electronics

MDPI

Article

Dynamic Telemetry and Deep Neural Networks for Anomaly Detection in 6G Software-Defined Networks

Grzegorz Rzym [1], Amadeusz Masny [2] and Piotr Chołda [1,*]

[1] AGH University of Krakow, Institute of Telecommunications, Al. Mickiewicza 30, 30-059 Krakow, Poland; rzym@agh.edu.pl
[2] Independent Researcher, 30-095 Krakow, Poland
* Correspondence: piotr.cholda@agh.edu.pl

Abstract: With the increasing availability of computational power, contemporary machine learning has undergone a paradigm shift, placing a heightened emphasis on deep learning methodologies. The pervasive automation of various processes necessitates a critical re-evaluation of contemporary network implementations, specifically concerning security protocols and the imperative need for swift, precise responses to system failures. This article introduces a meticulously crafted solution designed explicitly for 6G software-defined networks (SDNs). The approach employs deep neural networks for anomaly detection within network traffic, contributing to a more robust security framework. Furthermore, the paper delves into the realm of network monitoring automation by harnessing dynamic telemetry, providing a specialized and forward-looking strategy to tackle the distinctive challenges inherent in SDN environments. In essence, our proposed solution aims to elevate the security and responsiveness of 6G mobile networks. By addressing the intricate challenges posed by next-generation network architectures, it seeks to fortify these networks against emerging threats and dynamically adapt to the evolving landscape of next-generation technology.

Keywords: deep neural networks (DNN); software-defined network (SDN); 6G

Citation: Rzym, G.; Masny, A.; Chołda, P. Dynamic Telemetry and Deep Neural Networks for Anomaly Detection in 6G Software-Defined Networks. *Electronics* **2024**, *13*, 382. https://doi.org/10.3390/electronics13020382

Academic Editor: Ping-Feng Pai

Received: 19 December 2023
Revised: 4 January 2024
Accepted: 15 January 2024
Published: 17 January 2024

1. Introduction

Traditional methods of managing computer networks and monitoring these networks rely on data collection protocols such as SNMP (Simple Network Management Protocol), NetFlow, IPFIX, or NETCONF (Network Configuration Protocol) [1]. Many companies create their own tools based on these protocols, allowing insight into the state of their computer network and responding to failures. With the impressive increase in demand for network services, requirements for individual components and monitoring applications have also increased.

The amount of data generated by network devices, including control, statistical, and user data, is growing incredibly fast [2–5]. Traditional human-involved analysis methods require qualified human resources and working time, a trend increasingly displaced by automated solutions in today's era of computerization.

One solution enabling the implementation of modern methods for managing and monitoring telecommunication networks is the concept of software-defined networks (SDN) [6]. It introduces a centralized architecture for managing computer networks that is fully programmable. However, the analytical aspect remains, involving continuous network monitoring and anomaly detection. Collecting data in one central location has business benefits, facilitating the coordination and correlation of collected results. Nevertheless, searching such a database can pose a significant computational challenge known as "big data analytics" [7]. Tools facilitating the implementation, deployment, and utilization of massive datasets come to the rescue. An example is the open source solution proposed by Cisco—the Platform for Network Data Analytics (PNDA) [8]. PNDA allows for the collection and reading of data, along with real-time analysis. Considering the enormous

amount of data for analysis, this is where the possibility of harnessing artificial intelligence (AI) and machine learning (ML) comes into play.

Continuous data collection is an ideal solution for monitoring systems. However, it can be very disadvantageous in terms of device load and network throughput, whether it be in the transmission or management network. Therefore, dynamically changing data collection intervals based on the network's state (e.g., increasing polling frequency after anomaly detection) seems reasonable.

This article aims to showcase the potential applications of machine learning methods in tandem with software-defined networks to automate anomaly detection in computer networks, particularly within the dynamic framework of advancing 6G networks. The primary methodology presented involves automatically adjusting data collection intervals from network devices based on anomaly detection in traffic using deep neural networks (DNNs). The core objective is to conduct a comparative analysis of two distinct implementations, shedding light on how these variations influence the effectiveness of the proposed mechanism.

A significant differentiator lies in the application of fundamental data collection intervals orchestrated by the SDN controller. In the context of burgeoning 6G mobile networks, characterized by ultra-low latency and unprecedented data rates, the dynamics of data collection become pivotal. The intricacies of network behavior and anomalies in such cutting-edge environments necessitate a nuanced approach.

Moreover, the subsequent training of artificial neural networks further delineates the divergence between the implementations. As we traverse into the era of 6G, characterized by advanced machine learning integration and the Internet of Things (IoT) proliferation, the adaptability of these networks becomes paramount. The input data shaping the neural networks' understanding of network behavior becomes a crucial facet in this context, warranting tailored considerations for the intricacies of 6G environments.

In the realm of 6G mobile networks, traffic isolation can involve segregating different types of network traffic to prevent unauthorized access and mitigate potential security threats. Simultaneously, segmentation in this context plays a crucial role by dividing the network into isolated segments, curbing the lateral movement of threats and bolstering the overall security architecture.

Considering the unique characteristics of 6G networks, such as ultra-low latency and unprecedented data rates, the integration of these additional security measures can provide a robust multilayered defense strategy. By isolating critical services and segmenting network traffic effectively, the proposed deep neural network anomaly detection solution can work synergistically with these measures to ensure the security and integrity of 6G mobile networks.

While our solution is designed with a forward-looking perspective towards 6G networks, its underlying principles are generalizable and can be adapted to enhance anomaly detection in existing 5G or earlier networks. The choice to spotlight 6G is motivated by the advanced capabilities and flexibility it offers, allowing for a more straightforward and effective implementation.

In essence, this exploration transcends the traditional scope of anomaly detection mechanisms, delving into the domain of futuristic 6G mobile networks. The comparative analysis aims not only to uncover the implications of varied data collection intervals and diverse training data on the performance but also to illuminate the adaptability of the proposed mechanism within the dynamic context of these cutting-edge networks, offering insights that are paramount for shaping the future of network management in the 6G era.

The contributions of this paper are the following:

- Automated anomaly detection: Introducing an innovative method for automating anomaly detection in computer networks, specifically designed to address the dynamic nature of advancing 6G networks.

- Integration of machine learning and SDNs: Highlighting the synergistic application of machine learning methods, particularly DNNs, in conjunction with SDNs to enhance the capabilities of anomaly detection.
- Dynamic data collection intervals: Proposing a central method that involves the automatic adjustment of data collection intervals from network devices based on anomaly detection in traffic, presenting a nuanced and adaptive strategy.
- Comparative analysis: Conducting a comprehensive comparative examination of two distinct implementations, providing valuable insights into how variations in data collection intervals significantly influence the effectiveness of the proposed anomaly detection mechanism.
- Adaptability in 6G networks: Recognizing the unique challenges posed by 6G mobile networks, characterized by ultra-low latency and unprecedented data rates, the study explores the adaptability of the proposed mechanism within this cutting-edge context.
- Considerations for training neural networks: Extending the exploration into the training of artificial neural networks, emphasizing the divergence between implementations and the need for tailored considerations in the era of 6G and advanced machine learning integration.
- Future implications for network management: Transcending traditional anomaly detection mechanisms, this paper provides insights crucial for shaping the future of network management in the 6G era, where adaptability and innovation are paramount.

Without the proposed dynamic telemetry system, there might be delayed or suboptimal responses to emerging issues, leading to increased downtime or compromised network performance. This comparison could showcase reduced detection and mitigation times when our system is utilized, resulting in a more resilient and secure network environment.

In contrast, the absence of our dynamic telemetry system might result in a less agile response to anomalies, potentially leading to prolonged service disruptions or a slower adaptation to dynamic network conditions. This comparison aims to underscore the added value and efficiency introduced by our proposed solution in addressing and mitigating anomalies effectively within 6G SDNs.

2. Related Work

Machine learning applied to SDNs represents a cutting-edge approach to enhancing network performance, efficiency, and security. In the realm of SDNs, machine learning algorithms analyze vast amounts of network data, adapting and optimizing the network dynamically. These algorithms can predict traffic patterns, identify anomalies, and automate network management tasks. For an in-depth exploration of the applications of machine learning in software-defined networks, refer to the comprehensive survey presented in [9].

The exploration of the deployment of 6G networks is anticipated to usher in transformative enhancements to network architectures, with a specific emphasis on the integration of AI technologies. Reference [10] delves into the novel concept of knowledge-defined networking (KDN), wherein network intelligence is concentrated in the knowledge plane, achieved through a fusion of SDN, network telemetry, and ML algorithms. Notably, this paper underscores the utility of programming protocol-independent packet processors (P4), a technology facilitating SDN networks, and underscores the significance of in-band network telemetry (INT) for furnishing real-time network insights. Furthermore, it establishes a link between P4-SDN network architecture and reinforcement learning (RL), illustrating how network components and established techniques can be aligned with RL principles. The manuscript also delves into the potential of AI-driven network orchestration and expounds upon the conceptualization of networks as AI-based systems. While the work outlined in [10] offers a comprehensive framework for innovative mobile networks, it suggests the utilization of INT for latency measurement in networks. However, it is crucial to note that this framework remains in the proposal stage, lacking the implementation details or simulation results presented in this paper.

The study presented in [11] proposes a strategy to tackle the challenge of unpredictable topology states in Flying Ad Hoc Networks (FANET). The authors of this paper deployed an AI algorithm capable of discerning patterns in unmanned aerial vehicle (UAV) mobility, anticipating potential disconnections and proactively initiating rerouting or forwarding algorithms. The paper introduces a case of a software-defined FANET that offers wireless INT to an AI-equipped edge node situated at the ground station. It elaborates on the design of the subsystems housing the AI process and illustrates how a machine learning model can identify critical network situations without reliance on intricate neural networks.

In [12], the authors introduced integration of SDN with INT and deep reinforcement learning (DRL) to autonomously manage and enhance network performance. A QoS-routing use case and preliminary experimental evidence are presented to demonstrate the feasibility of the proposed paradigm. Additionally, some important challenges that need to be addressed are discussed. The authors advocate that addressing such challenges requires a truly interdisciplinary effort between the research fields of artificial intelligence, network science, and computer networks.

The authors of [13] assessed an ML-based soft-failure localization framework in scenarios involving partial telemetry. The framework, based on an artificial neural network (ANN), is trained using optical signal and noise power models simulating network telemetry across all potential failure scenarios. The ML-based framework demonstrates exceptional performance in partial telemetry scenarios, effectively interpolating missing data. The study also demonstrates that ANN training is expedited by principal component analysis and can be conducted using cloud-based services. Additionally, the authors emulated the evaluated ML-based framework in a software-defined networking-based setup using the gRPC Network Management Interface protocol for streaming telemetry.

In the study conducted by Faheem et al. [14], the authors explored a spectrum of machine learning techniques dedicated to estimating the resource requirements of intricate network entities, particularly Virtual Network Functions (VNFs) within a software-defined networking environment. Their focus primarily centered on deciphering the resource demands of VNFs, notably the central processing unit (CPU) consumption during the processing of input traffic. The experiments conducted in their research not only underscored the ML models' aptitude for learning the intricate behaviors of VNFs but also showcased their efficacy in accurately modeling the resource requirements. The findings put forth by the authors suggest that ML techniques can serve as highly effective tools for modeling the resource needs of diverse VNFs, providing valuable insights into optimizing resource allocation and enhancing the overall efficiency of network environments.

The study conducted by Alshahrani et al. [15] provides a comprehensive examination of the complexities introduced by the smart city initiative, characterized by a myriad of specifications and a diverse user base with distinct requirements. To address the challenges arising from this dynamic environment, the authors propose an innovative system that integrates SDN security controllers and ML models with optimization techniques. This strategic combination aims to effectively mitigate the impact of prevalent Distributed Denial of Service (DDoS) attacks on smart cities. The proposed approach is built upon an SDN infrastructure supported by security controllers, constituting a proactive line of defense against potential threats. Additionally, the detection mechanism embedded in the ML model, optimized for enhanced performance, plays a pivotal role in identifying and neutralizing common DDoS attacks within smart city networks. This dual-layered security strategy demonstrates the proposed system's capability to protect against evolving cybersecurity threats. Furthermore, Alshahrani et al. advocate for the implementation of binary classification as a crucial component of their proposed system. The adoption of this classification method not only enhances the efficiency of attack detection but also results in a commendable level of accuracy.

Given the increasing prevalence of Internet of Things (IoT) devices connected to the Internet, the annual rise in IoT-based attacks has prompted a need for more effective solutions. Existing approaches may struggle to sufficiently mitigate these attacks, especially

in network environments supporting both traditional and IoT protocols, and utilizing a centralized architecture like SDN. The study in [16] introduces a long short-term memory (LSTM)-based approach for detecting network attacks within IoT networks using an SDN-supported intrusion detection system. The performance of the machine learning and deep learning model is evaluated across two SDNIoT-focused datasets. Additionally, an LSTM-based architecture is proposed for the effective multiclass classification of network attacks in IoT networks. The evaluation demonstrates the model's effectiveness in identifying and classifying attacks, achieving a high level of accuracy. This study also employs various visualization methods to comprehend dataset characteristics and visualize embedding features.

Various models have been employed in the literature to identify anomalies. For instance, the paper presented in [17] proposes the Two-Step Graph Convolutional Neural Network (TS-GCN) framework. This framework, incorporating resampling techniques and adopting a streamlined architecture, establishes itself as the benchmark for addressing the identified problem. When applied to a specific satellite model, TS-GCN demonstrates notable success in state recognition and prediction accuracy. In comparison to established models, TS-GCN showcases considerable enhancements in state recognition accuracy. The conclusion suggests that TS-GCN, with its streamlined architecture and applicability for on-orbit deployment, holds promise for improved assessment and anomaly detection in satellite systems.

The authors of [18] explore the use of Markov models for anomaly detection in the Healthcare Internet of Things (HIoT). The proposed method leverages the simplicity, interpretability, and a well-developed theory of Markov models to enhance cybersecurity in HIoT. By evaluating the method using the ToN_IoT dataset, the paper aimed to address security concerns and contribute to safeguarding patients' wellbeing in healthcare services.

In [19], the authors delve into the application of undirected probabilistic graphical models, specifically the residual Gauss–Markov random field, for characterizing cloud telemetry. Acknowledging the complexities of cloud systems, the study proposes a unique data model and outlines an efficient estimation procedure. The primary focus is on anomaly detection and localization, demonstrated through experiments in synthetic and small-scale software system environments. The article underscores the computational attractiveness of fitting the model and addresses practical considerations in cloud system structure. However, it also highlights the challenge of validating anomaly detection techniques under the constraints of real-world production cloud systems.

The work presented in [20] addresses the challenges of anomaly detection in satellite telemetry data, proposing a novel model based on Bayesian deep learning. The model leverages Monte Carlo Dropout on a long short-term memory network (LSTM) and establishes the Bayesian LSTM for effective anomaly detection without domain knowledge. The research introduces the concept of uncertainty measures, including Monte Carlo Sampling Variance, Prediction Entropy, and Mutual Information, to enhance anomaly detection capabilities. Additionally, the study explores these uncertainties further and employs a variational auto-encoder (VAE) to re-evaluate high-uncertainty samples, improving the model's robustness on imbalanced datasets. The experimental results demonstrate the effectiveness of the proposed model, showcasing its superior performance over traditional neural networks and other Bayesian neural networks in handling imbalanced datasets.

3. Mechanism

In this chapter, we introduce a sophisticated machine learning mechanism that leverages the power of deep neural networks for the purpose of anomaly identification within the realm of SDNs. This innovative approach represents a significant stride toward enhancing the intelligence and adaptability of network management systems, particularly in the context of dynamic and programmable SDNs.

The proposed dynamic telemetry mechanism in network automation provides significant advantages through the continuous collection and real-time analysis of data from

network devices. This approach provides immediate visibility into the network's state, enabling prompt detection and response to anomalies. The adaptability of dynamic telemetry allows for efficient resource allocation, focusing monitoring efforts when needed most and reducing unnecessary overhead. This dynamic approach enhances the accuracy of anomaly detection mechanisms, contributing to proactive issue resolution and improved network reliability. By streamlining network management processes and automating routine tasks based on current conditions, dynamic telemetry supports scalable, flexible, and optimized network performance. In essence, it ensures that network automation strategies can adapt to changing conditions and requirements, ultimately contributing to more efficient and reliable network operations.

3.1. Architecture

The illustrated anomaly detection system architecture, as delineated in Figure 1, is underpinned by a multifaceted structure encompassing an SDN controller, an analytical server tailored for large datasets, a DNN model, and a discerning decision making algorithm. This comprehensive architecture synergizes these components to create a robust and intelligent system capable of effectively identifying anomalies in network behavior.

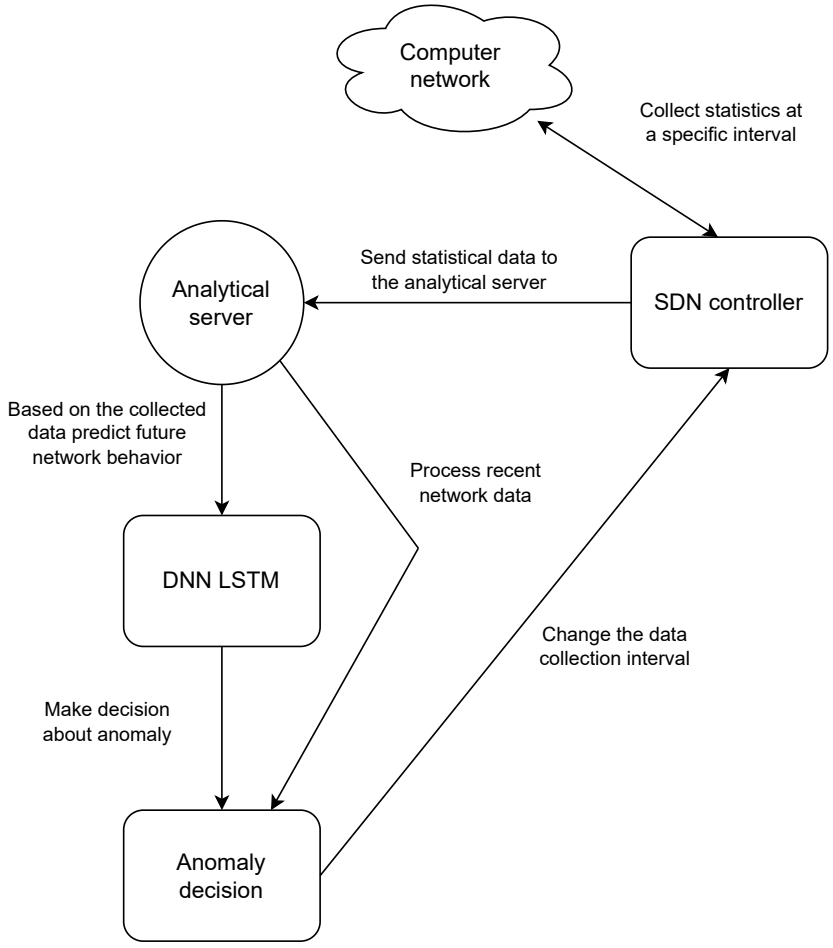

Figure 1. Anomaly detection mechanism operation diagram.

The general operation of the mechanism is as follows:

- The SDN controller queries network devices for their statistics at regular intervals. During normal network operation, this interval is increased to prevent excessive device and link load.
- The data collected by the SDN controller are sent to a database server for later analysis.

- Based on processed data, the DNN model predicts the network's behavior, specifically changes in network traffic (bandwidth) within certain intervals.
- The decision making algorithm then determines the probability that the current traffic deviates from the predicted intervals. If this probability is very low, an anomaly in the network is detected.
- Upon anomaly detection, the SDN controller's configuration is updated with a new, shorter data collection interval, enhancing data analysis accuracy at that moment.
- After detecting normal network traffic again, the algorithm allows updating the interval for statistics retrieval by the SDN controller, returning to its original value.

3.2. Neural Network Model

Artificial neural network models have been used for several decades, but a lack of computational resources have led to the dominance of other machine learning techniques. Nowadays, the trend associated with the type of model strongly focuses on deep learning [21,22].

In this article, a specific type of neural network called long short-term memory (LSTM) is utilized. It is employed for predicting time series and exhibits excellent characteristics in capturing periodic dependencies [21,22].

However, artificial neural networks have a significant disadvantage in practical applications. They are "black-box" models that apply numerous nonlinear dependencies during data processing. This raises questions about whether a given model is correct, to what extent it is correct, and how much trust can be placed in it. In comparison to classical machine learning methods, DNNs do not provide any information about the uncertainty associated with results. In recent years, various ideas have emerged to address this problem, one of which is the introduction of Bayesian neural networks [23]. However, the utility of this solution is questionable due to the increased number of parameters for optimization, resulting in a much more challenging and time-consuming optimization task.

The mechanism proposed by the authors implements a different solution—the application of the "dropout" functionality [24]. This method reduces the risk of overfitting in neural networks by randomly removing individual neurons (both hidden and not) with a certain probability. While this process is commonly used during the training phase of a neural network, implementing it in the testing phase (prediction) allows us to approximate uncertainty values [25].

Our emphasis on DNN LSTM in anomaly detection underscores its effectiveness, particularly within the dynamic context of 6G SDNs. Notably, DNNs have demonstrated superior performance compared to traditional machine learning techniques, including SVM, decision trees, and logistic regression, across diverse applications [26].

3.3. Decision Making Algorithm

The classification process determining whether a certain anomaly occurs in the network at a given moment has been divided into three parts. While anticipated uncertainties (in this specific context, standard deviations from the mean value) can be defined as originating from a normal distribution, the mechanism applies the 3-sigma rule [27].

The algorithm checks whether the current network traffic falls within three intervals: the predicted mean value $\pm$(one, three, four) predicted standard deviations. Conditions for confirming an anomaly are as follows:

- The measured value does not fall within the sigma range for 50 consecutive times (the probability of such an event is approximately 0.68^{50});
- The measured value does not fall within the 3-sigma range for four consecutive times (the probability of such an event is approximately 0.003^{4});
- The measured value does not fall within the 4-sigma range (the probability of such an event is approximately 0.000064).

The first two conditions pertain to detecting rare distributions in data. If a different distribution than usual occurs in network traffic, it likely indicates something unusual. The

third condition, however, relates to detecting values significantly deviating from normal network behavior.

The solution capitalizes on dynamic telemetry's real-time adaptability, enabling the continuous collection and analysis of data for a proactive response to the evolving conditions of next-gen networks. Additionally, the incorporation of deep neural networks elevates anomaly detection, empowering the system to dynamically adjust and respond to intricate patterns and changes in the dynamic network environment. In essence, the solution seamlessly integrates the inherent flexibility of dynamic telemetry with the robust pattern recognition capabilities of deep neural networks to adeptly navigate and mitigate challenges posed by the dynamic nature of next-generation network architectures.

4. Evaluation of the Mechanism

In this chapter, we delve into a comprehensive exposition of the environment employed for conducting our research, providing a detailed overview of the tools utilized in the experimental setup. Additionally, we present the obtained results, accompanied by a meticulous interpretation that sheds light on the intricacies and implications of our findings. This multifaceted exploration aims to offer readers a holistic understanding of the research context, methodology, and observations extracted from the experiments carried out.

4.1. Description of the Utilized Environment

To evaluate the proposed solution, a series of experiments were conducted using the Mininet environment. The routing mechanism, traffic statistics retrieval, and switch polling interval modification were implemented as a module in the Ryu controller [28]. Collected traffic statistics were sent by a custom Python script (referred to as "watchdog") to the Red PNDA environment [8] (Figure 2).

While the experimentation was conducted in an emulated environment, it is important to note that we utilized real traffic traces collected from a campus network. Testing the proposed solution in authentic operating environments like open Wifi, LTE, and 5G networks poses a challenge in acquiring genuine traffic traces due to restricted access as a non-operator. This approach was chosen deliberately to bridge the gap between emulation and real-world scenarios, ensuring that our evaluation closely reflects actual network conditions.

The use of real traffic traces adds a layer of authenticity to our results, enabling a more accurate assessment of the proposed solution's performance in a controlled yet realistic setting. By incorporating genuine network data, we aimed to enhance the reliability and relevance of our findings, providing insights that are more representative of potential real-world implementations.

Neural network training took place using the computational cloud provided by Google https://colab.research.google.com/ (accessed on 14 January 2024) (Google Colaboratory) with GPU support. The framework in which the deep neural networks were created is Keras, based on the TensorFlow environment. TensorFlow is a widely used open source deep learning framework that provides a versatile and cost-effective solution for organizations looking to implement deep neural networks, as it is freely accessible and extensively supported by a robust community. Moreover, contemporary practices often involve the application of transfer learning, where pretrained models are adapted for specific tasks, reducing the need for extensive training from scratch. This approach has become commonplace and significantly mitigates the resource-intensive aspects of developing or acquiring deep neural networks. The training time of the models significantly depended on the specific data collection interval used, ranging from 2 to 3 h.

Test traffic was generated based on daily network traffic collected in the AGH campus network [29]. The generator, written in Python, utilized the iperf3 tool for data transmission. Each generated sample was subjected to additional noise to mimic real network traffic. Anomalies were created by a separate instance of the iperf3 application, generating additional traffic at a rate of 40–45% of the average daily network traffic.

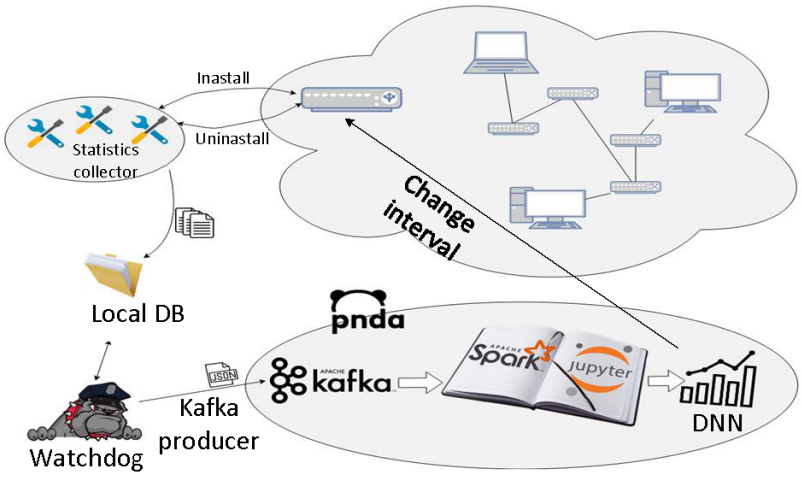

Figure 2. System architecture.

4.2. Parameters of Conducted Experiments

This article endeavors to showcase the relevance and adaptability of the aforementioned mechanism within the realm of computer networks, particularly within the context of evolving 6G mobile networks. The primary objective is to draw a comparative analysis between two distinct implementations, shedding light on how these variations impact the efficiency of the proposed mechanism.

The two following configurations were meticulously scrutinized and compared, each representing a unique set of parameters and data collection intervals.

1. Basic data collection interval: 100 s.
   Shortened data collection interval: 5 s.
   Interval at the input to DNN: 5 h.
   Interval predicted by DNN: 30 min.
2. Basic data collection interval: 300 s.
   Shortened data collection interval: 10 s.
   Interval at the input to DNN: 5 h.
   Interval predicted by DNN: 30 min.

In our anomaly detection system, the learning process is a critical phase that involves training the model to recognize patterns and anomalies within the network data. For our experiments, we opted for a substantial learning time, specifically setting the input interval to the DNN at 5 h. This extended learning duration was chosen to ensure that the model captures nuanced patterns and establishes a comprehensive understanding of the network's normal behavior. The deliberate choice of a 5 h interval aimed to enhance the DNN's adaptability and efficacy in detecting anomalies within the intricate dynamics of 6G SDNs.

The utilization of two different intervals, namely 300 s and 100 s, in our study is deliberate and serves the purpose of evaluating the effectiveness of our anomaly detection solution across varying time scales.

Initially, we chose the 300 s interval to represent a standard data collection timeframe. This duration provides a comprehensive overview of network behavior over a more extended period, allowing us to capture and analyze anomalies with a macro-level perspective. Subsequently, after detecting an anomaly, we opted for a shorter interval of 10 s for the first scenario and 5 s for the second scenario. This adjustment aims to intensify the focus on the post-anomaly period, offering a finer-grained analysis to detect and respond rapidly to changes in network behavior. The shorter intervals enable us to closely monitor the network in the aftermath of an anomaly, facilitating a more immediate and targeted response to emerging issues.

By employing this dual-interval approach, we aim to assess the robustness and adaptability of our anomaly detection solution across both normal and post-anomaly conditions, providing a nuanced understanding of its performance in different temporal contexts.

4.3. Results

In both scenarios, the neural networks adeptly processed minute-level traffic patterns, emphasizing the minute-by-minute granularity crucial for effective anomaly detection. However, the pivotal distinction lay in the data collection intervals orchestrated by the SDN controller. The nuanced exploration of these intervals shed light on their profound impact on the network's throughput characteristics.

The initial case, as illustrated in Figure 3, employed a fundamental data collection interval of 100 s. The outcome was a remarkably jagged throughput characteristic, vividly depicting the network's rapid fluctuations. The decision to detect anomalies and subsequently alter the interval to a mere 5 s introduced a degree of instability into the mechanism. This destabilization becomes more evident in the latter part of the article, underlining the delicate balance that exists when adjusting data collection intervals dynamically.

A closer examination of the predicted sigma interval, as portrayed in Figure 4, provides insights into the neural network's accuracy with varying data collection intervals. Notably, longer intervals yielded a more accurate representation of the network's expected behavior. Despite a conspicuous deviation at the beginning and end of the day, the overall trend indicated greater fidelity to the anticipated distribution. The inherent smoothing effect associated with longer intervals resulted in a more consistent throughput characteristic throughout the day.

This observed consistency could potentially serve as a rationale for considering longer data collection intervals, particularly in environments where network stability is paramount. The smoother trajectory could contribute to a reduction in false anomaly detection decisions, offering a more reliable basis for discerning genuine network anomalies from regular fluctuations. Nevertheless, this preference must be carefully weighed against the need for real-time responsiveness, as prolonged intervals may delay the detection of rapidly evolving network conditions. The optimal choice of data collection intervals, therefore, hinges on a nuanced understanding of the specific network dynamics and the desired trade-off between responsiveness and stability.

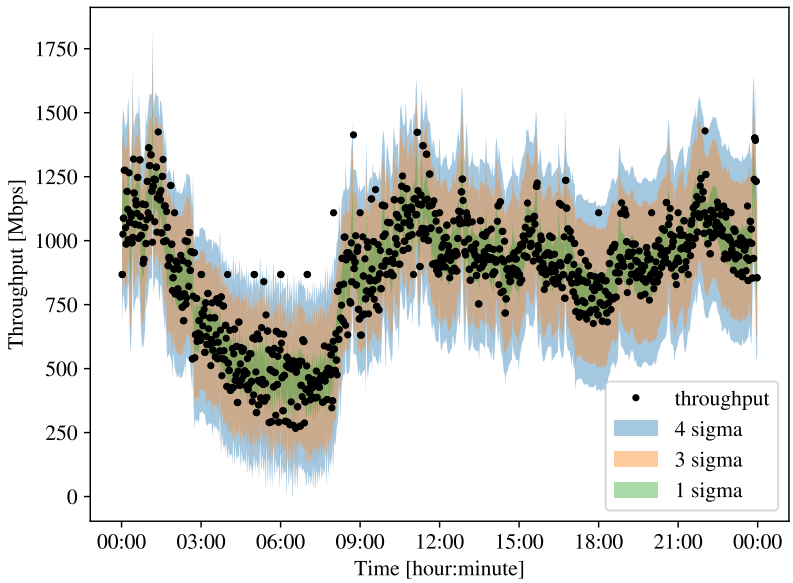

Figure 3. Traffic distribution for a 100 s interval along with predicted sigma intervals.

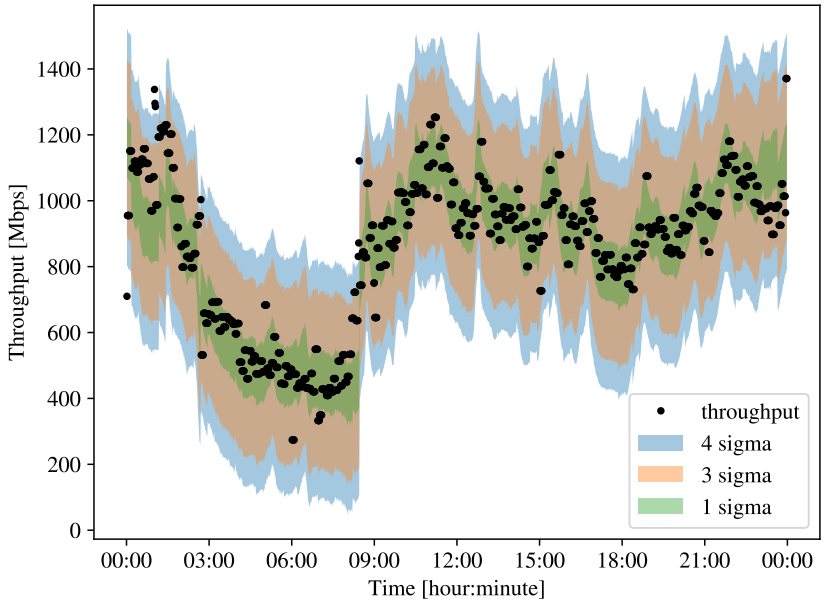

Figure 4. Traffic distribution for a 300 s interval along with predicted sigma intervals.

Figure 5 provides a comprehensive view of the distribution of falsely detected anomalies in comparison to correctly classified traffic samples within specific sigma intervals. This analysis pertains to two distinct cases observed over a single day of network traffic. Notably, the significance of this distribution lies in understanding the trade-offs associated with different data collection intervals. The implementation of shorter time intervals introduces a dynamic element to anomaly detection. While it allows for more frequent assessments of the network state, it also escalates the potential for incorrect classifications. The granularity of shorter intervals exposes the detection mechanism to transient fluctuations and noise in the network, which may lead to a higher incidence of false positives. This heightened risk is evident in Figure 5, where the number of falsely detected anomalies tends to rise within shorter sigma intervals.

Conversely, longer data collection intervals, as depicted in the discussed cases, showcase a relatively more stable distribution. While they may exhibit a slower response to sudden changes in network behavior, the extended time span between assessments provides a broader context for anomaly detection. This context, combined with the inherent smoothing effect of longer intervals, contributes to a more accurate representation of the network's normal state. Consequently, the trade-off lies in the potential delay in detecting anomalies versus a more resilient mechanism that reduces the likelihood of misclassifications.

The reader may naturally contemplate the extent to which alterations in the data collection interval and the presence of anomalies in the input data could impact the predictive capabilities of the neural network. particularly considering that the network was pretrained on a specific dataset derived from a distinct time interval. The investigation reveals that the neural network built upon a longer data collection interval proves to be more susceptible to variations in input data.

While the anticipated traffic pattern generally persists both before and during anomalies, it becomes apparent that the average value deviates from the established norm (see Figure 6). In contrast, the second configuration, featuring a shorter interval, demonstrates a more resilient and stable behavior. Despite anomalies introduced into the neural network input, the predicted time series exhibits a traffic distribution strikingly similar to the data without anomalies (refer to Figure 7).

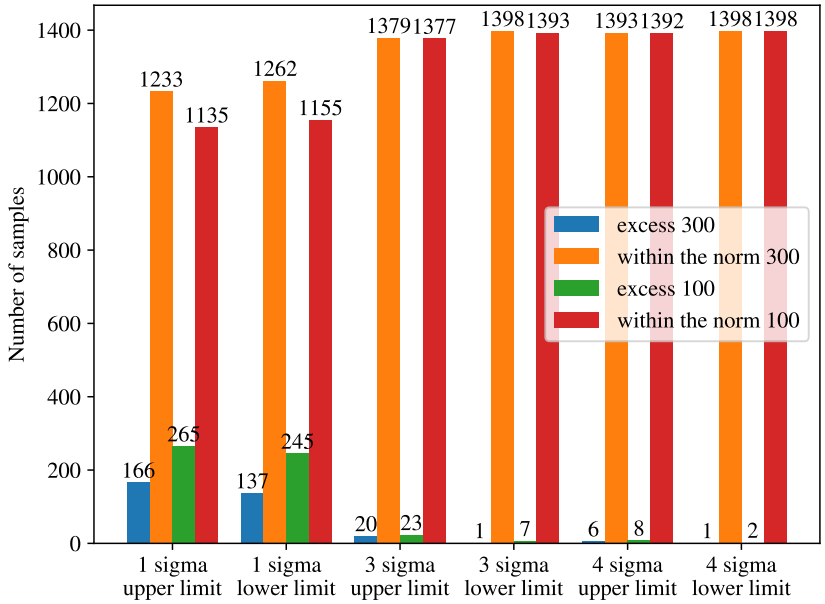

Figure 5. Distribution of misclassified anomalies throughout the day for two cases.

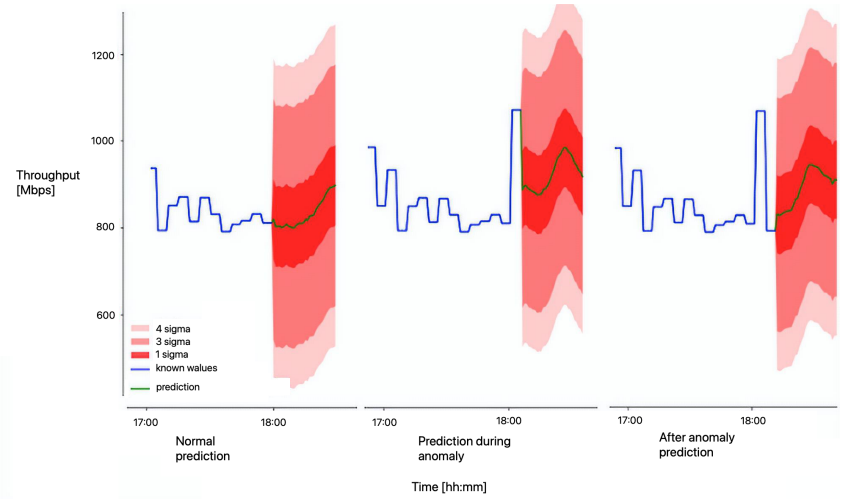

Figure 6. Predicted traffic for a 300 s interval (3 cases).

These findings may suggest a nuanced preference for the utilization of shorter data collection intervals, subsequently fed into the neural network. This approach appears to offer a more robust and adaptable response to changes in network conditions and the presence of anomalies. The ability to maintain a consistent traffic pattern, even in the face of anomalies, underscores the potential effectiveness of shorter intervals in enhancing the network's resilience and adaptability. Further exploration and validation of these observations could contribute to refining anomaly detection strategies and optimizing network management practices in dynamic computing environments.

The selection of longer measurement intervals, such as 300 s, may introduce the risk of overlooking certain anomalies due to the smoothing effect over time. In these extended intervals, network behavior is observed and recorded at a less frequent rate, potentially resulting in the omission of short-term irregularities or rapid changes in traffic patterns. This smoothing effect may lead to a more generalized representation of network stability, but it may also conceal transient anomalies that manifest within shorter timeframes.

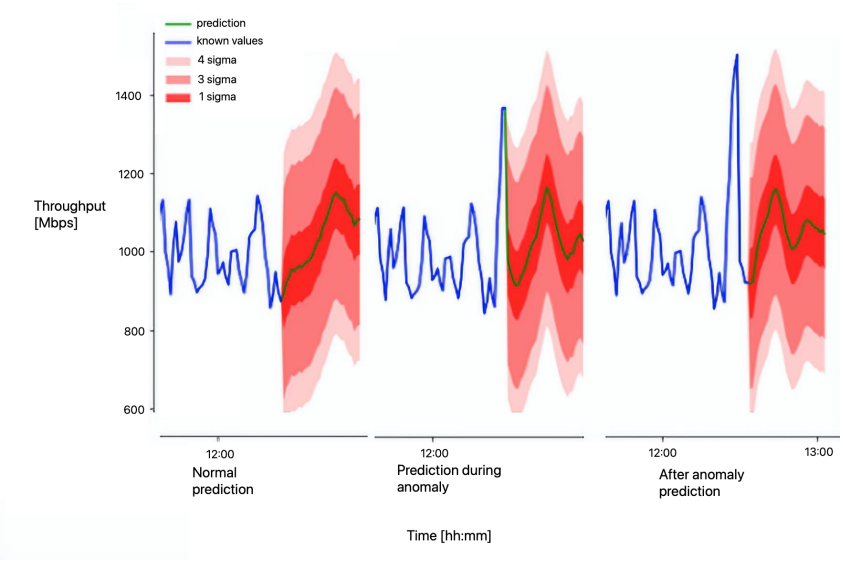

Figure 7. Predicted traffic for a 100 s interval (3 cases).

On the other hand, the use of shorter measurement intervals, such as 5 or 10 s, not only captures transient anomalies more effectively but also introduces an increased frequency of statistics retrieval. This heightened frequency implies a more frequent exchange of information and data between network devices, potentially leading to a higher volume of traffic on the network. While shorter intervals provide a more detailed and immediate observation of network dynamics, it is essential to consider the associated trade-off of increased network traffic. This elevated traffic load may result in a higher demand for computational resources and network bandwidth due to the more frequent collection and processing of statistics. Therefore, the decision to employ shorter measurement intervals involves a careful consideration of the network's capacity to handle the increased data flow without compromising overall performance.

Our proposed DNN-based anomaly detection system exhibits distinct advantages over traditional ML solutions. Traditional ML models, such as support vector machines (SVM), decision trees, and logistic regression, often rely on handcrafted feature engineering, requiring a priori knowledge of specific network characteristics. In contrast, our DNN approach excels in automatically learning intricate patterns and representations from raw data, eliminating the need for extensive feature engineering. This not only streamlines the model development process but also enhances adaptability to the dynamic and evolving nature of 6G software-defined networks.

Our proposal introduces an additional advantage in terms of parameterization compared to traditional ML solutions. In traditional ML models, manually tuning and optimizing parameters, a process known as hyperparameter tuning, is often a time-consuming and labor-intensive task that requires domain expertise. On the contrary, one of the inherent strengths of DNNs lies in their ability to learn and adapt their own internal representations, effectively automating the parameterization process. This self-optimization capability not only reduces the dependency on expert-defined hyperparameters but also facilitates a more agile response to the complex 6G networks. This represents a notable advantage over traditional ML solutions, where manual parameter tuning can be a bottleneck, especially in dynamic network environments.

Moreover, the inherent capacity of DNNs to capture complex relationships and dependencies within data makes them particularly well suited for anomaly detection in the sophisticated network environments envisioned in 6G. Traditional ML models may struggle to grasp the nuances of intricate network behaviors and may exhibit limitations in scalability and generalization. Our DNN-based approach, by leveraging deep learning capabilities, showcases a higher potential for discerning subtle anomalies and adapting

to the intricacies of 6G SDNs, offering a more robust and forward-looking solution for anomaly detection.

The utilization of DNNs for anomaly detection in 6G SDNs may entail certain limitations. One notable concern involves the computational complexity associated with deep neural networks, particularly for complex models, which can lead to extended processing times and increased resource utilization. Additionally, the reliance on substantial amounts of labeled training data poses a challenge, especially in the context of 6G scenarios where obtaining and curating such datasets may be inherently difficult. The real-time processing demands of deep learning models present another potential drawback, introducing latency that could impact the applicability of the solution in highly dynamic 6G networks. Sensitivity to hyperparameter selection, challenges in generalizing the solution across diverse scenarios, and the inherent "black-box" nature of deep neural networks, limiting interpretability, are further aspects that merit consideration. Despite these potential limitations, the utilization of our proposed approach with DNNs remains valuable for anomaly detection in 6G SDNs.

5. Conclusions

This article presents a leading trend in machine learning to address the partial automation of anomaly detection in computer networks. Two scenarios employing distinct time intervals for data collection were introduced and meticulously compared. The conducted experiment provided valuable insights, demonstrating the potential of deep neural networks in conjunction with software-defined networks for establishing reactive alert systems in 6G networks.

Our dynamic anomaly detection system in the context of 6G networks introduces practical implementations crucial for enhancing network security. By proactively identifying potential threats, the system ensures a robust defense mechanism against security breaches. Moreover, its adaptive approach to data collection intervals optimizes resource utilization, contributing to the efficient performance of 6G networks. This adaptability also facilitates swift incident response, providing timely and precise information for decision making during security incidents. Overall, these practical implementations underscore the system's effectiveness in fortifying the security posture of 6G networks in real-world scenarios.

It is noteworthy that the experiment relied on pretrained DNNs, emphasizing the significance of leveraging prior knowledge and models for effective implementation. However, the study also opens avenues for further research, particularly in the realm of continuous learning (online learning). This approach could enhance the accuracy of predicting the evolving behavior of computer networks as they scale and encounter new challenges.

These findings underscore the promising synergy between advanced machine learning techniques and the dynamic landscape of computer network management. As technology continues to evolve, the exploration of innovative methodologies and adaptive learning approaches will be crucial for developing robust, efficient, and scalable anomaly detection systems. This research serves as a foundation, and future studies may delve deeper into refining and extending these approaches to meet the evolving demands of modern network security.

In the context of our study, longer intervals, exemplified by the 300 s duration, possess the inherent risk of inadvertently overlooking crucial data points. This prolonged duration may result in a coarser representation of network dynamics, potentially missing short-lived anomalies or rapid changes in traffic patterns. On the other hand, the utilization of shorter intervals, exemplified by the 10 s duration, introduces a heightened frequency of statistics retrieval, intensifying network traffic. While these shorter intervals offer a finer-grained observation of network behavior, the increased frequency of data collection may contribute to a more resource-intensive process, requiring careful consideration of the network's capacity to manage the amplified data flow. This trade-off between temporal granularity and resource utilization underscores the importance of judiciously selecting data collection intervals based on the specific requirements and constraints of the network environment.

We plan to delve into an exploration of various runtime optimization techniques to augment the performance and scalability of our proposed solution. This investigation will include a comprehensive examination of strategies such as algorithmic optimizations, parallel processing, hardware acceleration, and other innovative approaches aiming to optimize the runtime efficiency of our anomaly detection system. By incorporating these enhancements, we anticipate achieving improved performance and scalability, ensuring the effectiveness of our solution in diverse and dynamic network environments.

Author Contributions: Conceptualization, G.R.; methodology, G.R.; software, A.M.; validation, G.R.; formal analysis, G.R.; investigation, G.R.; resources, G.R.; data curation, A.M.; writing—original draft preparation, G.R.; writing—review and editing, G.R. and P.C.; visualization, G.R. and A.M.; supervision, P.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Research Institute, grant number POIR.04.02.00-00-D008/20-01, on "National Laboratory for Advanced 5G Research" (acronym PL–5G) as part of the Measure 4.2 Development of modern research infrastructure of the science sector 2014–2020 financed by the European Regional Development Fund.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
ANN	Artificial Neural Network
DNN	Deep Neural Network
DRL	Deep Reinforcement Learning
INT	In-Band Network Telemetry
IoT	Internet of Things
IPFIX	Internet Protocol Flow Information Export
KDN	Knowledge-Defined Networking
LSTM	Long Short-Term Memory
ML	Machine Learning
P4	Programming Protocol-Independent Packet Processors
NETCONF	Network Configuration Protocol
RL	Reinforcement Learning
SDN	Software-Defined Network
SNMP	Simple Network Management Protocol
SVM	Support Vector Machines
UAV	Unmanned Aerial Vehicle
VNF	Virtual Network Function

References

1. Fernandes, G.; Rodrigues, J.J.; Carvalho, L.F.; Al-Muhtadi, J.F.; Proença, M.L. A Comprehensive Survey on Network Anomaly Detection. *Telecommun. Syst.* **2019**, *70*, 447–489. [CrossRef]
2. *Cisco Annual Internet Report (2018–2023) White Paper*; Technical Report; Cisco: San Jose, CA, USA , 2023.
3. *2022 Global Networking Trends Report*; Technical Report; Cisco: San Jose, CA, USA, 2022.
4. *2023 Global Internet Phenomena Report*; Technical Report, Sandvine Intelligent Broadband Networks; Sandvine Inc.: Waterloo, ON, Canada, 2022.
5. *Ericsson Mobility Report*; Technical Report; Ericsson: Stockholm, Sweden, 2022.
6. Karakus, M.; Durresi, A. A Survey: Control Plane Scalability Issues and Approaches in Software-Defined Networking (SDN). *Comput. Netw.* **2017**, *112*, 279–293. [CrossRef]
7. Cui, L.; Yu, F.R.; Yan, Q. When big data meets software-defined networking: SDN for big data and big data for SDN. *IEEE Netw.* **2016**, *30*, 58–65. [CrossRef]
8. Project PNDA Web Page. Available online: https://pnda.io (accessed on 15 December 2023).

9. Pathak, Y.; Prashanth, P.V.N.; Tiwari, A. AI Meets SDN: A Survey of Artificial Intelligent Techniques Applied to Software-Defined Networks. In *6G Enabled Fog Computing in IoT: Applications and Opportunities*; Kumar, M., Gill, S.S., Samriya, J.K., Uhlig, S., Eds.; Springer Nature Switzerland: Cham, Switzerland, 2023; pp. 395–412.
10. Zeman, D.; Zelinka, I.; Voznak, M. A Reinforcement Learning Framework for Knowledge-Defined Networking. In Proceedings of the 2023 15th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Ghent, Belgium, 30 October–1 November 2023; pp. 152–156.
11. Uomo, D.; Sgambelluri, A.; Castoldi, P.; De Paoli, E.; Paolucci, F.; Cugini, F. Failure Prediction in Software Defined Flying Ad-Hoc Network. In Proceedings of the Twenty-Fourth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing, New York, NY, USA, 23–26 October 2023; MobiHoc '23; pp. 355–357.
12. Yao, H.; Mai, T.; Xu, X.; Zhang, P.; Li, M.; Liu, Y. NetworkAI: An Intelligent Network Architecture for Self-Learning Control Strategies in Software Defined Networks. *IEEE Internet Things J.* **2018**, *5*, 4319–4327. [CrossRef]
13. Mayer, K.S.; Soares, J.A.; Pinto, R.P.; Rothenberg, C.E.; Arantes, D.S.; Mello, D.A.A. Machine-learning-based soft-failure localization with partial software-defined networking telemetry. *J. Opt. Commun. Netw.* **2021**, *13*, E122–E131. [CrossRef]
14. Faheem, S.M.; Babar, M.I.; Khalil, R.A.; Saeed, N. Performance Analysis of Selected Machine Learning Techniques for Estimating Resource Requirements of Virtual Network Functions (VNFs) in Software Defined Networks. *Appl. Sci.* **2022**, *12*, 4576. [CrossRef]
15. Alshahrani, M.M. A Secure and Intelligent Software-Defined Networking Framework for Future Smart Cities to Prevent DDoS Attack. *Appl. Sci.* **2023**, *13*, 9822. [CrossRef]
16. Chaganti, R.; Suliman, W.; Ravi, V.; Dua, A. Deep Learning Approach for SDN-Enabled Intrusion Detection System in IoT Networks. *Information* **2023**, *14*, 41. [CrossRef]
17. Liu, S.; Qiu, S.; Li, H.; Liu, M. Real-Time Telemetry-Based Recognition and Prediction of Satellite State Using TS-GCN Network. *Electronics* **2023**, *12*, 4824. [CrossRef]
18. Huang, H.C.; Liu, I.H.; Lee, M.H.; Li, J.S. Anomaly Detection on Network Traffic for the Healthcare Internet of Things. *Eng. Proc.* **2023**, *55*, 3.
19. Landolfi, N.C.; O'Neill, D.C.; Lall, S. Cloud Telemetry Modeling via Residual Gauss-Markov Random Fields. In Proceedings of the 2021 24th Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN), Paris, France, 1–4 March 2021; pp. 49–56.
20. Chen, J.; Pi, D.; Wu, Z.; Zhao, X.; Pan, Y.; Zhang, Q. Imbalanced satellite telemetry data anomaly detection model based on Bayesian LSTM. *Acta Astronaut.* **2021**, *180*, 232–242. [CrossRef]
21. Sainath, T.N.; Vinyals, O.; Senior, A.; Sak, H. Convolutional, Long Short-Term Memory, fully connected Deep Neural Networks. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, Australia, 19–24 April 2015; pp. 4580–4584.
22. Khan, A.; Fouda, M.M.; Do, D.T.; Almaleh, A.; Rahman, A.U. Short-Term Traffic Prediction Using Deep Learning Long Short-Term Memory: Taxonomy, Applications, Challenges, and Future Trends. *IEEE Access* **2023**, *11*, 94371–94391. [CrossRef]
23. Barber, D.; Bishop, C. Ensemble learning in Bayesian neural networks. In Proceedings of the Generalization in Neural Networks and Machine Learning, Cambridge, UK, 4–15 August 1997; Generalization in Neural Networks and Machine Learning, Ed.; Springer: Berlin/Heidelberg, Germany, 1998; pp. 215–237.
24. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
25. Caldeira, J.; Nord, B. Deeply uncertain: Comparing methods of uncertainty quantification in deep learning algorithms. *Mach. Learn. Sci. Technol.* **2020**, *2*, 015002. [CrossRef]
26. Taye, M.M. Understanding of Machine Learning with Deep Learning: Architectures, Workflow, Applications and Future Directions. *Computers* **2023**, *12*, 91. [CrossRef]
27. Pukelsheim, F. The Three Sigma Rule. *Am. Stat.* **1994**, *48*, 88–91.
28. Ryu Controller. Available online: https://ryu-sdn.org/ (accessed on 15 December 2023).
29. Jurkiewicz, P.; Rzym, G.; Boryło, P. Flow length and size distributions in campus Internet traffic. *Comput. Commun.* **2021**, *167*, 15–30. [CrossRef]