FUSING MULTI-VIEW SCORES VIA COUPLING FLOWS FOR TIME SERIES ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Time series anomaly detection is crucial for ensuring system reliability across various applications ranging from industrial monitoring to financial fraud detection. However, two fundamental challenges remain to be addressed: (1) Model bias caused by the inherent diversity of anomaly patterns; (2) Detection inflexibility caused by the scarcity of anomaly labels. We propose MSFlow (Multi-view Score Fusion via Coupling Flows), which constructs a coupling flow-based ensemble capable of modeling complex joint distributions of multi-dimensional scores through invertible transformations. Leveraging this flexible fusion framework, we strategically select four detection perspectives (clustering and reconstruction in both temporal and frequency domains). The coupling flows learn inter-view dependencies while preserving each perspective's unique detection capabilities, achieving effective integration that simple aggregation fails to accomplish. When labels are available, an uncertainty-guided enhancement mechanism identifies high-disagreement regions in ensemble predictions and selectively refines them through a learned soft router, enabling seamless adaptation from unsupervised to semi-supervised operation. Extensive experiments on 10 univariate and 8 multivariate benchmark datasets demonstrate that MSFlow achieves state-of-the-art performance across diverse anomaly types and label availability scenarios.

1 INTRODUCTION

Time series anomaly detection has become a critical capability across diverse real-world applications, from industrial equipment monitoring and financial fraud detection to fault diagnosis and automotive maintenance systems. The ability to automatically identify anomalous patterns in time series data is essential for ensuring system reliability, operational safety, and service continuity. The increasing complexity and scale of modern time series data demand advanced detection methods that can both capture various anomaly types and adapt to evolving operational conditions. Despite significant advances in anomaly detection techniques, two fundamental challenges continue to limit the performance of existing methods: the inherent diversity of anomaly patterns and the inflexibility in utilizing available supervision signals.

The first challenge stems from the remarkable diversity of time series anomaly types. Temporal anomalies manifest as either point anomalies affecting individual observations (including global and contextual anomalies) or subsequence anomalies spanning continuous segments (seasonal, trend, and shapelet anomalies). Researchers have developed various detection approaches, from statistical methods and classical machine learning to deep learning architectures, demonstrating excellent detection capabilities. However, different anomaly types exhibit vastly different behavioral patterns, and single detection mechanisms—constrained by their theoretical foundations and design biases—struggle

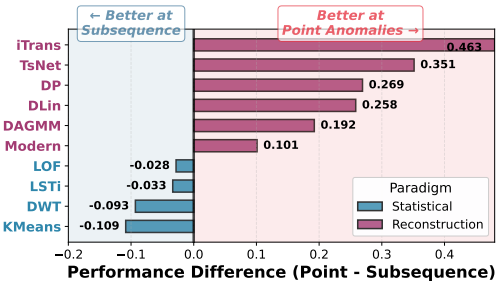


Figure 1: Method-specific detection biases. Positive values indicate better point anomaly detection; negative values indicate subsequence superiority.

to simultaneously capture these diverse characteristics (as shown in the figure B, detailed discussion in the appendix 1). This specialization inevitably limits the applicability of individual methods in complex real-world scenarios. The second challenge arises from the scarcity of anomaly labels in time series scenarios. This label scarcity has popularized unsupervised methods that operate without strict requirements on labeled data, while supervised and semi-supervised methods fail to handle limited annotation scenarios. This creates a fundamental gap: existing methods either completely discard available labels that could enhance detection performance, or require extensive labeling efforts that are impractical in most real-world scenarios. The absence of frameworks that can flexibly adapt to varying label availability prevents practitioners from leveraging valuable supervision signals that naturally accumulate in practical deployments.

We propose MSFlow (*M*ulti-view *S*core Fusion via Coupling *Flow*s), a framework that synergistically addresses the above challenges through a flexible multi-view ensemble architecture and adaptively utilizes available labels. For the first challenge, we develop an ensemble framework based on multiple coupling flows that can model the complex joint distribution of multi-view anomaly scores. Leveraging the flexibility of this ensemble framework, we strategically select four complementary detection perspectives spanning temporal/frequency domains and clustering/reconstruction paradigms, aiming to better cover diverse anomaly types. The coupling flow ensembles learn inter-view dependencies while preserving each view's unique detection capabilities. For the second challenge, we evaluate prediction disagreement among coupling flow ensembles as sample uncertainty—these uncertain samples represent the most valuable learning opportunities. Through uncertainty-guided selective supervision, we employ ranking loss to train a soft routing mechanism that ensures anomaly scores consistently exceed normal scores. This design enables MSFlow to effectively utilize valuable labels when available, while naturally degrading to unsupervised operation when labels are absent and identifying which samples most urgently require annotation.

We demonstrate MSFlow's effectiveness through extensive experiments on 18 diverse benchmarks (10 univariate and 8 multivariate datasets), showing consistent superiority over 20 state-of-the-art baseline methods. MSFlow achieves significant performance improvements when incorporating minimal labeled data and demonstrates effectiveness across multiple anomaly types, validating its practical applicability in various real-world scenarios.

- We propose MSFlow, a multi-view ensemble framework that models the joint distribution of diverse anomaly scores using coupling flows, effectively capturing different types of anomalies through complementary detection perspectives.
- We introduce uncertainty-guided selective supervision that leverages prediction disagreement to identify valuable annotation samples, enabling flexible adaptation from unsupervised to semi-supervised settings without requiring predefined label ratios.
- Extensive experiments on 18 benchmarks demonstrate MSFlow's consistent superiority over 20 state-of-the-art methods, particularly achieving significant performance improvements with minimal labeled data.

2 Related Work

2.1 Anomaly Detection Paradigms

Statistical Distribution-based Methods Statistical methods detect anomalies by finding data that differs from normal patterns (Chandola et al., 2009). Classic approaches include distance-based methods (Ahmed et al., 2020; Schubert et al., 2017; Li et al., 2003; Yi & Yoon, 2020; Ruff et al., 2018; Breunig et al., 2000; Guo et al., 2003; Yeh et al., 2016; Liu et al., 2008; Hariri et al., 2019) that identify outliers based on their proximity to normal data groups, density estimation (Goldstein & Dengel, 2012; Inoue & Shintani, 2006; Zong et al., 2018) that spots samples in low-density regions, and clustering-based methods (Yairi et al., 2001; He et al., 2003) that group normal patterns and identify deviations. Tests from the TAB benchmark (Qiu et al., 2025) show these traditional methods often beat deep learning at finding pattern anomalies. This is likely because they directly measure geometry and are sensitive to local structures. Recent work uses diffusion models (Zhang et al., 2025; Wyatt et al., 2022; Gathiaka et al., 2016) to learn complex patterns. These models spot anomalies by checking how well they can reconstruct data.

Reconstruction-based Methods Reconstruction methods work on a simple idea: models trained on normal data make larger errors when they see abnormal patterns (Sakurada & Yairi, 2014; Kingma & Welling, 2013). These methods have evolved with deep learning. They started with basic autoencoders (Zhou et al., 2021; Niu et al., 2020; Gong et al., 2019; Sakurada & Yairi, 2014). Then came recurrent networks (Niu et al., 2020; Su et al., 2019; Bhatnagar et al., 2021). Now we have attention-based models (Attention; Xu et al., 2021). Deep learning methods, especially Transformers (Xu et al., 2021; Yang et al., 2023; Wu et al., 2022; Nie et al., 2023; Liu et al., 2024a), have succeeded greatly due to their powerful representation advantages and are good at finding point anomalies. They learn complex features that catch single-point errors and local context problems (Qiu et al., 2025). Large pre-trained models (Chang et al., 2025; Goswami et al., 2024; Gao et al., 2024; Liu et al., 2024b) make this even better through transfer learning. However, these complex models can fail with simple outliers (Zeng et al., 2023). They also struggle with pattern anomalies because they focus too much on local quality and miss long-term shifts.

Discriminative Learning Methods Discriminative methods learn boundaries between normal and abnormal regions directly. They do not just rely on statistics or reconstruction quality. One-class classification (Ruff et al., 2018; Carmona et al., 2021; Schölkopf et al., 1999) builds tight boundaries around normal data in feature spaces. Contrastive learning (Yang et al., 2023; Yue et al., 2022; Zhuang et al., 2025) creates useful features by checking consistency across different views. It doesn't need anomaly labels. But these methods have problems. Without real anomalies for training, they must create fake ones that might not match real patterns. The learned boundaries might only work for specific anomaly types seen during training. New anomaly patterns are hard to handle.

2.2 Joint Temporal-Frequency Domain Analysis

Frequency analysis finds anomalies that time-based methods miss. These include broken cycles, warped harmonics, and spectral problems (Ren et al., 2019; Thill et al., 2017; Hyndman & Athanasopoulos, 2018). Different anomalies show up better in different domains. Seasonal anomalies appear clearly in frequency domain as broken patterns. Trend anomalies show better in time domain as long-term shifts (Qiu et al., 2025). Early work focused on single-variable spectral analysis (Feng et al., 2021; Ren et al., 2019). It used frequency maps and partial Fourier transforms to find anomalies. Modern methods (Wu et al., 2022; Yang et al., 2023; Zhang et al., 2022; 2019; Nam et al., 2024a; Wu et al., 2024) try to combine time and frequency analysis. They face several challenges: time and frequency details do not always match (Nam et al., 2024b), high-frequency information gets lost during processing, and cross-channel relationships are hard to model (Wu et al., 2024). Current fusion methods just combine features simply or merge them late. They fail to model the complex links between time and frequency scores.

3 Methodology

Consider a time series $\mathcal{X} = \langle x_1, x_2, \ldots, x_T \rangle$, where each point $x_t \in \mathbb{R}^d$ represents a d-dimensional measurement at time t. The training data $\mathcal{X}$ primarily contains normal behavior patterns (Chandola et al., 2009; Lai et al., 2021). Our goal is to build a detector that identifies anomalies in test sequences $\mathcal{X}_{\text{test}}$, producing binary labels $\mathcal{Y}_{\text{test}} = \langle y_1, y_2, \ldots, y_{T'} \rangle$ where $y_t \in \{0, 1\}$ indicates normal (0) or abnormal (1) behavior.

3.1 Framework Overview

MSFlow employs a staged approach to time series anomaly detection, centered on probabilistic modeling of multi-view anomaly scores through coupling flows. The framework consists of three main components: (1) an unsupervised foundation that models the joint distribution of anomaly scores from multiple detection perspectives using an ensemble of coupling flows; (2) a supervised enhancement mechanism that leverages prediction disagreement to identify and improve uncertain regions when labeled data is available; (3) an anomaly scoring system that adaptively combines unsupervised and supervised signals based on detection confidence. We provide detailed algorithmic descriptions of the three core components in Algorithms **??–??** in the Appendix for readers interested in implementation details.

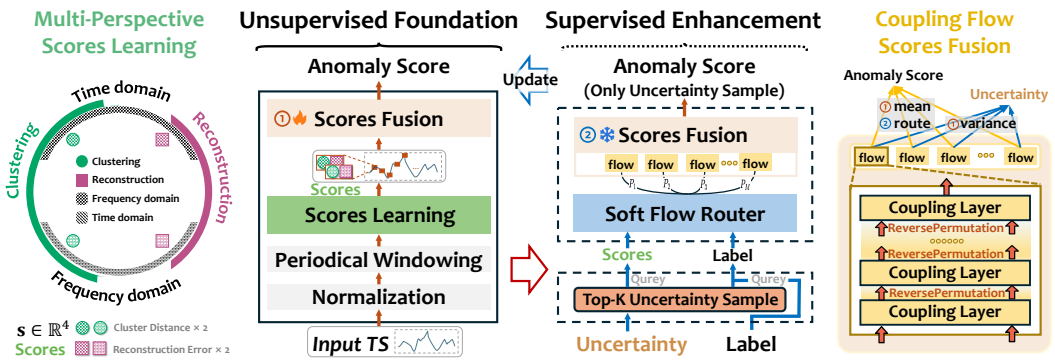


Figure 2: Architecture of MSFlow: A Multi-View Time Series Anomaly Detection Framework. The framework processes time series through adaptive windowing, parallel multi-view scoring across temporal and frequency domains, coupling flow-based probabilistic fusion, and optional uncertainty-guided enhancement.

Figure 2 illustrates the overall architecture. The framework first extracts anomaly scores from four complementary perspectives—clustering and reconstruction methods in both temporal and frequency domains—to capture diverse anomaly characteristics. These multi-view scores are then processed by an ensemble of coupling flows that learn their joint distribution through invertible transformations, producing fused anomaly scores. The key advantage of using multiple flows is that their prediction disagreement (variance) naturally quantifies detection uncertainty. When labeled data becomes available, this uncertainty guides selective supervision: during training, it identifies the Top-K most uncertain samples for training the soft router with ranking loss. During inference, the learned router is applied to all test samples when available, dynamically adjusting fusion weights based on the input score patterns. This design ensures the framework operates effectively in purely unsupervised settings while seamlessly incorporating supervision when available to enhance detection performance.

3.2 Unsupervised Foundation

Multi-View Score Learning The framework begins with standardization to ensure comparability across dimensions, followed by adaptive windowing that segments data into fixed-length windows $\mathbf{W}_i \in \mathbb{R}^{W \times d}$. The window size W can be automatically determined through FFT-based periodicity detection to align with natural data cycles, or set manually based on domain knowledge.

For each window, we compute anomaly scores from four complementary perspectives (see Appendix **??** for detailed rationale behind this design choice):

- **Temporal domain clustering:** We apply K-means clustering to identify K typical temporal patterns in the training data (Yairi et al., 2001; He et al., 2003). For each test window, we compute its Euclidean distance to the nearest cluster center. Windows that fall far from all learned clusters likely represent anomalous patterns not seen during training.
- **Temporal domain reconstruction:** We employ a Transformer model with inverted tokenization (Liu et al., 2024a; Nie et al., 2023), where each variable's time series serves as a token rather than each time point. The model learns to reconstruct normal temporal patterns through self-attention mechanisms (Xu et al., 2021). High reconstruction errors indicate deviations from learned normal dependencies.
- **Frequency domain clustering:** After applying FFT to convert windows to frequency domain, we perform K-means clustering on the magnitude spectra. This captures typical frequency patterns and identifies anomalies with unusual spectral characteristics that temporal analysis might miss (Ren et al., 2019).
- **Frequency domain reconstruction:** A specialized Transformer operates directly on frequency representations (both magnitude and phase), learning to reconstruct normal spectral patterns. This method excels at detecting subtle frequency anomalies like broken period-

icities, harmonic distortions, or abnormal frequency components (Thill et al., 2017; Nam et al., 2024a; Wu et al., 2024).

This multi-view approach leverages the observation that different anomaly types manifest distinctively across domains and detection paradigms. Point anomalies typically show high reconstruction errors, while pattern anomalies are better captured by clustering distances (Liu et al., 2008). Since we use overlapping windows with stride s, each timestamp may appear in multiple windows. We aggregate these window-level scores to point-level through averaging: $s_t^{(v)} = \frac{1}{|\mathcal{W}_t|} \sum_{w \in \mathcal{W}_t} s_w^{(v)}$, where $\mathcal{W}_t$ denotes all windows containing timestamp t (Yeh et al., 2016; Tatbul et al., 2018). The resulting score matrix $\mathbf{S}_t \in \mathbb{R}^{d \times 4}$ for each timestamp provides a comprehensive characterization of potential anomalies from multiple perspectives.

Coupling Flow-Based Score Fusion Traditional score aggregation methods (such as averaging or weighted combination) fail to capture the complex dependencies between different detection methods (Pevnỳ, 2016; Zong et al., 2018). Since the entire process operates without anomaly labels, we address this limitation through coupling flows, which perform self-supervised modeling of the complete joint distribution of multi-dimensional scores via invertible transformations (see Appendix **??** for motivation).

Before fusion, all scores are standardized within each variable to zero mean and unit variance, ensuring comparability across different detection methods (Bhatnagar et al., 2021). Each coupling flow $f_e : \mathbb{R}^{d \times 4} \to \mathbb{R}^{d \times 4}$ consists of L_c coupling layers that progressively transform the complex score distribution to a standard Gaussian distribution in latent space. We naturally organize the score matrix into temporal $\mathbf{T} \in \mathbb{R}^{d \times 2}$ (clustering and reconstruction scores) and frequency $\mathbf{F} \in \mathbb{R}^{d \times 2}$ (clustering and reconstruction scores) partitions. Each coupling layer applies conditional affine transformations through an alternating scheme:

$$\mathbf{T}^{(\ell+1)} = \mathbf{T}^{(\ell)}, \quad \mathbf{F}^{(\ell+1)} = \mathbf{F}^{(\ell)} \odot \exp(\mathbf{s}_\theta(\mathbf{T}^{(\ell)})) + \mathbf{t}_\theta(\mathbf{T}^{(\ell)}) \tag{1}$$

$$\mathbf{F}^{(\ell+1)} = \mathbf{F}^{(\ell)}, \quad \mathbf{T}^{(\ell+1)} = \mathbf{T}^{(\ell)} \odot \exp(\mathbf{s}_\phi(\mathbf{F}^{(\ell)})) + \mathbf{t}_\phi(\mathbf{F}^{(\ell)}) \tag{2}$$

where $\mathbf{s}$ and $\mathbf{t}$ are scale and translation parameters generated by coupling networks θ and ϕ. This alternating structure enables bidirectional modeling of dependencies between temporal and frequency domains, allowing the model to learn complex relationships such as correlations between specific frequency anomalies and temporal patterns.

Ensemble Strategy We train an ensemble of E coupling flows with different random initializations and data subsets to create diversity. Each flow independently optimizes the log-likelihood of observed score patterns:

$$\mathcal{L}_{\text{flow}}^{(e)} = \frac{1}{B} \sum_{i=1}^{B} \left[\frac{1}{2} \|\mathbf{f}_e(\mathbf{S}_i)\|_F^2 - \log |\det J_{f_e}(\mathbf{S}_i)| + C \right] \tag{3}$$

where J_{f_e} denotes the Jacobian determinant ensuring invertibility, and C is a normalization constant.

The ensemble serves three critical purposes: (1) *Robustness*—model averaging reduces individual flow biases and improves generalization; (2) *Uncertainty quantification*—prediction disagreement naturally identifies samples where the unsupervised detection is least confident, providing valuable guidance for selective supervision; (3) *Selective refinement*—enabling targeted improvement of uncertain regions without affecting confident predictions. The uncertainty for each sample is quantified as:

$$u_t = \text{Var}_{e \in \{1,\ldots,E\}}[r_e(\mathbf{S}_t)] \tag{4}$$

where $r_e(\mathbf{S}_t) \in [0, 1]$ is the percentile rank of the sample's likelihood relative to the training distribution, providing a normalized uncertainty measure robust to likelihood scale variations.

3.3 SUPERVISED ENHANCEMENT

When labeled data becomes available, MSFlow refines its detection capabilities through uncertainty-guided enhancement without requiring complete model retraining (Schmidl et al., 2022; Jacob et al., 2020). The key insight is that regions of high ensemble disagreement represent the most valuable improvement opportunities where supervision can have maximum impact.

Uncertainty-Guided Sample Selection We leverage the ensemble's prediction variance u_t to identify samples where supervision would be most beneficial. By selecting the Top-K samples with highest uncertainty (top $p_{\text{train}}\%$), we ensure that limited labels are strategically used to address the model's blind spots rather than redundantly confirming already-confident predictions. This active learning strategy maximizes the information gained from scarce labeled data.

Soft Flow Router For the selected high-uncertainty training samples, we introduce a lightweight routing network $g_\psi : \mathbb{R}^{d\times 4} \to \mathbb{R}^E$ that takes the original 4-dimensional anomaly scores $\mathbf{S}_t$ as input—specifically the clustering and reconstruction scores from both temporal and frequency domains—and learns to dynamically assign weights to each ensemble member. The router is trained exclusively on high-uncertainty samples using a ranking loss that encourages correct relative ordering between anomalous and normal samples:

$$\mathcal{L}_{\text{rank}} = \frac{1}{|\mathcal{P}|} \sum_{(i,j)\in\mathcal{P}} \log\left(1 + \exp\left(\phi(\mathbf{S}_j) - \phi(\mathbf{S}_i)\right)\right) \tag{5}$$

where $\mathcal{P}$ contains all valid pairs with anomaly i ($y_i = 1$) and normal sample j ($y_j = 0$) from the high-uncertainty subset. The routed score $\phi(\mathbf{S}) = \sum_{e=1}^{E} w_e(\mathbf{S}) \cdot r_e(\mathbf{S})$ combines individual flow predictions using learned weights w_e computed via temperature-controlled softmax. Once trained, the router is applied to all test samples, allowing it to adaptively select the most reliable flows for different score patterns across the entire test set.

3.4 ANOMALY SCORING

The framework converts coupling flow likelihoods to interpretable anomaly scores through percentile ranking. During training, we store the likelihood distribution of normal samples as a reference. At test time, each flow computes the negative log-likelihood for a test sample and converts it to a percentile rank $r_e(\mathbf{S}_t)$ by determining what percentage of training samples have lower likelihoods. This normalization ensures scores from different flows are comparable and provides an intuitive interpretation: a rank of 0.95 means the sample is more anomalous than 95% of training data.

The final anomaly score depends on whether supervised enhancement is available:

$$\text{AnomalyScore}(\mathbf{S}_t) = \begin{cases} \phi(\mathbf{S}_t) & \text{if supervised enhancement is trained} \\ \frac{1}{E}\sum_{e=1}^{E} r_e(\mathbf{S}_t) & \text{otherwise (pure unsupervised)} \end{cases} \tag{6}$$

where $\phi(\mathbf{S}_t) = \sum_{e=1}^{E} w_e(\mathbf{S}_t) \cdot r_e(\mathbf{S}_t)$ represents the router-enhanced scores using learned weights. When labeled data is available for training the soft router, the enhanced scoring is applied to all test samples, leveraging the learned weighting mechanism to improve detection performance. In the absence of labeled data, the framework defaults to simple ensemble averaging, maintaining strong unsupervised capabilities.

4 EXPERIMENT

4.1 EXPERIMENTAL SETTINGS

Datasets. We conduct comprehensive evaluation on 18 diverse time series anomaly detection benchmarks covering both 10 univariate datasets (GAIA, GHL, KDD21, MGAB, NASA-MSL, NAB, OPPORTUNITY, NASA-SMAP, SVDB, YAHOO) and 8 multivariate datasets (LTDB, MITDB, MSL,

SMD, NYC, PSM, SMAP, SVDB), more details of the benchmark datasets are included in Appendix Table 4 in Appendix A.1.

Baselines. We compare against 18 state-of-the-art methods spanning three major anomaly detection paradigms: statistical distribution-based approaches (6 methods), reconstruction-based methods (10 methods), and discriminative learning approaches (2 methods), as detailed in Table 1. The baselines encompass non-learning statistical methods, traditional machine learning algorithms, and modern deep learning architectures.

Table 1: Baseline methods for comparison

Paradigm	Method	Abbrev.	Key Technology
Statistical Distribution-based	• LOF (Breunig et al., 2000)	LOF	Density
	• KNN (Ramaswamy et al., 2000)	KNN	K-nearest Neighbors
	• Isolation Forest (Liu et al., 2008)	IF	Isolation Tree
	• HBOS (Goldstein & Dengel, 2012)	HBOS	Histogram
	• OC-SVM (Schölkopf et al., 1999)	OCSVM	One-class Classification
	• K-Means (Yairi et al., 2001)	KMeans	K-Means Clustering
Reconstruction-based	• VAE (Kingma & Welling, 2013)	VAE	Variational Autoencoder
	• DAGMM (Zong et al., 2018)	DAGMM	Deep Autoencoder + GMM
	• DeepPoint (Bhatnagar et al., 2021)	DP	Multilayer Perceptron
	• TranAD (Tuli et al., 2022)	TranAD	Transformer + Adversarial
	• Anomaly Transformer (Xu et al., 2021)	ATrans	Anomaly Attention
	• DualTF (Nam et al., 2024a)	DualTF	Time-Frequency Fusion
	• PatchTST (Nie et al., 2023)	Patch	Channel Independent + Patch
	• ModernTCN (Luo & Wang, 2024)	Modern	Enhanced TCN
	• DLinear (Zeng et al., 2023)	DLin	Linear + Decomposition
	• iTransformer (Liu et al., 2024a)	iTrans	Inverted Transformer
Discriminative Learning	• DCdetector (Yang et al., 2023)	DC	Contrastive Learning
	• ContraAD (Zhuang et al., 2025)	ConAD	Contrastive Learning

• Non Learning, • Machine Learning, • Deep Learning

Setup. We utilize TAB (Qiu et al., 2025) code for unified evaluation, with all baseline results also derived from TAB. To ensure fair comparison and demonstrate the robustness of our approach, we employ a single set of unified hyperparameters for all univariate datasets and another single set for all multivariate datasets, without dataset-specific tuning. We adopt Label-based metric: *Affiliated F1-score* (Huet et al., 2022) and Score-based metric: Area under the Receiver Operating Characteristics Curve (ROC) *AUROC* (Fawcett, 2006) as evaluation metrics. More implementation details are presented in the Appendix A.3.

4.2 Main Results

Tablese 2 present the evaluation results on univariate and multivariate datasets, respectively. MSFlow achieves the best performance on the majority of benchmarks, demonstrating strong generalization across diverse anomaly detection scenarios. For univariate time series, MSFlow obtains particularly impressive results on challenging datasets with complex temporal patterns, while maintaining consistent superiority in ranking quality. On multivariate benchmarks, our method delivers robust performance by effectively capturing cross-dimensional dependencies through the coupling flow mechanism. Compared to recent deep learning methods like ModernTCN, DualTF, and Anomaly Transformer, MSFlow demonstrates more stable performance across different data characteristics—while these baseline methods often excel on specific datasets but struggle on others, our approach maintains competitive results throughout.

4.3 Model Analysis

We conduct comprehensive experiments to analyze MSFlow from multiple perspectives. This section presents ablation studies to validate each component's contribution and visualizations of anomaly detection capabilities across different anomaly types. Additional analyses are provided in the appendix, including detailed performance breakdown for five fundamental anomaly types (Appendix B) and extensive parameter sensitivity analysis (Appendix **??**).

Table 2: Average A-R (AUC-ROC) and Aff-F (Affiliated-F1) accuracy measures for all datasets. The best results are highlighted in **Red**, and the second-best results are Blue.

Dataset	Metric	MSFlow	Modern	DualTF	iTrans	ATrans	DC	TranAD	ConAD	Patch	DLin	VAE	DAGMM	KNN	KMeans	OCSVM	IF	DP	LOF	HBOS
Univariate Datasets																				
GAIA	Aff-F	0.871	0.904	0.753	**0.907**	0.697	0.711	0.790	0.683	0.902	0.897	0.727	0.598	0.759	0.781	0.725	0.731	0.673	0.737	0.689
	A-R	**0.872**	0.821	0.564	0.823	0.446	0.411	0.756	0.486	0.819	0.801	0.748	0.648	0.821	0.738	0.726	0.695	0.755	0.817	0.687
GHL	Aff-F	**0.975**	0.723	0.682	0.709	0.652	0.669	0.577	0.667	0.735	0.730	0.555	0.465	0.679	0.598	0.667	0.687	0.667	0.679	0.638
	A-R	**0.998**	0.493	0.374	0.498	0.476	0.498	0.020	0.500	0.500	0.528	0.087	0.020	0.751	0.747	0.751	0.500	0.888	0.751	0.662
KDD21	Aff-F	**0.895**	0.718	0.698	0.767	0.698	0.686	0.723	0.685	0.713	0.726	0.652	0.640	0.678	0.822	0.669	0.675	0.667	0.675	0.666
	A-R	**0.887**	0.534	0.569	0.605	0.499	0.493	0.528	0.502	0.529	0.538	0.540	0.538	0.638	0.778	0.590	0.554	0.532	0.584	0.564
MGAB	Aff-F	**0.793**	0.678	0.675	0.691	0.671	0.673	0.670	0.667	0.669	0.668	0.664	0.620	0.666	0.688	0.666	0.664	0.667	0.667	0.665
	A-R	**0.773**	0.561	0.576	0.501	0.500	0.504	0.592	0.522	0.559	0.567	0.555	0.545	0.560	0.605	0.550	0.487	0.515	0.509	0.586
MSL	Aff-F	**0.922**	0.896	0.766	0.865	0.743	0.757	0.902	0.673	0.883	0.903	0.686	0.609	0.775	0.894	0.795	0.658	0.673	0.772	0.733
	A-R	**0.803**	0.726	0.537	0.724	0.426	0.469	0.616	0.500	0.727	0.710	0.652	0.609	0.673	0.725	0.666	0.561	0.636	0.614	0.607
NAB	Aff-F	0.828	0.838	0.782	0.832	0.751	0.714	**0.858**	0.701	0.848	0.850	0.739	0.620	0.730	0.818	0.722	0.711	0.680	0.695	0.744
	A-R	**0.640**	0.612	0.550	0.614	0.469	0.506	0.546	0.506	0.605	0.608	0.556	0.572	0.563	0.626	0.584	0.535	0.531	0.545	0.570
OPP	Aff-F	**0.788**	0.733	0.710	0.709	0.728	0.679	0.762	0.770	0.732	0.737	0.674	0.628	0.656	0.734	0.654	0.654	0.667	0.671	0.653
	A-R	0.664	0.215	0.411	0.224	**0.665**	0.503	0.559	0.504	0.217	0.304	0.514	0.578	0.461	0.285	0.487	0.242	0.260	0.545	0.522
SMAP	Aff-F	**0.967**	0.913	0.686	0.928	0.764	0.760	0.851	0.680	0.906	0.906	0.595	0.569	0.717	0.886	0.698	0.655	0.674	0.711	0.647
	A-R	**0.899**	0.694	0.535	0.752	0.480	0.487	0.520	0.502	0.707	0.615	0.596	0.586	0.594	0.802	0.586	0.513	0.655	0.551	0.526
SVDB	Aff-F	**0.833**	0.733	0.745	0.718	0.686	0.685	0.730	0.682	0.727	0.728	0.683	0.611	0.710	0.735	0.698	0.672	0.682	0.709	0.700
	A-R	**0.780**	0.573	0.586	0.576	0.504	0.460	0.549	0.500	0.569	0.580	0.570	0.549	0.520	0.683	0.567	0.544	0.531	0.516	0.553
YAHOO	Aff-F	**0.914**	0.793	0.748	0.897	0.671	0.629	0.755	0.678	0.800	0.825	0.615	0.453	0.661	0.604	0.675	0.897	0.669	0.664	0.653
	A-R	**0.936**	0.804	0.467	0.891	0.477	0.490	0.697	0.488	0.828	0.843	0.756	0.707	0.734	0.589	0.639	0.924	0.862	0.754	0.678
Multivariate Datasets																				
LTDB	Aff-F	**0.794**	0.777	0.455	0.756	0.702	0.710	0.790	0.752	0.767	0.764	0.670	0.653	0.768	0.724	0.754	0.644	0.724	0.763	0.759
	A-R	**0.833**	0.619	0.589	0.579	0.505	0.494	0.590	0.576	0.587	0.599	0.610	0.545	0.632	0.738	0.615	0.559	0.551	0.621	0.611
MITDB	Aff-F	**0.925**	0.803	0.843	0.806	0.700	0.689	0.848	0.694	0.813	0.722	0.713	0.689	0.709	0.834	0.700	0.709	0.678	0.700	0.687
	A-R	**0.767**	0.663	0.693	0.679	0.424	0.503	0.691	0.488	0.676	0.583	0.667	0.682	0.692	0.692	0.689	0.618	0.598	0.627	0.681
MSL	Aff-F	**0.738**	0.726	0.258	0.705	0.685	0.674	0.724	0.671	0.714	0.723	0.642	0.723	0.696	0.580	0.641	0.584	0.677	0.701	0.680
	A-R	**0.645**	0.621	0.499	0.589	0.502	0.500	0.478	0.548	0.621	0.601	0.530	0.569	0.623	0.603	0.524	0.524	0.488	0.557	0.574
SMD	Aff-F	0.794	**0.839**	0.679	0.817	c	0.674	0.789	0.023	0.830	0.834	0.450	0.023	0.696	0.686	0.742	0.626	0.674	0.682	0.629
	A-R	**0.749**	0.721	0.703	0.745	0.509	0.500	0.663	0.491	0.736	0.708	0.641	0.527	0.716	0.713	0.602	0.664	0.672	0.645	0.626
NYC	Aff-F	**0.946**	0.693	0.676	0.683	0.732	0.674	0.796	0.714	0.805	0.725	0.767	0.694	0.644	0.646	0.667	0.648	0.668	0.652	0.675
	A-R	0.808	0.696	0.723	0.594	**0.859**	0.526	0.676	0.443	0.667	0.699	0.661	0.573	0.466	0.833	0.456	0.475	0.476	0.464	0.446
PSM	Aff-F	0.739	0.823	0.725	**0.855**	0.634	0.671	0.748	0.648	0.836	0.832	0.295	0.463	0.695	0.735	0.531	0.620	0.694	0.694	0.658
	A-R	0.635	0.587	0.544	0.583	0.499	0.499	0.635	0.533	0.583	0.559	0.642	0.637	**0.744**	0.732	0.619	0.542	0.539	0.730	0.714
SMAP	Aff-F	0.551	0.616	0.674	0.577	**0.692**	0.681	0.538	0.430	0.629	0.607	0.487	0.430	0.630	0.517	0.503	0.512	0.681	0.642	0.509
	A-R	0.468	0.434	0.465	0.409	0.501	0.500	0.369	0.364	0.441	0.391	0.412	0.573	**0.629**	0.405	0.393	0.487	0.429	0.626	0.585
SVDB	Aff-F	**0.896**	0.746	0.585	0.746	0.701	0.711	0.752	0.694	0.787	0.727	0.697	0.692	0.733	0.820	0.723	0.708	0.692	0.734	0.735
	A-R	**0.913**	0.593	0.555	0.636	0.491	0.464	0.577	0.533	0.651	0.610	0.570	0.564	0.595	0.839	0.585	0.533	0.527	0.580	0.568

Impact of Supervised Enhancement. To evaluate the effectiveness of uncertainty-guided label utilization, we analyze MSFlow's performance with varying amounts of labeled data. Table 3 demonstrates the impact of different p_{train} percentages on detection performance.

Table 3: Performance (AUC-ROC) with varying amounts of labeled data for supervised enhancement. The baseline (0%) represents pure unsupervised MSFlow.

Dataset	Train AR (%)	MSFlow (Unsupervised)	p_{test} (%) 1	5	20	50	100
NAB	6.55	0.6402	0.6807	0.7003	0.7106	0.7004	0.7002
OPPORTUNITY	19.11	0.6643	0.6641	0.6645	0.6642	0.6644	0.6643
SVDB	5.48	0.7801	0.8103	0.8206	0.8204	0.8205	0.8203
YAHOO	5.48	0.9362	0.9413	0.9452	0.9447	0.9451	0.9453
LTDB	11.72	0.8334	0.8471	0.8532	0.8584	0.8603	0.8572
SVDB*	5.48	0.9131	0.9142	0.9154	0.9146	0.9152	0.9151

The results reveal a characteristic learning curve with diminishing returns: incorporating a small fraction of labeled data yields substantial performance improvements, while further increasing the label percentage provides progressively smaller gains. This pattern validates our uncertainty-guided sample selection strategy—by focusing on the most informative samples identified through ensemble disagreement, MSFlow achieves near-optimal performance with minimal supervision. The rapid initial improvement demonstrates that the soft router effectively learns to correct the ensemble's systematic errors in high-uncertainty regions, while the plateau at higher label percentages suggests that confident predictions require no correction. This label-efficient behavior is particularly valuable in practical deployments where obtaining comprehensive labels is costly or infeasible.

Anomaly Type Visualization. Figure 3 demonstrates MSFlow's multi-view scoring mechanism across five fundamental anomaly types: Global Point and Contextual Point (from YAHOO), and Seasonal, Trend, and Shapelet (from SVDB). The visualization reveals complementary detection patterns across different views. For point anomalies (first two columns), temporal reconstruction scores produce sharp, localized peaks precisely at anomaly positions, while frequency scores remain relatively stable. Conversely, for subsequence anomalies (last three columns), frequency reconstruction scores maintain sustained elevation throughout the entire anomalous intervals, effectively capturing extended pattern violations that temporal methods might fragment. Notably, the seasonal anomaly shows frequency clustering scores dominating the detection, while trend and shapelet anomalies benefit from both temporal and frequency reconstruction. This view-specific sensitivity validates our multi-perspective design—each detection method contributes uniquely based on anomaly characteristics, enabling MSFlow to achieve robust detection across diverse anomaly types through coupling flow-based fusion.

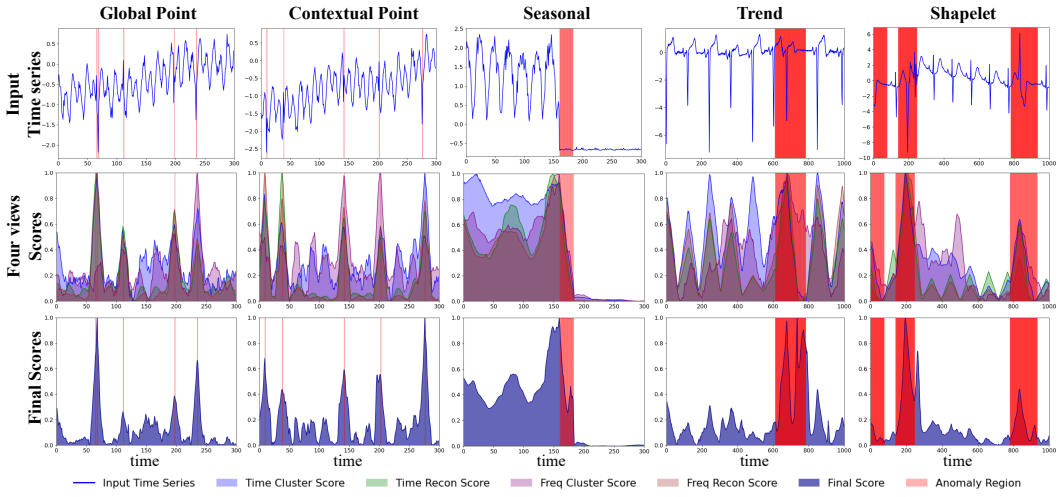


Figure 3: Multi-view anomaly detection across five types. Rows show: input time series, four view scores (temporal clustering, temporal reconstruction, frequency clustering, frequency reconstruction), and final fused scores. Red regions indicate ground-truth anomalies.

5 Conclusion

In this paper, we presented MSFlow, a multi-view anomaly detection framework that addresses two fundamental challenges in time series analysis: the diversity of anomaly patterns and the flexible utilization of available supervision. Through coupling flow-based score fusion, MSFlow effectively integrates complementary detection perspectives from temporal and frequency domains, capturing both point and subsequence anomalies that single methods often miss. The uncertainty-guided enhancement mechanism enables the framework to adaptively leverage labeled data when available while maintaining strong unsupervised performance. Extensive experiments on 18 diverse benchmarks demonstrate MSFlow achieves state-of-the-art performance.

Ethics Statement

Our work exclusively uses publicly available benchmark datasets that contain no personally identifiable information. The proposed anomaly detection framework is designed for beneficial applications in system reliability and safety monitoring. No human subjects were involved in this research.

Reproducibility Statement

To ensure reproducibility, we provide: (1) Complete implementation details including all hyperparameters in Appendix A.3; (2) Source code and scripts are provided in an anonymous repository at [`https://anonymous.4open.science/r/MSflow-020D`]; (3) All experiments use the standard public benchmark (TAB) protocols and implementations with documented preprocessing steps; (4) Fixed random seeds for all stochastic components.

References

L. Glass J. M. Hausdorff P. C. Ivanov R. G. Mark J. E. Mietus G. B. Moody C.-K. Peng A. L. Goldberger, L. A. Amaral and H. E. Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000.

Ahmed Abdulaal, Zhuanghua Liu, and Tomer Lancewicki. Practical approach to asynchronous multivariate time series anomaly detection and localization. In *SIGKDD*, pp. 2485–2494, 2021.

Subutai Ahmad, Alexander Lavin, Scott Purdy, and Zuha Agha. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing*, 262:134–147, 2017.

Mohiuddin Ahmed, Raihan Seraj, and Syed Mohammed Shamsul Islam. The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, 9(8):1295, 2020.

Differences Using Graph Attention. Multivariate time series anomaly detection based on reconstructed differences using graph attention networks. *Current Trends in Web Engineering*, pp. 58.

Pawel Benecki, Szymon Piechaczek, Daniel Kostrzewa, and Jakub Nalepa. Detecting anomalies in spacecraft telemetry using evolutionary thresholding and lstms. In *GECCO*, pp. 143–144, 2021.

Aadyot Bhatnagar, Paul Kassianik, Chenghao Liu, Tian Lan, Wenzhuo Yang, Rowan Cassius, Doyen Sahoo, Devansh Arpit, Sri Subramanian, Gerald Woo, Amrita Saha, Arun Kumar Jagota, Gokulakrishnan Gopalakrishnan, Manpreet Singh, K C Krithika, Sukumar Maddineni, Daeki Cho, Bo Zong, Yingbo Zhou, Caiming Xiong, Silvio Savarese, Steven Hoi, and Huan Wang. Merlion: A machine learning library for time series. 2021.

Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, pp. 93–104, 2000.

Chris U Carmona, François-Xavier Aubet, Valentin Flunkert, and Jan Gasthaus. Neural contextual anomaly detection for time series. *arXiv preprint arXiv:2107.07702*, 2021.

Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys*, 41(3):1–58, 2009.

Ching Chang, Wei-Yao Wang, Wen-Chih Peng, and Tien-Fu Chen. Llm4ts: Aligning pre-trained llms as data-efficient time-series forecasters. *ACM Transactions on Intelligent Systems and Technology*, 16(3):1–20, 2025.

Yuwei Cui, Chetan Surpur, Subutai Ahmad, and Jeff Hawkins. A comparative study of htm and other neural network models for online sequence learning with streaming data. In *IJCNN*, pp. 1530–1538, 2016.

Jesse Davis and Mark Goadrich. The relationship between precision-recall and roc curves. In *ICML*, pp. 233–240, 2006.

Tom Fawcett. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874, 2006.

Hsin-Yu Feng, Po-Ying Chen, and Janpu Hou. Sr-scatnet algorithm for on-device ecg time series anomaly detection. In *SoutheastCon 2021*, pp. 1–5. IEEE, 2021.

Pavel Filonov, Andrey Lavrentyev, and Artem Vorontsov. Multivariate industrial time series with cyber-attack simulation: Fault detection using an lstm-based predictive data model. *arXiv preprint arXiv:1612.06676*, 2016.

Shanghua Gao, Teddy Koker, Owen Queen, Tom Hartvigsen, Theodoros Tsiligkaridis, and Marinka Zitnik. Units: A unified multi-task time series model. *Advances in Neural Information Processing Systems*, 37:140589–140631, 2024.

Symon Gathiaka, Shuai Liu, Michael Chiu, Huanwang Yang, Jeanne A Stuckey, You Na Kang, Jim Delproposto, Ginger Kubish, James B Dunbar Jr, Heather A Carlson, et al. D3r grand challenge 2015: evaluation of protein–ligand pose and affinity predictions. *Journal of computer-aided molecular design*, 30(9):651–668, 2016.

Markus Goldstein and Andreas Dengel. Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm. *KI-2012: poster and demo track*, 1:59–63, 2012.

Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1705–1714, 2019.

Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. *arXiv preprint arXiv:2402.03885*, 2024.

S. D. Greenwald. *Improved detection and classification of arrhythmias in noise-corrupted electrocardiograms using contextual information*. 1990.

Ralf Greis, T Reis, and C Nguyen. Comparing prediction methods in anomaly detection: an industrial evaluation. In *Proceedings of the Workshop on Mining and Learning from Time Series*, 2018.

Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. Knn model-based approach in classification. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pp. 986–996. Springer, 2003.

Sahand Hariri, Matias Carrasco Kind, and Robert J Brunner. Extended isolation forest. *IEEE transactions on knowledge and data engineering*, 33(4):1479–1489, 2019.

Zengyou He, Xiaofei Xu, and Shengchun Deng. Discovering cluster-based local outliers. *Pattern recognition letters*, 24(9-10):1641–1650, 2003.

Niklas Heim and James E Avery. Adaptive anomaly detection in chaotic time series with a spatially aware echo state network. *arXiv preprint arXiv:1909.01709*, 2019.

Alexis Huet, Jose Manuel Navarro, and Dario Rossi. Local evaluation of time series anomaly detection algorithms. In *SIGKDD*, pp. 635–645, 2022.

Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom. Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding. In *SIGKDD*, pp. 387–395, 2018.

Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. 2018.

Atsushi Inoue and Mototsugu Shintani. Bootstrapping gmm estimators for time series. *Journal of Econometrics*, 133(2):531–555, 2006.

Vincent Jacob, Fei Song, Arnaud Stiegler, Bijan Rad, Yanlei Diao, and Nesime Tatbul. Exathlon: A benchmark for explainable anomaly detection over time series. *arXiv preprint arXiv:2010.05073*, 2020.

Dutta Roy T. Naik U. Agrawal Keogh, E. Multi-dataset time-series anomaly detection competition. In *SIGKDD*, 2021.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kwei-Herng Lai, Daochen Zha, Junjie Xu, Yue Zhao, Guanchu Wang, and Xia Hu. Revisiting time series outlier detection: Definitions and benchmarks. In *Proceedings of the conference on neural information processing systems datasets and benchmarks track (round 1)*, 2021.

N Laptev, S Amizadeh, and Y Billawala. S5-a labeled anomaly detection dataset, version 1.0 (16m), 2015.

Kun-Lun Li, Hou-Kuan Huang, Sheng-Feng Tian, and Wei Xu. Improving one-class svm for anomaly detection. In *Proceedings of the 2003 international conference on machine learning and cybernetics (IEEE Cat. No. 03EX693)*, volume 5, pp. 3077–3081. IEEE, 2003.

Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 eighth ieee international conference on data mining*, pp. 413–422. IEEE, 2008.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *ICLR*, 2024a.

Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer: Generative pre-trained transformers are large time series models. *arXiv preprint arXiv:2402.02368*, 2024b.

Donghao Luo and Xue Wang. Moderntcn: A modern pure convolution structure for general time series analysis. In *ICLR*, 2024.

Youngeun Nam, Susik Yoon, Yooju Shin, Minyoung Bae, Hwanjun Song, Jae-Gil Lee, and Byung Suk Lee. Breaking the time-frequency granularity discrepancy in time-series anomaly detection. In *WWW*, pp. 4204–4215. ACM, 2024a.

Youngeun Nam, Susik Yoon, Yooju Shin, Minyoung Bae, Hwanjun Song, Jae-Gil Lee, and Byung Suk Lee. Breaking the time-frequency granularity discrepancy in time-series anomaly detection. In *Proceedings of the ACM Web Conference 2024*, pp. 4204–4215, 2024b.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *ICLR*, 2023.

Zijian Niu, Ke Yu, and Xiaofei Wu. Lstm-based vae-gan for time-series anomaly detection. *Sensors*, 20(13):3738, 2020.

John Paparrizos, Paul Boniol, Themis Palpanas, Ruey S Tsay, Aaron Elmore, and Michael J Franklin. Volume Under the Surface: A New Accuracy Evaluation Measure for Time-Series Anomaly Detection. *Proc. VLDB Endow.*, 15(11):2774–2787, 2022.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, volume 32, 2019.

Tomáš Pevnỳ. Loda: Lightweight on-line detector of anomalies. *Machine Learning*, 102:275–304, 2016.

Xiangfei Qiu, Zhe Li, Wanghui Qiu, Shiyan Hu, Lekui Zhou, Xingjian Wu, Zhengyu Li, Chenjuan Guo, Aoying Zhou, Zhenli Sheng, et al. Tab: Unified benchmarking of time series anomaly detection methods. *arXiv preprint arXiv:2506.18046*, 2025.

Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim. Efficient algorithms for mining outliers from large data sets. In *Proceedings of the 2000 ACM SIGMOD international conference on management of data*, pp. 427–438, 2000.

Hansheng Ren, Bixiong Xu, Yujing Wang, Chao Yi, Congrui Huang, Xiaoyu Kou, Tony Xing, Mao Yang, Jie Tong, and Qi Zhang. Time-series anomaly detection service at microsoft. In *SIGKDD*, pp. 3009–3017, 2019.

Daniel Roggen, Alberto Calatroni, Mirco Rossi, Thomas Holleczek, Kilian Förster, Gerhard Tröster, Paul Lukowicz, David Bannach, Gerald Pirkl, Alois Ferscha, et al. Collecting complex activity datasets in highly rich networked sensor environments. In *INSS*, pp. 233–240, 2010.

Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International conference on machine learning*, pp. 4393–4402. PMLR, 2018.

Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *MLSDA*, pp. 4–11, 2014.

Sebastian Schmidl, Phillip Wenig, and Thorsten Papenbrock. Anomaly detection in time series: a comprehensive evaluation. *Proc. VLDB Endow.*, 15(9):1779–1797, 2022.

Bernhard Schölkopf, Robert C Williamson, Alex Smola, John Shawe-Taylor, and John Platt. Support vector method for novelty detection. In *NeurIPS*, volume 12, 1999.

Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21, 2017.

Mei-Ling Shyu, Shu-Ching Chen, Kanoksri Sarinnapakorn, and LiWu Chang. A novel anomaly detection scheme based on principal component classifier. In *Proceedings of the IEEE foundations and new directions of data mining workshop*, pp. 172–179, 2003.

Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2828–2837, 2019.

Jian Tang, Zhixiang Chen, Ada Wai-Chee Fu, and David W Cheung. Enhancing effectiveness of outlier detections for low density patterns. In *PAKDD*, pp. 535–548, 2002.

Nesime Tatbul, Tae Jun Lee, Stan Zdonik, Mejbah Alam, and Justin Gottschlich. Precision and recall for time series. *NeurIPS*, 31, 2018.

Markus Thill, Wolfgang Konen, and Thomas Bäck. Time series anomaly detection with discrete wavelet transforms and maximum likelihood estimation. In *ITISE*, volume 2, pp. 11–23, 2017.

Markus Thill, Wolfgang Konen, and Thomas Bäck. Markusthill/mgab: The mackey-glass anomaly benchmark. *Version v1. 0.1. Zenodo. doi*, 10, 2020.

Shreshth Tuli, Giuliano Casale, and Nicholas R Jennings. Tranad: Deep transformer networks for anomaly detection in multivariate time series data. *arXiv preprint arXiv:2201.07284*, 2022.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*, 2022.

Xingjian Wu, Xiangfei Qiu, Zhengyu Li, Yihang Wang, Jilin Hu, Chenjuan Guo, Hui Xiong, and Bin Yang. Catch: Channel-aware multivariate time series anomaly detection via frequency patching. *arXiv preprint arXiv:2410.12261*, 2024.

Julian Wyatt, Adam Leach, Sebastian M Schmon, and Chris G Willcocks. Anoddpm: Anomaly detection with denoising diffusion probabilistic models using simplex noise. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 650–656, 2022.

Haowen Xu, Wenxiao Chen, Nengwen Zhao, Zeyan Li, Jiahao Bu, Zhihan Li, Ying Liu, Youjian Zhao, Dan Pei, Yang Feng, et al. Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications. In *Proceedings of the 2018 world wide web conference*, pp. 187–196, 2018.

Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly transformer: Time series anomaly detection with association discrepancy. *arXiv preprint arXiv:2110.02642*, 2021.

Takehisa Yairi, Yoshikiyo Kato, and Koichi Hori. Fault detection by mining association rules from house-keeping data. In *i-SAIRAS*, volume 3, 2001.

Yiyuan Yang, Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. Dcdetector: Dual attention contrastive representation learning for time series anomaly detection. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 3033–3045, 2023.

Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. Matrix profile i: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In *2016 IEEE 16th international conference on data mining (ICDM)*, pp. 1317–1322. Ieee, 2016.

Jihun Yi and Sungroh Yoon. Patch svdd: Patch-level svdd for anomaly detection and segmentation. In *Proceedings of the Asian conference on computer vision*, 2020.

Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Congrui Huang, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pp. 8980–8987, 2022.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *AAAI*, volume 37, pp. 11121–11128, 2023.

Chaoli Zhang, Tian Zhou, Qingsong Wen, and Liang Sun. Tfad: A decomposition time series anomaly detection architecture with time-frequency analysis. In *Proceedings of the 31st ACM international conference on information & knowledge management*, pp. 2497–2507, 2022.

Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V Chawla. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 1409–1416, 2019.

Hui Zhang, Zheng Wang, Dan Zeng, Zuxuan Wu, and Yu-Gang Jiang. Diffusionad: Norm-guided one-step denoising diffusion for anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Yu Zhou, Xiaomin Liang, Wei Zhang, Linrang Zhang, and Xing Song. Vae-based deep svdd for anomaly detection. *Neurocomputing*, 453:131–140, 2021.

Zhihao Zhuang, Yingying Zhang, Kai Zhao, Chenjuan Guo, Bin Yang, Qingsong Wen, and Lunting Fan. Noise matters: Cross contrastive learning for flink anomaly detection. *Proc. VLDB Endow.*, 2025.

Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International conference on learning representations*, 2018.

A EXPERIMENTAL DETAILS

A.1 DATASET

Table 4: Summary of evaluation datasets

Type	Dataset	Domain	Dim	Series	Avg Total Length	Avg Test Length	Avg AR (%)	Train AR (%)
Univariate	GAIA ()	AIOps	1	184	9,777	8,799	1.26	0.19
	GHL (Filonov et al., 2016)	Machinery	1	1	200,001	180,001	0.43	0.00
	KDD21 (Keogh, 2021)	Multiple	1	243	65,903	47,294	0.58	0.00
	MGAB (Thill et al., 2020)	Mackey-Glass	1	6	100,000	90,000	0.20	0.00
	NASA-MSL (Benecki et al., 2021)	Spacecraft	1	22	5,167	2,897	3.85	0.00
	NAB (Ahmad et al., 2017)	Multiple	1	45	7,115	3,557	9.84	6.55
	OPPORTUNITY (Roggen et al., 2010)	Movement	1	462	31,359	28,223	4.12	19.11
	NASA-SMAP (Benecki et al., 2021)	Spacecraft	1	35	10,539	7,930	2.20	0.00
	SVDB (Greenwald, 1990)	Health	1	52	230,400	207,360	4.87	5.48
	YAHOO (Laptev et al., 2015)	Multiple	1	346	1,570	785	0.63	0.46
Multivariate	LTDB (A. L. Goldberger & Stanley, 2000)	Health	2	5	100,000	87,285	15.57	11.72
	MITDB (A. L. Goldberger & Stanley, 2000)	Health	2	6	141,667	106,250	2.72	0.00
	MSL (Hundman et al., 2018)	Spacecraft	55	1	132,046	73,729	5.88	0.00
	SMD (Su et al., 2019)	Server Machine	38	1	1,416,825	708,420	2.08	0.00
	NYC (Cui et al., 2016)	Transport	3	1	17,520	4,416	0.57	0.00
	PSM (Abdulaal et al., 2021)	Server Machine	25	1	220,322	87,841	11.07	0.00
	SMAP (Hundman et al., 2018)	Spacecraft	25	1	562,800	427,617	9.72	0.00
	SVDB (Greenwald, 1990)	Health	2	6	110,133	86,373	3.14	5.48

AR: anomaly ratio

In order to comprehensively evaluate the performance of MSflow, we evaluate 18 diverse time series anomaly detection benchmarks covering both 10 univariate datasets and 8 multivariate datasets, spanning multiple domains including spacecraft telemetry, server monitoring, healthcare, movement analysis, and industrial systems. These datasets exhibit varied characteristics in terms of dimensionality (1-55 channels), sequence lengths (785-1,416,825 points), test anomaly ratios (0.20%-15.57%), and training anomaly ratios (0.00%-19.11%), ensuring robust validation across different operational contexts. Table 4 lists statistics of the 18 datasets. As shown in Table 4, we conduct comprehensive evaluation on 18 diverse time series anomaly detection benchmarks covering both univariate (10 datasets) and multivariate (8 datasets) scenarios

As shown in Table 4, we conduct comprehensive evaluation on 18 diverse time series anomaly detection benchmarks covering both univariate (10 datasets) and multivariate (8 datasets) scenarios

A.2 METRICS

The metrics we support can be divided into two categories: Score-based and Label-based. Label-based metrics includes Accuracy (Acc), Precision (P), Recall (R), F1-score ($F1$), Range-Precision (R-P), Range-Recall (R-R), Range-F1-score (R-F) (Tatbul et al., 2018), Precision@k, Affiliated-Precision (Aff-P), Affiliated-Recall (Aff-R), and Affiliated-F1-score (Aff-F) (Huet et al., 2022). Score-based metrics includes the Area Under the Precision-Recall Curve (A-P) (Davis & Goadrich, 2006), the Area under the Receiver Operating Characteristics Curve (A-R) (Fawcett, 2006), the Range Area Under the Precision-Recall Curve (R-A-P), the Range Area under the Receiver Operating Characteristics Curve (R-A-R) (Paparrizos et al., 2022), the Volume Under the Surface of Precision-Recall (V-PR), and the Volume Under the Surface of Receiver Operating Characteristic (V-ROC) (Paparrizos et al., 2022). While we report Affiliated-F1 and AUC-ROC in the main paper for clarity, complete results across all metrics for every dataset and baseline method are provided in our repository at [`https://anonymous.4open.science/r/MSflow-020D`]. More implementation details are presented in the Appendix A.3.

A.3 IMPLEMENTATION DETAILS

Dataset Splitting. We follow the original train-test splits when provided. Otherwise, we use a 50%-50% split for training and testing. For baseline methods requiring validation, we extract the last 20% from the training set as validation data. Our method does not use validation sets and utilizes the full training portion.

Training Configuration. All baselines use their official implementations with default hyperparameters. We set batch size to 128 (or 64 when encountering OOM issues) and apply early stopping with patience of 10 epochs for deep learning methods. For memory-intensive models, we adopt stride-doubling to maintain computational feasibility.

Threshold Selection. Due to threshold sensitivity in anomaly detection, we evaluate across $\tau \in \{0.1, 0.5, 1, 2, 3, 5, 10, 15, 20, 25\}$ (percentiles) for both univariate and multivariate settings, reporting the best performance for each method-dataset pair. All metrics are reported as percentages.

Computational Setup. Experiments are conducted using Python 3.8 with PyTorch 1.12.0 (Paszke et al., 2019) on NVIDIA Tesla-A800 GPUs. Random seeds are fixed at 42 for reproducibility. We employ point-adjust (PA) evaluation (Xu et al., 2018) and use bias parameter 0.2 for segment-based metrics following (Huet et al., 2022).

B Method-Specific Detection Bias Analysis

To empirically validate the existence of method-specific biases, we conducted an extensive analysis across 35 anomaly detection methods spanning three paradigms, evaluating them on univariate datasets categorized by their primary anomaly patterns: point anomalies (including global and contextual anomalies), subsequence anomalies (seasonal, trend, and shapelet anomalies), and mixed cases combining multiple types. Table 5 lists all evaluated methods with the 18 baseline methods from our main experiments highlighted in blue. All experiments followed fair and consistent evaluation protocols from the TAB benchmark (Qiu et al., 2025), enabling systematic analysis of detection capabilities across different anomaly granularities.

Table 5: Evaluated methods across three detection paradigms

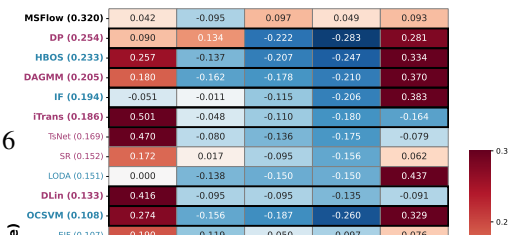
Paradigm	Methods (Abbreviation)
Statistical Distribution-based (14 methods)	LOF (Breunig et al., 2000), KNN (Ramaswamy et al., 2000), CBLOF (He et al., 2003), KMeans (Yairi et al., 2001), LSTi (Yeh et al., 2016), ZMS (Bhatnagar et al., 2021), HBOS (Goldstein & Dengel, 2012), Stat (Bhatnagar et al., 2021), DWT (Thill et al., 2017), OCSVM (Schölkopf et al., 1999), LODA (Pevnỳ, 2016), IF (Liu et al., 2008), EIF (Hariri et al., 2019), COF (Tang et al., 2002)
Reconstruction-based (19 methods)	ARIMA (Hyndman & Athanasopoulos, 2018), SARIMA (Greis et al., 2018), Torsk (Heim & Avery, 2019), LSTM (Bhatnagar et al., 2021), DP (Bhatnagar et al., 2021), SR (Ren et al., 2019), PCA (Shyu et al., 2003), DAGMM (Zong et al., 2018), iTrans (Liu et al., 2024a), TsNet (Wu et al., 2022), ATrans (Xu et al., 2021), Patch (Nie et al., 2023), Modern (Luo & Wang, 2024), TranAD (Tuli et al., 2022), DualTF (Nam et al., 2024a), AE (Sakurada & Yairi, 2014), VAE (Kingma & Welling, 2013), NLin (Zeng et al., 2023), DLin (Zeng et al., 2023)
Discriminative Learning (2 methods)	DC (Yang et al., 2023), ConAD (Zhuang et al., 2025)

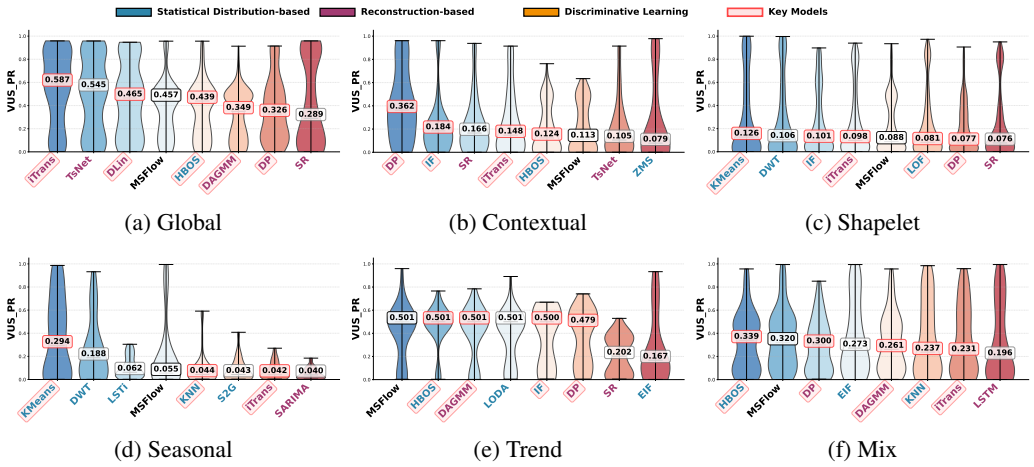
Note: Blue indicates baseline methods from main experiments.

Key Findings. The combined analysis of Figures 4 and 5 reveals three important observations about method-specific detection biases:

(1) High-performance methods exhibit specialized detection capabilities. Figure 4 shows that top-performing methods display pronounced specialization patterns. For instance, iTrans achieves exceptional performance on global anomalies with strong positive bias, yet shows reduced effectiveness and negative bias in seasonal detection. Similarly, DP demonstrates strong capabilities for contextual anomalies while showing limitations in subsequence detection, and KMeans excels at seasonal anomalies but shows negative bias for global detection. The heat map indicates that methods with higher average performance generally display more pronounced specialization patterns, whereas lower-ranked methods tend toward more uniform but modest performance across categories.

(2) Paradigm characteristics influence detection preferences. The performance distributions in Figure 5 indicate systematic tenden-





(a) Global (b) Contextual (c) Shapelet

(d) Seasonal (e) Trend (f) Mix

Figure 5: Performance distributions across anomaly types. Top performers shown per category, illustrating method specialization patterns.

cies across paradigms. Deep learning methods (iTrans, TsNet, DLin) frequently appear among top performers for global anomalies but are less prominent in subsequence categories. Statistical approaches demonstrate complementary strengths—KMeans and DWT dominate seasonal and trend detection, while HBOS excels at shapelet patterns. The heat map further supports this observation: reconstruction methods tend toward positive biases for point anomalies and negative biases for subsequence types, while statistical methods often exhibit the opposite pattern.

(3) Mixed anomalies present universal challenges. Figure 5f reveals that even the best-performing methods achieve only modest performance on mixed anomalies, with multiple methods (MSFlow, HBOS, DAGMM, LODA) plateauing at similar levels. This performance ceiling, considerably lower than peak performances on single-type categories, suggests that scenarios combining multiple anomaly patterns pose significant challenges that current single-method approaches struggle to address effectively.

MSFlow's Superior Performance. These observations validate MSFlow's multi-view fusion strategy. By leveraging coupling flows to combine complementary detection perspectives, MSFlow successfully addresses individual method limitations. Empirical results confirm MSFlow achieves the highest average performance and consistently ranks among the top performers across all six anomaly categories. Unlike specialized methods that excel in specific domains but struggle elsewhere (as shown by the extreme biases in Figure 4), MSFlow maintains balanced performance with minimal bias variations across categories, demonstrating that principled fusion

of complementary perspectives provides a more robust solution than pursuing universal capability within single architectures.