Proceedings of the Annual General Donald R. Keith Memorial Conference
West Point, New York, USA
April 28, 2022
A Regional Conference of the Society for Industrial and Systems Engineering

Counter-AI Tool System Design for AI System Adversarial Testing and Evaluation

Nathan Byington[1], Carter Davis[1], Matthew Meehan[1], Caroline Vincent[1], David Woodward[1], and Nathaniel Bastian[1,2]

[1]Department of Systems Engineering
United States Military Academy
West Point, NY 10996

[2]Army Cyber Institute
United States Military Academy
West Point, NY 10996

Corresponding author's Email: nathaniel.bastian@westpoint.edu

Author Note: Cadets Byington, Davis, Meehan, Vincent, and Woodward are fourth year students in the Department of Systems Engineering (DSE) at the United States Military Academy. This project is the culminating event for graduating DSE students to showcase the academic progress made throughout the previous years of learning within DSE. This project is supported by MAJ(P) Nathaniel Bastian, our Capstone Advisor and Assistant Professor, and sponsored by Mr. Frank Geck, Acting Chief of the Defensive Cyber Innovations Branch at U.S. Army DEVCOM C5ISR Center, under Support Agreement No. USMA21056.

Abstract: This work consists of the initial recommendations and conclusions found while soliciting functional requirements for the research, design and development of a Counter-AI Tool for conducting adversarial testing and evaluation of artificial intelligence (AI) systems. The report includes a literature review of relevant AI concepts and extensive research within the adversarial AI domain. An intensive stakeholder analysis, consisting of requirement elicitation from over twenty governmental and non-governmental organizations, assisted in determining what functional requirements should be included in the system design of a Counter-AI Tool. The subsequent system architecture diagram takes user input, tests for various types of adversarial AI attacks, and outputs the vulnerabilities of the AI model. Prior to the operationalization of this tool, iterative experimentation will be conducted by partner organizations, which is the next step in the development and deployment of this Counter-AI Tool.

Keywords: Adversarial AI, System Design Architecture, AI Security, AI Resiliency, Testing and Evaluation

1. Introduction

The U.S. Department of Defense (DoD) has become increasingly interested in using artificial intelligence (AI) technology to enhance both military mission capabilities and everyday tasks. The DoD defines AI as an "artificial system designed to think or act like a human, including cognitive architectures and neural networks" (Sayler, 2020). It defines adversarial AI as the "countermeasures that adversaries may deploy against our AI systems, and the evaluation steps and defenses needed to safeguard performance" (DoD, 2018). The DoD has pledged to research new theory, techniques, and tools to make AI systems more resilient and exhibit less unexpected behavior. The DoD's strategy overview includes delivering AI-enabled capabilities that address key missions, scaling AI's impact across the DoD through a common foundation, cultivating a leading AI workforce, engaging with various partners, and leading the world in military ethics and AI safety (DoD, 2018).

However, with the increase in AI system implementation and adoption, adversaries have threatened the attack and manipulation of these systems; currently, there is no tooling readily available to help conduct adversarial testing and evaluation (T&E) of AI systems to assess vulnerabilities and failure models prior to their operationalization. In mission use cases, the DoD should not deploy these AI systems without a prior assessment of the effectiveness of security, or Counter-AI measures. Designing and building resilient AI systems is vital to AI defenses because these systems are more explainable, trustworthy, and secured against various identified methods of adversarial attacks.

As such, the DoD aims to ensure that deployed AI systems are more secure to protect against adversarial manipulation. Adversaries will attack AI based on three access paradigms: white box, black box, grey box. White box attacks give the attacker the highest capability, which occurs when the adversary has access to all model components (Kurakin, 2018). In black box attacks, the adversary does not have a fully transparent view of the model but is able to probe the model to infer its structure

and components (Kurakin, 2018). The last in order of attacker capability is grey box (or hidden box) attacks, which is when the adversary has no direct access to the model and can only make assumptions about the model's structure (Kurakin, 2018).

Potential threats to the AI system include various modes of attack, such as poisoning, evasion, and model inversion. Poisoning attacks are ones that pollute the training data to skew model behavior, such as incorrectly classifying user inputs into the AI system (Bae, 2021). Evasion attacks do not directly impact the training data, but effectively obscure the content it delivers, making the attack both invisible to human observers, AI system recognition and classification (Bae, 2021). Model inversion (stealing) attacks occur when the adversary probes the AI system to extract information about the model configuration or training data to effectively reconstruct the model (Bae, 2021). All three of these adversarial attacks pose a diverse array of consequences for deployed AI systems, most notably with user privacy and data security.

Given the perceived threat and lack of tooling to adequately assess adversarial AI vulnerabilities, our work seeks to understand how a Counter-AI Tool could be designed, developed, and utilized to help protect AI systems against these newfound adversarial threat vectors. Specifically, our work contributed to and supported the research, design, and development of a Counter-AI Tool for adversarial T&E of AI systems to be used by AI Red Teamers to improve AI system resiliency.

2. Literature Review

To enable problem definition and further develop our knowledge on adversarial AI, we conducted a thorough literature review covering five distinct topic areas: adversarial AI attacks, adversarial AI defense techniques, counter-AI red teaming performance metrics, AI engineering and AI DevSecOps, and software system design architecture and AI system integration.

First, there are many tactics and techniques for adversaries to attack AI systems. The three main categories are poisoning attacks, evasion attacks, and model inversion (Bae, 2021). Each attack targets a different aspect of the AI system, and each attack depends on the degree of accessibility an adversary has: white box, grey box, or black box (Kurakin, 2018). As a result, adversaries will use different algorithms and tools that allow them to manipulate data and/or certain coding to employ their own functions within the AI system. These perturbations can cause the AI system to make incorrect classifications, predictions and inferences. To protect again adversarial AI attacks, there are limited existing, yet prototype, tools that can help improve and defend against AI attacks, such as Adversarial Robustness Toolbox, Foolbox, and CleverHans (Chang, 2020).

Second, AI is important to understand from a defensive standpoint because for the AI system to be deployed, trusted and adopted by users, it must be able to be robust and resilient against adversarial AI attacks. A strong knowledge base of AI defense techniques allowed us to better understand how to architect an example user-interface of a Counter-AI Tool that outputs recommended AI defense mechanisms to the user. This also informed us of current trends in defending against adversarial AI attacks, which include adversarial training, randomization, denoising, provable defenses, and ensemble defenses. Adversarial training defenses train the model to recognize and defend against attacks through improving the robustness of the model. Denoising and randomization are both defenses that mitigate a poisoning attack by disrupting the perturbation of the attacker. Finally, provable defenses are defenses that have been tested and proven while ensemble defenses are a combination of defenses designed to prevent multiple attacks. (Byington et al., 2021).

Third, essential to the development of a Counter-AI Tool is the ability for Red Teamers to test the performance and evaluate the robustness of the AI system through mimicked adversarial AI attacks. Red Teamers provide the important function of identifying weaknesses in the system, which, therefore, allows the system to be updated and become more resilient against AI attacks. Red Teamers must take various approaches, utilizing multiple modes of attack while also attacking at various levels of AI system knowledge such as white box, gray box, or black box attacks (Byington et al., 2021). Several studies in the existing literature have considered a diverse array of Red Team performance metrics that are both qualitative and quantitative in nature (MITRE, 2020). These Red Teaming assessments help AI developers to understand adversarial considerations such as origins, common types of attacks, flexibility, and knowledge of systems to better evaluate how to create more robust AI systems overall.

Fourth, AI engineering and AI DevSecOps strive to create AI systems to mimic human decision-making and solve complex problems. The only constant in AI systems is change; they are constantly evolving. The complexity of such systems widens the attack surface and makes them vulnerable to adversarial AI threats. AI engineers must design robust and secure AI systems that manage the inherent risk. The AI engineering framework includes AI technologies and components, the AI system, human-machine teaming, infrastructure, and process and policy (Yasar, 2020); there now exists foundations for AI engineers to design secure systems (Horneman et al., 2019). AI DevSecOps emphasizes the collaboration of all developers and stakeholders throughout the process while valuing the integration of security into AI system design (Yasar, 2020).

Lastly, the purpose of our research into software system design architectures and AI system integration was to explore the different types of system architectures along with AI integration requirements for a Counter-AI Tool to be able to assess AI models for vulnerabilities an adversary could exploit. Our findings aided in the architecture design to determine the most critical functional requirements necessary for the tool, evaluate preexisting options to find the best software system architecture approach, and how to integrate software system components into a tool that detects vulnerabilities (Byington et al., 2021).

3. Stakeholder Analysis

To facilitate system design, we followed the stakeholder analysis process depicted in Figure 1, which started with stakeholder identification. We identified over 20 organizations in the Army, DoD, Intelligence Community, Federally Funded R&D Centers, and University-Affiliated Research Centers that had expertise and/or interest in adversarial AI. We split the stakeholders into categories based upon defined perceived roles in the research, design, development and/or use of a Counter-AI Tool. Stakeholder roles included: client, consumer, decision authority, owner, partner, and user (Byington et al., 2021). While some stakeholders qualify as multiple types, we assigned the best fit classification for each stakeholder in Table 1.

Figure 1. Systems Engineering Stakeholder Analysis Process

Table 1. List of the Identified Stakeholders

Owner	User	Partner (1)	Partner (2)	Partner (3)	Consumer	Decision Authority
• Army Cyber Command (ARCYBER) • National Security Agency (NSA)	• Army Artificial Intelligence Integration Center (AI2C) • Army Testing and Evaluation Command (ATEC) • DoD Joint Artificial Intelligence Center (JAIC)	• Army Research Laboratory (ARL) • Defense Advanced Research Projects Agency (DARPA) • Intelligence Advanced Research Projects Activity (IARPA) • The MITRE Corporation (MITRE)	• Carnegie Melon University Software Engineering Institute (CMU SEI) • Pennsylvania State University Applied Research Laboratory (PSU ARL) • Georgia Tech Research Institute (GTRI)	• Massachusetts Institute of Technology Lincoln Laboratory (MIT-LL) • Research and Development Corporation (RAND) • Institute for Defense Analyses (IDA)	• Program Executive Office Command, Control, Communications-Tactical (PEO C3T) • Program Executive Office Enterprise Information Systems (PEO EIS) • Program Executive Office Intelligence Electronic Warfare & Sensors (PEO IEWS)	• Army Command, Control, Communications, Computers, Cyber, Intelligence, Surveillance and Reconnaissance (C5ISR) Center • Army Cyber Institute, United States Military Academy (ACI/USMA)

Next, we developed and employed both broad and specific questions to elicit system design requirement information from the stakeholders; this provided guidance for functional requirement development. Through interviews and surveys of over 15 of the 20 stakeholders, we obtained a better picture of each organization's role in the adversarial AI domain and their interests in developing this tool. The questions included in the stakeholder interviews and surveys are shown in Table 2.

Table 2. The Interview and Survey Questions Posed to the Identified Stakeholders

Structured Interview Questions	Unstructured Survey Questions
What makes your organization interested in adversarial AI?	If you were an end user of such a tool, what would the functions of the system do? What would the system components be? Inputs/outputs?
What are some key military mission use cases for AI that you think need to be tested for adversarial attack vulnerabilities via a Counter-AI Tool?	What sort of metrics and information should the tool output to the end user providing an assessment of the AI system's adversarial attack vulnerabilities (e.g., fooling rate, system confidence, recommendations of defensive measures)?
What should be the focus in terms of adversarial AI threat modeling? How worried should we be about the risks of adversarial vulnerabilities of AI systems (e.g., Data modality – where is data coming from? Are there redundant sensors? Data providence – is the data correct? Model providence – is the model appropriate?)	What would the system design architecture consist of to ensure the system is adaptable and flexible to new, evolving adversarial threats? How should the tool operationalize these threats in terms of attacker objective, attacker knowledge, and attacker capabilities?
How does adversarial AI T&E connect to traditional software testing as part of DevSecOps and AI engineering, more broadly? How does this relate to software testing in DevSecOps and AI Engineering?	From a system architecture perspective, how should the system account for the adversary access paradigm (white box, black box, grey box)?
What is your understanding of the process/workflow of AI system red teamers?	From an infrastructure and architecture perspective, how should this tool integrate with other AI system testing and evaluation tools? How would you handle maintenance of such a tool? Who should maintain this tool? What are the risks of application programming interfaces (APIs)?
What is your organization's timeline on the need of said technology? What is the echelon of the said technology focus (enterprise, edge, etc.)?	What would the user interface consist of? How do we make this adversarial attack vulnerability assessment useful for end users and decision makers? How would we visualize the severity of the risk assessed by the counter-AI tool?

ISBN: 97819384962-2-6

4. Functional Requirements Analysis

Upon completing stakeholder analysis, we conducted functional requirement analysis using the process shown in Figure 2. The functional hierarchy, depicted in Figure 3, provides an initial breakdown of the composition and functionality for the system design of the Counter-AI Tool. The top layer indicates the overarching functional requirement, which is the development of the Counter-AI Tool. Below the top layer are various sub-requirements that can be further broken down into more specific sub-tasks; these sub-tasks must be included in the design and development of the tool to maximize effectiveness.

Figure 2. The Process of Developing Final Functional Requirements Recommendations

To analyze the functional requirements for a Counter-AI Tool, we utilized a mixed-methods approach that included both qualitative and quantitative analyses. Qualitative analysis provided a surface-level understanding of the tool requirements by manually analyzing and grouping similar phrases and functions into groups, allowing us to sort through a wide range of stakeholder responses. Quantitative analysis included text mining, which is an analytical process for deriving frequencies of key words and insights from the text corpus of the stakeholder response data (after cleaning and wrangling the data).

To confirm our findings from the qualitative analysis, we used text mining as a data-driven approach to help identify additional un-discovered key takeaways from the stakeholder feedback. Specifically, we ingested and curated the stakeholder interview and survey data, tokenized the responses, removed stop words, used term frequency analysis, and then built bigram network plots to see the frequency of two words used together and what other words they were connected to in the stakeholders' responses. This allowed us to concentrate on specific key words and confirm our qualitative analysis of stakeholder feedback.

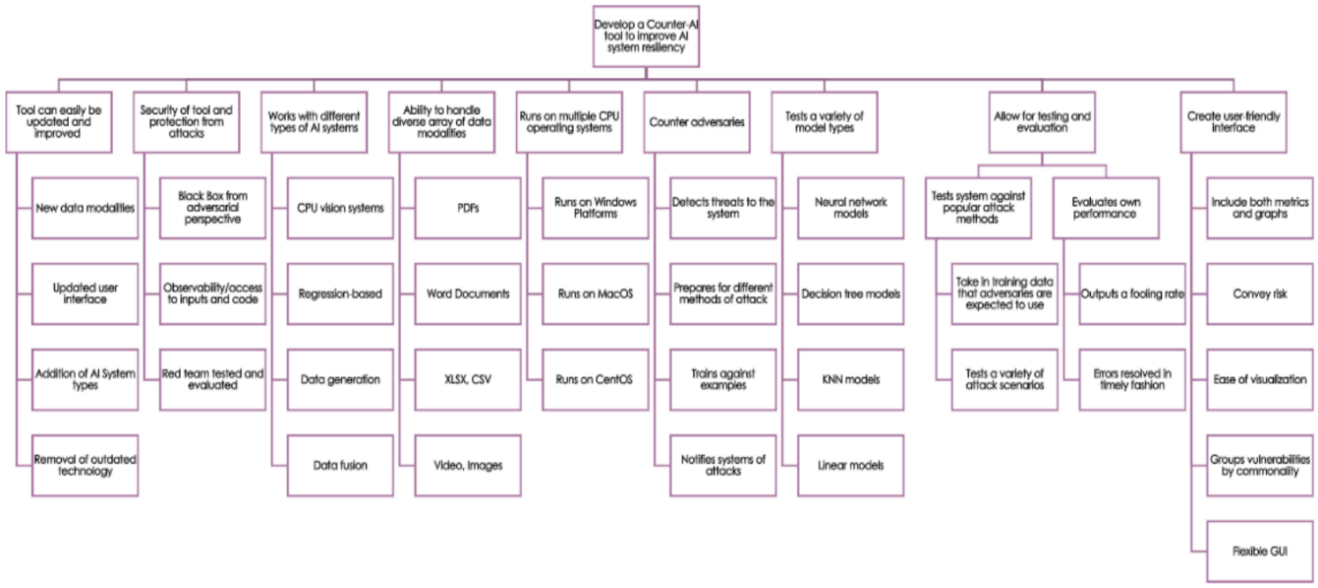

Figure 3. Counter-AI Tool System Design Functional Hierarchy

Next, we created a Findings, Conclusions, and Recommendations (FCR) matrix to summarize findings, which is a method that helped condense results drawn from previous portions of the research. Our findings are specifically summarized from the stakeholder surveys and interviews, and the conclusion provides the "so what" or significance behind the findings. The recommendations provide clear and specific development considerations for what the engineering team should consider in building the Counter-AI Tool. Table 3 provides a summary of recommendations drawn from each individual FCR matrix.

Table 3. Summary of Recommendations Drawn from the Final FCR Matrices

Recommendations	
Stakeholder organizations are interested in working together to fund and promote both governmental and private development. These organizations should publicly publish clear findings to encourage interest by larger organizations and the advancement of knowledge within the growing field.	Create and standardize the AI red teaming workflow and processes. Implement the AI red teaming processes using the Counter-AI Tool with a focus on gray box attack adversarial scenarios.
Key military use cases include satellite imagery protection, cyber-attack detection, and malware detection. With detection there will be false positives and false negatives and it is important to determine which case is worst because the model will be skewed to one false over the others. Identify which risk we want to counter first and build up the tool one use-case at a time.	While the timeline is not as important as developing a fully functional Counter-AI Tool, the said technology is needed now.
The tool should focus on use cases where the adversary can realistically manipulate the data, whether that be poisoning it or perturbing it to evade. All data is vulnerable, so focusing on how to assess those vulnerabilities is a key function of the tool.	The Counter-AI Tool needs to test against multiple attacks to determine robustness and generate stimuli inputs to respond to a range of confounding information.
In general, the software acquisition community has not yet understood that AI technology is software. Because of this, we recommend increased awareness and a holistic look at AI capability development through the lens of software development with additional tools for testing AI unique vulnerabilities.	The Counter-AI Tool needs to be representative and adaptive. It should test against realistic, complex adversarial examples and continuously learn from each attack.

5. System Design Architecture

Utilizing findings from the functional requirements analysis, we determined an AI Red Teaming process and how a system workflow would be depicted through the system design architecture. The system design architecture, depicted in Figure 4, is focused on assessing vulnerabilities from evasion, poisoning, and inversion (stealing) attacks. The end user of the Counter-AI Tool provides input and answers questions through the User Interface, which will then determine the types of tests performed on the AI system components input (model, data, etc.) to determine and assess the vulnerabilities of the end user's input.

User Input: User Input provides the input needed by the Counter-AI Tool for vulnerability assessment. It can range from a model, code, and/or labeled data. The User Input will be used to test for various types of attacks to determine its vulnerabilities.
User Interface: This is what the end user will see. It contains a series of questions to determine the proper modules to activate assess the vulnerabilities of the user input. The questions include the AI use case and type of data in the input and the level of access a potential adversary may have. These answers will determine the vulnerability assessments conducted on the user input.
Evasion Module: The degree of adversary access will determine the types of evasion attacks that will be replicated. If it is a white box, meaning the adversary would have full access to the input, then adversarial examples of the model would be generated to determine how the model can be fooled. If it is a black box, a surrogate model will be created to replicate the model as close as possible and determine at what level the model can be fooled; transfer attacks would be tested.
Poisoning Module: The module detects for data poisoning within the input, specifically in the data used to train the model.
Stealing Module: The way to determine the vulnerability for stealing the AI model (model inversion) will be to attempt to recreate the model strictly from output of the model and backwards program a similar model and derive the data (a privacy, security concern). This module will test for the vulnerabilities of replicability of a model similar to the user input.
Vulnerability Assessment Module: This module is critical to the function of the entire Counter-AI Tool. This summarizes the findings of all the attack modules (Evasion, Poisoning, and Stealing) activates and displays, via the User Interface, the greatest AI system vulnerabilities to the end user. This will show the vulnerability areas, the spoofing rate, the accuracy rate, the robustness rate, and recommend AI defense techniques to make the AI model more robust and resilient to adversarial AI attacks.
Data Storage: The data storage layer is interwoven throughout the system design. All data will be stored within a data hub. The data storage includes the user input, user interface responses, attack module creations, and the vulnerability assessments.

6. Conclusions and Recommendations

The continuing trend of increased use and reliability on AI systems for DoD mission operations requires a tool that can help assess the security of AI systems prior to operationalization. Many organizations are interested in the development of such a tool that includes aspects of adversarial T&E, integrated AI system red teaming processes, and applicability to a wide range of AI use cases and military mission areas. The system design architecture depicted in Figure 4 provides our

recommended base structure for a Counter-AI Tool. For next steps, we recommend that user stakeholder organizations help identify prioritized Counter-AI use cases to provide the necessary development scope for a tool that can test against multiple realistic, adaptive attack scenarios, as well as create and standardize the AI Red Teaming workflow and processes. We also recommend that organizations publicly release findings on emerging adversarial AI threats to increase awareness to senior DoD leaders, as the most prominent limitation on tool development is sufficient funding for both development and sustainment. Opportunities for further work include expanding the system design architecture with system specification reference document, crafting Counter-AI Tool user stories, drafting user interface mockups, and continuing to improve adversarial AI awareness.

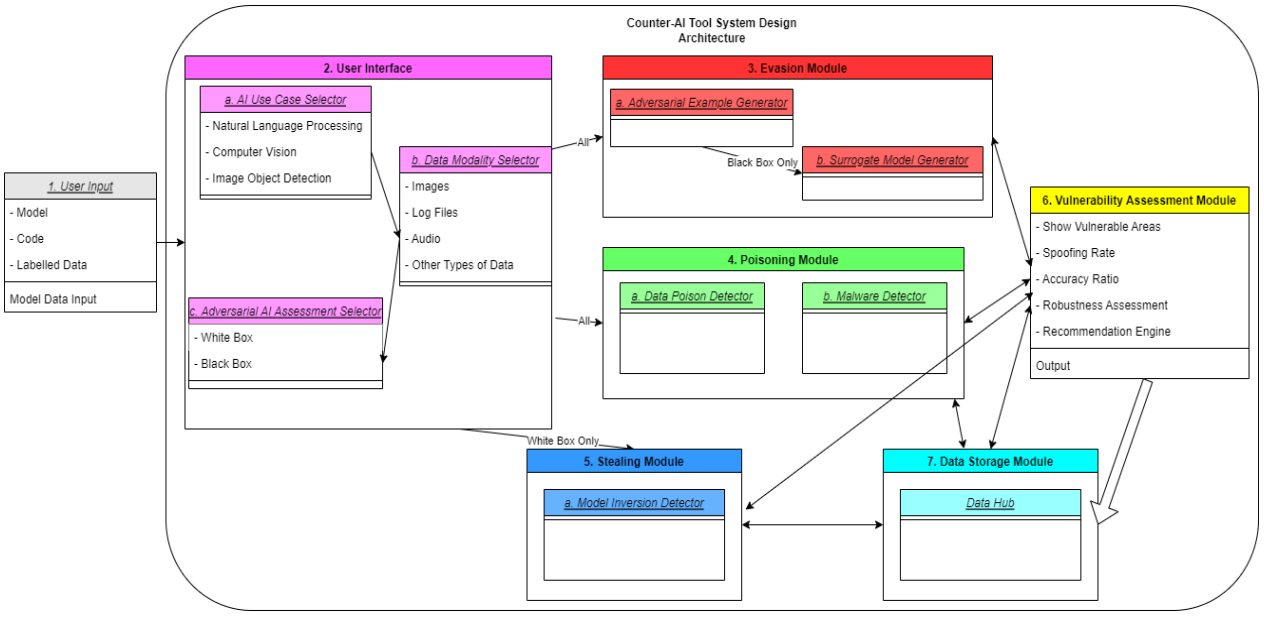

Figure 4. Prospective System Design Architecture for the Counter-AI Tool

7. References

Bae, H., Jang, J., Jung, D., Ha, H., Lee, H., & Yoon, S. (2021, March 10*). Security and Privacy Issues in Deep Learning.*

Byington, N., Davis, C., Meehan, M., Vincent, C., Woodward, D. and Bastian, N. (2021, December). *Counter-AI Tool System Design for AI System Adversarial Testing and Evaluation*. SE/EM 402 Interim Technical Report, Department of Systems Engineering, United States Military Academy, West Point, NY.

Chang, C.-L., Hung, J.-L., Tien, C.-W., Tien, C.-W., & Kuo, S.-Y. (2020, October 6). *Evaluating Robustness of AI Models against Adversarial Attacks (SPAI '20: Proceedings of the 1st ACM Workshop on Security and Privacy on AI).*

Department of Defense. *Artificial Intelligence Strategy*. (2018). https://media.defense.gov/2019/Feb/12/2002088963/-1/-1/1/SUMMARY-OF-DOD-AI-STRATEGY.PDF

Horneman, A., Mellinger, A., & Ozkaya, I. (2019, August). *AI Engineering: 11 Foundational Practices* (No. AD1099280). Carnegie Mellon University Software Engineering Institute. https://apps.dtic.mil/sti/pdfs/AD1099280.pdf

Kurakin, A., Goodfellow, I., Bengio, S., Dong, Y., Liao, F., Liang, M., Pang, T., Zhu, J., Hu, X., Wang, J., Zhang, Z., Ren, Z., Yuille, A., Hang, S., Zhao, Y., Zhao, Y., Han, Z., Long, J., Berdibekov, Y., . . . Abe, M. (2018, September 28). *Adversarial Attacks and Defenses Competition* (The Springer Series on Challenges in Machine Learning).

The MITRE Corporation. *Creating an AI Red Team to Protect Critical Infrastructure.* (2020, June 30). https://www.mitre.org/publications/project-stories/creating-an-ai-red-team-to-protect-critical-infrastructure.

Sayler, K. *Artificial Intelligence and National Security*. (2020, November 10). Congressional Research Service. https://sgp.fas.org/crs/natsec/R45178.pdf

Yasar, H. (2020, January). *Leveraging DevOps and DevSecOps to Accelerate AI Development and Deployment* (No. AD1110415). Carnegie Mellon University Software Engineering Institute. https://apps.dtic.mil/sti/pdfs/AD1110415.pdf