PaAno: Patch-Based Representation Learning for Time-Series Anomaly Detection

Jinju Park, Seokho Kang*

Department of Industrial Engineering, Sungkyunkwan University, Korea

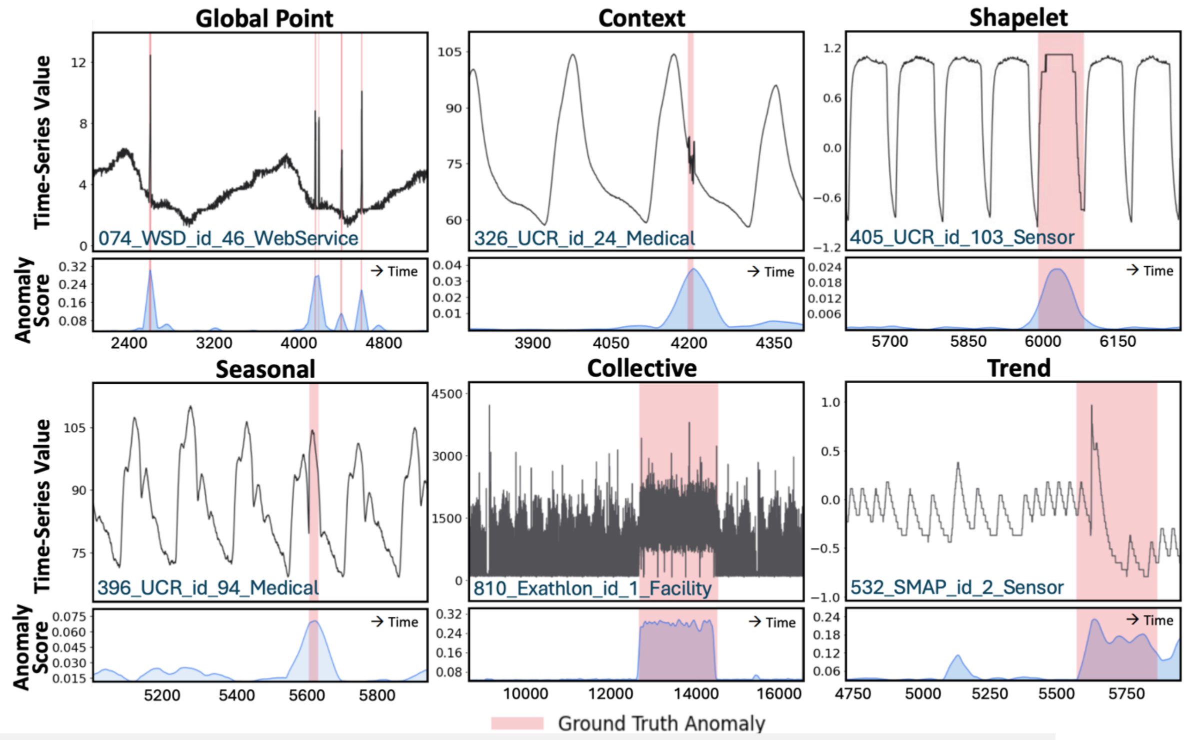
Apr 2026, International Conference on Learning Representations (ICLR 2026)

Paper Code

1. Introduction

1-1. Time-Series Anomaly Detection

Time-series Anomaly Detection aims to identify time points or segments within a sequence whose patterns deviate significantly from expected normal behavior.



[Figure 1] Examples of several types of time-series anomalies. (Black: Normal, Red: Anomaly)

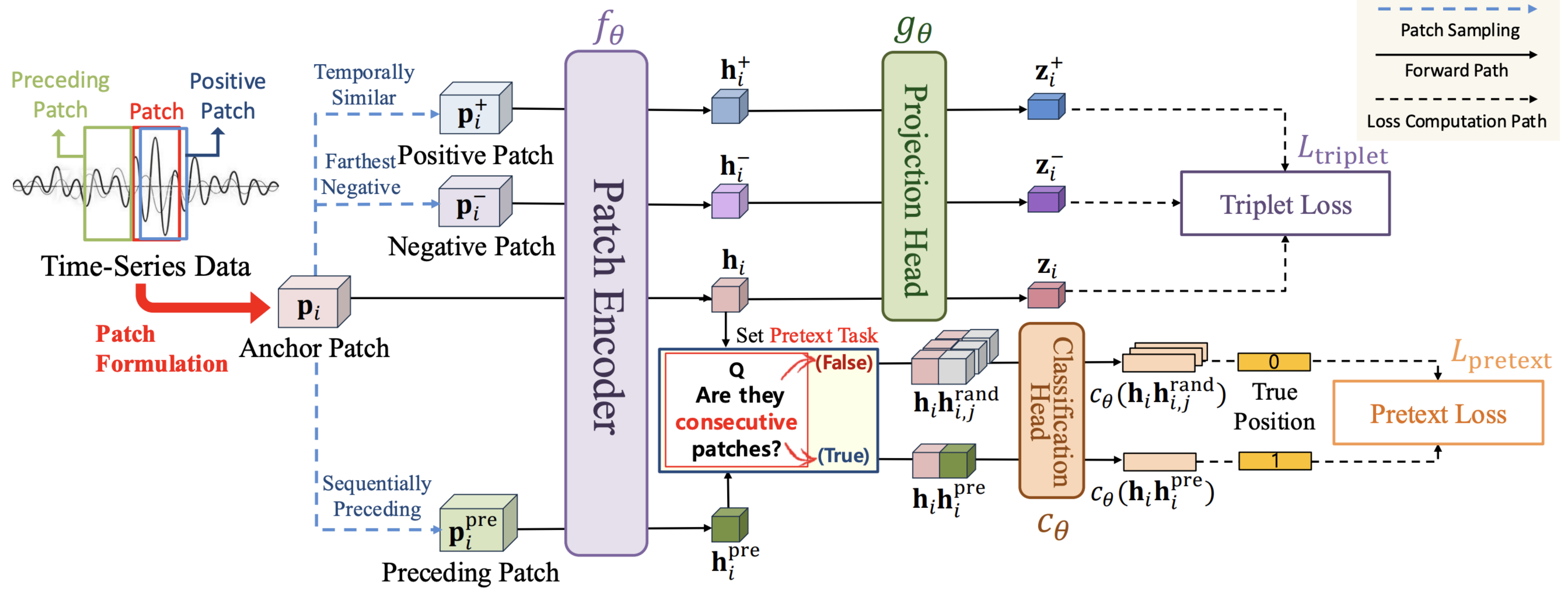
1-2. Issues in Recent Time-Series Anomaly Detection

However, recent studies have revealed that common practices in time-series anomaly detection evaluation suffer from serious flaws. **1) Benchmark datasets contain mislabeled anomalies or unrealistic assumptions about anomaly distributions**, and **2) Evaluation Protocols rely on point-adjustment and method-specific threshold tuning**, which can artificially inflate performance. Furthermore, Recent studies also show that, under corrected protocols, recent state-of-the-art transformer-based time-series anomaly detection methods did not provide meaningful improvements over simpler methods.

2. Proposed Method

2-1. Training Phase

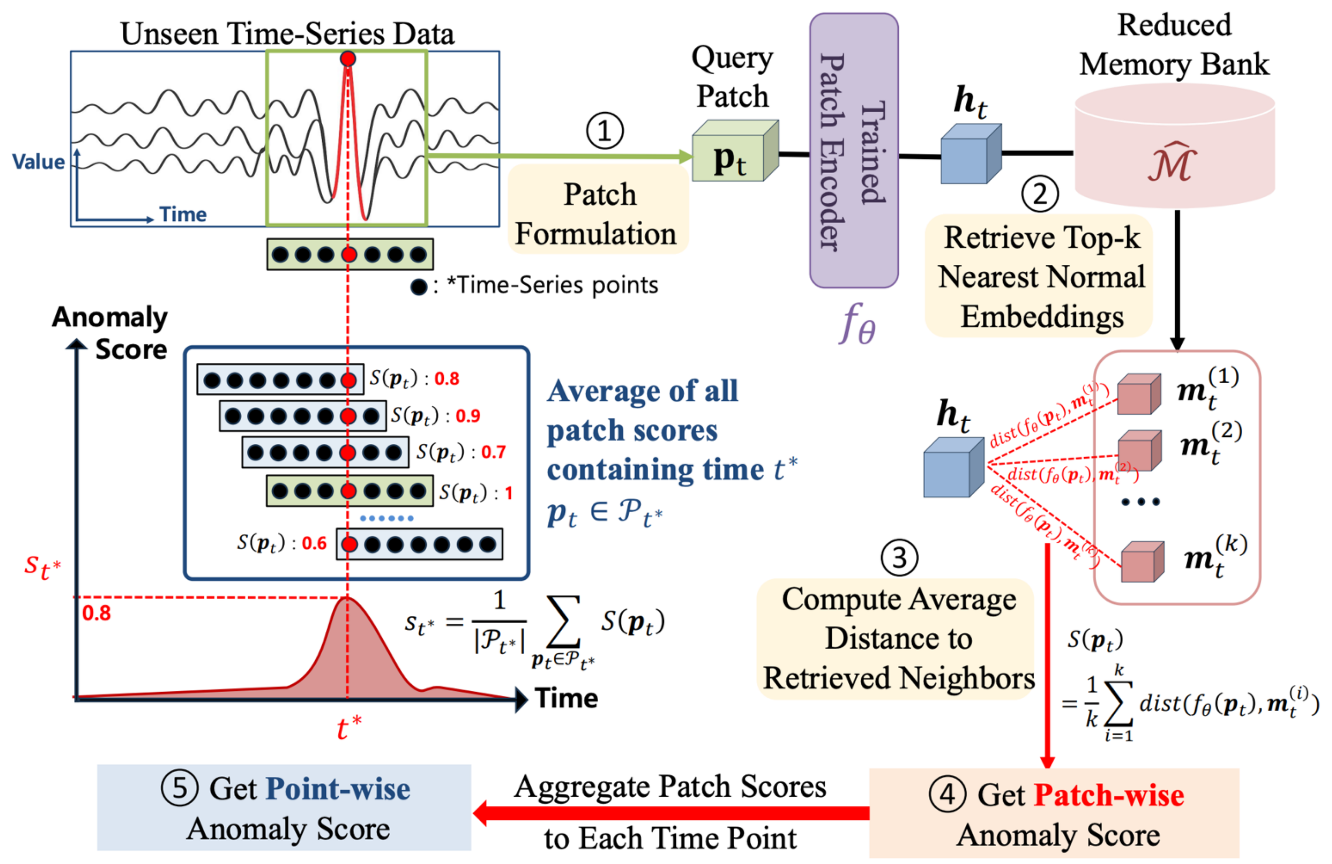
PaAno embeds instance-normalized patches into a discriminative embedding space using a **lightweight 1D-CNN encoder**. The training objective minimizes a **triplet loss** to ensure that patches with similar temporal patterns are embedded close together while dissimilar ones are pushed apart, creating **well-organized clusters of normal patches**. To stabilize this representation learning, a **pretext loss** is applied during the **early stage of training** to capture sequential relationships by predicting whether two patches are temporally consecutive. This joint optimization encourages the encoder to extract features sensitive to meaningful temporal dynamics, resulting in a **robust and structured embedding space** where unseen anomalies can be effectively identified.



[Figure 2] Training procedure of PaAno.

2-2. Anomaly Detection

During inference, PaAno computes anomaly scores using the trained **patch encoder** and a **reduced memory bank**. Each patch containing the query time step is embedded and compared against its **k-nearest neighbors** in the memory bank via cosine distance. This patch-level score measures the deviation from **learned normal clusters**, as anomalous embeddings fail to align with normal patterns. The final anomaly score is determined by **averaging the scores of all overlapping patches**.



[Figure 3] Anomaly detection procedure of PaAno.

3. Experimental Results

3-1. Univariate Time-Series Anomaly Detection

For evaluation metrics, we used three range-based metrics (VUS-PR, VUS-ROC, Range-F1), and three point-wise metrics (AUC-PR, AUC-ROC, Point-F1). We used TSB-AD dataset for evaluation. In univariate time-series anomaly detection, PaAno ranked first across all six performance measures, outperforming all baseline methods.

[Table 1] Experiment results of PaAno on TSB-AD-U.

*Best is **bold** and Second-Best is underlined. All scores are reported with their rankings as "Score/Rank".

		Range-Wise Measure ↑			Point-Wise Measure ↑			Computational Cost ↓	
	Method	**VUS-PR**	VUS-ROC	Range-F1	AUC-PR	AUC-ROC	Point-F1	# Params	Run Time
Stat&ML	(Sub)-PCA (2017)	0.42/3	0.76/9	0.41/3	0.37/3	0.71/11	0.42/3	–	1.5s
	KShapeAD (2017)	0.40/4	0.76/9	0.40/4	0.35/4	0.74/5	0.39/4	–	8.0s
	DLinear (2023)	0.25/20+	0.74/19	0.22/20+	0.21/20+	0.62/20+	0.26/20+	< 0.1M	2.9s
	NLinear (2023)	0.23/20+	0.72/20+	0.20/20+	0.18/20+	0.62/20+	0.23/20+	< 0.1M	5.8s
NN	DeepAnT (2018)	0.34/13	0.79/5	0.35/9	0.33/5	0.71/11	0.38/5	< 0.1M	2.0s
	USAD (2020)	0.36/10	0.71/20+	0.40/4	0.32/7	0.66/20+	0.37/8	< 0.1M	1.7s
	TimesNet (2022)	0.26/20+	0.72/20+	0.21/20+	0.18/20+	0.61/20+	0.24/20+	< 0.1M	11.2s
	FITS (2023)	0.26/20+	0.73/20+	0.20/20+	0.17/20+	0.61/20+	0.23/20+	< 0.1M	3.1s
	DADA (2025)	0.31/17	0.77/8	0.31/19	0.29/14	0.71/11	0.38/18	1.84M	0.8s
	KAN-AD (2025)	<u>0.43</u>/2	<u>0.82</u>/2	<u>0.43</u>/2	<u>0.41</u>/2	<u>0.80</u>/2	<u>0.44</u>/2	< 0.1M	12.1s
Transformer	AnomalyTransformer (2021)	0.12/20+	0.56/20+	0.14/20+	0.08/20+	0.50/20+	0.12/20+	4.8M	48.9s
	DCdetector (2023)	0.09/20+	0.56/20+	0.10/20+	0.05/20+	0.50/20+	0.10/20+	0.9M	5.8s
	Lag-Llama (2023)	0.27/21	0.72/20+	0.31/19	0.25/20+	0.65/20+	0.30/20+	2.5M	1220.8s
	OFA (2023b)	0.24/20+	0.71/20+	0.20/20+	0.16/20+	0.59/20+	0.22/20+	81.9M	171.1s
	PatchTST (2023)	0.26/20+	0.75/17	0.22/20+	0.21/20+	0.63/20+	0.25/20+	0.5M	26.3s
	iTransformer (2024)	0.22/20+	0.74/19	0.18/20+	0.16/20+	0.61/20+	0.21/20+	0.6M	9.8s
	MOMENT (FT) (2024)	0.39/5	0.76/9	0.35/9	0.30/12	0.69/15	0.35/12	109.6M	43.6s
	MOMENT (ZS) (2024)	0.38/8	0.75/17	0.36/7	0.30/12	0.68/16	0.35/12	109.6M	42.9s
	TimesFM (2024)	0.30/18	0.74/19	0.34/14	0.28/17	0.67/19	0.34/16	203.5M	83.8s
	PaAno (Ours)	**0.52/1**	**0.89/1**	**0.48/1**	**0.46/1**	**0.86/1**	**0.51/1**	**0.3M**	**6.9s**

3-2. Multivariate Time-Series Anomaly Detection

In multivariate time-series anomaly detection, PaAno again ranked first across all six performance measures. PaAno showed highly competitive run time, highlighting its practical efficiency for real-time applications. While majority of recent Transformer-based baselines required significantly longer run times, PaAno was substantially faster with superior performances.

[Table 2] Experiment results of PaAno on TSB-AD-M.

*Best is **bold** and Second-Best is underlined. All scores are reported with their rankings as "Score/Rank".

		Range-Wise Measure ↑			Point-Wise Measure ↑			Computational Cost ↓	
	Method	**VUS-PR**	VUS-ROC	Range-F1	AUC-PR	AUC-ROC	Point-F1	# Params	Run Time
Stat&ML	KMeansAD (2001)	0.29/11	0.73/6	0.33/7	0.25/14	0.69/6	0.31/13	–	62.0s
	PCA (2017)	0.31/3	0.74/4	0.29/11	0.31/4	0.70/4	0.37/3	–	0.1s
	DLinear (2023)	0.29/11	0.70/12	0.26/15	0.27/9	0.66/12	0.32/9	< 0.1M	14.8s
	NLinear (2023)	0.29/11	0.70/12	0.28/12	0.24/15	0.65/14	0.31/13	< 0.1M	15.0s
NN	DeepAnT (2018)	0.31/3	<u>0.76</u>/2	<u>0.37</u>/4	<u>0.32</u>/3	<u>0.73</u>/2	0.37/3	< 0.1M	9.5s
	OmniAnomaly (2019)	0.31/3	0.69/16	<u>0.37</u>/4	0.27/9	0.65/14	0.32/9	< 0.1M	9.1s
	TimesNet (2022)	0.19/20+	0.64/20+	0.17/20+	0.13/20+	0.56/20+	0.20/20+	< 0.1M	52.1s
	FITS (2023)	0.21/21	0.66/20+	0.16/20+	0.15/20+	0.58/20+	0.22/20+	< 0.1M	16.7s
	DADA (2025)	0.31/3	0.73/6	0.25/18	0.31/4	0.69/6	0.35/6	1.84M	2.1s
	KAN-AD (2025)	<u>0.41</u>/2	0.75/3	**0.41**/1	**0.38**/1	<u>0.73</u>/2	<u>0.42</u>/2	< 0.1M	31.9s
Transformer	AnomalyTransformer (2021)	0.12/20+	0.57/20+	0.14/20+	0.07/20+	0.52/20+	0.12/20+	4.8M	55.8s
	PatchTST (2023)	0.28/15	0.71/9	0.26/15	0.26/12	0.65/14	0.32/9	0.5M	66.9s
	OFA (2023b)	0.21/20+	0.63/20+	0.17/20+	0.15/20+	0.55/20+	0.21/20+	81.9M	532.9s
	DCdetector (2023)	0.10/20+	0.56/20+	0.10/20+	0.06/20+	0.50/20+	0.10/20+	0.9M	15.0s
	iTransformer (2024)	0.29/11	0.70/12	0.23/20+	0.23/18	0.63/20+	0.28/18	0.6M	24.4s
	CATCH (2025)	0.30/8	0.73/6	0.27/14	0.24/15	0.67/8	0.30/16	210.8M	40.1s
	PaAno (Ours)	**0.43/1**	**0.79/1**	**0.41/1**	**0.38/1**	**0.76/1**	**0.43/1**	**0.3M**	**12.8s**

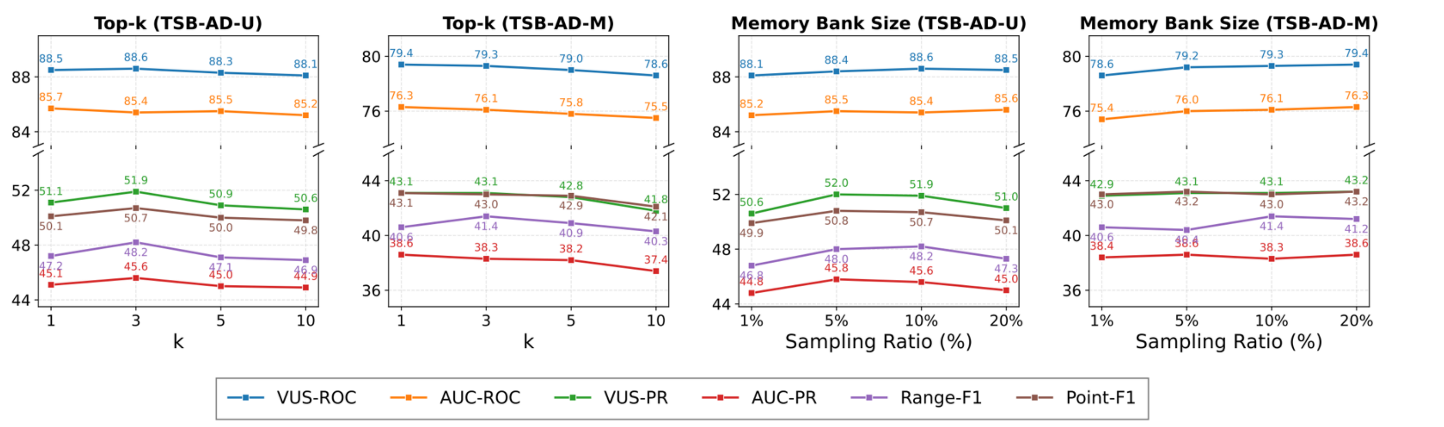
3-3. Comprehensive Experiments

We conducted a comprehensive experiment of PaAno. PaAno relies on all its components, and removing or altering any of them leads to a substantial drop in performance, as shown in **Table 3**. It is also robust to hyperparameter variations, with **Figure 4** presenting results for different memory bank sizes and numbers of nearest neighbors used in anomaly scoring, while additional settings (including the patch encoder architecture, loss weight, patch size, and minibatch size) are detailed in the paper. Furthermore, PaAno achieves competitive run time on TSB-AD-U **[Figure 5]**, and its efficiency advantage becomes more pronounced on TSB-AD-M, where it is even faster than linear models **[Figure 6].**

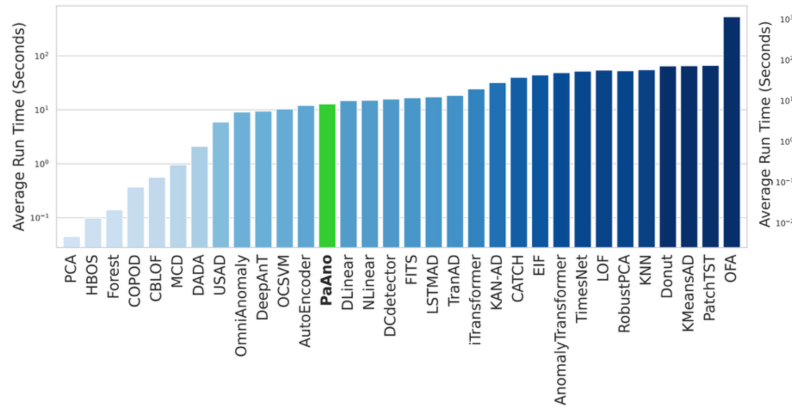
[Table 3] Ablation study on the core components of PaAno.

	Ablation Variant	VUS-PR	VUS-ROC	Range-F1	AUC-PR	AUC-ROC	Point-F1
TSB-AD-U	w/o InstanceNorm	46.3	85.0	46.3	42.6	82.5	47.3
	w/o $\mathcal{L}_{\text{triplet}}$ and $\mathcal{L}_{\text{pretext}}$	45.1	86.9	42.5	38.6	83.9	45.0
	w/o Negative Selection in $\mathcal{L}_{\text{triplet}}$	49.6	87.2	47.0	44.5	84.3	49.2
	Replace $\mathcal{L}_{\text{triplet}}$ with InfoNCE loss	48.4	86.7	45.3	42.6	83.6	47.8
	w/o $\mathcal{L}_{\text{pretext}}$	49.5	88.2	46.6	43.6	85.4	49.2
	Continuous Use of $\mathcal{L}_{\text{pretext}}$	46.8	87.1	45.3	40.4	84.0	45.8
	w/o Linear Decay on $\mathcal{L}_{\text{pretext}}$	50.7	88.4	48.2	45.6	85.4	50.7
	PaAno (Ours)	**51.9**	**88.6**	**48.2**	**45.6**	**85.7**	**50.7**
TSB-AD-M	w/o InstanceNorm	32.5	74.5	39.9	28.5	72.9	36.6
	w/o $\mathcal{L}_{\text{triplet}}$ and $\mathcal{L}_{\text{pretext}}$	35.5	76.2	34.8	30.5	72.9	36.5
	w/o Negative Selection in $\mathcal{L}_{\text{triplet}}$	36.9	76.7	35.1	32.4	73.5	37.2
	Replace $\mathcal{L}_{\text{triplet}}$ with InfoNCE loss	37.0	76.2	35.4	32.8	72.8	37.2
	w/o $\mathcal{L}_{\text{pretext}}$	41.0	79.1	38.8	36.2	76.0	41.3
	Continuous Use of $\mathcal{L}_{\text{pretext}}$	40.2	76.6	39.6	35.4	73.1	40.0
	w/o Linear Decay on $\mathcal{L}_{\text{pretext}}$	43.0	79.1	40.6	38.1	76.0	42.9
	PaAno (Ours)	**43.1**	**79.3**	**41.4**	**38.3**	**76.1**	**43.0**

[Figure 4] Sensitivity analysis on Top-k and memory bank size of PaAno across TSB-AD-U/M.



[Figure 5] Average Run Time on TSB-AD-U.



[Figure 6] Average Run Time on TSB-AD-M.