role of software automation in improving industrial operations and efficiency,” Int. J. Eng. Res. Updat., vol. 7, no. 1, pp. 22–35, 2024.

11. S. Shilpi, “Workflow-Aware Resilience In Enterprise Wire-Payment Systems: A Practical Framework For Minimizing Risk And Downtime In Production Support,” Available SSRN 5696863, 2025.

12. A. Park, S. Y. Jung, I. Yune, and H.-Y. Lee, “Applying robotic process automation to monitor business processes in hospital information systems: mixed method approach,” JMIR Med. Inform., vol. 13, p. e59801, 2025.

13. L. Husselman, “Anomaly Detection with Windows Event Logs: A comparative study between traditional and ML based approaches,” Master Thesis, 2024.

14. S. Tadi, “Process mining driven by deep learning for anomaly detection in intelligent automation systems,” J. Sci. Eng. Res., vol. 11, no. 1, pp. 317–329, 2024.

15. A. Vervaet, “Automated log-based anomaly detection within cloud computing infrastructures,” Sorbonne Université, 2023.

16. L. Zhen, N. H. Kamarudin, V. J. Kok, and F. Qamar, “Anomaly detection model in network security situational awareness based on machine learning: Limitation, techniques, future trends,” IEEE Access, 2025.

17. A. K. Dohare, “A Hybrid Machine Learning Framework for Financial Fraud Detection in Corporate Management Systems,” EKSPLORIUM-Bul. Pus. Teknol. BAHAN GALIAN Nukl., vol. 46, no. 02, pp. 139-154-139–154, 2025.

18. S. Kongkiatpaiboon et al., “Anomaly Detection on Compressor Application: Detailed Evaluation of Statistical and Machine Learning Techniques,” in SPE Asia Pacific Oil and Gas Conference and Exhibition, SPE, 2024, p. D021S012R004.

19. D. Paolini, P. Dini, A. Elhanashi, and S. Saponara, “Advanced Fault Detection and Diagnosis Exploiting Machine Learning and Artificial Intelligence for Engineering Applications,” Electronics, vol. 15, no. 2, p. 476, 2026.

20. A. Agarwal, “A unified machine learning framework for enterprise portfolio forecasting, risk detection, and automated reporting,” Future Technol., vol. 5, no. 1, pp. 337–354, 2026.

21. M. K. Alzaylaee, “Enhancing Cybersecurity Through Artificial Intelligence: A Novel Approach to Intrusion Detection.,” Int. J. Adv. Comput. Sci. Appl., vol. 16, no. 4, 2025.

22. P. Kumar, “AI-Powered Fraud Prevention in Digital Payment Ecosystems: Leveraging Machine Learning for Real-Time Anomaly Detection and Risk Mitigation,” J. Inf. Syst. Eng. Manag., vol. 9, no. 4, 2024.

23. Rascanu Dragos, “BPI Challenge 2017.” 2025.