WBHT: A Generative Attention Architecture for Detecting Black Hole Anomalies in Backbone Networks

Kiymet Kaya*,‡,¶, Elif Ak§ Sule Gunduz Oguducu†,¶,
*Istanbul Technical University, Department of Computer Engineering, Türkiye
†Istanbul Technical University, Department of Artificial Intelligence and Data Engineering, Türkiye
§Memorial University, Canada ‡BTS Group, Türkiye
¶ITU AI Research and Application Center, Istanbul, Türkiye
Email: kayak16@itu.edu.tr, elif.ak@mun.ca, sgunduz@itu.edu.tr

***Abstract*—We propose the Wasserstein Black Hole Transformer (WBHT) framework for detecting black hole (BH) anomalies in communication networks. These anomalies cause packet loss without failure notifications, disrupting connectivity and leading to financial losses. WBHT combines generative modeling, sequential learning, and attention mechanisms to improve BH anomaly detection. It integrates a Wasserstein generative adversarial network with attention mechanisms for stable training and accurate anomaly identification. The model uses long-short-term memory layers to capture long-term dependencies and convolutional layers for local temporal patterns. A latent space encoding mechanism helps distinguish abnormal network behavior. Tested on real-world network data, WBHT outperforms existing models, achieving significant improvements in F1 score (ranging from 1.65% to 58.76%). Its efficiency and ability to detect previously undetected anomalies make it a valuable tool for proactive network monitoring and security, especially in mission-critical networks.**

***Keywords*—generative artificial intelligence, black hole, anomaly detection, self attention, transformer, wasserstein**

I. Introduction

The increasing reliance on communication networks for mission-critical applications, such as emergency services, industrial Internet of Things (IoT), autonomous transportation, and critical infrastructure monitoring, makes anomaly detection a paramount concern in ensuring both network security and reliability [1]. In particular, anomalies like black holes (BH), which silently drop data packets without issuing failure notifications, pose significant threats to these mission-critical infrastructures [2]. Due to their stealthy nature, they can cause prolonged undetected interruptions, severely impacting applications such as real-time monitoring in industrial IoT systems, disrupting command-and-control channels in autonomous transportation systems, and impeding timely communication in emergency response scenarios.

Despite advancements in generative AI and sequential learning, existing solutions still face fundamental limitations in accurately identifying BH anomalies in backbone networks [3]. Traditional autoencoder-based methods struggle to reconstruct complex network traffic, while adversarial learning techniques, such as generative adversarial networks (GANs), often suffer from unstable training and lack structured latent space representations for effective anomaly detection. Furthermore, existing transformer-based approaches, although promising in capturing long-range dependencies, fail to fully exploit spatial-temporal relationships within network traffic data.

Recognizing that each learning architecture possesses distinct advantages, it remains essential to empirically investigate their efficacy in specific anomaly detection contexts. Motivated by these challenges and building upon our prior research that employed an unsupervised convolutional autoencoder (ConvAE) combined with density-based spatial clustering of applications with noise (DBSCAN) [4], this study evaluates the performance of Wasserstein GANs integrated with attention mechanisms specifically for BH anomaly detection. While GANs and transformers have individually or jointly demonstrated robust performance across various anomaly scenarios, including intrusion detection, fraud detection, network traffic irregularities, sensor malfunctions, and industrial IoT anomalies [5], their combined capabilities have yet to be fully explored and validated in the critical case of BH anomalies. To validate the effectiveness of our WBHT, we utilize a substantial dataset in collaboration with an Internet technology provider [1]. The main contributions of this study can be summarized as follows:

- Unlike conventional GAN-based approaches, WBHT integrates transformer architectures to capture temporal dependencies in sequential network traffic.
- WBHT leverages a hybrid approach using Wasserstein-GAN (WGAN) for stable training and improved anomaly score estimation, combined with multi-head attention to refine feature extraction from complex network data.
- WBHT benefits from transformer-based parallelization and Wasserstein loss stability, making it more efficient than traditional GAN-based detection methods, especially in large-scale network monitoring.

II. Literature Review

Recent studies have concentrated on exploring unique characteristics of various anomaly types and leveraging available data through diverse machine learning approaches, ranging

[1]https://www.btsgrp.com

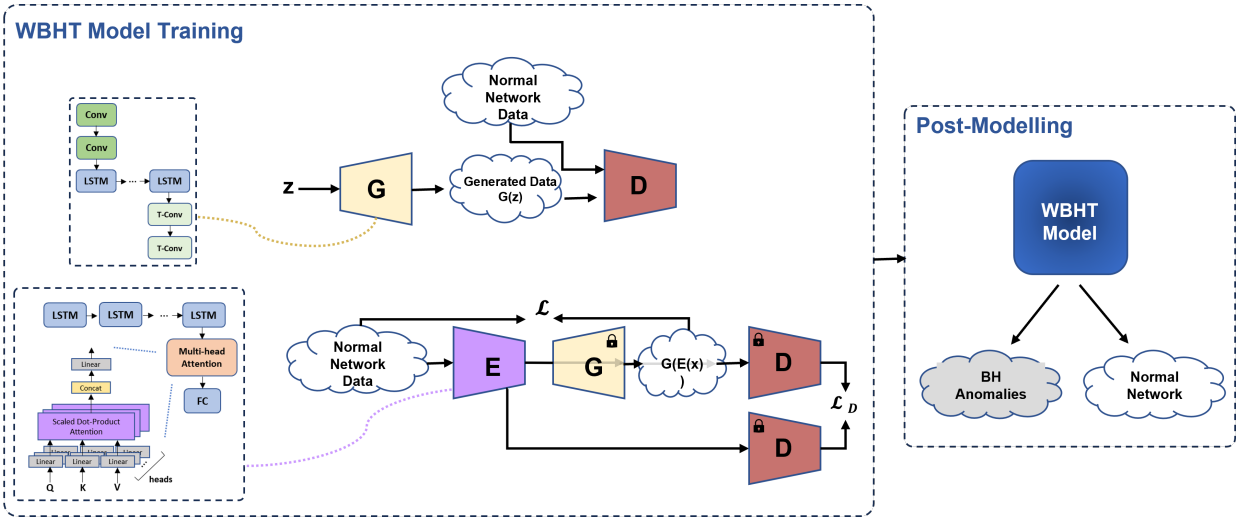

Fig. 1: WBHT: Wasserstein Black Hole Transformer Anomaly Detection Framework.

from supervised and semi-supervised to unsupervised techniques [6].

Autoencoder (AE) models are particularly relevant for unsupervised anomaly detection, as they learn the underlying data structure and flag deviations as anomalies. Various AE architectures have been explored for this purpose, including deep neural networks for network intrusion detection [7], convolutional autoencoders for capturing spatial patterns [8], GRU-based AEs for sequential data [9], and LSTM-AEs tailored to detecting anomalies in wide area networks [10].

While AE models are valuable in unsupervised intrusion detection, recent developments have led to the exploration of GANs, where they employ a generator and discriminator in an adversarial process. This setup enables GANs to detect anomalies by contrasting real and generated data, while also generating new samples that improve the model's ability to distinguish normal from anomalous behavior. Among the notable approaches, DCT-GAN integrates dilated convolutions and Transformers within a GAN framework to improve time series anomaly detection by addressing mode collapse and enhancing generalization [11]. TransEC-GAN combines a Transformer-enhanced GAN with Wasserstein distance and adaptive differential privacy for robust anomaly detection in industrial CPS [12].

Transformers, meanwhile, have advanced anomaly detection by effectively capturing temporal dependencies, outperforming traditional RNNs in sequential tasks. Adformer applies a Transformer-based adversarial framework to IoT sensor data, enhancing sensitivity to subtle anomalies through attention mechanisms [13]. Similarly, Transformer-based GANs have demonstrated superior capability over Autoencoder-based variants in generating high-quality adversarial samples for intrusion and malware detection, underscoring the advantage of attention-driven models in security contexts [14].

GANs combined with Transformer models have also been applied beyond time series and IoT contexts. One approach employs a Transformer-based GAN with Wasserstein GAN-GP for adversarial USB keystroke attack detection, improving robustness and accuracy [15]. Another framework leverages Transformer classifiers trained on GAN-generated data for intrusion detection, achieving state-of-the-art accuracy and enhanced resilience against evolving threats in metaverse security contexts [16].

Despite these advancements, challenges remain in detecting BH anomalies due to the difficulty of obtaining labeled datasets [17]–[20]. As a result, most studies rely on unsupervised or semi-supervised methods with limited or unlabeled data [6]. In this study, we take a different approach by having access to a dataset known to contain exclusively normal traffic and use Wasserstein GANs with multi-head self-attention to learn its representation. This allows the model to detect anomalies as deviations without needing explicit BH examples, providing a practical and effective alternative to fully unsupervised approaches.

III. Methodology

The flow of the proposed WBHT is given in Fig. 1. The model leverages a combination of generative modeling, sequential and adversarial learning, and attention mechanisms to accurately differentiate between normal network traffic and anomalous patterns indicative of BH anomalies. The training phase of WBHT consists of two primary components: the generative phase and the encoder-decoder phase.

The generative phase is based on the WGAN architecture, which improves training stability and alleviates common issues, such as mode collapse, often encountered in vanilla GANs. When comparing loss functions, WGAN uses the Wasserstein distance (WD) (also known as Earth Mover's Distance) as part of the loss function to learn the probability distribution, whereas GAN uses Jensen-Shannon (JS). JS takes values from 0 to $\log 2$, whereas WD addresses the issue that when JS distributions do not overlap, the derivative becomes 0 in the region where $\log 2$ applies. WGAN's WD is achieved by enforcing the Lipschitz constraint on the discriminator through weight-clipping [21].

In the generative phase of WBHT, minimizing WD ensures a smooth and more meaningful optimization landscape, allowing the generator G to produce high-quality network traffic representations that align with the normal data distribution. On the other hand, the discriminator D is tasked with distinguishing between the real and generated data, optimizing this process through the WD, and ensuring stable and efficient adversarial learning. Here, G combines LSTM layers for capturing long-term dependencies and convolutional layers for extracting local temporal patterns. The convolutional layer C_e^i is formulated as in Equation 1, where W_i and b_i represent learnable weights and biases, while ReLU introduces non-linearity. The deconvolution layer T_d^i is formulated as in Equation 2, focusing on reconstructing realistic sequences, where $W_i^{'}$ and $b_i^{'}$ denote transposed convolution filters and biases, ensuring temporal consistency in the generated data.

$$C_e^i(x_i) = \text{ReLU}(W_i * x_i + b_i), \quad i = 0, 1, .., N \tag{1}$$

$$T_d^i(y_i) = \text{ReLU}(W_i^{'} \otimes y_i + b_i^{'}), \quad i = 0, 1, .., N \tag{2}$$

WGAN training yields a generator $G(z)$ that maps from the latent space Z (noise) to the data space X (training data), but does not provide the inverse mapping from X to Z, which is essential for anomaly detection in WBHT. To address this limitation, an encoder E is introduced, which transforms input network traffic into a compact latent representation. The encoder is trained separately, while keeping the parameters of the pre-trained G and D models fixed, such that it learns the inverse mapping $x \rightarrow z$.

The encoder consists of stacked LSTM layers and a multi-head self-attention (MHSA) [22] mechanism. The LSTM layers capture sequential dependencies, while MHSA enhances the model's ability to focus on relevant time steps. The MHSA module processes three key components: the Query (Q), Key (K), and Value (V) vectors. The Q computes similarity scores between the current time step and all other time steps, while the K provides information about the time steps being compared. The V contains the actual data at each time step. Attention scores are calculated using a dot product similarity function, determining the importance of each time step relative to the current one. These scores are then used to compute a weighted sum of the V vectors, producing a context vector that retains critical information from the entire time series.

The overall loss function for WBHT model optimization is defined as in Equation 3. The first part of the Loss function comes from the Generator, and the second one is related to the Discriminator. Here, $f(\cdot)$ represents the discriminator features extracted from an intermediate layer, n_d is the dimensionality of the intermediate feature representation, and k is a weighting factor balancing the feature residual, and n denotes the total number of time steps in the input sequence.

$$\mathcal{L} = \frac{1}{n}\sqrt{\sum_{t=1}^{n} x - G(E(x))} + \frac{k}{n_d}\sqrt{\sum_{t=1}^{n} f(x) - f(G(E(x)))} \tag{3}$$

The encoder-decoder phase refines the model's ability to represent and reconstruct network traffic. The encoder E transforms input network data into a lower-dimensional latent representation, which G then attempts to reconstruct. The D evaluates the reconstructed data, enhancing the model's ability to detect deviations from learned normal patterns. This dual-stage training process enables WBHT to develop a deep understanding of the statistical characteristics of network traffic, making it robust against subtle and sophisticated BH anomalies.

After training, the WBHT model is deployed for post-modeling anomaly detection. Incoming network traffic is processed through the trained model, which classifies sequences as either normal network activity or BH anomalies. Classification relies on reconstruction errors and D's confidence score. Sequences that significantly deviate from learned normal patterns are flagged as anomalies, enabling the identification of potential BH attacks in communication networks.

IV. Experimental Results

The real ISP network data containing BH traffic used in this study is collected, processed and cleaned according to procedures explained in our previous study [4]. Beyond the previous study, this approach gives a semi-supervised learning performed with all candidate forecasting models and the BH labeled samples were used only in the evaluation of the test set results. The subsections elaborate on the following aspects: Section IV-A presents the formation of the WBHT, while Section IV-B provides a comparative analysis of state-of-the-art models with WBHT for BH detection.

A. Formation of the Wasserstein Black Hole Transformer

We conducted a series of experiments to develop the WBHT model, as summarized in Table I. Our primary objectives are: (i) to compare the performance of WGAN and vanilla GAN, (ii) to identify the most effective E model, and (iii) to determine the best-performing G model. These experiments select an optimal architecture that enhances model robustness and generalization.

We evaluated various E and G architectures, including Fully Connected Neural Networks (FCNN), Conv, LSTM, ConvLSTM, and Transformer-based architectures incorporating Multi-Head Attention mechanisms (ConvMultiHead, LSTMMultiHead). To assess model performance, we utilized several key evaluation metrics: Detection Rate (DR), False Alarm Rate (FAR), F1 Score (F1), and Accuracy (Acc). Among these, the F1 Score was prioritized for model selection due to its balanced consideration of precision and recall, which is crucial for mitigating the trade-off between false positives-negatives.

Based on the results, the WGAN consistently outperformed the vanilla GAN across multiple configurations. Additionally, among E architectures, the LSTMMultiHead encoder demonstrated superior performance, particularly in capturing temporal dependencies and complex patterns. For G architectures, the ConvLSTM model achieved the best results, effectively balancing spatial and temporal feature extraction. As a result of these experiments, WBHT was formed as a model incorporating WGAN as the generative AI method, LSTMMultiHead as the E, and ConvLSTM as the G. This combination leverages the strengths of both Transformer-based

TABLE I: WBHT Performance: Evaluating GAN vs. WGAN with Different E and G Architectures

WGAN

	G: FCNN				G: Conv.				G: LSTM			
	DR	**FAR**	**F1**	**Acc.**	**DR**	**FAR**	**F1**	**Acc.**	**DR**	**FAR**	**F1**	**Acc.**
E: FCNN	0.9447	0.0842	0.9174	0.9253	0.9430	0.0804	0.9194	0.9272	0.9490	0.0825	0.9202	0.9278
E: Conv	0.9481	0.0825	0.9199	0.9275	0.9498	0.0817	0.9211	0.9286	0.9124	0.0747	0.9118	0.9211
E: LSTM	0.9498	0.0842	0.9194	0.9269	0.9473	0.0870	0.9164	0.9242	0.9354	0.0895	0.9101	0.9186
E: ConvLSTM	0.9473	0.0813	0.9205	0.9281	0.9379	0.0903	0.9105	0.9189	0.9179	0.0527	0.9199	0.9275
E: ConvMultiHead	0.9515	0.0813	0.9221	0.9294	0.9379	0.0767	0.9201	0.9281	0.9498	0.0796	0.9226	0.9300
E: LSTMMultiHead	0.9507	0.0796	0.9229	0.9303	0.9405	0.0796	0.9190	0.9269	0.9481	0.0804	0.9214	0.9289

	G: ConvLSTM				G: ConvMultiHead				G: LSTMMultiHead			
	DR	**FAR**	**F1**	**Acc.**	**DR**	**FAR**	**F1**	**Acc.**	**DR**	**FAR**	**F1**	**Acc.**
E: FCNN	0.9252	0.0957	0.9019	0.9111	0.9456	0.0800	0.9207	0.9283	0.9311	0.0920	0.9068	0.9156
E: Conv	0.9515	0.0829	0.9209	0.9283	0.9371	0.0957	0.9064	0.9150	0.9515	0.0821	0.9215	0.9289
E: LSTM	0.9532	0.0792	0.9242	0.9314	0.9456	0.0837	0.9181	0.9258	0.9558	0.0817	0.9234	0.9306
E: ConvLSTM	0.9490	0.0767	0.9243	0.9317	0.9439	0.0862	0.9157	0.9236	0.9532	0.0792	0.9242	0.9314
E: ConvMultiHead	0.9515	0.0821	0.9215	0.9289	0.9515	0.0817	0.9218	0.9292	0.9524	0.0804	0.9230	0.9303
E: LSTMMultiHead	0.9575	0.0788	0.9261	0.9331	0.9422	0.0846	0.9162	0.9242	0.9532	0.0780	0.9250	0.9322

GAN

	G: FCNN				G: Conv.				G: LSTM			
	DR	**FAR**	**F1**	**Acc.**	**DR**	**FAR**	**F1**	**Acc.**	**DR**	**FAR**	**F1**	**Acc.**
E: FCNN	0.8478	0.0802	0.8826	0.8964	0.9354	0.0858	0.9127	0.9211	0.9498	0.0837	0.9197	0.9272
E: Conv	0.9184	0.0932	0.9010	0.9106	0.9022	0.0982	0.8913	0.9019	0.9405	0.0870	0.9138	0.9219
E: LSTM	0.9396	0.0800	0.9184	0.9264	0.9507	0.0825	0.9209	0.9283	0.9439	0.0854	0.9162	0.9242
E: ConvLSTM	0.9379	0.0854	0.9140	0.9222	0.8963	0.1011	0.8870	0.8981	0.9473	0.0813	0.9205	0.9281
E: ConvMultiHead	0.8997	0.1019	0.8878	0.8986	0.9405	0.0809	0.9182	0.9261	0.9515	0.0825	0.9212	0.9286
E: LSTMMultiHead	0.9388	0.0982	0.9053	0.9139	0.8520	0.0759	0.8872	0.9006	0.9507	0.0821	0.9212	0.9286

	G: ConvLSTM				G: ConvMultiHead				G: LSTMMultiHead			
	DR	**FAR**	**F1**	**Acc.**	**DR**	**FAR**	**F1**	**Acc.**	**DR**	**FAR**	**F1**	**Acc.**
E: FCNN	0.9303	0.0870	0.9099	0.9186	0.9490	0.1341	0.8844	0.8931	0.9498	0.0833	0.9200	0.9275
E: Conv	0.9362	0.0788	0.9180	0.9261	0.9515	0.0895	0.9162	0.9239	0.9490	0.0796	0.9223	0.9297
E: LSTM	0.9532	0.0825	0.9218	0.9292	0.9464	0.0833	0.9187	0.9264	0.9532	0.0804	0.9233	0.9306
E: ConvLSTM	0.9524	0.0813	0.9224	0.9297	0.9507	0.0821	0.9212	0.9286	0.9507	0.0784	0.9238	0.9311
E: ConvMultiHead	0.9481	0.0776	0.9234	0.9308	0.9490	0.0813	0.9211	0.9286	0.9473	0.0780	0.9228	0.9303
E: LSTMMultiHead	0.9532	0.0796	0.9239	0.9311	0.9473	0.0821	0.9199	0.9275	0.9515	0.0796	0.9232	0.9306

encoders and spatiotemporal feature extractors, ultimately improving overall model performance.

B. WBHT Black Hole Detection Evaluation

To validate the effectiveness of our proposed WBHT model, we compare its performance against state-of-the-art baseline models, as explained in the following:

AE consists of only $Linear$ layers. This model has four $Linear$ layers, the first two used for encoding and the last two for decoding, which respectively transform {#features, 16, 8, 16, #features}.

Con-AE has 3 $Conv1D$ and 3 $T-Conv1D$ layers, the $filters$ are 32, 16, 8, 16, 32, 1 and $dropout\ rates$: 0.2, 0, 0.2, 0, 0 from the left side, respectively. **LSTM-AE** consists of $LSTM$ encoder and decoders with the hyperparameter values: $hiddensize$: 8.

ConvLSTM-AE is Con-AE's version with an $LSTM$ layer added between encoding and decoding.

ConvMultiHead-AE is Con-AE's version with MHSA mechanisms with $numberofattentionheads$: 4 after first $Conv1D$ and $T-Conv1D$. MHSA allows the model to attend to different parts of the time series simultaneously, capturing long-range dependencies and temporal patterns in the data.

LSTMMultiHead-AE is LSTM-AE's version with MHSA mechanisms with $numberofattentionheads$: 4

AnoGAN [23] is an anomaly detection model based on vanilla GAN with a single G and a single D. AnoGAN is trained using the JSD loss, which can sometimes cause unstable training. A major drawback of AnoGAN is the absence of an E, requiring an iterative optimization process to find the best latent representation of an input during inference.

MADGAN [24] improves time series anomaly detection by introducing multiple D's but suffers from training instability due to its vanilla GAN's JSD and lacks an E for fast inference like f-AnoGAN. However, since MADGAN still uses a vanilla GAN's JSD loss instead of WGAN-GP, training stability can be an issue, particularly in complex datasets. Additionally, MADGAN does not include an E, meaning it does not benefit from the fast inference provided by f-AnoGAN.

f-AnoGAN [25] improves upon AnoGAN by introducing an E network, which maps real input data to the latent space, bypassing the need for optimization, in addition to G and D. f-AnoGAN also replaces JSD loss with WGAN-GP, resulting in more stable training and a better latent space representation.

AutoFormer [26] employs a decomposition-based architecture that separates time series into trend and seasonal components as $X_t = \text{AvgPool}(\text{Padding}(X)), \quad X_s = X - X_t$ where X_t is the smoothed trend component, and X_s captures seasonal variations. Additionally, Autoformer introduces an *Auto-Correlation Mechanism* to identify periodic dependencies

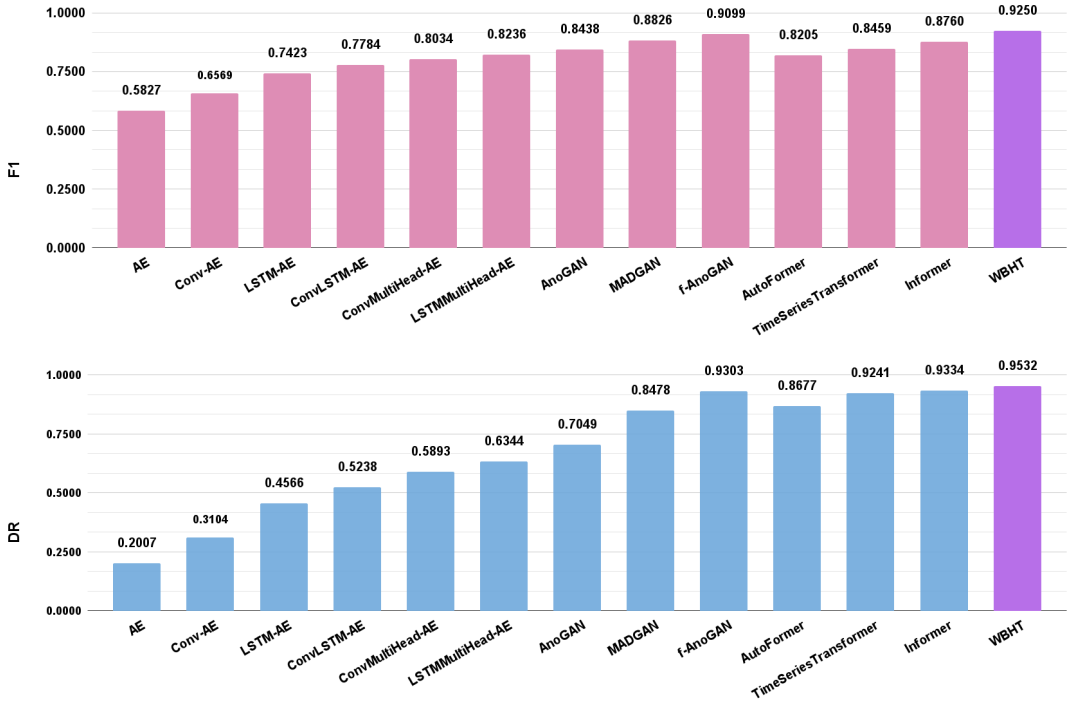

Fig. 2: Comparative Performance Evaluation of BH Detection Models

efficiently.

TimeSeriesTransformer is an encoder-decoder architecture designed for long-term time series forecasting. It leverages self-attention to capture long-range dependencies efficiently, making it suitable for applications like energy forecasting, finance, and health monitoring. Its self-attention mechanism computes dependencies between inputs as $A(Q, K, V) = \text{Softmax}\left((QK^T)/\sqrt{d}\right)V$ where Q, K, and V represent query, key, and value matrices, and d is the dimensionality.

Informer [27] addresses the computational inefficiencies of self-attention by introducing *ProbSparse Self-Attention*, which selectively focuses on the most significant attention scores. The attention mechanism is defined as $A(Q, K, V) = \text{Softmax}\left((Q'K^T)/\sqrt{d}\right)V$, where Q' contains only the top-u queries based on sparsity constraints, reducing complexity.

TABLE II: Performance Benchmarking of WBHT

	DR	FAR	F1	Acc.
AE	0.2007	0.0062	0.5827	0.7347
Conv-AE	0.3104	0.0136	0.6569	0.7656
LSTM-AE	0.4566	0.0190	0.7423	0.8097
ConvLSTM-AE	0.5238	0.0194	0.7784	0.8314
ConvMultiHead-AE	0.5893	0.0309	0.8034	0.8450
LSTMMultiHead-AE	0.6344	0.0330	0.8236	0.8583
AnoGAN	0.7049	0.0499	0.8438	0.8700
MADGAN	0.8478	0.0802	0.8826	0.8964
f-AnoGAN	0.9303	0.0870	0.9099	0.9186
AutoFormer	0.8677	0.2183	0.8205	0.8394
TimeSeriesTransformer	0.9241	0.2485	0.8459	0.8673
Informer	0.9334	0.1922	0.8760	0.8920
WBHT (proposed)	**0.9532**	**0.0780**	**0.9250**	**0.9322**

The results are presented in Table II. The complexity requirements of our model are assessed by benchmarking it against more primitive AE-based methods, namely LSTM-AE, ConvLSTM-AE. These models, while effective in capturing temporal dependencies, struggle with high-dimensional network traffic patterns and lack the adversarial learning necessary to distinguish subtle BH anomalies. Additionally, we investigate the adequacy of incorporating only Transformer-based architectures by comparing WBHT against Transformer-based baselines. While primitive Transformer-based models, such as ConvLSTM-MultiHeadAE, introduce self-attention mechanisms for improved feature extraction and temporal dependencies, they still lack the generative modeling needed for anomaly detection in general. More advanced Transformer architectures, including Informer, AutoFormer, and TimeSeriesTransformer, excel in capturing long-range dependencies in anomalies as an enhancement but fail to effectively localize anomalies. This is mainly due to the key characteristic of BH anomalies, occurring over short, bursty time intervals. Finally, we contrast our approach with advanced generative models that do not incorporate Transformers, such as AnoGAN, f-AnoGAN, and MADGAN. This comparison also enables us to analyze the impact of using WGAN (w/ f-AnoGAN) versus standard GAN architectures (w/ AnoGAN and MADGAN). The closest model to our WBHT approach for BH anomaly detection is obtained in f-AnoGAN model due to its improvement in training stability, unlike other GAN baseline models. However, its performance is slightly lower than that of WBHT, likely due to its lack of structured spatial-temporal awareness. Overall, WBHT successfully integrates WGAN for stable training, LSTM-based encoding for sequential learning, and Multi-Head Attention for fine-grained feature extraction, allowing it to outperform all baseline models. For enhanced

visual interpretability, Fig. 2 provides a comparative performance evaluation in terms of F1 and DR. The results demonstrate that our WBHT model consistently outperforms state-of-the-art alternatives, achieving the best F1 and DR scores. This confirms the effectiveness of our proposed approach in BH detection tasks, surpassing existing methodologies in terms of both accuracy and robustness.

V. Conclusion

In this study, we proposed the WBHT framewotk for BH anomaly detection in communication networks using time series tabular data. Our approach integrates generative modeling, sequential learning, and attention mechanisms, building on WGAN architecture to enhance the model's stability and performance.

Through a series of evaluations, we showed that WBHT outperforms traditional GAN-based models, Transformer-based architectures, and other advanced generative methods in BH detection tasks. Specifically, WBHT's ability to leverage LSTMMultiHead as the encoder and ConvLSTM as the generator enabled the model to effectively capture both spatial and temporal dependencies, making it highly suitable for real-time network anomaly detection. Our findings confirm that the combination of WGAN's stability, the encoder's temporal learning capabilities, and the generator's spatial feature extraction significantly enhances the accuracy and robustness of BH detection.

Acknowledgements

This research is supported by the Scientific and Technological Research Council of Turkey (TUBITAK) 1515 Frontier R&D Laboratories Support Program for BTS Advanced AI Hub: BTS Autonomous Networks and Data Innovation Lab. project number 5239903, TUBITAK 1501 project number 3220892, and the ITU Scientific Research Projects Fund under grant numbers MÇAP-2022-43823 and YESAP-2024-45920.

References

[1] Z. Kaleem, F. A. Orakzai, W. Ishaq, K. Latif, J. Zhao, and A. Jamalipour, "Emerging trends in uavs: From placement, semantic communications to generative ai for mission-critical networks," *IEEE Transactions on Consumer Electronics*, pp. 1–1, 2024.

[2] K. Kaya, E. Ak, S. Bas, B. Canberk, and S. G. Oguducu, "X-cba: Explainability aided catboosted anomal-e for intrusion detection system," in *ICC 2024 - IEEE International Conference on Communications*, 2024, pp. 2288–2293.

[3] M. Polverini, A. Cianfrani, M. Listanti, G. Siano, F. G. Lavacca, and C. C. Campanile, "Investigating on black holes in segment routing networks: Identification and detection," *IEEE Transactions on Network and Service Management*, vol. 20, no. 1, pp. 14–29, 2023.

[4] K. Kaya, E. Ak, E. Ozaltun, L. Maglaras, T. Q. Duong, B. Canberk, and S. G. Oguducu, "Black hole prediction in backbone networks: A comprehensive and type-independent forecasting model," *IEEE Transactions on Network and Service Management*, pp. 1–1, 2025.

[5] M. Sabuhi, M. Zhou, C.-P. Bezemer, and P. Musilek, "Applications of generative adversarial networks in anomaly detection: A systematic literature review," *IEEE Access*, vol. 9, pp. 161 003–161 029, 2021.

[6] B. Al-Musawi, P. Branch, and G. Armitage, "Bgp anomaly detection techniques: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 1, pp. 377–396, 2017.

[7] S. J. Moore, F. Cruciani, C. D. Nugent, S. Zhang, I. Cleland, and S. Sani, "Deep learning for network intrusion: A hierarchical approach to reduce false alarms," *Intelligent Systems with Applications*, vol. 18, p. 200215, 2023.

[8] M. A. Khan and J. Kim, "Toward developing efficient conv-ae-based intrusion detection system using heterogeneous dataset," *Electronics*, vol. 9, no. 11, p. 1771, 2020.

[9] E. Mushtaq, A. Zameer, and R. Nasir, "Knacks of a hybrid anomaly detection model using deep auto-encoder driven gated recurrent unit," *Computer Networks*, vol. 226, p. 109681, 2023.

[10] S. Longari, D. H. Nova Valcarcel, M. Zago, M. Carminati, and S. Zanero, "Cannolo: An anomaly detection system based on lstm autoencoders for controller area network," *IEEE Transactions on Network and Service Management*, vol. 18, no. 2, pp. 1913–1924, 2021.

[11] Y. Li, X. Peng, J. Zhang, Z. Li, and M. Wen, "Dct-gan: Dilated convolutional transformer-based gan for time series anomaly detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 4, pp. 3632–3644, 2023.

[12] J. Liang, Z. Li, and M. Sadiq, "Transec-gan: A transformer-enhanced ids for robust detection and privacy in industrial cps," in *2025 IEEE Wireless Communications and Networking Conference (WCNC)*, 2025, pp. 1–6.

[13] F. Zeng, M. Chen, C. Qian, Y. Wang, Y. Zhou, and W. Tang, "Multivariate time series anomaly detection with adversarial transformer architecture in the internet of things," *Future Generation Computer Systems*, vol. 144, pp. 244–255, 2023.

[14] S. Laudanna, A. Di Sorbo, P. Vinod, C. A. Visaggio, and G. Canfora, "Transformer or autoencoder? who is the ultimate adversary for attack detectors?" *Int. J. Inf. Secur.*, vol. 24, no. 1, Nov. 2024. [Online]. Available: https://doi.org/10.1007/s10207-024-00934-9

[15] A. K. Chillara, P. Saxena, and R. R. Maiti, "Transformer-based gan-augmented defender for adversarial usb keystroke injection attacks," ser. ICDCN '25. New York, NY, USA: Association for Computing Machinery, 2025, p. 94–103. [Online]. Available: https://doi.org/10.1145/3700838.3700871

[16] Y. Djenouri, A. Nabil Belbachir, A. Belhadi, T. Michalak, and G. Srivastava, "Next-gen metaverse security through intrusion detection enhanced by transformers and gans," *IEEE Internet of Things Journal*, vol. 12, no. 12, pp. 20 640–20 651, 2025.

[17] M. B. Yasin, Y. M. Khamayseh, and M. AbuJazoh, "Feature selection for black hole attacks." *J. Univers. Comput. Sci.*, vol. 22, no. 4, pp. 521–536, 2016.

[18] S. Pandey and V. Singh, "Blackhole attack detection using machine learning approach on manet," in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*. IEEE, 2020, pp. 797–802.

[19] T. Nagalakshmi, A. Gnanasekar, G. Ramkumar, and A. Sabarivani, "Machine learning models to detect the blackhole attack in wireless adhoc network," *Materials Today: Proceedings*, vol. 47, pp. 235–239, 2021.

[20] I. Almomani, B. Al-Kasasbeh, and M. Al-Akhras, "Wsn-ds: A dataset for intrusion detection systems in wireless sensor networks," *Journal of Sensors*, vol. 2016, 2016.

[21] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*. PMLR, 2017, pp. 214–223.

[22] A. Rogozhnikov, "Einops: Clear and reliable tensor manipulations with einstein-like notation," in *International Conference on Learning Representations*, 2022.

[23] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *Information Processing in Medical Imaging*. Cham: Springer International Publishing, 2017, pp. 146–157.

[24] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S.-K. Ng, "Mad-gan: Multivariate anomaly detection for time series data with generative adversarial networks," in *International conference on artificial neural networks*. Springer, 2019, pp. 703–716.

[25] T. Schlegl, P. Seeböck, S. M. Waldstein, G. Langs, and U. Schmidt-Erfurth, "f-anogan: Fast unsupervised anomaly detection with generative adversarial networks," *Medical image analysis*, vol. 54, pp. 30–44, 2019.

[26] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in neural information processing systems*, vol. 34, pp. 22 419–22 430, 2021.

[27] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 12, 2021, pp. 11 106–11 115.