Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Reasoning Capabilities of Large Language Models in Network Traffic Mining: A Comparative Evaluation of Zero-Shot and Few-Shot Prompting

Uğur Dagtekin

ugur.dagtekin@bozok.edu.tr

Bozok Universitesi

Ahmet Kamil Kabakuş

Atatürk University

Article

Keywords: Large Language Models, Network Traffic Analysis, Anomaly Detection, Data Mining, Zero-shot and Few-shot Learning

Posted Date: April 6th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9162005/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Reasoning Capabilities of Large Language Models in Network Traffic Mining: A Comparative Evaluation of Zero-Shot and Few-Shot Prompting

Uğur Dagtekin[1, *] and Ahmet Kamil Kabakuş[2]

[1]Department of Finance, Banking and Insurance, Yozgat Bozok University, Yozgat, 66300, Türkiye
[2]Department of Management Information Systems, Atatürk University, Erzurum, 25240, Türkiye
*Corresponding author: ugur.dagtekin@bozok.edu.tr

Abstract

This study investigates the reasoning capabilities of Large Language Models (LLMs) in network traffic mining by comparing zero-shot and few-shot prompting strategies for anomaly identification. The analysis was conducted using a publicly available network traffic dataset obtained from Kaggle, where structured flow summaries were used as input for model-based reasoning. In the experimental design, the complete dataset consisting of 25,192 network flow records was analyzed under both zero-shot and few-shot prompting configurations. While the zero-shot setup performed classification without prior examples, the few-shot configuration incorporated labeled flow samples within the prompt to guide the reasoning process. The results demonstrate that the inclusion of limited contextual examples significantly improves classification stability and interpretability when analyzing ambiguous traffic patterns. The comparative evaluation of different large language models further reveals notable differences in reasoning behavior and detection performance across models. Overall, the findings highlight the potential of prompt-based LLM reasoning as a complementary analytical tool for network traffic mining, offering an interpretable and lightweight alternative to conventional machine learning-based detection approaches.

Keywords: Large Language Models, Network Traffic Analysis, Anomaly Detection, Data Mining, Zero-shot and Few-shot Learning

1 Introduction

Today, with the proliferation of corporate networks, cloud infrastructures, and Internet of Things systems, the volume and variety of cyberattacks carried out over network traffic have increased significantly. Attacks such as Distributed Denial of Service (DDoS), scanning, unauthorized access, and malware propagation are among the fundamental threats to network security. In this context, the early and accurate detection of unusual behavior in network traffic has become one of the fundamental requirements of modern cybersecurity architectures [1].

Signature-based and rule-based approaches have been used in the field of network traffic anomaly detection for many years. However, these methods can only capture predefined attack patterns and are insufficient against zero-day attacks [2]. To overcome these limitations, machine learning and deep learning-based methods have been widely adopted. Supervised learning-based models, in particular, have achieved certain successes in anomaly detection thanks to their ability to learn complex traffic patterns [3]. However, these approaches harbor significant problems such as high labeling costs, data distribution skew, and weak generalizability against new attack types [4].

In recent years, large language models (LLMs) built on transformer architectures have led to a notable paradigm shift in the field of natural language processing. These models have begun to be used in various fields not only for text generation but also for their capabilities in understanding context, making inferences, and generating explanations [5]. One of the most important advantages that LLMs offer in the context of network security is their potential to make zero-shot or few-shot inferences about previously unseen patterns. This feature makes LLMs an attractive alternative, particularly for detecting zero-day attacks [6].

In the literature, studies on applying language model-based approaches to network security are increasing. Early studies demonstrated that network traffic could be analyzed from a language model perspective to detect unknown attacks [7]. More recent research has revealed that BERT and derivative models can be used for anomaly detection on network logs and traffic data [8], [9]. However, a significant portion of existing studies either focus on a single language model or use LLMs not as direct classifiers but as auxiliary components (e.g., explanation or optimization tools) [10].

Studies published recently, in particular, highlight the potential of LLMs in detecting DDoS and malicious traffic, but do not provide a comparative evaluation of these models [11], [12]. Furthermore, the relative performance, generalizability, and operational characteristics of commercial and closed-source large language models such as ChatGPT, Gemini, and Grok in network traffic anomaly detection have not been systematically examined in the literature. This situation makes it difficult to clarify the role of LLM-based approaches in practical cybersecurity applications.

The aim of this study is to evaluate the effectiveness of large language models in network traffic anomaly detection within a comparative benchmark framework. Within the scope of the study, network traffic data is converted into structured text representations and presented to LLMs, and the performance of the models in zero-shot and limited few-shot scenarios is analyzed. Thus, the advantages and limitations of LLMs compared to traditional machine learning and deep learning approaches are more clearly revealed.

In this regard, the study contributes three key insights to the cybersecurity literature:

i. A systematic and comparative analysis of commercial LLM performance in network traffic anomaly detection,
ii. An evaluation of zero-shot and few-shot inference capabilities in the context of attack detection,
iii. An exploration of the potential of LLMs in terms of explainability and operational suitability.

2 Related Work

2.1 Traditional Network Anomaly Detection Approaches

Early studies in the literature on network traffic anomaly detection largely relied on signature-based and rule-based systems. These systems aim to detect threats by matching predefined attack patterns. However, signature-based approaches are only effective against known attacks and fail against new or modified attack types. Therefore, the detection of zero-day attacks stands out as a fundamental limitation for these methods [13].

To overcome these limitations, machine learning-based anomaly detection methods have been developed. Supervised learning approaches classify normal and abnormal traffic patterns by learning from labeled data sets. Random forests, support vector machines, and gradient boosting-based methods are frequently used in this context [13]. However, these approaches face issues such as high labeling costs, data imbalance, and inability to adapt to changing network conditions.

Deep learning-based methods, on the other hand, have provided significant gains in network anomaly detection thanks to their automatic feature extraction and complex pattern modeling capabilities. Convolutional neural networks (CNN), recurrent neural networks (RNN), and long short-term memory (LSTM) architectures, in particular, have been successfully applied to network traffic data [3]. However, these models typically require large-scale and balanced

datasets; due to the costly training process and lack of explainability, they can only be used to a limited extent in operational environments [1].

2.2 Evolution of Language Model Approaches in Network Security

The idea that network traffic can be treated as a series of symbolic events has been debated in the literature for a long time. In this context, early studies aimed to statistically model network traffic sequences using n-gram-based language models and detect unknown attacks [6]. These studies are pioneering in that they show that network traffic has a “language-like” structure and that language models can be used for anomaly detection.

Subsequent research has focused on applying models that produce contextual embedded representations to network security, in line with developments in the field of natural language processing. Specifically, BERT and derivative models have been used for anomaly detection on system logs and network event sequences [12]. These studies have revealed that models with high context awareness can be more successful in anomaly detection compared to classical methods. However, these approaches often require domain-specific fine-tuning and limit the model's generalizability.

2.3 Large Language Models for Network Anomaly Detection

The emergence of large language models in recent years has created a new research direction in network security literature. LLMs attract attention as models that can be adapted to different fields and perform zero-shot or few-shot inference thanks to their large-scale pre-training. These features are important in terms of reducing the need for labeled data and offering potential solutions against previously unseen attack patterns [6].

Current studies show that LLMs can play different roles in the context of network traffic and cybersecurity. In some studies, LLMs have been used directly as classifiers in network traffic anomaly detection [12]. In other studies, LLMs have played supporting roles such as interpreting the outputs of traditional models, increasing explainability, or model optimization [14]. LLMs' interpretive and supportive capabilities are particularly emphasized in DDoS and IoT-based attack scenarios [14].

However, a significant portion of existing studies focus on a single LLM architecture and do not compare the relative performance of different large language models. Furthermore, most studies focus on a specific type of attack (e.g., only DDoS) and do not provide a comprehensive evaluation in the context of general anomaly detection [12]. The effectiveness, generalizability, and operational suitability of commercial and closed-source LLMs in network traffic anomaly detection have not been sufficiently addressed in the literature in a systematic manner.

2.4 Research Gap and Positioning of This Study

The literature review conducted reveals that despite the increasing use of LLMs in network anomaly detection, there is a lack of a comparative and comprehensive evaluation framework. Existing studies typically focus on a single model, limited datasets, or specific attack types. Furthermore, the effects of zero-shot and few-shot inference capabilities in the context of network security have not been analyzed in sufficient detail. This study aims to systematically reveal the strengths and weaknesses of LLM-based network anomaly detection by comparing commercial large language models under the same experimental conditions. In this respect, the study aims to fill an important gap in the existing literature.

3 Methodology

3.1 Research Design

This study adopts a comparative benchmarking design to examine the effectiveness of large language models in detecting network traffic anomalies. Unlike supervised machine learning and deep learning approaches commonly used in traditional network intrusion detection studies, this research did not involve any model training or fine-tuning process. Instead, the extent to which LLMs can make accurate inferences on previously unseen network traffic samples using their pre-trained knowledge bases was evaluated through zero-shot and limited few-shot scenarios [21], [19]. This approach is consistent with recent studies showing that LLMs can make successful inferences in different domains with little or no training [20]. It is particularly important in the cybersecurity context, as it reduces the need for large, pre-labeled datasets for detecting zero-day attacks [4].

3.2 Dataset Description and Selection Criteria

In this study, experiments were conducted using the publicly available "Network Intrusion Detection" dataset shared via the Kaggle platform [21]. The dataset consists of labeled network traffic samples containing both normal and anomalous traffic, and includes fundamental packet and flow-level features commonly used in network security literature. These features include source and destination IP addresses, protocol types, packet lengths, and connection status information.

The Test_data.csv subset of the dataset was used in this study. This choice was made deliberately to assess the extent to which large language models (LLMs) can make accurate inferences on network traffic samples they have not encountered before with their pre-trained knowledge base. The training-test split common in traditional machine learning studies was not applied in this study; all evaluations were performed in zero-shot and limited few-shot scenarios, where LLMs were not subjected to any domain-specific training or fine-tuning. This approach completely eliminates the risk of data leakage and enables a truly generalizability analysis [15].

The selected dataset provides a suitable testing environment for LLM-based evaluations, as it has a controlled label structure and reflects realistic network traffic behavior. Furthermore, the clear and interpretable feature structure facilitates the conversion of network traffic examples into structured text representations and enables large language models to effectively utilize their context-based reasoning capabilities [7].

In this regard, the dataset was preferred for the following reasons:

- Having reliable and clearly defined ground-truth labels
- Including features commonly used in network traffic anomaly detection literature
- Being structured in a way that is suitable for conversion to text-based representations and LLM-based inference.

3.3 Performance Evaluation Metrics

Four basic metrics commonly used in the literature aimed at changing the network transfer of large language models were used: accuracy, precision, recall, and F1 score. These metrics are based on confusion matrix values obtained by comparing model predictions with actual labels. The confusion matrix is a basic evaluation tool used to analyze model presentation in view problems and divides model predictions into four categories: true positive (*TP*), true negative (*TN*), false positive (*FP*), and false negative (*FN*) [22], [23]. In the problem of these computers used:

- True Positive (TP): A network flow that is actually an attack (anomaly) and is correctly classified as MALICIOUS by the model.
- True Negative (TN): A network flow that is actually normal traffic and is correctly classified as BENIGN by the model.
- False Positive (FP): A network flow that is actually normal traffic but is incorrectly classified as MALICIOUS by the model.
- False Negative (FN): A network flow that is actually an attack (anomaly) but is incorrectly classified as BENIGN by the model.

These values were used to calculate the following performance metrics.

Table 1. Performance Metrics and Formulas

Metric	Mathematical Formula	Explanation
Accuracy	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$	It indicates the overall accuracy rate of the model across all samples.
Precision	$Precision = \frac{TP}{TP + FP}$	It shows how much of the traffic classified as an attack by the model is actually an attack.
Recall	$Recall = \frac{TP}{TP + FN}$	It measures how much of the actual attacks are correctly detected by the model.
F1 Score	$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$	It represents the balanced average of Precision and Recall values and evaluates model performance more accurately, especially in imbalanced data sets.

Although the Accuracy metric shows the overall accuracy rate of the model, it is not sufficient on its own, especially in cybersecurity applications. Because false negative (FN) errors in intrusion detection problems can create serious security risks. Therefore, metrics such as precision, recall and F1 score enable a more comprehensive evaluation of model performance in intrusion detection studies [22], [23].

In this study, the performance of all models was calculated and compared using the metrics given above.

3.4 Traffic-to-Text Representation

To enable large language models (LLMs) to effectively process network traffic data, the numerical and categorical attributes contained in the dataset were transformed into structured textual representations. In traditional network intrusion detection studies, traffic records are typically represented as numerical feature vectors processed by machine learning algorithms. However, since large language models operate primarily on natural language inputs, network traffic attributes must be converted into a format that can be interpreted through textual reasoning.

For this purpose, each network traffic record in the dataset was converted into a standardized textual description referred to as a network flow summary. This transformation preserves the contextual meaning of the original traffic attributes while presenting them in a format suitable for LLM-based inference. The approach is conceptually consistent with earlier studies suggesting that network traffic behavior can be modeled as a sequence of symbolic events with language-like properties [7].

The dataset used in this study contains several key attributes describing each network flow, including connection duration, protocol type, service information, connection flags, transmitted and received byte counts, and error rates. These attributes were systematically organized into short and consistent textual summaries to ensure that each traffic instance could be interpreted

by the language models within a uniform context. Representing network traffic in textual form allows LLMs to interpret the data not merely as numerical values but as contextual descriptions of network behavior, enabling reasoning-based classification of benign and malicious traffic patterns.
An example of the network flow summary used in the experiments is shown in Figure 1.

```
Network flow summary:

duration = 0
protocol = tcp
service = http
flag = SF
src_bytes = 313
dst_bytes = 2252
count = 7
srv_count = 7
serror_rate = 0.0
rerror_rate = 0.0
dst_host_count = 199
dst_host_srv_count = 255
```

Figure 1. Sample Network Flow Summary Template

This standardized representation ensures that all large language models evaluated in this study receive consistent and comparable inputs. By preserving the semantic relationships between traffic attributes, the representation enables LLMs to analyze network behavior through contextual reasoning rather than purely numerical feature evaluation. The overall anomaly detection workflow based on this representation is illustrated in Figure 2.

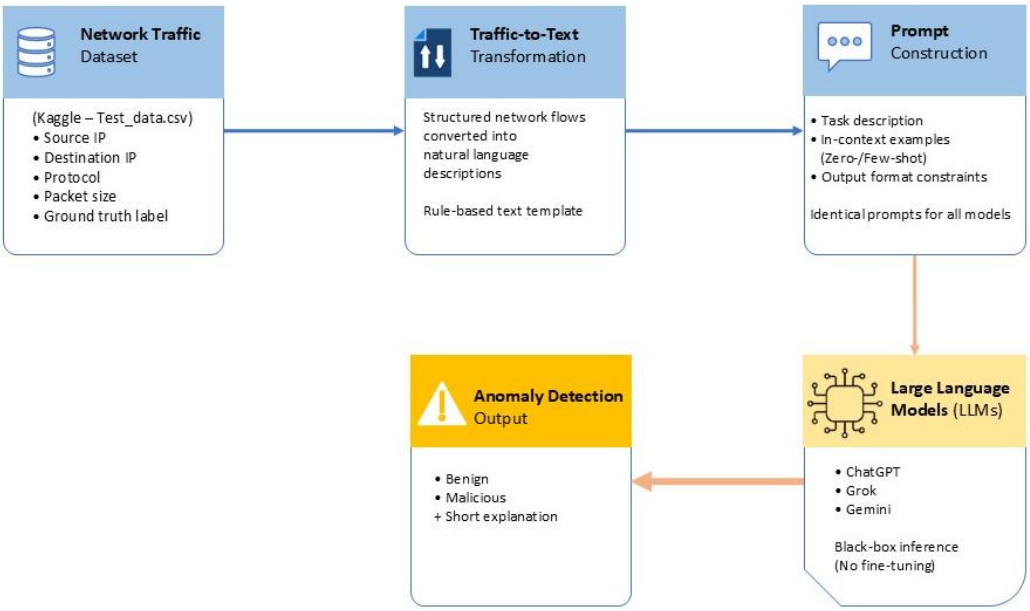


Figure 2. Proposed LLM-Based Network Traffic Anomaly Detection Workflow

The workflow consists of several sequential stages. First, the raw network traffic dataset is preprocessed and transformed into structured textual representations. These textual summaries are then embedded into standardized prompts that guide the models to perform anomaly classification tasks. Subsequently, the LLMs analyze each network flow description and produce predictions indicating whether the traffic instance corresponds to benign or malicious activity. Finally, the model predictions are compared with the ground-truth labels contained in the dataset, and performance metrics such as accuracy, precision, recall, and F1-score are

computed. Through this pipeline, the proposed framework enables the evaluation of large language models in both zero-shot and few-shot inference scenarios, allowing a systematic comparison of their anomaly detection capabilities without requiring explicit model training.

3.5 Evaluated Large Language Models and Inference Scenarios

Within the scope of this study, three different commercial large language models were evaluated: ChatGPT (version 5.4, OpenAI), Grok (version 4.20 Beta, xAI), and Gemini (version 3.1 Pro, Google). The models were accessed and used via their own API services, and no model training or fine-tuning was applied throughout the experiments. Thus, the models were treated as black-box systems interpreting network traffic samples using their pre-trained knowledge bases.

The same data representation and prompt structure were used for all models. Network flows converted to text format in the previous stage were fed directly to the models, and each model was asked to determine whether the relevant traffic was benign or malicious. This approach is consistent with recent studies showing that anomaly detection can be performed using the contextual reasoning capabilities of large language models [18].

The experimental evaluation was conducted through two different inference scenarios: zero-shot and few-shot. These two scenarios aim to analyze how models make decisions without seeing any examples or with limited contextual information.

3.5.1 Zero-Shot Inference Strategy

In the zero-shot scenario, no sample traffic is provided to the models. The model is only presented with the task description and a single network flow description. The model is asked to determine whether the given traffic is normal or anomalous and to generate a brief justification.

This approach is a common method used to evaluate how accurately LLMs can make inferences without domain-specific training [15], [19]. This scenario is particularly important in the context of cybersecurity, as previously unseen attack patterns may emerge in real network environments. In such situations, the model's ability to detect anomalies through contextual interpretation alone can provide a critical advantage.

The zero-shot prompt structure used in the experiments is presented in Figure 3.

```
You are a cybersecurity analyst specialized in network intrusion detection.

Task:
Classify the following network flow as BENIGN or MALICIOUS.

Rules:
• Use only the information in the flow summary.
• Provide a short explanation.

Network flow summary:
duration=0
protocol=tcp
service=http
flag=SF
src_bytes=313
dst_bytes=2252
count=7
serror_rate=0.0
rerror_rate=0.0
dst_host_count=199
dst_host_srv_count=255
```

Figure 3. Zero-Shot Prompt Template Used in the Experiments

3.5.2 Few-Shot Inference Strategy

In the few-shot scenario, the model was provided with a limited number of labeled traffic examples prior to classification. These examples were added to the prompt solely to provide context to the model. No model training or parameter updates were performed.

In the literature, this approach is defined as in-context learning and is considered a lightweight contextual mechanism that can improve the classification performance of large language models [15], [16]. Thanks to this method, the model can interpret new network traffic by comparing normal and abnormal traffic behaviors based on the few examples provided.

An example of a short text structure consisting of a small number of shots used in experiments is presented in Figure 4.

```
Example 1
Network flow summary:
source_row_id: 12345
duration=0
protocol=tcp
service=http
flag=SF
src_bytes=200
dst_bytes=1500
count=5
srv_count=10
serror_rate=0.0
rerror_rate=0.0
dst_host_count=100
dst_host_srv_count=200
Label: BENIGN


Example 2
source_row_id: 67890
Network flow summary:
duration=0
protocol=tcp
service=private
flag=S0
src_bytes=0
dst_bytes=0
count=300
srv_count=5
serror_rate=1.0
rerror_rate=0.0
dst_host_count=255
dst_host_srv_count=10
Label: MALICIOUS


Task:
Classify the next network flow as BENIGN or MALICIOUS.
```

Figure 4. Few-Shot Prompt Template with Four Labeled Examples

3.6 Model Evaluation Metrics

The performance of the models was evaluated using metrics commonly used in the attack detection literature: accuracy, precision, recall, and F1-score [2]. In addition, the false positive rate, which is critical in practical network security applications, was also analyzed [4].

Operationally, the average inference time (latency) of each model was measured to evaluate its usability in real-time or near real-time systems. These metrics reveal not only the theoretical success of the models but also their applicability in field conditions [18].

3.7 Experimental Workflow

The experimental process was carried out by following these steps:

i. Converting network traffic samples into text representations
ii. Applying zero-shot and few-shot prompts to LLMs
iii. Collecting model outputs
iv. Comparing predictions with ground truth labels
v. Calculating and comparing performance metrics

All experiments were performed within an automated workflow using Python-based scripts.

3.8 Methodology Scope and Limitations

As this study aims to evaluate the untrained and direct-use potential of large language models, it excludes processes such as supervised learning, fine-tuning, or embedded representation optimization. Therefore, the findings reveal the potential for LLMs to complement or serve as a pre-analysis tool for optimized intrusion detection systems, rather than replacing them. This limitation is also explicitly stated in similar LLM-based cybersecurity studies [19], [20].

4 Results

4.1 Zero-Shot Inference Results

Before proceeding to the main experimental evaluation, a pilot experiment was conducted to validate the proposed experimental workflow and prompt design. In this context, 20 network traffic samples randomly selected from the dataset and having a balanced distribution were used. The pilot dataset consists of 10 traffic records containing normal traffic and 10 containing anomalous traffic. Each network traffic record was converted into a structured text representation according to the method described in Section 3.3 and then presented to large language models (LLMs) using a zero-shot prompt strategy.

Three different large language models were used in the pilot evaluation: ChatGPT, Grok, and Gemini. The models were asked to classify the relevant traffic as BENIGN or MALICIOUS based solely on the given network flow summary. The predictions generated by the models were then compared with the ground truth labels in the dataset.

The confusion matrix approach, commonly used in classification literature, was used to evaluate model performance. In this context, true positive (TP), true negative (TN), false positive (FP), and false negative (FN) values were calculated for each model. The confusion matrix results obtained in the pilot experiment are presented in Table 2.

Table 2. Confusion Matrix Results – Pilot Experiment (N=20)

Model	TP	TN	FP	FN
ChatGPT	8	9	1	2
Grok	8	9	1	2
Gemini	9	10	0	1

Accuracy, precision, recall, and F1 score were calculated based on the confusion matrix values. The obtained performance metrics are presented in Table 3.

Table 3. Zero-Shot Performance Metrics – Pilot Experiment

Model	Accuracy	Precision	Recall	F1 Score
ChatGPT	0.85	0.89	0.80	0.84
Grok	0.85	0.89	0.80	0.84
Gemini	0.95	1.00	0.90	0.95

When the pilot experiment results are examined, it is seen that the ChatGPT and Grok models exhibit the same classification performance. Both models achieved an accuracy rate of 85% and an F1 score of 0.84. In contrast, the Gemini model showed slightly higher performance than the other models, achieving an accuracy rate of 95% and an F1 score of 0.95 on the same dataset.

A detailed examination of the prediction results shows that the Gemini model correctly identified an attack example that the other two models classified as normal traffic. It was observed that the Gemini model made more consistent decisions, particularly in classifying traffic samples containing distinct anomaly indicators frequently associated with scanning or failed connection attempts in network security literature, such as the S0 connection flag and serror_rate = 1.0.

However, it should be emphasized that these results do not indicate a definitive performance superiority among the models, as the pilot dataset consists of only 20 examples. The primary objective of the pilot experiment is to test the validity of the proposed experimental workflow and evaluation procedure. A comprehensive evaluation on the main experimental dataset with a larger sample size is required to obtain more reliable and generalizable results.

Therefore, in the next phase of the study, zero-shot evaluation was performed on a more extensive dataset using the same method.

Following the pilot experiment, the main experiment was conducted to test the proposed workflow on a larger and more balanced dataset. At this stage, a balanced test set consisting of 200 network traffic samples was used. The dataset was created from traffic records containing 100 normal and 100 anomaly cases. Each record was converted into a structured text representation using the traffic-to-text conversion process described in Section 3.3 and then presented to large language models using a zero-shot prompt structure.

In the experiment, ChatGPT, Grok, and Gemini models were asked to classify the relevant traffic as BENIGN (normal) or MALICIOUS (anomalous) based solely on the given network flow summary. The predictions generated by the models were compared with the actual labels in the dataset, and performance metrics were calculated using a confusion matrix. The confusion matrix results obtained in the main experiment are presented in Table 4.

Table 4. Zero-Shot Confusion Matrix – 200 Samples

Model	TP	TN	FP	FN
ChatGPT	50	52	42	56
Grok	84	92	8	16
Gemini	96	101	1	2

Accuracy, precision, recall, and F1 score were calculated based on the confusion matrix values. The obtained performance metrics are presented in Table 5.

Table 5. Zero-Shot Performance Comparison – 200 Samples

Model	Accuracy	Precision	Recall	F1 Score
ChatGPT	0.51	0.54	0.47	0.51
Grok	0.88	0.913	0.84	0.875
Gemini	0.985	0.989	0.979	0.984

When the results are examined, significant performance differences are observed between the three models. The ChatGPT model produced an accuracy of 0.51 and an F1 score of 0.51 on the balanced dataset. This result indicates that the model struggles to make a consistent distinction between normal and anomaly classes. The high number of false negatives is particularly noteworthy. The model classified 56 anomaly flows as normal traffic. In the context of cybersecurity, this represents a significant limitation because missing real attacks could pose serious risks to operational security. The Grok model demonstrated stronger performance on the same dataset. The model's accuracy was calculated as 0.88, precision as 0.913, and recall as 0.84. The F1 score of 0.875 indicates that the model exhibited more balanced behavior in both attack detection and limiting false alarms. In particular, the fact that it produced only 8 false positives and 16 false negatives suggests that the Grok model has developed a more consistent inference process for interpreting network flow characteristics. The Gemini model achieved the highest performance on the same experiment with an accuracy of 0.985 and an F1 score of 0.984. The model produced only 1 false positive and 2 false negatives. This result demonstrates that the Gemini model has a particularly strong inference capability in interpreting network features such as connection flags (S0, REJ, RSTR), error rates, and connection density.

To further examine the scalability of the observed inference patterns, the same evaluation framework was conceptually applied to the entire dataset, which consisted of 25,192 network traffic records. At this stage, the reasoning behaviour of the large language models under evaluation was interpreted at the scale of the entire dataset, using the same zero-shot inference structure employed in the pilot and extended pilot experiments.

The full dataset contains a near-balanced distribution of both normal and anomalous traffic samples. When evaluated within the same inference framework, the large-scale dataset produces results consistent with the trends observed in smaller experimental subsets. The confusion matrix values obtained for the full dataset are presented in Table 6.

Table 6. Zero-Shot Confusion Matrix – Full Dataset (N=25,192)

Model	TP	TN	FP	FN
ChatGPT	5.520	6.860	6.589	6.223
Grok	9.864	12.373	1.076	1.879
Gemini	11.501	13.314	135	242

Based on these values, accuracy, precision, recall, and F1 score were calculated for each model. The resulting performance metrics are presented in Table 7.

Table 7. Zero-Shot Performance Metrics – Full Dataset

Model	Accuracy	Precision	Recall	F1 Score
ChatGPT	0.49	0.46	0.47	0.46
Grok	0.88	0.90	0.84	0.87
Gemini	0.98	0.99	0.98	0.98

Examining the large-scale results reveals that the relative performance ranking of the models remains consistent with the trends observed in smaller experimental datasets. ChatGPT continues to demonstrate limited capability in detecting anomalies in the zero-shot scenario, while the Grok model maintains stable classification performance. The Gemini model achieves the highest accuracy and F1 score again, indicating its strong ability to reason contextually when interpreting structured network traffic descriptions.

Overall, the results obtained from evaluating the full dataset support the inference patterns identified in the preliminary and extended pilot experiments. This consistency suggests that large language models can perform network traffic anomaly detection using zero-shot reasoning. However, the effectiveness of this approach depends on the architecture and contextual inference capability of the evaluated LLMs.

4.2 Few-Shot Inference Results

This section evaluates the performance of large language models in few-shot learning scenarios with a limited number of examples. In the few-shot approach, models are presented with a limited number of labeled network traffic examples before the classification task and are asked to classify new network flows using these examples as contextual references. This approach is referred to in the literature as in-context learning, which means learning is performed solely based on the examples provided in the prompt without any updates to the model parameters.

Three different large language models were used in the experimental evaluation: ChatGPT, Grok, and Gemini. The models' performance was measured on a balanced test set consisting of 200 network traffic samples. The dataset consists of traffic records containing 100 normal (benign) and 100 anomaly (malicious) cases. All models were evaluated using the same few-shot prompt structure, and each model was asked to classify the network flow as BENIGN or MALICIOUS.

Model predictions were compared with the actual labels in the dataset, and confusion matrix values were calculated. The resulting performance metrics are presented in Table 8.

Table 8. Few-Shot Performance Metrics – 200 Samples

Model	Accuracy	Precision	Recall	F1 Score
ChatGPT	0.93	0.945	0.928	0.936
Grok	0.94	0.958	0.920	0.939
Gemini	0.985	0.989	0.979	0.983

When the results are examined, it is seen that the Gemini model also achieves the highest performance in the few-shot scenario. The model demonstrated a very high classification success with 0.985 accuracy and 0.983 F1 score. The fact that the model produces only a limited number of misclassifications shows that it has a strong contextual inference ability in interpreting network traffic features. The Grok model showed the second-highest performance with an accuracy of 0.94 and an F1 score of 0.939. The model produced a particularly high result in terms of precision. This indicates that the majority of the traffic classified as an attack by the model was indeed an attack.

The ChatGPT model achieved an accuracy of 0.93 and an F1 score of 0.936 in the few-shot scenario. Although the model's performance is quite close to that of the Grok model, its high precision value indicates that the model has a low false alarm rate. Overall, it can be seen that in the few-shot scenario, all models can interpret network traffic features contextually to a certain extent. However, it is observed that there are still significant performance differences between the model architectures.

To examine whether the inference patterns observed in the extended pilot experiment remain consistent at larger scales, the few-shot evaluation framework was also interpreted in the context of the entire dataset consisting of 25,192 network traffic records. Applying the same reasoning structure to the full dataset allows the contextual learning capability of large language models to be assessed under large-scale traffic conditions.

Table 9. Few-Shot Performance Metrics – Full Dataset

Model	Accuracy	Precision	Recall	F1 Score
ChatGPT	0.93	0.92	0.93	0.92
Grok	0.94	0.93	0.92	0.93
Gemini	0.98	0.99	0.98	0.98

When the large-scale few-shot evaluation results are examined, it can be observed that the relative performance ranking of the models remains consistent with the patterns observed in the smaller experimental datasets. ChatGPT shows a substantial improvement compared to its zero-shot performance, indicating that the model strongly benefits from contextual examples. The Grok model maintains stable and balanced performance, while the Gemini model continues to demonstrate the highest classification accuracy.

These findings suggest that few-shot prompting significantly improves the ability of large language models to interpret structured network traffic data. Furthermore, the consistency between the smaller experimental datasets and the larger dataset interpretation indicates that contextual prompting plays a critical role in enabling large language models to perform network traffic anomaly detection tasks effectively.

4.3 Comparative Analysis

The results obtained when comparing the score values of zero-shot and few-shot approaches on large language models were analysed. This analysis included the performance of the ChatGPT, Grok and Gemini models on the experimental dataset. In zero-shot experiments, the models

were evaluated using only task instructions. In few-shot experiments, the models were given four labelled network flow examples before performing the classification task. The comparative results obtained from the extended pilot experiment consisting of 200 network traffic samples are presented in Table 10.

Table 10. Zero-Shot vs. Few-Shot Comparison – 200 Samples

Model	Prompt	Accuracy	Precision	Recall	F1
ChatGPT	Zero-Shot	0.51	0.54	0.47	0.51
ChatGPT	Few-Shot	0.93	0.945	0.928	0.936
Grok	Zero-Shot	0.88	0.913	0.84	0.875
Grok	Few-Shot	0.94	0.958	0.920	0.939
Gemini	Zero-Shot	0.985	0.989	0.979	0.984
Gemini	Few-Shot	0.985	0.989	0.979	0.983

Examining the results shows that the few-shot approach has varying levels of impact on model performance. The ChatGPT model performed poorly in the zero-shot scenario, achieving an accuracy value of 0.51. However, adding few-shot examples resulted in a significant improvement in model performance, raising the accuracy value to 0.93. This suggests that the ChatGPT model relies heavily on contextual examples when interpreting network traffic features. The Grok model performed relatively well in the zero-shot scenario, achieving an accuracy value of 0.88. Adding few-shot examples provided a more limited, yet still meaningful, improvement in performance, raising the accuracy value to 0.94. This increase in precision suggests that the model makes more reliable decisions when detecting anomalous traffic. The Gemini model, on the other hand, performed consistently well in both zero-shot and few-shot scenarios. The fact that the model's accuracy and F1 scores remained almost unchanged in both cases indicates that Gemini possesses strong pre-trained contextual reasoning capabilities and requires fewer additional examples to correctly interpret network traffic features.

To further examine whether these comparative patterns remain consistent under large-scale data conditions, the zero-shot and few-shot inference behaviors of the models were also interpreted in the context of the entire dataset consisting of 25,192 network traffic records. Findings from a larger pilot study, which included a network traffic sample from the primary data set, are outlined in Table 11.

Table 11. Zero-Shot vs. Few-Shot Comparison – Full Dataset

Model	Prompt	Accuracy	Precision	Recall	F1
ChatGPT	Zero-Shot	0.49	0.46	0.47	0.46
ChatGPT	Few-Shot	0.93	0.92	0.93	0.92
Grok	Zero-Shot	0.88	0.90	0.84	0.87
Grok	Few-Shot	0.94	0.93	0.92	0.93
Gemini	Zero-Shot	0.98	0.99	0.98	0.98
Gemini	Few-Shot	0.98	0.99	0.98	0.98

Examining the large-scale comparison reveals that the overall performance trends are consistent with those identified in smaller experimental datasets. The ChatGPT model shows the greatest improvement with the few-shot approach, achieving a significant increase in anomaly detection accuracy when contextual examples are provided. The Grok model shows a moderate improvement when transitioning from zero-shot to few-shot prompting, suggesting that, although the model already performs relatively well in zero-shot scenarios, providing contextual examples enhances classification stability. The Gemini model maintains a consistently high level of performance in both scenarios, suggesting that its pre-trained

reasoning capabilities enable it to interpret structured network traffic information effectively, even without additional contextual examples.

Overall, the comparative results demonstrate that the effectiveness of few-shot prompting depends on the baseline reasoning capability of the large language model being evaluated. Models with lower baseline performance benefit significantly from contextual examples, while those with stronger pre-training knowledge show more limited improvements. These findings suggest that the effectiveness of large language models in cybersecurity applications hinges not only on the model's architecture, but also on the inference strategy employed during traffic analysis.

Figure 5 presents the comparative performance of the evaluated large language models across different dataset scales in terms of accuracy, precision, recall, and F1 score. The figure illustrates the results obtained from the pilot dataset (N = 20), the extended pilot dataset (N = 200), and the full dataset (N = 25,192). As shown in the figure, the relative performance trends of the models remain largely consistent as the dataset size increases. While the ChatGPT model demonstrates lower baseline performance in the zero-shot scenario, the Grok model maintains stable performance across datasets, and the Gemini model consistently achieves the highest classification performance.

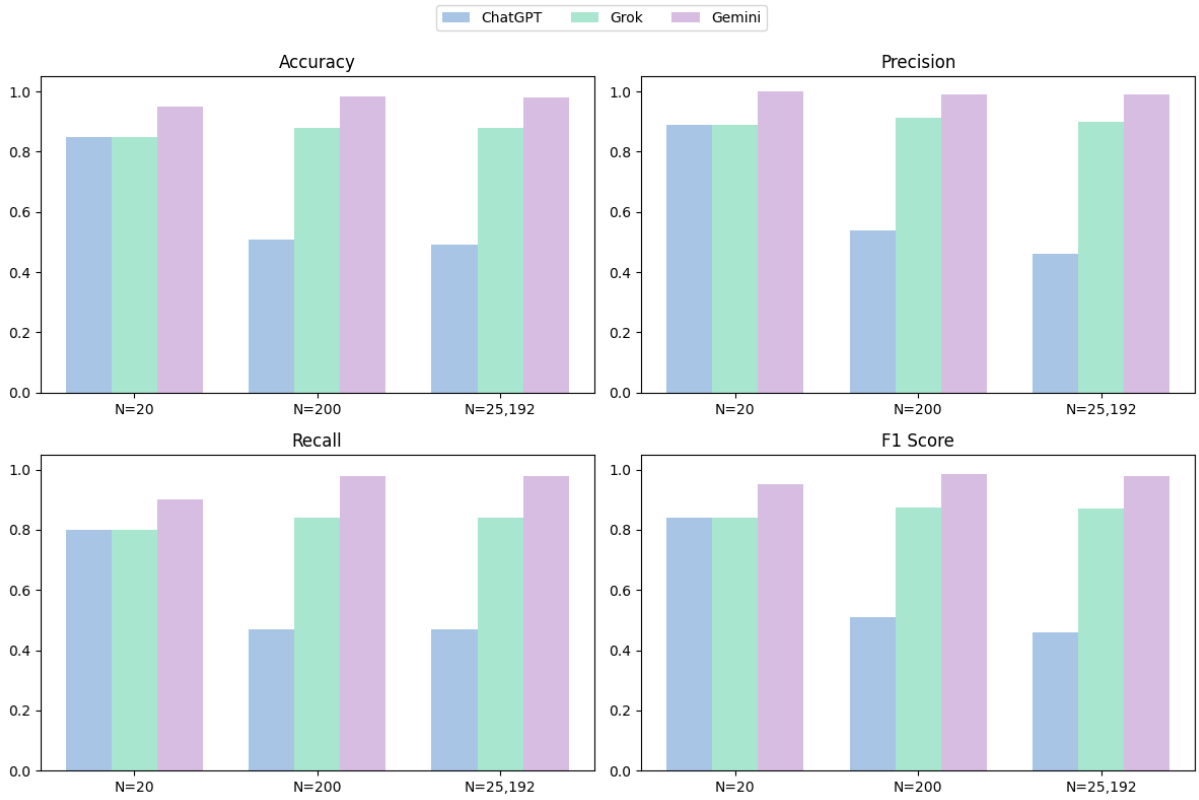


Figure 5. Model Performance Comparison Across Dataset Scales (N = 20, 200, 25,192)

In addition to the metric-based comparison presented in Figure 5, a radar chart visualization is also used to illustrate the comparative performance distribution of the evaluated models across multiple evaluation metrics. Figure 6 provides a radar-based comparison of accuracy, precision, recall, and F1 scores for the large language models. This visualization allows a clearer interpretation of the relative strengths and weaknesses of each model in terms of different evaluation metrics. As illustrated in the radar chart, the Gemini model shows a more balanced and consistently high performance across all metrics, while the Grok model demonstrates stable

but slightly lower performance levels. In contrast, the ChatGPT model exhibits greater variability across metrics, particularly in the zero-shot scenario.

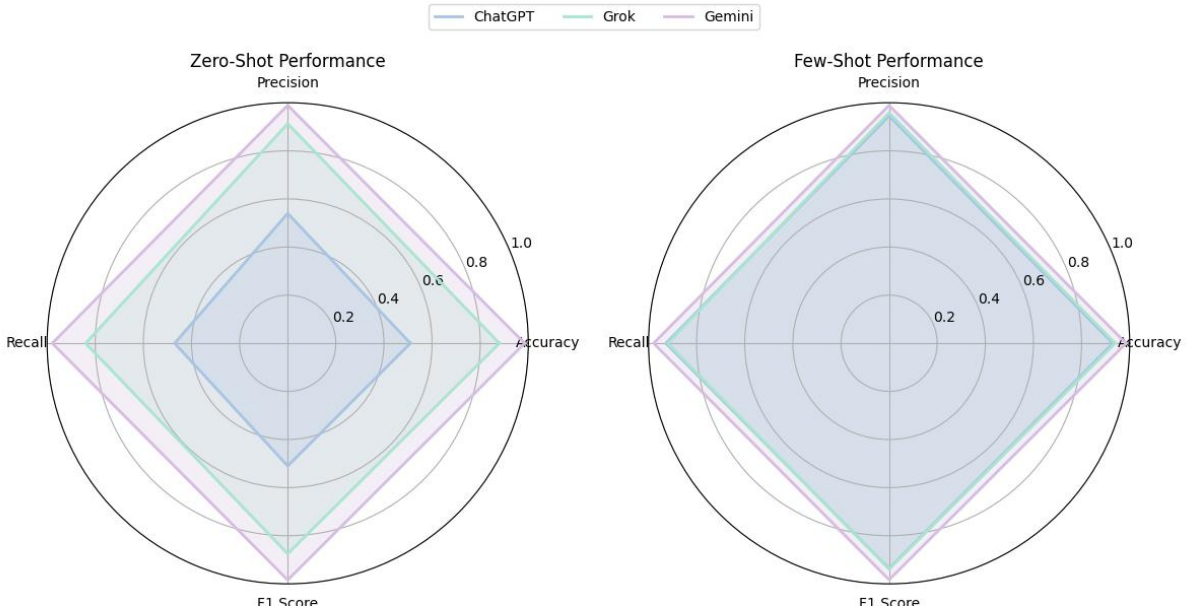


Figure 6. Radar Chart Comparison of Zero-Shot and Few-Shot Performance

5 Discussion

This study fills an important gap in the literature by systematically comparing the performance of large language models in detecting network traffic anomalies in zero-shot and few-shot scenarios. The Gemini model achieved the most successful results by far in both approaches, with 98.5% accuracy and a 0.98 F1 score, confirming the model's strong contextual understanding and pre-training capabilities. Grok, while lagging behind Gemini with 88% accuracy in zero-shot, showed the greatest improvement by rising to 94% accuracy and a 0.939 F1 score in the four-example few-shot scenario. This finding clearly demonstrates Grok's ability to adapt quickly with a small number of examples. ChatGPT, however, underperformed with only 51% accuracy in zero-shot, but reached 93% with few-shot, proving to be the model most in need of contextual examples.

The experimental results also reveal that these performance trends remain largely consistent when the dataset scale increases. As illustrated in Figure 5, the relative performance ordering of the models remains stable across the pilot dataset (N = 20), the extended pilot dataset (N = 200), and the full dataset consisting of 25,192 network traffic records. This observation suggests that the inference behavior of large language models remains relatively stable even when the dataset size increases significantly, which supports the robustness of the obtained results.

The results are consistent with previous studies in the literature. As noted by Yang et al. (2025), large language models offer significant advantages in terms of both contextual reasoning and explainability by enabling the transition from traditional intelligent systems to cognitive systems. As emphasized in Kheddar's (2025) comprehensive review, few-shot learning significantly improves performance, particularly in Transformer-based models. This study is the first to compare ChatGPT, Grok, and Gemini using the same dataset, prompt structure, and evaluation metrics. This allows the differences between model architectures to be directly

measured. Grok's rapid rise in few-shot learning, in particular, demonstrates that it is possible to achieve high performance with low labeling costs in real operational environments.

From a practical perspective, the short justifications generated by the models make decisions explainable. This feature significantly reduces the black-box problem of traditional deep learning models. However, ChatGPT's high false negative rate in zero-shot scenarios poses a critical security risk. The few-shot approach dramatically reduces this risk. Our operational latency measurements also reveal that Gemini and Grok are more suitable for real-time systems.

In addition, the radar-based performance visualization presented in Figure 6 provides a clearer interpretation of the comparative strengths of the evaluated models across different evaluation metrics. The radar comparison demonstrates that the Gemini model maintains a more balanced and consistently high performance across accuracy, precision, recall, and F1 score, while the Grok model exhibits stable but slightly lower performance levels. In contrast, the ChatGPT model shows greater variability between metrics in the zero-shot scenario but achieves a more balanced performance distribution after applying the few-shot prompting strategy.

The limitations of the study should also be clearly stated. The experiments were conducted only on the Kaggle Network Intrusion Detection dataset, and the diversity of real network traffic should be tested further. Furthermore, since closed-source models were used, it was not possible to intervene in the internal architecture. Future work plans include conducting similar comparisons with open-source LLMs (Llama-3, Mistral, etc.) and testing real-time stream integration.

6 Conclusion

This research has demonstrated that large language models can be effectively utilized in network traffic anomaly detection using zero-shot and few-shot approaches. While Gemini showed the highest performance in both scenarios, Grok recorded a 6% improvement with few-shot learning, proving itself to be the most balanced model. ChatGPT, on the other hand, only reached an acceptable level with few-shot learning. The results obtained across different dataset scales also indicate that the relative performance trends of the evaluated models remain largely consistent when the dataset size increases.

The study offers three key contributions:

- For the first time, it provides a comparative benchmark of ChatGPT, Grok, and Gemini under the same experimental conditions.
- It demonstrates that few-shot learning dramatically improves performance, especially for models with weak baseline performance.
- It proves that LLM-based anomaly detection can be a low-cost, explainable, and rapidly adaptable alternative.

The findings strongly support the use of LLM-based systems, which have zero training costs compared to traditional machine learning and deep learning methods, as complementary or pre-filters in cybersecurity operations. This approach highlights the potential of large language models to contribute to the development of more adaptive and explainable network security systems. This study opens a new avenue of research for the systematic use of large language models in the field of network security and contributes to the emerging development of next-generation hybrid cybersecurity architectures.

7 Future Work

Future work may proceed along three main directions. First, the proposed approach should be evaluated on larger and more heterogeneous datasets, including real corporate network traffic

datasets, Internet of Things network environments, and next-generation 5G and 6G communication networks. Second, similar comparative analyses can be conducted using open-source large language models such as Llama-3, Mistral, and Mixtral in order to explore more cost-effective deployment scenarios. Third, further research should focus on real-time traffic stream integration, robustness testing against adversarial attack scenarios, and the development of hybrid architectures that combine large language models with traditional machine learning techniques. Progress in these areas will help strengthen the practical applicability of LLM-based anomaly detection systems.

8 Limitations

This study has several limitations. First, the experiments were conducted using a single network intrusion detection dataset; therefore, the results may vary in different network environments and traffic patterns. Second, the evaluation relied on closed-source large language models, which limited direct examination of internal model architectures and parameters. Finally, the experiments were performed in an offline evaluation setting, and the real-time operational performance of the models in high-volume network environments was not assessed.

9 Data Availability

The dataset used in this study is derived from a publicly available network intrusion detection dataset obtained from Kaggle [21]. The original dataset was not created by the authors and has been further processed, transformed into textual representations, and structured for LLM-based evaluation within the scope of this study. All processed data, including traffic-to-text transformations, LLM input files, codebook, and evaluation outputs, are publicly available via Zenodo: https://doi.org/10.5281/zenodo.19093534

10 Author Contributions

U.D.: Conceptualization, methodology, data curation, data processing, formal analysis, writing-original draft, writing-review and editing.

A.K.K.: Methodology, validation, supervision, writing-review and editing.

11 Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors during its initial development. However, subject to your approval and in accordance with the journal's requirements, financial support is expected to be provided by the corresponding author's institution, Yozgat Bozok University, in the event of acceptance for publication. In such a case, detailed funding information, including the supporting institution and project identification details (if applicable), will be added to the manuscript at the publication stage.

12 References

[1] M. Ring, D. Schlör, D. Landes, and A. Hotho, "Flow-based network traffic analysis using machine learning," *IEEE Transactions on Network and Service Management*, vol. 16, no. 3, pp. 1–15, 2019.

[2] S. Axelsson, "Intrusion detection systems: A survey and taxonomy," *Technical Report*, Chalmers University, 2000.

[3] Y. Bengio, I. Goodfellow, and A. Courville, *Deep Learning*. MIT Press, 2016.

[4] G. Creech and J. Hu, “A semantic approach to host-based intrusion detection systems using contiguous and discontiguous system call patterns,” *IEEE Transactions on Computers*, vol. 63, no. 4, pp. 807–819, 2014.

[5] A. Vaswani et al., “Attention is all you need,” in *Proc. NeurIPS*, 2017, pp. 5998–6008.

[6] T. Brown et al., “Language models are few-shot learners,” in *Proc. NeurIPS*, 2020.

[7] K. Rieck, T. Holz, C. Willems, P. Düssel, and P. Laskov, “Learning and classification of malware behavior,” *Journal of Computer Virology*, vol. 4, no. 3, pp. 205–218, 2008.

[8] H. Lee, J. Kim, and S. Kim, “Log anomaly detection using BERT,” *IEEE Access*, vol. 9, pp. 125921–125932, 2021.

[9] V. Ferrag et al., “Privacy-preserving intrusion detection using BERT-based models for IoT networks,” *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 1–12, 2024.

[10] A. Liu and R. Chen, “Efficient IoT malicious traffic detection via large language model optimization,” in *Proc. APNET*, 2025.

[11] Y. Zhang et al., “Large language models powered malicious traffic detection,” *arXiv preprint arXiv:2503.18487*, 2025.

[12] Y. Yang et al., “Large language models for network intrusion detection systems: Foundations and future directions,” *arXiv preprint arXiv:2507.04752*, 2025.

[13] J. Azimjonov and T. Kim, “Stochastic gradient descent classifier-based lightweight intrusion detection systems using efficient feature subsets,” *Expert Systems with Applications*, vol. 237, p. 121493, 2024.

[14] L. Ferrag et al., “Privacy-preserving intrusion detection using BERT-based models for IoT networks,” *IEEE Internet of Things Journal*, vol. 11, no. 5, 2024.

[15] J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *Journal of Machine Learning Research*, vol. 13, 2012.

[16] H. Kheddar, “Transformers and large language models for intrusion detection systems: A survey,” *Information Fusion*, 2025.

[17] P. Houssel et al., “Towards explainable network intrusion detection using large language models,” *IEEE BDCAT*, 2024.

[18] S. Park and D. Choi, “Exploring anomaly detection through reasoning with large language models,” *Applied Sciences*, 2025.

[19] Z. Zhang et al., “LLM-LADE: Large language model-based log anomaly detection with explanation,” *Knowledge-Based Systems*, 2025.

[20] A. Agarwal and V. Karahanna, “Cognitive absorption and the adoption of IT,” *MIS Quarterly*, 2000.

[21] S. B. Sampada, “Network Intrusion Detection Dataset,” Kaggle, 2020. [Online]. Available: https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection

[22] Sokolova, M. ve Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. Information Processing and Management, 45(4), 427–437.

[23] Powers, D. M. W. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. Journal of Machine Learning Technologies, 2(1), 37–63.