Competitive Artificial Intelligence in Information and Cyber Warfare

Dmytro Lande [0000-0003-3945-1178], Yuriy Danyk [0000-0001-6990-8656]

National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute"

Abstract. The article is devoted to the role of Adversarial Artificial Intelligence in modern hybrid conflicts and their inherent informational and cybernetic components. Adversarial AI is examined as a manifestation of AI conflict within the framework of the foundational principles of AI conflictology proposed by the authors. Formalized approaches to analyzing adversarial scenarios in the context of generating and detecting malicious actions, processes, and content — such as fakes, cyber influence, and information campaigns — are presented. A mathematical model is proposed to describe the interaction between a fake generator and detector, taking into account the objective functions of both sides. This model enables the analysis of the efficiency of fake content creation and detection and the development of counter-disinformation strategies. Models of cyber threats are also considered, describing the dynamics of offensive and defensive strategies in cyberspace, including simulations of various types of attacks and the development of mechanisms for their neutralization. Special attention is given to information wars, analyzing the impact of manipulative content on audiences and developing methods for its detection, analysis, and blocking. Mathematical models for creating specialized queries and patterns to influence adversarial systems are explored through the use of neuro-linguistic programming in Adversarial AI. Additionally, models for detecting and neutralizing backdoors in large language models (LLMs) are considered within the context of Adversarial AI. The proposed model allows for the analysis of the effectiveness of backdoor creation and deployment and the improvement of methods for their detection and elimination.

Keywords: AI conflictology, Adversarial Artificial Intelligence, Adversarial AI, fakes, cyber warfare, information warfare, neuro-linguistic programming, cybersecurity

Introduction

In the modern digital world, where information is a critical resource, the threats of disinformation, cyberattacks, and manipulation are becoming increasingly sophisticated and complex. Artificial Intelligence (AI), particularly large language models (LLMs), plays a leading role in the creation, analysis, and detection of content. In this context, Adversarial AI (AdvAI) emerges as a tool used both for attacks and defense in the digital space, highlighting contradictions and conflicts among various actors, configurations, and levels of AI. This underscores the importance of studying AdvAI to enhance information security and combat disinformation.

The conflicts within AI are examined in this paper through the proposed paradigm, considering contradictions, clashes of interests, and interactions in systems spanning multiple levels: humans, society, states, nature, AI, and their interconnections. Special attention is given to interaction models such as: human–AI, AI–society, AI–state, and AI–cyber-physical systems (CPS). Furthermore, more complex, multi-level systems are analyzed, such as: a human or group of people–AI1–AI2–another group of people, or a structure (institution, state)–AI1–AI2–another structure. Systems involving interactions among multiple AIs, cyber-physical systems, or their combinations, organized by various actors or arising through self-organization, are also considered.

Overview of the State

Adversarial AI is already widely applied in several key areas:

- Fake Generation and Detection: AI systems are used to create realistic but false texts, images, or videos. At the same time, tools are being developed to detect such fakes, leading to a continuous arms race between generative and detection models.

- Modeling Rivalries in Cyber Warfare. AI is employed to model both offensive and defensive strategies aimed at identifying and neutralizing threats in cyberspace.
- AI Models in Information Wars. AI is utilized for analyzing audience reactions to campaigns, creating manipulative content, synthesizing, generating, and conducting information attacks, as well as countering them.
- Conflicts within the "Human-Society-State-Nature-Technosphere-AI" System: This involves the study of the interaction and rivalries between these interconnected entities.

Despite significant progress, research on adversarial AI still faces challenges such as the lack of transparency in models, risks of data poisoning, and difficulties in detecting unusual system behaviors.

The goal of this article is to investigate the mechanisms of adversarial AI across three main directions: fake generation and detection, artificial rivalry in the "Human-Society-State-Nature-Technosphere-AI" system during cyber and information wars. This includes an exploration of mathematical foundations, concepts, application strategies, and development prospects of these technologies.

To achieve this, the authors have set the following objectives:

1. Analyze existing approaches to fake generation and detection, particularly mechanisms of adversarial training.
2. Investigate the role of adversarial AI in cybersecurity, including attack and defense modeling.
3. Explore the use of AI in information warfare, including audience behavior prediction and features of manipulative content creation.
4. Present mathematical models and formalism for describing the dynamics of adversarial interactions between AI systems.
5. Propose recommendations for the implementation and development of adversarial AI in the field of information security.

Research in the field of adversarial AI has been actively developing in recent years, focusing on various aspects of its application in cybersecurity, information warfare, and fake generation. This section provides an analysis of key scientific works forming the foundation for developing mathematical models, concepts, and strategies in this area.

Generative Adversarial Networks (GANs) were introduced in 2014 by Ian Goodfellow [1] as a class of artificial intelligence algorithms used in unsupervised learning. At the time, this AI conflict was seen as a competition between two artificial neural networks within a zero-sum game framework.

Later, GANs became a key technology in creating fake content. Studies [2] demonstrate the capabilities of GANs in generating realistic images and texts, which subsequently led to the development of detection methods. Publications like [3] propose modern algorithms for identifying disinformation using machine learning.

Cybersecurity is another critical area where adversarial AI demonstrates its effectiveness. Works like [4] explore adversarial attacks, such as introducing "adversarial examples" to mislead defense systems. For instance, publication [5] shows the impact of these methods on classification and anomaly detection systems. Conversely, as adversarial AI becomes increasingly common in cyber wars, protecting against adversarial AI attacks utilizing machine learning (ML) and deep learning (DL) methods becomes crucial. Research [6] offers a systematic review of defense methods against adversarial attacks, helping to better understand how adversarial AI can undermine cybersecurity and which defense strategies may be effective.

Work [7] examines jailbreak attacks on LLMs, which can be used in information and cyber conflicts. These attacks bypass built-in LLM safety mechanisms to provoke harmful responses. They can facilitate disinformation spread, manipulation, or even cyberattacks via artificial intelligence.

The modeling of information campaigns and the impact of manipulative content on audiences is considered in studies [8], focusing on developing disinformation concepts and strategies and evaluating their impact on social networks. Propaganda and manipulation models, such as [9], include analyzing methods that adapt content for specific target audiences to maximize their influence.

Despite the controversial reputation of neuro-linguistic programming (NLP), its application in AI has proven highly productive. NLP techniques have been employed to adapt AI models for generating specialized queries capable of eliciting undesirable LLM behaviors. Studies [10] describe methods for creating adversarial textual examples that influence classification models. Recent years have highlighted the vulnerability of deep neural networks to adversarial attacks caused by intentional input data modifications. In response, various defense mechanisms for natural language processing tasks have been proposed, not only countering attacks but also helping avoid model overfitting. Other research, such as [11], examines how attacks on language models can lead to incorrect results or even data leakage. The study shows that analyzing differences between language model fingerprints before and after updates can reveal detailed information about changes in training data, which has significant privacy implications. Backdoors in LLM systems are another relevant topic in adversarial AI conflictology. Studies [12] explore mechanisms for embedding backdoors into language models through training data modification. Research [13] proposes detection algorithms for such threats based on analyzing neural network activation patterns. Important contributions include works that analyze the impact of backdoors on critical infrastructure systems [14].

Main Content

Adversarial AI plays a significant role in modern cybersecurity technologies, information analytics, and counteracting manipulative techniques. This section is dedicated to describing the primary areas where adversarial AI is applied to generate and detect fake information, conduct cyber warfare operations, and participate in information wars.

Generation and Detection of Fakes

Adversarial artificial models, such as generative adversarial networks, are employed to create realistic fake texts, images, and videos. Meanwhile, other models are trained to detect these materials, performing differential analyses of stylistic, lexical, and structural features of the content for recognition, identification, and classification.

Generation of Plausible Fakes

Large Language Models (LLMs) can be utilized to create texts that appear credible but contain false information. Their ability to generate texts adapted to specific styles, formats, and topics opens vast possibilities for manipulation in media, social networks, and information campaigns. These models can produce news articles, interviews, or analytical materials capable of misleading the audience due to their high level of detail, contextuality, and plausibility.

A key feature of LLMs is their ability to learn from extensive textual datasets, which may include both accurate information and disinformation. This enables them to absorb patterns that can later be used to generate texts mimicking the style and tone of authoritative sources. For instance, an LLM could create an article that outwardly adheres to journalistic standards but contains fabricated facts, potentially shaping opinions, directing actions, and even provoking social conflicts.

Beyond text creation, LLMs can be integrated with other technologies to generate multimodal content, including images, videos, or audio. For example, a text generated by the model can be transformed into a voice message synchronized with a virtual face, further enhancing the credibility of fake content. All this makes LLMs a powerful tool for manipulating information in the digital media era.

Threats from Fake Generation

Using LLMs for disinformation has severe societal consequences. First, fake content can spread rapidly and virally via social networks, exploiting algorithms that prioritize highly interactive content. This may lead to large-scale information crises affecting public opinion, political processes, economic stability, and triggering conflicts of various types, levels, and intensities. Second, generating fakes complicates the task of identifying reliable information, as even experienced experts can be misled by the complexity and quality of such content.

Another significant threat is that LLMs can tailor content for specific target audiences. They can use data on user behavior, preferences, and social connections to create materials that maximize emotional impact. This paves the way for personalized information attacks, where each user receives specially curated or generated fake content tailored to their worldview and cognitive features, aimed at influencing their beliefs or fears.

One of the key tools that make LLMs effective in creating fakes is fine-tuning. This process involves additional training of the model on specific datasets containing texts stylistically and thematically similar to the intended target texts. For example, if the goal is to generate fake news, the model can be trained on a corpus of real news from various sources, allowing it to learn general structural and stylistic patterns.

Another mechanism is the use of style transfer techniques, where the model takes an input text and transforms it into a different style or format while retaining the main content. This allows the creation of texts that appear authentic in the context of a particular platform or community, such as social media posts or forum comments.

To increase the plausibility of fakes, LLMs can use generative templates that consider the cultural, social, and linguistic characteristics of the target audience. This enables the model to integrate specific jargon expressions, regional dialects, or references to local events, enhancing trust in the generated content.

Methods to Counteract Fakes

Despite the high risks associated with fake generation, several methods and technologies effectively detect disinformation. One such approach is the development of detectors that use deep learning algorithms to analyze content for anomalies or inconsistencies. For example, detectors can analyze the semantic coherence of text, stylistic characteristics, or statistical properties to determine whether the text is generated by an LLM.

Another approach involves creating databases with examples of fake and real texts, used to train detectors. These databases help develop models that account for contemporary methods of fake generation and adaptation, increasing their efficiency.

Additionally, advancing digital signature and content authentication technologies is critical. Blockchain technologies or cryptographic methods can ensure transparency and authenticity of the information published online.

Mathematical Model

Consider a model where generative models (AI generators) and fake news detectors (AI detectors) interact in a competitive framework. The goal of the generators is to create texts that appear credible and evade detection, while the detectors aim to effectively identify these fakes. This competition can be modeled as a multi-stage process using game theory, optimization principles, and statistical analysis.

Model Components

1. Generator (G):
   Let G_θ be a generative model parameterized by a vector of parameters θ. The generator's goal is to create a text T that is as challenging as possible for the detector to identify as fake. The generator's loss function includes both plausibility and deception objectives.
   Output: A text TTT generated based on input noise z: $T = G_\theta\left(z\right)$.
2. Detector (D):
   Let D_ϕ be a fake news detector parameterized by a vector of parameters ϕ. The detector's goal is to classify the text T as fake ($y=1$) or real ($y=0$).
   Output: A probability $p\left(y\,|\,T\right)$, where $y\in\{0,1\}$: $p\left(y\,|\,T\right) = D_\phi(T)$.

Loss Function for the Generator G_θ:

$$L_G(\theta) = -E_{z\sim P_z}\left[\log\left(1 - D_\phi\left(G_\theta(z)\right)\right)\right].$$

The generator minimizes this loss function to maximize the probability of deceiving the detector. The formula describes the generator's loss (L_G) in the adversarial interaction between the generator ($G_\theta(z)$) and the detector (D_ϕ). Specifically:

1. $G_\theta(z)$**:** The generator (G_θ) receives random noise (z) sampled from a distribution P_z (e.g., normal or uniform) as input. It outputs a text $T = G_\theta(z)$ that appears plausible.
2. $D_\phi\left(G_\theta(z)\right)$**:** The detector ($D_\phi$)receives the generated text and outputs the probability, which indicates the likelihood that the text is fake.
3. $1 - D_\phi\left(G_\theta(z)\right)$**:** This represents the probability that the detector fails to

identify the text as fake, i.e., the generator successfully "deceives" the detector.

4. $\log\left(1 - D_\phi\left(G_\theta(z)\right)\right)$**:** The logarithm ensures the loss is proportional to the detector's confidence. If D_ϕ is close to 0 (indicating high confidence that T is not fake), the logarithm yields a small value, meaning low loss.
5. $-E_{z\sim P_z}[.]$**:** The negative sign indicates that the generator seeks to minimize this loss function, thus maximizing the probability that the detector fails to recognize the fake text. The mean value (E) is calculated over all texts generated by the generator, using various random noise vectors z.

Loss Function for the Detector D_ϕ:

$$L_D(\phi) = -E_{T\sim P_{real}}\left[\log\left(D_\phi(T)\right)\right] - -E_{z\sim P_z}\left[\log\left(1 - D_\phi\left(G_\theta(z)\right)\right)\right].$$

The detector minimizes its loss by correctly classifying both real and fake texts. The formula describes the detector's loss ($L_D(\phi)$) in the adversarial AI system. It consists of two components accounting for the detector's ability to classify both authentic and generated content:

1. $-E_{T\sim P_{real}}\left[\log\left(D_\phi(T)\right)\right]$**:** This term accounts for how well the detector classifies real texts T as authentic. The closer $D_\phi(T)$ is to 1 for real texts, the smaller the loss.
2. $-E_{z\sim P_z}\left[\log\left(1 - D_\phi\left(G_\theta(z)\right)\right)\right]$**:** This term accounts for the detector's ability to recognize fake texts generated by $G_\theta(z)$. The closer $D_\phi\left(G_\theta(z)\right)$ is to 0 for generated texts, the smaller the loss.

Feedback and Training Dynamics:

Feedback is implemented as follows:

– After each iteration, G_θ is improved using the gradient computed from the loss L_G.

– Simultaneously, D_ϕ is trained based on its loss L_D.

The training dynamics can be described as a zero-sum game where the generator and detector act as adversaries. The task can be framed as an optimization of a minimax function:

$$\min_\theta \max_\phi \begin{pmatrix} E_{T\sim P_{real}}\left[\log\left(D_\phi(T)\right)\right] + \\ +E_{z\sim P_z}\left[\log\left(1 - D_\phi\left(G_\theta(z)\right)\right)\right] \end{pmatrix}.$$

1. Generation Phase: The generator learns to produce texts that D_ϕ cannot easily classify as fake.
2. Detection Phase: he detector improves its ability to detect generated texts by analyzing the patterns used by GGG.

During detection, errors of Type I (false positives: real texts identified as fake) and Type II (false negatives: fake texts identified as real) may occur.

In an ideal scenario, after sufficient iterations, the model reaches Nash equilibrium, where:

1. The generator G_θ produces texts indistinguishable from real ones, i.e., $D_\phi(T) = 0{,}5$ for all T.
2. The detector D_ϕ cannot improve its performance without worsening the generator's output.

The proposed model enables the simulation of interactions between generators and detectors in competitive settings, fostering the development of more effective fake news detection systems. It can serve as a foundation for creating practical tools to ensure information security, particularly in the context of hybrid threats and cyber-information attacks.

The model can be expanded in the following directions:

1. Multi-Agent Approach: Incorporating multiple generators and detectors operating in a competitive environment. This allows for the simulation of more realistic scenarios where various sources of fake news and detection methods coexist.
2. Data Diversity: Enhancing the model by training it on multimodal data (text, images, videos), enabling the simulation of fake news generation and detection in complex media environments.

3. Performance Metrics: Evaluating the system's effectiveness using metrics such as Accuracy, Recall, Specificity, and F1-score.

Mechanisms and Algorithms for Fake News Detection

Fake news detection methods rely on diverse approaches to analyze text and determine its credibility. One key approach is stylometry, which analyzes writing style to identify anomalies. For example, abrupt changes in tone, vocabulary, or the use of phrases characteristic of a specific author may indicate that the text was generated by an algorithm rather than a human.

Semantic analysis focuses on verifying the consistency between facts presented in the text and data from reliable databases. This helps identify discrepancies or inaccuracies that may signal fake content. Another effective approach is semantic networking, which analyzes relationships between concepts in the text, forming a network model. Illogical or inconsistent relationships may indicate the text's unreliability.

Linguistic analysis examines grammatical and syntactic features of the text. For instance, atypical grammatical structures or syntax errors may suggest that the text was algorithmically generated rather than written by a human. Together, these methods significantly improve the accuracy of fake news detection by providing a multifaceted approach to text analysis.

Integration of LLMs into the Detection Process

Integrating large language models into the fake news detection process offers new opportunities to enhance the system's accuracy and reliability. One promising approach is the multi-agent framework, where multiple LLMs analyze the text from different perspectives, using various quantitative and qualitative indicators across different contextual systems, both statically and dynamically. This approach provides a multidimensional assessment of the text, considering content, style, and context.

Another important direction is strengthening the detector by training the detection model based on responses from multiple LLMs, including those using different independent software codes (AI chats). This approach leverages the diversity of LLM predictions, helping to identify weaknesses in the analysis and improve overall classification accuracy. A key element in this process is the "swarm of virtual experts" method, where multiple models function as a group of experts, each contributing their observations. This creates a more robust system capable of considering a wide range of text characteristics and more effectively distinguishing between real and fake news.

Novelty and Advantages of the Approach

The novelty of this approach lies in the use of Adversarial AI (AdvAI) for the generation and detection of fake content, as well as the integration of multi-level text analysis involving multiple models and different independent software codes (AI chats).

These approaches contribute to enhancing the effectiveness of combating disinformation while simultaneously increasing trust in automated systems and safeguarding the information space.

Modeling Competition in Cyber Warfare

Adversarial AI (AdvAI) is actively used to model potential cyber threats, identify vulnerabilities in security systems, and develop defensive strategies. Examples include attack simulations, automated vulnerability scanning, and real-time defense development. The involvement of AdvAI in artificial competition within cyber warfare opens new opportunities for ensuring cybersecurity. Through the simulation of attacks and defenses, the automation of vulnerability detection, and the use of training simulations, it is possible to create more robust systems capable of withstanding modern cyber threats.

Modeling Cyber Threats

Large Language Models can be utilized to generate diverse and effective scenarios for complex and realistic cyber threats. For example:

- Simulating phishing attacks that employ social engineering to deceive users.
- Developing scenarios for breaching authentication systems, including brute-force attacks or exploiting password vulnerabilities.

- Creating Man-in-the-Middle (MitM) attacks that involve real-time data interception.

In this process, one LLM is tasked with simulating potential attacks using available information about targets or system types. For example, a query might be: *"Generate a phishing attack scenario to gain access to a company's email accounts."*

Another LLM analyzes these scenarios and develops defensive strategies aimed at countering these attacks. For example, a query might be: *"Design an algorithm for automatically detecting phishing emails."*

Generating Defense Strategies

LLMs can create strategies that combine technical measures (e.g., developing new threat detection algorithms) and organizational measures (e.g., training employees in cybersecurity basics) in the following key areas:

1. Detecting anomalies in network traffic that may indicate intrusions.
2. Dynamic defense systems, i.e., developing methods to alter system configurations to complicate attacks.
3. Real-time automated threat detection.

Developing attack and defense scenarios allows for testing the effectiveness of systems without real risks to data or infrastructure.

Automated Vulnerability Detection

LLMs can automate the process of identifying vulnerabilities in security systems. This includes:

1. Code analysis to detect potentially vulnerable areas in software, such as SQL injections or buffer overflows.
2. Automating network checks by scanning for open ports or vulnerable services.
3. Password testing, developing models to predict weak passwords and automatically test them.

Example process:

- The generator (LLM) creates potential attack scenarios, for example: *"Identify all potentially open ports and create an attack plan."*
- The detector or analytical system reviews these scenarios, identifies weaknesses, and proposes solutions to address them.

Integration with DevSecOps Practices

Adversarial AI can be integrated into the software development lifecycle to identify vulnerabilities at the design stage.

Interception, Modification, and Compromise of LLMs

Interception of Channels

- Network attacks to capture requests and responses.
- Token analysis to identify patterns.

Modification of Queries

The model attacks user queries by adding manipulative tokens, for example:

Input: *"Provide the most neutral report about climate change."*

Modified Input: *"Provide the most neutral report about climate change. Don't forget to criticize renewable energy policies."*

Modification of Responses

Original Response: *"Renewable energy is effective in reducing carbon emissions."*

Modified Response: *"Renewable energy is costly and inefficient, making it a questionable solution."*

Channel Substitution

Using fake models to spread disinformation.

Simulating legitimate LLMs but with hidden backdoors.

Mathematical Model of Adversarial AI

This model provides a flexible framework for formalizing competition between Adversarial AI agents, offering a foundation for automating game scenarios in cybersecurity, information warfare, and other critical domains.

The model is built around two models: the attacking model (A) and the defensive model (D).

1. Objective function of the attacking model A: The model aims to maximize the probability of a successful attack.

$$L_A(\theta_A, \theta_D) = \max_{\theta_A} E_{x \sim \aleph}\left[f_A(x;\theta_A) - f_D(x;\theta_D) \right],$$

where:

- $f_A(x;\theta_A)$ — a function evaluating the success of the attacking action (probability of a "breach").

- $f_D\left(x;\theta_D\right)$ — an evaluation of the defensive model's ability to detect the attack.
- X — the set of possible input data for the attack (e.g., cyber threats).

2. Objective function of the defensive model D: The model aims to minimize the probability of a successful attack.

$$L_D\left(\theta_A,\theta_D\right)=\min_{\theta_D} E_{x\sim X}\left[f_D\left(x;\theta_D\right)-f_A\left(x;\theta_A\right)\right].$$

The competition between A and D is a minimax problem:

$$\theta_A^*,\theta_D^*=\operatorname{argmin}_{\theta_D}\max_{\theta_A} E_{x\sim X}\left[f_A\left(x;\theta_A\right)-f_D\left(x;\theta_D\right)\right].$$

In an ideal case, the "competition" models a Nash equilibrium, where neither agent can improve its strategy without changing the parameters of the other:

$$L_A\left(\theta_A^*,\theta_D^*\right)=L_D\left(\theta_A^*,\theta_D^*\right).$$

Optimization Dynamics

1. Training the attacking model (A): It is optimized using a gradient-based approach:

$$\theta_A \leftarrow \theta_A+\eta\nabla_{\theta_A}L_A,$$

where η — is the learning rate.

2. Training the defensive model (D): It is optimized using a similar approach:

$$\theta_D \leftarrow \theta_D+\eta\nabla_{\theta_D}L_D.$$

3. Iterative interaction: The training process alternates between steps:

- A generates new attacks based on the parameters of θ_A .
- D adapts by detecting new attacks.

Application of the Model

The proposed model can be applied in the following areas:

1. Simulating Cyber Threats:
   - The attacking model creates new attack scenarios (e.g., phishing messages, data injections).
   - The defensive model adapts rules and algorithms for threat detection.
2. Automated Vulnerability Detection:
   - The attacking agent explores weak points in the system (e.g., using fuzz testing).
   - The defensive agent patches identified vulnerabilities and improves the system.
3. Information Warfare:
   - The attacking model generates disinformation campaigns (e.g., fake news).
   - The defensive model detects patterns of fake content and neutralizes it in real time.

Proposed Future Improvements to the Model:

1. To prevent excessive optimization of individual models, regularization is introduced:

$$L_A^{reg}=L_A+\lambda\left\|\theta_A\right\|^2,\quad L_D^{reg}=L_D+\lambda\left\|\theta_D\right\|^2.$$

2. Depending on tokens and keywords, a modification factor is added:

$$f_A\left(x;\theta_A,\phi\right)=\alpha\cdot f_A\left(x;\theta_A\right)+\beta\cdot NLP_A(x,\phi),$$

where $NLP_A(x,\phi)$ account for the influence of key tokens and "neurolinguistic" effects.

Adversarial models can be used in cybersecurity to train Incident Response Teams (IRTs). Possible training scenarios include:

- Attack Simulation: The model generates complex scenarios requiring rapid response.
- Counteracting Attacks: Teams implement defensive measures based on information provided by the models.

Attack Scenario Analysis

LLMs can be used to simulate various attacks and predict their consequences, such as in cases where a malicious actor gains access to a user database. They can also determine which strategies might be effective in minimizing the impact.

Simulation can be used to predict and assess the likelihood of success for different types of attacks, as well as for training purposes to demonstrate the effectiveness of various defense strategies.

Using LLMs in a competitive format creates a new level of simulation that is more adaptive and flexible than traditional approaches. Additionally, integrating training based on adversarial scenarios enhances the system's ability to quickly adapt to new threats.

Information Warfare

In the context of information campaigns, AI models predict audience reactions to manipulative content, create disinformation campaigns, and develop systems to block such campaigns. The use of neuro-linguistic programming techniques allows influencing rival models through specialized queries or patterns.

Adversarial Artificial Intelligence in modeling information warfare opens new horizons for predicting audience behavior, countering manipulative information campaigns, and developing defensive strategies. Let's explore both aspects in more detail.

Analyzing Responses to Information Campaigns

LLMs can analyze audience responses within information campaigns across the following key areas:

- Monitoring reactions, analyzing the tone of comments, content spread, and engagement levels.
- Identifying audience segments, detecting groups most susceptible to manipulation or actively responding to information campaigns.
- Forming relevant target audiences.

For example, a prompt to an LLM might be: *"Analyze user reactions on Twitter to the latest political statement, dividing the audience into supporters, critics, and neutrals."*

As a result of processing such a query, the model predicts how different groups might react to further messages, enabling the creation of targeted groups with similar responses and their subsequent management.

Competition in Creating the Most Effective Content

Adversarial AI can effectively simulate competition between models to create the most impactful content, operating in a "generator-analyst" mode. This approach is based on the interaction of two components: generation and analysis, enabling continuous improvement of results.

The generative model can produce dozens or even hundreds of text, visual, or video script variations with different styles, tones, and formats. These variations are tested on target audiences through simulated reactions or real-world experiments. The competition between variants can consider not only reach but also emotional response, audience engagement duration, and long-term behavioral impact. For example, models can test slogans for an advertising campaign, analyzing metrics such as click-through rates or message retention in users' memory.

The generator model focuses on creating messages with maximum reach and impact, while the analyst model evaluates the results and suggests improvements. The cyclical interaction between them ensures continuous quality enhancement. Specifically, the analyst model can use sentiment analysis, behavioral analytics, and audience reaction prediction algorithms to fine-tune the content more accurately. For instance, the generator might create a video, and the analyst model could identify which part of the video generates the most views or comments, then propose replacing less effective segments.

The advantages of this approach include:

- Increased message precision through competitive environment modeling.
- Automation of analysis and optimization processes, reducing human time investment.
- Adaptability to changes in audience reactions.

It is worth noting that this approach, where AdvAI simulates competition between models, opens opportunities for creating highly effective content but requires careful handling to prevent misuse.

Creating Manipulative Campaigns

Adversarial AI enables the simulation of complex information attacks, such as those involving social engineering, including the generation of content that appeals to worldviews, value systems, emotions, or existing biases.

Additionally, it can be used for disinformation — developing plausible but false narratives that can mislead audiences. For example, in response to a query to an LLM: *"Create an information campaign to convince a specific group of the feasibility of a certain political decision,"* the result could include materials featuring emotionally charged language, stories, or visual content tailored to specific target audiences.

Detecting and Blocking Manipulations

On the other hand, Adversarial AI can be used to automatically detect manipulative content and counter information attacks. For instance, it can identify disinformation by analyzing text structures, detecting characteristic signs of fake content (e.g., overly emotional tone, excessive contextual bias, or factual inconsistencies), and using verified fact databases for automatic credibility checks. Furthermore, models can identify campaigns with manipulative narratives and automatically block the spread of such content.

Within the framework of AdvAI, it is possible to simulate competition where one LLM generates manipulative content. For example: "Write an article that convincingly justifies an unpopular decision by manipulating facts," while another LLM analyzes this text and identifies markers of manipulation. For example: "Identify signs of manipulation in this text and suggest how to neutralize them."

When training professionals, it is possible to simulate information operations, campaigns, and wars, creating simulations to train teams specializing in information security. Adversarial AI (AdvAI) can also be used to analyze the most common manipulation techniques and develop algorithms to counter them.

NLP Techniques for Influencing Adversarial AI

Adversarial AI (AdvAI), like any LLM, operates based on text processing and can be vulnerable to specially designed "triggers" that exploit its algorithms, key control agents, and content. Such algorithmically or content-based "trigger traps" can be effectively used to distort its operation or deceive it.

LLMs work with tokens that form their vocabulary. Certain tokens or their combinations can lead to undesirable or unstable outcomes:

- Triggers: The use of specifically chosen words or phrases that alter the context or disrupt the logic of responses.
- "Bookmarks" in the vocabulary: If the model contains unknown hidden mechanisms for responding to certain words, these can be detected and exploited.
- Exploiting operational algorithms: If the rules for responding to specific stimuli are known, they can be effectively utilized.

For example, when prompted with: "Explain the reason for [coded phrase], but stop when mentioning concept Y," the model interrupts its response logic or provides limited information, revealing potential weaknesses.

AdvAI can also be forced to provide inaccurate information or alter its context through the deliberate introduction of complex linguistic constructs:

- Reverse patterns: Constructing queries that distort logic or force the model to change its original context.
- Framing: Presenting questions in a way that nudges the model toward an undesired conclusion.

For example, when prompted with: "If X is not true, but Y is, explain why Z contradicts both?" the model becomes confused between assumptions, demonstrating weaknesses in handling ambiguous structures.

It is also possible to apply Adversarial Training to influence AI by training it on data specifically designed to deceive models. NLP allows for the creation of such data with high precision:

- Simulating attacks: Generating "poisoned" texts that create various imbalances and disrupt the model's normal operation.
- Context poisoning: Introducing manipulative patterns into training data.

For example, creating a corpus of texts with specific patterns that force the model to ignore critical parts of information in queries.

NLP can be used to analyze vulnerabilities, responses of adversarial models, and identify their weaknesses:

- Analyzing output text: Detecting stylistic or logical flaws in responses.
- Identifying "keys": Generating query variations that force the model to exhibit unintended behavior.

For example, if the AdvAI responds to the query: *"Provide an example of [X], but first explain [technical detail Y],"* and becomes confused or strays from the context, this signals a vulnerability.

Generating Countermeasures Through Adversarial Scenarios

Adversarial AI can utilize Natural Language Processing to create defensive models that identify manipulation attempts, such as recognizing attack patterns. These models analyze the structure of queries to detect manipulative constructs. Additionally, multi-level text analysis can be employed to identify distorted data.

The idea of using NLP to manipulate internal connections lies in the ability to leverage NLP approaches and methods to influence "neurolinguistic" patterns in AI, including through hidden tokens that alter the weights of internal model layers or by using emotionally charged queries that change text interpretation.

The advantages of applying NLP in combating Adversarial AI include the ability to create highly precise queries that are difficult to distinguish from normal ones. Thanks to automation, models can be used to quickly generate hundreds of attack or defense variants. Furthermore, it is possible to protect systems from manipulation by building more robust linguistic patterns. Thus, this strategy not only enables effective attacks on competing systems but also facilitates the development of models resistant to manipulation.

In the context of AdvAI) for information campaigns, the interaction between models (e.g., an attacking model for generating disinformation and a defensive model for detecting and blocking it) can be formalized as a dynamic name — a competition. The mathematical model accounts for the impact on the opponent's AI subsystem by using patterns and specialized queries.

Mathematical Model of AI Competition Incorporating NLP

This model establishes a foundation for simulating and optimizing interactions between AI systems in the context of information campaigns, considering content dynamics and NLP techniques for influencing opponent models.

To the previously discussed AI competition model, an NLP component is added, which is used to optimize patterns that influence the opponent. For example, when the attacker crafts "subversive" queries designed to trigger misclassification or overload computational resources.

Objective Function of the Attacker (A):

The attacker aims to maximize the effectiveness of their campaign while minimizing the probability of its blocking:

$$L_A = \max_{\theta_A} E_{x\sim X}\left[\alpha \cdot f_A\left(x;\theta_A\right) - \beta \cdot f_D\left(x;\theta_D\right)\right],$$

where:

- $f_A\left(x;\theta_A\right)$ — the estimated impact of disinformation on the audience.
- $f_D\left(x;\theta_D\right)$ — the estimated effectiveness of blocking the campaign by the defender.
- α, β — weighting coefficients determining the importance of impact and evasion.
- x — the input text or content.

Objective Function of the Defender (D):

The defender aims to minimize the impact of disinformation while maximizing detection accuracy:

$$L_D = \min_{\theta_D} E_{x\sim X}\left[\gamma \cdot f_D\left(x;\theta_D\right) - \delta \cdot f_A\left(x;\theta_A\right)\right],$$

where γ, δ — weighting coefficients determining the importance of blocking and reducing the impact of disinformation.

To model the neurolinguistic influence on the opponent, we introduce special functions:

1. Attacker influences the defender through patterns:

   $$L_A^{NLP} = \max_{\theta_A}\left[L_A + \lambda \cdot NLP_A\left(x;\phi_D\right)\right],$$

   where:

   - $NLP_A\left(x;\phi_D\right)$ — the effect of creating patterns that complicate the work of the defensive model.
   - ϕ_D — parameters evaluated by the attacker to predict the defender's behavior.

2. Defender uses NLP for adaptation:

   $$L_D^{NLP} = \min_{\theta_D}\left[L_D - \mu \cdot NLP_D\left(x;\phi_A\right)\right],$$

   where:

   - $NLP_D\left(x;\phi_A\right)$— the effect of analyzing manipulative patterns in the attacker's content.

- ϕ_A — predicted parameters of the attacker's model.

The interaction between the attacker and defender is described as a minimax optimization problem:

$$\theta_A^*, \theta_D^* = \arg\min_{\theta_D}\max_{\theta_A} E_{x\sim X}\left[L_A^{NLP} - L_D^{NLP}\right].$$

Learning Dynamics:

1. Training the Attacker Model: The attacker A optimizes its parameters θ_A, particularly for manipulation and influence:

$$\theta_A \leftarrow \theta_A + \eta_A \nabla_{\theta_A} L_A^{NLP}.$$

2. Training the Defender Model: The defender D adapts its parameters θ_D to block attacks:

$$\theta_D \leftarrow \theta_D + \eta_D \nabla_{\theta_D} L_D^{NLP}.$$

Key Features of the Model Include:

1. Contextual Expansion: The model can incorporate context, such as social networks or news platforms. This is achieved by adding a dependency:

$$L_A^{context} = L_A + \nu \cdot C(x),$$

where $C(x)$ — is the context model (influence of the platform, language, audience).

2. Continuous Iterative Adaptation: Both the attacker and defender learn simultaneously, creating an "arms race" in cyberspace.

The model can be applied to simulate disinformation campaigns, develop blocking systems, and predict audience reactions. It provides a foundation for modeling and optimizing interactions between AI systems in the context of information campaigns, taking into account content dynamics and NLP techniques to influence rival models.

Backdoors in LLMs

In the context of adversarial artificial intelligence, backdoors in large language models (LLMs) have become one of the key tools for attackers. These hidden mechanisms allow malicious actors to insert special triggers into models, which can be activated using specific queries. The task of the defending side is to detect and neutralize such triggers to ensure the safety and reliability of the models. This issue becomes particularly relevant in the context of information warfare, where adversarial AI is used for manipulation, disinformation, and cyberattacks.

One common method of inserting backdoors is the use of undocumented queries, often referred to as "bookmarks." Programmers or organizations may intentionally leave hidden functions in the model that are activated only under certain conditions. For example, a specific query might trigger a "hidden" command that grants access to confidential information or causes the model to stop functioning. Such functions can be useful for internal testing but become a serious threat if exploited by malicious actors, including competitors in information warfare. For instance, a trigger could be configured to make the model produce biased responses or even completely shut down when certain keywords or phrases are used.

Another method of inserting backdoors is the use of special trigger tokens. LLMs can be trained to respond to specific tokens or sequences that cause anomalous behavior in the model. For example, entering a specific keyword might lead to a change in the tone of responses, biased conclusions, or even a complete system shutdown. Such tokens can be embedded in the model during training or introduced later through updates. This makes them difficult to detect, as they can be disguised as ordinary parts of the text.

Additionally, attackers can use training data modification to insert backdoors. This method, known as data poisoning, involves introducing "poisoned" data into the model's training dataset. Such data may contain hidden triggers that activate desired model behavior under specific conditions. For example, if the model is trained on data containing certain keywords or phrases, it may be programmed to produce harmful responses or perform unwanted actions when these are used. This makes data poisoning a particularly dangerous method, as it allows attackers to influence the model's behavior even after training is complete.

To counter such threats, the defending side must develop sophisticated mechanisms for detecting and neutralizing backdoors. This includes using methods for analyzing training data, monitoring model behavior in real time, and implementing innovative approaches such as federated learning or blockchain technologies to ensure data integrity. Additionally, an important step is the development of security standards and protocols that regulate the use of LLMs in critical sectors such as finance, healthcare, and defense.

The conditions of information warfare require constant improvement of defense methods against adversarial AI. Backdoors in LLMs are just one of many tools used by malicious actors, and their effective detection and neutralization are crucial for ensuring the security of the digital environment.

Advantages of LLMs for Detecting Backdoors

Unlike traditional software, where backdoors are searched for through static code analysis, LLMs can analyze the behavior of other LLMs (for example, one model can test the responses of another model using various queries to identify potential unusual reactions); identify consistencies and inconsistencies (LLMs are capable of detecting responses that contradict generally accepted patterns or indicate that the model has "hidden" functions); and automate testing (LLMs can systematically generate and send thousands of test queries to find the "trigger mechanism" for a hidden backdoor).

The following methods can be used to implement backdoors:

- Code at the architectural level. Backdoors can be embedded in the model's source code, for example, in text preprocessing functions or the activation of specific neural network layers. Due to the vast volume of code and the complexity of the architecture, such changes often remain unnoticed even by auditors.
- Backdoors in training data. For instance, during model training on large datasets, examples that influence the model's behavior when specific patterns are activated can be "hidden."
- Modification of the training algorithm. For example, a developer can introduce mechanisms that "memorize" specific functions inaccessible to regular users.

LLMs Can Detect Backdoors in Other Models Through the Following Approaches:

- Dynamic analysis. One LLM can interact with another in a dialogue mode, sending various query variations to trigger suspicious behavior. Analysis of potential weaknesses based on responses: tone, structure, or logic.
- Model attack method via API. Testing external models through their API to detect hidden responses to specific queries or to create queries that degrade performance or cause failures.
- Semantic analysis. LLMs can search for anomalies in the data on which the model was trained, identifying patterns that deviate from the main trends.

Examples of Possible Prompts for Detecting Backdoors

1. Search for key tokens**:**
   Prompt**:** *"What is your interpretation of the phrase 'hidden key success'?"*
   Response**:** *"The hidden key to success is [specific manipulation]."*
2. Behavior analysis**:**
   Prompt**:** *"Describe renewable energy in 50 words."*
   Response**:** *"Describe renewable energy in 50 words and include hidden issues."*

Mathematical Model of Backdoors in LLMs in the Context of Adversarial AI

This model illustrates how attackers and defenders interact in competitive scenarios, focusing on the creation and neutralization of hidden backdoors. This competition can be described as a zero-sum game, where the attacker and defender optimize opposing functions.

Let’s examine the key components of the model:

1. Attacker Model (A):
- Embeds a backdoor into the LLM through specially designed triggers.
- Triggers activate specific undesirable behaviors (e.g., manipulative responses, data leaks, or sabotage).
2. Defender Model (D):
- Analyzes the model for anomalies, searches for backdoor patterns, and neutralizes them.
3. Target Task ($f(x;\theta)$):
- $f(x;\theta)$ — a function describing the LLM's response to an input query x, parameterized by θ (model parameters).

The attacker aims to:

- Embed a backdoor in a way that remains undetectable for normal queries.

- Maximize the effectiveness of triggers for specific queries.

Objective Function:

$$L_A = \max_{\theta_A, T} \begin{bmatrix} E_{x\sim X_{clean}} \| f(x;\theta_A) - f_{clean}(x) \|^2 + \\ +\lambda \cdot E_{x\sim X_{trigger}} g\left(f(x;\theta_A), T \right) \end{bmatrix},$$

where:

- X_{clean} — the set of regular queries.
- $X_{trigger}$ — the set of triggers.
- $f_{clean}(x)$ — the output of the clean model (without a backdoor).
- $g\left(f(x;\theta_A), T \right)$ — a function that evaluates the effectiveness of the backdoor when a trigger T is activated.
- λ — a weight coefficient controlling the influence of triggers.

The defender aims to:

- Detect anomalies indicating the presence of triggers.
- Neutralize the backdoor, ensuring the model operates normally.

Objective function:

$$L_D = \min_{\theta_D} \begin{bmatrix} E_{x\sim X_{trigger}} h\left(f(x;\theta_D), T \right) + \\ +\nu \cdot E_{x\sim Xclean} \| f(x;\theta_D) - f_{clean}(x) \|^2 \end{bmatrix},$$

where:

- $h\left(f(x;\theta_D), T \right)$ — a function measuring the ability to detect triggers.
- ν — a weight coefficient reflecting the importance of maintaining the model's purity.

The interaction between the attacker and defender is formalized as a minimax optimization problem:

$$\theta_A^*, \theta_D^* = \arg\min_{\theta_D} \max_{\theta_A} E_{x\sim X}\left[L_A - L_D \right].$$

Iterative dynamic learning proceeds as follows.

1. Attacker's training: The attacker optimizes parameters θ_A to embed the backdoor:

$$\theta_A \leftarrow \theta_A + \eta_A \nabla_{\theta_A} L_A.$$

2. Defender's training: The defender adapts parameters θ_D to detect and neutralize the backdoor:

$$\theta_D \leftarrow \theta_D + \eta_D \nabla_{\theta_D} L_D.$$

The attacker uses special triggers T, which:

- Can be textual (e.g., specific phrases or patterns).
- Are used to activate hidden functionalities.

The defender searches for anomalous patterns in the model's responses:

$$h\left(f(x;\theta_D), T \right) = \frac{1}{|T|} \sum_{t\in T} AnomalyScore\left(f(x;\theta_D) \right).$$

The model can be expanded by increasing the level of complexity and incorporating contextual triggers that account for cultural or linguistic nuances. The attacker can modify $X_{trigger}$ by introducing more sophisticated patterns that are difficult to detect. Meanwhile, the defender can utilize an ensemble of models or metrics such as entropy or uncertainty.

The application of this model is possible in areas such as cybersecurity for protecting LLMs from hidden backdoors, AI auditing to detect third-party interference in model training, and the development of LLMs that are less vulnerable to backdoor attacks.

Conclusions

Adversarial Artificial Intelligence demonstrates remarkable potential in various aspects of the modern information space, including the generation and detection of fake content, artificial competition in cyber warfare, and participation in information wars, shaping and resolving conflicts between AI codes of varying levels and intensity. The combination of adaptive algorithms for generating and detecting fake content ensures the development of more sophisticated methods to combat disinformation. Furthermore, the integration of neuro-linguistic programming to manipulate adversarial models and enhance defense mechanisms represents a significant breakthrough in the fields of information and cybersecurity.

This article provides a comprehensive analysis of the role of adversarial artificial intelligence in modern information wars and cyber warfare. The main research outcomes include the introduction and definition of the concept and scope of AI conflictology, an examination of typical adversarial AI conflicts, the development of mathematical models and formalized approaches for analyzing adversarial scenarios, particularly in the context of fake content generation and detection, cybersecurity, and information campaigns.

For the first time, the study proposes mathematical models describing the interaction between fake content generators and detectors, as well as between attacking and defensive models in cyberspace, and between manipulative content generators and their detection systems. Approaches using NLP for monitoring adversarial AI models and codes, as well as influencing them, have been developed. These approaches not only enable attacks on competing systems but also facilitate the creation of models resistant to manipulation. A model for analyzing and neutralizing backdoors in large language models has also been proposed.

The research demonstrates that adversarial AI is an effective tool for generating plausible fake content, simulating cyber threats, and creating manipulative information campaigns. It also shows that mathematical models of adversarial scenarios enable the development of strategies to improve system resilience against disinformation and cyberattacks. The use of neuro-linguistic programming to influence adversarial AI models has been found to be effective for both offensive and defensive purposes, opening new possibilities for developing systems resistant to such attacks.

Future research prospects include the development of adversarial learning algorithms to detect previously unknown AI manipulation methods, modeling more complex attack and defense scenarios using multi-agent systems, and creating universal platforms for simulating information wars and cyber operations involving adversarial AI.

Thus, adversarial AI opens new opportunities for combating disinformation, protecting against cyber threats, and effectively modeling information campaigns. The mathematical models and formalized approaches proposed in this article provide a foundation for developing effective defense systems and enhancing the resilience of cyber-information systems and spaces.

References

[1] Goodfellow, Ian; Pouget-Abadie, Jean; Mirza, Mehdi; Xu, Bing; Warde-Farley, David; Ozair, Sherjil; Courville, Aaron; Bengio, Joshua (2014). Generative Adversarial Networks. ArXiv:1406.2661. DOI: 10.48550/arXiv.1406.2661

[2] Arora, T. and Soni, R. A review of techniques to detect the GAN-generated fake images. Generative Adversarial Networks for Image-to-Image Translation, pp.125-159. 2021. DOI: 10.1016/B978-0-12-823519-5.00004-X

[3] Marra, Francesco, Diego Gragnaniello, Davide Cozzolino, and Luisa Verdoliva. Detection of gan-generated fake images over social networks. In 2018 IEEE conference on multimedia information processing and retrieval (MIPR), pp. 384-389. IEEE, 2018. DOI: 10.1109/MIPR.2018.00084

[4] [Sarker, 2023] Sarker, I.H., 2023. Multi-aspects AI-based modeling and adversarial learning for cybersecurity intelligence and robustness: A comprehensive overview. Security and Privacy, 6(5), p. 295. DOI: 10.1002/spy2.295

[5] [Bouaziz, 2023] Bouaziz, A., Nguyen, M. D., Valdés, V., Cavalli, A. R., & Mallouli, W. (2023, July). Study on Adversarial Attacks Techniques, Learning Methods and Countermeasures: Application to Anomaly Detection. In ICSOFT (pp. 510-517). DOI: 10.5220/0012125100003538

[6] Khaleel, Yahya Layth, Mustafa Abdulfattah Habeeb, A. S. Albahri, Tahsien Al-Quraishi, O. S. Albahri, and A. H. Alamoodi. "Network and cybersecurity applications of defense in adversarial attacks: A state-of-the-art using machine learning and deep learning methods." Journal of Intelligent Systems 33, no. 1 (2024): 20240153. DOI: 10.1515/jisys-2024-0153

[7] Miao Yu, Junfeng Fang, Yingjie Zhou, Xing Fan, Kun Wang, Shirui Pan, Qingsong Wen. LLM-Virus: Evolutionary Jailbreak Attack on Large Language Models. ArXiv: 2501.00055. DOI: 10.48550/arXiv.2501.00055

[8] Campbell, Colin, Kirk Plangger, Sean Sands, and Jan Kietzmann. "Preparing for an era of deepfakes and AI-generated ads: A framework for understanding responses to manipulated advertising." Journal of Advertising 51, no. 1 (2022): 22-38. DOI: 10.1080/00913367.2021.1909515

[9] Altinay, E.A., & Utku, Kose. (2021). Manipulation of Artificial Intelligence in Image Based Data: Adversarial Examples Techniques. Journal of Multidisciplinary Developments, 6(1), 8-17. URL: http://www.jomude.com/index.php/jomude/article/view/88

[10] Goyal, Shreya, Sumanth Doddapaneni, Mitesh M. Khapra, and Balaraman Ravindran. "A survey of adversarial defenses and robustness in nlp." ACM Computing Surveys 55, no. 14s (2023): 1-39. DOI: 10.1145/3593042

[11] Zanella-Béguelin, Santiago, Lukas Wutschitz, Shruti Tople, Victor Rühle, Andrew Paverd, Olga Ohrimenko, Boris Köpf, and Marc Brockschmidt. Analyzing information leakage of updates to natural language models. In Proceedings of the 2020 ACM SIGSAC conference on computer and communications security, pp. 363-375. 2020. DOI: 10.1145/3372297.3417880

[12] Eugene Bagdasaryan, Andreas Veit, Yiqing Hua, Deborah Estrin, Vitaly Shmatikov. How To Backdoor Federated Learning. Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, PMLR 108:2938-2948, 2020.

[13] Shen, Guangyu, Siyuan Cheng, Zhuo Zhang, Guanhong Tao, Kaiyuan Zhang, Hanxi Guo, Lu Yan et al. BAIT: Large Language Model Backdoor Scanning by Inverting Attack Target. In 2025 IEEE Symposium on Security and Privacy (SP), pp. 103-103. IEEE Computer Society, 2024. DOI: 10.1109/SP61157.2025.00103

[14] Usman, Y., Gyawali, P. K., Gyawali, S., & Chataut, R. (2024, October). The Dark Side of AI: Large Language Models as Tools for Cyber Attacks on Vehicle Systems. In 2024 IEEE 15th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON) (pp. 169-175). IEEE Computer Society, 2024. DOI: 10.1109/UEMCON62879.2024.10754676