

Z-ROCKET: A Lightweight Method for Whole-Time-Series Anomaly Detection Using Random Convolutional Kernels

Matthias Rüb*, Ahmed Hassan Mohamed Farrag*, Christoph Lipps* and Hans D. Schotten*[†]

*German Research Center for Artificial Intelligence (DFKI), Kaiserslautern, Germany

E-mail: matthias.rueb@dfki.de

[†]RPTU University Kaiserslautern-Landau, Kaiserslautern, Germany

E-mail: schotten@rptu.uni-kl.de

Abstract—Efficient time series anomaly detection (AD) is becoming increasingly important for emerging wireless networks, where large-scale monitoring and autonomous operation require accurate yet computationally lightweight solutions. While random convolutional kernel methods have shown strong performance in time series analysis, existing AD approaches typically rely on additional classifiers, increasing computational complexity. In this work, we introduce Z-ROCKET, a semi-supervised whole time series AD algorithm that combines the efficient random convolutional kernel transformation of MiniRocket with an efficient top-k z-score-based thresholding approach. Z-ROCKET enables AD directly in the transformed feature space without requiring additional classifiers. Extensive experiments on 93 heterogeneous UCR datasets, used in existing literature, demonstrate that Z-ROCKET consistently outperforms fast baseline methods, including k-nearest neighbors (KNN), Isolation Forest (iForest), and Local Outlier Factor (LOF), achieving an AUROC of 0.86 at a comparable computational cost. Furthermore, Z-ROCKET approaches the performance of the random convolutional kernel-based ROCKAD method (AUROC of 0.90) while reducing training time by an average factor of 50×, confirming the feasibility of the fast MiniRocket transformation for AD. These results demonstrate that random convolutional kernel feature extraction enables effective AD using simple statistical thresholding, without complex downstream models.

Index Terms—Anomaly detection, Time series, Random convolutional kernels

I. INTRODUCTION

Time series Anomaly Detection (AD) is becoming increasingly important in emerging wireless networks, particularly as agentic systems are expected to take over autonomous network orchestration. In such environments, large volumes of heterogeneous data are continuously generated across multiple layers and components, requiring robust and efficient monitoring mechanisms [1]. Detecting anomalous behavior in these data streams is critical for reliable, secure, and sustainable network operation.

At the same time, current communication networks face strong competition for computational resources, alongside increasing the concerns regarding energy efficiency and sustainability [2]. This pushes the demand for lightweight AD methods that provide sufficient accuracy while maintaining

low computational overhead. Efficient solutions are especially valuable in resource-constrained or large-scale deployments, where traditional, computationally expensive approaches may not be feasible.

Beyond networking, efficient time series AD has a wide range of additional applications. For instance, in healthcare environments, such methods can be used to ensure the integrity of patient data or support early disease diagnosis through continuous monitoring of physiological signals.

In recent years, random convolutional kernel methods, particularly the ROCKET family of algorithms [3], [4], have shown strong potential time series classification tasks. These approaches transform input time series into highly informative feature representations using large sets of randomly generated convolutional kernels, followed by simple downstream classifiers. Several recent works have attempted to adapt these feature transformations to AD, typically by combining them with additional classifiers such as k-nearest neighbors (KNN) [5], one-class Support Vector Machine (SVM) [6], and iForest [7].

Following the paradigm of the original ROCKET family, we argue that the expressive feature space generated by the lightweight MiniRocket is sufficiently rich to enable effective AD without the need for complex downstream models. Instead, we show that a simple thresholding mechanism in the transformed feature space is sufficient to achieve strong performance.

In this work, we introduce Z-ROCKET, a semi-supervised whole time series AD algorithm. Our method leverages the fast random convolutional kernel transformation of MiniRocket [4] and combines it with an efficient top-k z-score computation. Based on feature statistics derived from normal training data, Z-ROCKET defines calibrated thresholds using percentile-based criteria, enabling fast and accurate AD without additional classifiers. Z-ROCKET consistently outperforms fast baselines such as KNN, iForest, and Local Outlier Factor (LOF) at a similar computational cost, achieving an AUROC of 0.86. At the same time, it approaches the performance of the random convolutional kernel-based method ROCKAD [5]

(AUROC of 0.90) while being, on average, 50× faster to train, primarily due to the faster MiniRocket transformation.

The main contributions in this work are:

- Introduction of Z-ROCKET, a semi-supervised whole time series AD method based on MiniRocket features and z-score thresholds.
- We demonstrate that the feature space induced by random convolutional kernels is sufficiently informative to enable a z-score-based threshold in this feature space for AD, without an additional classifier.

The remainder of this work is structured as follows: Section II positions the proposed method within the existing literature. Section III presents Z-ROCKET in detail. Section IV provides an empirical evaluation and comparison with existing methods. Section V concludes this work and provides an outlook on future research directions.

II. RELATED WORK

Recent developments driven by artificial intelligence, such as automated agent systems for network orchestration in 6G communication networks and other applications in healthcare foster the need for robust AD in time series in various domains [8], [9]. Frequently discussed approaches include Matrix Profile (MP) [10] and Catch22 features [11]. An extensive overview of current methods is given in [12]. Meanwhile, time series classification with supervised learning paradigms has shown significant progress with the efficient feature extraction via random convolutional kernels, namely the ROCKET family of algorithms [3], [4], [13]. The idea of applying random convolutional kernels to whole time series AD has already been discussed in recent literature, for instance by *Theissler, Wengert, and Gerschner* [5]. In their approach, ROCKET-based feature extraction is followed by a Yeo-Johnson power transformation to improve normality, notably not using the faster MiniRocket framework, which we address in this work. Finally, KNN is applied, either as a single detector or as an ensemble, to derive anomaly scores based on Euclidean distances from normal training data. In contrast, we do not utilize any of their architecture components, but follow only the basic idea of applying ROCKET-based feature extraction for AD. Particularly, our proposed method omits the training of a dedicated anomaly detector, such as KNN, completely. The only learning step is the deterministic computation of thresholds based on the z-scores of the feature space of a non-anomalous distribution, allowing for one order of magnitude faster training and inference. Additional applications of convolutional kernels to AD are in conjunction with iForest [7] and One-Class Support Vector Machine (SVM) [6]. Those methods, however, do not provide guidelines for whole time series AD and present very limited use cases instead of heterogeneous benchmarks. In addition, both rely on the maximum as a pooling operator in the feature space of random convolutional kernels. In contrast, we utilize only the Proportion of Positive Values (PPV), the single feature of MiniRocket [4].

A major advance of random convolution-based methods is the fact that the feature space is so information-rich that simple aggregate features (such as the maximum) in this feature space are sufficient to outperform many state-of-the-art models in classification problems when combined with a linear classifier [14]. However, this is not used in the literature for AD without the use of an additional learning step such as one-class SVM. To address this gap, we restrict our anomaly decision to a fast-computing z-score threshold. Z-scores applied to the time series itself would only be sensitive to absolute deviations; however, in the feature space of convolutional kernels, such thresholds can detect the absence or introduction of arbitrary patterns.

III. METHODS

Z-ROCKET introduces a semi-supervised whole time series AD method that learns from normal data during training. The method uses MiniRocket’s [4] random convolutional kernels to extract features, then detects anomalies based on z-score deviations in this feature space.

A. Prerequisites

Whole time series AD assigns a single anomaly score to a time series. For consistency with [5], an existing application of ROCKET-based features in whole time series AD, we denote the anomaly score as α and the time series as x_T .

The first part in the proposed approach is the feature extraction via MiniRocket. The idea is to transform an input time series with a sliding dot product with numerous random kernels. A detailed description of the creation rules for such kernels within the MiniRocket framework for time series classification is given by *Dempster et al.* in [4]. Then a single aggregate feature is computed across the whole time series for each kernel: the proportion of positive values (PPV). Intuitively, a high PPV indicates that the corresponding random kernel frequently exhibits a strong match with the time series. Rocket-based approaches inherently apply z-normalization to the raw time series. This should be distinguished from the z-score computation used in this work, which is performed in feature space rather than on the raw time series.

The experimental results in Section IV are limited to univariate time series. However, since modern implementations of random convolutional kernel methods such as MiniRocket support multivariate inputs, our proposed approach is directly applicable to multivariate time series. A comprehensive empirical evaluation on multivariate data is part of future work.

B. Rationale behind the proposed approach

Random convolutional kernels have been shown to extract rich features from time series data, achieving exceptional accuracy in time series classification tasks [14], [15].

On the other hand, z-score computation on raw time series serves as a simple baseline for AD, identifying absolute values that deviate significantly from a non-anomalous distribution. This simplistic baseline is agnostic to patterns and frequencies. The goal of our proposed approach is to combine the rich

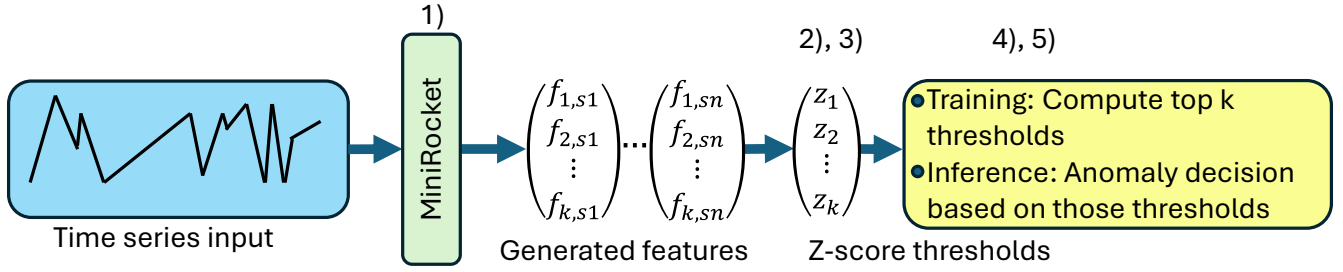


Fig. 1: Z-ROCKET: A semi-supervised approach for whole time series anomaly detection. 1) Transform the n non-anomalous training segments via MiniRocket. 2) Compute global statistics on those transformed series producing n feature vectors. 3) For each kernel, compute the corresponding z-scores of an input segment based on the global intra-feature mean and standard deviation. The latter are computed horizontally across all non-anomalous training segments of the same kernel features. Next, compute the mean of the top k largest absolute values in $\bar{z}$. 4) Define normal thresholds based on percentiles of the mean of the top k z-score absolute values. 5) Compare new inference samples against these z-score thresholds to detect deviations from the normal distribution. While z-scores on raw time series are only sensitive to absolute value deviations, the application in the convolutional kernel feature space can detect complex, arbitrary patterns or their absence.

feature representation of random convolutional kernels with an ultra-fast z-score computation on kernel features instead of raw time series. We assume that for each anomaly, there exists at least one kernel whose activation is notably affected by that specific anomaly. A reduction in kernel activation corresponds to the absence of patterns that are normally present, while an increase in activation reflects the introduction of new, anomalous patterns.

By employing a sufficiently large and diverse set of kernels, any anomaly will significantly alter at least one kernel activation. Experiments in [5] and Section IV suggest that these assumptions are easily fulfilled, and significantly fewer kernels are required than MiniRocket’s default of 10,000 for classification [4].

C. Architecture of Z-ROCKET

The architecture of Z-ROCKET consists of the following steps. A schematic overview is given in Fig. 1.

1) *Feature Extraction (Training)*: Inputs are normal time series segments $x_{T,n}$ from the training set. The original time series is transformed via MiniRocket with k random convolutional kernels. Each kernel produces one feature per segment by computing pooling operations, in MiniRocket namely the PPV. The result is a feature matrix (n_k kernels, n train segments).

2) *Compute global statistics (Training)*: Compute mean μ and standard deviation σ per kernel-feature. Here, the kernel feature is the PPV from MiniRocket. This results in one mean vector and one standard deviation vector across all training samples. A small epsilon ($1e-8$) is added to sigma to avoid division by zero.

3) *(Optional) Feature selection (Training)*: Depending on a desired focus on a certain anomaly type, the highest or lowest mean values in $\bar{\mu}$ can be pruned. High μ values allow detecting the absence of normal patterns as an anomaly. Accordingly, low μ allow detecting newly introduced patterns as anomalies.

4) *Computing Anomaly Scores (Training)*: For each normal training segment, compute a z-score vector as follows:

$$z_i = \frac{(x_i - \mu_i)}{\sigma_i} \quad (1)$$

where x_i denotes the feature value of the current normal segment for kernel index i . Next, we aggregate the z-scores into a single anomaly score α : First, take the absolute value $|z_i|$. This treats upward and downward deviations due to anomalies equally. Second, select the top k largest absolute z-scores. The choice of k is arbitrary, detailed results in Section IV show that values between 3 and 100 yield very similar performance. Third, compute the mean of these top k values. This mean defines the anomaly score α for one particular non-anomalous training segment x_T .

5) *Setting the Threshold (Training)*: Determine what score constitutes an anomaly: The threshold T_z for this anomaly score is set, so that 99 % of normal training segments have scores $\leq$ threshold. Accordingly, 1 % of normal training segments have scores $>$ threshold. Changing this parameter shifts the performance between precision and recall.

IV. EXPERIMENTAL RESULTS

In this section, hyperparameter settings of the proposed method are explored empirically. Second, the method is compared with existing solutions. As a representative of existing random convolutional kernel-based approaches, we utilize ROCKAD by Theissler, Wengert, and Gerschner [5]. Other approaches with random kernels such as RandomAD [16] and Rocket-LRP [17] do not offer a framework for whole time series AD and are therefore not included in the empirical comparison. As the proposed Z-ROCKET is 50 times faster than ROCKAD during training, suitable other ultra-fast methods such as KNN with $n_neighbor=20$, LOF [18], and iForest [19] are included. For direct performance comparison with existing methods, the dataset creation described in [5] is

utilized, which is briefly summarized in the following, using the same notation:

The foundational dataset is the UCR time series classification data set [20]. The UCR anomaly dataset is not suited for whole time series semi-supervised mode, therefore the UCR classification set is utilized. For AD, the class with the most time series is defined as the normal class C_N , the class with the fewest samples is defined accordingly as anomaly class C_A . As the training is semi-supervised, the training set X_{tr} consists of 80 % of the C_N data. The remaining 20 % of C_N contribute to the test set X_{te} in conjunction with anomalies from the least represented class in the original dataset, so that 10 % of the test set are anomalies. As some datasets have only limited samples, and at least one anomalous sample is required per test set, 93 UCR datasets are viable for evaluation. According to [20], 10 resamples of this approach are used with random initialization for all empirically evaluated methods, including Z-ROCKET for robust results. Experimental setup: All experiments are conducted on an AMD Ryzen 9 5900X CPU.

A. Z-ROCKET-Parameter 1: Feature Pruning

The proposed approach can be modified through optional pruning of especially high or low kernel activations (step 3 in Section III-C). When sorting random kernels by their activation from high to low on the normal training data in $\bar{\mu}$, the following interpretation can be distinguished: If the mean across all training segments (which is computed in step 2 in Section III-C), is low for a specific kernel i , that means that this random kernel does not match the given time series well. If this kernel feature is suddenly high during inference, this can be interpreted as the anomalous emergence of a new pattern, that was not prevalent in normal data. Accordingly, high kernel activations in normal training data represent a match of the kernel and existing patterns in the time series, therefore a reduction of such features during inference can be interpreted as the absence of patterns existing in normal data. For empirical evaluation, three cases are distinguished: Pruning the top 33 % in the mean vector of step 2 in Section III-C), the bottom 33 % and the full set.

The experiments on the UCR dataset in Table I show no difference between the three options. However, this only indicates that, with the chosen approach, the class differences in the UCR dataset were reflected equally by the emergence of new patterns and the disappearance of normal patterns. This can be different in production depending on the anomaly types that should be addressed. This is relevant when the application scenario specifically targets the continuity of existing patterns in normal data or the introduction of anomalous patterns.

B. Z-ROCKET-Parameter 2: Kernel number

The number of kernels n_k is a hyperparameter that affects not only performance but also computation time. Experimental results are shown in Fig. 2. As a reference, MiniRocket's recommended 10,000 kernels are used. The x-axis shows the difference in AUROC for each kernel count compared to this

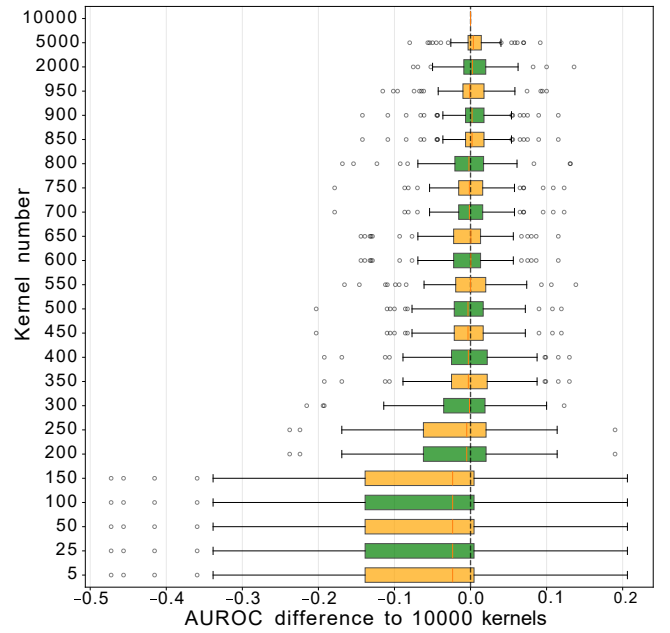


Fig. 2: AUROC for different numbers of kernels. The reference is 10000 kernels. Whiskers indicate the most extreme values within $1.5 \times \text{IQR}$ (interquartile range).

reference. It can be seen that the number of kernels in the range of 550–5000 yields very similar results, with AUROC values between 0.85 and 0.86. Due to the faster computation time, we recommend 550 kernels as the default.

The mean training time ranges from 45 ms (550 kernels) to 150 ms (2000 kernels), showing roughly linear behavior, but without statistically significant performance improvements. 550 kernels are used by default when comparing a single Z-ROCKET with other methods. In particular, the mean performance hardly improves; only the variance decreases with more kernels. This empirically determined parameter setting is consistent with the results in [5], which is notable since they used ROCKET [3], while Z-ROCKET uses MiniRocket [4].

C. Z-ROCKET-Parameter 3: Number of top-k kernel activations for threshold computation

In step 4 in Section III-C, the top-k largest absolute z-scores are aggregated into a mean as anomaly score. Results on the 93 UCR datasets are shown in Table II. It can be observed that the results are very stable with respect to the choice of the parameter k in the range $k = 3-100+$. For our code implementation, we arbitrarily set $k=5$, but note that this is not a sensitive hyperparameter, as long as $k \ll n_k$.

As an additional hyperparameter, the percentile used in step 5 of Section III-C determines the anomaly score threshold separating normal and anomalous inputs. The AUROC is invariant to the choice of this percentile threshold: changing it only affects the trade-off between precision and recall. Since the optimal balance depends on the specific application, this aspect is not discussed in detail here. In our experiments, we arbitrarily set $p=0.99$.

TABLE I: Results of empirical experiments of Z-ROCKET and existing methods on all 93 UCR datasets used in existing literature.

Method	AUROC	Mean train time (s)	Inference time (ms)
Z-ROCKET ($n_k = 550$)	0.85	0.04	0.08
Z-ROCKET ($n_k = 2000$)	0.86	0.15	0.26
ROCKAD	0.90	2.13	15.09
iForest	0.78	0.10	0.11
KNN	0.79	0.03	0.59
LOF	0.79	0.07	0.58

TABLE II: Performance of Z-ROCKET for varied top-k absolute z-scores for anomaly score aggregation with $n_kernels = 2000$.

Top k aggregated z-scores	AUROC
1	0.838
2	0.848
3	0.854
4	0.857
5	0.859
10	0.860
50	0.857
100	0.852

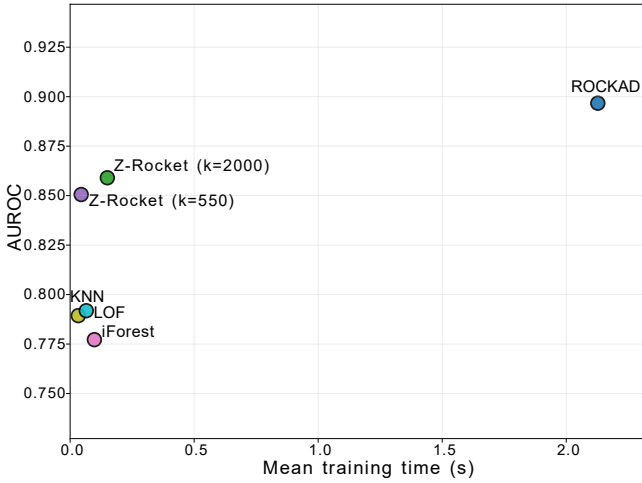


Fig. 3: AUROC over mean training time on all 93 AD comparison datasets from UCR. Our proposed approach, Z-ROCKET, reaches performance moderately lower than ROCKAD while being 50 times faster in terms of mean training time. Lightweight baseline methods such as KNN, LOF, and iForest are consistently outperformed by Z-ROCKET.

D. Comparison to other methods

This section presents the empirical experimental results of Z-ROCKET in direct comparison with the only other whole time series convolutional kernel-based method ROCKAD as well as fast baseline methods. A comparison to more computationally expensive methods, such as LSTMs or tsfresh-based feature extraction methods are given in [5]. The results

of 10 resamples on all 93 valid UCR datasets (as described in Section III) are given in Table I.

Overall, the proposed Z-ROCKET method offers a compelling trade-off between predictive performance and computational efficiency. While ROCKAD achieves the highest mean AUROC of 0.90, this gain comes at a substantial computational cost due to the combined use of PPV [3], maximum-based features, and an additional KNN stage, resulting in runtimes that exceed fast baselines by an order of magnitude. In contrast, Z-ROCKET is approximately 14× faster with $n_kernels = 2000$, and up to 50× faster in training for a more efficient configuration with $n_kernels = 550$, while incurring only a modest AUROC reduction of 0.05. The speedup compared to ROCKAD is primarily due to the faster MiniRocket, which omits the maximum pooling feature of the slower ROCKET. An important finding of this work is that this omission does not prevent competitive whole time series AD. In active deployment (inference time), Z-ROCKET is twice as fast as KNN and LOF, slightly faster than iForest and 189× faster than ROCKAD.

This trade-off of AUROC and training time is further visualized in Fig. 3. It can be seen that Z-ROCKET (ours) forms a cluster that combines the low training time of lightweight methods with the high AUROC performance of ROCKAD. Within Z-ROCKET, it is possible to adjust performance and computation time as needed by varying the number of kernels.

These results indicate that Z-ROCKET achieves competitive training and inference efficiency while maintaining superior detection performance compared to similarly efficient baselines. This balance highlights the practical value of the z-score-based approach as a scalable and effective contribution to AD.

E. Limitations of this work

The experiments in this work are limited to univariate data, however the framework is inherently compatible with multivariate inputs, enabling future work in this direction. Other autoencoder-based models are not compared in this work as baselines, as this is already done in [5]. Further, a sensitivity analysis of the percentile threshold, introduced in Section III is omitted, as the requirements for precision and recall are use-case dependent. This work uses a threshold of 0.99.

F. Adaptability of the method against evolving attacks

The random convolutional kernels that are used for training, can be initialized based on a random seed according to [3], [4]. This allows fast retraining and redeployment. If certain anomalies are suspected to fool the existing set of kernels, they can be recreated on demand or as a routine measure in deployment. This changes the exact patterns that the system is sensitive to, however as long as a sufficiently large and diverse set of kernels is used, the system can detect any type of zero-day anomaly that induces a change in any kernel activation.

V. CONCLUSION AND OUTLOOK

In this work, we presented Z-ROCKET, a lightweight semi-supervised AD algorithm for whole-time-series. By leveraging the fast random convolutional kernel transformation of MiniRocket alongside a top- k z -score thresholding scheme, Z-ROCKET eliminates the need for complex downstream classifiers. Our empirical evaluation across 93 heterogeneous UCR benchmark datasets demonstrates that Z-ROCKET delivers strong AD performance, achieving an AUROC of 0.86 and consistently outperforming fast baselines like KNN, iForest, and LOF at a similar computational cost. Moreover, it closely approaches the performance of state-of-the-art methods such as ROCKAD, while offering an average $50\times$ speedup in training time. This speedup is mainly attributed to the faster MiniRocket transform; the main contributions of this work are the demonstration that MiniRocket provides sufficient features for kernel-based whole time series AD despite omitting the maximum feature used by ROCKET. Furthermore, we show that z -score thresholds in that feature space can achieve competitive AD performance without a complex downstream classifier. Future work will focus on detailed ablation studies to systematically explore which pooling operators yield the best AD performance when paired with the proposed z -score thresholding approach.

ACKNOWLEDGMENT

This work has been supported by the Federal Ministry of Research, Technology, and Space of the Federal Republic of Germany (Förderkennzeichen 16KIS2239K, SUSTAINET_guarDian, 16KIS2453, PATHCARE, and 16KIS2402K, Open6GHub+). The authors alone are responsible for the content of the paper.

REFERENCES

- [1] M. Rüb, J. Herbst, O. Mey, *et al.*, "Large Agent Models in Patient Care and Clinical Communication Infrastructure: Applications and Challenges," in *2024 IEEE 6th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA)*, 2024, pp. 39–44. DOI: 10.1109/ICCCMLA63077.2024.10871706.
- [2] K. Dev, S. A. Khawaja, E. Zeydan, K. Singh, and M. Debbah, "Advanced Architectures Integrated With Agentic AI for Next-Generation Wireless Networks," *IEEE Communications Standards Magazine*, pp. 1–8, 2025. DOI: 10.1109/MCOMSTD.2025.3632205.
- [3] A. Dempster, F. Petitjean, and G. I. Webb, "ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels," *Data Mining and Knowledge Discovery*, vol. 34, no. 5, 1454–1495, Jul. 2020. DOI: 10.1007/s10618-020-00701-z.
- [4] A. Dempster, D. F. Schmidt, and G. I. Webb, "MiniRocket: A Very Fast (Almost) Deterministic Transform for Time Series Classification," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, New York: ACM, 2021, pp. 248–257.
- [5] A. Theissler, M. Wengert, and F. Gerschner, "ROCKAD: Transferring ROCKET to Whole Time Series Anomaly Detection," in *Advances in Intelligent Data Analysis XXI*. Springer Nature Switzerland, 2023, 419–432. DOI: 10.1007/978-3-031-30047-9_33.
- [6] A. A. Melakhsou and M. Batton-Hubert, "Explainable Abnormal Time Series Subsequence Detection Using Random Convolutional Kernels," in *Deep Learning Theory and Applications*, D. Conte, A. Fred, O. Gusikhin, and C. Sansone, Eds., Cham: Springer Nature Switzerland, 2023, pp. 280–294.
- [7] X. Shu, S. Zhang, Y. Li, and M. Chen, "An anomaly detection method based on random convolutional kernel and isolation forest for equipment state monitoring," *Eksplotacja i Niezawodność – Maintenance and Reliability*, vol. 24, no. 4, pp. 758–770, 2022. DOI: 10.17531/ein.2022.4.16.
- [8] Y. Liu and K. Yang, "Asynchronous Decentralized Federated Anomaly Detection for 6G Networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 11, no. 5, pp. 3384–3396, 2025. DOI: 10.1109/TCCN.2025.3538020.
- [9] H. Goumudi and S. Pierre, "Real-Time Anomaly Detection in IoMT Networks Using Stacking Model and a Healthcare-Specific Dataset," *IEEE Access*, vol. 13, pp. 70352–70365, 2025. DOI: 10.1109/ACCESS.2025.3563158.
- [10] C.-C. M. Yeh, Y. Zhu, L. Ulanova, *et al.*, "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)*, 2016, pp. 1317–1322. DOI: 10.1109/ICDM.2016.0179.
- [11] C. H. Lubba, S. S. Sethi, P. Knaute, *et al.*, "catch22: CAnonical Time-series CHaracteristics: Selected through highly comparative time-series analysis," *Data Mining and Knowledge Discovery*, vol. 33, no. 6, 1821–1852, Aug. 2019. DOI: 10.1007/s10618-019-00647-x.
- [12] S. Schmidl, P. Wenig, and T. Papenbrock, "Anomaly detection in time series: a comprehensive evaluation," *Proc. VLDB Endow.*, vol. 15, no. 9, 1779–1797, May 2022. DOI: 10.14778/3538598.3538602.
- [13] C. W. Tan, A. Dempster, C. Bergmeir, and G. I. Webb, "MultiRocket: multiple pooling operators and transformations for fast and effective time series classification," *Data Mining and Knowledge Discovery*, vol. 36, no. 5, 1623–1646, Jun. 2022. DOI: 10.1007/s10618-022-00844-1.
- [14] M. Middlehurst, P. Schäfer, and A. Bagnall, "Bake off redux: a review and experimental evaluation of recent time series classification algorithms," *Data Mining and Knowledge Discovery*, vol. 38, no. 4, 1958–2031, Apr. 2024. DOI: 10.1007/s10618-024-01022-1.
- [15] A. Bagnall, J. Lines, A. Bostrom, J. Large, and E. Keogh, "The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances," *Data Mining and Knowledge Discovery*, vol. 31, no. 3, 606–660, Nov. 2016. DOI: 10.1007/s10618-016-0483-9.
- [16] W. Xi and J. Lin, "RandomAD: A Random Kernel-Based Anomaly Detector for Time Series," in *Machine Learning and Knowledge Discovery in Databases. Research Track*. Springer Nature Switzerland, Oct. 2025, 159–175. DOI: 10.1007/978-3-032-05962-8_10.
- [17] Z. Ling, T. Aoyama, K. Yano, and E. Cohen, "ROCKET-LRP: Explainable Time Series Classification with Application to Anomaly Prediction in Manufacturing," in *2025 IEEE 21st International Conference on Automation Science and Engineering (CASE)*, 2025, pp. 576–583. DOI: 10.1109/CASE58245.2025.11163765.
- [18] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," *SIGMOD Rec.*, vol. 29, no. 2, 93–104, May 2000. DOI: 10.1145/335191.335388.
- [19] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in *2008 Eighth IEEE International Conference on Data Mining*, 2008, pp. 413–422. DOI: 10.1109/ICDM.2008.17.
- [20] H. A. Dau, E. Keogh, K. Kamgar, *et al.*, *The UCR Time Series Classification Archive*, https://www.cs.ucr.edu/~eamonn/time_series_data_2018/, 2018.