ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

A New Positional Encoding Loss for Anomaly Transformer in Time series Anomaly Detection

Khuu Trong Quan and Huynh Cong Viet Ngu*
khuutrongquan220405@gmail.com and nguhcv@fe.edu.vn

FPT University - Ho Chi Minh Campus
Lot E2a-7, D1 Street, Saigon Hi-Tech Park, Tang Nhon Phu Ward, HCMC - Vietnam

Abstract. Time-series anomaly detection (TSAD) is central to automated monitoring, where early detection of unexpected behaviors helps prevent system failures. Despite its strong performance, Anomaly Transformer remains limited by conventional positional encoding, which can cause positional duplication across input tokens and weaken temporal separability. To address this issue, we propose a learnable positional encoding (PE) module trained with a PE loss that explicitly penalizes duplicated positional representations, thereby improving temporal distinguishability. Experiments on three benchmarks—PSM, MSL, and SMAP—show consistent gains over the Anomaly Transformer baseline, improving F_1 by 0.76, 0.63, and 0.21 percentage points, respectively. These results suggest that regularizing positional representations is a simple and general way to strengthen Transformer-based TSAD. The main code is available in Github.

1 Introduction

Unexpected failures are common in long-running automated systems—including healthcare [1] and finance [2]—where continuous human supervision is impractical. Time series anomaly detection (TSAD) [3] is thus essential for identifying such failures accurately and promptly across diverse temporal data.

A wide range of TSAD methods has been studied, with deep models [3] proving effective at capturing temporal dependencies. Unsupervised formulations [3] are especially attractive because anomalies are rare and labeling is costly. In this unsupervised setting, we make two contributions:

- We propose a learnable positional encoding (PE) module with a PE loss that penalizes duplicated positional representations, improving token-level temporal distinguishability.
- We demonstrate consistent improvements over Anomaly Transformer and other SOTA methods on PSM, MSL, and SMAP.

The next section presents related work, followed by a detailed description of our main method in Section 3. Experimental results are reported in Section 4. Finally, we conclude the paper in the last section.

*Corresponding Author

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964.
Available from http://www.i6doc.com/en/.

2 Related Work

2.1 Anomaly Transformer and Positional Encoding

Among unsupervised deep TSAD methods [3], Transformer-based models [4] are effective at capturing long-range temporal dependencies via self-attention [4]. Anomaly Transformer [5] detects anomalies by exploiting discrepancies in attention distributions between normal and anomalous points. We adopt it as our baseline and improve temporal anomaly detectability by addressing a limitation of conventional positional encodings: they can yield insufficiently discriminative positional vectors for input tokens (Sec. 3).

Sinusoidal absolute positional encodings [4] are fixed functions of the time index and feature dimension, which can limit adaptation to diverse time-series distributions. To overcome this, we introduce a learnable PE module regularized by a PE loss that discourages positional duplication and enforces an ordered structure (Subsec. 3.2.2–3.3).

3 Main Contribution

Problem Definition Time series data is a collection of data points recorded over time. Each time step x_t has a d-dimensional vector of values, where $x_t \in R^d$. In the unsupervised TSAD task, each time step must be determined as anomalous or normal without requiring input labels, where anomalies are defined as points that deviate from general data distribution.

3.1 Overall Architecture

We propose a learnable positional encoding (PE) module regularized by a PE loss that minimizes positional duplication and enforces a prescribed structure (Subsec. 3.3). Before injecting PE into token representations, we introduce two learnable coefficient vectors to balance the contributions of the input features and positional information for subsequent processing. For input embedding, we adopt a Hybrid Encoder instead of a 1D-CNN (Subsec. 3.2). The overall architecture is illustrated in Fig. 1.

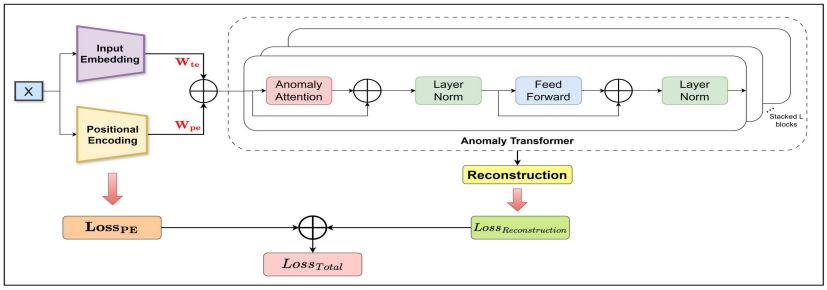

Fig. 1: Main Architecture of our method

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964.
Available from http://www.i6doc.com/en/.

In the following, we explain all improved components in detail, consisting of enhanced Data Embedding (Subsec. 3.2) and the proposed PE loss (Subsec. 3.3).

3.2 Data Embedding

The data embedding stage consists of two components: (i) an input embedding module that maps raw observations to richer token representations, and (ii) a positional encoding module that injects temporal position information into each token (Fig. 2).

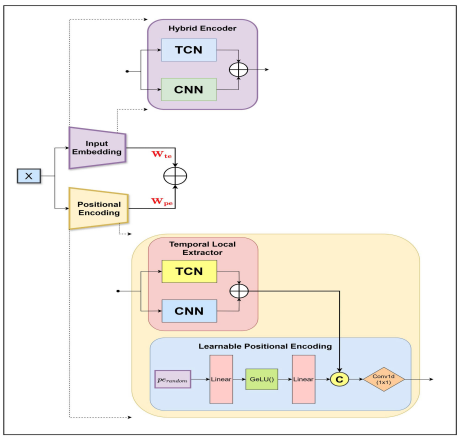

Fig. 2: Data Embedding: (1) Input Embedding by utilizing Hybrid Encoder block and (2) Positional Encoding using designed learnable Positional Encoding

3.2.1 Input Embedding module: Hybrid Encoder

The Hybrid Encoder integrates Temporal Convolutional Networks (TCN) [6] with dilated causal convolutions and CNN to extract local temporal dependencies. To overcome the limited temporal modeling capability of 1D-CNN, we replace it with the proposed Hybrid Encoder, as shown in Fig. 2.

3.2.2 Positional Encoding module: Learnable Positional Encoding

For PE module, we propose a learnable PE to replace the original one. After initializing positional vectors, we employ a Multi-Layer Perceptron (MLP) to learn these vectors, then concatenate by dimension with the output of Temporal Local Extractor (similar to the Hybrid Encoder) to integrate input features into the PE, and reduce the dimensionality using CNN with kernel 1, as demonstrated in Fig. 2.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

3.2.3 Coefficients for Module Adjustment

We utilize two coefficient vectors, W_{te} and W_{pe}, to adjust the contribution of input and positional information, as displayed in Eq. 1.

$$DataEmbedding = W_{te} * InputEmbedding \ + \ W_{pe} * PositionalEncoding \quad (1)$$

3.3 Proposed PE Loss

Prior work has analyzed the duplication of PE vectors for input tokens by measuring the similarity between these vectors of distinct tokens [7]. When each token corresponds to a time step, this analysis suggests that the larger the temporal distance between two time points, the lower the similarity between their PE vectors should be, thereby reducing the likelihood of positional duplication.

Motivated by this assumption, we propose a PE loss (L_{PE}) that refines positional vectors to better distinguish input tokens along the temporal axis. In this formulation, $L(i,k)$, as defined in Eq. 2, denotes the PE loss at time step i for a given temporal distance k. The overall PE loss is then computed as the average over all such time−distance pairs, as shown in Eq. 3.

$$L(i,k) = \Big[u(i,i+1) - u(i,i+k) - |k-1|.\epsilon\Big]^2 \quad (2)$$

$$L_{PE} = \frac{1}{(L-2)^2} \sum_{i=0}^{L-2} \Big[\sum_{k=2}^{L-i-1} L(i,k) + \sum_{k=-i}^{-2} L(i,k) \Big] \quad (3)$$

where $u(i,j)$ represents the Cosine Similarity between $PE[i]$ and $PE[j]$ (all similarities are scaled to the range $[0,1]$ by adding 1, and then dividing by 2), i and k are running variables for the current time point and the temporal distance to i, respectively. In addition, we use $\epsilon = \frac{1}{len(input)}$ to enforce that the similarity values decrease linearly as the temporal distance increases.

4 Experimental Result

4.1 Benchmark Datasets

We evaluate TSAD on widely used benchmarks [5, 6] spanning six multivariate time-series categories. ECG contains six electrocardiogram subsets (ECG-A–ECG-F). 2D-Gesture captures 2D hand-gesture trajectories from videos. PSM is a 25-channel server-monitoring dataset from eBay, and SMD is a 38-dimensional dataset synthesized from continuous resource-usage traces across 28 machines over four weeks. MSL and SMAP are NASA spacecraft telemetry benchmarks derived from monitored incident periods.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

4.2 Hyperparameters Setup

In all experiments, we tune dataset-specific hyperparameters to maximize validation performance, including the initial learning rate, learning-rate scheduler, training epochs, and the latent dimensionality and TCN channel widths, as detailed in Github. For all benchmarks, we fix the window size to 127, batch size to 64, and use the Adam optimizer.

4.3 Detailed Performance Compared to Baseline

We do experiments on four high-dimensional datasets, then collect the results and summarize them in Table 1, compared to the baseline.

Table 1: Comparison with baseline on Precision, Recall and F_1-score metrics

No.	Dataset	Anomaly Transformer			Our proposed method		
		Precision	Recall	F_1	Precision	Recall	F_1
1	PSM	96.91	98.90	97.89	99.01	98.29	**98.65**
2	SMD	89.40	95.45	**92.33**	**89.74**	87.84	88.78
3	MSL	92.09	95.15	93.59	94.72	93.73	**94.22**
4	SMAP	94.13	99.40	96.69	95.53	98.31	**96.90**

4.4 Comparison of performance with state-of-the-art methods

To demonstrate the robustness, we compare our method with 11 SOTA methods [6], surpassing all of them on several datasets, as shown in Table 2.

Table 2: Comparison with SOTA across five benchmark datasets by $F_1(\%)$

Methods	**ECG**	**2DGesture**	**PSM**	**SMD**	**MSL**
MAD-GAN	48.81	42.47	36.17	91.50	91.69
DAGMM	82.35	38.91	80.08	94.91	84.82
MSCRED	85.37	60.17	84.64	84.14	93.63
CAE-M	62.32	97.94	72.04	93.67	87.33
OmniAnomaly	78.21	41.11	80.83	94.01	87.65
TranAD	33.33	90.34	73.86	96.05	94.94
GDN	77.68	91.68	72.92	83.42	**95.91**
Anomaly Transformer	94.47	98.47	97.89	92.33	93.59
MTAD-GAT	79.19	88.87	82.06	86.83	87.68
USAD	62.33	93.36	73.86	94.95	88.22
CL-TAD	97.00	99.93	95.74	**99.62**	95.38
Ours	**99.515**	**100.00**	**98.65**	88.78	94.22

To assess generality, we integrate our learnable PE module and PE loss into TranAD [8], a strong TSAD baseline, and evaluate F_1 on SMAP and NAB

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

before and after the modification. As shown in Table 3, the augmented TranAD achieves higher performance, reaching $F_1 = 0.9061$ on SMAP and $F_1 = 0.9411$ on NAB, which supports the robustness of our approach.

Table 3: F_1 score comparison between original TranAD and loss-applied TranAD

Datasets	**Original TranAD**	**Loss-applied TranAD**
SMAP	0.8915	**0.9061**
NAB	0.9364	**0.9411**

5 Conclusion

In conclusion, we propose a PE loss with learnable PE module to decrease the positional duplication among input tokens. Empirical experiments show a superior performance on averaged ECG, 2D-Gesture, PSM, MSL, and SMAP, with F_1 equals 99.515%, 100%, 98.65%, 94.22%, and 96.90% respectively, whereas the performance on SMD is relatively poor, with an F_1 score of only 88.78%.

References

[1] Xue Yang, Xuejun Qi, and Xiaobo Zhou. Deep learning technologies for time series anomaly detection in healthcare: A review. *Ieee Access*, 11:117788–117799, 2023.

[2] Stéphane Crépey, Noureddine Lehdili, Nisrine Madhar, and Maud Thomas. Anomaly detection in financial time series by principal component analysis and neural networks. *Algorithms*, 15(10):385, 2022.

[3] Zahra Zamanzadeh Darban, Geoffrey I Webb, Shirui Pan, Charu Aggarwal, and Mahsa Salehi. Deep learning for time series anomaly detection: A survey. *ACM Computing Surveys*, 57(1):1–42, 2024.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

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

[6] Huynh Cong Viet Ngu and Keon Myung Lee. Cl-tad: A contrastive-learning-based method for time series anomaly detection. *Applied Sciences*, 13(21):11938, 2023.

[7] Navid Mohammadi Foumani, Chang Wei Tan, Geoffrey I Webb, and Mahsa Salehi. Improving position encoding of transformers for multivariate time series classification. *Data mining and knowledge discovery*, 38(1):22–48, 2024.

[8] 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.