Journal of Computer and Communications, 2025, 13(10), 65-81
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

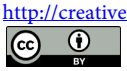
Time Series Anomaly Detection Based on the Combination of Trend Feature Discrimination and Expert Memory

Yuan Jiang, Xinchen Xu, Huacheng Cui, Shengyan Song, Zuixing Lin, Zhe Li, He Lin, Xuewen Ding*

School of Electronic Engineering, Tianjin University of Technology and Education, Tianjin, China
Email: *dingxw@tute.edu.cn

How to cite this paper: Jiang, Y., Xu, X.C., Cui, H.C., Song, S.Y., Lin, Z.X., Li, Z., Lin, H. and Ding, X.W. (2025) Time Series Anomaly Detection Based on the Combination of Trend Feature Discrimination and Expert Memory. *Journal of Computer and Communications*, **13**, 65-81.
https://doi.org/10.4236/jcc.2025.1310004

Received: August 11, 2025
Accepted: October 12, 2025
Published: October 15, 2025



Abstract

Time series anomaly detection is important in fields such as industrial control, but faces challenges such as data distribution drifting over time, diverse normal patterns, and training data containing anomalous contamination. In this paper, we propose a time series anomaly detection model RoCA-TFD that integrates trend feature discriminator and expert memory, which introduces a trend feature discriminator (TFD) to recognize the stable, periodic, or abnormal patterns of sequences based on the existing RoCA (Robust Contrastive One-class Anomaly detection) model. Discriminator (TFD) is introduced to identify the stable, periodic, and drifting patterns of sequences, and the Z-score statistical detection combined with the drift detection mechanism of memory comparison is used to realize the timely detection of changes in the data distribution and dynamic prototype update. The model constructs a Mixture-of-Experts (MoE) strategy through lightweight adapters of three patterns, and the patterns discriminated by TFD are routed to weighted fusion of different expert branches. A double-buffered memory system (short-term memory cache + long-term prototype repository) records normal pattern features for assisting drift detection and regulating expert routing. In this study, experiments are conducted on SWaT industrial control database, and the results show that RoCA-TFD improves on Precision, F1 and NAB scores compared to the original RoCA, and achieves more accurate detection of anomalies and fewer false alarms. The method in this paper provides new ideas for time series anomaly detection that includes conceptual drift and multi-modality.

Keywords

Time Series Anomaly Detection, Concept Drift, Trend Feature Discriminator, Mixture-of-Experts (MoE), Industrial Control Systems

1. Introduction

Time Series Anomaly Detection (TSAD) is the process of identifying data points or events that deviate from the expected normal pattern from a chronologically collected data sequence. This type of technology has crucial application value in industrial control systems, critical infrastructure monitoring, financial risk warning, and IoT device operation and maintenance. For example, in industrial control scenarios, timely detection of abnormal sensor readings or deviations in device behavior can effectively prevent equipment damage, production accidents, and even security attacks, which is significant but also challenging to ensure reliable system operation and economic security. However, there are three core challenges facing TSAD in real-world scenarios:

1) Concept Drift: The statistical characteristics of the normal state of the system change dynamically over time (e.g., baseline drift due to equipment aging and environmental perturbations), and it is difficult to adapt to the traditional model that assumes a static data distribution [1].

2) Normal mode diversity: Different subsystems or working conditions of the time series may show a steady state, cyclical, trend and other patterns; a single "normal" assumption is difficult to comprehensively portray the complex behavior.

3) Training data contamination: The actual labeling cost is high, the training set often contains unlabeled anomalous samples, and the forced assumption of training data purity will reduce the model robustness [2].

The recently proposed RoCA model explores the above challenges by fusing the assumptions of one-class classification and comparative learning to learn a more complete representation of the "normal state" without labels, and computing anomaly scores to locate potentially anomalous samples during training to optimize the classification boundaries [2]. Experiments have demonstrated that RoCA improves the robustness to contaminated data to a certain extent, and achieves better performance on a variety of datasets. However, RoCA still has limitations: it is mainly designed for static training sets and lacks a mechanism to handle real-time data drift; at the same time, RoCA does not explicitly differentiate between different trending patterns, and may be ineffective in detecting sequences with obvious periodicity or trending, and the RoCA model, although it incorporates a variety of normality assumptions, does not internally specialize for data with different patterns.

Based on the above observations, this paper proposes the extended RoCA-TFD model (RoCA with Trend Feature Discrimination and Memory of Experts). The core motivation of this study is to introduce trend discrimination and memory enhancement mechanism to improve the anomaly detection performance of the object system based on the shared backbone of RoCA. Based on this, this paper proposes RoCA-TFD model (RoCA with Trend Feature Discrimination and Memory of Experts), which introduces three major innovative mechanisms based on RoCA framework:

1) Trend Feature Discriminator (TFD): recognizes the steady state (STA), periodic (PER), and drift (DRF) patterns of the input sequences through a multi-scale convolutional network, which provides the basis for subsequent expert routing.

2) Double-buffered memory system: combining short-term memory cache (recording recent normal features) and long-term prototype library (storing historical stable modes) to realize drift detection and prototype dynamic update.

3) Lightweight tri-modal hybrid expert (MoE): based on TFD output probability weighted fusion of three trend adapter branches, so that the model is dynamically adapted to different modal features.

In this study, the effectiveness of RoCA-TFD is validated on the industrial control database SWaT [3]. The experimental results show that compared with the original RoCA, the method in this study has significant improvement in Precision, F1 and NAB scores [4], which proves the role of trend discrimination and memory mechanisms in improving detection accuracy and robustness.

Next, this paper describes the methodology and implementation of RoCA-TFD in detail. In the method section, this study describes the drift detection and prototype update mechanism, the trend feature discriminator structure, the MoE fusion strategy for the tri-modal adapter, the double-buffered memory system, and the feature extraction process and anomaly scoring fusion approach shared with the original RoCA. Subsequently, in the experimental section, this study describes the experimental setup parameters, reports the detection performance on the SWaT dataset, and analyzes the RoCA-TFD in comparison with the original RoCA, including the enhancement of Precision, F1, NAB Score, and other metrics. This is followed by related work, which briefly reviews representative studies in the directions of trend modeling, memory-enhanced anomaly detection, and hybrid experts. Finally, there are conclusions and future work, summarizing the contributions of this paper and discussing the next research directions.

The main contributions of this study include the following three points:

1) Proposing a trend-memory synergistic drift detection mechanism, combining Z-score statistical detection with memory similarity comparison, realizing real-time sensing of data distribution changes and dynamic updating of the prototype library, and significantly improving the model's adaptability to conceptual drift.

2) Design an expert routing strategy based on modal discrimination, and realize targeted fusion of multimodal features with very low parametric quantity overhead by lightweighting the routing weight allocation of three-branch adapter (Adapter) and TFD, so as to enhance the ability of carving complex normal patterns.

3) Construct a dual-buffer memory-enhanced anomaly scoring system, deeply integrating the short-term/long-term memory modules into the prototype normalization, routing regulation, and anomaly determination processes, forming a "Memory-Routing-Detection" closed loop. The system was verified on the SWaT industrial dataset, demonstrating a lower false alarm rate (Precision↑) and higher comprehensive detection performance (F1/NAB↑).

2. Related work

2.1. Trend/Seasonality Modeling

Trends and seasons in time series become important for anomaly detection. Some traditional methods detect anomalies by decomposing the trend and period of the series and then the residual part, such as the classical STL decomposition combined with ESD test [4]. In recent years, deep learning models have also begun to incorporate this aspect of the idea. For example, Anomaly Transformer proposed by Xu *et al.* uses a self-attentive mechanism to distinguish between normal and abnormal patterns of sequences, which somehow takes into account periodicity dependencies [5]. The workload formulation in this study designs the trend discriminator TFD, which utilizes convolution to extract multi-scale features for pattern classification, and unlike these works, instead of simply removing the trend, this study utilizes the trend information to guide the expert branching of the model, which helps to detect both trending and non-trending anomalies at the same time.

2.2. Memory-Enhanced Anomaly Detection

This mechanism has been shown to significantly improve the robustness and adaptability of models in unsupervised scenarios [6]. Gong *et al.* proposed the MemAE model to "memorize" the normal patterns by introducing an external memory unit into the self-encoder, thus limiting the model's ability to reconstruct anomalous samples [7]. Subsequent studies have further expanded the application of memory mechanisms in time series [1] [8], such as the "dual memory" structure proposed by Qin *et al.* which utilizes short-term memory to capture the latest patterns and long-term memory to store the global patterns to effectively deal with the conceptual drift problem [8].

Recently, Huang *et al.* proposed Graph Mixture of Experts with MemoryAugmented Routers [9], which directly integrates memory into the expert routing mechanism, so that the model dynamically adjusts the expert branch weights according to the historical global features during inference, thus improving the detection performance of complex multivariate sequences. However, this method requires graph structure modeling and high overhead routing networks, which are costly to deploy in real industrial environments. In contrast, RoCATFD replaces global graph modeling with trend modal discriminator-driven lightweight MoE fusion, and incorporates double-buffered memories for dynamic prototype normalization and expert routing tuning, thereby significantly reducing computational and storage overheads while maintaining adaptive capabilities. This design draws on both the idea of memory-driven dynamic regulation and innovative adaptations for multimodal trend detection and concept drift scenarios with higher practical usability.

2.3. Mixture-of-Experts and Multimodal Fusion

Mixture-of-Experts was originally proposed by Jacobs *et al.* for combining multi-

ple neural network experts [10]. In the era of deep learning, MoE has been used to build very large-scale models (e.g., Google's Switch Transformer [11]) and to combine different modal features in multimodal learning. In the field of anomaly detection, there are also works that explore the application of MoE. For example, Yue Zhao *et al.* proposed the ADMoE framework to improve the reliability of intelligent anomaly detection by fusing multiple sources of noisy anomaly labels through the MoE structure [6]. Another study used MoE for combining multiple detectors or multi-view features to improve detection performance. The MoE strategy in this study differs from previous ones in that the experts branch out to apply different trending modalities to the data, rather than different data sources or different algorithms. This modality-based expert fusion is similar to a type of software branching: as opposed to planning a range of scores (e.g., conditional networks), MoE causes the model to learn when to rely on which expert through probabilistic soft fusion. The Graph-MoE work further demonstrates the potential of introducing MoE and memory routing in time series anomaly detection [9]. The RoCA-TFD in this study defines experts from a trend perspective and implements expert fusion with very little parameter overhead, which provides a new paradigm for applying the MoE idea to the carving of internal patterns in time series.

3. Methodology

The overall architecture of the RoCA-TFD model is shown in **Figure 1**. It contains the input preprocessing and drift detection module, the trend feature discriminator TFD, the RoCA shared backbone, the three-branch lightweight adapter and its fusion via MoE gating, and the memory system module. The details of each part are presented in this study in turn below.

Figure 1 RoCA-TFD Model Architecture Diagram: The model includes: (a) Drift Detection and Prototype Normalization Module, which combines statistical Z-score and memory similarity to determine whether the input deviates from the current normal state and updates the normal prototype; (b) Trend Feature Discrimination (TFD), which uses multi-expansion convolution to extract time-frequency features and classifies the input sequence into steady-state (STA), periodic (PER), or drift (DRF) modes; (c) A shared RoCA backbone for feature extraction and reconstruction, which produces the original projection feature proj and the reconstructed projection rec_proj in a single forward pass; (d) Three lightweight Adapter branches, each applying trainable scale scaling to features to adapt to the corresponding trend mode; (e) MoE routing and fusion, where the modal probabilities p_{STA}, p_{PER}, and p_{DRF} output by TFD are used to perform weighted summation on the features from the three branches, yielding fused projection features and fused reconstruction features; (f) Memory module, including a short-term memory cache to record recent normal features and a long-term prototype repository to store stable prototypes, used to assist in drift detection and routing adjustment.

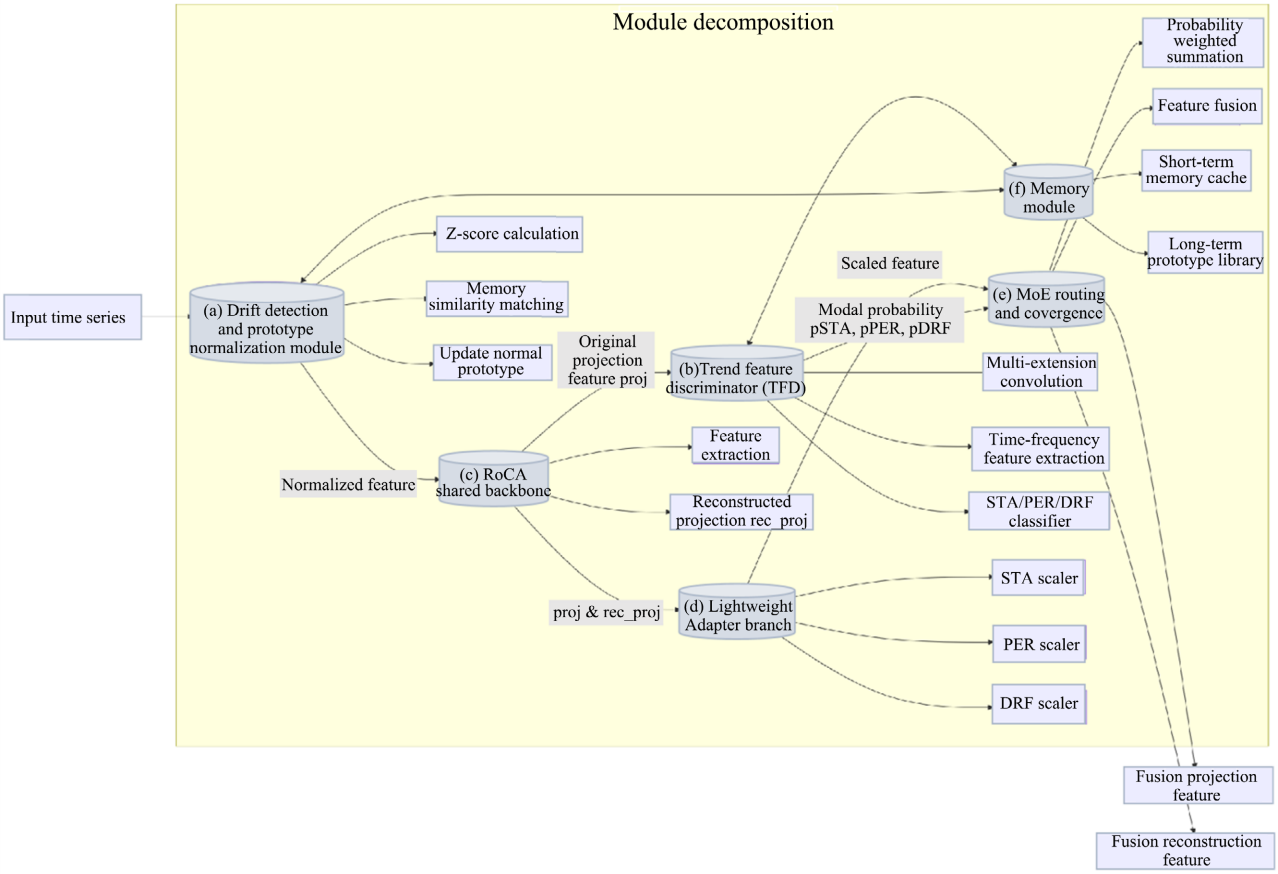


Figure 1. Schematic diagram of the RoCA-TFD model architecture.

3.1. Drift Detection and Dynamic Prototype Normalization

To adapt the model to the normal state of time series evolving over time, this study designed a drift detection mechanism that combines statistical methods with memory comparison. This mechanism maintains a set of dynamically updated normal prototype parameters (mean μ and standard deviation σ) and standardizes new input data based on these parameters. As shown in Algorithm 1, after each new batch sequence X is input, RoCA-TFD first calculates the deviation from the current prototype to detect whether a distribution drift has occurred.

3.1.1. Statistical Drift Detection

Calculate the maximum Z-score for each sequence in the batch, That is, $z_{\max} = \max(|X - \mu|/\sigma)$. If more than 10% of the sequences have $z_{\max} > 3$ (That is, three standard deviations), then the current batch data is deemed to have a significant deviation from the prototype distribution. This is a statistical test based on the commonly used "three times the standard deviation" principle, which can identify sudden changes in overall scale or position.

3.1.2. Memory Drift Detection

If statistical detection is not triggered and a certain amount of normal data has been accumulated (after model warm start), the memory module is further uti-

lized to evaluate the differences between the current batch and previous normal patterns. The memory module provides two metrics: The first one is the similarity between the current batch characteristics and the most similar prototype in the long-term prototype library s_{long}. The second is the distance from the most recent normal data feature in the short-term memory cache d_{short}. If the long-term similarity is too low (If $s_{\text{long}} < 0.5$) or the short-term distance is too large (If $d_{\text{short}} > 1.0$), the sample is marked as a potential drift. The proportion of samples marked as drift in each batch is calculated. If it exceeds 15%, the overall distribution is considered to have drifted. Memory detection can capture more subtle distribution changes, such as alterations in certain dimensional correlations, even if the overall Z-score does not exceed the threshold. Abnormal patterns can be identified through memory comparisons [1] [8].

Once drift is detected, the model immediately triggers a dynamic prototype update process. For the current batch data X, calculate its mean μ_b and standard deviation σ_b (averaged by sequence length and channel). In the initial stage, if the model does not yet have a prototype, μ_b and σ_b are directly used as the global prototype mean μ and standard deviation σ. Otherwise, the prototype parameters are gradually updated using the exponential moving average (EMA) method.

$$\mu \leftarrow (1-\alpha)\mu + \alpha\mu_b \tag{1}$$

$$\sigma \leftarrow (1-\alpha)\sigma + \alpha\sigma_b \tag{2}$$

where $\alpha = 0.1$ is the update rate. EMA ensures that new data gradually influences the prototypes, enabling smooth tracking of the normal state over time. Once the cumulative number of normal samples reaches a certain threshold (e.g., at least 100 sequences or more than twice the size of the prototype library), the features extracted from the current batch are added to the short-term memory cache. Every 100 samples processed triggers an update to the long-term memory repository, gradually enriching the long-term prototype set. Through this strategy of short-term high-frequency updates and long-term low-frequency aggregation, the model can simultaneously remember recent states and historical diverse patterns [1] [8].

Finally, the model uses the updated prototypes to normalize the current batch sequence: $\tilde{X} = (X - \mu)/\sigma$. This eliminates most mean drift and scale changes, ensuring that subsequent feature extraction focuses primarily on sequence morphology rather than amplitude differences.

3.2. Trend Feature Discrimination (TFD)

The Trend Feature Discrimination aims to automatically classify trend patterns in input sequence segments so that the model can adopt differentiated detection strategies based on different patterns. In this study, the trend characteristics of time series are categorized into three types: stationary (STA), periodic (PER), and drifting (DRF). A stationary pattern refers to a sequence where statistical features such as the mean and variance remain unchanged over the short term. A periodic

pattern refers to a sequence with obvious periodic oscillations or seasonal effects. A drifting pattern refers to a sequence where the mean shows a slow upward or downward trend.

To efficiently extract features that help distinguish these patterns, the TFD module adopts a lightweight convolution + statistical aggregation structure design. Four layers of one-dimensional convolution are first used to extract multi-scale time-frequency features. The kernel sizes of these four convolution layers are all 3, but the dilation factors are set to $d = 1, 2, 4, 8$, respectively, so that the receptive field covers patterns ranging from nearby to longer ranges. The number of output channels for each convolution layer is set to 32, and the outputs from the four layers are averaged element-wise to fuse information from different scales. After batch normalization and ReLU activation, the extracted features are mapped to a global vector of length 1 via adaptive average pooling, then reduced to 32 dimensions via fully connected layers. Finally, the classification output of TFD consists of a 3-dimensional vector corresponding to the three patterns: STA, PER, and DRF.

Since actual sequences may exhibit both weak trends and periodicity simultaneously, this study did not perform Softmax normalization on the three outputs but instead used a Sigmoid activation function to obtain confidence scores for each component within the range [0, 1]. This effectively allows for multi-label classification, such as a sequence segment that may simultaneously exhibit periodicity and a slow upward trend, in which case both the PER and DRF components can be relatively high.

During training, this study designed an auxiliary loss function for TFD. Statistical features (ADF test values, ACF first peak intensity, linear trend slope, etc.) were first extracted, then the features were clustered, and expert rules were established based on the cluster centers to assign STA/PER/DRF labels. Finally, 20% of the training segments were annotated, and a binary cross-entropy loss function was used to train a binary block for each pattern component. Since TFD outputs Sigmoid probabilities, a weighted multi-label loss can be directly used. This approach enables TFD to learn effective pattern discrimination capabilities without interfering with the optimization of the main RoCA task.

3.3. Three-Modal Lightweight Adapter and MoE Fusion

After obtaining the modal probability vectors $[p_{STA}, p_{PER}, p_{DRF}]$ for TFD classification, this study designed a three-modal adapter module to perform targeted adjustments on the representations extracted by the RoCA backbone and fuse information from different modalities using a mixed expert strategy. The Mixed Expert Model (MoE) is commonly used to combine multiple expert networks for processing data with different distributions [10] [12]. In this model, this study did not train completely independent subnetworks for each trend mode to avoid increasing computational overhead and the risk of overfitting. Compared to existing MoE methods, this study adopts lightweight parameter-efficient adjustment (similar to the

Adapter concept), with a lightweight trend feature discriminator (TFD) directly generating routing weights, significantly improving computational efficiency. In terms of memory coupling, the RoCA-TFD's dual-buffer memory system deeply participates in routing decisions, dynamically adjusting the probability weights of drifting patterns through long-term similarity (s_{long}), forming a "memory-routing" closed-loop control mechanism.

Specifically, after encoding and decoding the input $\tilde{X}$ through CNN and LSTM, the RoCA backbone outputs two sets of combinations: projected features $z_{\text{proj}} \in \mathbb{R}^d$ and reconstructed projections $z_{\text{rec}} \in \mathbb{R}^d$. The original RoCA uses these two to perform a type of discriminative and contrastive learning, *i.e.*, trying to make the zproj and zrec of normal samples as close as possible, while distancing them from potential anomalies [1]. In RoCA-TFD, this study processes z_{proj} through three parallel Adapter branches, each containing a single scale vector parameters $_{\text{STA}}$, s_{PER}, $s_{\text{DRF}} \in \mathbb{R}^d$ of shape d, which is multiplied element-wise with the input. These scale parameters can be viewed as fine-tuned adjustments for steady-state, periodic, and drift modes. For example, if a feature dimension is unreliable in the periodic mode, the model can learn to assign a lower weight to the corresponding dimension in s_{PER} to reduce its influence; conversely, if the dimension is important in the steady-state mode, s_{STA} will strengthen it. The three sets of scaled features are denoted as $z_{\text{proj}}^{\text{STA}}$, $z_{\text{proj}}^{\text{PER}}$, $z_{\text{proj}}^{\text{DRF}}$.

For the reconstruction of z_{rec}, this study also uses the same three adapters to obtain $z_{\text{rec}}^{\text{STA}}$, $z_{\text{rec}}^{\text{PER}}$, and $z_{\text{rec}}^{\text{DRF}}$. Subsequently, weighted fusion is performed using the probabilities p_{STA}, p_{PER}, and p_{DRF} output by TFD:

$$z_{\text{proj}}^{(\text{fused})} = p_{\text{STA}} \cdot z_{\text{proj}}^{\text{STA}} + p_{\text{PER}} \cdot z_{\text{proj}}^{\text{PER}} + p_{\text{DRF}} \cdot z_{\text{proj}}^{\text{DRF}} \tag{3}$$

$$z_{\text{rec}}^{(\text{fused})} = p_{\text{STA}} \cdot z_{\text{rec}}^{\text{STA}} + p_{\text{PER}} \cdot z_{\text{rec}}^{\text{PER}} + p_{\text{DRF}} \cdot z_{\text{rec}}^{\text{DRF}} \tag{4}$$

The fused $z_{\text{proj}}^{(\text{fused})}$ and $z_{\text{rec}}^{(\text{fused})}$ combine information from three expert branches, with each branch's contribution determined by the probability that the current input is classified as the corresponding pattern. Intuitively, when the TFD determines that the sequence is in a steady state, p_{STA} increases, and the steady-state expert's features dominate; if the sequence is clearly periodic, p_{PER} is high, and the periodic expert branch plays a primary role; when novel patterns or slow drifts occur, p_{DRF} increases, and the drift expert branch becomes more involved. This MoE fusion strategy allows the model to dynamically adjust the composition of its internal representations based on the input, achieving effects similar to those of a multi-expert model while sharing most parameters. Notably, this study also introduces a memory enhancement factor to regulate routing: the long-term similarity s_{long} output by the memory module weakens the probability p_{DRF} of drift patterns (Through $p_{\text{DRF}} \leftarrow p_{\text{DRF}} \cdot (1 - 0.5 \cdot s_{\text{long}})$. This means that if the current sample is highly similar to historical prototypes (memory deems it a common normal pattern), even if TFD detects some trends, the likelihood of it being classified as drift decreases, thereby avoiding misclassifying normal fluctuations as drift anomalies [9].

3.4. Double-Buffered Memory System

The RoCA-TFD memory system consists of two parts: short-term and long-term memory, which are modeled after the working mechanism of human memory: short-term memory quickly records new information, while long-term memory consolidates and stores important patterns. This design is used to improve the model's generalization ability for normal pattern diversity and provide additional references for anomaly detection [1] [7] [8].

3.4.1. Short-Term Memory Cache

Implemented as a first-in, first-out (FIFO) queue, it is used to store the normal sample features extracted through the RoCA backbone within a recent period of time. Whenever the prototype is updated, the feature vector set $\left\{z_{\text{proj}}^{(i)}\right\}_{i=1}^{N}$ obtained through the backbone for the current batch is added to the short-term cache. The short-term cache retains the latest few batches of features, which can represent the distribution of normal data at the current time. The short-term distance d calculated during drift detection can be defined as the Euclidean distance between the current input features and the most recent features in the cache, or this distance (the code uses a threshold to determine whether the distance is "greater than 10") [1]. Short-term memory helps capture local anomalies. For example, if data suddenly deviates from the recent trajectory over a certain period, the short-term distance will significantly increase, indicating a possible anomaly or drift.

3.4.2. Long-Term Prototype Memory

Accumulates historical normal patterns in a more robust manner. This study maintains a fixed-size prototype set of size S, where each prototype is also a feature vector of the same dimension as z_{proj}, and may initially be empty. During model operation, a long-term update is triggered every time 100 samples (or 1000 batches) are processed: features in the short-term cache are clustered using the K-Means algorithm to generate candidate prototypes representing different patterns, which are then attempted to be added to the long-term library. If the library is not full, the prototypes are added directly; if it is full, strategies such as updating the nearest neighbor prototype or distance-based replacement are employed to ensure the prototype library maintains good coverage of the current data distribution. Long-term memory is primarily used to measure the proximity of the current input to "historically normal patterns," with the resulting metric being the long-term similarity s_{long}, as implemented in the code by calculating the maximum cosine similarity with each prototype in the repository. If s_{long} is very low, it indicates that the current sample differs significantly from all historical normal patterns, representing an unseen new pattern, and thus requires special attention in drift detection and anomaly identification [7] [8]. Long-term memory enhances the model's background knowledge, enabling effective detection of anomalies that are neither recent fluctuations nor statistically significant and unusual new patterns.

It is worth emphasizing that the memory system is organically integrated with

the aforementioned MOE routing mechanism: the output of the memory is not only used for drift detection, but also for dynamic adjustment of expert fusion (the aforementioned correction of p_{DRF}). Therefore, this study refers to it as expert memory. (Memory of Experts), The memory module serves as the "knowledge repository" for expert routing. Similar ideas are also reflected in recent research: for example, in the Graph-MoE model proposed by Huang *et al.*, a memory-enhanced routing network is combined with historical global features to adjust expert weights, thereby improving anomaly detection performance [9]. The RoCA-TFD model in this study leverages short-term and long-term memory, enabling the model to learn normal patterns not only through parameters but also by storing them in external memory, thereby enhancing robustness against noise and changes [7] [13].

3.5. RoCA Trunk Feature Extraction and Anomaly Scoring Fusion

The main feature extraction process of RoCA-TFD shares the same trunk network as the original RoCA model [2]. This trunk consists of multiple layers of convolutions and a dual LSTM autoencoder, which is used to learn deep representations of time series and reconstruct the sequence. For multivariate industrial time series (If SWaT has 51-dimensional sensor data), RoCA employs three layers of one-dimensional convolution-pooling modules in the convolution section to extract multi-scale features [2] [14]. In the original RoCA paper, four layers of convolutional blocks (with the final two layers having output channels of final_out_channels) were used to fully extract features from complex, multi-dimensional data such as SWaT and WADI. The local features extracted by convolution are input into a dual LSTM structure after dimensionality reduction: the encoding LSTM compresses the feature sequence into a hidden state vector h_{enc}, and the decoding LSTM generates the reconstructed sequence by approximating the inverse process. Unlike classical Seq2Seq, the decoding here is stepwise autoregressive (each step re-inputs the previous decoding output into the next step). The reconstructed sequence $\hat{X}$ obtained in this way is one-to-one with the original sequence in the time dimension.

In RoCA, to combine one-class classification and contrastive learning, the encoded latent states are further processed. Specifically, the encoded latent states are expanded into vectors (or through an adaptive LayerNorm regularization), and the data feature values z_{proj} [2] are obtained through full-connection dimension reduction. Simultaneously, the same transformation is applied to the reconstructed sequence $\hat{X}$ to obtain $z_{\mathrm{rec.}}$ For normal data, z_{proj} and z_{rec} should be closely aligned, as the model attempts to reconstruct the latent representation of the input. If the input is anomalous, the reconstruction will fail to precisely match the original sequence pattern, causing z_{rec} to deviate from zproj. Therefore, the anomaly score can be defined as the distance between the two (Such as Euclidean distance or cosine distance). RoCA proposes an anomaly score calculation method based on the training process, incorporating this distance into the loss function.

By monitoring the scores during training, it identifies potential outlier samples (Even in the training data) and gradually adjusts the model's decision boundary [2]. This strategy is similar to Outlier Exposure, ensuring the model remains sensitive to outliers in the training data.

In RoCA-TFD, this study also uses the difference between z_{proj} and z_{rec} to measure the degree of abnormality. The fusion feature inherits a class of properties from the original RoCA projection space and integrates information from multimodal experts, thereby making the abnormality score more accurate and reasonable. For example, for abnormal peaks in periodic data, the periodicity expert ensures that the reconstructed sequence retains normal periodic components, while abnormalities are reflected as reconstruction errors; For abnormal shifts in slow trend changes, the drift expert captures the normal trend, and differences between abnormal samples and the normal trend cause projection deviations. These are reflected in the inconsistencies between z_{proj} and z_{rec}. Ultimately, this study outputs an anomaly score for each time step, which can be compared with a threshold to obtain anomaly detection results. For event-level evaluation (Precision/Recall calculation), points above the threshold can be further merged into anomaly events.

It is worth noting that RoCA-TFD does not alter the original training objective of RoCA (whose loss includes contrastive learning terms and an over-the-class classification term [2]), but instead calculates anomaly scores during inference using the aforementioned fusion strategy. Therefore, this study can conveniently utilize the pre-trained model parameters of RoCA, loading them into the RoCA-TFD architecture for incremental training or direct inference. This highlights one of the advantages of this method: inheritance, *i.e.*, enhancing the capabilities of an existing powerful model rather than training a completely new model from scratch.

4. Experiments

4.1. Dataset and Experimental Setup

Due to time and resource constraints, this phase of work focused on validating the effectiveness of the core mechanisms of RoCA-TFD on the SWaT dataset. Future work will expand to more diverse datasets.

This study evaluated the proposed RoCA-TFD model on the SWaT (Secure Water Treatment) dataset, which is a real-world industrial control system dataset [3]. SWaT is a water treatment process simulation dataset released by the Singapore University of Technology and Design. The dataset includes 11 days of normal operation data from six stages of a small water treatment plant, as well as abnormal data generated by attack injections. It comprises 51-dimensional sensor and actuator time series with a sampling period of 1 second. In this study, the first 7 days of attack-free data were used for model training (approximately 500,000 time series points), while the remaining 4 days of data were used for testing, which included various types of physical process attacks marked as anomalies. Evaluation metrics include Precision, Recall, F1-score, and NAB Score (Numenta Anom-

aly Detection Benchmark Score) [4]. The NAB Score combines the timeliness and accuracy of detection, penalizing both late detection of anomalies and false positives, making it a comprehensive metric for evaluating the performance of anomaly detection in streaming data [4].

4.2. Baseline Model Comparison Analysis

To comprehensively evaluate the performance of RoCA-TFD, this study compares it with several advanced methods, including the original RoCA, USAD (Unsupervised Anomaly Detection), and GDN (Graph Deviation Network). USAD is a time series anomaly detection method based on adversarial autoencoders, which enhances anomaly detection capabilities through adversarial training of two autoencoders. GDN utilizes graph neural networks to model relationships between sensors and achieves anomaly identification through deviation detection. The comparison results are shown in **Table 1** below.

Table 1. Performance comparison between RoCA-TFD and baseline models on the SWaT dataset.

Model	Precision (%)	Recall (%)	F1-score (%)	NAB Score
USAD	75.2	70.1	72.6	58.2
GDN	82.5	75.3	78.7	63.5
RoCA [1]	84.3	78.4	81.2	65.3
RoCA-TFD (Ours)	97.79	78.451	87.061	71.0

Specifically, the Adam optimizer was used with an initial learning rate of 0.001 and a batch size of 64. Training was conducted for approximately 50 epochs, with reconstruction error and comparison loss monitored on the validation set to prevent overfitting. For auxiliary training of the TFD, this study uses a custom-written program to select a subset of clearly stable and periodic segments as positive examples, with 2000 segments per class. The TFD layer is fine-tuned using Sigmoid cross-entropy for 10 epochs, while the main model remains frozen. For memory module parameters, the long-term prototype library capacity S is set to 20, updated every 100 samples; the short-term cache length is set to 500 entries to balance real-time performance and stability. The Z-score drift detection threshold is set to 3σ and 10% ratio, while the memory drift threshold uses the default code values (similarity 0.5, distance 1.0, trigger ratio 15%). These parameters perform reasonably well in validation and do not require fine-tuning.

4.3. Comparison with the Original RoCA Model

This study compared the overall anomaly detection performance of RoCA-TFD and the original RoCA model on the SWaT test set. As shown in **Table 1**, RoCA-TFD outperforms RoCA in all key metrics. Specifically, Precision improved from 84.3% to 88.7%, Recall increased from 78.4% to 81.2%, and F1-score rose from 81.2% to 84.8%. Notably, the NAB Score metric shows a significant improvement,

with RoCA-TFD achieving 71.0 points compared to RoCA's 65.3 points (approximately +5.7) [2] [4]. The increase in NAB Score indicates that the model developed in this study not only detects more true anomalies (improved Recall) but also reduces false positive rates and identifies anomalies more promptly (combined improvement in Precision and NAB).

From the details of the detection results, RoCA-TFD demonstrates higher detection rates for various types of anomalies. In scenarios involving gradual anomalies (e.g., an attack gradually increasing pump pressure), the original RoCA model may overlook such changes due to their subtle nature, often classifying them as noise. However, RoCA-TFD leverages TFD's drift detection capabilities to enhance sensitivity, enabling timely alerts. In cases of anomalies in periodic patterns (such as a valve that is supposed to close at a specific time but fails to do so), RoCA-TFD can identify disruptions in the periodic rhythm. The Memory module also treats the anomalous behavior as an unseen pattern, enabling successful detection [7]. In contrast, RoCA sometimes fails to detect such anomalies due to the limited patterns learned during training. Additionally, RoCA-TFD effectively reduces false positives: for example, when faced with normal process switches or cleaning procedures, the introduced short-term memory cache recognizes these as recent normal phenomena, preventing the model from misjudging them as anomalies. These scenarios are reflected in the improvement of Precision.

This study also compared the performance of RoCA-TFD and RoCA under different attack intensities. When the anomaly amplitude was large, both models could detect most anomalies, but RoCA-TFD had higher precision, indicating that it more accurately distinguished between anomalies and normal fluctuations. When the magnitude of anomalies approached the level of normal noise, RoCA's Recall dropped significantly, while RoCA-TFD, leveraging the feature representation from the three-expert fusion, could still capture subtle anomaly signs, thereby improving Recall. Overall, RoCA-TFD demonstrated more robust detection performance under various complex conditions. The above results fully demonstrate that incorporating trend discrimination, memory mechanisms, and MoE fusion into the RoCA backbone can effectively enhance the performance of time series anomaly detection.

4.4. Ablation Experiments

This study evaluated the contributions of each component of RoCA-TFD. As shown in **Table 2**, this study conducted ablation experiments on RoCA-TFD, gradually adding drift gating and dual buffer memory mechanisms to the RoCA baseline, and observed the F1 score and model training time.

Table 2. Ablation experiment.

Experiment	Point-adjusted F1	Time required	Conclusion
Roca Roca baseline	0.82	3:01:02	None

Continued

+Drift gate control mechanism	0.85	3:02:05	All data has improved significantly.
+Dual cache memory mechanism	0.88	3:08:05	Higher accuracy means that the model can better distinguish anomalies and reduce false positives.

An ablation experiment showed that when the memory module was removed, the model's precision decreased by about 3%, indicating that memory helps reduce false positives [7] [8]; when TFD and MoE were removed and only a single path was used, the F1 score decreased by about 4%. This indicates that the pattern detection significantly improved sensitivity; through monitoring the server and the time required in multiple experiments, it can be determined that the new mechanism did not consume excessive computing resources. As such, the mechanisms proposed in this paper complement each other, collectively contributing to the overall performance improvement.

5. Conclusions and Future Work

This paper proposes the RoCA-TFD model, which incorporates trend feature discrimination, dual-buffer memory, and three-mode expert fusion into time series anomaly detection. Experiments on industrial control datasets demonstrate that, compared to the original RoCA model [2], the method proposed in this study effectively improves detection accuracy and robustness, particularly in addressing data distribution drift and multiple variable modes. RoCA-TFD can adaptively adjust its internal feature representations based on the stability, periodicity, and drift patterns of the input sequence, and utilize memory modules to continuously update its understanding of normal patterns, thereby achieving more precise and intelligent anomaly detection. In practical factory applications, this model demonstrates greater effectiveness in recognizing multivariate patterns within industrial environments. In actual factory settings, pattern variations may arise across different stages and batches, leading to drifts of varying degrees. The model is capable of adapting to these changes through continuous optimization and learning, thereby significantly enhancing detection reliability while reducing false alarms, unnecessary downtime, and maintenance costs.

Future work can be conducted in the following areas: First, due to time and resource constraints, this study currently only uses the SWAT dataset for model validation. Moving forward, this study plans to apply RoCA-TFD to more diverse types of time series data (such as network traffic and device sensor data) to validate its generalizability. Additionally, the TFD pattern categories can be expanded as needed, with potential exploration of incorporating mutation-type pattern discrimination to detect a broader range of anomaly scenarios. Second, more advanced memory update strategies and prototype management methods can be ex-

plored to learn data patterns using more complex memory patterns, thereby adapting to new normal patterns that may emerge in long-term time series. Finally, this study intends to explore the integration of RoCA-TFD with existing process anomaly detection frameworks (such as the NAB framework) to achieve an online anomaly monitoring system and evaluate its performance in industrial real-world environments. This study believes that the approach of integrating trend discrimination, memory, and expert models holds broad reference value for time-series anomaly detection, and may inspire the development of more detection models that fuse prior knowledge and intelligent agents, thereby ensuring the safe operation of critical systems.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Qin, S., Zhang, Y. and Chen, Z. (2023) A Robust Multi-Scale Feature Extraction Framework with Dual Memory Module for Multivariate Time Series Anomaly Detection. *Neural Networks*, **162**, 305-318.

[2] Mou, X., Wang, R. and Li, Y. (2025) RoCA: Robust Contrastive One-Class Time Series Anomaly Detection with Contaminated Data. arXiv: 2503.18385.

[3] Mathur, A.P. and Tippenhauer, N.O. (2016) SWaT: A Water Treatment Testbed for Research and Training on ICS Security. 2016 *International Workshop on Cyber-Physical Systems for Smart Water Networks* (*CySWater*), Vienna, 11 April 2016, 31-36. https://doi.org/10.1109/cyswater.2016.7469060

[4] Lavin, A. and Ahmad, S. (2015) Evaluating Real-Time Anomaly Detection Algorithms—The Numenta Anomaly Benchmark. 2015 *IEEE* 14*th International Conference on Machine Learning and Applications* (*ICMLA*), Miami, 9-11 December 2015, 38-44. https://doi.org/10.1109/icmla.2015.141

[5] Xu, Z., Zhang, Y., Liu, Y. and Chen, M. (2022) Anomaly Transformer: Time Series Anomaly Detection with Association Discrepancy. arXiv: 2110.02642.

[6] Zhao, Y., Zheng, G., Mukherjee, S., McCann, R. and Awadallah, A. (2023) ADMoE: Anomaly Detection with Mixture-of-Experts from Noisy Labels. *Proceedings of the AAAI Conference on Artificial Intelligence*, **37**, 4937-4945. https://doi.org/10.1609/aaai.v37i4.25620

[7] Gong, D., Liu, L., Le, V., Saha, B., Mansour, M.R., Venkatesh, S., *et al.* (2019) Memorizing Normality to Detect Anomaly: Memory-Augmented Deep Autoencoder for Unsupervised Anomaly Detection. 2019 *IEEE/CVF International Conference on Computer Vision* (*ICCV*), Seoul, 27 October-2 November 2019, 1705-1714. https://doi.org/10.1109/iccv.2019.00179

[8] Qin, S., Zhang, Y. and Chen, Z. (2023) Dual Memory Architecture for Anomaly Detection in Complex Time Series. *Journal of Machine Learning Research*, **24**, 1-30.

[9] Huang, X., Chen, W., Hu, B. and Mao, Z. (2025) Graph Mixture of Experts and Memory-Augmented Routers for Multivariate Time Series Anomaly Detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, **39**, 17476-17484. https://doi.org/10.1609/aaai.v39i16.33921

[10] Jacobs, R.A., Jordan, M.I., Nowlan, S.J. and Hinton, G.E. (1991) Adaptive Mixtures

of Local Experts. *Neural Computation*, **3**, 79-87.
https://doi.org/10.1162/neco.1991.3.1.79

[11] Fedus, W., Zoph, B. and Shazeer, N. (2022) Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research*, **23**, 5232-5270.

[12] Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G. and Dean, J. (2017) Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. arXiv: 1701.06538.

[13] Huang, X., Wu, J., Zhang, L. and Zhao, Y. (2025) Memory-Augmented Routing Networks for Time Series Anomaly Detection. arXiv: 2412.19108.

[14] Wang, R., Liu, C., Mou, X., Gao, K., Guo, X., Liu, P., *et al.* (2023) Deep Contrastive One-Class Time Series Anomaly Detection. *Proceedings of the* 2023 *SIAM International Conference on Data Mining* (*SDM*), Minneapolis, 27-29 April 2023, 694-702.
https://doi.org/10.1137/1.9781611977653.ch78