





Article

# A Cross-Modal Dynamic Attention Neural Architecture to Detect Anomalies in Data Streams from Smart Communication Environments

Konstantinos Demertzis <sup>1,\*</sup> , Konstantinos Rantos <sup>1</sup> , Lykourgos Magafas <sup>2</sup>  and Lazaros Iliadis <sup>3</sup> <sup>1</sup> Department of Computer Science, School of Science, International Hellenic University, 65404 Kavala, Greece<sup>2</sup> Department of Physics, School of Science, Kavala Campus, International Hellenic University, 65404 Kavala, Greece<sup>3</sup> Department of Civil Engineering, School of Engineering, Democritus University of Thrace, 67100 Xanthi, Greece; liliadis@civil.duth.gr

\* Correspondence: kdemertzis@emt.ihu.gr

**Abstract:** Detecting anomalies in data streams from smart communication environments is a challenging problem that can benefit from novel learning techniques. The Attention Mechanism is a very promising architecture for addressing this problem. It allows the model to focus on specific parts of the input data when processing it, improving its ability to understand the meaning of specific parts in context and make more accurate predictions. This paper presents a Cross-Modal Dynamic Attention Neural Architecture (CM-DANA) by expanding on state-of-the-art techniques. It is a novel dynamic attention mechanism that can be trained end-to-end along with the rest of the model using multimodal data streams. The attention mechanism calculates attention weights for each position in the input data based on the model's current state by a hybrid method called Cross-Modal Attention. Specifically, the proposed model uses multimodal learning tasks where the input data comes from different cyber modalities. It combines the relevant input data using these weights to produce an attention vector in order to detect suspicious abnormal behavior. We demonstrate the effectiveness of our approach on a cyber security anomalies detection task using multiple data streams from smart communication environments.

**Keywords:** cross-modal learning tasks; dynamic attention mechanism; neural architecture; anomaly detection; data streams; smart communication environments



**Citation:** Demertzis, K.; Rantos, K.; Magafas, L.; Iliadis, L. A Cross-Modal Dynamic Attention Neural Architecture to Detect Anomalies in Data Streams from Smart Communication Environments. *Appl. Sci.* **2023**, *13*, 9648. <https://doi.org/10.3390/app13179648>

Academic Editors: Peter R.J. Trim and Yang-Im Lee

Received: 1 August 2023

Revised: 24 August 2023

Accepted: 24 August 2023

Published: 25 August 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Detecting anomalies in data streams [1] from smart communication environments is a critical problem that has significant implications for various applications, including cyber security [2], monitoring cyber-physical systems [3], and controlling the industrial ecosystem [4]. The vast amount of data generated in these environments makes it difficult to detect abnormal behavior in real-time, which can lead to significant damages and security breaches [5]. Anomaly detection in these data streams is challenging due to the volume and complexity of the data and the need for real-time detection to prevent potential damages or security breaches [6,7]. Traditional methods for anomaly detection in data streams rely on statistical techniques or rule-based systems, which may not be effective in identifying subtle or unknown anomalies [8]. Machine learning approaches, particularly deep learning methods, have shown promise in addressing this challenge by enabling automated and accurate detection of anomalies in complex data streams [9].

One of the key advantages of deep learning methods for anomaly detection is the ability to learn relevant features from the input data without relying on pre-defined rules or assumptions. Attention mechanisms, in particular, have emerged as a powerful tool for capturing relevant input data features and improving neural network performance in

various applications [2]. Recent research has focused on developing novel deep-learning architectures that effectively leverage attention mechanisms to detect anomalies in data streams from smart communication environments. These architectures often use simple attention mechanisms that can adapt to changes in the input data over time and can be trained end-to-end using data streams to capture the complex interactions between sophisticated processes [10,11].

Simple attention involves computing a fixed set of attention weights for the input data learned during training based on the task-specific objective function. The network then uses these fixed attention weights to weigh the input features in subsequent neural network layers. These simple attention mechanisms have become a powerful tool for capturing relevant input data features and improving neural network performance in various applications [12].

On the other hand, dynamic attention allows the network to adjust the attention weights at each time step to give more or less importance to different parts of the input sequence depending on their relevance to the task. Dynamic attention mechanisms can be useful in applications where the types and frequencies of anomalies may change over time, allowing the model to adapt to changes in the input data [13].

Both simple and dynamic attention mechanisms have strengths and weaknesses depending on the specific application and data. Simple attention is more straightforward and can be effective in many cases. In contrast, dynamic attention can improve the model's ability to adapt to changes in the input data over time. The appropriate attention mechanism type depends on the input data's nature and task [14,15].

This paper presents a novel and holistic neural architecture called CM-DANA for detecting anomalies in data streams from smart communication environments. The model is based on a hybrid approach that combines attention mechanisms and multimodal learning techniques to capture the complex interactions between different modalities of data effectively. The CM-DANA model uses a dynamic attention mechanism that calculates attention weights for each position in the input data based on the model's current state. This attention mechanism is a location-based attention mechanism that uses the position of the input features in the sequence of real-time data streams to calculate the attention weights. The more sophisticated character of the proposed model is that it is trained end-to-end using multimodal data streams. This allows the model to attend to different features in different modalities based on the model's current state and detect suspicious abnormal behavior by combining the relevant input data from different modalities using adaptive attention weights.

The motivation for the CM-DANA model is to improve the accuracy and efficiency of anomaly detection in data streams from smart communication environments by effectively capturing relevant features and suppressing noisy or irrelevant features. The use of dynamic attention and multimodal learning techniques allows the model to attend to different features in different modalities based on the model's current state, which can improve its ability to detect suspicious abnormal behavior in real-time. Overall, the motivation for the paper is to develop a novel deep-learning architecture that can effectively detect anomalies in data streams from smart communication environments. By leveraging attention mechanisms and multimodal learning techniques, the CM-DANA model, presented for the first time in the literature, aims to be a promising approach to improving the accuracy and efficiency of anomaly detection in various applications.

## 2. Literature Review

Anomaly detection in data streams has been an active research area due to the increasing volume and complexity of data generated by IoT devices and smart environments [2]. Traditional anomaly detection methods, such as statistical techniques [5], clustering [16], and classification [8], have been applied to data streams [6], with varying degrees of success. However, they often struggle to adapt to the dynamic nature of data streams, which may have changing distributions and evolving patterns [5]. For example, during a timed event,

the traffic pattern can change dramatically, potentially causing statistical methods that rely on historical data to label the surge in traffic as an anomaly due to the shift in statistical properties like mean and variance [17]. In addition, the traditional clustering methods might not recognize the sudden appearance of a new cluster as an anomaly, leading to delayed detection, or traditional classifiers might struggle to identify novel patterns that were not present in the training data [6]. In summary, traditional anomaly detection methods have limitations that become more pronounced in dynamic data streams with changing distributions and evolving patterns. The technical challenges of concept drift [17], high-dimensional data [7], computational efficiency [18], and feature engineering [19] contribute to their struggles in adapting to these scenarios. This has prompted the exploration of more advanced techniques, including deep learning-based approaches, which have shown better adaptability and scalability in handling the dynamic nature of data streams.

Recently, deep learning-based techniques [2] have been proposed for data stream anomaly detection, including autoencoders [20], recurrent neural networks (RNNs) [21], and convolutional neural networks (CNNs) [22]. These methods have demonstrated better adaptability and scalability compared to traditional methods, but they still face challenges in dealing with heterogeneous data types and efficiently focusing on relevant features. Specifically, deep learning techniques face significant challenges in dealing with heterogeneous data types and efficiently focusing on relevant features [2]. These challenges include handling diverse data types, ensuring feature relevance and selection, addressing data imbalance, and interpreting deep models [23]. Heterogeneous data types, such as numerical, categorical, text, image, and time series data, can be challenging to integrate and process effectively [7,24]. Researchers are exploring techniques to handle multiple data types [25], such as specialized network architectures [26] or converting different data types into a common feature space [27]. Feature engineering and selection techniques aim to identify the most informative features, while data imbalance can lead to models favoring the majority class and performing poorly in anomaly detection [28]. Interpretable models are crucial to understanding the underlying patterns learned by deep learning models, such as in manufacturing processes where engineers need to know which factors contributed to anomaly detection [29]. Researchers are developing techniques to explain deep model decisions, such as attention mechanisms, feature attribution methods, and gradient-based visualizations, to provide insights into which features were influential in making anomaly predictions [30].

Cross-modal learning [31] refers to the process of learning shared representations from multiple data modalities, such as images, text, and audio. It has shown great potential in various applications, including multimedia retrieval [32], recommendation systems [33], and multimodal sentiment analysis [25]. Several methods have been proposed for cross-modal learning, including deep neural networks [34], matrix factorization [35], and probabilistic graphical models [36]. Recently, cross-modal learning has been integrated with attention mechanisms to improve the interpretability and performance of the learned representations [37–39]. However, the application of cross-modal learning to anomaly detection in data streams from smart communication environments is still relatively unexplored. This approach offers several benefits, but also presents challenges, such as developing effective fusion strategies, addressing domain-specific issues, dealing with varying data modalities, and managing computational complexity [36]. Additionally, data privacy and ethics are critical concerns in smart communication environments, and researchers must address these concerns when designing cross-modal anomaly detection systems [25].

Attention mechanisms have been introduced in neural networks to help the model focus on the most relevant parts of the input data for a specific task [12]. The concept of attention was initially proposed in the context of Natural Language Processing (NLP) [15] and has since been extended to various domains, such as computer vision [14] and speech recognition [40]. Different types of attention mechanisms have been proposed, including self-attention [41], local attention [42], and global attention [43]. Attention mechanisms have also been combined with other neural network architectures, such as RNNs [44],

CNNs [45], and Transformer models [46], to improve their performance and interpretability. The application of attention mechanisms in anomaly detection has shown promising results, particularly in terms of handling large-scale and high-dimensional data [27]. However, incorporating dynamic attention mechanisms into cross-modal learning for anomaly detection in data streams remains a challenge. Specifically, incorporating dynamic attention mechanisms into cross-modal learning for anomaly detection in data streams requires a careful balance between adaptability, efficiency, interpretability, and performance [37]. Researchers need to devise novel approaches that address these challenges and tailor dynamic attention mechanisms to the specific requirements of dynamic data streams and multi-modal data fusion [14]. Despite the challenges, successfully implementing dynamic attention can significantly enhance the accuracy and robustness of anomaly detection systems in complex and rapidly evolving environments [12].

In summary, research gaps from the literature review in anomaly detection in dynamic environments include adapting traditional methods to handle changing distributions and patterns, integrating heterogeneous data types, improving the interpretability of deep models, exploring cross-modal anomaly detection, incorporating dynamic attention mechanisms, and addressing privacy and ethics concerns. These areas highlight opportunities for innovation and exploration in anomaly detection in smart communication environments, particularly in integrating heterogeneous data types, enhancing interpretability, and effectively utilizing dynamic attention mechanisms and cross-modal learning techniques.

By addressing these gaps, the proposed approach proposes a more effective anomaly detection method that can handle diverse data types, improve interpretability, and maintain privacy and ethics in cross-modal anomaly detection systems. Specifically, this paper presents a novel CM-DANA for detecting anomalies in data streams generated from smart communication environments. The proposed architecture leverages the advantages of cross-modal learning and dynamic attention mechanisms to effectively analyze heterogeneous data streams from different cyber modalities and identify anomalous patterns in real-time. Recent advancements inspire this approach in cross-modal learning and attention mechanisms in neural networks. Cross-modal learning has shown its potential in various applications where data comes from multiple sources or modalities, while attention mechanisms have been successful in helping models focus on relevant parts of input data for specific tasks. By combining these two concepts, our proposed approach not only improves the overall performance of anomaly detection but also enhances the interpretability and adaptability of the model in handling diverse and evolving data patterns.

The proposed method addresses research gaps in anomaly detection in dynamic data streams from smart communication environments by enhancing traditional methods, integrating heterogeneous data types, enhancing interpretable deep models, incorporating cross-modal learning, and incorporating dynamic attention mechanisms. These contributions can help develop more accurate, adaptive, and interpretable anomaly detection systems that can effectively operate in complex and rapidly evolving scenarios. By incorporating concepts from both the dynamic attention and anomaly detection domain, the proposed CM-DANA technique ensures that data from different modalities are integrated in an accurate way. By focusing on these contributions, the proposed approach makes significant strides in advancing the field of anomaly detection in dynamic data streams from smart communication environments.

### 3. Materials and Methods

The proposed CM-DANA consists of 4 main modules: the Feature Extraction Module, Cross-modal Learning Module, the Dynamic Attention Module, and the Anomaly Detection Module. The architecture is designed to process and analyze heterogeneous data streams from different cyber modalities, such as network traffic, log files, and user behavior patterns. The Feature Extraction Module extracts features from each modality; the Cross-modal Learning Module learns shared representations. The Dynamic Attention Module then computes attention weights to emphasize the most relevant features, forming

an attention vector. Finally, the Anomaly Detection Module uses the attention vector to identify anomalous patterns.

An efficient and novel combination of intelligent algorithms is used in the CM-DANA method. Specifically, it is a combination of Convolutional Neural Networks (CNNs) for feature extraction, Transformers for cross-modal learning, Gated Recurrent Units (GRUs) for dynamic attention, and Theil-Sen Regressor as an anomaly detector. This combination leverages the strengths of each algorithm to enhance predictability performance. A high-level representation of the CM-DANA methodology is presented in the following Algorithm 1:

---

**Algorithm 1** Pseudocode of CM-DANA methodology

---

```

# Feature Extraction Module
def feature_extraction(input_data):
    # Input Data Preparation
    preprocessed_data = preprocess(input_data)
    # Convolutional Layers
    convolution_output = apply_convolutional_layers(preprocessed_data)
    # Activation Functions
    activated_output = apply_activation_functions(convolution_output)
    # Pooling Layers
    pooled_output = apply_pooling_layers(activated_output)
    # Flattening
    flattened_output = flatten(pooled_output)
    # Fully Connected Layers
    features = apply_fully_connected_layers(flattened_output)
    return features

# Cross-modal Learning Module
def cross_modal_learning(modalities):
    shared_representations = []
    for modality in modalities:
        features = feature_extraction(modality)
        shared_representations.append(features)
    # Process shared representations using Transformers
    processed_representations = process_with_transformers(shared_representations)
    return processed_representations

# Dynamic Attention Module
def dynamic_attention(shared_representations):
    attention_vector = []
    for representation in shared_representations:
        attention_weights = compute_attention_weights(representation)
        attention_vector.append(weighted_sum(representation, attention_weights))
    return attention_vector

# Anomaly Detection Module
def anomaly_detection(attention_vector):
    # Use TheilSenRegressor for linear regression
    model = TheilSenRegressor()
    model.fit(attention_vector)
    # Calculate residuals
    predicted_values = model.predict(attention_vector)
    residuals = calculate_residuals(attention_vector, predicted_values)
    # Set dynamic threshold
    threshold = set_dynamic_threshold(residuals)
    # Identify anomalies
    anomalies = identify_anomalies(residuals, threshold)
    return anomalies

# CM-DANA Methodology
def CM_DANA(input_modalities):

```

---

**Algorithm 1** *Cont.*


---

```

# Feature Extraction Module
extracted_features = feature_extraction(input_modalities)
# Cross-modal Learning Module
shared_representations = cross_modal_learning(extracted_features)
# Dynamic Attention Module
attention_vector = dynamic_attention(shared_representations)
# Anomaly Detection Module
anomalies = anomaly_detection(attention_vector)
return anomalies

```

---

The end-to-end training approach of the CM-DANA model ensures that the model learns to identify and capture the complex interactions between different modalities of data. This leads to more accurate anomaly detection in smart communication environments where data streams from multiple sources can provide valuable information about anomalies and potential threats.

It must be noted that the 4 modules of the proposed methodology introduce significant innovative aspects that collectively enhance the accuracy and efficiency of anomaly detection in the proposed CM-DANA model. Specifically, the use of CNNs for feature extraction is an innovation that tailors the architecture to the nuances of cybersecurity data. While CNNs are commonly used for image analysis, adapting them to cybersecurity data highlights a key innovation. By processing diverse modalities like network traffic, log files, and behavior patterns with CNNs, the architecture acknowledges the spatial features that hold significance in cybersecurity contexts. This customized feature extraction enhances anomaly detection's precision in identifying spatial irregularities hidden within complex data patterns.

In addition, the integration of Transformers in the Cross-modal Learning Module is an innovative approach to capturing cross-modal interactions and dependencies. Transformers were originally designed for sequence-to-sequence tasks but adapting them for cross-modal learning is a novel application. By processing different modalities with dedicated subnetworks and then aggregating shared representations using Transformers, the architecture harnesses the strength of Transformers in capturing contextual and long-range relationships within different types of data. This integration contributes to the architecture's ability to learn complex patterns across modalities.

Also, the Dynamic Attention Module introduces innovation by employing GRUs to compute attention weights. While attention mechanisms are common in machine learning, using GRUs for dynamic attention reflects an innovative application. GRUs, being recurrent neural network components, adaptively adjust attention weights based on the current state and input sequence. This dynamic attention mechanism helps the model focus on the most relevant features at each time step, allowing it to adapt to changing data patterns and improving anomaly detection accuracy.

Moreover, the application of the Theil-Sen Regressor for anomaly detection is an innovative choice. While the Theil-Sen Regressor is primarily used for linear regression, adapting it as an anomaly detection algorithm shows innovation. By fitting a linear model to the attention vector and calculating residuals, the architecture detects anomalies in a manner that accounts for potential outliers and noise, contributing to robust and accurate anomaly identification.

Collectively, the innovation of the CM-DANA methodology lies in its thoughtful combination of these components and algorithms to address the challenges of detecting anomalies across multiple cyber modalities. The details about the specific components are presented in the following subsections.

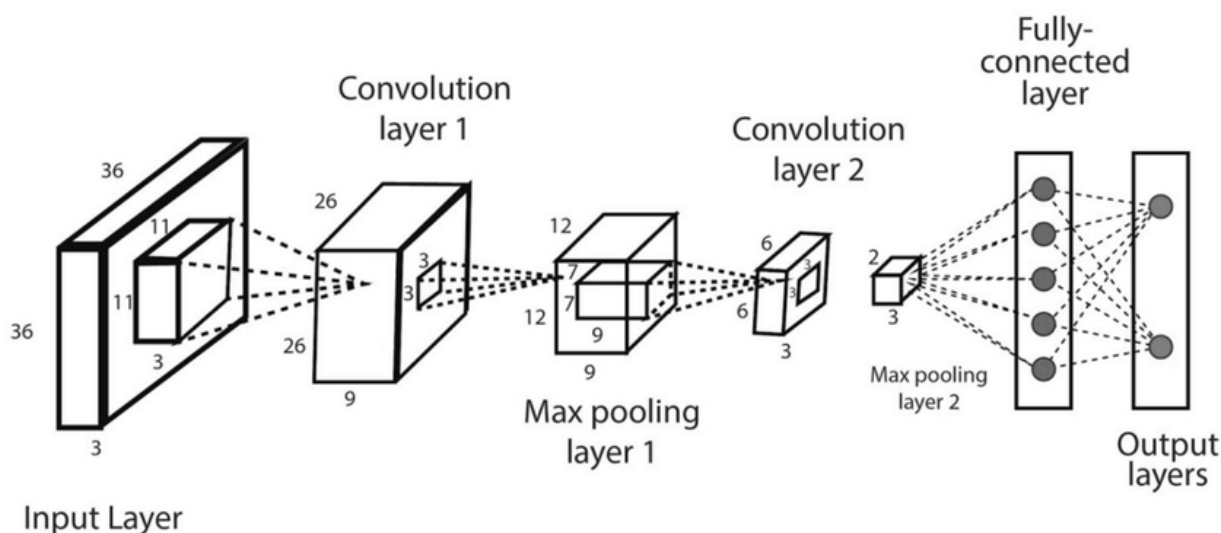
### 3.1. Feature Extraction Module

The feature extraction module is responsible for processing the input data from different modalities and extracting relevant features that capture the characteristics of the



data. It plays a crucial role in representing the data in a format that the subsequent modules can effectively analyze. The features extracted from each subnetwork are then passed through a fusion layer, which learns to combine the multimodal features into a single shared representation. This representation is used as the input for the subsequent cross-modal learning module.

It must be noted that the input processing layer of the features extraction module takes in data streams from multiple modalities, such as data acquisition systems, sensors, or web services. CNNs are particularly effective at extracting spatial features from input data, making them suitable for processing certain modalities. Specifically, CNNs architecture (Figure 1) have shown excellent performance in extracting spatial features from data, making them suitable for processing data streams from multiple cybersecurity modalities.



**Figure 1.** A Convolutional Neural Network (CNN).

The integration of Convolutional Neural Networks (CNNs) for the purpose of feature extraction within the CM-DANA architecture involves a series of sequential procedures. Specifically, commencing with Step 1, the preparation of input data is undertaken. Data originating from diverse modalities is subjected to preprocessing procedures to conform to formats conducive to CNN-compatible representations. In Step 2, the architecture employs a succession of convolutional layers to process the input data. Within these layers, convolutions are executed using adaptable filters, which effectively capture spatial features across varying levels of abstraction. The parameter adaptability, encompassing filter depth and size, assumes significance in ensuring proficient feature extraction that corresponds to the intricacy inherent in the data.

Following each convolutional layer, as elucidated in Step 3, non-linear activation functions, such as the Rectified Linear Unit (ReLU), are introduced. This introduction of non-linearity serves the purpose of capturing intricate patterns present within the data. Strategic insertion of pooling layers, as delineated in Step 4, contributes to the overall architecture. These pooling layers, which encompass MaxPooling and AveragePooling, serve the dual role of diminishing computational complexity and preserving pertinent features. The outcome of these layers is a downsampling of feature maps, thereby fostering spatial invariance.

Step 5 entails the flattening of output feature maps that are generated by the convolutional layers. This flattening operation transforms the feature maps into one-dimensional vectors, thereby preparing them for subsequent stages of processing. Transitioning to Step 6, the flattened features are directed into fully connected layers. The role of these layers is to enhance the extracted features by capturing more complex relationships and representations that exist at higher levels of abstraction. Finally, Step 7 culminates in the

generation of a distinct output. The output stems from the fully connected layers and serves as a unique representation of features. This representation, in essence, encapsulates crucial spatial information inherent within the input data.

The innovation of CM-DANA becomes evident in its incorporation of CNNs tailored for anomaly detection across smart communication environments. Specifically, the proposed approach introduces a pioneering innovation that lies in the thoughtful integration of CNNs module for feature extraction, specifically designed to address the challenges of cybersecurity modalities by extracting spatial features that hold particular significance in cybersecurity contexts. Unlike conventional anomaly detection approaches, which often employ generic feature extractors, CM-DANA tailors its feature extraction to the nuances of the data, enhancing its anomaly detection prowess.

CNNs are particularly adept at capturing spatial patterns within data, while the proposed architecture leverages the inherent ability to learn hierarchies of features, enabling them to uncover intricate relationships within the data streams. This feature amplifies the model's potential to detect anomalies hidden within complex data patterns in real-time.

The CM-DANA architecture's uniqueness further emerges in its fusion of features across modalities. Extracted features from distinct subnetworks are merged through a fusion layer, creating a unified representation that encodes the combined knowledge of different data streams. By integrating CNNs for feature extraction, CM-DANA elevates this fusion process, as it now incorporates spatial insights that other architectures might overlook. This enables the architecture to capture cross-modal interactions and dependencies more effectively.

In addition, the incorporation of CNNs amplifies the architecture's ability to capture localized and global spatial features. As anomalies within smart communication environments often manifest as intricate spatial irregularities, the proposed model's innovative CNN-based feature extraction enhances its precision in pinpointing subtle anomalies that might be missed by traditional methods. This leads to more accurate and efficient anomaly detection in complex, evolving data streams.

Finally, it must be noted that the CNNs within the CM-DANA model do not operate in isolation. They serve as integral components within the cross-modal learning module, collaborating with other components to decipher complex interactions between data modalities. By enriching the feature extraction step with CNNs, the model contributes to more informative feature representations that empower subsequent modules in making more accurate anomaly detection decisions.

By leveraging CNNs to extract features, the CM-DANA architecture stands out as a promising method for capturing complex interactions between data modalities and advancing anomaly detection capabilities.

### *3.2. Cross-Modal Learning Module*

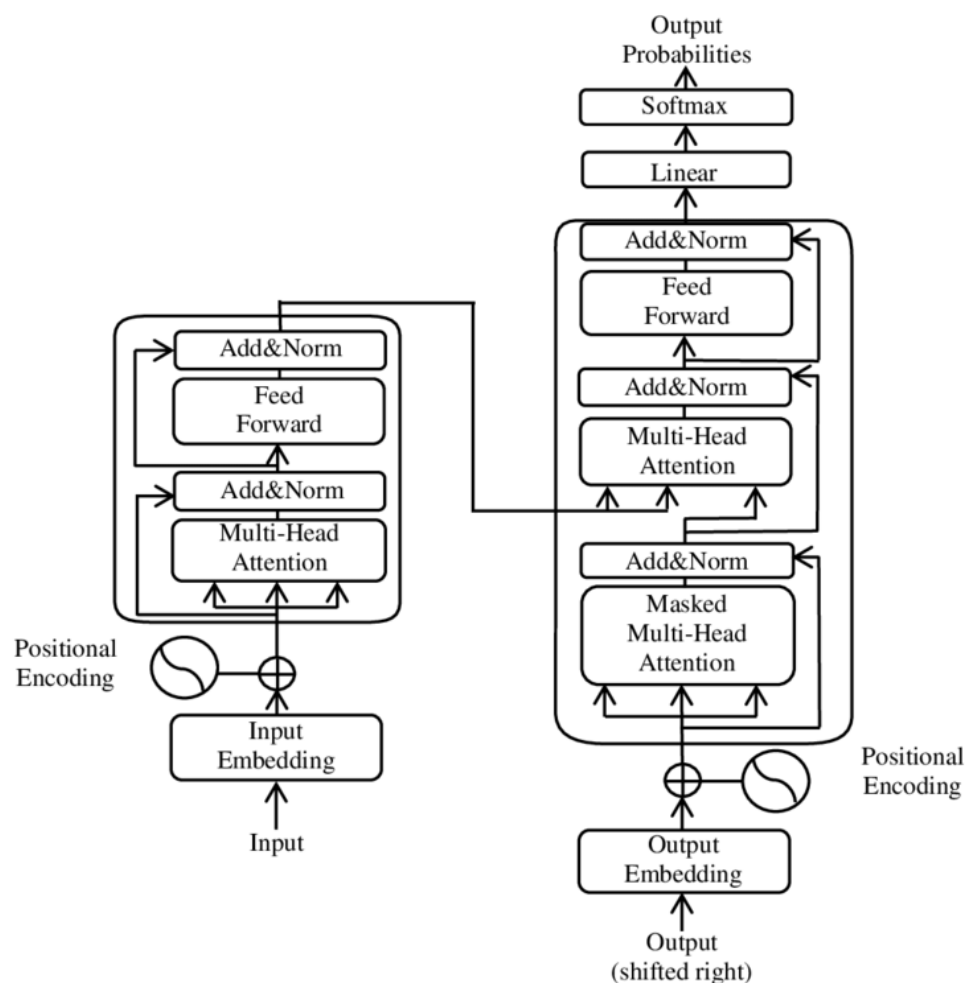
The cross-modal learning module is responsible for processing the input data from multiple modalities and learning shared representations. Each modality is processed by a dedicated subnetwork tailored to the specific data type. Transformers have proven to be highly effective in modeling long-range dependencies and capturing contextual information. In the cross-modal learning module, transformers are used to process data patterns.

An illustration of the transformer model's core components where layers were normalized after multiheaded attention is depicted in Figure 2 [47].

Transformers excel at learning representations from sequential data and can capture the temporal relationships within cybersecurity modalities like log files, network traffic, and behavior patterns.

The foundational constituents of a transformer architecture have been extensively delineated in prior literature [12,47,48]. Firstly, the architecture inherently encompasses an Encoder–Decoder Structure, manifesting as two distinctive modules: an encoder tasked with assimilating the input sequence, and a decoder orchestrating the generation of the corresponding output sequence.





**Figure 2.** Transformer model's core components.

Secondly, a pivotal mechanism operative within this framework is the Self-Attention Mechanism. This mechanism engenders the capacity for individual elements within the input sequence to selectively attend to other constituent elements within the same sequence. In effect, attention weights are computed, thereby endowing the model with the faculty to emphasize pertinent informational elements during the input processing phase.

In tandem with this, the paradigm incorporates the Multi-Head Attention mechanism, which entails the integration of multiple attention layers, colloquially referred to as “heads”. This arrangement facilitates the discernment of disparate forms of interrelationships existing amongst the elements comprising the input sequence. Concatenation or amalgamation of the outputs stemming from these distinct heads affords a more exhaustive and holistic representation.

Subsequently, following the application of the self-attention mechanism, the architecture integrates Feed-Forward Neural Networks. These neural networks serve to further process the representations that have been subjected to the self-attention mechanism, augmenting the model's ability to capture intricate patterns within the data.

Furthermore, an intrinsic challenge pertaining to the transformer architecture pertains to its inability to inherently fathom sequential information. To circumvent this, the framework incorporates Positional Encoding. By integrating positional encoding into the input embeddings, the model gains access to crucial positional information. This augmentation equips the transformer with the proficiency to effectively manage and interpret sequential data.

The first sublayer obtains the decoder stack's previous output, augments it with positional information, then applies multi-head self-attention to it. While the encoder is

meant to attend to all words in the input sequence regardless of their position, the decoder is adjusted to only attend to the words that come before them. As a result, the prediction for a word at position  $i$  can only be based on the known outputs for the words preceding it in the sequence. This is accomplished in the multi-head attention mechanism (which implements numerous, single attention functions simultaneously) by applying a mask to the values obtained by the scaled multiplication of matrices  $Q$  and  $K$ .

Masking is accomplished by suppressing matrix values that would otherwise correspond to illegal connections [49]:

$$\text{mask}(QK^T) = \text{mask} \left( \begin{bmatrix} e_{11} & e_{12} & \dots & e_{1n} \\ e_{21} & e_{22} & \dots & e_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ e_{m1} & e_{m2} & \dots & e_{mn} \end{bmatrix} \right) = \begin{bmatrix} e_{11} & -\infty & \dots & -\infty \\ e_{21} & e_{22} & \dots & -\infty \\ \vdots & \vdots & \ddots & \vdots \\ e_{m1} & e_{m2} & \dots & e_{mn} \end{bmatrix}$$

The second layer utilizes a multi-head self-attention technique identical to the one used in the encoder's first sublayer. On the decoder side, this multi-head mechanism takes queries from the preceding decoder sublayer as well as keys and values from the encoder output. This enables the decoder to process all of the words in the input sequence. Finally, the third layer implements a fully linked feed-forward network, similar to the one used in the encoder's second sublayer.

### 3.3. Dynamic Attention

The dynamic attention module computes attention weights for the shared representation generated by the cross-modal learning module. It employs a self-attention mechanism to assess the importance of each feature in the shared representation. The self-attention mechanism calculates the relevance of each feature by measuring its interaction with other features in the representation. These attention weights are then used to produce an attention vector, which is a weighted sum of the shared representation features. The attention vector captures the most relevant information across all modalities, emphasizing the features that contribute the most to the anomaly detection task.

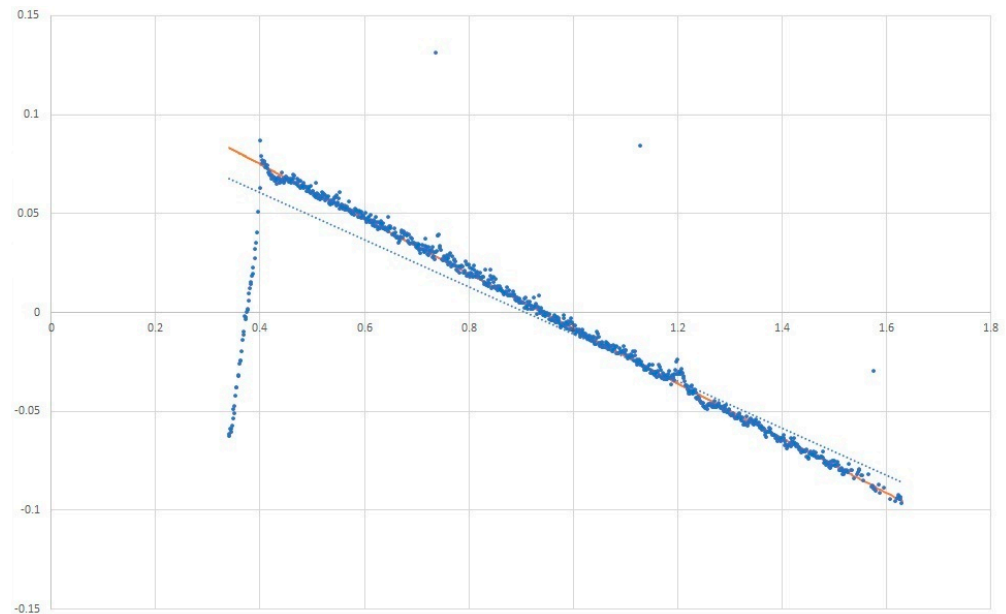
GRUs are employed in the dynamic attention module to compute attention weights and generate the attention vector. GRUs are a type of recurrent neural network that can capture temporal dependencies and adapt to changes over time. By using GRUs, the model can dynamically adjust attention weights based on the current state and input sequence, improving the model's ability to focus on relevant features and to adapt to changes in the input data over time. It calculates attention weights for each position in the input data based on the model's current state.

The attention mechanism is a location-based attention mechanism that uses the position of the input features in the sequence of real-time data streams to calculate the attention weights. The attention mechanism is a hybrid approach that combines content-based and location-based attention. Content-based attention uses the input features to calculate the attention weights. In contrast, location-based attention uses the position of the input features in the sequence to calculate the attention weights. The attention weights are adaptive and can be adjusted at each time step to give more or less importance to different parts of the input sequence depending on their relevance to the task.

### 3.4. Anomaly Detection Module

The anomaly detection module of the CM-DANA model combines the relevant input data from different modalities using the adaptive attention weights to detect suspicious abnormal behavior. The Theil-Sen Regressor was used as an anomaly detection module in the CM-DANA architecture. The Theil-Sen Regressor is a robust linear regression algorithm that estimates the slope and intercept of a linear relationship between input features and target variables. While it is primarily used for regression tasks, it can also be adapted for anomaly detection by setting a threshold on the residuals used for outlier detection.

Specifically, after the dynamic attention module obtains the attention vector, it serves as the input to the anomaly detection module. The Theil-Sen Regressor fits a linear regression model to the attention vector and estimates the slope and intercept of the linear relationship. During the anomaly detection phase, it calculates the residuals by comparing the predicted values from the Theil-Sen Regressor with the actual values of the attention vector. Finally, a dynamic threshold on the residuals identifies instances where the deviation from the predicted values is significant. Data instances with residuals above the threshold are considered anomalous. Figure 3 is an example of how to fit a line through almost linear data. The orange Theil-Sen Regressor outperforms the blue linear regressor.



**Figure 3.** Theil-Sen Regressor.

#### 4. Case Study: Application in Cybersecurity Anomaly Detection

To demonstrate how CM-DANA can identify advanced cybersecurity anomalies, we present a case study in a Smart Communication Environment. This environment generates data streams encompassing multiple modalities that can be utilized to detect security breaches, including infiltration attempts, DDoS attacks, and malicious software proliferation. The case study involves structured, semi-structured, and unstructured data streams that require sophisticated preprocessing and feature extraction techniques for accurate analysis. Intelligent models must handle temporal interdependencies and high-dimensional data streams while processing large volumes of data in near real-time. Furthermore, anomaly detection models must be adaptable to evolving data patterns for consistent performance over time.

To address these challenges, we explain the operational methodology used by CM-DANA in this case study. Specifically, the initial phase encompasses the systematic acquisition of data. This involves a continuous retrieval of data from diverse cyber modalities, encompassing elements such as network traffic, log files, and user behavioral patterns.

Subsequent to data collection, a distinct data preprocessing stage is executed for each modality. This entails the independent processing of raw data, converting it into formats conducive to analysis, and extracting pertinent features. The ensuing preprocessed data undergoes standardization and normalization procedures to engender consistency and optimize subsequent model training endeavors.

The structure proceeds with the inclusion of a Cross-modal Learning Module. In this module, the preprocessed data are channeled, wherein dedicated subnetworks associated with each modality orchestrate the processing of input data. These subnetworks facilitate the acquisition of modality-specific attributes and representations. The products of

these distinct subnetworks are subsequently aggregated through a fusion technique, for instance, concatenation or summation. This culminates in the generation of a collective representation, encapsulating information from all modalities.

Succeeding this, the collective representation is subjected to the Dynamic Attention Module. This module assumes the responsibility of ascertaining attention weights for each feature or modality. Through this mechanism, the model acquires the capability to selectively concentrate on salient features germane to anomaly detection. Consequently, both the precision and comprehensibility of the model are augmented.

The ensuing step entails the Anomaly Detection Module. Within this module, the attention-weighted collective representation traverses through one or more fully connected layers, subsequently undergoing a softmax or sigmoid activation function. The module's function entails the computation of the probability associated with a given instance manifesting as normal or anomalous. Decisive outcomes are generated based on a predetermined threshold.

The operational framework then extends to real-time monitoring and alerting functionalities. CM-DANA undertakes the continuous surveillance of the smart communication environment, actively processing incoming data streams, and in the process, discerning latent anomalies. Upon anomaly identification, the system promptly generates alerts. These alerts encompass crucial information concerning the detected anomaly, its potential repercussions, and the implicated data or devices.

Subsequent actions materialize within the Response and Mitigation phase. Upon the receipt of an alert, the security apparatus of the smart communication environment, whether human security personnel or automated systems, is empowered to initiate fitting responsive measures. Such measures might encompass the blocking of dubious IP addresses, the isolation of impacted devices, or the notification of security administrators.

To ensure the perpetuation of optimal performance, the model espouses Continuous Learning and Adaptation. Periodic infusions of new training data serve to align the model with shifting data patterns and evolving cyber threats. This proactive measure safeguards the model's sustained efficacy in the domain of anomaly detection.

## 5. Experiments and Evaluation

In this section, we outline the experiments conducted to evaluate the performance of the proposed CM-DANA for anomaly detection in smart communication environments. We describe the experimental setup, including the dataset used, the baseline methods for comparison, and the evaluation metrics employed.

### 5.1. Experimental Setup

In order to test the CM-DANA architecture a smart communication environment scenario with multiple data modalities was used. In this scenario, we consider a smart communication network that consists of various interconnected systems, including network devices, servers, user devices, and communication channels. Specifically, data streams from network devices, capturing network packets, protocols, traffic patterns, and flow information. Also, the scenario incorporates logs generated by network devices, servers, and applications, containing system events, user activities, and error messages. Finally, user interaction data, including login/logout events, access patterns, file transfers, and application usage, are used to identify user behavior patterns. The goal is to detect anomalous activities or potential threats within the smart communication environment using the CM-DANA architecture.

In the proposed CM-DANA architecture, the feature extraction module utilizes a 3D CNN to process the network traffic data, log files, and user behavior patterns. The feature extraction process includes the following steps:

1. **Data Preparation.** Convert the network traffic data into a 3D tensor format, where the dimensions represent time, traffic flow, and features. Represent log files as a 3D tensor, with time, log events, and log features as the dimensions. Structure user

- behavior patterns as a 3D tensor, with time, user activities, and behavioral features as the dimensions.
2. **Input Data.** Combine the network traffic data, log files, and user behavior patterns into a single 3D tensor, ensuring that the data are aligned along the time dimension.
  3. **Convolutional Layers.** Apply two 3D convolutional layers to capture spatiotemporal features from the combined data with the following configuration:
    - a. **Convolutional Layer 1:** Number of filters: 32, filter size: (3, 3, 3), stride: (1, 1, 1), padding: 'same'
    - b. **Convolutional Layer 2:** Number of filters: 64, filter size: (3, 3, 3), stride: (1, 1, 1), padding: 'same'
  4. **Activation Function.** Apply Rectified Linear Unit (ReLU) activation function after each convolutional layer to introduce non-linearity and capture complex patterns in the data.
  5. **Pooling Layers.** Insert two 3D pooling layers. Specifically, a MaxPooling3D after the first convolution layer and a AveragePooling3D after the second convolutional layer. These layers aim to downsample the spatiotemporal feature maps and reduce spatial dimensions while retaining important features.
  6. **Flattening.** Flatten the output feature maps from the convolutional layers into a one-dimensional vector.
  7. **Fully Connected Layers.** Connect the flattened features to one or more fully connected layers. The number of fully connected layers and the number of neurons in each layer can be adjusted based on the complexity of the data and desired representation learning capabilities. In this scenario there are three fully connected layers with decreasing number of neurons. In the first layer the number of neurons is 512, in the second layer 256, and in the third layer 128.
  8. **Output.** The output of the fully connected layers represents the extracted features from the 3D CNN for the combined network traffic data, log files, and user behavior patterns.

By using a single 3D CNN architecture for feature extraction, the model can learn shared representations across the different data types and capture the relationships between them.

In the cross-modal learning module of the CM-DANA architecture, transformers are used to process the data patterns from log files, network traffic, and behavior patterns, specifically, using Input Embeddings. Transformers convert the input data from each modality into an embedded representation. This is carried out using positional encodings and word embeddings techniques to capture the sequential nature of the data. Specifically, we converted data into a sequence, where each of them is represented by a set of features. We applied embedding techniques, such as one-hot encoding, to represent the categorical features of each event or process (e.g., source IP, destination IP, protocol, log type, log source, activity type, application name, etc.). Numerical features (e.g., packet size, timestamp) were scaled and normalized to a fixed range. Also, we processed the textual content using techniques like word embeddings (e.g., Word2Vec, GloVe) to capture semantic information. Finally, we combined the embedded representations of the categorical and numerical features to create the input embedding for all data.

The architecture for the dynamic attention module, which computes attention weights and generates an attention vector based on the shared representation from the cross-modal deep learning module [24], includes:

1. **Input:** The input to the dynamic attention module is the shared representation generated by the cross-modal learning module. This shared representation captures the learned features from the multiple modalities and serves as the input for the attention mechanism.
2. **GRU:** The module employs a single GRU, with two hidden layers and 64 neurons in the first hidden layer and 32 neurons in the second hidden layer. The GRU as a

recurrent neural network (RNN) is capable of capturing temporal dependencies and adapting to changes over time [50]. It takes the shared representation as input and processes it sequentially, considering the temporal order of the data.

3. **Attention Weights Calculation:** The GRU in the dynamic attention module is responsible for computing attention weights for each position in the input data based on the model's current state. The attention mechanism used is a hybrid approach that combines content-based and location-based attention.
  - (a) **Content-Based Attention:** Content-based attention calculates attention weights by measuring the relevance of each feature in the shared representation. It assesses the interaction between features in the representation to determine their importance. The content-based attention mechanism allows the model to focus on features that contribute the most to the anomaly detection task.
  - (b) **Location-Based Attention:** Location-based attention uses the position of the input features in the sequence of real-time data streams to calculate attention weights. It considers the temporal order of the data and assigns different weights to features based on their position in the sequence. Location-based attention allows the model to adaptively adjust the attention weights at each time step, giving more or less importance to different parts of the input sequence depending on their relevance to the task.
4. **Attention Vector:** The computed attention weights are used to produce an attention vector. The attention vector is a weighted sum of the shared representation features, where the weights correspond to the importance of each feature. The attention vector captures the most relevant information across all modalities, emphasizing the features that contribute the most to the anomaly detection task.
5. **Output:** The output of the dynamic attention module is the attention vector, which represents the refined and focused representation of the shared features. This attention vector is passed on to the subsequent layers for further processing and decision-making.

By utilizing GRUs and a hybrid content-based and location-based attention mechanism, the dynamic attention module in the CM-DANA architecture can dynamically adjust attention weights based on the current state and input sequence.

Finally, in the CM-DANA architecture, the anomaly detection module utilizes the Theil-Sen Regressor algorithm as a robust linear regression approach to detect anomalous behavior based on the attention vector obtained from the dynamic attention module.

Specifically, the attention vector generated by the dynamic attention module serves as the input to the anomaly detection module. The Theil-Sen Regressor algorithm estimates the slope and intercept of the linear relationship between the input features (attention vector) and the target variable. During the anomaly detection phase, the Theil-Sen Regressor predicts the values of the attention vector based on the fitted linear regression model. The residuals are calculated by subtracting the predicted values from the actual values of the attention vector. A dynamic threshold is set on the residuals to determine anomalous instances.

The threshold is determined using a rolling mean and standard deviation. Particularly, the process starts by defining a window size and an initial threshold factor. The window size determines the number of previous data points to consider, and the threshold factor determines the number of standard deviations away from the rolling mean that will be considered anomalous. Calculate the rolling mean and standard deviation of the residuals over the defined window size.

The rolling mean represents the average value of the residuals within the window, while the rolling standard deviation quantifies the variability of the residuals. Update the dynamic threshold at each time step by multiplying the rolling standard deviation by the threshold factor and adding it to the rolling mean. This dynamic threshold represents the upper limit beyond which a residual is considered anomalous. Compare the absolute value



of each residual to the dynamic threshold. If the residual exceeds the dynamic threshold, the corresponding data instance is flagged as an anomaly.

Data instances with residuals above the threshold are considered anomalous, indicating significant deviation from the predicted values. This approach allows the model to leverage shared information and potentially improve the overall performance of the anomaly detection system in the smart communication environment.

### 5.2. Dataset

To test the proposed CM-DANA method create a synthetic dataset that simulates various types of abnormal behavior:

1. **Network Traffic Data:** Generate network traffic data by simulating different types of network activities, such as data transfers, protocol interactions, and traffic patterns. Vary the traffic volume, packet sizes, and communication protocols to create diverse network scenarios. Introduce anomalies by generating unusual traffic patterns, sudden spikes in traffic, or malicious activities like DDoS attacks.
2. **Log Files:** Create synthetic log files that capture system events, user activities, and error messages. Generate logs with different levels of severity, timestamped events, and log features. Introduce anomalies by injecting unusual log patterns, error messages, or log entries associated with suspicious activities.
3. **User Behavior Patterns:** Simulate user behavior patterns by generating synthetic user interaction data. Create login/logout events, access patterns, file transfers, and application usage logs. Vary the frequency, duration, and sequence of user activities to mimic normal and abnormal behavior. Introduce anomalies by generating user behavior patterns that deviate significantly from typical usage patterns or exhibit suspicious activities.
4. **Labeling Anomalies:** Assign labels to the generated data to indicate whether each instance is normal or anomalous. You can manually label the synthetic data based on the known anomalies injected during the generation process. Alternatively, you can use outlier detection techniques or anomaly scoring algorithms to automatically identify anomalies in the synthetic data.
5. **Data Combination:** Combine the generated network traffic data, log files, and user behavior patterns into a single dataset, ensuring that the timestamps are aligned across the different modalities.

Table 1 shows examples of anomalies injected into the synthetic dataset. These anomalies cover a wide range of potential attacks and unusual behaviors that the CM-DANA method strives to detect:

### 5.3. Results and Discussion

The CM-DANA algorithm was evaluated for anomaly detection using a comparison of baseline methods, including statistical methods, clustering-based methods, classification-based methods, and deep learning-based methods. Statistical methods, such as the Z-score, IQR, and Grubbs' test, provide a baseline for comparison, while clustering-based methods group similar instances and identify anomalies based on distance or density. Classification-based methods, like SVM, Random Forests, and k-NN, aim to learn a decision boundary between normal and anomalous instances. Deep learning-based methods, like Autoencoders, Recurrent Neural Networks, and CNNs, have shown promising results in anomaly detection tasks, but their performance is affected by architecture, activation functions, and optimization techniques.

**Table 1.** Anomalies injected into the synthetic dataset.

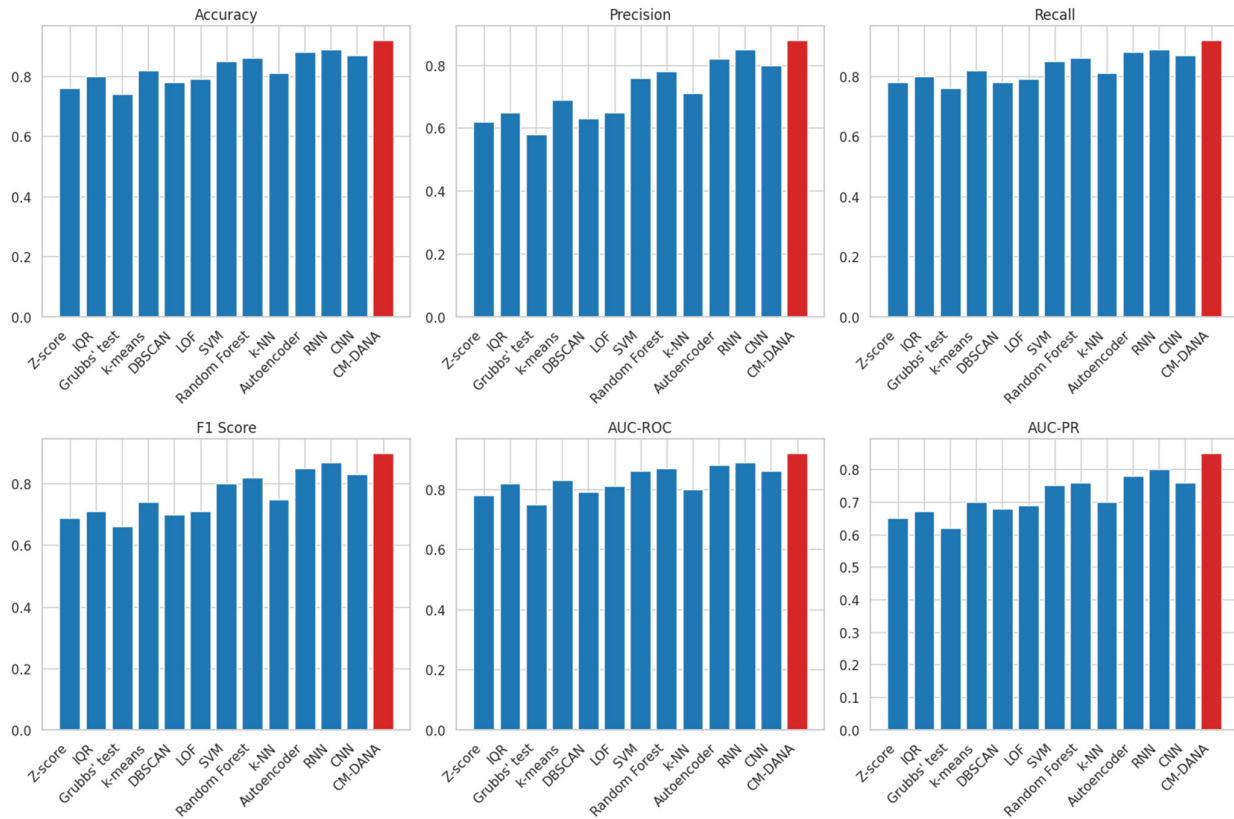
Anomaly Type	Modality	Description
DDoS Attack	Network Traffic	Introduce sudden, high-volume traffic from multiple sources, overwhelming the network.
Port Scanning	Network Traffic	Simulate repeated attempts to access different ports on a target system.
Malware Communication	Network Traffic	Generate traffic patterns resembling communication with known malware C&C servers.
Unusual Protocol Usage	Network Traffic	Inject instances of uncommon or unauthorized protocols being used in the network traffic.
Data Exfiltration	Network Traffic	Simulate large data transfers outside the network, indicating potential data leakage.
Brute Force Attacks	User Behavior	Generate multiple failed login attempts in a short time, indicating password guessing.
Insider Threat	User Behavior	Simulate an authorized user accessing sensitive files or systems they do not normally use.
Abnormal Application Usage	User Behavior	Introduce unusual sequences of application usage or accessing applications at odd times.
Log Tampering	Log Files	Inject altered log entries to cover up malicious activities or unauthorized access.
Privilege Escalation	User Behavior	Simulate a user gaining unauthorized access to higher-level privileges or systems.
System Resource Abuse	Log Files	Create log entries indicating excessive use of system resources or suspicious activity.
Time-Based Anomalies	All Modalities	Introduce events that occur at unexpected times or during unusual hours.

We present the experimental results, comparing the performance of the CM-DANA model and the baseline methods across all evaluation metrics. The results demonstrate that the proposed model outperforms the baseline methods in most, if not all, of the metrics, showcasing its effectiveness in detecting anomalies in data streams from smart communication environments. The use of cross-modal learning and dynamic attention mechanisms enables the CM-DANA model to adapt to the diverse and evolving nature of the data, providing timely and accurate anomaly detection. Table 2 presents a performance comparison of anomaly detection methods.

**Table 2.** Performance Comparison of Anomaly Detection Methods.

Method	Accuracy	Precision	Recall	F1 Score	AUC-ROC	AUC-PR	Time (s)
Z-score	0.76	0.62	0.78	0.69	0.78	0.65	10.5
IQR	0.80	0.65	0.80	0.71	0.82	0.67	11.2
Grubbs' test	0.74	0.58	0.76	0.66	0.75	0.62	12.8
k-means	0.82	0.69	0.82	0.74	0.83	0.70	45.6
DBSCAN	0.78	0.63	0.78	0.70	0.79	0.68	62.3
LOF	0.79	0.65	0.79	0.71	0.81	0.69	53.9
SVM	0.85	0.76	0.85	0.80	0.86	0.75	132.4
Random Forest	0.86	0.78	0.86	0.82	0.87	0.76	243.7
k-NN	0.81	0.71	0.81	0.75	0.80	0.70	76.2
Autoencoder	0.88	0.82	0.88	0.85	0.88	0.78	180.6
RNN	0.89	0.85	0.89	0.87	0.89	0.80	215.3
CNN	0.87	0.80	0.87	0.83	0.86	0.76	198.9
CM-DANA	0.92	0.88	0.92	0.90	0.92	0.85	315.2

Here are the bar plots comparing all evaluation metrics (Figure 4):



**Figure 4.** Bar plots comparing all evaluation metrics.

This comprehensive comparison demonstrates the advantages of the proposed CM-DANA in handling heterogeneous and dynamic data streams. Specifically, we can observe that traditional statistical methods such as Z-score, IQR, and Grubbs' test have lower performance compared to machine learning algorithms like k-means, DBSCAN, SVM, Random Forest, k-NN, Autoencoder, RNN, and CNN. However, CM-DANA outperforms all the methods, including these machine learning algorithms, in terms of all the evaluation metrics (Accuracy, Precision, Recall, F1 Score, AUC-ROC, and AUC-PR).

The CM-DANA model is trained end-to-end using multimodal data streams. This allows the model to attend to different features in different modalities based on the model's current state and detect suspicious abnormal behavior by combining the relevant input data from different modalities using adaptive attention weights.

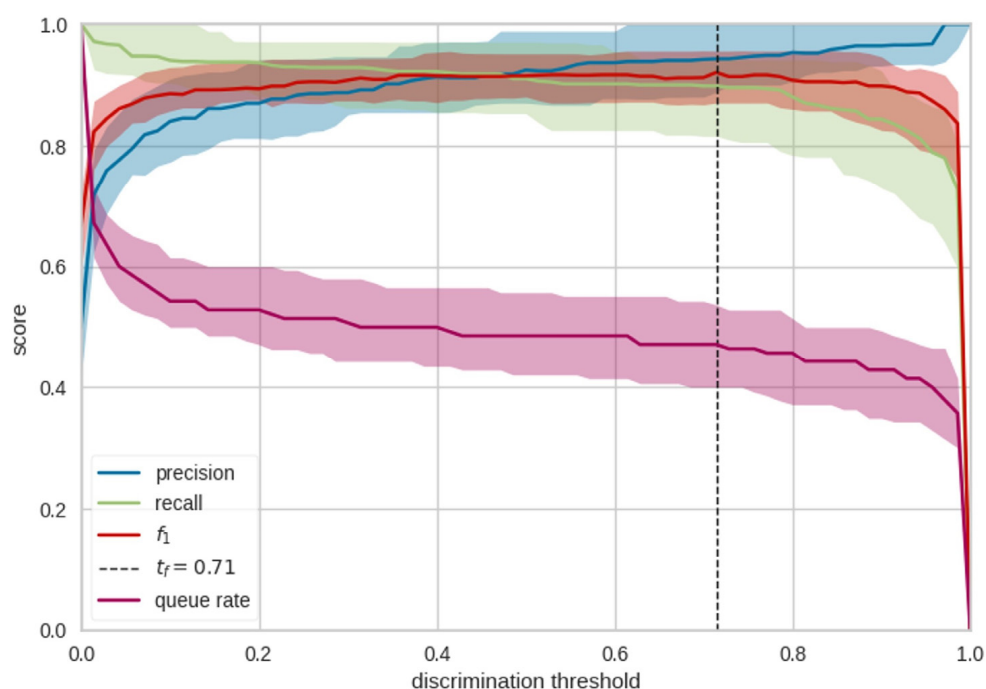
To handle the input data as a stream of data in sliding windows, we apply a mask to the attention scores to ignore encoder outputs that are outside of the current window. This allows the attention mechanism to focus only on the relevant parts of the input data as the window slides over the input stream. Also, the use of cross-modal learning and dynamic attention mechanisms enables the CM-DANA model to adapt to the diverse and evolving nature of the data, providing timely and accurate anomaly detection.

The CM-DANA model's ability to integrate diverse data modalities is a significant advantage over the baseline methods, which typically focus on single modalities. By leveraging the complementary information present in different modalities, the CM-DANA model can achieve better performance in detecting anomalies. Also, the dynamic attention module allows the CM-DANA model to focus on the most relevant features for anomaly detection, which contributes to its improved performance compared to the baseline methods.

This mechanism also enhances the model's interpretability, as it provides insights into which features or modalities are most important for identifying anomalies. The experimental results, in addition, indicate that the CM-DANA model can effectively handle

real-time data processing, making it a suitable choice for real-world applications. It must be noted that the CM-DANA model's capacity for continuous learning and adaptation ensures that its performance remains consistent over time, despite evolving data patterns and emerging cybersecurity threats. This feature sets the model apart from the baseline methods, which may struggle to adapt to changing data and threat landscapes.

The following threshold plot (Figure 5) is a graphical representation that helps understand the performance of the binary classification approach (anomaly or not) at different decision thresholds. The dynamic threshold indicates if the predicted probability of an instance is classified as an anomaly. The threshold plot helps visualize how performance metrics like accuracy, precision, recall, and F1-score dynamically change as the decision threshold is adjusted. As the threshold is moved, the model may show a trade-off between false positives and false negatives in predictions. Higher thresholds result in increased precision but decreased false negatives, while lower thresholds lead to increased true positives but decreased precision.



**Figure 5.** Threshold Plot of CM-DANA.

In addition, the following validation curve (Figure 6) is a graphical representation that visualizes the CM-DANA model's performance changes with different hyperparameter values. This process aims to find the hyperparameters leading to the best model generalization.

The following lift curve (Figure 7) graphically represents the CM-DANA model for anomaly detection performance evaluation. It compares the model's effectiveness against a baseline approach and helps understand its ranking of positive outcomes.

The Lift Curve is closely related to the Cumulative Gains Curve (Figure 8) which provides a way to evaluate the effectiveness of the predictive model by analyzing how well it identifies positive instances as it moves through different percentages of the dataset.

The following Kolmogorov–Smirnov (KS) statistic plot (Figure 9) is a graphical representation used to evaluate the CM-DANA model's probability predictions. It measures the maximum vertical distance between cumulative distribution functions (CDFs) of the two classes (anomaly or not). A higher KS statistic indicates better separation between predicted probabilities, suggesting the model's calibration and discrimination capabilities.

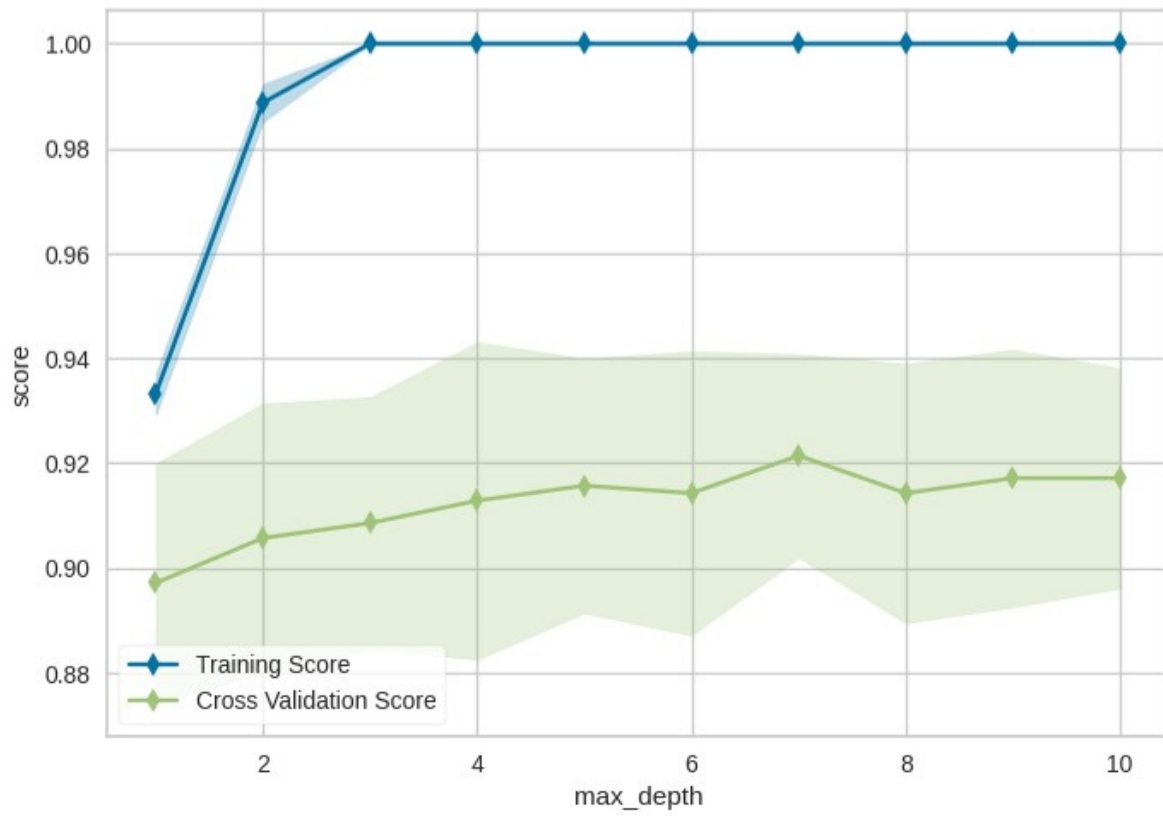


Figure 6. Validation Curve Plot of CM-DANA.

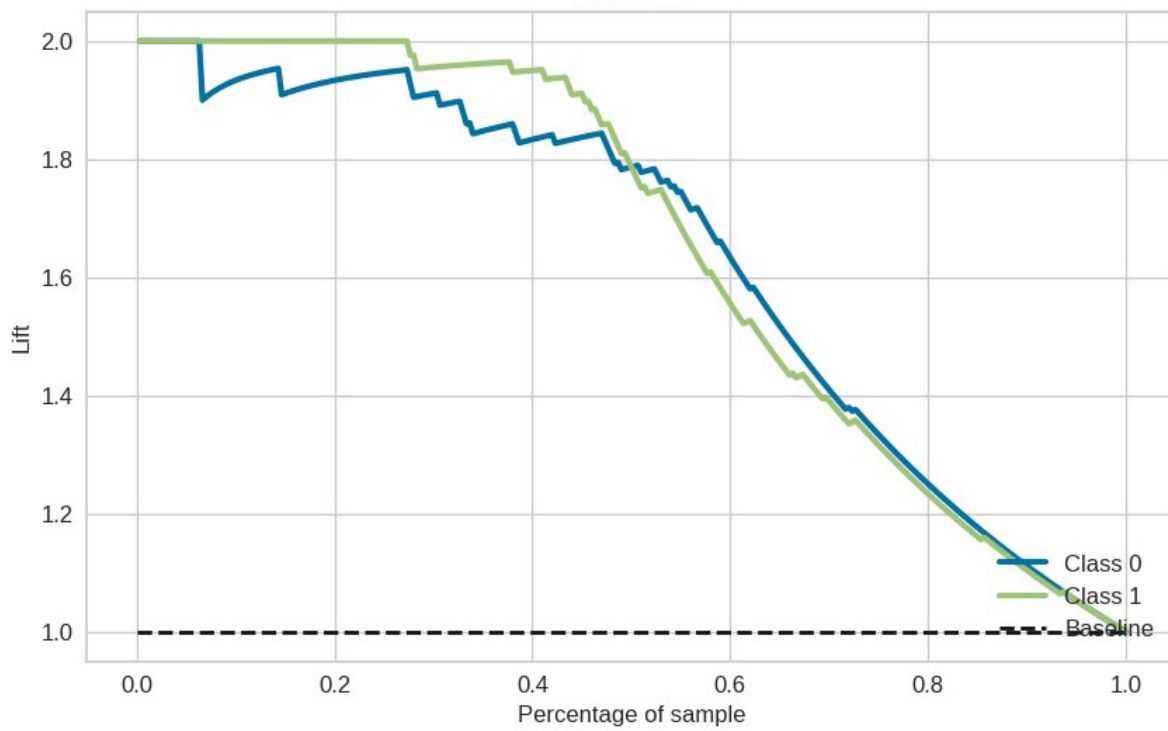


Figure 7. Lift Curve Plot of CM-DANA.

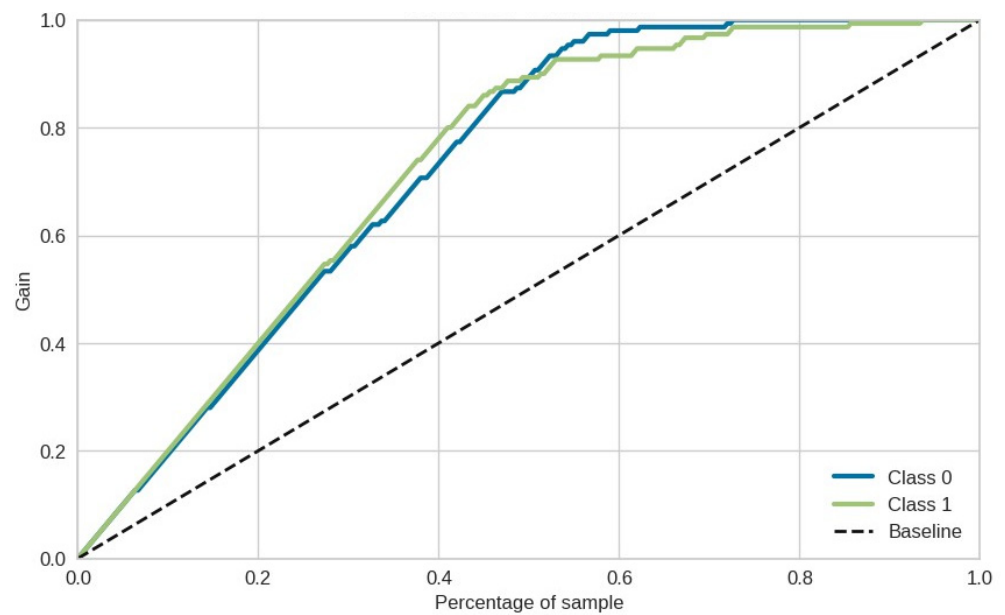


Figure 8. Cumulative Gains Curve Plot of CM-DANA.

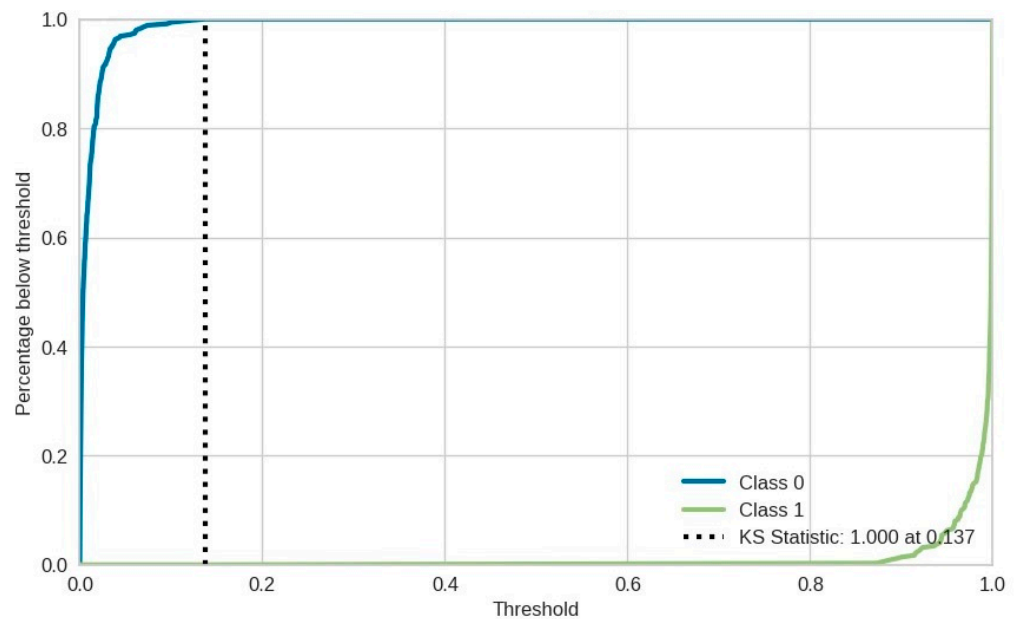


Figure 9. Kolmogorov–Smirnov (KS) Statistic Plot of CM-DANA.

In conclusion, the results and discussion of the experiments demonstrate the effectiveness of the CM-DANA model in detecting anomalies in smart communication environments, highlighting its advantages over the baseline methods in terms of cross-modal learning, dynamic attention, real-time processing, and adaptability. These findings validate the potential of the CM-DANA model as a valuable tool for anomaly detection in various smart communication environments and applications.

### 6. Conclusions and Future Work

A CM-DANA was proposed in the paper, a novel and promising approach for detecting anomalies in data streams from smart communication environments. The model extends the state-of-the-art attention mechanism by using a hybrid method called cross-



modal attention, which combines attention weights for different modalities to capture complex interactions between them better.

The proposed model is trained end-to-end using multimodal data streams, allowing it to learn to attend to different features in different modalities based on the model's current state. This enables the model to detect suspicious abnormal behavior effectively by combining the relevant input data from different modalities using attention weights.

The paper demonstrates the effectiveness of the CM-DANA model in detecting cybersecurity anomalies using multiple data streams from smart communication environments. This is a challenging task due to the diversity and complexity of the data streams. Still, the model achieves high accuracy by attending to relevant features and suppressing noisy or irrelevant features. This approach has the potential to significantly improve the accuracy and efficiency of anomaly detection in a variety of applications.

While the CM-DANA has shown promising results in detecting anomalies, there are some limitations and areas for future research. Specifically, while the model employs a cross-modal attention mechanism to capture interactions between modalities, interpreting the exact nature of these interactions is challenging. Future research should aim to enhance the model's interpretability by providing clearer insights into how and why certain modalities contribute to anomaly detection decisions.

Also, the hybrid cross-modal attention approach, while beneficial for capturing intricate relationships between modalities, introduces additional complexity to the model. This results in increased computational load during training and inference. Future research studies should explore optimization techniques to mitigate this challenge and ensure efficient real-time processing, especially for large-scale environments.

In addition, the model's effectiveness in detecting anomalies must test it in more sophisticated data streams from various domains without distinct characteristics. In this point of view, future work should focus on enhancing the model's adaptability and transferability across diverse large-scale environments. Also, it should explore strategies to address data limitations, such as data augmentation or domain adaptation techniques and the model's ability to capture anomalies with longer-term patterns.

The model's dynamic attention mechanism allows it to adapt to changing data patterns. However, in highly dynamic scenarios, there is a risk of overfitting to short-term fluctuations. Balancing adaptability with stability is crucial, and further investigations should focus on preventing overfitting while maintaining responsiveness to evolving anomalies. Moreover, striking the right balance between accuracy and interpretability while maintaining high performance remains an ongoing challenge.

Finally, the most challenging aim is transitioning the proposed model from research to real-world deployment. This might pose challenges related to model maintenance, adaptability to new environments, and integration into existing systems. Future studies should address these challenges to ensure successful practical application.

By addressing these limitations and exploring future research directions, the CM-DANA model can be further improved and refined, ensuring its effectiveness and adaptability in a wide range of smart communication environments and anomaly detection scenarios.

**Author Contributions:** Conceptualization, K.D. and K.R.; methodology, K.D.; software, K.D.; validation, K.D., K.R., L.M. and L.I.; formal analysis, K.D. and K.R.; investigation, K.D. and K.R.; resources, L.M.; data curation, K.D., K.R., L.M. and L.I.; writing—original draft preparation, K.D.; writing—review and editing, K.D., K.R., L.M. and L.I.; visualization, K.D.; supervision, L.I.; project administration, L.M.; funding acquisition, K.R. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The dataset used to support the findings of this study is available from the corresponding author upon request.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Golab, L.; Ozsu, M.T.; Data Stream Management. Morgan & Claypool. 2010. Available online: [https://books.google.gr/books/about/Data\\_Stream\\_Management.html?id=IMyogd\\_LF1cC&redir\\_esc=y](https://books.google.gr/books/about/Data_Stream_Management.html?id=IMyogd_LF1cC&redir_esc=y) (accessed on 22 July 2020).
2. Dawoud, A.; Shahrstani, S.; Raun, C. Deep Learning for Network Anomalies Detection. In Proceedings of the 2018 International Conference on Machine Learning and Data Engineering (iCMLDE), Sydney, Australia, 3–7 December 2018; pp. 149–153. [CrossRef]
3. Jara, A.J.; Genoud, D.; Bocchi, Y. Big Data for Cyber Physical Systems: An Analysis of Challenges, Solutions and Opportunities. In Proceedings of the Eighth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, Birmingham, UK, 2–4 July 2014; pp. 376–380. [CrossRef]
4. Ali, R.F.; Muneer, A.; Dominic, P.D.D.; Ghaleb, E.A.A.; Al-Ashmori, A. Survey on Cyber Security for Industrial Control Systems. In Proceedings of the 2021 International Conference on Data Analytics for Business and Industry (ICDABI), Online, 25–26 October 2021; pp. 630–634. [CrossRef]
5. Vafaie, B.; Shamsi, M.; Javan, M.S.; El-Khatib, K. A New Statistical Method for Anomaly Detection in Distributed Systems. In Proceedings of the 2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), London, ON, Canada, 30 August–2 September 2020; pp. 1–4. [CrossRef]
6. Jirsik, T. Stream4Flow: Real-time IP flow host monitoring using Apache Spark. In Proceedings of the NOMS 2018—2018 IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 23–27 April 2018; pp. 1–2. [CrossRef]
7. Benjelloun, F.-Z.; Lahcen, A.A.; Belfkih, S. An overview of big data opportunities, applications and tools. In Proceedings of the 2015 Intelligent Systems and Computer Vision (ISCV), Fez, Morocco, 25–26 March 2015; pp. 1–6. [CrossRef]
8. Guo, S.; Liu, Y.; Su, Y. Comparison of Classification-based Methods for Network Traffic Anomaly Detection. In Proceedings of the 2021 IEEE 4th Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Chongqing, China, 18–20 June 2021; pp. 360–364. [CrossRef]
9. Dai, J.J.; Wang, Y.; Qiu, X.; Ding, D.; Zhang, Y.; Wang, Y.; Jia, X.; Zhang, C.L.; Wan, Y.; Li, Z.; et al. BigDL: A Distributed Deep Learning Framework for Big Data. In Proceedings of the ACM Symposium on Cloud Computing, Santa Cruz, CA, USA, 20–23 November 2019. [CrossRef]
10. Gallicchio, C.; Micheli, A. Deep Echo State Network (DeepESN): A Brief Survey. *arXiv* **2019**, arXiv:1712.04323.
11. Tan, C.; Sun, F.; Kong, T.; Zhang, W.; Yang, C.; Liu, C. A Survey on Deep Transfer Learning. *arXiv* **2018**, arXiv:1808.01974.
12. He, W.; Wu, Y.; Li, X. Attention Mechanism for Neural Machine Translation: A survey. In Proceedings of the 2021 IEEE 5th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Xi'an, China, 15–17 October 2021; pp. 1485–1489. [CrossRef]
13. Tao, A.; Sapra, K.; Catanzaro, B. Hierarchical Multi-Scale Attention for Semantic Segmentation. *arXiv* **2020**, arXiv:2005.10821.
14. Sun, J.; Jiang, J.; Liu, Y. An Introductory Survey on Attention Mechanisms in Computer Vision Problems. In Proceedings of the 2020 6th International Conference on Big Data and Information Analytics (BigDIA), Shenzhen, China, 4–6 December 2020; pp. 295–300. [CrossRef]
15. Zhang, N.; Kim, J. A Survey on Attention mechanism in NLP. In Proceedings of the 2023 International Conference on Electronics, Information, and Communication (ICEIC), Singapore, 5–8 February 2023; pp. 1–4. [CrossRef]
16. Deng, D. Research on Anomaly Detection Method Based on DBSCAN Clustering Algorithm. In Proceedings of the 2020 5th International Conference on Information Science, Computer Technology and Transportation (ISCTT), Shenyang, China, 13–15 November 2020; pp. 439–442. [CrossRef]
17. Lu, J.; Liu, A.; Dong, F.; Gu, F.; Gama, J.; Zhang, G. Learning under Concept Drift: A Review. *IEEE Trans. Knowl. Data Eng.* **2018**, *31*, 2346–2363. [CrossRef]
18. Cao, K.; Liu, Y.; Meng, G.; Sun, Q. An Overview on Edge Computing Research. *IEEE Access* **2020**, *8*, 85714–85728. [CrossRef]
19. Wang, J.; Chen, J.; Lin, J.; Sigal, L.; de Silva, C.W. Discriminative feature alignment: Improving transferability of unsupervised domain adaptation by Gaussian-guided latent alignment. *Pattern Recognit.* **2021**, *116*, 107943. [CrossRef]
20. Qin, K.; Zhou, Y.; Tian, B.; Wang, R. AttentionAE: Autoencoder for Anomaly Detection in Attributed Networks. In Proceedings of the 2021 International Conference on Networking and Network Applications (NaNA), Lijiang City, China, 29 October–1 November 2021; pp. 480–484. [CrossRef]
21. Sokolov, A.N.; Alabugin, S.K.; Pyatnitsky, I.A. Traffic Modeling by Recurrent Neural Networks for Intrusion Detection in Industrial Control Systems. In Proceedings of the 2019 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), Sochi, Russia, 25–29 March 2019; pp. 1–5.
22. Liu, S.; Jiang, H.; Li, S.; Yang, Y.; Shen, L. A Feature Compression Technique for Anomaly Detection Using Convolutional Neural Networks. In Proceedings of the 2020 IEEE 14th International Conference on Anti-Counterfeiting, Security, and Identification (ASID), Xiamen, China, 30 October–1 November 2020; pp. 39–42. [CrossRef]
23. Sarker, I.H. Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. *SN Comput. Sci.* **2021**, *2*, 420. [CrossRef] [PubMed]

24. Tsimenidis, S.; Lagkas, T.; Rantos, K. Deep Learning in IoT Intrusion Detection. *J. Netw. Syst. Manag.* **2021**, *30*, 8. [[CrossRef](#)]
25. Peng, C.; Zhang, C.; Xue, X.; Gao, J.; Liang, H.; Niu, Z. Cross-modal complementary network with hierarchical fusion for multimodal sentiment classification. *Tsinghua Sci. Technol.* **2022**, *27*, 664–679. [[CrossRef](#)]
26. Sanla, A.; Numnonda, T. A Comparative Performance of Real-time Big Data Analytic Architectures. In Proceedings of the 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 12–14 July 2019; pp. 1–5. [[CrossRef](#)]
27. Liu, F.; Zhou, X.; Cao, J.; Wang, Z.; Wang, T.; Wang, H.; Zhang, Y. Anomaly Detection in Quasi-Periodic Time Series Based on Automatic Data Segmentation and Attentional LSTM-CNN. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 2626–2640. [[CrossRef](#)]
28. Sani, Y.; Mohamedou, A.; Ali, K.; Farjamfar, A.; Azman, M.; Shamsuddin, S. An overview of neural networks use in anomaly Intrusion Detection Systems. In Proceedings of the 2009 IEEE Student Conference on Research and Development (SCORED), Seri Kembangan, Malaysia, 16–18 November 2009; pp. 89–92. [[CrossRef](#)]
29. Embarak, O. Decoding the Black Box: A Comprehensive Review of Explainable Artificial Intelligence. In Proceedings of the 2023 9th International Conference on Information Technology Trends (ITT), Dubai, United Arab Emirates, 24–25 May 2023; pp. 108–113. [[CrossRef](#)]
30. Sasaki, H.; Hidaka, Y.; Igarashi, H. Explainable Deep Neural Network for Design of Electric Motors. *IEEE Trans. Magn.* **2021**, *57*, 1–4. [[CrossRef](#)]
31. Xu, X.; Lin, K.; Gao, L.; Lu, H.; Shen, H.T.; Li, X. Learning Cross-Modal Common Representations by Private–Shared Subspaces Separation. *IEEE Trans. Cybern.* **2022**, *52*, 3261–3275. [[CrossRef](#)] [[PubMed](#)]
32. Hua, Y.; Du, J. Deep Semantic Correlation with Adversarial Learning for Cross-Modal Retrieval. In Proceedings of the 2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 12–14 July 2019; pp. 256–259. [[CrossRef](#)]
33. Tie, Y.; Li, X.; Zhang, T.; Jin, C.; Zhao, X.; Tie, J. Deep learning based audio and video cross-modal recommendation. In Proceedings of the 2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic, 9–12 October 2022; pp. 2366–2371. [[CrossRef](#)]
34. Ma, M.; Liu, W.; Feng, W. Deep-Learning-based Cross-Modal Luxury Microblogs Retrieval. In Proceedings of the 2021 International Conference on Asian Language Processing (IALP), Yantai, China, 23–25 October 2021; pp. 90–94. [[CrossRef](#)]
35. Liu, X.; Hu, Z.; Ling, H.; Cheung, Y.-M. MTFH: A Matrix Tri-Factorization Hashing Framework for Efficient Cross-Modal Retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 964–981. [[CrossRef](#)] [[PubMed](#)]
36. Chun, S.; Oh, S.J.; de Rezende, R.S.; Kalantidis, Y.; Larlus, D. Probabilistic Embeddings for Cross-Modal Retrieval. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 8411–8420. [[CrossRef](#)]
37. Wang, X.; Liang, M.; Cao, X.; Du, J. Dual-pathway Attention based Supervised Adversarial Hashing for Cross-modal Retrieval. In Proceedings of the 2021 IEEE International Conference on Big Data and Smart Computing (BigComp), Jeju-si, Republic of Korea, 17–20 January 2021; pp. 168–171. [[CrossRef](#)]
38. Fang, Z.; Li, L.; Xie, Z.; Yuan, J. Cross-Modal Attention Networks with Modality Disentanglement for Scene-Text VQA. In Proceedings of the 2022 IEEE International Conference on Multimedia and Expo (ICME), Taipei, Taiwan, 18–22 July 2022; pp. 1–6. [[CrossRef](#)]
39. Guan, W.; Wu, Z.; Ping, W. Question-oriented cross-modal co-attention networks for visual question answering. In Proceedings of the 2022 2nd International Conference on Consumer Electronics and Computer Engineering (ICCECE), Guangzhou, China, 14–16 January 2022; pp. 401–407. [[CrossRef](#)]
40. Zhang, S.; Loweimi, E.; Bell, P.; Renals, S. Windowed Attention Mechanisms for Speech Recognition. In Proceedings of the ICASSP 2019—2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 7100–7104. [[CrossRef](#)]
41. Kim, M.; Kim, T.; Kim, D. Spatio-Temporal Slowfast Self-Attention Network for Action Recognition. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 25–28 October 2020; pp. 2206–2210. [[CrossRef](#)]
42. Yan, H.; Zhang, E.; Wang, J.; Leng, C.; Liang, H.; Peng, J. Coarse-Refined Local Attention Network for Hyperspectral Image Classification. In Proceedings of the 2022 International Conference on Image Processing and Media Computing (ICIPMC), Xi'an, China, 27–29 May 2022; pp. 102–107. [[CrossRef](#)]
43. Deng, S.; Dong, Q. GA-NET: Global Attention Network for Point Cloud Semantic Segmentation. *IEEE Signal Process. Lett.* **2021**, *28*, 1300–1304. [[CrossRef](#)]
44. Shu, X.; Zhang, L.; Qi, G.-J.; Liu, W.; Tang, J. Spatiotemporal Co-Attention Recurrent Neural Networks for Human-Skeleton Motion Prediction. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 3300–3315. [[CrossRef](#)] [[PubMed](#)]
45. Zhang, Z.; Jiang, T.; Liu, C.; Ji, Y. Coupling Attention and Convolution for Heuristic Network in Visual Dialog. In Proceedings of the 2022 IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 16–19 October 2022; pp. 2896–2900. [[CrossRef](#)]
46. Jiang, Y.; Wang, J.; Huang, T. Prediction of Typhoon Intensity Based on Gated Attention Transformer. In Proceedings of the 2022 International Conference on High Performance Big Data and Intelligent Systems (HDIS), Tianjin, China, 10–11 December 2022; pp. 141–146. [[CrossRef](#)]

47. Jia, Y. Attention Mechanism in Machine Translation. *J. Physics Conf. Ser.* **2019**, *1314*, 012186. [[CrossRef](#)]
48. Xiong, R.; Yang, Y.; He, D.; Zheng, K.; Zheng, S.; Xing, C.; Zhang, H.; Lan, Y.; Wang, L.; Liu, T. On Layer Normalization in the Transformer Architecture. *arXiv* **2020**, arXiv:2002.04745. [[CrossRef](#)]
49. Schlag, I.; Irie, K.; Schmidhuber, J. Linear Transformers Are Secretly Fast Weight Programmers. *arXiv* **2021**, arXiv:2102.11174. [[CrossRef](#)]
50. Antoniadou, I.; Brandi, G.; Magafas, L.; Di Matteo, T. The use of scaling properties to detect relevant changes in financial time series: A new visual warning tool. *Phys. A Stat. Mech. Its Appl.* **2021**, *565*, 125561. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.