Proposed Uses of Generative AI in a Cybersecurity-Focused Soar Agent

Indelisio Prieto, Benjamin Blakely
[1]Argonne National Laboratory
9700 South Cass Avenue
Lemont, Illinois 60439
iprieto@anl.gov, bblakely@anl.gov

Abstract

With the rapidly increasing use of AI and machine learning in recent years and the current generative AI revolution, it is no surprise that the malicious use of AI has begun to establish itself in the realm of cybersecurity. At risk of being left behind in this "arms race", it's imperative that autonomous intelligent cybersecurity agent (AICAs) are developed to counter this emerging threat. Currently, a project at Argonne National Laboratory is using Soar as a starting point for developing a cognitive-architecture based AICA, but the utilization of Soar in this project has shortcomings, in particular the lack of modern AI principles to generate novel analysis in the face of novel situations. Generative AI has the potential to allow a Soar cognitive agent to consider a much broader range of contextual information, and learn from past episodic knowledge in novel ways by using transformer architectures. This paper focuses on the theoretical integration of Generative AI into the Soar cognitive architecture from a cybersecurity standpoint, and discuss the advantages to doing so.

1 Introduction

Argonne has been a leading member of an international consortium to develop autonomous intelligent cyberdefense agents (AICAs). This work grew out of a NATO Research Task Group (152) and continues in the form of an international working group based in France. Argonne built the initial prototype of this agent and open-sourced it for community use and contributions [1]. While this is currently more of a research environment than functional agent (no AI has yet been implemented in it), the Argonne team is using this as a launching point for several avenues of research. Given Argonne's focus on energy and national security issues, particular attention is being paid to the manner in which agents can protect critical infrastructures such as power, water, or communications; or other autonomous systems such as emergency service vehicles.

One avenue of this research concerns utilizing cognitive agents, such as Soar, to implement a flexible agent that ideally can learn and reason from its environment. Other approaches of interest include the use of graph neural networks (e.g., over knowledge graphs) and reinforcement learning (e.g., Deep Q). Over the past several months, the authors have been investigating the ability of Soar to integrate with a test environment similar to the open source AICA prototype, and develop initial input and output capabilities. During this time, generative AI has greatly increased in popularity and capability, and so naturally we have also begun to consider how those techniques factor into our ongoing investigations.

In this paper we will summarize the AICA architecture and its objectives, discuss our progress to-date on implementing an AICA-like agent using Soar, and make suggestions for ways we think Soar could leverage generative AI to meet the needs in a cybersecurity context. As this is an ongoing area of research for the authors, by the time of publication this work will likely have advanced beyond the state represented here.

2 The AICA Architecture

The AICA architecture (Kott et al. 2018a,b, 2019; Kott and Theron 2020) was created on the premise that future cyber attacks will be increasingly automated and even autonomously intelligent. This was before the advent of generative AI, but even in 2018 it was apparent that was a serious threat to be considered. The early work was done under NATO auspices, and specifically focused on the types of military systems that might come into a direct conflict. However, the architecture and ideas that went into it are highly generalizable. At Argonne, we are investigating ways to use this architecture and variations on it for protecting energy grids, first responder systems, and thinking about future directions such as space systems. Details of considerations for different operational environments for AICA-like agents are covered in detail in (Blakely et al. 2023c).

Figure 1, reproduced from (Blakely et al. 2023a), shows the main components of the AICA architecture, as implemented in Argonne's prototype. It consists of portions that ingest data from various inputs, reason over that data, constrain actions, and enact actions. More information about the development of this prototype is available in (Blakely 2022; Blakely et al. 2023b). As it currently is implemented, it is more of a sandbox for research into possible AI/ML approaches than an ML approach in itself, as we have not yet integrated anything beyond static rule-based reasoning into the decision-making engine. However, we have begun investigatory work into graph neural networks so that we

[1] https://github.com/aica-iwg/aica-agent/

can leverage node classification and link prediction to guide AICA's decision making process. The is enabled by AICA's use of a knowledge graph as it's primary information storage location.

Early demonstrations of this prototype, such as that presented in (Blakely et al. 2023a) have made it clear that interactions with human operators as paramount for the success of any autonomous cyberdefense agent. It's not enough to capture data, or even to appropriately act on the data. Agents must be able to present those decisions, or information required for operators to help them make those decisions in a manner that significantly reduces cognitive load of those operators. It's also clear that these agents must be able to constantly evolve, learn from past experience, and collaborate between themselves to accomplish their function. These are areas where an approach such as graph neural network classification or link prediction may struggle, and additional flexibility may be required. For that reason, while we continue to pursue that approach we are also investigating the use of cognitive agent-style agents in parallel. These have shown great promise, but may also suffer from some limitations that can be aided by the appropriate integration of generative AI components.

3 Soar-based Cyber Defense Agent

To investigate the potential use of cognitive models for AICA-like purposes, Argonne has begun building a project under the umbrella of a broader initiative called Descartes. Descartes is an initiative funded by Argonne through royalty funds, through the Department of Energy (DOE) Science Undergraduate Research Internship (SULI) program, and by the Department of Homeland Security Science and Technology (DHS S&T) Commercialization Accelerator Program (CAP). Additionally the initiative has a number of outstanding DHS and DOE funding proposals, and several fledgling industry partnerships. The existing funding streams are intended to accelerate development of autonomous cyberdefense agents and result in industry partnerships with field validation and deployments.

Moving from the AICA architecture to a Soar-based architecture requires a slightly different architecture, but fundamentally the data flow is the same. The agent(s) ingest data about their environment (or are pre-loaded with contextual information, such as semantic knowledge regarding network protocols & ports, software vulnerabilities and common weaknesses, etc), use the information to build an internal world state and reason about potential threats, and enact responses to protect their defended environment. In this way they are much like any other control system through their combination of sensors, decision making, and actuators and some concepts from areas such as PID controllers (proportional, integral, derivative) may be warranted as preprocessing steps for input data. This idea is discussed in additional detail in (Blakely 2021) using Netflow as a case study. However, the obvious difference is the sophistication and complexity of the decision making process, as well as the large variety of "senses" and "actions" the agent may need to consider. Additionally, agents are unlikely to perform fully autonomously - they need to collaborate with other agents and potentially human operators.

The authors have designed a Soar cyber agent architecture that is similar in many ways to its AICA counterpart. Like the Argonne AICA implementation, it is containerized within a single Docker container, and is designed to be as lightweight as possible for deployment to devices with fewer available resources, such as embedded systems on autonomous vehicles. Inside of the container, it runs a RabbitMQ server for communication between components of the Soar agent itself and to the outside world via an exposed communication port for accepting input data from other processes (e.g., in other containers) or other agents. Information from these inputs is pre-processed into standardized Python objects, and a "translate" was written to transform these into Soar "pseudo-productions". This "pseudo-production" concept is an extension of the fact that Soar provides a method of performing parallel conditional "for-each" style searches. It is theoretically possible that a Soar production can be described by a set of Soar working memory elements (WMEs) stored in an agent's semantic memory, and then executed by a set of interpreting "propose" and "apply" productions, letting Soar's hierarchical task decomposition gather the necessary information detailed in the vulnerability model sent to Soar. Pseudo-productions can be thought of as the missing pieces of an existing master set of productions.

Outputs from Soar are sent to a primary actuator system. This system translates them into OpenC2[2] framework-compatible output commands that can be relayed to individual actuators (or directly to external systems). This also allows integration with user interfaces for alerting or confirmation requests to interact with operators (a notable gap in AICA from (Blakely et al. 2023a)), as well as potential communication between agents. This also enables communication of threat information to a Trusted Automated Exchange of Intelligence Information (TAXII)[3] server via conversion to a Structured Threat Information Expression (STIX). As STIX is a node-relation ontology, it would potentially be an easy fit for converting to Soar WMEs. An additional input connection would likewise enable consuming threat intelligence from a TAXII server in STIX format. The current prototype uses a custom and loosely defined ontology instead of STIX, but consideration is being given to switching to STIX 2 if all concepts can be suitably expressed in it.

A notable benefit of utilizing the Soar agent is the ability to create a federated threat sharing system the offloads expensive machine learning tasks from edge devices to a central server and provides information Soar can ingest on potentially undiscovered vulnerabilities (zero days) and novel attack patterns. Such items are especially pernicious for cyber defenders as by their nature they are unforeseen and difficult to prevent. The usage of symbolic AI along with the diversity of inputs considered enables Soar to learn to, and thus potentially teach other agents, how to recognize these even if they haven't yet been communicated in the security community through formal notifications or threat sharing channels.

[2] https://openc2.org/

[3] https://oasis-open.github.io/cti-documentation/

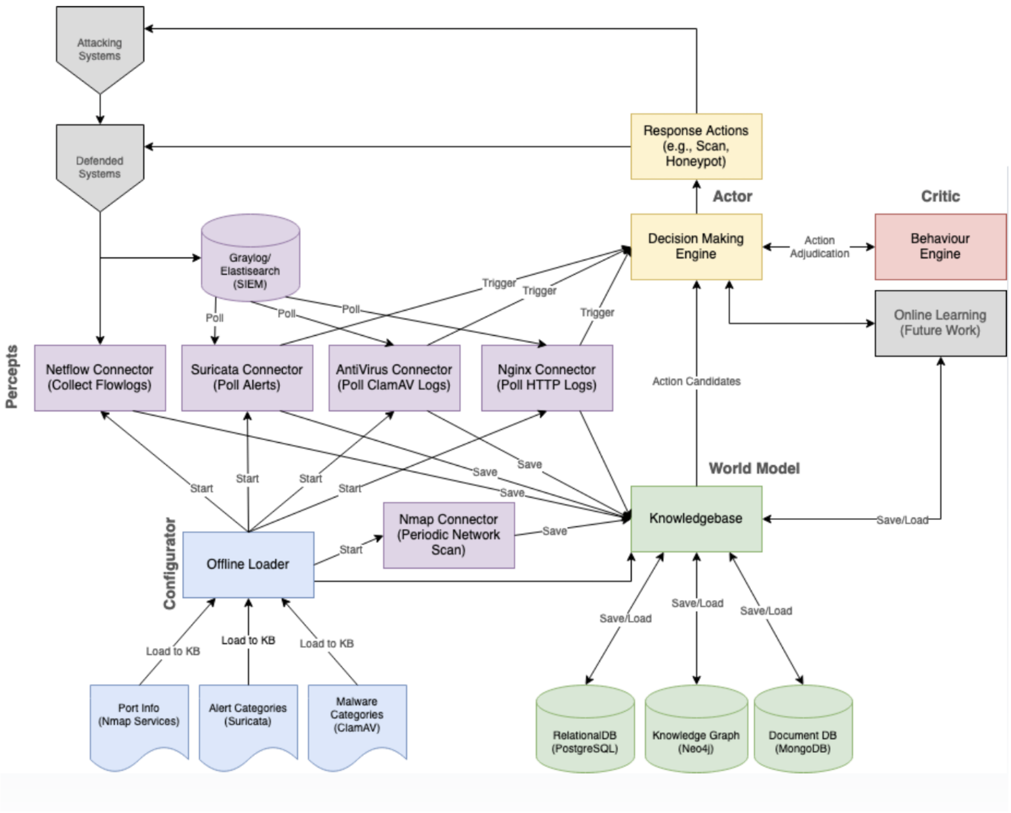
Attacking Systems
Defended Systems
Response Actions (e.g., Scan, Honeypot)
Actor
Critic
Decision Making Engine
Action Adjudication
Behaviour Engine
Graylog/ Elastisearch (SIEM)
Trigger
Trigger
Trigger
Poll
Poll
Poll
Online Learning (Future Work)
Percepts
Netflow Connector (Collect Flowlogs)
Suricata Connector (Poll Alerts)
AntiVirus Connector (Poll ClamAV Logs)
Nginx Connector (Poll HTTP Logs)
Action Candidates
Save
Save
Save
World Model
Start
Start
Start
Start
Nmap Connector (Periodic Network Scan)
Save
Knowledgebase
Save/Load
Configurator
Offline Loader
Start
Save/Load
Save/Load
Save/Load
Load to KB
Load to KB
Load to KB
Port Info (Nmap Services)
Alert Categories (Suricata)
Malware Categories (ClamAV)
RelationalDB (PostgreSQL)
Knowledge Graph (Neo4j)
Document DB (MongoDB)

Figure 1: AICA Architecture as implemented in prototype

The backbone of this threat-sharing is a dedicated system analyzing three sources. The first source is data from trusted cybersecurity publications, such as the the National Institute of Standards and Technology (NIST) National Vulnerability Database, particularly CWE (Common Weakness Enumeration) and CVE (Common Vulnerability & Exposure) information. These are semi-structured taxonomies with a defined ontology. Still, a considerable amount of the information in these is written in prose, so some level of natural language processing (NLP) would be needed to extract the full informative value.

The second source is feedback from individual Descartes agent instances, containing efficient representations of logs, audit information, and system resource usage values. This serves as the key component of zero-day detection, as an AI can be trained to recognize suspicious patterns within multiple reports of attacks, and pin down exactly what was exploited. This is akin to traditional threat intelligence sharing, but instead of sharing single indicators of compromise or signatures, the agent can share abstract representations of a likely-bad environmental state that can be incorporated into the semantic memory of other agents in the form of a pseudo-production or intelligence data. One can consider this to be something of a "post-mortem" report from a potentially hostile or damaging action and represents and entirely new form of threat intelligence sharing.

The third source is data coming from trusted TAXII servers, which use STIX to describe all manner of cybersecurity-relevant information, which can be used to complement information coming from the other two sources by providing explicit graph-like descriptions of cybersecurity intelligence.

To process the information mined from these data sources into a format ingestible by Soar, a centralized system with non-negligible computational resources would be needed. A trusted party such as US CISA or a national laboratory would be needed for this purpose (in the general public use case), but the advantage is that heavy processing for NLP or consolidating and abstracting post-mortem reports into a format that can be used by another agent. These use cases are similar to the federated learning models of machine learning in the latter case, and could potentially benefit from fine-tuned foundation models in the former case.

4 Limitations of Soar Approach

Originally, the project was envisioned to solely use a local Soar agent in order to accomplish its objectives. However, Soar has some inherent limitations in its design when faced with open ended-problems. Soar has to know patterns ahead of time, and its machine learning does not perform well on highly-dimensional input data. Additionally, it's difficult for the layman to interpret a Soar program (i.e., productions and operators) due to its highly implicit nature. This very different programming paradigm limits development to only specifically trained users. It should be noted, however, that the very nature of the Soar production syntax is what lends itself to the integration of Generative AI with the Soar Suite.

5 Opportunities for Generative AI Integration

The current envisioning of the threat-sharing system is one that provides a structure containing threat information, detection steps, and remediation steps to a soar agent on request. This structure can be injected into a Soar agent's semantic memory, which can be interpreted by the agent as a set of "pseudo-productions" that can be processed by a universal production to provide protection against the threat. Since this information now resides in the agent's semantic memory, it can also be recalled in the event of an active cyber-attack or during creation of a post-mortem report.

One of the main difficulties facing the development of the AICA offshoot is the implementation of the analysis system central to the threat-sharing model. This system needs to be capable of ingesting both AICA reports and cybersecurity publications, and outputting information usable by Soar. The core idea is to have a single "central AI" creating the information and behaviours for the smaller "edge" AIs.

A solution to this problem lies in the fact that the composition of a Soar production is an efficient representation of an arbitrary graph search and manipulation as a discrete time sequence, that is, a sequence of tokens forming graph manipulation commands and variable bindings, with a fixed vocabulary (excluding the variables, but these can be generalized into special tokens like VARIABLE 1, VARIABLE 2, etc...). This makes both soar productions and soar WME's prime targets for the output of a Transformer-based Generative AI approach to transform loosely-structured or unstructured data into a format Soar can ingest.

This leads us to our suggestion for a worthwhile addition to the Soar suite. Soar could incorporate two separate WME transformers. One would provide the ability to convert a natural language descriptor of a graph-like object into a valid soar semantic memory entry. The other would do the same sort of conversion, but instead with a process description into a set of Soar production memory elements (PMEs). By creating the basic natural-language to soar bridge, projects can build off of these AI for more specific usage scenarios. By bridging the gap between modern generate AI and symbolic AI, a system can be created with the advantages of both.

6 Conclusion

The authors have proposed a novel theoretical cyberdefense system architecture that utilizes the strengths of both symbolic and transformer AI, and taken first steps at implementing a prototype onto which this could be built. The authors envision the development of two separate transformer AIs for the Soar suite, these being an AI capable to transforming natural language descriptions into soar PMEs, and another for converting natural language descriptions into a soar semantic WMEs. Previous work by the authors and others on generating knowledge graphs from cybersecurity knowledge, combined with the obvious potential of generative AI, make it likely that this would be highly advantageous in constructing a functional implementation of the AICA architecture, as well as for othe problem domains where exchange of data between agents or consuming natural language data may be beneficial.

Acknowledgments

The work presented in this paper was partially supported by the U.S. Department of Energy, Office of Science under DOE contract number DE-AC02-06CH11357. The submitted manuscript has been created by UChicago Argonne, LLC, operator of Argonne National Laboratory. Argonne, a DOE Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.

References

Blakely, B. 2021. Cyber Senses: Modeling Network Situational Awareness after Biology. Washington, DC, USA.

Blakely, B. 2022. An Experimental Platform for Autonomous Intelligent Cyber-Defense Agents: Towards a collaborative community approach (WIPP). In *Resilience Week 2022*. National Harbor, Maryland, USA: IEEE.

Blakely, B.; Billings, H.; Evans, N.; Landry, A.; and Domingo, A. 2023a. Evaluation of an autonomous intelligent cyberdefense agent at NATO cyber coalition exercise 2022. In Wysocki, B. T.; Holt, J.; and Blowers, M., eds., *Disruptive Technologies in Information Sciences VII*, 13. Orlando, United States: SPIE. ISBN 978-1-5106-6200-1 978-1-5106-6201-8.

Blakely, B.; Horsthemke, W.; Evans, N.; and Harkness, D. 2023b. Case Study A: A Prototype Autonomous Intelligent Cyber-Defense Agent. In Kott, A., ed., *Autonomous Intelligent Cyber Defense Agent (AICA): A Comprehensive Guide*, Advances in Information Security, 395–408. Cham: Springer International Publishing. ISBN 978-3-031-29269-9.

Blakely, B.; Horsthemke, W.; Harkness, D.; and Evans, N. 2023c. Deployment and Operation. In Kott, A., ed., *Autonomous Intelligent Cyber Defense Agent (AICA): A Comprehensive Guide*, Advances in Information Security, 295–310. Cham: Springer International Publishing. ISBN 978-3-031-29269-9.

Kott, A.; Blakely, B.; Henshel, D.; Wehner, G.; Rowell, J.; Evans, N.; Muñoz-González, L.; Leslie, N.; French, D. W.; Woodard, D.; Krutilla, K.; Joyce, A.; Linkov, I.; Mas-Machuca, C.; Sztipanovits, J.; Harney, H.; Kergl, D.; Nejib, P.; Yakabovicz, E.; Noel, S.; Dudman, T.; Trepagnier, P.; Badesha, S.; and Møller, A. 2018a. Approaches to Enhancing Cyber Resilience: Report of the North Atlantic Treaty Organization (NATO) Workshop IST-153. Technical report, Army Research Laboratory, Munich, Germany.

Kott, A.; and Theron, P. 2020. Doers, Not Watchers: Intelligent Autonomous Agents Are a Path to Cyber Resilience. *IEEE Security & Privacy*, 18(3): 62–66.

Kott, A.; Thomas, R.; Drašar, M.; Kont, M.; Poylisher, A.; Blakely, B.; Theron, P.; Evans, N.; Leslie, N.; Singh, R.; Rigaki, M.; Yang, S. J.; LeBlanc, B.; Losiewicz, P.; Hourlier, S.; Blowers, M.; Harney, H.; Wehner, G.; Guarino, A.; Komárková, J.; and Rowell, J. 2018b. Toward Intelligent Autonomous Agents for Cyber Defense: Report of the 2017 Workshop by the North Atlantic Treaty Organization (NATO) Research Group IST-152-RTG. Technical report.

Kott, A.; Théron, P.; Drašar, M.; Dushku, E.; LeBlanc, B.; Losiewicz, P.; Guarino, A.; Mancini, L.; Panico, A.; Pihelgas, M.; and Rzadca, K. 2019. Autonomous Intelligent Cyber-defense Agent (AICA) Reference Architecture. Release 2.0.