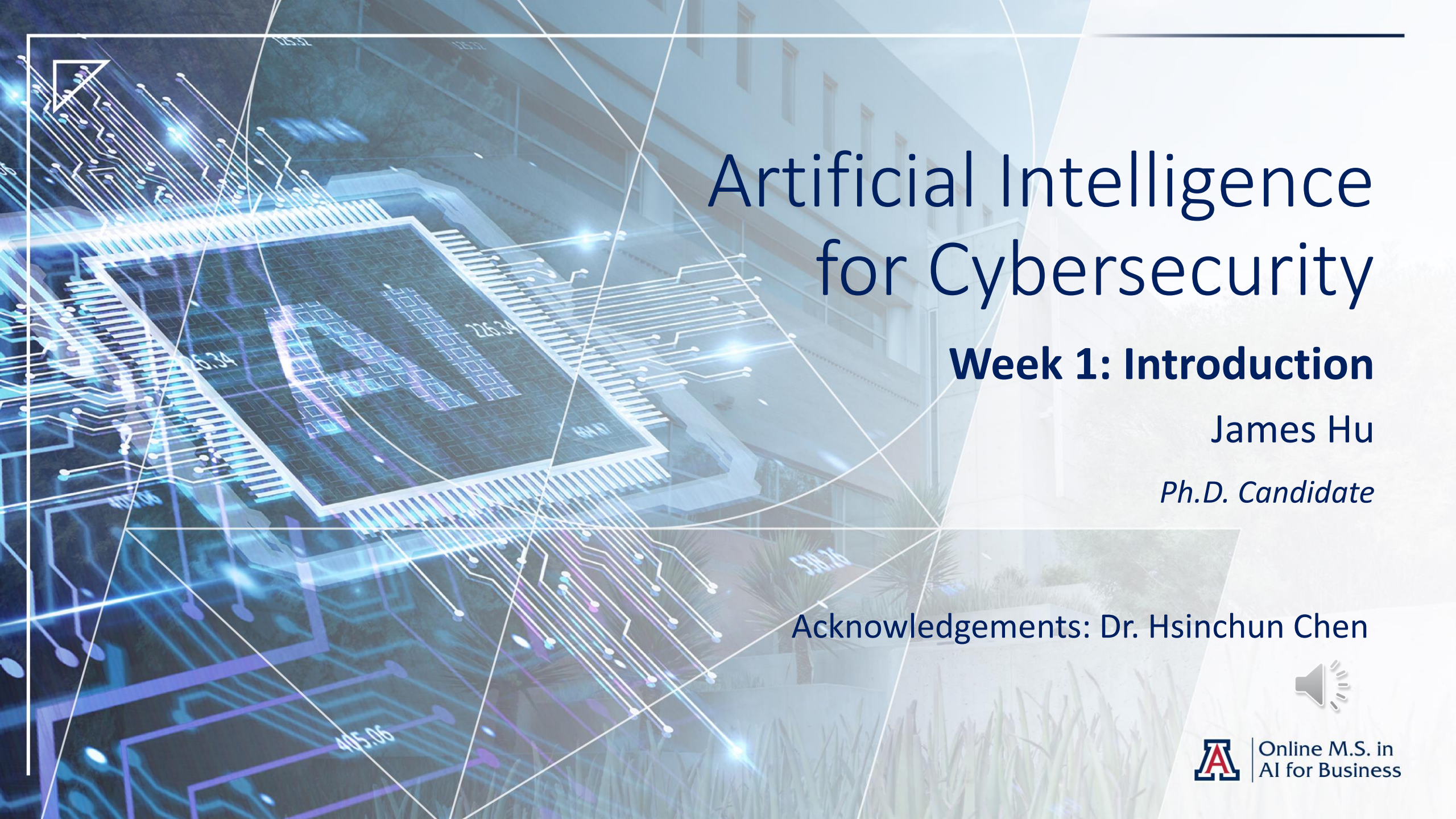




Artificial Intelligence for Cybersecurity

Week 1: Introduction

James Hu

Ph.D. Candidate

Acknowledgements: Dr. Hsinchun Chen



Online M.S. in
AI for Business

Course Agenda

- ▶ Week 1: Course Overview and the AI-for-Cyber (AI4Cyber) Lifecycle

[Overview](#)

- ▶ Week 2: Machine Learning Analytics
- ▶ Week 3: Deep Learning Analytics
- ▶ Week 4: Agentic AI Analytics

[Artificial Intelligence](#)

- ▶ Week 5: AI Analytics for Cybersecurity
- ▶ Week 6: Adversarial AI
- ▶ Week 7: Threat Analysis of AI Analytics

[Cybersecurity](#)



Online M.S. in AI

Grading Policy

- ▶ Exams

- ▶ Final Exam 100 pts (25%)

- ▶ Course Project

- ▶ Milestone 1 40 pts (10%)

- ▶ Milestone 2 100 pts (25%)

- ▶ Milestone 3 60 pts (15%)

- ▶ Final Report 200 pts (50%)





Final Exam

- ▶ Final exam will contain 3-4 multi-part questions
 - ▶ Comprehensive. Covers topics from all weeks.
 - ▶ Each question is essay-formatted.
 - ▶ Open book and open note
 - ▶ One week to complete the exam from when it opens



Course Project

- ▶ **Members:** 1-3 students per group
 - ▶ 2+ students per group is recommended
- ▶ **Task:** Develop actionable insights with AI analytics by implementing a complete AI4Cyber cycle.

Milestone	Task
Proposal	• Project team selection • Identify key stakeholders and task characterization
Milestone 1	• Review limitations of current approaches • Collect and pre-process data
Milestone 2	• Implement select AI analytics
Milestone 3	• Emulate adversaries and degrade AI analytics
Final Report	• IEEE-formatted Final Writeup





Today's Learning Objectives

- ▶ **Understand how cybersecurity** secures digital technologies and current challenges faced by cybersecurity professionals.
- ▶ **Explain how Artificial Intelligence (AI)** can enhance cybersecurity while also introducing certain cybersecurity risks.
- ▶ **Understand the AI4Cyber lifecycle**, including context identification, data extraction, AI-enabled analytics, and operationalization.
- ▶ **Recognize the iterative nature of AI4Cyber** and how each cycle informs the next with AI4Cyber insights disseminated via literature.



Motivation

- ▶ Society is becoming more digital:
 - ▶ Paper Records → Digital Databases
 - ▶ Traditional Appliances → Internet-of-Things
 - ▶ Local Software → Cloud-Based Application
 - ▶ On-Site Work → Remote Work
 - ▶ Legacy Infrastructure → Smart Cities
- ▶ These digital assets and systems are under constant threat by adversaries:
 - ▶ Unauthorized access to private data (e.g., Data Leaks).
 - ▶ Corruption of sensitive information (e.g., Stuxxnet)
 - ▶ Crippling of entire digital infrastructures (e.g., DDoS)



Cybersecurity

- ▶ Cybersecurity secures digital technologies from adversaries' cyberattacks.
 - ▶ Involves simultaneously fulfilling several functions and traditionally done manually:

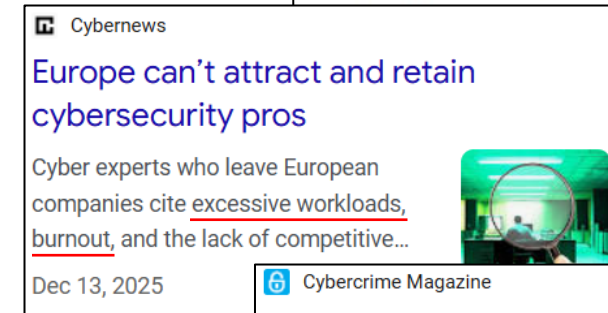
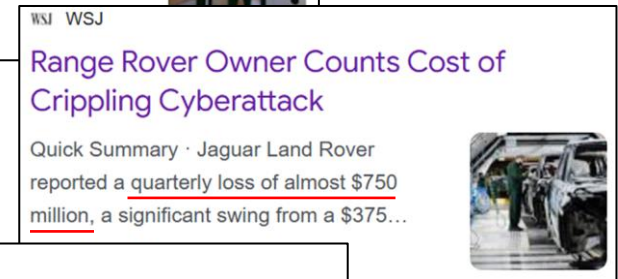
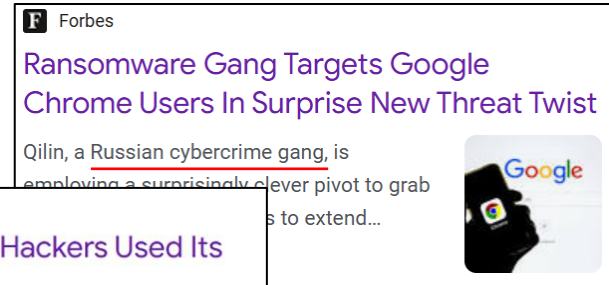
National Institution for Standards and Technology's Cybersecurity Framework 2.0 (NIST's CSF 2.0)

Function	Description	Example Task
Identify	Understand the system being secured	IT Manager cataloging all devices on a network
Protect	Proactive guarding against cyberattacks	System Admin implementing strict login requirements
Detect	Monitor for possible cyberattacks	Data Analyst flagging suspicious network traffic
Respond	Address ongoing cyberattacks	Incident Responder isolating a device infected with malware
Recover	Restore compromised systems	Data Engineer restoring systems from an offline backup
Govern	Enforce cybersecurity policies	Chief Information Security Officer setting a firm's general cyber-risk strategy



Cybersecurity

- ▶ With society increasingly digitalized, new challenges are emerging:
 - ▶ **Scale of Protection:** Many new digital assets need constant monitoring for potential attacks.
 - ▶ **Costs of Attacks:** One successful cyberattack can result in significant financial losses, consumer mistrust, and national security (sometimes all three).
 - ▶ **Lack of People:** There is a global shortage of cybersecurity professionals.
- ▶ It is no longer practical to just rely on the manual efforts of cybersecurity professionals.
 - ▶ Solution: Artificial Intelligence (AI)





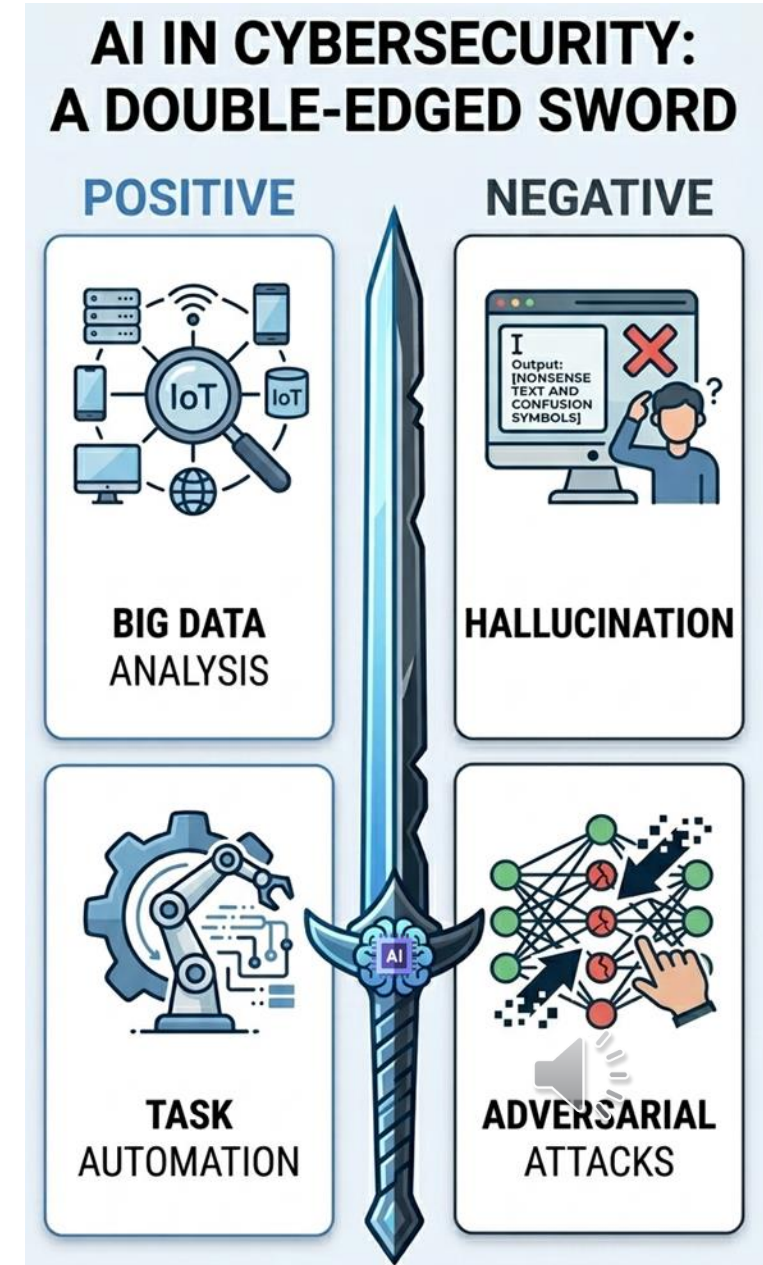
AI for Cybersecurity

- ▶ AI refers to computational techniques that allow machines to automatically make decisions by learning from data.
- ▶ AI can enable small teams to effectively defend large digital infrastructures by automatically:
 - ▶ **Conducting** routine cybersecurity tasks.
 - ▶ **Discovering** exploitable weak points that were otherwise missed by human eyes.
 - ▶ **Synthesizing** threat intelligence to proactively update defenses.
 - ▶ **Analyzing** streams of data in real-time to identify potential threats.
 - ▶ **Responding** immediately to cyberattacks to minimize damages.

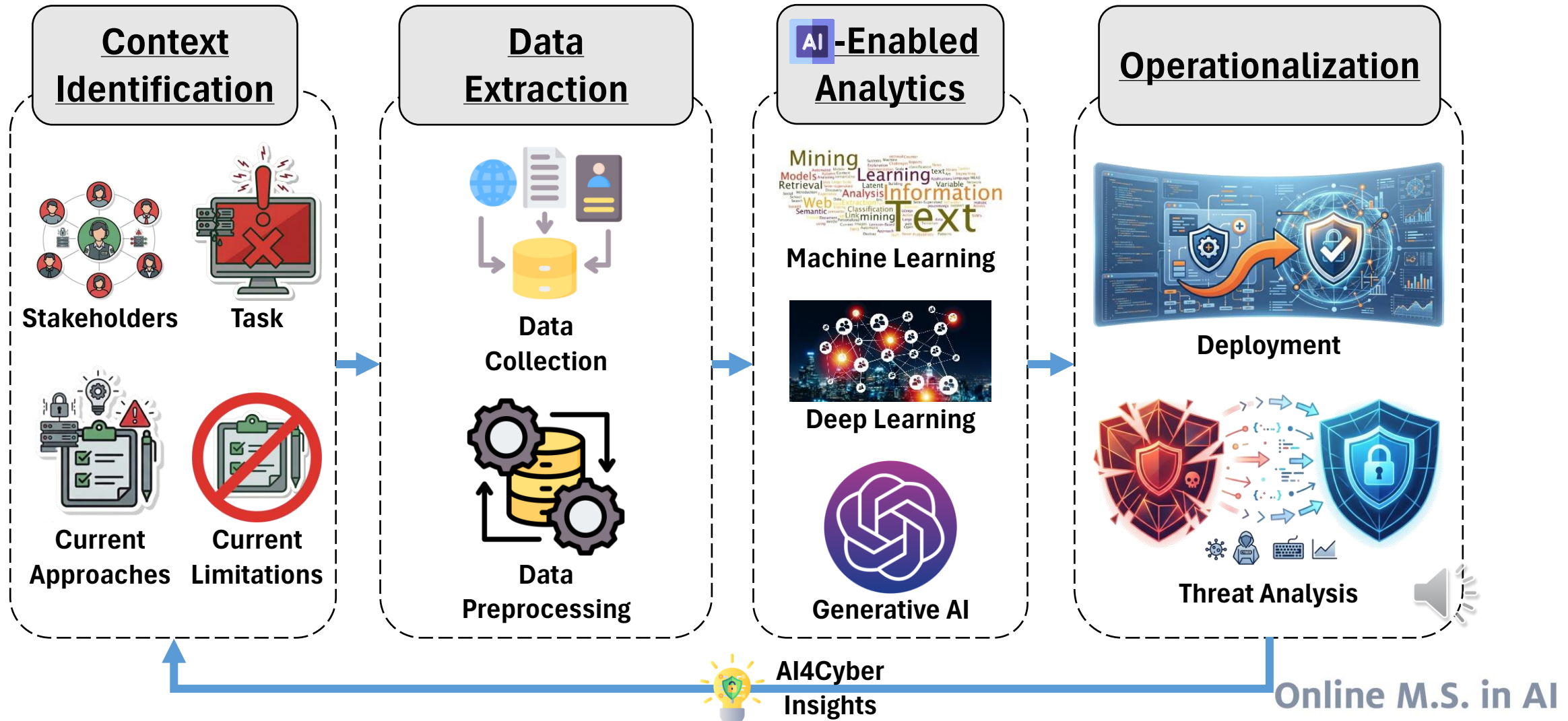


AI for Cybersecurity

- ▶ However, AI for cybersecurity is not a “fix-all” solution:
 - ▶ AI is not always right. They may learn incorrect patterns from the data and hallucinate.
 - ▶ Cybersecurity is an inherently adversarial environment. Adversaries will try to trick and exploit AI-based systems.
- ▶ Remember, **AI itself is also a digital technology!**
 - ▶ The security of AI systems must also be considered.
- ▶ Improper use of AI for cybersecurity can lead to unexpected and fatal security gaps.
 - ▶ Thus, AI must be leveraged for cybersecurity in a **systematic, efficient, and iterative** process.



AI4Cyber Lifecycle



Context Identification



Stakeholder

- **Primary User:** The administrators, managers, and analysts who will actively use the solution.
- **Critical Asset:** What is the digital technology being secured?



Task

- **Objective:** What is the goal that the stakeholders are trying to achieve?
- **Current Data:** What data is relevant and available? Where and how was the data sourced?



Current Approaches

- **Core Question:** What is the cybersecurity decision being made by the method?
- **Current Method:** Do they use a specific rule-based method, an analytic method, or do it manually?



Current Limitations

- **Data-Wise:** Is the scope or complexity of data problematic? How and why?
- **Method-Wise:** Is the current approach highly manual, inefficient, or ineffective? How and why?



Context Identification (Example): Phishing Detection



Stakeholder

- **Primary User:** Email Security Administrators.
- **Critical Asset:** Corporate email accounts and sensitive employee information.



Task: Phishing Detection

- **Objective:** Phishing emails are blocked. Suspicious emails properly labeled to alert employees.
- **Current Data:** The sender address and text body of prior normal/phishing emails.



Current Approaches

- **Core Question:** Should this email be delivered to the inbox, moved to junk, or stripped of its links?
- **Current Method:** Manual analysis of emails by IT team. Employee training.



Current Limitations

- **Data-Wise:** Dataset is missing crucial metadata (Header, embedded URLs, etc.).
- **Method-Wise:** Manual effort is not scalable. Employees are inconsistent with their evaluations.



Context Identification (Example): Malware Detection



Stakeholder

- **Primary User:** SOC Analysts and Endpoint Security Engineers.
- **Critical Asset:** Corporate workstations and critical servers.



Task: Malware Detection

- **Objective:** Malware files be detected and prevented from infecting digital assets.
- **Current Data:** Known malware signatures.



Current Approaches

- **Core Question:** Is this particular incoming file benign, suspicious, or malicious ?
- **Current Method:** Signature-based matching against database of known malware signatures.



Current Limitations

- **Data-Wise:** Dataset of known malware signatures is old and outdated.
- **Method-Wise:** High manual effort to analyze malware files without a known signature yet.





Data Extraction: Sources

- ▶ Our next step is to extract the necessary data from our identified context.
- ▶ We first explore and prioritize potential data sources to identify their:
 - ▶ Data type(s): Different types of data
 - ▶ Data volume: The total amount of data available from a source
 - ▶ Data frequency: How often new data is made available in the source.
- ▶ Understanding the nature of our data sources helps with selecting the appropriate data collection method.



Data Extraction: Sources

- There are five data types that are most common in for cybersecurity:

Data Type	Description	Examples
Numerical	Quantities that can be measured or counted	Packet payload size, CVSS base scores, number of failed login attempts, port numbers
Textual	Language data, composed of a sequence of words	Report descriptions, phishing email bodies, shell command strings, source code, assembly
Categorical	Groups or categories	Protocol types (TCP/UDP), threat severity (High/Med/Low), file types (.exe/.pdf), alert status
Datetime	Specific points in time, typically a calendar date combined with a time of day	SIEM event arrival time, timestamp of initial breach, certificate expiration dates.
Images	Pixelated images where individual pixels comprise of red, green, and blue values	Logos, biometrics, security film, email attachments.

- Each data source yields several datapoints. Each datapoint may contain several features with differing types.



Data Extraction: Sources (Examples)

Data Source	Individual datapoint	Features & Data Types	Data Volume & Frequency	Example Task
Wireshark Readings: Network Traffic Monitoring	Network Packets	• Source IP (text) • Destination IP (text) • Port / protocol (categorical) • Payload size (numerical) • Packet content (text)	Millions of packets daily or hourly	• DDoS attack detection • Unusual protocol flagging
Reddit: Social Media	Forum Posts	• Username (text) • Post text (text) • Timestamp (datetime) • Upvotes/likes (numerical)	1M+ a day, updated in real time	• Threat actor identification • Sentiment analysis of hacker communities
cve.org: Open Vulnerability Database	Vulnerability Reports	• CVE ID (text) • Upload time (datetime) • Report description (text) • Severity (categorical)	50-100+ a day, updated hourly	• Vulnerability discovery • System patch prioritization



Data Extraction: Collection

- ▶ There are four common methods for data collection.

Method	Description	Common Tools and Packages	Characteristics	Examples
Database Querying	Direct access to structured data from databases.	SQL clients, Python	Highly, structured, efficient for large volumes.	Internal company data, network logs
Web Scraping / Crawling	Programmatically extracts data from public websites by parsing HTML.	Python (bs4, Scrapy)	Accesses vast external web data, flexible.	Webpages, darknet websites
API Integration	Programmatic access to services via Application Programming Interfaces (API).	Python (requests), API-specific libraries	Structured & reliable, efficient, but sometime with usage limits	News articles, social media, network/sensor readings
Manual	Direct collection from individuals via forms or downloading from websites.	Custom web forms, spreadsheets.	Targeted info, captures qualitative data	Government reports, client feedback 

Data Extraction: Preprocessing

- ▶ A raw collected dataset can often contain too much irrelevant information.
- ▶ Data preprocessing isolates only the useful pieces of information and organizes them into a structured dataset for analytics. For example:
 - ▶ Filter only for network packets that originated from outside the company's network
 - ▶ Extract the hash signature from malware files
 - ▶ Identify key hackers that are mentioned often within a community
- ▶ **Common tools:** Python (pandas, spacy, nltk, Gensim), OCR tools, ETL platforms



Data Extraction (Example)

- ▶ **Task:** Identify key hackers within a community.
- ▶ **Data Extraction Steps:**
 - ▶ **Data Source:** Online community of hackers (r/hacking).
 - ▶ **Individual datapoint:** One post in r/hacking.
 - ▶ **Data Collection:** Web crawler saves the raw HTML of each post.
 - ▶ **Data Preprocessing:** Web parser systematically extracts important information (author, post date, etc.).
- ▶ Python code for this example is provided in supplemental material.
 - ▶ Leverages libraries for web scrapping (selenium) and html parsing (bs4).



AI-enabled Analytic



Descriptive analytic:

- **Purpose:** Summarize and interpret historical data to understand past events.
- **Question to answer:** What happened?
- Often supported by Unsupervised Learning



Predictive analytic:

- **Purpose:** Forecast future outcomes or probabilities based on historical data
- **Question to answer:** What will happen?
- Often supported by Supervised Learning

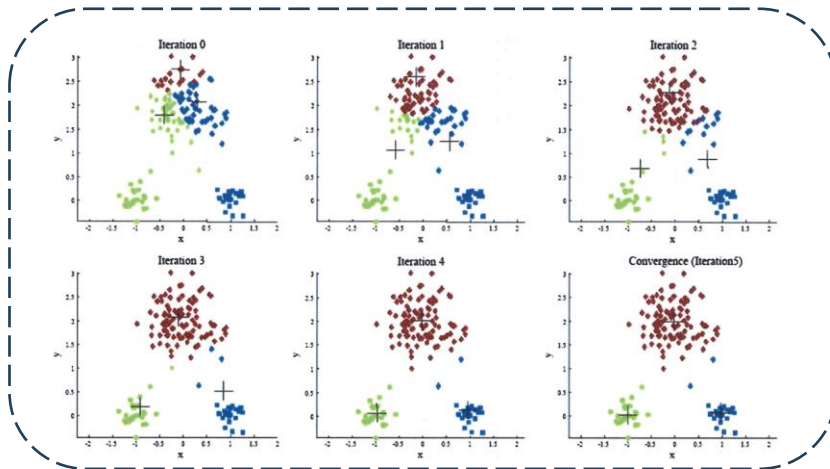


Prescriptive analytic:

- **Purpose:** Recommend specific actions to take to achieve the objective.
- **Question to answer:** What should I do?
- Often supported by Agentic AI



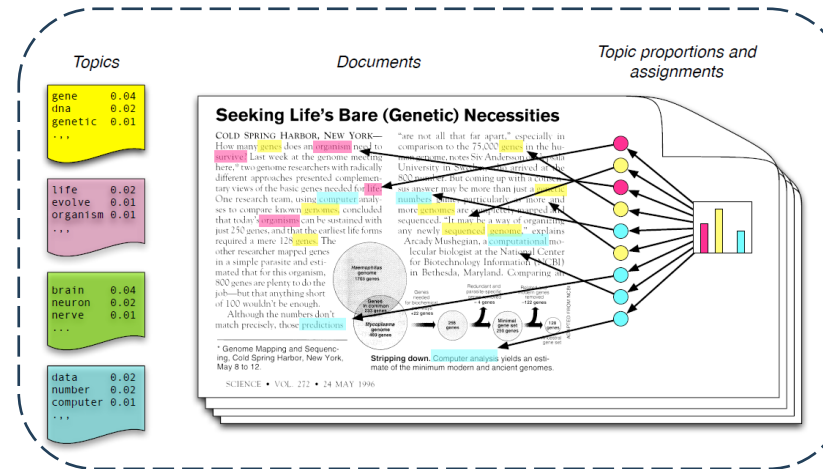
AI-enabled Analytic – Descriptive



Clustering

Examples

- **Network Device Segmentation:** Grouping devices in a network based on similar network behaviors.



Topic Modeling

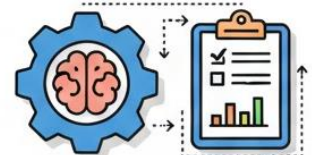
Examples

- **Dark Web Discovery:** Parsing posts from hacker forums or Reddit to identify emerging adversary trends.

1. Data



2. Training

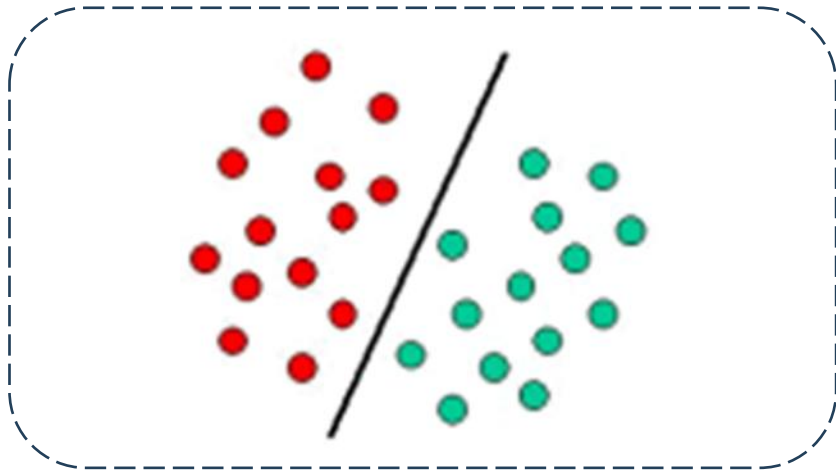


3. Insights



Online M.S. in AI

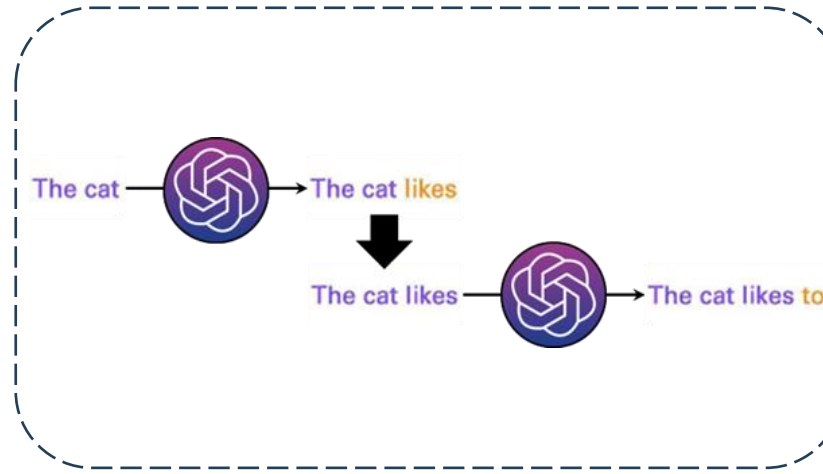
AI-enabled Analytic – Predictive



Classification

Examples

- **Phishing Detection:** Flagging emails that have a high probability of being phishing
- **Malware Detection:** Determine which files are suspicious or malicious



Generation

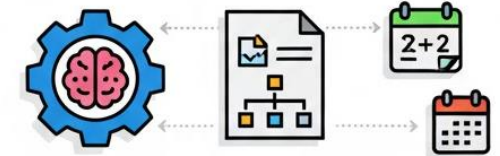
Examples

- **After Action Summarization:** Generate reports that summarize key points of a cybersecurity incident.

1. Data



2. Training & Testing

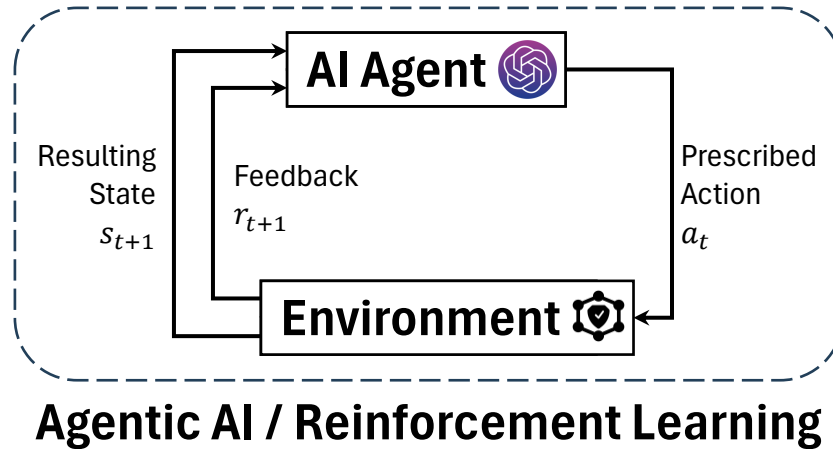


3. Prediction & Forecasting



Online M.S. in AI

AI-enabled Analytic – Prescriptive



Examples

- **Automated PenTesting:** Determine the optimal PenTesting actions to successfully compromise a client's IT system.

1. Simulation



2. Optimization

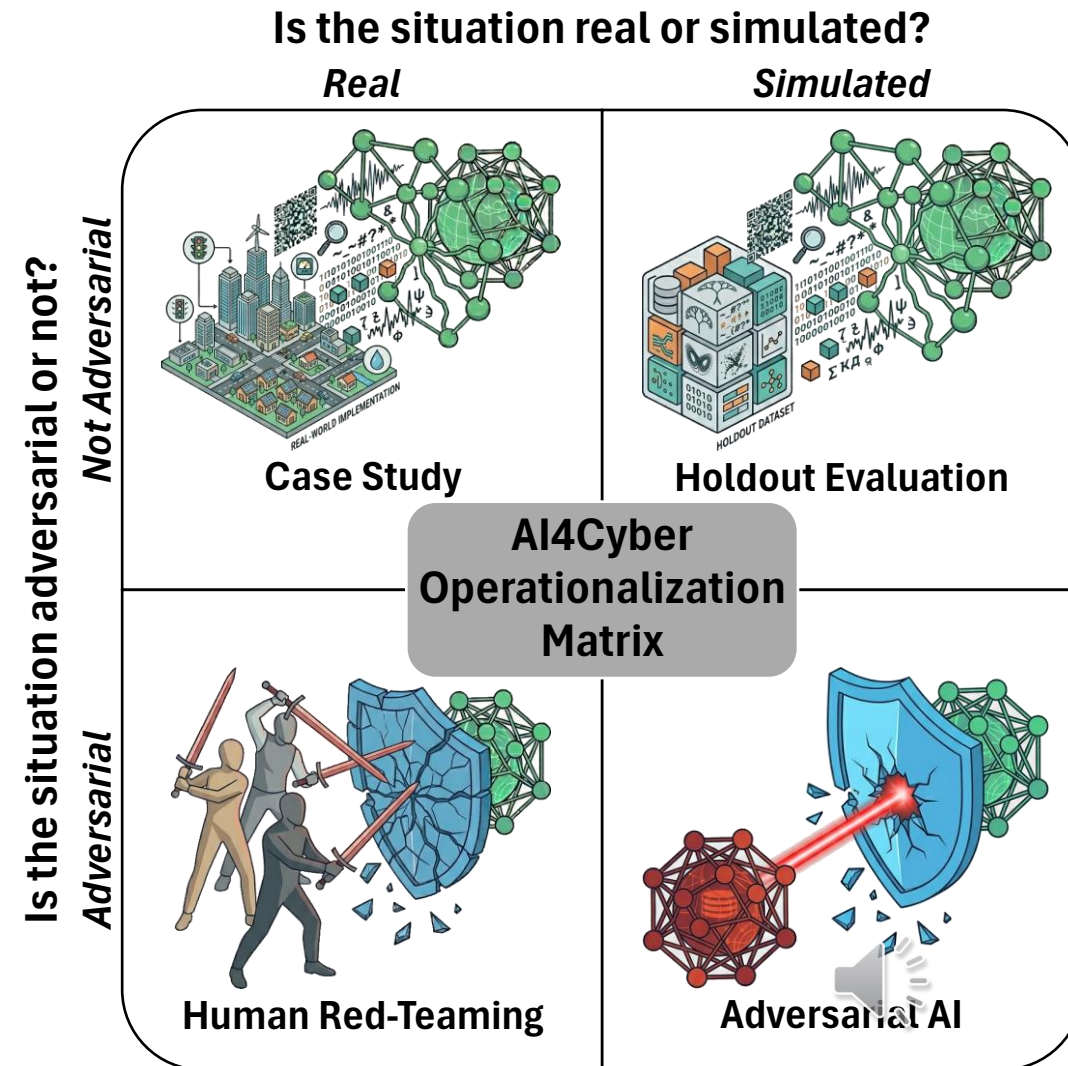


3. Recommended Action



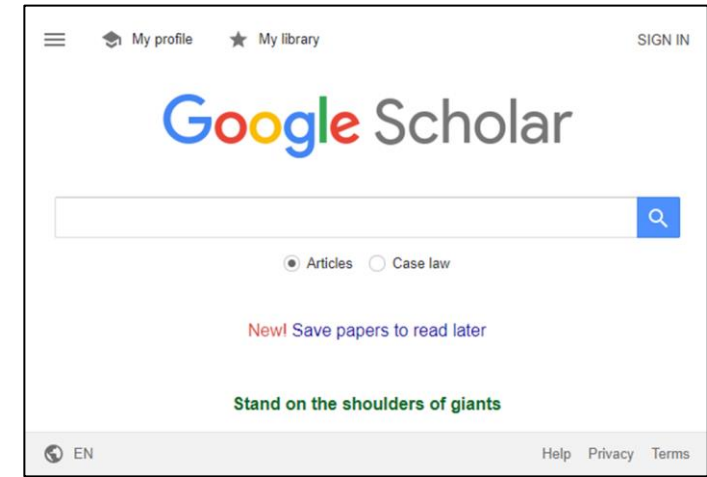
Operationalization

- ▶ After development, the effectiveness of AI-enabled analytics must be demonstrated via situated implementations:
 - ▶ **Deployment:** Implementing the AI-enabled analytic outside of the development environment. Examples include:
 - ▶ Real-world case study
 - ▶ Evaluate on holdout datasets
 - ▶ **Threat Analysis:** Identifying how adversaries can exploit the AI-enabled analytic. Examples include:
 - ▶ Manually with Human Red-Teaming
 - ▶ Automatically with Adversarial AI



Dissemination of Insights

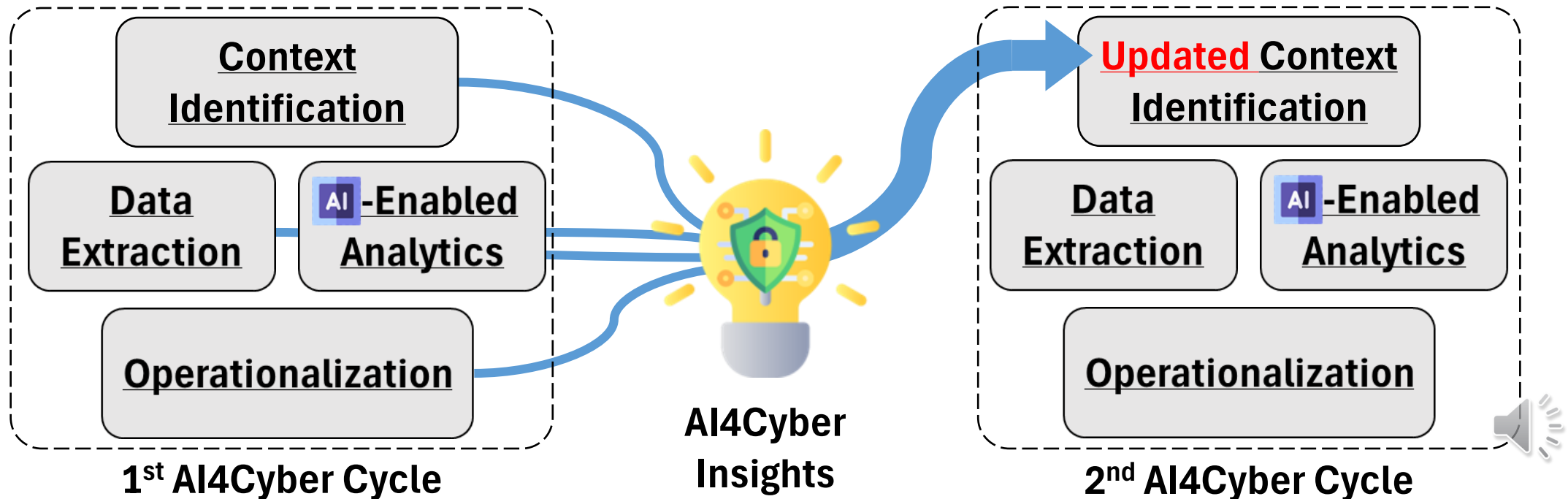
- ▶ One cycle of the AI4Cyber lifecycle should yield answers to the following questions:
 - ▶ What did you do? (**Data Extraction** and **AI-Enabled Analytics**)
 - ▶ Why did you do it? (**Context Identification**)
 - ▶ How well did it work? (**Operationalization**)
 - ▶ What is takeaway for future AI4Cyber analysts?
- ▶ These insights are compiled and disseminated via literature:
 - ▶ Academic Publications and White Papers (arXiv, IEEE, ACM, AAAI)
 - ▶ Opensource Code Repositories (Github, Huggingface)
 - ▶ Industry Reports and Memos (Company blogs)



Online M.S. in AI

Iterative Process

- ▶ Insights from one AI4Cyber cycle inform subsequent AI4Cyber cycles.
 - ▶ Likewise, when beginning an AI4Cyber cycle, reviewing related literature is crucial towards context identification.



Iterative Process (Example – 1st Cycle) : Malware Detection



Stakeholder

- **Primary User:** SOC Analysts and Endpoint Security Engineers
- **Critical Asset:** Corporate workstations and critical servers



Task: Malware Detection

- **Objective:** Classify incoming files as benign, suspicious, or malicious in real-time
- **Current Data:** Known malware signatures.



Current Approaches

- **Core Question:** Is this particular file benign or malicious?
- **Current Method:** Signature-based matching against database of known malware signatures

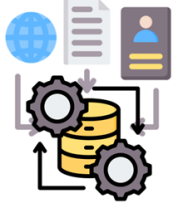


Current Limitations

- **Data-Wise:** Signature dataset is old and outdated
- **Method-Wise:** High manual effort analyze malware files without a known signature yet



Iterative Process (Example – 1st Cycle) : Malware Detection



Data Extraction

- **Data Collection:** Collect malware files from online database (VirusTotal) with API.
- **Data Preprocessing:** Extract each file's binary content



AI-Enabled Analytics

- **Description:** Three ML models analyzing a file's binary content to label as benign/malicious
- **Limitation Addressed:** Automates malware detection. Can try to predict labels of unknown files



Operationalization

- **Deployment:** Evaluation against holdout malware dataset from Microsoft.
- **Threat Analysis (Red-Teaming):** Machine Learning Security Evasion Competition at Defcon 2020.

DEFCON



AI4Cyber Insights

- **Summary of Insights:** Models all work well, but Defcon participants efficiently fooled all of them.
- **Disseminated Outputs:** Competition results published in company blog.

 **CUJO AI**



Online M.S. in AI

Machine Learning Security Evasion Competition

Getting Started

Visit the [GitHub project](#) for detailed information.

This contest involves functional malicious binaries. By participating, you agree to the [terms of service](#).

Dates

[Defender Challenge]: Jun 15 – Jul 23, 2020 AoE*

[Attacker Challenge]: Aug 06 – Sep 18, 2020 AoE

*all times are Anywhere on Earth (AoE)

Resources

Attacker Challenge - CLOSED

- download and setup (with disabled networking) a [free Windows 10 Virtual Machine](#) for use over 90 days to validate that your samples are functional
- [sample solution](#) to modify

Defender Challenge - CLOSED

- [sample solution](#) to modify
- [archive \(ZIP\)](#) containing malware and variants to tune your solution (encryption password: infected)

Prizes

Grand Prize (one per Challenge): US \$2,500 in Azure credits

First Prize (one per Challenge): US \$500 in Azure credits

Judging criteria are detailed in the [contest rules](#), but include:

- Defender Challenge**
 - highest true positive rate when deployed in the Attacker Challenge
 - the solution must also satisfy modest (<1%) FP requirements
- Attacker Challenge**
 - most black-box evasions against defensive countermeasures
 - fewest number of model queries breaks a tie

You must publish your solution to win.

Logged in as

Test Zoltan Balazs



Nickname: hapnoksq



ZOLTAN BALAZS • October 26, 2020 • 11 min read

Machine Learning Security Evasion Competition 2020 Results and Behind the Scenes

Analysis of the winning solutions

Please check out all the great write-ups from the participants.

First place in the attacker track and second at the defender track

<https://secret.inf.ufpr.br/2020/09/29/adversarial-malware-in-machine-learning-detectors-our-mlsec-2020-secrets/>

The previous one, but white-paper format, defender track only

<https://ieeexplore.ieee.org/document/8636415>

First place in the defender track, authors are Erwin Quiring, Lukas Pirch, Michael Reimsbach, Daniel Arp, Konrad Rieck. They are from the Technische Universitat Braunschweig, Germany.

<https://arxiv.org/pdf/2010.09569.pdf>

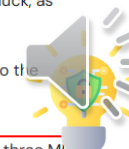
Second place in the attacker track

<https://embracethered.com/blog/posts/2020/microsoft-machine-learning-security-evasion-competition/>

When checking Fabricio and Marcus' solution, the high-level overview is that they first tried an XOR crypter. On top of that, they added a lot of dead imports to the import table – as recommended by us in the presentation multiple times 😊 I won't call this luck, as the second ML engine they bypassed was the one they submitted.

Reminder for the participants of the MLSEC 2021 challenge – if you submit to the defender challenge, you have better chances at the attacker challenge!

Anyway, they later changed the XOR based obfuscation to Base64, and voila, all three ML models were bypassed. I also expect that in the future some malware developers implement algorithms that decrease the entropy of the encrypted section or even encode it to something which looks like natural text.



Iterative Process (Example – 2nd Cycle) : Malware Detection



Stakeholder

- **Primary User:** SOC Analysts and Endpoint Security Engineers
- **Critical Asset:** Corporate workstations and critical servers



Task: Malware Detection

- **Objective:** Classify incoming files as benign, suspicious, or malicious in real-time
- **Current Data:** Known malware signatures. **Benign/malicious files and their binary data**



Current Approaches

- **Core Question:** Is this particular file benign or malicious?
- **Current Method:** **Three predictive ML models analyzing a file's binary content**



Current Limitations

- **Data-Wise:** **Dataset does not capture other features of malware files (e.g., source code).**
- **Method-Wise:** **Models have all been demonstrated to be vulnerable to base64 encryption.**



Review Questions

- ▶ In the AI4Cyber lifecycle, what are the key sub-components of the context identification component?
- ▶ Give one example each of descriptive, predictive, and prescriptive analytics.
- ▶ What AI4Cyber insights does each AI4Cyber cycle yield, and how does prior AI4Cyber insights influence future AI4Cyber cycles?



