Budapest University of Technology and Economics
Faculty of Electrical Engineering and Informatics
Department of Networked Systems and Services

RealTime-AT: Enhancing Real-time Anomaly Detection for Streaming Multivariate Time Series

Scientific Students' Association Report

Author:

Burak Colak

Advisor:

dr. Károly Farkas

2023

Contents

Kivonat

A digitális átalakulás korszakában a többváltozós idősorok valós idejű megfigyelése és analízise rendkívül fontosá vált. Bonyolult ipari rendszerek telemetriájától komplex pénzügyi hálózatokig a rendellenes viselkedés azonnali észlelése lényeges eszköz, mely gyakran képes megakadályozni rendszerszintű zavarokat vagy pénzügyi rendellenességeket.

A szakirodalomban egyre elterjedtebb megközelítés a gépi tanulás alapú anomáliadetekció alkalmazása nagy méretű adatsorok kezelésére. A valódi kihívás azonban ezen modellek alkalmazása valós idejű adatok esetén. Az Anomaly Transformer, melyet Jiehui Xu és munkatársai mutattak be, kiemelkedő eredményeket mutat többváltozós, de eltárolt historikus adatokon végzett idősoros anomáliadetekció esetén. Valós idejű alkalmazása azonban továbbra is kutatásra váró terület.

Célom így az Anomaly Transformer valós idejű helyzetekre történő alkalmazása, melynek eredményeként létrehoztam továbbfejlesztett algoritmusomat, a RealTime-AT-t. Felismerve a valós idejű adatok belső összefüggéseit, megközelítésem alapja, hogy finomhangoljam a modellben található csúszóablak-mechanizmust. Ez biztosítja, hogy a RealTime-AT folyamatosan a legújabb beérkező adatpontokat vegye figyelembe, lehetővé téve a gyors, valós idejű anomáliadetekciót.

Vizsgálataim eredményei alapján a RealTime-AT összemérthető teljesítményt mutat az eredeti Anomaly Transformer algoritmussal a tradícionális anomáliadetekciós metrikák tekintetében, mint Accuracy, Precision, Recall és F-score. Ellenben, a működéséhez szükséges erőforrások tekintetében, így a GPU memóriahasználatot illetően jelentősen alacsonyabb (közel nyolcadakkora) igénnyel rendelkezik, ami előnyös a szűkös erőforrású, valós idejű anomáliadetekció esetén.

Meggyőződésem, hogy a bemutatott megoldásokkal a RealTime-AT megfelel a valós idejű adatfolyamok követelményeinek, új megközelítést teremtve a pontos, valós idejű anomáliadetekció területén.

Abstract

In the era of digital transformation, real-time monitoring and analysis of multivariate time series data have become paramount. From telemetry of intricate industrial systems to complex financial networks, the ability to detect abnormal behavior promptly can provide critical insights, potentially preventing system disruptions or financial irregularities.

Anomaly detection using machine learning, with its capability to handle vast datasets, has emerged as a promising tool in this domain. However, the real challenge is in adapting these models to handle data as it streams. Anomaly Transformer, introduced by Xu et al. stands out with its impressive results in offline settings for multivariate time series anomaly detection. Yet, its adaptability to real-time scenarios remains an area ripe for exploration.

Our research is centered on refining the Anomaly Transformer for online scenarios, resulting in our improved algorithm named RealTime-AT. Recognizing the inherent complexities of streaming data, our approach focuses on fine-tuning the existing sliding window mechanism within the model. This ensures that RealTime-AT is continuously fed with the most recent points of streaming data, allowing for fast online anomaly detection.

The results of our experiments presented in this study show that RealTime-AT achieved comparable performance with the original Anomaly Transformer in the traditional anomaly detection metrics of Accuracy, Precision, Recall and F-score. However, resource demands such as GPU memory utilization decreased significantly (to almost an eighth of the original need), which is paramount in resource-constrained real-time environments.

We are optimistic that with targeted refinements, the Anomaly Transformer can be optimized to cater to the demands of real-time data streams, ushering in a new approach for timely and precise anomaly detection.

Chapter 1

Introduction

The digital transformation epoch has brought to the fore the significance of instantaneously analyzing and monitoring multivariate time series data. Whether it's the telemetry from sophisticated industrial setups or the intricate web of financial systems, timely identification of anomalies can be the key to averting potential mishaps. These anomalies might range from disruptions in system operations to financial discrepancies that could have profound implications if left unnoticed.

Historically, the task of detecting anomalies in time series data has been a compelling pursuit with substantial challenges and complexities. In the early days of anomaly detection, techniques predominantly relied on pointwise representation or pairwise association methodologies. [2] These methods, while groundbreaking in their time, had their inherent limitations. They provided the early steps towards understanding the behavior of time series data, but their intrinsic designs often meant struggling with comprehending the elaborate patterns and relationships manifested over longer time horizons.

Time series anomalies are essentially outliers, deviating significantly from expected patterns or behaviors. Due to their rarity within datasets, earlier methods often struggled to detect them reliably. Unlike typical data points that form consistent relationships within the broader dataset, anomalies are characterized by distinct differences, especially when juxtaposed with their immediate temporal neighbors. Because of this subtle distinction, traditional techniques sometimes produced false positives or failed to identify these anomalies altogether.

With technological advancements and a deeper understanding of time series data dynamics, the field witnessed a paradigm shift with the introduction of Transformer models [3]. Known for their sophisticated ability to process sequential data, these models, originally designed for natural language processing tasks, found a natural home in time series anomaly detection. Their superior capability to understand both pointwise and pairwise associations presented a renewed approach to the challenges posed by anomaly detection.

A standout representative in this new type of models is Anomaly Transformer [1], created by Xu et al., which has exhibited commendable performance in offline multivariate time series anomaly detection. Not only does it harness the self-attention mechanism innate to Transformer architectures, but it also introduces novel techniques tailored specifically for anomaly detection. One of its key innovations is the ability to leverage the self-attention weight distribution to precisely discern how each time point in a series associates with every other point. This intricate understanding led to the discovery of the 'Association Discrepancy' principle. This principle highlights the adjacent-concentration bias often found in anomalies. It posits that anomalies, due to their aberrant nature, tend to have

stronger associations with immediate neighbors rather than with distant data points. Such insights provide a robust and reliable criterion to delineate between regular patterns and potential anomalies.

The journey from traditional anomaly detection techniques to the advanced Transformer-based approaches paints a picture of a field in continuous evolution, with each step building on the last, driving towards more precise, reliable, and effective solutions. Anomaly Transformer has achieved commendable milestones in the realm of offline anomaly detection. Its prowess in discerning intricate patterns within large datasets has set it apart from its predecessors. However, the dynamic nature of modern data, particularly in industrial and mission-critical settings, underscores the need for online or real-time processing. The complexity arises when attempting to harness the power of Anomaly Transformer, inherently designed for static datasets, to meet the demands of continuously streaming data. The challenge is two-fold: ensuring real-time adaptability without compromising the depth of analysis.

The real-time detection of anomalies isn't merely a technical challenge but has profound real-world ramifications. In the case of service monitoring, for example, even minor disruptions can cascade into significant service outages if not addressed promptly, potentially leading to customer dissatisfaction and financial setbacks. In realms like space and earth exploration, the window to act upon a unique or critical discovery can be incredibly narrow, making timely insights indispensable. Similarly, in environments such as water treatment facilities, a delay in detecting anomalies can escalate into severe health hazards. Hence, the move from offline to online isn't just a progression in computational techniques but a requisite for modern-day challenges.

The foundational architecture of Anomaly Transformer, while revolutionary, was crafted with static datasets in mind. When deployed in real-time scenarios, certain inherent limitations become apparent. Its capacity to dissect and understand historical data is unparalleled, but this depth of analysis often comes at the cost of speed – making it less suitable for streaming data. The transition from offline to online, however, demands a radical rethinking of the model's architecture and processing mechanisms. Thus, while Anomaly Transformer performs adequately in offline settings, there's an impending need for an adaptive variant tailored for online environments.

Building upon this understanding, our research aims to bridge the gap between the offline prowess of Anomaly Transformer and the needs of real-time data processing. Our goal isn't just to make the model faster but to concurrently ensure speed, adaptability, and accuracy. By incorporating novel techniques and refining existing mechanisms, we have created a version of Anomaly Transformer that is attuned to the nuances of online data, ensuring that it remains the gold standard in anomaly detection, irrespective of the nature of data.

We introduce an advanced iteration of Anomaly Transformer, named RealTime-AT, which is tailored for online scenarios. Our emphasis has been on refining the sliding window mechanism, a critical component ensuring the model's timely response to streaming data. By doing so, we aspire to strike a balance between swift anomaly detection and maintaining model accuracy. Our ambition is to pave the way for a paradigm shift in anomaly detection, making it more aligned with the exigencies of real-time data streams.

Our exmerimental results highlighted that the performance of RealTime-AT in the traditional anomaly detection metrics (Accuracy, Precision, Recall and F-score) was comparable with the original Anomaly Transformer. But, the resource demands of our approach

such as GPU memory utilization decreased significantly, to approximately an eighth of the original requirement, which is crucial in resource-constrained real-time environments.

The rest of this study is organized as follows. Chapter 2 details related work in the field of multivariate anomaly detection, categorized by the type of approach taken. Afterwards, in Chapter 3, we present RealTime-AT, our proposed improvement of Anomaly Transformer. Chapter 4 shows our experimental results, evaluating RealTime-AT compared to its predecessor, Anomaly Transfer. Through our experimentation, we shed light on the capabilities of RealTime-AT in an online context. Finally, we conclude this study with Chapter 5, where we reflect on the improvements achieved by RealTime-AT, and possible future areas of research.

Chapter 2

Related work

In this chapter, we review various multivariate anomaly detection algorithms, emphasizing their adaptability and performance in both offline and online settings.

2.1 Graph Based Anomaly Detection Algorithms

Graph-based anomaly detection algorithms represent one of the promising methodologies explored in the field. This approach involves representing data as a graph and detecting anomalies based on their relationship and position within that graph.

The work by Deng and Hooi [4] proposes a method called Graph Deviation Networks (GDN). The GDN method aims to learn relationships between sensors as a graph and then identifies and explains deviations from the learned patterns. It involves four main components:

- Sensor Embedding: This uses embedding vectors to capture the unique characteristics of each sensor.
- Graph Structure Learning: This learns a graph structure representing dependence relationships between sensors.
- Graph Attention-Based Forecasting: This forecasts future values of each sensor based on a graph attention function over its neighbors.
- Graph Deviation Scoring: This identifies deviations from the learned relationships and localizes and explains these deviations.

Figure 2.1 shows the architecture of Deng and Hooi's work.

The paper also discusses the use of Graph Neural Networks (GNNs) and their variants, such as Graph Convolution Networks (GCNs) and Graph Attention Networks (GATs), which have shown success in time-dependent problems.

Similarly, Zhao et al. [5] present Multivariate Time-series Anomaly Detection via Graph Attention Network (MTAD-GAT), a self-supervised framework for multivariate time series anomaly detection. This approach considers each univariate time series as an individual feature and includes two graph attention layers in parallel to learn the complex dependencies of multivariate time-series in both temporal and feature dimensions. The approach jointly optimizes a forecasting-based model and a reconstruction-based model, obtaining

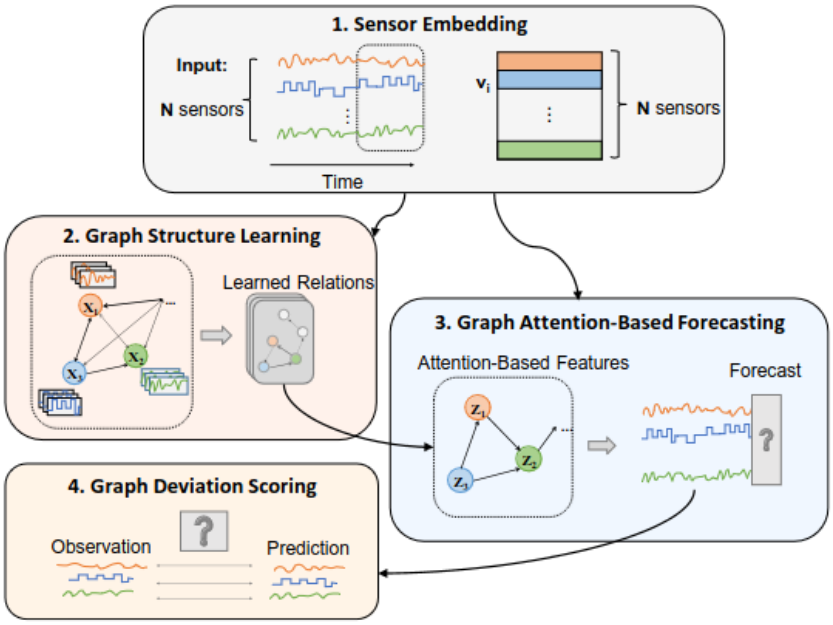

Figure 2.1: GDN's proposed framework [4]

better time-series representations through a combination of single-timestamp prediction and reconstruction of the entire time-series. This approach is shown in Figure 2.2.

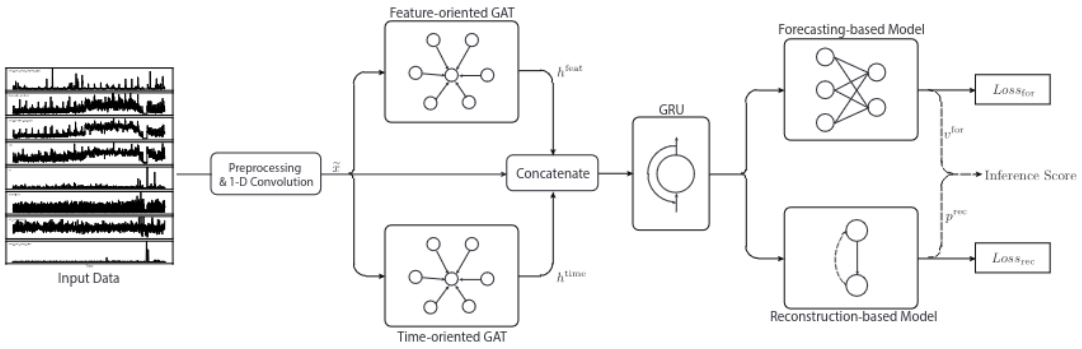

Figure 2.2: MTAD-GAT's proposed framework [5]

In summary, while Deng and Hooi focus on learning relationships between sensors and identifying deviations from these relationships using the GDN method, Zhao et al. with MTAD-GAT propose a self-supervised framework that uses GATs to learn complex dependencies in multivariate time-series data.

2.2 Generative Adversarial Networks Based Algorithms

In this section, we introduce Generative Adversarial Networks (GANs) [6] based algorithms. We note that both algorithms came from similar authors, thus the latter constitutes at the improvement of the former.

Both articles propose methods for anomaly detection in complex networked Cyber-Physical Systems (CPSs) using Generative Adversarial Networks (GANs) and Long-Short-Term-Memory Recurrent Neural Networks (LSTM-RNN) [7].

The first one by Li et al. presents Generative Adversarial Networks-based Anomaly Detection (GAN-AD)[8] with the following key points.

- The GAN-AD method uses LSTM-RNN to capture the distribution of multivariate time series of sensors and actuators under normal working conditions of a CPS.
- It is designed to classify deviant behaviors as possible attacks.
- The method was tested on a six-stage Secure Water Treatment (SWaT) system and showed high detection rates and low false positive rates compared to existing methods [9].

Figure 2.3 shows the framework published by Li et al..

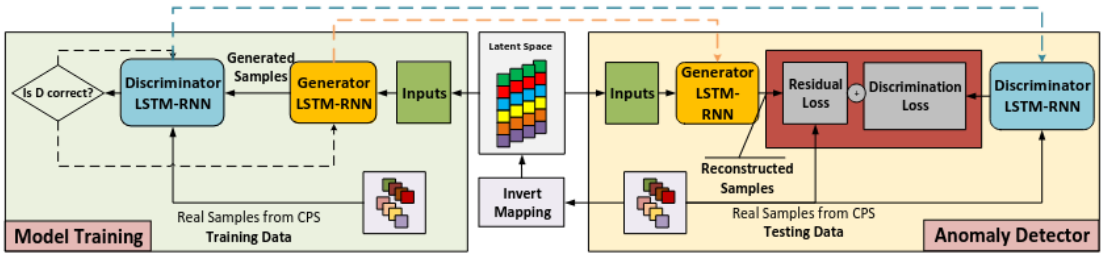

Figure 2.3: GAN-AD's proposed framework [8]

As an improvement, Li et al. created Unsupervised Multivariate Anomaly Detection using Generative Adversarial Networks (MAD-GAN)[10], which is summarized below, and presented in Figure 2.4.

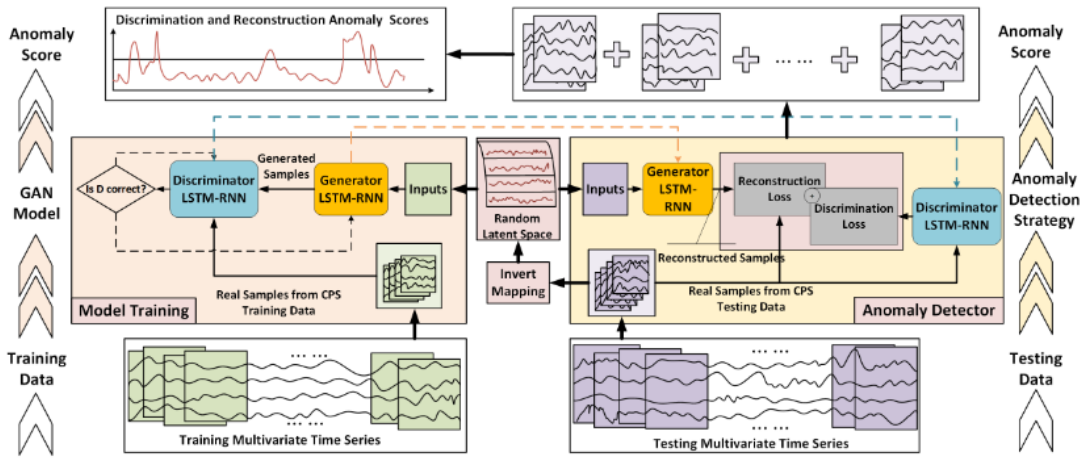

Figure 2.4: MAD-GAN's proposed framework [10]

- The MAD-GAN method considers the entire variable set concurrently to capture the latent interactions amongst the variables.

- It is designed to capture the temporal correlation of time series distributions.
- The authors tested MAD-GAN using two recent datasets collected from real-world CPS and showed that it is effective in reporting anomalies caused by various cyber-intrusions compared in these complex real-world systems.

In summary, while both methods use GANs and LSTM-RNN for anomaly detection in CPSs, MAD-GAN focuses on capturing latent interactions amongst variables and temporal correlations of time series distributions, while GAN-AD focuses on capturing the distribution of multivariate time series under normal conditions and classifying deviant behaviors as possible attacks.

2.3 Negative Selection Methods

We regard this approach as among the most promising ways to perform multivariate anomaly detection. In this section, we discuss two papers based on Negative Selection Algorithms (NSA) which generally employ search algorithms emulating how antibodies distinguish pathogens from body cells [11].

Negative Selection Algorithm for Anomaly Detection by Dasgupta and Majumdar [12] is summarized in the following points:

- The authors discuss the use of the Negative Selection Algorithm, which is based on the mechanisms of the human immune system, for anomaly detection in single and multidimensional data sets.
- Dasgupta and Majumdar argue that conventional approaches to anomaly detection are limited and do not adapt well to changes in data patterns over time.
- The immune-based approach offers a distributed and adaptive pattern recognition mechanism that can make self/non-self discrimination and learn to recognize relevant patterns.
- The authors suggest that this approach has the potential to provide effective solutions for anomaly detection in sensitive personnel data.

The other approach based on the Negative Selection Method is called Unsupervised Anomaly Detection via Variational Auto-Encoder for Seasonal KPIs in Web Applications, and was published by Sipple [11]. We summarize their main concepts as follows:

- This paper proposes an unsupervised approach for detecting anomalies in IoT devices using negative sampling to train a classifier to distinguish between normal and anomalous states.
- The authors report that their approach yields significantly higher AUC scores compared to state-of-the-art approaches against benchmark anomaly detection datasets.
- The method has been successfully deployed at large scale to predict failures in real-time in over 15,000 climate-control and power meter devices in 145 office buildings within the California Bay Area.
- The paper explores the problem of finding patterns in data that do not conform to expected behavior and how anomaly detection can be used to detect anomalous measurements from incoming data streams.

2.4 Gaussian Mixture Methods

Multidimensional Time Series Anomaly Detection is a GRU-based Gaussian Mixture Variational Autoencoder Approach [13] by Guo et al., and is summarized in the following points:

- The article introduces a new type of neural network model called Neural Ordinary Differential Equations (Neural ODEs).
- The model is based on the concept of ODEs, where the derivative of the hidden state is parameterized with a neural network.
- The model is trained using a method called adjoint sensitivity method, which is a variant of backpropagation that is suitable for ODEs.
- The model is evaluated on various tasks including image recognition, density estimation, and time series prediction.

Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection [14] (DAGMM) by Zong et al. (see Figure 2.5) consists of two major components, a compression network and an estimation network. The compression network performs dimensional reduction for input samples by a deep auto-encoder, prepares their low-dimensional representations from both the reduced space and the reconstruction error features, and feeds the representations to the subsequent estimation network; the estimation network takes the feed, and predicts their likelihood/energy in the framework of Gaussian Mixture Model (GMM).

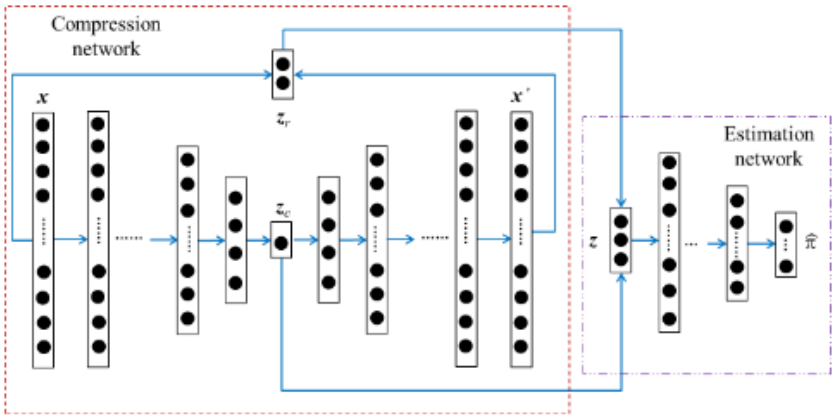

Figure 2.5: DAGMM's proposed model [14]

- The method consists of a compression network and an estimation network. The compression network is an autoencoder that reduces the dimensionality of the data, and the estimation network is a GMM that estimates the density of the compressed data.
- The method is trained using an end-to-end training strategy, where the parameters of both networks are optimized simultaneously.
- The method is evaluated on various datasets including KDDCUP, Thyroid, Arrhythmia, and KDDCUP-Rev.

In summary, the first article introduces a neural network model based on ODEs and uses adjoint sensitivity method for training, while the second article introduces a method for unsupervised anomaly detection that combines an autoencoder and a GMM and uses end-to-end training.

2.5 Other Methods

Zhang et al. present their algorithm named A Deep Neural Network (DNN) for Unsupervised Anomaly Detection and Diagnosis in Multivariate Time Series Data [15]. The paper proposes a Multi-Scale Convolutional Recurrent Encoder-Decoder (MSCRED) to detect and diagnose anomalies in multivariate time series data. The proposed model jointly considers temporal dependency, noise resistance, and the interpretation of severity of anomalies. The model constructs multi-scale signature matrices to characterize multiple levels of the system statuses across different time steps. A convolutional encoder is employed to encode the inter-sensor correlations patterns and an attention-based Convolutional Long-Short Term Memory (ConvLSTM) network is developed to capture the temporal patterns. Finally, with the feature maps which encode the inter-sensor correlations and temporal information, a convolutional decoder is used to reconstruct the signature matrices and the residual signature matrices are further utilized to detect and diagnose anomalies. See the DNN framework in Figure 2.6.

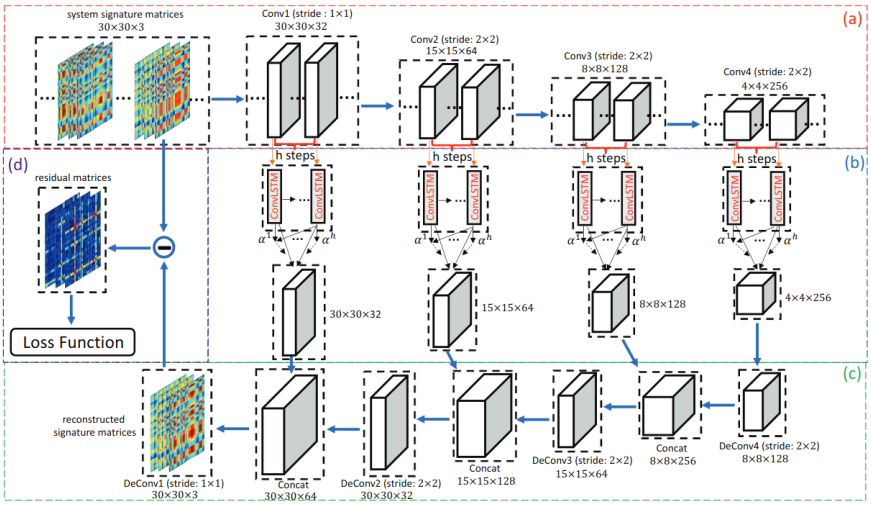

Figure 2.6: DNN's proposed model [15]

We detail the methodologies and strong points used by Zhang et al. in the following points:

- The model constructs multi-scale signature matrices to characterize multiple levels of the system statuses across different time steps.
- A convolutional encoder is employed to encode the inter-sensor correlations patterns and an attention-based Convolutional Long-Short Term Memory (ConvLSTM) network is developed to capture the temporal patterns.
- With the feature maps which encode the inter-sensor correlations and temporal information, a convolutional decoder is used to reconstruct the signature matrices and the residual signature matrices are further utilized to detect and diagnose anomalies.

- The proposed model jointly considers temporal dependency, noise resistance, and the interpretation of severity of anomalies.
- The model constructs multi-scale signature matrices to characterize multiple levels of the system statuses across different time steps.
- A convolutional encoder is employed to encode the inter-sensor correlations patterns and an attention-based Convolutional Long-Short Term Memory (ConvLSTM) network is developed to capture the temporal patterns.
- The model outperforms the baseline models in detecting and diagnosing anomalies in multivariate time series data.

Su et al. propose a novel approach called OmniAnomaly [16]. The approach uses a stochastic recurrent neural network to learn robust latent representations of normal patterns in multivariate time series data, considering both temporal dependence and stochasticity. The approach also includes an anomaly interpretation method to provide insights into the detected anomalies. The approach is evaluated on three datasets and compared with four state-of-the-art unsupervised approaches and a univariate time series anomaly detection approach (see Figure 2.7).

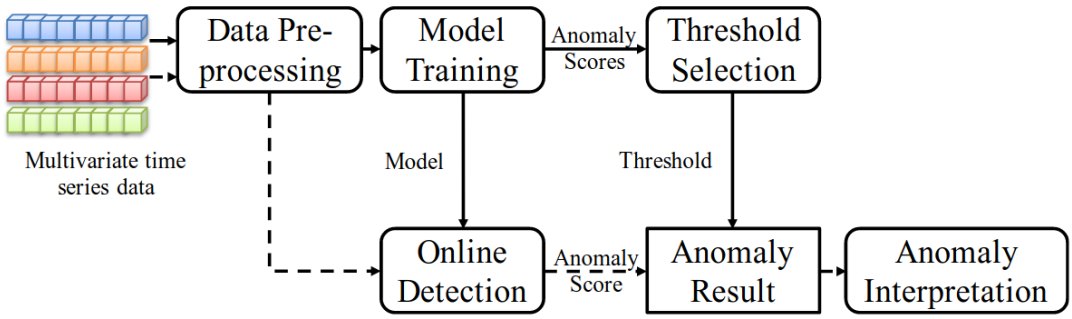

Figure 2.7: OmniAnomaly's proposed model [16]

We detail some key methodologies and advantages of OmniAnomaly in the points below:

- Stochastic recurrent neural network with explicit temporal dependence among stochastic variables.
- Stochastic variable connection technique: Linear Gaussian State Space Model.
- Planar Normalizing Flows to learn non-Gaussian posterior distributions in latent stochastic space.
- Anomaly interpretation approach based on reconstruction probabilities of individual dimensions of the detected anomaly.
- First multivariate time series anomaly detection algorithm that can deal with explicit temporal dependence among stochastic variables to learn robust representations of input data.
- Anomaly interpretation approach that works with not only OmniAnomaly, but also other algorithms.

- Great effect of the four key techniques in OmniAnomaly: GRU, planar NF, stochastic variable connection, and an adjusted Peaks-Over-Threshold method for automatic anomaly threshold selection.

Audibert et al. propose a new method for unsupervised anomaly detection in multivariate time series data called USAD (UnSupervised Anomaly Detection) [17]. The method is based on autoencoders and trained using an adversarial training approach inspired by Generative Adversarial Networks. The authors evaluate the performance of USAD on five public reference datasets and an internal dataset from Orange, a telecommunications company. The results show that USAD outperforms state-of-the-art techniques in terms of standard F1-score and is scalable for use in an industrial setting (see Figure 2.8).

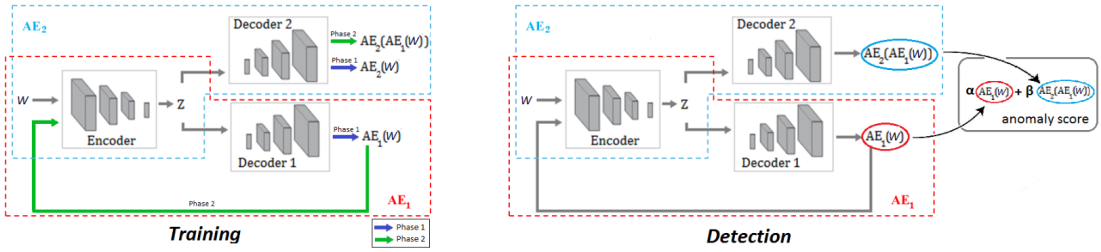

Figure 2.8: USAD's proposed model [17]

The points below highlight key aspects from USAD:

- USAD is an unsupervised anomaly detection method based on autoencoders and adversarial training.
- The method is evaluated on five public reference datasets and an internal dataset from Orange.
- Performance is assessed using precision, recall, and F1-score metrics.
- The method is scalable for use in an industrial setting.
- USAD provides the ability to parameterize its sensitivity and produce a set of detection levels, making it highly customizable for different use cases.
- The feasibility study on Orange's internal dataset showed promising results for the automation of IT systems supervision.

2.6 Transformers in Anomaly Detection

Transformer models, particularly Anomaly Transformer [1], have emerged as powerful tools for anomaly detection in time series data (see Figure 2.9). Their inherent self-attention mechanism is adept at capturing intricate temporal dependencies, which traditional methods often overlook. This capability allows them to model both pointwise representation and pairwise association, offering a comprehensive understanding of time series dynamics.

Central to the Anomaly Transformer is the concept of Association Discrepancy. This measures the difference between the actual association of a time point with the entire series and its expected association based on adjacent-concentration bias. Such a discrepancy becomes a distinguishable criterion, especially in an unsupervised setting, to identify

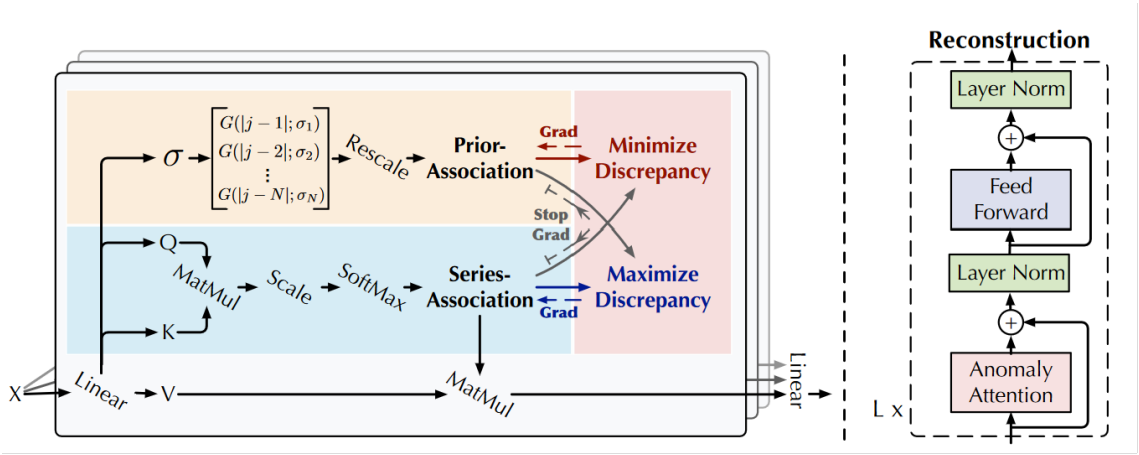

Figure 2.9: Anomaly Transformer's proposed model [1]

anomalies. The Anomaly Transformer leverages this by introducing a novel Anomaly-Attention mechanism, which computes the association discrepancy effectively.

In offline settings, the Anomaly Transformer has demonstrated state-of-the-art results across various benchmarks, spanning applications like service monitoring, space and earth exploration, and water treatment. While its offline performance is commendable, there exists potential for adaptation and enhancement in online scenarios. Recognizing these existing gaps and the potential of the Anomaly Transformer in real-time anomaly detection can pave the way for future advancements in the field. Our research aims to bridge this gap by modifying Anomaly Transformer to be applicable for real-time scenarios.

Chapter 3

RealTime-AT

The Anomaly Transformer has demonstrated strong performance in time series anomaly detection, extracting intricate multi-level features from data. However, its static nature does not cater efficiently to real-time data streams, where anomalies need to be detected on-the-fly without retraining on the entire historical data.

3.1 Anomaly Transformer's Architecture

The Anomaly Transformer employs a novel attention mechanism, termed Anomaly-Attention, tailored for time series anomaly detection. Unlike traditional self-attention mechanisms [3], which are incapable of concurrently modeling prior-associations and series-associations, the Anomaly-Attention uses a dual-branch structure:

1. Prior-Association Branch: It employs a trainable Gaussian kernel to capture relationships based on temporal proximity. The design inherently emphasizes nearby temporal relations, adapting to various time series patterns.
2. Series-Association Branch: It directly learns associations from raw time series data, adaptively determining the most pertinent relationships.

Both branches preserve temporal dependencies, capturing richer context than point-wise representations, which enables a clear distinction between normal and anomalous patterns.

A pivotal component of Anomaly Transformer is the Association Discrepancy. This metric, derived using symmetrized Kullback–Leibler divergence, quantifies the disparity between the prior and series associations. Such a divergence serves as a reliable indicator of anomalies.

The Association Discrepancy, as defined by the given equation, quantifies the difference between the prior and series associations of data points. For the Anomaly Transformer model to effectively discern anomalies, it relies on the principle that anomalies exhibit different association characteristics compared to regular data points. Therefore, this mathematical formulation (see Equation 3.1), which computes the divergence between these associations, is indispensable. It offers a quantifiable metric by which the AT can identify and classify anomalous behavior.

$$\text{AssDis}(\mathcal{P}, \mathcal{S}; \mathcal{X}) = \left[\frac{1}{L} \sum_{l=1}^{L} \left(\text{KL}\left(\mathcal{P}_{i,:}^{l} \| \mathcal{S}_{i,:}^{l} \right) + \text{KL}\left(\mathcal{S}_{i,:}^{l} \| \mathcal{P}_{i,:}^{l} \right) \right) \right]_{i=1,\cdots,N} \tag{3.1}$$

Where:

- KL represents the Kullback-Leibler (KL) divergence, a measure of the difference between two probability distributions. Specifically, $\mathrm{KL}(\cdot\|\cdot)$ computes the KL divergence between two discrete distributions corresponding to each row of $\mathcal{P}$ and $\mathcal{S}$.
- $\mathrm{AssDis}(\mathcal{P}, \mathcal{S}; \mathcal{X})$ is the point-wise association discrepancy of $\mathcal{X}$ with respect to the prior-association $\mathcal{P}$ and the series-association $\mathcal{S}$ from multiple layers. It results in a matrix of size $N \times 1$, where N is the number of time points.
- i represents an index corresponding to a specific time point in $\mathcal{X}$. For instance, the i-th element of $\mathrm{AssDis}(\mathcal{P}, \mathcal{S}; \mathcal{X})$ corresponds to the i-th time point of $\mathcal{X}$.

From observations, anomalies tend to have smaller $\mathrm{AssDis}(\mathcal{P}, \mathcal{S}; \mathcal{X})$ values compared to normal time points, making this measure useful for distinguishing them.

Minimax Association Learning, also a necessary component of Anomaly Transformer, utilizes a two-phase iterative learning approach designed to emphasize distinctions between regular and anomalous data points within time series datasets. By integrating a unique discrepancy loss alongside a reconstruction loss, this methodology accentuates the contrasts between the typical and the exceptional.

The comprehensive loss function, $\mathcal{L}_{\text{Total}}$, which underpins the minimax learning process, is formulated as in Equation 3.2.

$$\mathcal{L}_{\text{Total}}\left(\widehat{\mathcal{X}}, \mathcal{P}, \mathcal{S}, \lambda; \mathcal{X}\right) = \|\mathcal{X} - \widehat{\mathcal{X}}\|_{\text{F}}^2 - \lambda \times \|\,\mathrm{AssDis}(\mathcal{P}, \mathcal{S}; \mathcal{X})\|_1 \tag{3.2}$$

Where:

- $\widehat{\mathcal{X}}$ symbolizes the reconstruction of $\mathcal{X}$.
- $\|\cdot\|_{\text{F}}$ and $\|\cdot\|_1$ stand for the Frobenius and l_1-norms, respectively.
- λ acts as a balancing factor, ensuring an equilibrium between reconstruction and association discrepancy.

The dual phases integral to this strategy comprise:

1. Minimize Phase: The aspiration here is to synchronize the prior-association $\mathcal{P}$ with the series-association $\mathcal{S}$. This is mathematically represented in Equation 3.3 shows.

$$\mathcal{L}_{\text{Total}}\left(\widehat{\mathcal{X}}, \mathcal{P}, \mathcal{S}_{\text{detach}}, -\lambda; \mathcal{X}\right) \tag{3.3}$$

2. Maximize Phase: The intent in this phase is to accentuate disparities in the series-association, particularly spotlighting anomalies (see Equation 3.4).

$$\mathcal{L}_{\text{Total}}\left(\widehat{\mathcal{X}}, \mathcal{P}_{\text{detach}}, \mathcal{S}, \lambda; \mathcal{X}\right) \tag{3.4}$$

As illustrated in Figure 3.1, the dual-phase learning structure ensures a sophisticated attention mechanism for time series analysis. In the minimize phase, the prior-association adjusts to prevalent temporal patterns. Meanwhile, the maximize phase encourages the series-association to underscore non-adjacent patterns. This strategic differentiation makes

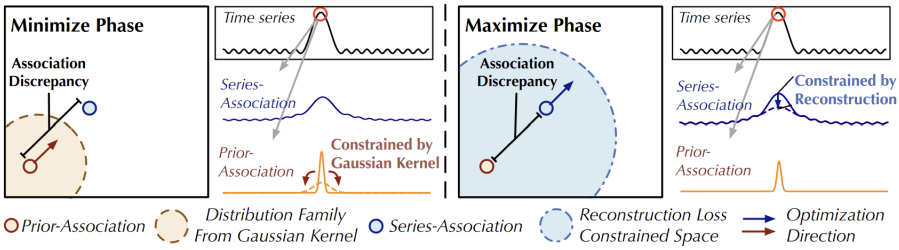

Figure 3.1: Minimax Association Learning [1]

anomalies more conspicuous, aiding their detection. Consequently, the association discrepancy emerges as a robust tool, allowing for effective distinction between regular and anomalous time points.

To robustly identify anomalies in time series data, it's crucial to harness the strengths of both temporal representation and association discrepancy. The proposed association-based anomaly criterion fuses these two metrics, ensuring that even subtle anomalies—ones that might decrease the association discrepancy—are distinctly captured.

Formally, the anomaly score for a given time series $\mathcal{X} \in \mathbb{R}^{N \times d}$ is defined in Equation 3.5.

$$\text{AnomalyScore}(\mathcal{X}) = \text{Softmax}\left(-\text{AssDis}(\mathcal{P}, \mathcal{S}; \mathcal{X})\right) \odot \left[\left\|\mathcal{X}_{i,:} - \widehat{\mathcal{X}}_{i,:}\right\|_2^2\right]_{i=1,\cdots,N} \tag{3.5}$$

Where $\odot$ represents the element-wise multiplication.

This anomaly score offers a point-wise anomaly criterion for $\mathcal{X}$. By fusing the reconstruction error with the association discrepancy, the model utilizes both metrics' advantages, potentially elevating anomaly detection performance.

3.2 RealTime-AT's Architecture

In dynamic environments where data streams continuously, the need for immediate anomaly detection becomes paramount. The original Anomaly Transformer, being offline, has the following limitations:

- Scalability Issues: For large datasets, repeatedly retraining the model becomes computationally expensive.
- Latency: Offline models cannot provide immediate feedback as new data arrives.

In response to the dynamic challenges of real-time data streams, we introduce RealTime-AT that embeds online learning principles into the traditional anomaly detection model. Our methodology preserves the foundational architecture of the Anomaly Transformer, but with substantial enhancements, notably in the Anomaly-Attention mechanism. The model's central features are the following:

- The model is designed for continuous adaptation and incremental learning. Unlike offline learning, where models are trained on the entire dataset, incremental learning focuses on updating the model's parameters with each new data entry or a mini-batch. This approach offers several advantages. Firstly, by learning incrementally,

the model eliminates the need to store and revisit the entire historical data, making it memory efficient. Moreover, as data dynamics shift over time, incremental learning ensures the model remains relevant and up-to-date, thereby retaining its predictive accuracy in real-time scenarios. Lastly, given that only a segment of data is processed at any given time, the model can effortlessly scale to accommodate larger streams of incoming data.

- The introduced sliding window approach emphasizes the most recent data points by defining a window or subset of the incoming data stream. By constraining the model's focus to a subset of recent data, we substantially reduce the computational burden, ensuring faster processing speeds. The sliding window inherently prioritizes recent events, making the model attuned to current data trends, which is particularly crucial for anomaly detection in real-time scenarios. The window size is adjustable. While a larger window captures more historical context, a smaller window ensures quicker adaptability to recent changes.

Incorporating these features, the Anomaly-Attention mechanism within RealTime-AT is specifically optimized to simultaneously achieve continuous data processing.

3.3 Implementation of Sliding Window Approach

One of the pivotal enhancements in the RealTime-AT algorithm is its architectural shift to address the intricate challenges posed by real-time data streams. To effectively manage and harness the incessant influx of data, we used a sophisticated dual-buffer mechanism, fundamentally rooted in the principles of circular buffers. This mechanism operates by dividing incoming data into fixed windows of 100 data points each. As one buffer deque (short for double-ended queue) processes a window of data through the anomaly detection algorithm, its counterpart concurrently accumulates the next 100 data points. This alternating procedure not only ensures that data ingestion is seamless but also facilitates uninterrupted algorithmic processing. The choice of a 100-data point window size serves as a balanced trade-off between computational agility and granularity of detection, ensuring that the algorithm remains both responsive and accurate in dynamic environments.

At its core, the dual-buffer mechanism consists of two primary buffers: the Active Buffer and the Processing Buffer. The data stream populates the Active Buffer in real-time. Once this buffer fills up to a predefined window size, its contents switch to the Processing Buffer, making room for the next set of data points in the Active Buffer. This switching ensures that while one buffer is being populated, the other is simultaneously processed, ensuring no latency in data handling.

Additionally, to ensure data integrity and coherence, we introduce two auxiliary label buffers corresponding to each primary buffer. These label buffers store associated metadata or labels for each data point, ensuring synchronized processing.

Benefits of the dual-buffer approach include the following:

- Continuous Data Ingestion: The algorithm doesn't need to pause or wait. Data ingestion is uninterrupted, matching the real-time requirements.
- Parallel Processing: While one buffer ingests data, the other processes it, maximizing computational efficiency.

- Memory Efficiency: With a fixed window size, we prevent excessive memory usage, ensuring that the algorithm remains scalable and resource-efficient.
- Latency Reduction: Immediate switching and parallel processing eliminate the delay between data ingestion and processing, making real-time anomaly detection a reality.

This dual-buffer mechanism is a key component of RealTime-AT, ensuring that the algorithm not only keeps pace with real-time data streams but also processes them efficiently, laying the foundation for subsequent analysis modules.

In Figure 3.2, the dual-buffer mechanism of RealTime-AT is visually represented with circular buffers. The choice of circular buffers aptly symbolizes the continuous and cyclic nature of the data processing within our proposed algorithm.

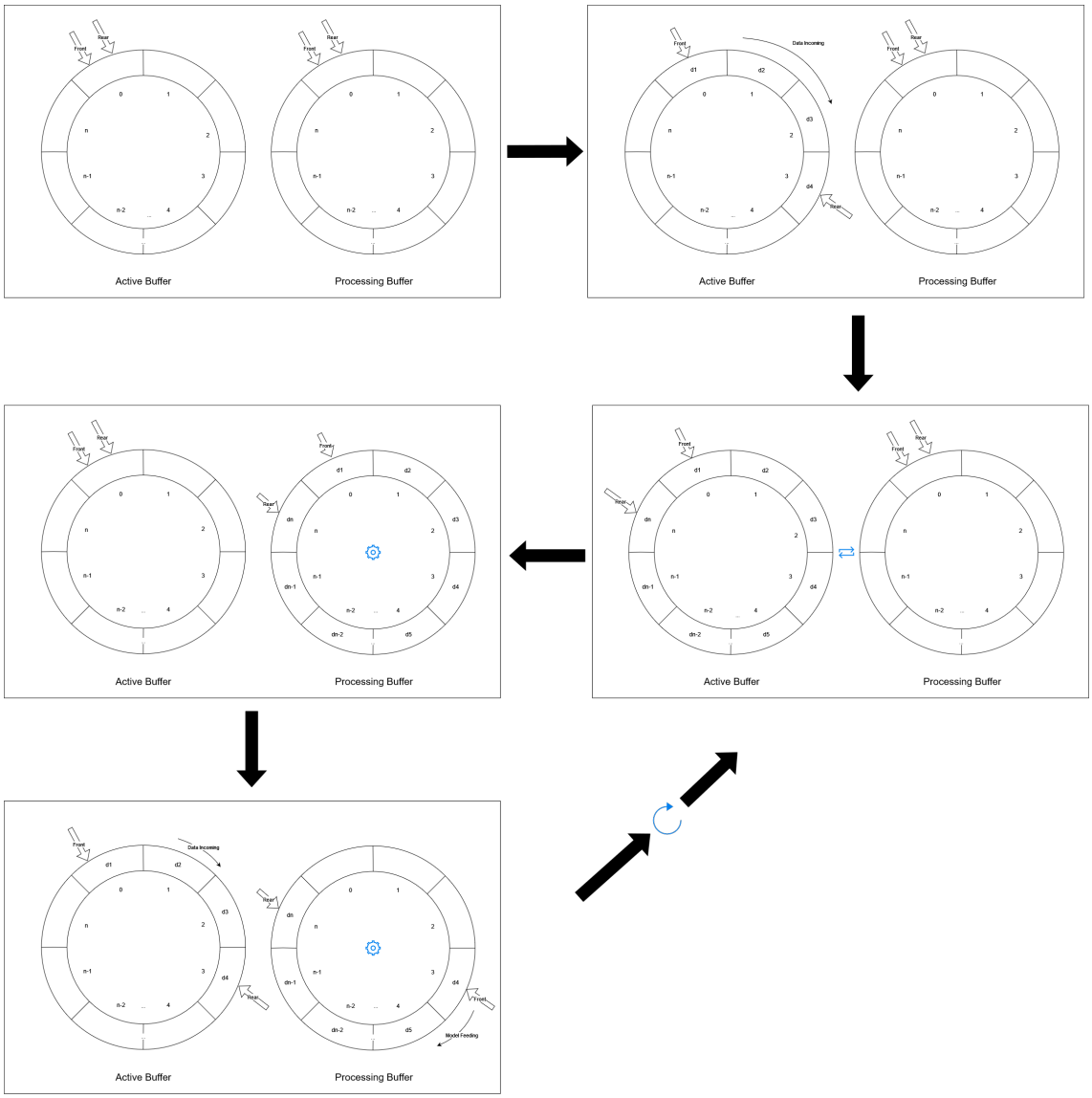

Figure 3.2: RealTime-AT's dual-buffer mechanism

Each buffer is divided into segments, with each segment representing a data point $(d_1, \ldots, d_n)$ or a mini-batch of data points. The total number of segments, denoted by n, signifies the window size of the buffer. In essence, the window size n dictates how many

of the most recent data points (or mini-batches) the buffer can accommodate at any given time.

As new data arrives, it is appended to the active buffer. When the active buffer reaches its capacity, the algorithm switches its focus, making the previously passive buffer the new active buffer. This allows the system to process data in the filled buffer while simultaneously collecting new data in the other, ensuring seamless and uninterrupted data processing.

The circular nature of these buffers ensures that the oldest data point is automatically replaced by the newest data point once the buffer is full. This represents the sliding window mechanism integral to online algorithms, especially in scenarios where it is impractical or inefficient to consider the entire data history.

Through this dual circular buffer system, RealTime-AT achieves a balance between continuous data ingestion and real-time processing, addressing both the scalability concerns of large data streams and the need for immediate feedback.

The introduction of RealTime-AT underscores a significant shift towards catering to the challenges of real-time data processing. It has many advantages. Firstly, real-time detection means the model provides immediate insights as soon as data is ingested. Thus, stakeholders are alerted without delay, allowing for prompt remedial actions. This also allows for enhanced responsiveness, as the quick turnaround ensures that systems remain agile, adapting to new information and ensuring minimal latency between data arrival and insight generation.

Another improvement, scalability results in better adaptability to data volume. Regardless of the volume of incoming data, the architecture can scale seamlessly, accommodating large influxes without compromising on performance.

Memory efficiency is also strongly reduced. By focusing on the most recent data through the sliding window approach, there's a considerable reduction in the memory footprint, ensuring optimal use of resources. Minimizing memory usage translates to savings, especially in scenarios where storage costs can escalate with larger datasets.

These advantages coalesce to make RealTime-AT an ideal choice for dynamic and demanding environments. In practical terms, industries that rely on real-time monitoring, such as finance, healthcare, and manufacturing, stand to benefit immensely. The ability to detect anomalies instantaneously can translate to significant cost savings, enhanced system reliability, and improved decision-making. In essence, by embedding online learning principles, the Anomaly Transformer is not just an algorithmic enhancement but a major step towards proactive and efficient data analytics.

Chapter 4

Experiments

In this chapter, we detail the empirical evaluations designed to measure the performance of RealTime-AT. We describe the experimental setup, datasets used, evaluation metrics, and the obtained results.

4.1 Experimental Setup

Hardware specifications: Experiments were conducted on a computing environment powered by an AMD Ryzen 9 7940HS processor coupled with Radeon 780M Graphics, clocked at 4.00 GHz. The system was equipped with 16.0 GB RAM (15.2 GB usable) and was based on a 64-bit operating system with an x64-based processor architecture.

Software and libraries: For software dependencies, our experimental setup heavily relied on a range of Python libraries. These include: numpy, Pillow, scikit-learn, scipy, tensorflow (GPU accelerated, leveraging Nvidia CUDA), pandas, websockets, asyncio, torch (utilizing the cu118 variant for optimized CUDA support), torchvision, torchaudio [18].

WebSocket API setup: To facilitate real-time data acquisition and processing, we established a WebSocket API using FastAPI [19]. The choice of FastAPI, a modern, fast (high-performance) web framework, ensured that the API was scalable, reliable, and efficient in handling concurrent data streams.

Client-Side Setup: The WebSocket client we developed interfaces seamlessly with the FastAPI server and incorporates the following features:

- Initialization: Upon being provided the server's WebSocket URI, the client employs an internal asyncio queue and event flags. These tools ensure efficient data storage and constant monitoring of the connection state.
- Data Handling: At its core, the client is designed to retrieve and deserialize data swiftly from the WebSocket connection. This mechanism ensures data is primed for subsequent real-time processing.
- Robust Connection Management: Our client has an innate resilience against potential disruptions. If disconnections or server errors emerge, it will attempt reconnections after short intervals to ensure consistent data flow.
- Asynchronous Data Retrieval: The client is equipped with asynchronous methods, facilitating efficient extraction of data from the internal queue, in line with the real-time processing requirements.

- Active Connection Maintenance: To preemptively address potential server timeouts and sustain an uninterrupted link, our client sends periodic ping messages to the server. This approach ensures the connection remains lively and responsive.

Server-Side Setup: We developed a FastAPI-based server to seamlessly interface with the WebSocket client, emphasizing robustness, scalability, and real-time data streaming for the RealTime-AT pipeline. This server caters to multiple clients concurrently, demonstrating the scalability of our architecture.

- CORS Configuration: Ensured smooth client-server interactions with Cross-Origin Resource Sharing (CORS) setup, especially beneficial during the developmental phase.
- Data Preparation: The dataset was preloaded and primed for real-time streaming.
- WebSocket Functionality: Introduced a WebSocket endpoint (‘/ws‘) to stream data points and their corresponding labels. A deliberate delay, emulating real-world data streams, was factored between consecutive data points.
- Robustness Measures: Incorporated provisions to gracefully manage unforeseen events like client disconnections and potential exceptions.
- Server Accessibility: Ensured universal server execution, facilitating access from diverse IP addresses. Configured WebSocket settings to avert unwarranted disconnections.

4.2 Dataset Description

The primary dataset employed for this research is the Server Machine Dataset (SMD) [16], published by Su et al. 2019. Sourced from a renowned Internet company, the SMD covers a timespan of 5 weeks. It classifies entities into three distinct categories, each labeled as ‘machine-<group-index>-<index>‘.

A distinguishing characteristic of the SMD is its data derived from 28 separate machines. It is imperative to recognize that each of these subsets should be trained and tested separately. To maintain consistency in evaluations, each subset is evenly divided into training and testing segments.

In our methodology, the training set served an offline role to train the RealTime-AT model. Post this initial phase, we transitioned to an online operational mode, processing data from the test set in real-time. This approach effectively combines offline model preparation with online anomaly detection, capitalizing on the strengths of both strategies.

To provide a clearer perspective, the SMD is comprised of the following components:

- **train:** This encompasses the initial half of the dataset designated for training.
- **test:** This forms the latter half, reserved for testing purposes.
- **test_label:** Accompanying the test dataset, this label indicates whether a given data point is an anomaly.
- **interpretation_label:** It provides a list of dimensions that contribute to each identified anomaly, offering a deeper understanding of the data’s intricacies.

Given the structured and comprehensive nature of SMD, it serves as an apt dataset for evaluating RealTime-AT, especially in the realm of anomaly detection.

4.3 Results

To assess the performance of RealTime-AT, we focused on crucial evaluation metrics that measure the efficiency and resource usage of the algorithm, especially when dealing with large datasets and real-time processing. The key metrics employed are:

- Anomaly detection metrics: Accuracy, Precision, Recall, F-score.
- RAM utilization: It assesses the amount of system memory used by the algorithm during its operation.
- GPU memory utilization: It focuses on the memory allocation on the GPU, emphasizing the effective use of Nvidia CUDA capabilities.

Figure 4.1 shows the comparison between the Original Anomaly Transformer and RealTime-AT in regard to anomaly detection metrics. RealTime-AT demonstrates comparable performance to that of the original Anomaly Transformer in the metrics of Accuracy, Precision, Recall and F-score. Note that the RealTime-AT is optimized for online processing, exhibiting a reduced computational footprint. This adaptation to an online context is a significant advantage for real-time applications where resource efficiency is paramount. Despite the marginal decrease in performance metrics, the RealTime-AT remains a robust solution for immediate anomaly detection tasks.

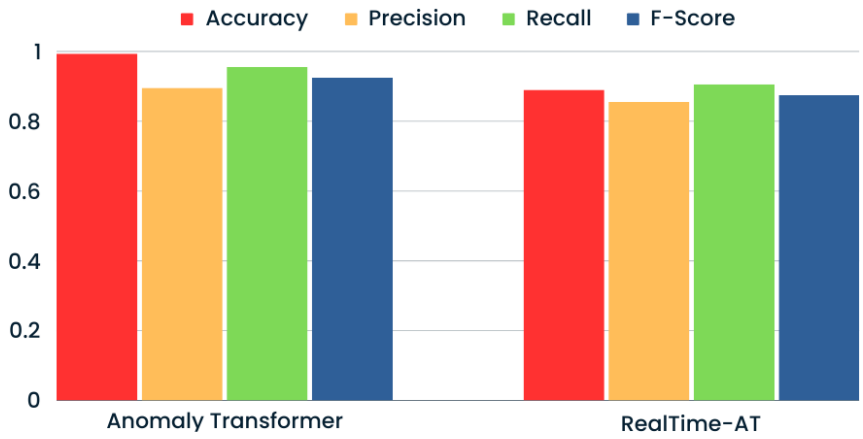

Figure 4.1: Performance comparison between the original Anomaly Transformer and RealTime-AT in regard to anomaly detection metrics

Figure 4.2 provides the results of our resource utilization experiments.

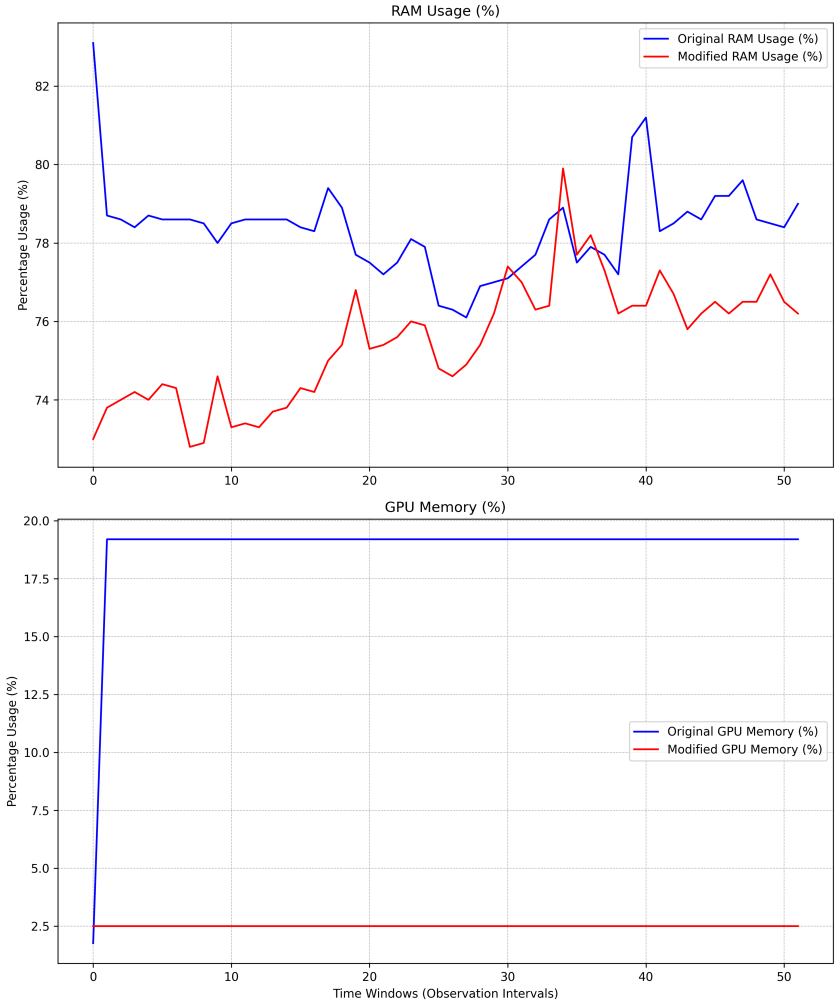

Figure 4.2: Resource utilization comparison between the original Anomaly Transformer and RealTime-AT

The figure elucidates several key observations:

- GPU Memory: RealTime-AT demonstrated a noticeable (approx. eight times) decrease in GPU memory consumption relative to its predecessor. This stands as testament to the advancements we have made in GPU memory optimization.
- RAM Utilization: There was a discernible, albeit modest, reduction in RAM consumption with the introduction of RealTime-AT. In real-time systems, even marginal RAM savings can accumulate to offer significant performance advantages.
- CPU and GPU Peaks: The observed increments in CPU and GPU utilization are likely artifacts of concurrent server operations during our experiments.

From these empirical findings, we can deduce the following:

- RealTime-AT is highly amenable to real-time operations, thanks to its efficient resource management.
- The diminished memory demands, both in terms of GPU and RAM, render it feasible to run multiple algorithm instances concurrently or to achieve comparable results with lesser memory allocations.

- The resilience of performance metrics in the face of reduced resource consumption underscores the potential of RealTime-AT in telemetry and related domains.

As shown in Table 4.1 below, RealTime-AT demonstrates improved efficiency in terms of both RAM and GPU memory usage when compared to the original algorithm. As we see again, the average and peak RAM and GPU memory utilization were substantially decreased in RealTime-AT, which is essential when we use anomaly detection in a resource-constrained real-time environment.

Table 4.1: Comparison of Anomaly Transformer's and RealTime-AT's resource usage

Metric	Anomaly Transformer	RealTime-AT
Average RAM Usage (%)	78.36	75.50
Peak RAM Usage (%)	83.10	79.90
Average GPU Memory Usage (%)	18.86	2.49
Peak GPU Memory Usage (%)	19.20	2.49
Cumulative RAM Usage Score	4074.90	3926.10
Cumulative GPU Memory Usage Score	980.96	129.48

Drawing from the aforementioned observations, our primary conclusions include:

- RealTime-AT exhibits a propensity for efficient resource management, making it particularly well-suited for real-time applications.
- The reduced memory requirements suggest potential benefits for telemetry systems, such as running multiple algorithm instances concurrently or achieving comparable performance with reduced memory allocations.
- The performance stability observed, even with lower resource consumption, accentuates RealTime-AT's promise for diverse applications beyond the realms of this study.

The implementation of a deque-based buffering system facilitated uninterrupted data processing. One buffer deque was employed for the processing of data via the anomaly detection algorithm, while another was tasked with concurrently accumulating the succeeding 100 data points. Such a circular buffer interchange guaranteed uninterrupted algorithmic operations, eliminating substantial idle periods. This mechanism not only bolstered real-time processing capabilities but also emphasized the scalability and adaptability of RealTime-AT in dynamic data environments. Nevertheless, this segmented processing approach did yield some variations in accuracy metrics, evident in the presented results.

When juxtaposed against conventional batch processing techniques, RealTime-AT presented a pronounced reduction in computational latency, largely attributed to the innovative deque-based buffering strategy. However, the segmentation of data into fixed 100-point chunks might inherently introduce discrepancies in anomaly detection compared to a holistic dataset processing approach. The predetermined chunk size, in this instance 100, may inherently influence the fidelity of anomaly detection, and a change in this parameter could potentially yield different outcomes. Additionally, we observed heightened CPU and GPU utilization, potentially due to server processes running concurrently during our experiments. This observation warrants further experimentation, which we intend to pursue to ascertain the cause and implications.

Chapter 5

Conclusions

In this final chapter, we synthesize our findings and contributions stemming from this research, emphasizing the significance and innovations introduced by the optimized RealTime-AT algorithm. The milestones achieved in online data processing are underscored, accompanied by a visionary perspective on prospective advancements in this domain.

Our investigation was initiated with the aspiration to advance the state-of-the-art in anomaly detection. To this end, we introduced RealTime-AT to address and overcome the limitations of the existing Anomaly Transformer. RealTime-AT is characterized by its dynamic data ingestion capabilities. The empirical evaluations we conducted subsequently revealed its advantages in terms of efficient resource utilization. These findings reinforced the foundational motivations that spurred the development of RealTime-AT and accentuated its potential for wide-scale adoption in pertinent domains.

As our contributions, we carried out a comprehensive analysis of the foundational Anomaly Transformer, shedding light on its limitations and areas that posed challenges in real-time operations. Furthermore, we introduced RealTime-AT, our new algorithm with dynamic data ingestion and swift real-time analytics.

While RealTime-AT delineates a significant leap in real-time anomaly detection, there remains a plethora of promising avenues ripe for exploration:

1. Delving into alternative computational architectures that could potentially improve the latency of anomaly detection.
2. Introducing cutting-edge artificial intelligence and machine learning techniques to imbue the algorithm with enhanced adaptability to evolving data patterns.
3. Examining potential scalability challenges, particularly when confronted with massive data streams.
4. Engaging in collaborations with industry leaders to ensure that the algorithm aligns seamlessly with current industrial requirements.

This research represents a meaningful step forward in the realm of real-time anomaly detection. Our RealTime-AT highlights the importance of continuous refinements and improvements. As the dynamics of online data processing evolve, the findings and methodologies presented here offer valuable perspectives, potentially serving as a foundation for future academic endeavors.

Acknowledgements

I wish to express my profound gratitude to my advisor, dr. Károly Farkas for his enduring mentorship and support throughout both the conception and realization of this research.

Similarly, my sincere thanks go to Dániel Vajda for his invaluable input and unwavering encouragement that greatly contributed to this work.

List of Figures

List of Tables

Bibliography

[1] Jiehui Xu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Anomaly transformer: Time series anomaly detection with association discrepancy. *arXiv preprint arXiv:2110.02642*, 2022.

[2] Guansong Pang, Anton Van Den HENGEL, and Chuanhua SHEN. Weakly-supervised deep anomaly detection with pairwise relation learning. 2019.

[3] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

[4] Ailin Deng and Bryan Hooi. Graph neural network-based anomaly detection in multivariate time series. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(5):4027–4035, May 2021. DOI: 10.1609/aaai.v35i5.16523. URL https://ojs.aaai.org/index.php/AAAI/article/view/16523.

[5] Hang Zhao, Yujing Wang, Juanyong Duan, Congrui Huang, Defu Cao, Yunhai Tong, Bixiong Xu, Jing Bai, Jie Tong, and Qi Zhang. Multivariate time-series anomaly detection via graph attention network. In *2020 IEEE International Conference on Data Mining (ICDM)*, pages 841–850, 2020. DOI: 10.1109/ICDM50108.2020.00093.

[6] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[7] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[8] Dan Li, Dacheng Chen, Jonathan Goh, and See kiong Ng. Anomaly detection with generative adversarial networks for multivariate time series, 2019.

[9] Riccardo Taormina, Stefano Galelli, Nils Ole Tippenhauer, Elad Salomons, Avi Ostfeld, Demetrios G. Eliades, Mohsen Aghashahi, Raanju Sundararajan, Mohsen Pourahmadi, M. Katherine Banks, B. M. Brentan, M. Herrera, Amin Rasekh, Enrique Campbell, I. Montalvo, G. Lima, J. Izquierdo, Kelsey Haddad, Nikolaos Gatsis, Ahmad Taha, Saravanakumar Lakshmanan Somasundaram, D. Ayala-Cabrera, Sarin E. Chandy, Bruce Campbell, Pratim Biswas, Cynthia S. Lo, D. Manzi, E. Luvizotto, Jr, Zachary A. Barker, Marcio Giacomoni, M. Fayzul K. Pasha, M. Ehsan Shafiee, Ahmed A. Abokifa, Mashor Housh, Bijay Kc, and Ziv Ohar. The battle of the attack detection algorithms: Disclosing cyber attacks on water distribution networks. *Journal of Water Resources Planning and Management*, 144(8):04018048, August 2018. DOI: 10.1061/(ASCE)WR.1943-5452.0000969.

[10] Dan Li, Dacheng Chen, Baihong Jin, Lei Shi, Jonathan Goh, and See-Kiong Ng. Madgan: Multivariate anomaly detection for time series data with generative adversarial

networks. In Igor V. Tetko, Věra Kůrková, Pavel Karpov, and Fabian Theis, editors, *Artificial Neural Networks and Machine Learning – ICANN 2019: Text and Time Series*, pages 703–716, Cham, 2019. Springer International Publishing. ISBN 978-3-030-30490-4.

[11] John Sipple. Interpretable, multidimensional, multimodal anomaly detection with negative sampling for detection of device failure. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9016–9025. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/sipple20a.html.

[12] D. Dasgupta and N.S. Majumdar. Anomaly detection in multidimensional data using negative selection algorithm. In *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, volume 2, pages 1039–1044 vol.2, 2002. DOI: 10.1109/CEC.2002.1004386.

[13] Yifan Guo, Weixian Liao, Qianlong Wang, Lixing Yu, Tianxi Ji, and Pan Li. Multidimensional time series anomaly detection: A gru-based gaussian mixture variational autoencoder approach. In Jun Zhu and Ichiro Takeuchi, editors, *Proceedings of The 10th Asian Conference on Machine Learning*, volume 95 of *Proceedings of Machine Learning Research*, pages 97–112. PMLR, 14–16 Nov 2018. URL https://proceedings.mlr.press/v95/guo18a.html.

[14] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BJJLHbb0-.

[15] Chuxu Zhang, Dongjin Song, Yuncong Chen, Xinyang Feng, Cristian Lumezanu, Wei Cheng, Jingchao Ni, Bo Zong, Haifeng Chen, and Nitesh V. Chawla. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33 (01):1409–1416, Jul. 2019. DOI: 10.1609/aaai.v33i01.33011409. URL https://ojs.aaai.org/index.php/AAAI/article/view/3942.

[16] 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 amp; Data Mining*, KDD '19, page 2828–2837, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450362016. DOI: 10.1145/3292500.3330672. URL https://doi.org/10.1145/3292500.3330672.

[17] 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 amp; Data Mining*, KDD '20, page 3395–3404, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379984. DOI: 10.1145/3394486.3403392. URL https://doi.org/10.1145/3394486.3403392.

[18] Python Software Foundation. Python package index, 2023. URL https://pypi.org/.

[19] Tiangolo. Fastapi, 2018. URL https://fastapi.tiangolo.com/.