LLM-based Anomaly Detection for Digital Substation Cybersecurity

Liam McGevna, Jason Chow, Jeffery Luo, Priscilla Danso
Department of Computer Science
Stony Brook University, New York, USA
{liam.mcgevna, jason.chow, jeffery.luo, priscilla.danso}@stonybrook.edu

***Abstract*—The security of digital substations is pivotal in the operations of smart grid. These substations rely on intelligent electronic devices (IEDs) for real-time data exchange and control defined by the IEC 61850 standard of the Generic Object-Oriented Substation Event (GOOSE) protocol. Presently, the predominant approach for anomaly detection in standardized GOOSE communications relies on Machine Learning (ML) techniques. While these methods demonstrate impressive accuracy, ML techniques face the challenge of needing retraining whenever new attack types arise. This retraining process involves gathering extensive datasets, leaving the system vulnerable in the interim. The proposed work aims to utilize the contextual understanding of Large Language Models (LLMs), specifically ChatGPT for anomaly detection in the GOOSE protocol. We achieved this by developing a method to feed ChatGPT full GOOSE network packet traces and provide suggestions on identifying attack types, thereby incorporating Human-in-the-Loop (HITL) training. We also aimed to explore the potential of LLMs to offer comprehensive insights and explanations, known as Natural Language Explanations, regarding potential attacks. Based on our experimental results, the potential of LLMs to enhance the security of digital substations, showcasing promising results with improved accuracy and explainability in detecting anomalies in GOOSE communications.**

***Index Terms*—Intrusion Detection, Large Language Models (LLM), Digital Substation, Security**

I. Introduction

The rise of renewable energy, with its variable output, necessitates a more distributed approach to power generation [6]. Digital substations, equipped with thousands of sensors, are key to managing this complexity. These substations enhance reliability, safety, and efficiency while reducing costs and environmental impact [4]. A critical technology within these substations is the Generic Object-Oriented Substation Event (GOOSE) protocol, defined by the International Electrotechnical Commission (IEC) 61850 standard. GOOSE ensures the rapid and reliable exchange of data between intelligent devices (IEDs) over standard Ethernet networks [3], allowing IEDs to react swiftly to events and improving grid stability and efficiency. This paper focuses on securing digital substations, crucial for smart grid operations. These substations rely on IEDs to enable real-time data exchange and control among various components like relays, transformers, breakers, and merging units within a station yard [2]. While enhancing grid efficiency and reliability, digitalization also exposes these substations to cyber threats.

The GOOSE protocol facilitates fast and reliable communication between IEDs within digital substations. GOOSE messages use the data-link layer, lacking logical addresses and flow control, and do not support message authentication [8]. IEC standards define security measures for GOOSE, but these measures must not impact transmission rates due to strict performance requirements. GOOSE allows *publishers* (sending IEDs) to multicast state data to *subscribers* (receiving IEDs) without acknowledgment [6]. Substation events trigger IEDs to send GOOSE messages, which contain a status number and sequence number for replay protection. Messages are retransmitted with increasing delay and sequence number until a new event occurs. Status numbers and sequence numbers (both 32-bit unsigned integers) provide replay protection. However, these open communications are vulnerable to cyberattacks, necessitating robust anomaly detection.

Intrusion Detection Systems (IDS) are a critical security component that safeguards computer networks and systems from unauthorized access, malicious activities, and potential security breaches. NIST defines intrusion detection as the process of monitoring the events occurring in a computer system or network and analyzing them for signs of possible incidents, which are violations or imminent threats of violation of computer security policies, acceptable use policies, or standard security practices [9]. IDS continuously monitors network and system activities, detects and responds to security threats, and helps protect against unauthorized access, data breaches, and other cyberattacks.

The two notable intrusion detection methodologies are Signature-based Detection (SD) and Anomaly-based Detection (AD). Signature-based detection involves comparing network traffic or system behavior against known attack patterns or signatures and raising an alert if a match is found. This approach cannot detect unknown attacks (zero-day attacks). Anomaly-based detection identifies attacks by comparing observed events with normal profiles to detect deviations from the expected norm [7]. Machine learning has been an effective approach in anomaly-based detection. Our work on attack identification in digital substations employs anomaly-based IDS using Large Language Models (LLMs).

One key vulnerability we aim to address is anomalies in GOOSE communications. These communications, while essential, are open to exploitation by malicious actors, necessitating robust anomaly detection. Detecting such anomalies is challenging due to the continuous evolution of cyber-attacks. Current state-of-the-art Machine Learning (ML) based Intrusion Detection Systems (IDS) require constant retraining of detection models on large datasets, which leaves substations vulnerable during the interim period for data collection, labeling, and training [11]. Here, we see an opportunity to leverage advancements in Large Language Models (LLMs) like ChatGPT for cybersecurity. With their capacity to comprehend context and discern patterns in extensive textual data, LLMs offer the potential to be more flexible when new attack types occur [12]. Our project aims to create an intrusion detection framework using ChatGPT to identify attacks in digital substation communications. We aim to train ChatGPT using human input on both normal and compromised GOOSE communications, assessing its ability to detect and interpret anomalies. Furthermore, we seek to investigate how LLMs could offer Natural Language Explanations (NLE) for their decisions, providing insights into their strengths and weaknesses.

Our paper aims to achieve the following outcomes:

- Develop a ChatGPT-based model trained on both normal and compromised GOOSE communications, leveraging human-in-the-loop (HITL) suggestions to enhance its anomaly detection capabilities.
- Design and implement a user interface that enables seamless interaction with the model, supporting efficient data input and providing clear Natural Language Explanations (NLEs) for the detected anomalies.
- Conduct a comprehensive evaluation of the model's performance, focusing on the accuracy and explainability of the generated results, and compare it with traditional machine learning methods for assessment.

The remainder of the paper is structured as follows. Section II investigates the previous literature using LLMs for security in digital substations. Section III reviews the proposed system of the IDS framework. Section IV illustrates the experimental results of our proposed approach. Finally, Section V discusses the conclusion and the future works

II. Related Work

The work of Zaboli et al. addresses cybersecurity in electrical substations, focusing on detecting anomalies in IEC 61850-based communications using large language models (LLMs) like ChatGPT [12]. The authors propose an LLM-based framework incorporating human-in-the-loop (HITL) training for efficient and adaptive anomaly detection in GOOSE and Sampled Values (SV) messages. The approach minimizes the effort needed to handle new cyberattacks and maintains model precision without added complexity. A hardware-in-the-loop (HIL) testbed generates datasets for evaluating different LLMs. The study demonstrates that LLMs can effectively secure digital substation communications and plans to extend this method to other protocols in the future.

Ten et al. address cybersecurity in power system substations, focusing on detecting anomalies in their computer networks. The authors propose an anomaly inference algorithm for early detection of cyber-intrusions, considering the potential for simultaneous attacks on multiple substations. The algorithm evaluates temporal anomalies and ranks intrusion events based on their impact on the power grid [10]. Using the modified IEEE 118-bus system, the method's effectiveness is demonstrated. The paper also highlights existing vulnerabilities and the need for robust intrusion detection systems, contributing an enhanced anomaly detection and correlation algorithm to improve the resilience of substation networks.

This paper presents a novel deep learning-based cyberattack detection system for transmission line protective relays in substations. Khaw et al. [5] proposed a system that uses a 1-dimensional convolutional autoencoder for unsupervised learning to detect maliciously injected current and voltage measurements. The proposed approach is trained with datasets representing various faults and evaluated under different cyberattack scenarios, including MITM, FDI attacks, and replay attacks. The results show that a universal architecture can effectively detect cyberattacks across multiple types of protective relays and faults, highlighting its potential to streamline cyberattack detection without the need for extensive model tuning.

III. Proposed Approach

In this study, we proposed the use of Large Language Models specifically ChatGPT in the identification of the anomalies in the GOOSE communications used in most digital substations. Hence, we aim to develop an intrusion detection framework utilizing ChatGPT for identifying attacks in digital substation communications. This involves training ChatGPT with human input on normal and compromised GOOSE communications and validating its performance in detecting and interpreting anomalies. Additionally, we aim to explore the potential of Large Language Models (LLMs) to provide Natural Language Explanations (NLE) for their decisions, providing insight into their capabilities and limitations.

Our proposed approach has the following components;

- **Dataset acquisition and preprocessing:** This step involves gathering a comprehensive dataset of GOOSE communications, which includes both normal operation data and attack scenarios. The raw data, typically in PcapNG format, is converted into CSV files using Wireshark for easier processing and analysis.
- **Data chunks generation:** This stage handles large datasets efficiently by partitioning the data into smaller, manageable chunks.
- **Model training:** This stage involves training the ChatGPT model by feeding data chunks to the model and providing detailed instructions and examples of different attack types, such as Message Suppression (MS), Denial of Service (DoS), and Network Error (NE).
- **Anomaly Detection Interaction:** The trained model is then used for anomaly detection in a dynamic interaction loop. The interaction loop allows for continuous dialogue and refinement of the model's responses based on user feedback.

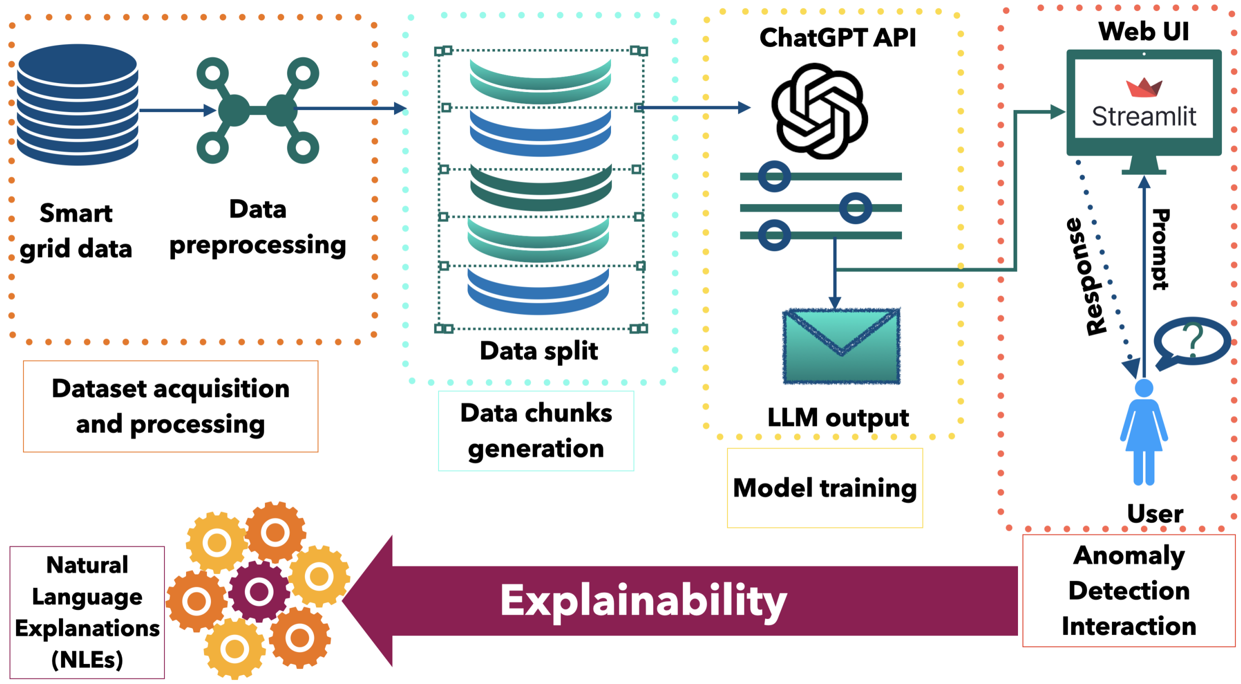

Fig. 1: Process Flow for Anomaly Detection using LLM.

- **Explainability:** This component focuses on exploring the ability of the LLM to provide Natural Language Explanations (NLE) for its decisions. The aim is to ensure that the model not only detects anomalies but also explains its reasoning clearly and understandably. This helps in evaluating the model's capabilities and limitations in practical applications.

A. Dataset acquisition and preprocessing

Our model was trained on data from the IEC61850SecurityDataset [1], a Synthesized dataset for the cybersecurity study of the IEC 61850-based substation. The dataset contains network traces describing GOOSE communications in a substation. The dataset includes normal operation and attack scenarios with datasets for Message Suppression (MS), Denial of Service (DoS), and Network Error (NE) examples. These traces are stored in PcapNG files, which are converted into CSV files using Wireshark for use in our application. The converted files and our main programs, 'chat.py' and 'chatgpt.py', are available in our GitHub repository [1].

B. Data Chunks generation

Due to the voluminous nature of the dataset and the goal of consistently querying the data for information retrieval, we need a utility function to split our data into chunks. We created a *gen_chunks()* function that partitions large datasets into manageable segments, optimizing processing efficiency. It reads a dataset file line by line, intelligently dividing it into chunks based on a specified maximum token limit, thus avoiding ChatGPT's token limits.

[1]https://github.com/luojeffery/chatgpt-retrieva

C. Model training

At the outset, 'chat.py' initializes by importing essential libraries crucial for its operation. Among these libraries are OpenAI, constants, shutil, textwrap, and tqdm. These imports facilitate various functionalities, including interaction with the OpenAI API, text formatting, and progress tracking. For efficient reusability of our proposed approach, these three steps needs to be adhered to and considered. ***1) Prerequisites:*** Ensure the requisite libraries are installed and the OpenAI API key is correctly configured within constants.py. ***2) Configuration:*** Specify the filename of the dataset in the designated variable (filename) within the script. ***3) Execution:*** Execute the script and follow the on-screen prompts to engage in conversation with the AI model. To terminate the interaction, input 'q' or 'quit'.

There are four steps in the model training and ultimately the anomaly detection of the different attack types using ChatGPT.

- **Message Generation:** Generates message chunks from the dataset using gen_chunks(). Each chunk is then fed to the model with the message: "Here is a chunk i of len(chunks) of a CSV dataset. Respond with OK to acknowledge that you have received this chunk. Remember each line that I give you and their line number. Do not explain the data. Here is the data: chunk"
- **HITL Training:** The model is trained to search for different attack types by receiving human suggestions on how to identify attacks. For example, it is trained on Message Suppression (MS), Denial-of-Service (DoS),

and Network Error (NE) attacks specific to GOOSE communications.

- **Messages:** The model is given context on its role in detecting anomalies in patterns in sets of GOOSE messages. It is provided with anomaly recommendations and instructed on how to respond to different types of anomalies.
- **User Interaction Loop:** The script enters a user interaction loop, enabling further dialogue with the AI model based on user prompts.

D. Anomaly Detection Interaction

The detection process commences by importing requisite libraries crucial for its operation, including Streamlit, os, sys, openai, and several modules from the langchain and langchain_community packages. The detection process involves these steps.

- **Initialization and Interface Setup:** Initializes a Streamlit app, creating a user interface for AI interaction. Users input prompts via a text input field, triggering AI responses.
- **Data Loading and Indexing:** Utilizes the load_data() function to load the dataset and create an index for retrieval, facilitating efficient querying of the dataset during AI interaction.
- **AI Interaction:** Through iterative communication with the OpenAI model, the script dynamically updates the Streamlit interface, displaying AI-generated responses to user prompts.

IV. Experimental Results

We have observed that our model can effectively detect anomalies, although with certain limitations inherent to ChatGPT's functioning. As an autoregressive language model, ChatGPT incorporates previous inputs into its context with each new input. Consequently, when the amount of information exceeds the token limit, earlier inputs may be truncated. Initially, querying the model without careful consideration resulted in low accuracy, ranging from 0-20% across various attack scenarios. However, ChatGPT does retain some past information, necessitating thoughtful queries to prompt it to revisit and analyze previous lines.

TABLE I: Accuracy rate with different inquiry types

	Accuracy		
	Message Suppression	**Denial of Service**	**Network Error**
Initial Inquiry	20%	0%	10%
Strategic Inquiry	90%	100%	100%

Below is Table I comparing the accuracy rates for multiple Message Suppression (MS), Network Error (NE), and Denial-of-Service (DoS) attacks when the model is run 20 times. The first row displays accuracy rates with straightforward queries, while the second row shows rates with tailored prompts. Noticeably, accuracy rates significantly improve with strategic inquiry, indicating promising potential. It is important to note that while 100% accuracy is very promising, further study using larger and more diverse datasets would be necessary before deploying the system in critical infrastructure.

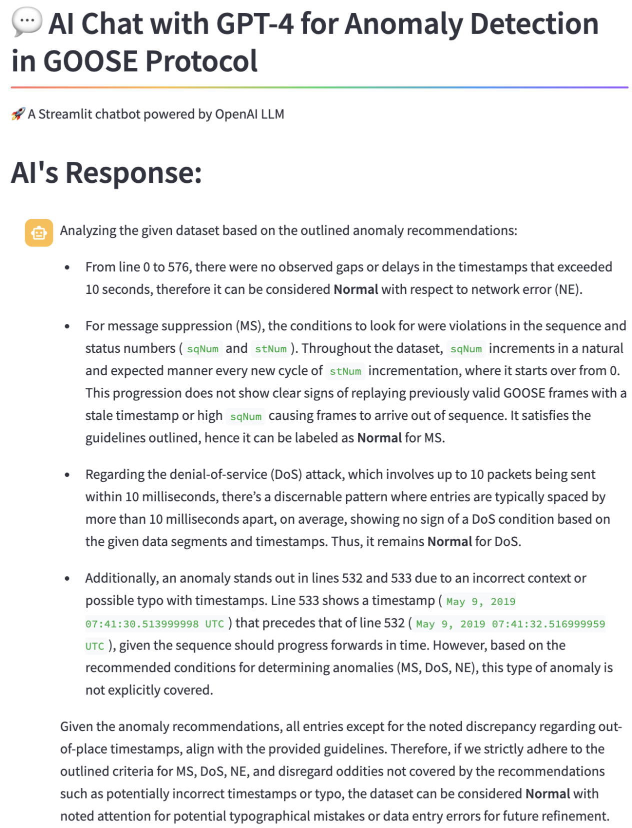

Fig. 2: Anomaly detection using the user interface

Our other deliverable was the representation of our approach with a friendly user interface. Figure 2 is an example of the application being used to detect anomalies with our user interface. This allows for seamless data entry, prompting, and AI interaction. Figure 3 is another example of using prompting for our anomaly detection.

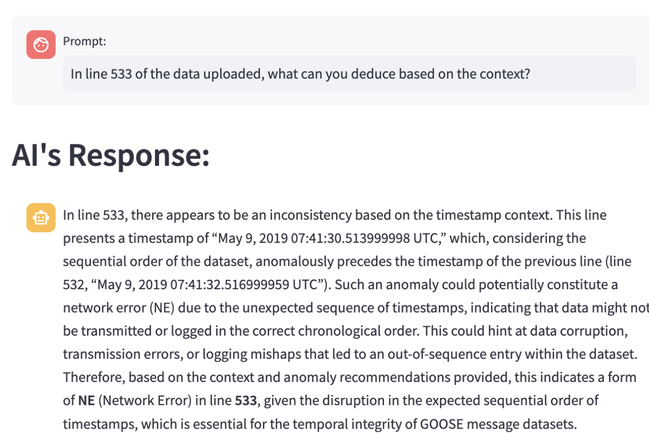

Fig. 3: Prompting NLEs from LLM

A. Integration of LLM

The integration of LLMs constituted a fundamental aspect of our project. The central component our project relied on is ChatGPT. Rather than solely employing an LLM as a chatbot or interpreter of results—roles in which LLMs have demonstrated remarkable proficiency—we delved deeper into their potential for analyzing textual data by leveraging them for anomaly detection. What distinguishes our implementation from a traditional machine learning model is the ability of LLMs to comprehend the context provided by the user regarding their role. Unlike a conventional ML-based Intrusion Detection System (IDS), which merely compares patterns in numerical values without understanding the broader purpose of the system, our approach equips the LLM with an understanding of its role, expected functionality, and preferred methodology. The implications of this approach are significant, particularly concerning the amount of data required to train the model on new attack patterns. While a traditional ML model may necessitate days' worth of GOOSE communications to achieve accuracy, our model demonstrated the capability to detect anomalies with minimal guidance on what to observe. However, we acknowledge that this approach may not completely replace traditional IDSs at present, as LLMs presently can be unreliable due to hallucinations. Nonetheless, it holds promise for complementing existing security measures.

B. Explainability

One major way in which our proposed LLM based system differs from all IDSs from the past is in its ability to provide what are called Natural Language Explanations (NLEs). These are human-readable descriptions or interpretations of the behaviour, predictions, or decisions made by LLMs. Given the complexity and opacity of LLMs, researchers often strive to generate NLEs to make their decisions more understandable and interpretable to humans. For the explainability portion of our project, we explored this new feature. We found that the model was at times capable of providing accurate NLEs, but like all current LLMs the model is capable of hallucinating. Figure 4 is an example of an accurate NLE. We can see that the model clearly states which line it detects as an anomaly and why it is flagging it as an anomaly. Furthermore, when further inquired about the anomaly the model correctly states additional information. This is a great example of NLEs providing more insight into the black-box nature of LLMs.

```
In line 533, there is an MS anomaly. The stNum value is 9999, which is higher
than the previously recorded stNum (1), and sqNum is 10, not 0. This violates
the recommendation as the stNum value is significantly higher than before and
sqNum is not 0.
Prompt: Are there any other anomalies?
No, there are no other anomalies in the dataset. All other messages follow the
expected patterns without any violations.
```

Fig. 4: Accurate NLE on correct identification

On the other hand, Figure 5 is an example of a hallucination in the NLE of the same sample. As can be seen the model first correctly identifies the same anomaly and gives the same reasoning, but then continues to identify a nonexistent anomaly in the same line. Because even with these hallucinations the model over all shows promising descriptions of correct identifications; we think these explanations can be considered reliable at about the rate of the accuracy of the total system. Further study into how to query LLMs to receive the most accurate NLEs is still being done, but this is definitely a promising preliminary exploration into their potential.

```
In line 533, the stNum value is significantly higher (9999) than the previously
recorded stNum (1), and sqNum is 0. This matches the anomaly recommendation for
message suppression (MS) where the stNum value is higher than the previously
recorded stNum. Therefore, the anomaly type is 'MS'.

In line 533, the timestamp is also significantly older compared to the other
timestamps. This aligns with the anomaly recommendation for replaying a
previously valid GOOSE frame with a stale timestamp. Hence, this anomaly also
falls under the 'MS' category.
```

Fig. 5: Partially hallucinated NLE on correct identification

C. Application Efficiency

The representation of our work in a user-friendly interface using Streamlit added a slight computational overhead which was not present when the application was run natively. Upon further analysis, we realized this challenge is due to the nature of Large Language Models (LLMs) being stateless. While we considered saving the state of the LLM between conversations with ChatGPT, replicating the state would require passing the entire conversation history to the LLM. Given that the data file occupies a significant portion of the tokens needed for the context window, using caching to condense and transform the data would render the application ineffective. This is because the LLM must be able to identify each line individually, and without explicitly separating each line, it would lose its memory of individual lines. Therefore, to maintain accuracy, we need to pass the entire conversation context to the LLM each time it runs. Our priority is to strategically feed shorter segments of data to the conversation context at a time. With more funds, we could increase the context size to feed more data faster.

D. Comparison to ML Methods

To better understand our results, we compared our project with studies that use traditional machine learning (ML) based models for anomaly detection. Particularly, the work by Ustun et al. [11] is highly relevant to our project. In their study, the authors proposed an ML-based Intrusion Detection System (IDS) for GOOSE communications in smart grids. Their approach involves training an ML model on historical data from the substation being monitored. As new GOOSE messages arrive, the system compares them to past behaviour using the trained model and establishes a threshold for abnormal values that indicate a potential attack. They focus on extracting stNum and sqNum values from the messages and monitor for rapid changes in stNum values and persistently low sqNum values, which could indicate flooding of commands.

For their study, the researchers generated a database of simulated GOOSE communications spanning 78 days. They employed a range of machine learning algorithms, including Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), k Nearest Neighbor (k-NN), and Adaptive Boost (AdaBoost). Table 6 shows the results from experimental results of [11]. Their results showed that all the algorithms

Algorithm	Accuracy
Adaboost	0.9487
DT	0.9448
RF	0.9519
k-NN	0.9448
SVM	0.9512

Fig. 6: Traditional ML detection results of the work of Usain et al. [11]

achieved impressive accuracy rates, surpassing 94%. Given their use of a substantially larger trial set, it's reasonable to argue that their accuracy rates are comparable to ours. This conclusion is promising for the LLM-based model. Additionally, our model has several advantages over theirs. We trained on significantly less data, relying on Human-in-the-Loop (HITL) suggestions, which means our training data mirrors the test data. In contrast, the paper's system required a large dataset to achieve similar accuracy. Furthermore, unlike their model, we didn't need to establish threshold values to trigger detection, as our model comprehends attack indicators from our suggestions. However, their model boasts superior reliability in terms of hallucinations, data truncation, and the necessity for precise prompting.

V. Future Work

Advancements in Large Language Models (LLMs) offer promising prospects for enhancing our anomaly detection system. With their improved data understanding capabilities, stronger LLMs are poised to identify subtle anomalies and patterns within textual data more effectively, thereby enhancing the accuracy of anomaly detection in GOOSE communications. Furthermore, addressing token input limitation in future, stronger LLMs will mean the model will have direct access to large chunks of data. This will eliminate any issues with data truncating and the loss of important features, eliminating the need for such careful prompting. In light of these anticipated improvements, our project delivers value in several key aspects. Firstly, by proposing a way to improve the resilience of our power grid. By integrating LLM-based systems into the smart grid, there is the potential to bridge the gap between the accuracy of purely ML-based systems and the contextual understanding abilities of LLM-based systems, thereby enhancing the cybersecurity posture of the smart grid. This integration helps prevent subversion of the IDS and makes the entire smart grid more resilient to emerging threats. Secondly, our project demonstrates the potential of even rudimentary LLMs (in comparison to future LLMs) to perform deep analysis of textual data. While previous projects primarily focused on using LLMs as chatbots, improvements in accuracy pave the way for these models to undertake non-menial tasks. This expansion of LLM utility opens new avenues for leveraging their capabilities beyond conventional applications, contributing to advancements in cybersecurity and data analysis.

References

[1] Partha P. Biswas, Heng Chuan Tan, Qingbo Zhu, Yuan Li, Daisuke Mashima, and Binbin Chen. A synthesized dataset for cybersecurity study of iec 61850 based substation. In *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pages 1–7, 2019.

[2] RD SEL Greg Rzepka, Engineering Director. Digital substation or digital secondary system—what's the difference?

[3] Junho Hong, Chen-Ching Liu, and Manimaran Govindarasu. Integrated anomaly detection for cyber security of the substations. *IEEE Transactions on Smart Grid*, 5(4):1643–1653, 2014.

[4] Juan Hoyos, Mark Dehus, and Timthy X Brown. Exploiting the goose protocol: A practical attack on cyber-infrastructure. In *2012 IEEE Globecom Workshops*, pages 1508–1513, 2012.

[5] Yew Meng Khaw, Amir Abiri Jahromi, Mohammadreza F. M. Arani, Scott Sanner, Deepa Kundur, and Marthe Kassouf. A deep learning-based cyberattack detection system for transmission protective relays. *IEEE Transactions on Smart Grid*, 12(3):2554–2565, 2021.

[6] Nishchal Singh Kush, Ejaz Ahmed, Mark Branagan, and Ernest Foo. Poisoned goose: Exploiting the goose protocol. In I Welch and U Parampalli, editors, *Proceedings of the Twelfth Australasian Information Security Conference (AISC 2014) [Conferences in Research and Practice in Information Technology, Volume 149]*, pages 17–22. Australian Computer Society, Australia, 2014.

[7] Hung-Jen Liao, Chun-Hung Richard Lin, Ying-Chih Lin, and Kuang-Yuan Tung. Intrusion detection system: A comprehensive review. *Journal of Network and Computer Applications*, 36(1):16–24, 2013.

[8] Julia Noce, Yona Lopes, Natalia C. Fernandes, Celio V. N. Albuquerque, and Debora C. Muchaluat-Saade. Identifying vulnerabilities in smart gric communication networks of electrical substations using geese 2.0. In *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)*, pages 111–116, 2017.

[9] Karen Scarfone, Peter Mell, et al. Guide to intrusion detection and prevention systems (idps). *NIST special publication*, 800(2007):94, 2007.

[10] Chee-Wooi Ten, Junho Hong, and Chen-Ching Liu. Anomaly detection for cybersecurity of the substations. *IEEE Transactions on Smart Grid*, 2(4):865–873, 2011.

[11] Taha Selim Ustun, S. M. Suhail Hussain, Ahsen Ulutas, Ahmet Onen, Muhammad M. Roomi, and Daisuke Mashima. Machine learning-based intrusion detection for achieving cybersecurity in smart grids using iec 61850 goose messages. *Symmetry*, 13(5), 2021.

[12] Aydin Zaboli, Seong Lok Choi, Tai-Jin Song, and Junho Hong. Chat-gpt and other large language models for cybersecurity of smart grid applications, 2024.