LLM-Based Threat Query System

Non Technical Actors In AI Failures

A Thesis Prospectus
In STS 4500
Presented to
The Faculty of the
School of Engineering and Applied Science
University of Virginia
In Partial Fulfillment of the Requirements for the Degree
Bachelor of Science in Computer Science

By
Vishal Kamalakrishnan

December 13, 2024

On my honor as a University student, I have neither given nor received unauthorized aid on this assignment as defined by the Honor Guidelines for Thesis-Related Assignments.

ADVISORS

Ben Laugelli, Department of Engineering and Society

Wajih Ul Hassan, Department of Computer Science

Introduction

The introduction of generative AI tools, such as ChatGPT in 2022, marked a significant shift in the adoption and integration of artificial intelligence (AI) across many industries. Generative AI is highly efficient due to Natural Language Processing (NLP), allowing AI tools to reason through human language and perform complex tasks such as answering questions, generating code, or engaging in meaningful interactions with users (Stryker & Holdsworth, 2024). AI is used in cybersecurity applications by network administrators to generate Kusto Query Language (KQL) queries, which are employed by systems such as Microsoft Defender and Microsoft Sentinel. However, due to "AI hallucinations," large language models (LLMs) can sometimes generate incorrect, misleading, or simply bizarre outputs. Experts have also raised concerns about the potential misuse of AI generated code by malicious actors, further expanding the need for robust and accurate AI systems (Synk, 2023).

To address these challenges, I will propose a structured approach to refining LLM-based KQL query generation incorporating STRUCTCHEM (Ouyang et al.) principles and other learning techniques such as feedback loops. Using this approach, this project aims to improve the reliability and security of AI-based cybersecurity systems.

Despite significant concerns about the reliability of generative AI responses, the "AI hype-train" has led companies to invest heavily in AI technologies without fully addressing the risks associated with their use. This phenomenon is driven by both technical and social factors that contribute to the reliability issues in large language model (LLM)-generated responses and understanding these factors is crucial for developing successful generative AI products. To examine such mechanisms, I will draw on the STS framework of actor-network theory (ANT) to

analyze the failure of Air Canada's chatbot, which misled a mourning customer about phony flight discounts (Garcia, 2024). Specifically, I will investigate how interactions among technical and social factors such as the premature adoption of AI products, social pressures to adopt AI, and the flawed legal stance that the chatbot represents a separate entity from Air Canada led to this failure.

If new LLMs are developed with higher accuracy in generating KQL queries without properly analyzing the underlying social, corporate, and legal dimensions of AI tools, they might fail under more complex use cases and in real-time scenarios like the Air Canada chatbot. Therefore, because the challenge of LLM generation in cybersecurity is sociotechnical in nature, it requires attending to both its technical and social aspects. In what follows, I set out two related research proposals: a technical project proposal for a structured approach to refining LLM-based KQL query generation and an STS project proposal for examining the technical, legal, and social failures of Air Canada's chatbot.

Technical Project Proposal

The increasing reliance on AI in cybersecurity has highlighted the limitations of current AI models in generating accurate Kusto Query Language (KQL) queries. KQL is widely used in security operations centers (SOCs) for threat detection and response and is utilized by tools like Microsoft Defender and Microsoft Sentinel to manage security incidents (Birch, 2024). While Large language models (LLMs), such as ChatGPT, can assist in generating KQL queries for Microsoft Sentinel incidents, their responses are often inaccurate and require manual tweaking (Trent 2023). In order to fully harness the benefits of AI in cybersecurity applications, a more structured approach is needed for LLM-generated KQL queries for use in cybersecurity applications that will prioritize accuracy and deal with "AI hallucinations."

Many startups are emerging to address the challenges of using LLMs in cybersecurity. Companies like DeepKeep and Lasso Security focus on protecting AI models from adversarial attacks and ensuring that AI-generated code and queries are secure without needing access to sensitive data (Dang, 2023). Additionally, several open source models such as SecurityLLM from ZySec-AI, are available on websites such as Hugging Face or Kaggle. These models are fine-tuned for security applications and specifically trained on cybersecurity threats, compliance frameworks, and risk management (Siddi, 2024). Despite these advancements, there is still a significant gap in the accuracy and reliability in the underlying LLM model architecture, and the problem of "AI hallucinations" remains unaddressed.

The first step in my proposed technical solution involves creating a ground-truth dataset to validate existing LLM models. This dataset will consist of prompts paired with accurate KQL queries extracted from public GitHub repositories using Python-based web scrapers. The ground truth dataset will serve as a benchmark for evaluating existing LLMs like GPT-4 and Claude 3.5,

and Gemini. These AI tools will be measured based on how well their generated KQL Queries match with the queries in the dataset and will also be tested in a simulated network environment to verify whether they produce the expected results.

The second step is to create a framework that will apply principles from STRUCTCHEM, a structured learning framework used in chemistry reasoning (Ouyang et al., 2024), to generate KQL accurate queries. This framework incorporates structured instruction, formulae generation, and a confidence-based iterative review process that will help refine AI-generated results by providing AI tools with additional context. This additional context will help mitigate the "AI hallucination" problem by making LLMs validate their responses. Additionally, refinements will be made by incorporating advanced learning techniques such as gradient-based learning and feedback loops.

The third step involves validating the results of the STRUCTCHEM approach using the ground-truth dataset and iteratively refining the approach to ensure maximum accuracy in KQL query generation. The results will then be tested in real-world environments incorporating tools such as Microsoft Sentinel, Microsoft Defender, and Microsoft Defender XDR. Finally, all results and instructions for reproducing the project will be posted on GitHub to ensure that other developers have the opportunity to improve and learn from this project. By directly tackling the deficiencies of existing AI tools when tested against the ground-truth dataset and by applying this structured approach, I assert that my approach will achieve greater accuracy than existing LLMs for generating KQL Queries.

STS Project Proposal

Introduced in 2022, Air Canada's chatbot was designed to assist customers with inquiries related to flight bookings and policies. The chatbot incorrectly informed customer Jake Moffatt about the airline's bereavement fare policy and stated he could book a full-price flight and apply for a partial refund within 90 days. In reality, Air Canada's bereavement policy requires that bereavement discounts be applied for prior to booking a flight. This misinformation led Moffatt to book a $1200 ticket following the death of his grandmother, and when he later sought the discount promised by the chatbot, Air Canada refused to honor it (Hawley, 2024).

Previous analyses of AI failures have often focused on purely technical shortcomings or human error. A common argument is that failures such as Air Canada's chatbot incident stem from inadequate training data or poor algorithmic design (Fui-Hoon Nah et al., 2023). Critics argue that because AI projects are so heavily dependent on the data they are trained from, AI failures are due to data quality, quantity, and the divide between training and real-world data. This discrepancy is most adequately summed up by the classic phrase - "Garbage In, Garbage Out" (Rschmelzer, 2023). These analyses, however, tend to overlook interactions between technical and social factors that contribute to such incidents. For this reason, in the case of Air Canada's chatbot, I contend that it was not merely technical failure, but a combination of social factors - such as the pressure to adopt AI features, corporate policies, and legal stances that led to its failure.

Current discourse has not adequately discussed how premature adoption of AI products, which are largely driven by corporate pressures for innovation, can exacerbate AI failures. The prevailing view of generative AI is shaped by technological determinism and the belief that AI is inevitable, compelling organizations to either embrace it or risk falling behind. Due to this,

companies often rush to implement AI technologies without fully understanding their limitations or thoroughly testing their product (Constantin, 2024). This “AI hype-train” has also prompted a strong opposing response from academic institutions, which have resorted to banning AI generated content due to fear that it will lead to students cheating on assignments (David, 2023).

Current discourse has also not addressed the pressing issue of the legality of AI generated content and about who is to blame for AI failures. In response to the lawsuit regarding the misleading information provided by its chatbot, Air Canada attempted to deflect responsibility by blaming the customer for not verifying the chatbot's response. The airline argued that it was Moffatt’s obligation to verify the bereavement policy by clicking on the link provided by the chatbot, which directed them to another section of the website containing the correct bereavement policy (Forbes, 2024). Additionally, Air Canada also claimed that the chatbot is a separate entity from the company and so could not be held accountable for its responses (Garcia, 2024).

To frame my analysis of the failure of Air Canada chatbot, I will draw on the science, technology, and society (STS) concept of actor-network theory (ANT). ANT is a framework that describes how human and non-human actors participate in networks that shape sociotechnical systems. ANT describes these systems by studying the “associations between heterogeneous actors” to describe how networks gain and lose power and shape technologies (Cressman, 2009). Using ANT, I will analyze how various actors such as Air Canada’s corporate leadership, the “AI hype train”, customers’ expectations, and legal stance interacted within a networked system to create a misleading Chatbot and Air Canada’s response to the lawsuit. To support my argument, I will draw on evidence from news media articles, Air Canada’s AI adoption plan, employee memos, and legal policies.

Conclusion

Both technical and social aspects of AI-driven systems must be addressed to ensure that AI-generated KQL queries can effectively be integrated into security applications without resulting in the "AI hallucination" problem when tested in real conditions. The technical project will improve KQL generation using STRUCTCHEM principles and maintaining response quality by testing and refining prompts iteratively using the ground-truth dataset. This approach will benefit cybersecurity professionals who are utilizing KQL queries for applications involving network administration, network security, and organizational management.

The STS project will provide a deeper understanding of the social factors that lead to premature adoption and failure of generative AI projects. Using actor-network theory (ANT), I will analyze how human and non-human actors such as corporate policies, profit-driven executives, and social pressures lead to defective AI products. By drawing insights from the STS project, I can help mitigate risks and apply a holistic approach to the technical project to ensure accuracy and reliability when deployed in real-world environments.

1750 words

References

Birch, D. (2024, January 30). *Unlocking the potential: Large language models in security operations centres*. LLMs In SoC. https://cybanetix.com/llms-in-soc/

Constantin, L. (2024, September 19). *Companies skip security hardening in rush to adopt AI*. CSO Online. https://www.csoonline.com/article/3529615

Cressman, D. (2009). A Brief Overview of Actor-Network Theory: Punctualization, Heterogeneous Engineering & Translation.

Dang, W. L. (2023, September 6). *LLMS in security: What are the opportunities for startups?*. Unusual Ventures. www.unusual.vc/post/llms-security

David, E. (2023, February 23). *The rise of AI is inevitable. It's time to embrace ChatGPT*. Business Insider. https://www.businessinsider.com/generative-ai-chatbot-chatgpt-bill-gates-business-schools-2023-2

Fui-Hoon Nah, F., Zheng, R., Cai, J., Siau, K., & Chen, L. (2023). Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. *Journal of Information Technology Case and Application Research*, 25(3), 277–304. https://doi.org/10.1080/15228053.2023.2233814

Garcia, M. (2024, February 20). *What Air Canada lost in "remarkable" lying AI chatbot case*.

Forbes. https://www.forbes.com/sites/marisagarcia/2024/02/19/what-air-canada-lost-in-remarkable-lying-ai-chatbot-case

Hawley, M. (2024, April 2). *Exploring Air Canada's AI Chatbot Dilemma*. CMSWire.com. https://www.cmswire.com/customer-experience/exploring-air-canadas-ai-chatbot-dilemma

Ouyang, S., Zhang , Z., Yan , B., Liu , X., Choi , Y., Han , J., & Qin, L. (2024). Structured Chemistry Reasoning with Large Language Models (dissertation). arXivLabs. Retrieved from https://arxiv.org/abs/2311.09656

Rschmelzer. (2023, December 26). *Top reasons why AI projects fail*. Cognilytica. https://www.cognilytica.com/top-10-reasons-why-ai-projects-fail

Siddi, V. (2024, February). *ZYSEC-ai/SecurityLLM · hugging face*. ZySec-AI/SecurityLLM. Hugging Face. https://huggingface.co/ZySec-AI/SecurityLLM

Stryker, C., & Holdsworth, J. (2024, August 11). *What is NLP (Natural Language Processing)?*. IBM. https://www.ibm.com/topics/natural-language-processing

Synk. (2023). *AI code, security, and trust in modern development*. Snyk. https://go.snyk.io/2023-ai-code-security-report-dwn-typ.html

Trent, R. (2023, March 24). *Generating KQL from Microsoft Sentinel incidents with chatgpt*.

Rod's Blog. https://rodtrent.substack.com/p/generating-kql-from-microsoft-sentinel