RESTAD: Reconstruction and Similarity Based Transformer for Time Series Anomaly Detection

Ghorbani, Ramin; Reinders, Marcel J.T.; Tax, David M. J.

Citation (APA)
Ghorbani, R., Reinders, M. J. T., & Tax, D. M. J. (2024). RESTAD: Reconstruction and Similarity Based Transformer for Time Series Anomaly Detection. In *Proceedings of the 2024 IEEE 34th International Workshop on Machine Learning for Signal Processing (MLSP)* IEEE. https://doi.org/10.1109/MLSP58920.2024.10734755

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

RESTAD: RECONSTRUCTION AND SIMILARITY BASED TRANSFORMER FOR TIME SERIES ANOMALY DETECTION

Ramin Ghorbani, Marcel J.T. Reinders, David M.J. Tax

Delft University of Technology, Delft, Netherlands

ABSTRACT

Anomaly detection in time series data is crucial across various domains. The scarcity of labeled data for such tasks has increased the attention towards unsupervised learning methods. These approaches, often relying solely on reconstruction error, typically fail to detect subtle anomalies in complex datasets. To address this, we introduce *RESTAD*, an adaptation of the Transformer model by incorporating a layer of Radial Basis Function (RBF) neurons within its architecture. This layer fits a non-parametric density in the latent representation, such that a high RBF output indicates similarity with predominantly normal training data. RESTAD integrates the RBF similarity scores with the reconstruction errors to increase sensitivity to anomalies. Our empirical evaluations demonstrate that RESTAD outperforms various established baselines across multiple benchmark datasets.

Index Terms— Time Series, Anomaly Detection, Radial Basis Function (RBF) kernel, Transformer

1. INTRODUCTION

Anomalies in time series data, i.e., unexpected patterns or deviations from normal behavior, can signify critical issues across various domains, from financial fraud to life-threatening health conditions. Hence, accurate anomaly detection is important. Given the rarity of anomalies and, thus, the lack of sufficient labeled data, fully supervised methods are less suited. Consequently, unsupervised learning methods have gained increasing attention [1]. These methods do not explicitly require labeled anomaly examples, making them ideal for the detection of unknown or unexpected anomalies [2].

Various classic unsupervised techniques like distance-based One-Class SVM (OC-SVM) [3] or density-based Local Outlier Factor (LOF) [4], have been widely used. However, they struggle with the temporal dependencies, high dimensionality, and complex generalization demands of time series data [5]. Recent developments in deep learning offer promising solutions for handling these challenges [6]. Architectures like Transformers and LSTMs excel at capturing temporal

patterns and automatically learning hierarchical and non-linear features from time series data [7, 8].

Building on these advancements, several effective anomaly detection methods have been developed, largely focusing on the reconstruction error as a primary anomaly criterion [9, 10, 11]. These methods typically assess the deviations between a given input and its reconstruction to identify anomalies. The underlying assumption is that typical data will have lower reconstruction errors, whereas anomalous data will exhibit higher errors due to the unfamiliarity of the model with these patterns [10, 12].

A major issue of using the reconstruction error for anomaly detection is over-generalization [13]. Models fitted to capture the predominant patterns in the training data, generalize these patterns to include subtle variations as well. Therefore subtle anomalies can also be reconstructed well by these models. As a result, these anomalies are less distinguishable from typical patterns, reducing the model's detection sensitivity [14]. This effect is depicted in Figure 1.a, where the original signal includes a subtle anomaly at time point t_0 and a significant anomaly at t_1. The reconstructed signal is a slightly smoothed version of the original signal, and by using the reconstruction error alone, the subtle anomaly is missed as the reconstruction error remains below the detection threshold, as shown in Figure 1.b.

Efforts have been made to improve unsupervised anomaly detection by adding other types of scores to the conventional reconstruction error-based anomaly scores. For instance, AnomalyTrans [15] utilizes the concept of association discrepancy, which considers the similarity of a time point with its adjacent time points. It then reweights the reconstruction error accordingly to formulate the final composite anomaly score. However, in this method a normalization is performed, which can exaggerate the discrepancy scores for normal time points when no anomalies are present, potentially leading to false positives. This can misleadingly highlight normal data points as anomalies. Although this approach is effective for identifying clear outliers, it can inadvertently misrepresent subtle normal fluctuations as anomalies.

To overcome the challenges of scoring based on reconstruction error and the limitations of the association discrepancy method, we propose combining the reconstruction error with
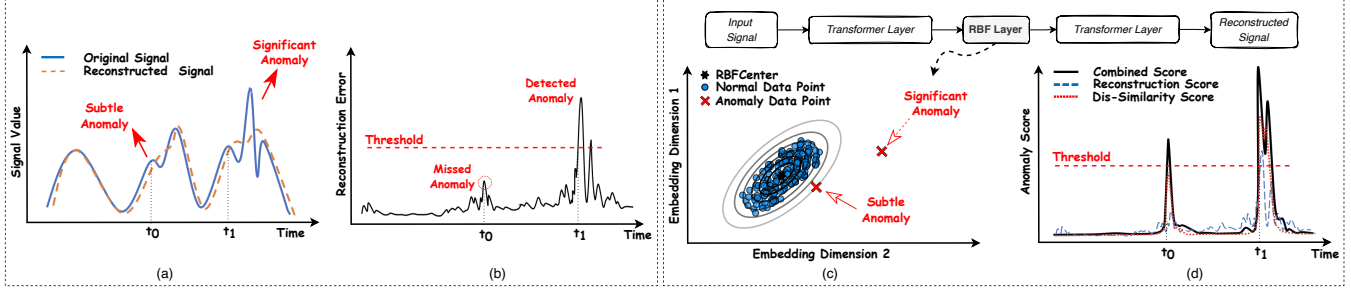

Fig. 1: *Comparison of traditional reconstruction and RBF-enhanced anomaly detection: a) Original signal with subtle and significant anomalies compared to its reconstruction. b) Reconstruction errors for the signals in (a), highlighting challenges in detecting subtle anomalies. c) Visualization of a model integrated with an RBF, shown via a 2D scatter plot that includes typical data, subtle and significant anomalies, and the RBF center with its influence radius, showing the RBF's ability to differentiate typical points from anomalies. d) Enhanced anomaly score using the RBF, which shows improved detection of subtle anomalies.*

a specialized non-linear transformation like the Radial Basis Function (RBF) kernel [16]. The RBF kernel generates a similarity score that measures how close a data point is to a reference point or center, making it highly effective for anomaly detection. Anomalies, data points that deviate (are far away) from typical patterns, yield lower similarity scores with the RBF kernel, thus directly measuring how anomalous a point is. This score can effectively complement the reconstruction error and improve the sensitivity to subtle anomalies that might be overlooked by the reconstruction error. The effectiveness of combining RBF scores with reconstruction error is illustrated in Figure 1.c, where an RBF kernel is applied to typical data in the latent representation of a Transformer. By combining reconstruction error with RBF similarity scores, we create a comprehensive composite anomaly score that not only captures deviations from expected patterns but also ensures that subtle anomalies are still flagged. This composite anomaly score is shown in Figure 1.d, where the anomaly scores for both anomaly types are now above the detection threshold.

This paper presents an adaptation of the Transformer model, chosen for its ability to capture temporal dependencies in sequential data. By integrating the RBF neurons into the Transformer architecture, we develop a model that synergistically utilizes both similarity scores and reconstruction error to compute a distinctive anomaly score. Through an extensive evaluation, we show that this new REconstruction and Similarity based Transformer for time series Anomaly Detection, RESTAD, outperforms existing baselines across a range of benchmark datasets.

2. METHODOLOGY

Assume that the observed time series dataset consists of N sequences with length T. Each sequence in this dataset is denoted by $\boldsymbol{X}_i = \{\boldsymbol{x}_{i,t}\}_{t=1}^{T}$ where $\boldsymbol{x}_{i,t}$ represents the observed time point for i-th sequence at time t, having d dimensions, i.e., $\boldsymbol{x}_{i,t} \in \mathbb{R}^d$. Our task is to determine if a given $\boldsymbol{x}_{i,t}$ shows any anomalous behavior or not.

2.1. RESTAD Framework

In our study, we incorporate the anomaly detection mechanism into the vanilla Transformer [7] through a specific layer of RBF neurons, see Figure 1.c. This RBF layer operates on the latent representations from the preceding layer, denoted by $\mathcal{H}_i = \{\boldsymbol{h}_{i,t}\}_{t=1}^{T}$, where $\boldsymbol{h}_{i,t} \in \mathbb{R}^{d_h}$. This layer computes the similarity of each data point $\boldsymbol{h}_{i,t}$ to a set of learnable reference points (centers), denoted by $\mathcal{C} = \{\boldsymbol{c}_m\}_{m=1}^{M}$, where $\boldsymbol{c}_m \in \mathbb{R}^{d_h}$. This computation results in the RBF output, $\mathcal{Z}_i = \{\boldsymbol{z}_{i,t}\}_{t=1}^{T}$, where $\boldsymbol{z}_{i,t} \in \mathbb{R}^M$, which then serves as the input to subsequent layer of the model. Specifically, the RBF similarity output for each data point relative to each center is defined by:

$$z_{i,t}^{m}(\boldsymbol{h}_{i,t}, \boldsymbol{c}_m) = \exp\left(-\frac{1}{2}e^{\gamma}\|\boldsymbol{h}_{i,t} - \boldsymbol{c}_m\|^2\right) \qquad (1)$$

Here, the parameter γ controls the width of the RBF, influencing how it considers data points at varying distances from the center. This parameter is initialized and adjusted during training. Using the exponential of γ ensures the positivity of the scale parameter, simplifying the optimization process without enforcing a positivity constraint.

Anomaly Score: RESTAD is trained by minimizing the Mean Squared Error (MSE) to achieve accurate reconstruction. For anomaly detection, a composite anomaly score, $RESTAD_{score}$, is introduced by combining the normalized RBF similarity scores and reconstruction errors. The normalization is based on MinMax to ensure comparability. The RBF similarity score measures how closely $\boldsymbol{x}_{i,t}$ aligns with the learned centers. A higher similarity suggests normal behavior, whereas a lower similarity (or greater distance to the RBF centers) signals anomalies. This score is derived from averaging the RBF output $\boldsymbol{z}_{i,t}$ across all centers. The reconstruction error is the squared difference between the actual data $\boldsymbol{x}_{i,t}$ and its reconstruction $\hat{\boldsymbol{x}}_{i,t}$. The $RESTAD_{score}$ is formulated as:

$$RESTAD_{score}(\boldsymbol{x}_{i,t}) = \epsilon_r \times \epsilon_s \qquad (2)$$
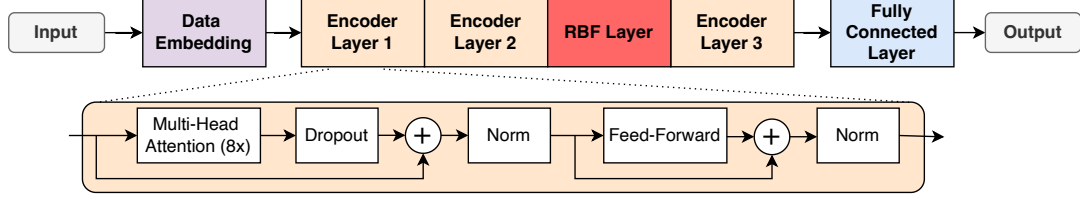
Fig. 2: *Overview of the proposed RESTAD model. Here, the RBF layer is added after the second encoder layer.*

where $\epsilon_r = ||\boldsymbol{x}_{i,t} - \hat{\boldsymbol{x}}_{i,t}||^2$ represents the reconstruction error, and $\epsilon_s = \left(1 - \frac{1}{M}\sum_{m=1}^{M} z_{i,t}^m\right)$ measures dissimilarity. This combination highlights subtle anomalies characterized by both low reconstruction errors and RBF scores, as well as significant anomalies with high reconstruction errors or low RBF scores.

Initialization of RBF Layer Parameters: Proper initialization of the RBF parameters, including the centers $\boldsymbol{c}$ and scale parameter γ, is crucial for our methodology. We explore two initialization strategies: *Random* and *K-means*, to assess their impact on model performance. For *Random* initialization, parameters $\boldsymbol{c}$ and γ are drawn from a normal distribution with zero mean and unit standard deviation. Although it is simple, it may lead to slower convergence, risk of local minima, and may not effectively represent the data distribution initially, possibly resulting in instability. In contrast, *K-means* initialization uses the inherent data structure for a more representative starting point. In this approach, initially, a base model (without the integrated RBF layer) is trained to minimize the MSE of reconstruction:

$$\mathcal{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left\|\boldsymbol{X}_i - \hat{\boldsymbol{X}}_i\right\|_F^2 \qquad (3)$$

After achieving satisfactory reconstruction accuracy from the base model, the latent representation is extracted from the specific layer where the RBF layer is intended to subsequently be integrated. This representation is then used to initialize $\boldsymbol{c}$ via the K-means clustering algorithm. The scale parameter γ is initialized using $\tilde{\sigma}^2$, the mean squared distance from each data point to its nearest cluster center:

$$\tilde{\sigma}^2 = \frac{1}{NT}\sum_{i=1}^{N}\sum_{t=1}^{T}\min_{m}\|\boldsymbol{h}_{i,t} - \boldsymbol{c}_m\|^2, \quad \forall m \in [1, M] \quad (4)$$

Here, $\boldsymbol{h}_{i,t}$ denotes the latent representation vector of the i-th sample at the t-th time step, and $\boldsymbol{c}_m$ is the m-th cluster center obtained from the K-means algorithm. This value, $\tilde{\sigma}^2$, is used to initialize γ as $\gamma = \frac{1}{\tilde{\sigma}^2}$, ensuring that the RBF function has a spread informed by the average dispersion of the data points around their respective centers.

3. EXPERIMENTAL SETUP

3.1. Datasets and Preprocessing

We use three public widely used benchmark datasets for our experiments: 1) Server Machine Dataset (SMD) [10], 2) Mars Science Laboratory (MSL) Rover [8], and 3) Pooled Server Metrics (PSM) [17]. Further information on each dataset is available in our code repository[1].

Data preprocessing involves normalizing each feature to zero mean and unit variance across the time dimension. Subsequently, the normalized signal is segmented into non-overlapped sliding windows with a fixed length of 100 data points, a common setting based on previous related works [15, 9].

3.2. Implementation

RESTAD Model: The RESTAD model is an adaptation of a vanilla Transformer, incorporating an RBF kernel layer as detailed in Figure 2. It includes a DataEmbedding module that combines both token and positional embeddings, followed by an encoder with three layers. Each layer includes a multi-head self-attention mechanism and feed-forward networks. The model has a latent dimension of 32, an intermediate feed-forward network layer with a dimension of 128, and 8 attention heads. The RBF layer is placed after the second encoder layer (other placements are also possible, see section 4.1). Optimization is performed using the ADAM optimizer, and hyperparameters are determined through systematic search to optimize reconstruction task performance. Additional hyperparameter details are available in our code repository[1].

Evaluation: Anomaly scores (Eq. 2) exceeding a threshold δ are identified as anomalies. Performance is evaluated using the F1-score for threshold-dependent evaluation. Here, we follow [15] by setting δ to label a predefined proportion of data points as anomalies (0.5% for SMD, 1% for others). For threshold-independent analysis, we use AUC-ROC, AUC-PR, VUS-ROC, and VUS-PR metrics [18]. We exclude the point-adjustment method [19] due to its overestimation [20]. Our model is compared against baselines and state-of-the-arts models: LSTM [8], vanilla Transformer [15], USAD [12], PatchAD [11], AnomalyTrans [15], and DCdetector [9].

---

[1]https://github.com/Raminghorbanii/RESTAD

Table 1: *Performance metrics of baselines and RESTAD on test sets.* Initialization methods are denoted as (R) for *Random* and (K) for *K-means*. For all measures, a higher value indicates better anomaly detection performance.

Dataset	SMD					MSL					PSM				
Models	F1-Score	AUC-ROC	AUC-PR	VUS-ROC	VUS-PR	F1-Score	AUC-ROC	AUC-PR	VUS-ROC	VUS-PR	F1-Score	AUC-ROC	AUC-PR	VUS-ROC	VUS-PR
LSTM	0.12	0.74	0.17	0.79	0.20	0.06	0.56	0.14	0.63	0.19	0.11	0.73	0.50	0.72	0.51
USAD	0.13	0.63	0.11	0.72	0.14	0.06	0.53	0.14	0.59	0.18	0.07	0.60	0.41	0.61	0.43
PatchAD	0.01	0.50	0.04	0.61	0.08	0.03	0.50	0.10	0.57	0.15	0.02	0.50	0.28	0.55	0.33
Transformer	0.11	0.75	0.19	0.80	0.22	0.06	0.56	0.14	0.63	0.19	0.13	0.71	0.49	0.70	0.50
AnomalyTrans	0.03	0.49	0.04	0.50	0.07	0.02	0.49	0.10	0.52	0.14	0.02	0.51	0.30	0.53	0.34
DCDetector	0.01	0.50	0.04	0.51	0.08	0.02	0.50	0.11	0.58	0.15	0.02	0.50	0.28	0.52	0.32
RESTAD (R)	**0.23**	**0.78**	**0.23**	**0.82**	**0.24**	**0.07**	**0.68**	**0.18**	**0.72**	**0.23**	**0.15**	**0.79**	**0.59**	**0.76**	**0.57**
RESTAD (K)	**0.20**	**0.79**	**0.24**	**0.83**	**0.25**	**0.07**	**0.66**	**0.18**	**0.71**	**0.23**	**0.14**	**0.79**	**0.57**	**0.76**	**0.56**

4. RESULTS

Our empirical results, as detailed in Table 1, highlight the effectiveness of the RESTAD for anomaly detection. RESTAD outperforms all baseline models across the benchmark datasets and evaluation metrics, regardless of the RBF initialization strategy. While there are slight performance differences between initialization methods, these variations are not significant enough to establish the superiority of one method over another.

To visually show detection differences, Figure 3 displays anomaly scores for a short segment of the SMD dataset. PatchAD, DCdetector, and AnomalyTrans models reveal many false detections, with DCdetector showing a pattern of repeated false positives and PatchAD resembling random scoring. LSTM, USAD, and Transformer models miss some anomalies or detect them weakly; for example, the first anomaly area is undetected by USAD, and only weakly detected by LSTM and Transformer. In contrast, the RESTAD model demonstrates robust detection, effectively identifying all anomaly sections.
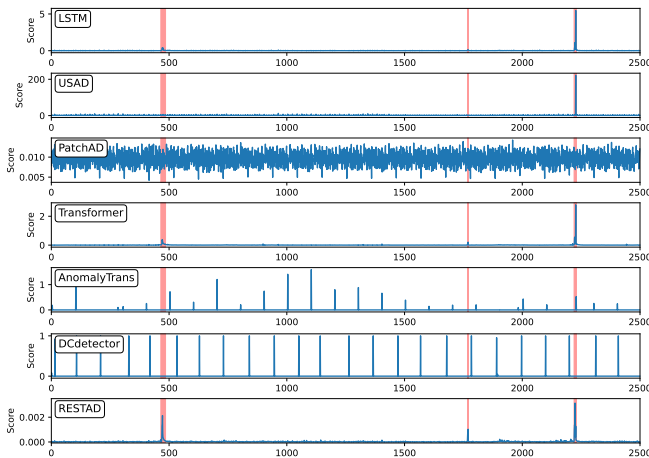

Fig. 3: *Anomaly scores of different models for a segment of SMD dataset. The highlighted regions in red indicate the true anomaly periods (labeled by an expert).*

4.1. Ablation Analysis

The ablation experiments are based on the RBF layer with random initialization. This decision is based on our findings that random initialization is as effective as the K-means strategy (see Table 1), while offering greater simplicity and computational efficiency.

Anomaly Score Criterion: Table 2 highlights the impact of integrating the RBF score into anomaly detection. Multiplying the RBF layer's dissimilarity score (ϵ_s) with the reconstruction error (ϵ_r) to form the composite anomaly score ($\epsilon_s \times \epsilon_r$) is found to be the most effective, consistently enhancing detection across all benchmarks and metrics. Adding the RBF layer to the vanilla Transformer with only reconstruction error ϵ_r as the anomaly score offers marginal improvements on some datasets. In contrast, using only the dissimilarity score (ϵ_s) or adding it directly to the reconstruction error ($\epsilon_s + \epsilon_r$) shows no significant benefits.

Figure 4 visually illustrates the superiority of our composite anomaly score over the traditional reconstruction score (ϵ_r) by showing subsets from all three datasets and the corresponding anomaly scores. Our anomaly score effectively identifies anomalies that are overlooked by the model relying solely on reconstruction error, with detections notably stronger and typically exceeding the threshold. Note that the thresholds depicted in the figures are the best optimized ones based on the entire dataset. Altering this threshold for the subset of data presented in the figures could diminish the overall performance and is therefore not possible.

RBF Layer Placement: We explored the flexibility of RBF layer placement within the vanilla Transformer by integrating it after each of the three encoder layers. Figure 5 demonstrates that performance remains robust across all datasets, irrespective of the RBF layer's location. Note that placing the RBF layer after the second encoder layer results in marginally better performance across all datasets. This slight advantage influenced our decision to position the RBF layer after the second layer in the final model architecture (see Figure 2).

Table 2: *Effect of integrating RBF layer and the choice of anomaly score. For all measures, a higher value indicates better anomaly detection performance.*

Architecture	Anomaly Criterion	SMD					MSL					PSM				
		F1-Score	AUC-ROC	AUC-PR	VUS-ROC	VUS-PR	F1-Score	AUC-ROC	AUC-PR	VUS-ROC	VUS-PR	F1-Score	AUC-ROC	AUC-PR	VUS-ROC	VUS-PR
Transformer	ϵ_r	0.11	0.75	0.19	0.80	0.22	0.06	0.56	0.14	0.63	0.19	0.13	0.71	0.49	0.70	0.50
RESTAD	ϵ_r	0.11	0.77	0.18	0.81	0.21	0.07	0.63	0.16	0.69	0.21	0.13	0.75	0.56	0.74	0.55
RESTAD	ϵ_s	0.01	0.44	0.03	0.52	0.07	0.01	0.43	0.08	0.48	0.12	0.01	0.32	0.20	0.37	0.25
RESTAD	$\epsilon_r + \epsilon_s$	0.04	0.57	0.06	0.60	0.10	0.07	0.61	0.16	0.65	0.20	0.01	0.68	0.49	0.59	0.45
RESTAD	$\epsilon_r \times \epsilon_s$	**0.23**	**0.78**	**0.23**	**0.82**	**0.24**	**0.07**	**0.68**	**0.18**	**0.72**	**0.23**	**0.15**	**0.79**	**0.59**	**0.76**	**0.57**

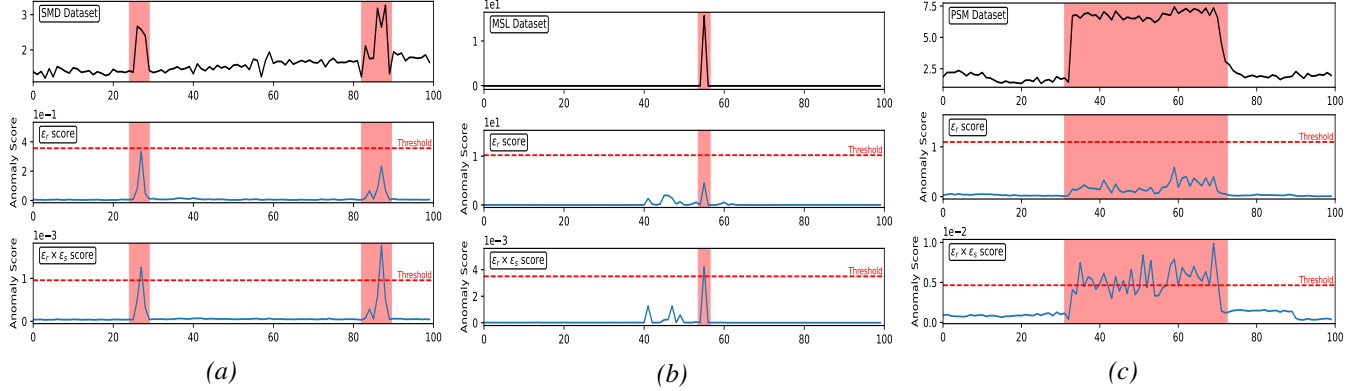
(a) *(b)* *(c)*

Fig. 4: *Effect of our composite anomaly score ($\epsilon_r \times \epsilon_s$) compared to reconstruction error (ϵ_r) across segments of all datasets. The highlighted regions in red indicate the true anomaly periods (labeled by an expert).*
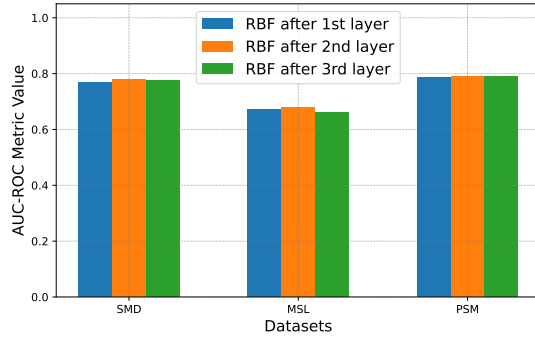

Fig. 5: *RESTAD Performance with varying RBF layer placements.*

Fig. 6: *RESTAD mean performance across varying numbers of RBF centers. Shaded areas indicate ± standard deviation, illustrating variability across multiple runs.*

Number of RBF Centers: Figure 6 represents the impact of the number of centers, ranging from 8 to 512, in the RBF layer of *RESTAD*. Results indicate that the optimal number of RBF centers is data-dependent. Additionally, beyond a certain threshold, increasing the number of centers does not enhance performance and may even reduce it.

5. DISCUSSION AND CONCLUSION

We introduced RESTAD, an adaptation of Transformers for unsupervised anomaly detection that improves on the limitations of using only reconstruction error as the anomaly score. By integrating an RBF layer into the Transformer, we combined RBF similarity scores with reconstruction error, enhancing 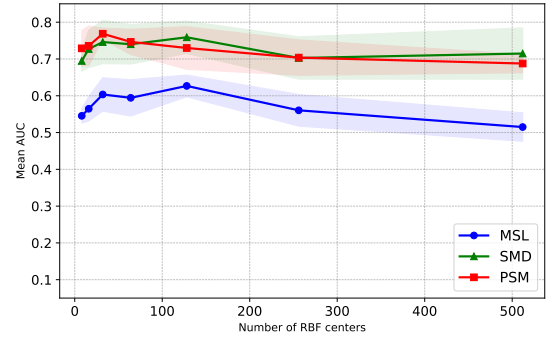the sensitivity to subtle anomalies. RESTAD consistently outperforms established baselines across various datasets and evaluation metrics.

Our findings reveal that RESTAD's performance is relatively invariant to RBF layer initialization methods, indicating robustness against initialization variability. The significant performance gains are primarily due to the multiplicative fusion of RBF similarity scores with reconstruction error, markedly improving anomaly detection capabilities. The RBF layer's placement within the architecture did not significantly affect performance, demonstrating architectural flexibility in integrating the RBF layer. However, the optimal number of RBF centers is data-dependent. These findings motivate future studies for the exploration of integrating RBF layers into other deep learning architectures for anomaly detection tasks.

6. REFERENCES

[1] Varun Chandola, Arindam Banerjee, and Vipin Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[2] Ramin Ghorbani, Marcel JT Reinders, and David MJ Tax, "Personalized anomaly detection in ppg data using representation learning and biometric identification," *Biomedical Signal Processing and Control*, vol. 94, pp. 106216, 2024.

[3] Bernhard Schölkopf, Robert C Williamson, Alex Smola, John Shawe-Taylor, and John Platt, "Support vector method for novelty detection," *Advances in neural information processing systems*, vol. 12, 1999.

[4] 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*, 2000, pp. 93–104.

[5] Nesryne Mejri, Laura Lopez-Fuentes, Kankana Roy, Pavel Chernakov, Enjie Ghorbel, and Djamila Aouada, "Unsupervised anomaly detection in time-series: An extensive evaluation and analysis of state-of-the-art methods," *arXiv preprint arXiv:2212.03637*, 2022.

[6] Kukjin Choi, Jihun Yi, Changhwa Park, and Sungroh Yoon, "Deep learning for anomaly detection in time-series data: review, analysis, and guidelines," *IEEE Access*, vol. 9, pp. 120043–120065, 2021.

[7] 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*, 2017.

[8] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, and Tom Soderstrom, "Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 387–395.

[9] 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*, 2023, pp. 3033–3045.

[10] 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*, 2019, pp. 2828–2837.

[11] Zhijie Zhong, Zhiwen Yu, Yiyuan Yang, Weizheng Wang, and Kaixiang Yang, "Patchad: Patch-based mlp-mixer for time series anomaly detection," *arXiv preprint arXiv:2401.09793*, 2024.

[12] Julien Audibert, Pietro Michiardi, Frédéric Guyard, Sébastien Marti, and Maria A Zuluaga, "Usad: Unsupervised anomaly detection on multivariate time series," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 3395–3404.

[13] Tianzi Zhao, Liang Jin, Xiaofeng Zhou, Shuai Li, Shurui Liu, and Jiang Zhu, "Unsupervised anomaly detection approach based on adversarial memory autoencoders for multivariate time series.," *Computers, Materials & Continua*, vol. 76, no. 1, 2023.

[14] Haoyi Zhong, Yongjiang Zhao, and Chang Gyoon Lim, "Abnormal state detection using memory-augmented autoencoder technique in frequency-time domain," *KSII Transactions on Internet and Information Systems (TIIS)*, vol. 18, no. 2, pp. 348–369, 2024.

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

[16] Mark JL Orr et al., "Introduction to radial basis function networks," 1996.

[17] Ahmed Abdulaal, Zhuanghua Liu, and Tomer Lancewicki, "Practical approach to asynchronous multivariate time series anomaly detection and localization," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021.

[18] 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," *Proceedings of the VLDB Endowment*, vol. 15, no. 11, pp. 2774–2787, 2022.

[19] 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*, 2018, pp. 187–196.

[20] Siwon Kim, Kukjin Choi, Hyun-Soo Choi, Byunghan Lee, and Sungroh Yoon, "Towards a rigorous evaluation of time-series anomaly detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, vol. 36, pp. 7194–7201.