EADS: An Early Anomaly Detection System for Sensor-based Multivariate Time Series

Yihao Ang,†‡ Qiang Huang,†Anthony K. H. Tung,† Zhiyong Huang†‡
†School of Computing, National University of Singapore, Singapore
‡NUS Research Institute in Chongqing, Chongqing, China
{yihao_ang, huangq, atung, huangzy}@comp.nus.edu.sg

***Abstract*—Early Anomaly Detection (AD) in sensor-based Multivariate Time Series (MTS) is crucial for addressing signs of operational failures. However, existing AD methods either struggle to identify anomalies at an early stage or lean heavily on intricate neural networks and extensive data for model training, compromising clarity and interpretability. To bridge this gap, we pioneered CAD, a novel AD framework based on correlation analysis. It harnesses Time-Series Graphs (TSGs) to monitor sensor correlation changes. Through meticulous analysis of these changes, CAD excels in ascertaining the precise time of anomalies and identifying the implicated sensors. In this demonstration, we introduce EADS, an Early Anomaly Detection System built upon CAD for sensor-based MTS. We navigate multiple scenarios to illustrate the prowess of EADS in serving as an early AD benchmark platform, offering insightful abnormal time interpretability, and facilitating timely predictive maintenance. The source code is available at https://anonymous.4open.science/r/EADS-9A80/.**

***Index Terms*—Early Anomaly Detection, Outlier Detection, Multivariate Time Series, Correlation Analysis**

I. Introduction

Sensor-based Multivariate Time Series (MTS) are extensively employed across various industries. They play a crucial role in tasks like monitoring manufacturing processes for assembly lines [11] and recording operating states of infrastructures such as server machines [15]. A primary application of sensor-driven MTS is to enable predictive maintenance. By detecting anomalies in their earliest stages, it becomes possible to prevent severe damage and ensure timely restoration.

Early Anomaly Detection (AD) in sensor-based MTS, however, is challenging. (1) MTS often emerges from a diverse array of sensor measuring readings. This diversity makes it hard to pinpoint consistent patterns that represent normal and abnormal events. (2) A minor machine failure can escalate and impact adjacent components; thus, identifying affected sensors along with the abnormal time is crucial. (3) Industries prioritize methods that are interpretable yet user-friendly. An intuitive interface is paramount to aid users in comprehending the nature of failures and identifying faulty components.

While much of prior research has emphasized unsupervised AD, many such methods fall short in early anomaly detection. Data-mining-based methods [5], [8], [10] stand out for their efficiency and interpretability. Yet, as they neglect the temporal dependency of MTS, they predominantly detect outliers and often miss early anomaly signs. On the other hand, advanced deep-learning-based methods [1], [3], [9], [15] excel in accuracy by tapping into latent features and harnessing temporal information. However, they typically require complex neural networks and extensive training data, and their outcomes are hard to interpret. Thus, deploying them in real-world industries for early anomaly detection is troublesome.

To tackle the aforementioned challenges, we recently introduced CAD [2], a novel AD method based on correlation analysis. CAD focuses on monitoring underlying sensor correlations and identifying anomalies by looking at the unusual changes in their correlations. CAD first converts the MTS into a sequence of Time Series Graphs (TSGs). In these TSGs, each vertex represents an individual sensor, and the edges are established between sensors and their highly correlated neighbors. To detect anomalies, CAD begins by finding subsets of vertices exhibiting altered correlations. It then tracks these unusual correlation variations across a series of TSGs. Even if the sensors do not manifest significant anomalous behaviors, their correlations are likely to break down when an anomaly occurs. By analyzing such changes, CAD can report affected sensors and detect anomalies in their early stages.

In this demonstration, we propose EADS, an Early Anomaly Detection System built upon CAD for sensor-based MTS. EADS is a web-based application comprising three main components: (1) Early AD Performance Evaluation: Users can rigorously juxtapose CAD detection outcomes with the results from other AD methods. (2) Community Change Visualization: It assists users in understanding CAD's operational mechanics and its superior efficacy rationale. (3) Sensor Correlation Exploration: It permits users to explore the sensor correlation through TSG visualization. Upon deployment, EADS provides invaluable insights to both academic researchers and industry experts through three key scenarios:

(1) **Early AD Benchmark Platform:** While various AD benchmark platforms exist [7], [13], none specifically focus on early anomaly detection capabilities. EADS fills this gap by establishing itself as an early AD benchmark platform, granting users the opportunity to test, compare, and assess the performance of different AD methods in terms of early detection.

(2) **Abnormal Time Interpretability:** While CAD excels at early anomaly detection by identifying unusual community changes, its interpretability has remained largely unexplored. EADS demystifies the CAD process and community shifts during the anomalies, enhancing the interpretability of abnormal timings.

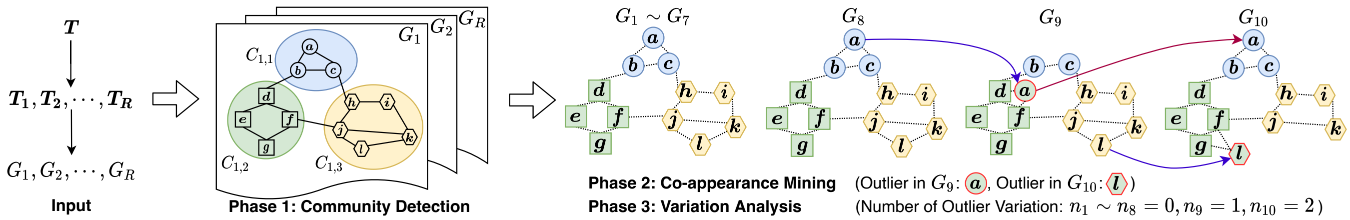


Fig. 1. An overview of the CAD framework.

(3) **Predictive Maintenance Support:** In numerous real-world maintenance scenarios, having an intuitive system is paramount for industry professionals to enable effective predictive maintenance. EADS assists industry experts in exploring potentially abnormal sensors and undertaking prompt repairs.

II. CAD: The Methodology

A. Problem Formulation

We assume that an MTS $\boldsymbol{T}$ with n sensors is represented as a matrix, i.e., $\boldsymbol{T} = (\boldsymbol{s}_1; \cdots; \boldsymbol{s}_n)^{\mathsf{T}}$. Let l be the time series length. Each time series $\boldsymbol{s}_i$ ($1 \leq i \leq n$) can thus be denoted as an l-dimensional vector, i.e., $\boldsymbol{s}_i = (x_{i,1}; \cdots; x_{i,l})$. Here, $x_{i,j}$ ($1 \leq j \leq l$) signifies a sensor reading from $\boldsymbol{s}_i$ at a specific time point t_j. As anomalies typically manifest within a brief period, it is hard to detect anomalies directly from a long MTS. To address this, using a sliding window w and a step s ($1 \leq s < w$), we define $R = (l - w)=s + 1$ and partition $\boldsymbol{T}$ into R overlapping sub-matrices $\{\boldsymbol{T}_1; \cdots; \boldsymbol{T}_R\}$ of length w, where $\boldsymbol{T}_1 = \boldsymbol{T}[1 : w]$, $\boldsymbol{T}_2 = \boldsymbol{T}[1 + s : w + s]$, etc.

To track sensor correlations and promptly detect anomalies through notable variations, we convert each $\boldsymbol{T}_r$ into a k-Nearest Neighbor Graph $G_r = (V; E_r)$. In G_r, each vertex $\boldsymbol{v} \in V$ represents an individual sensor, and E_r connects each $\boldsymbol{v}$ to its top k correlated vertices. The edge weight $!(e)$, for $e = (\boldsymbol{u}; \boldsymbol{v}) \in E_r$, is set by the Pearson Correlation [14] of $\boldsymbol{u}$ and $\boldsymbol{v}$. We prune edges in G_r with absolute weights below a correlation threshold , yielding a Time Series Graph (TSG).

Unlike MTS, TSGs emphasize strong sensor correlations through edges and often exhibit community structures [12]. In a round r, G_r has m communities $\{C_{r,c}\}_{c=1}^{m}$. Any anomalous sensor shifts can then immediately manifest as community changes between consecutive rounds. Given this backdrop, we define the affected vertices as follows.

Definition 1 (Affected Vertices)**:** *For a TSG $G_r = (V; E_r)$ ($r \in (1; R]$) encompassing m communities, a vertex subset $V_r \subseteq V$ is affected if every $\boldsymbol{v} \in V_r$ moves into or out of $C_{r,c}$ between two consecutive rounds, i.e., $V_r = \{\boldsymbol{v} \in V \mid (\boldsymbol{v} \notin C_{r-1,c}$ and $\boldsymbol{v} \in C_{r,c})$ or $(\boldsymbol{v} \in C_{r-1,c}$ and $\boldsymbol{v} \notin C_{r,c})\}$.*

We denote each affected sensor $\boldsymbol{v} \in V_r$ as one *variation* if it changes across two consecutive rounds.

Definition 2 (Abnormal Time and Abnormal Sensors)**:** *Given an abnormal time threshold $\in [1; n]$ and R rounds of affected vertices $\{V_1; \cdots; V_R\}$, the abnormal time R_Z comprises consecutive rounds with $|V_r| \geq$ for each $r \in R$, and the abnormal sensors V_Z are the union of V_r across all $r \in R_Z$.*

B. The CAD Framework

Figure 1 depicts an overview of CAD for anomaly detection through a sequence of TSGs.

Phase 1: Community Detection. For each round $r \in [1; R]$, we employ the Louvain method [4] to partition the TSG G_r into m communities. This enables CAD to track the unusual correlation variations among vertices. Changes in community structure often signify alterations in machine operating states. We will elucidate this aspect in Section IV.

Phase 2: Co-appearance Mining. We mine the co-appearance relationships of vertices from their communities. By assessing the co-appearance between two vertices over consecutive rounds, one can monitor community changes of all vertices in each round. For a round $r \in (1; R]$ and a vertex $\boldsymbol{v}$ transitioning from its previous community $C_{r-1,c}$ to current $C_{r,c'}$, its co-appearance $S_r(\boldsymbol{v}; \boldsymbol{u})$ with any $\boldsymbol{u} \in V$ is computed as:

$$S_r(\boldsymbol{v}; \boldsymbol{u}) = \begin{cases} 1; & \text{if } \boldsymbol{u} \in C_{r-1,c} \text{ and } \boldsymbol{u} \in C_{r,c'} \\ 0: & \text{otherwise} \end{cases}$$

Furthermore, the co-appearance number $S_r(\boldsymbol{v})$ for each vertex $\boldsymbol{v}$ is given by $S_r(\boldsymbol{v}) = \sum_{\boldsymbol{u} \in V \wedge \boldsymbol{u} \neq \boldsymbol{v}} S_r(\boldsymbol{v}; \boldsymbol{u})$. This number, $S_r(\boldsymbol{v})$, counts the total vertices that have co-appeared with $\boldsymbol{v}$. Note that $S_r(\boldsymbol{v})$ may vary across rounds, especially when an anomaly occurs. To gain deeper insights into the co-appearance dynamics, we introduce the Ratio of Co-appearance Number (RC). For each $\boldsymbol{v} \in V$, its RC for round $r \in (1; R]$ is determined by: $RC_{\boldsymbol{v},r} = \frac{1}{r(n-1)} \sum_{i=1}^{r} S_i(\boldsymbol{v})$. We then assess if $\boldsymbol{v}$ is an outlier in round r by comparing $RC_{\boldsymbol{v},r}$ with a predefined outlier threshold . The set of outliers for a given round r is denoted as O_r and defined as $O_r = \{\boldsymbol{v} \in V \mid RC_{\boldsymbol{v},r} < \ \}$.

Phase 3: Variation Analysis. To precisely determine the abnormal time, we focus on the transitions of a vertex between its normal and abnormal states. For a round $r \in (1; R]$, n_r represents the number of vertices undergoing such a transition, i.e., $n_r = \sum_{\boldsymbol{v} \in V}((\boldsymbol{v} \Subset O_{r-1}$ and $\boldsymbol{v} \in O_r)$ or $(\boldsymbol{v} \in O_{r-1}$ and $\boldsymbol{v} \Subset O_r))$. According to the Weak Law of Large Numbers and Chebyshev's inequality, we derive:

$$P(|n_r - \ | \geq \ \cdot \ ) \leq \frac{1}{\eta^2};$$

where is a constant; and are the mean and standard deviation of n_r, respectively. We set $= 3$ for precise anomaly detection. Hence, the abnormal time threshold, as defined in Definition 2, is $= 3$. A round r is abnormal if $|n_r - \ | \geq$ 3 , as the communities in this round change significantly from prior rounds, deeming the outliers as abnormal sensors.

III. EADS: System Overview

We now introduce EADS. Its Graphical User Interface (GUI) is a stand-alone web application developed using Python 3.7 and the Dash framework [6]. The system architecture of EADS is showcased in Figure 2, and screenshots capturing the essence of EADS are presented in Figure 3. EADS incorporates basic functions, such as a control panel (as depicted in Figure 3(a.1)) for data uploading and parameter setting and a "Sensor-based Multivariate Time Series Overview" canvas (visible in Figure 3(a.2)) that provides tools for data selection and raw MTS visualization. Notably, EADS is built around three principal components that enhance user interaction: (1) Early AD Performance Evaluation, (2) Community Change Visualization, and (3) Sensor Correlation Exploration.

Component 1: Early AD Performance Evaluation. This component is designed to equip users to systematically gauge different AD methods, particularly the capability of early anomaly detection. For CAD, users can initiate it by setting the values of and , followed by clicking the "Run CAD" button (visible in Figure 3(a.3)). Upon completion, the detected abnormal times are displayed on the main screen (as shown in Figure 3(a.4)). If ground truth data is provided, it will be juxtaposed with the detected anomalies for a more intuitive comparison. One of the characteristics of CAD is its adeptness in identifying abnormal sensors. When users navigate to the detected abnormal times, these abnormal sensors are automatically highlighted, facilitating a profound comprehension of the anomaly's nature. In addition to CAD, EADS is compatible with other AD methods, such as ECOD [8], IForest [10], and RCoders [1]. Users can execute these by clicking on the "Run Selected Methods" button. Their outcomes are displayed along with CAD's results, enabling a comprehensive comparison. The system also supports a zoom-in feature, allowing users to closely inspect each identified anomaly.

In the second row of "Early AD Performance Evaluation" (Figure 3(a.5)), users can evaluate various AD methods under the modes of Point Adjustment (PA) [3], [15] and Delay-Point Adjustment (DPA) [2]. The measures encompass three aspects: (1) Effectiveness measures like F1-score (F1), Precision, Recall, and the confusion matrix values, computed using provided ground truths. EADS automatically determines the optimal abnormal threshold by performing a grid search ranging from 0 to 1, with a step of 0.001; (2) Relative comparisons between two methods, Ahead and Miss, providing clarity on the timeliness of detected anomalies; (3) Efficiency measures such as the training and testing time.

Component 2: Community Change Visualization. EADS allows users to visualize community changes for each round, as displayed in Figure 3(b.1). The flow diagram depicts community alterations. Within each column, distinct black line segments represent specific communities for a round, while the flows trace the transitions of communities tied to individual sensors. To enhance the viewing experience, EADS offers two display modes: (1) The "by Range" mode lets users specify a range of rounds. (2) The "by Anomaly" mode is

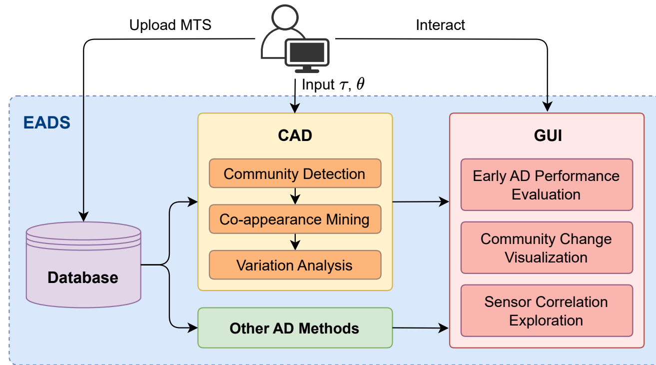


Fig. 2. The system architecture of EADS.

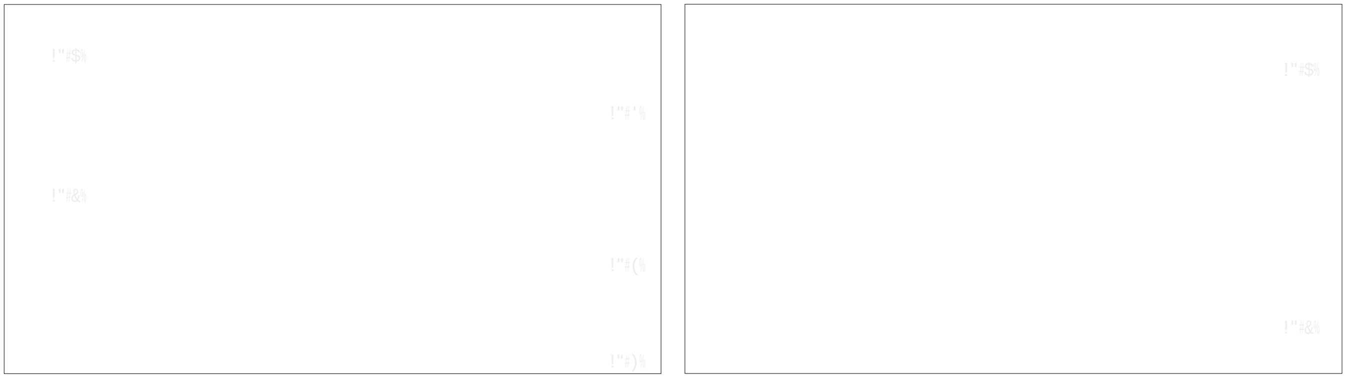
tailored to facilitate the exploration of community dynamics around an anomaly, providing deeper insight into community transitions during selected rounds. It is crucial to note that even if sensors do not show pronounced abnormal behaviors when an anomaly occurs, their correlations could be disrupted, leading to community changes. CAD adeptly captures such changes, timely identifying outliers.

Component 3: Sensor Correlation Exploration. Delving into community changes, EADS provides an exploration of sensor correlations through TSG visualization, as seen in Figure 3(b.2). Each vertex in the TSG represents an individual sensor, with edges representing the correlation intensity between sensors–the thicker the edge, the higher the correlation. Vertices are color-coded by their associated communities. For enhanced user interaction, clicking on a vertex reveals comprehensive information about that sensor, given it's provided. In rounds detected as abnormal, the abnormal sensors stand out with a square shape outlined in yellow color. Hovering over these vertices highlights a ranked list of sensors correlating with the abnormal one, ordered by decreasing correlation strength. Such visualization is paramount, offering users insights into the abnormal sensors' role within a TSG and their surrounding correlations. It facilitates timely maintenance actions, allowing users to address potentially faulty sensors.

IV. Demonstration Scenarios

This demonstration pursues three key objectives: (1) functions as an early AD benchmark, facilitating the evaluation of diverse AD methods for early anomaly detection; (2) enriches users understanding of the CAD process, bolstering abnormal time interpretability; (3) guides industry experts towards timely predictive maintenance upon anomaly detection.

Scenario 1: Early AD Benchmark Platform. In this scenario, EADS serves as an early AD benchmark platform, catering to both researchers and industry experts to evaluate the early AD capabilities of various methods. When inputting a real-world sensor-based MTS, e.g., SMD 1_6 [15], two prolonged anomalies are highlighted in yellow. After executing CAD and other AD methods, their results are evident in Figure 3(a.4). Users can further zoom into specific rounds to discern when each method reports the anomaly. Figure 3(a.5) provides a comparative evaluation by various measures, guiding users in



(a) Early AD Performance Evaluation.

(b) Community Change Visualization and Sensor Correlation Exploration.

Fig. 3. Screenshots of EADS.

selecting suitable AD methods. Moreover, EADS facilitates the addition of new AD methods through a user-friendly control panel checklist, enhancing adaptability.

Scenario 2: Abnormal Time Interpretability. The second scenario offers users a deeper understanding of the CAD process, especially how communities change across different rounds during an anomaly, thereby improving the interpretability of abnormal times. We proceed with the SMD 1_6 dataset, activate the "by Anomaly" mode, and select "Anomaly 6," as depicted in Figure 3(b.1). EADS then pinpoints rounds before and after this anomaly is detected, expanding the visualization range from Round 229 to Round 239. A significant shift is observed starting from Round 234, with swift community alterations and rapid sensor membership changes, especially from Round 235. These insights offer researchers a clear interpretation of anomaly timelines and equip industry experts with actionable data to optimize operations. Take the case of grid energy managers, who need prompt, informed decisions. Understanding community change patterns and sensor interactions helps determine anomaly credibility and severity. When many stable communities shift significantly, it probably hints at a critical concern, necessitating immediate intervention.

Scenario 3: Predictive Maintenance Support. The last scenario assists users in identifying potential abnormal sensors, thereby supporting timely predictive maintenance. Referencing "Anomaly 6" in Figure 3(b.2), EADS presents the TSG for Round 235. The five communities are visible in different colors, and the detected abnormal sensors are distinctly marked with yellow squares. Inspecting each abnormal sensor, EADS ranks its associated sensors based on their correlation. This granularity not only enhances interpretability for researchers but also provides additional insights for industries that need urgent action. Take the manufacturing sector as an example: a minor ignition in machinery might initially yield a subtle temperature rise, causing certain sensors to be flagged as abnormal. Yet, if maintenance technicians delve into sensors linked to these abnormal sensors, they can rapidly identify vulnerable machinery and institute prompt countermeasures. Thus, EADS helps professionals preemptively tackle anomalies, reducing disruption risks and ensuring timely restoration.

V. Conclusions

This demonstration introduces EADS, a novel early AD system for sensor-based MTS. Leveraging CAD–an advanced early AD method grounded in sensor correlation variations–EADS serves as a pivotal tool for both academia and industry. It benchmarks the capabilities of diverse AD methods for early anomaly detection, enhances abnormal time interpretability, and streamlines predictive maintenance processes.

References

[1] A. Abdulaal, Z. Liu, and T. Lancewicki, "Practical approach to asynchronous multivariate time series anomaly detection and localization," in *KDD*, 2021, pp. 2485–2494.

[2] Y. Ang, Q. Huang, A. K. Tung, and Z. Huang, "A stitch in time saves nine: Enabling early anomaly detection with correlation analysis," in *ICDE*, 2023, pp. 1832–1845.

[3] J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, "USAD: Unsupervised anomaly detection on multivariate time series," in *KDD*, 2020, pp. 3395–3404.

[4] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[5] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *SIGMOD*, 2000, pp. 93–104.

[6] P. T. Inc., "Dash," https://plotly.com/dash/, 2021.

[7] V. Jacob, F. Song, A. Stiegler, B. Rad, Y. Diao, and N. Tatbul, "Exathlon: A benchmark for explainable anomaly detection over time series," *Proc. VLDB Endow.*, vol. 14, no. 11, pp. 2613–2626, 2021.

[8] Z. Li, Y. Zhao, X. Hu, N. Botta, C. Ionescu, and G. Chen, "ECOD: Unsupervised outlier detection using empirical cumulative distribution functions," *TKDE*, 2022.

[9] Z. Li, Y. Zhao, J. Han, Y. Su, R. Jiao, X. Wen, and D. Pei, "Multivariate time series anomaly detection and interpretation using hierarchical intermetric and temporal embedding," in *KDD*, 2021, pp. 3220–3230.

[10] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *ICDM*, 2008, pp. 413–422.

[11] L. Martí, N. S. Pi, J. M. Molina, and A. C. B. Garcia, "Anomaly detection based on sensor data in petroleum industry applications," *Sensors*, vol. 15, no. 2, pp. 2774–2797, 2015.

[12] M. E. Newman, "Modularity and community structure in networks," *PNAS*, vol. 103, no. 23, pp. 8577–8582, 2006.

[13] J. Paparrizos, Y. Kang, P. Boniol, R. S. Tsay, T. Palpanas, and M. J. Franklin, "Tsb-uad: an end-to-end benchmark suite for univariate time-series anomaly detection," *Proc. VLDB Endow.*, vol. 15, no. 8, pp. 1697–1711, 2022.

[14] K. Pearson and F. Galton, "Notes on regression and inheritance in the case of two parents," *Proceedings of the Royal Society of London*, vol. 58, pp. 240–242, 1895.

[15] Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in *KDD*, 2019, pp. 2828–2837.