# ATTACKS, DEFENSES, AND TOOLS: A FRAMEWORK TO FACILITATE ROBUST AI/ML SYSTEMS

**Mohamad Fazelnia, Igor Khokhlov & Mehdi Mirakhorli**
Rochester Institute of Technology
Rochester, NY 14623, USA
{mf8754,ixk8996,mxmvse}@rit.edu

## ABSTRACT

Software systems are more increasingly relying on Artificial Intelligence (AI) and Machine Learning (ML) components. The emerging popularity of AI techniques in various application domains attracts malicious actors and adversaries. Therefore, the developers of AI-enabled software systems need to take into account various novel cyber-attacks and vulnerabilities that these systems may be susceptible to. This paper presents a framework to characterize attacks and weaknesses associated with AI-enabled systems and provide mitigation techniques and defense strategies. This framework aims to support software designers to take proactive measures in developing AI-enabled software, understanding the attack surface of such systems and develop products that are resilient to various emerging attacks associated with ML. The developed framework covers a broad spectrum of attacks, mitigation techniques, and defensive and offensive tools. In this paper, we demonstrate the framework architecture and its major components, describe their attributes, and discuss the long-term goals of this research.

## 1 INTRODUCTION

Today's software systems are more increasingly relying on various artificial intelligence (AI) and machine learning (ML) components. For instance, flight control software of commercial airlines rely on AI-enabled autopilot components, medical diagnostic software products rely on AI for image analysis, autonomous driving vehicles heavily rely on object detection and recognition, and modern intrusion detection systems rely on various ML-based classification models and anomaly detection. This increase in AI and ML (AI/ML) deployment in software products can be explained by various reasons, such as availability of the data needed for AI/ML models training, more powerful computers capable of much faster model training, drastic advances in the AI/ML algorithms, etc.

Followed by this popularity of AI/ML-based systems, attack and defense strategies on AI/ML systems have been widely studied (Szegedy et al., 2014; Goodfellow et al., 2015; Moosavi-Dezfooli et al., 2016; Papernot et al., 2016). Also, several works have done surveys on this domain (Liu et al., 2018; Xu et al., 2020; Pitropakis et al., 2019; Yuan et al., 2019) and tried to categorize attacks and defenses based on their attributes and assumptions. However, we still lack a comprehensive analysis of the techniques and tactics used in this domain.

During software design and development, designers have to understand all potential threats related to the AI/ML algorithms deployment and leverage mitigation techniques in order to address them. While there have been extensive resources for weaknesses (MITRE, 2021) and attacks (ATT&CK, 2021) on software systems, there are not many organized knowledge bases for attacks, weaknesses, and mitigation techniques for AI-enabled systems.

To this end, we present an online framework[1] that aims to enumerate possible attacks to the AI-enabled systems, identifies appropriate mitigation techniques, tools and toolchains used by adversaries, and characterizes attackers. This framework provides a structured and scalable comprehensive knowledge base that can help researchers, practitioners, and AI capability builders be better

[1] www.design.se.rit.edu/programs/ai-ml-framework

informed of attacks, toolchains, and mitigation techniques and guide them to perform attack modeling of their AI-enabled software systems and reason about adversaries, identify them, and respond intelligently.

While this paper presents the framework's architecture, the long-term goal is to develop a framework that enables cyber-risk analysis of AI-enabled Software systems. To the best of our knowledge, this is the first work that provides a comprehensive framework based on a systematic literature review that categorizes adversarial actors and explains their behavior, techniques and tactics used by them, their appropriate mitigation techniques, and the offensive and defensive tools used in AI/ML systems. The proposed project is a fundamental step toward developing a knowledge base for various offensive and defensive measurements, representations, and other mechanisms for securing AI-enabled software systems.

## 2 METHODOLOGY

This work aims to provide a comprehensive framework that contains all the attacks, mitigation techniques, and tools in AI/ML domain. To this end, we conducted a systematic literature review of articles published in the two of the most popular digital libraries, IEEE Xplore [2] and ACM Digital Library. [3] We explored all peer-reviewed papers published between 2000 and 2020 years. The first step includes searching in papers' abstract using such keywords (and their variations) as "artificial intelligence," "machine learning," "deep learning," "neural networks," "mitigation," "threat," and "adversarial."

As the initial result, we gathered around 14500 papers in total. Then, two experts in the area filtered out irrelevant papers based on their titles, which resulted in 2500 filtered papers. The next step included paper filtration based on their abstracts. Next, we performed a detailed literature review. During reviewing each paper, we also employed a snowballing process to review other related works mentioned in the literature that have not been published in the two aforementioned digital libraries. After we analyzed the first random 10% of the selected papers, we were able to develop a primary skeleton of the proposed framework. Based on this review, we identified the attributes that could characterize different attacks, as well as the techniques and tactics they use to violate the systems.

In order to build the proposed framework, we perform the following steps:

1. Conduct a systematic literature review to gather information representing attacks on AI/ML algorithms, mitigation strategies, and tools related to attacks or mitigation techniques.
2. Analyze and identify emerging attacks on AI/ML algorithms.
3. Classify attacks, tools, and mitigation strategies from various perspectives and develop appropriate components of the framework. This step resulted in a set of attributes for each component of the framework, which are presented in Section 3.
4. Develop a bi-directional mapping between all components and implement the developed framework in a form that can be used by security practitioners and system developers.
5. Finally, the framework has to be validated by gathering feedback from system designers that used our framework, which is out of this paper's scope.

This paper briefly discusses the first two steps and focuses on the third step, and briefly touches on the fourth step. The fifth step will be presented in future work.

## 3 FRAMEWORK TO DESCRIBE AI/ML ATT&CK

Based on the conducted systematic literature review and the analysis of the selected papers, we developed the framework with three major components: *attacks*, *mitigation techniques*, and *tools*, and a set of attributes for each component. Each component provides a comprehensive review of its area. The framework's mindmap is presented in figure 1.

[2] www.ieeexplore.ieee.org

[3] www.dl.acm.org

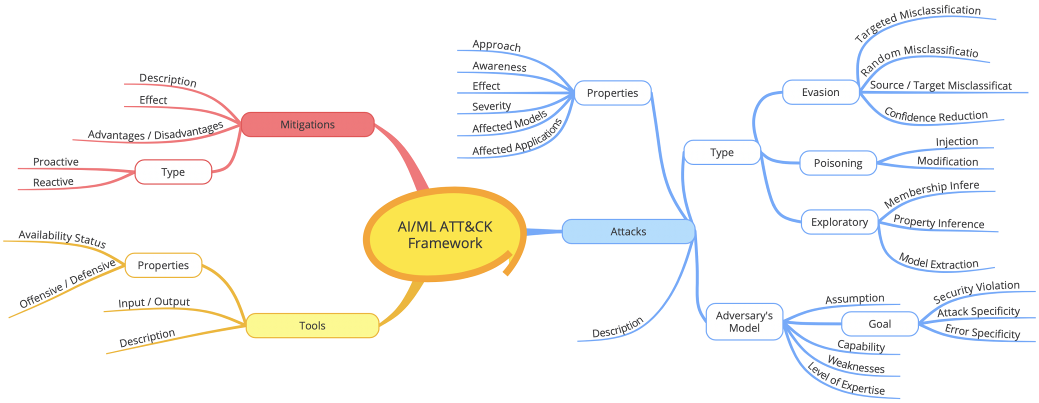

Figure 1: An overview of the meta model

### 3.1 Attacks

This section expands the *attacks* component and describes its characteristics briefly, as well as its relationship to the *mitigation* and *tools* components.

#### 3.1.1 Types

Attacks on AI/ML-enabled systems are categorized into three types, *Poisoning attacks, Exploratory attacks, and Evasion attacks*. In **Poisoning Attacks** the attacker tries to undermine the learning process during model training via two approaches, *injection* and *modification*, in which the attacker injects malicious samples into the training process (Muñoz González et al., 2017) and the attacker directly changes the training data (Biggio et al., 2011), respectively. **Exploratory Attacks** aim to extract private information from the system, which can be categorized to the following groups: *Membership inference*, in which the attacker aims to determine whether a particular data exists in the training dataset (Shokri et al., 2017). In *property inference* the attacker tries to extract information about the model and the training data, like data distribution, or the model hyper-parameters. Finally, in *model extraction*, the attacker tries steal the ML model and architecture (Tramèr et al., 2016). **Evasion Attacks** refer to those attacks that try to create a malicious sample that the model classifies mistakenly. The attacker could reduce the confidence level of the predicted output which is called (*confidence reduction*), or based on the input and target class, the attacks are categorized to *misclassification*, *target misclassification* and *source-target misclassification* (Papernot et al., 2016).

#### 3.1.2 Attributes

To better characterize the attacks on AI/ML systems, understanding the attack's attributes is extremely important. To this end, we considered the following attributes for attacks: **Approach** explains the mechanism of the attack. **Effect** shows the consequences of the attack.**Awareness** illustrates the level of information that is available to the attacker.*White-box* (Carlini & Wagner, 2017) denotes full access to the model's architecture, data and the parameters, while in a *black-box* setting, neither data nor the model is available to the attacker. The *gray-box* setting refers to the attacks with limited knowledge about the target model and its data (Xiang et al., 2021). **Severity** indicates the level of the violation. **Mitigation** presents appropriate mitigation strategies and maps the attack to the related part of the "mitigation" component of the framework. **Affected Models** shows the models that are vulnerable to the attack. **Affected Applications** mentions some of the datasets, systems, and real-world applications that are susceptible to the attack.

#### 3.1.3 Adversary's Model

In addition to the attack attributes, we include adversary model to capture a threat agent that has the power to, exploit a vulnerability or conduct other damaging activities in AI-enabled software. To model the adversary, the following characteristics are considered: **Goal** expresses the specific attack goal and motivation. **Security Violation** illustrates the type of violation, which could be one or a combination of these types: *integrity violation*, in which the adversary enters the model without undermining the normal system's performance, in *availability violation*, the attacker compromises the normal performance of the model, and in *privacy violation* the attacker obtains private information about the model or the data (Muñoz González et al., 2017). **Attack specificity** shows if the attacker aims to affect only a specific subset of the samples, which is called *targeted*, or any sample in the system, which is called *indiscriminate*. **Error specificity** shows whether the attacker wants to misclassify the samples into the desired class or any classes other than the true class, which is called *error specific* and *error generic*, respectively (Muñoz González et al., 2017). **Assumption** explains the adversary's access over the model and its resources. **Capabilities** indicates the abilities and the malicious potentials of the attack. **Weaknesses** represents the adversary's weaknesses and its blind-spots. Finally, **Level of Expertise** shows the required skill level to perform the attack.

### 3.2 Mitigation Techniques

The "mitigation" component presents available techniques and approaches that can be used to mitigate or defend from the attacks. Each entry in this component is mapped to the entries of the "attacks" component and has the following attributes: **Approach** briefly describes the mitigation method. **Effect** explains how utilizing the mitigation technique makes the system more robust against attacks. **Type** indicates if the technique is defending the model before the attack happens (*proactive*) (Papernot et al., 2016), or the technique is responding to the attack after the attack happened (*reactive*) (Grosse et al., 2017). **Advantage & Disadvantage** assesses the benefits and drawbacks of utilizing the mitigation technique, considering its efficiency, cost, and side effects.

### 3.3 Tools

This component provides information about tools that enable users to deploy attacks on AI/ML algorithms, assess the model robustness, and provide mitigation techniques to make their model more robust against the attacks. Similar to the two previous components, entries of this component are mapped to entries of "attacks" and "mitigation" components. For each tool, the following attributes are considered: **Input** and **Output** indicate the type and format of the inputs and outputs of the tool. **Offensive/Defensive** indicates whether a tool is called to carry out an attack or is designed to defend against the attacks. **Availability Status** describes the tool's availability, whether it is available to the public, open-source, commercial, revised from a prior toolchain, or it is entirely new.

### 3.4 Implementation

We collect the information we obtained from our systematic literature review in online knowledge base captured as tables. The tables' rows correspond to the techniques discussed in the paper, and the columns represent different attributes discussed earlier in this section. Then we used this information to develop the online framework[4]. The framework is provided with several filters, which allows users to focus on a specific problem and get into details as necessitated, evaluate the problem from different viewpoints, and find an appropriate solution with respect to the advantages, disadvantages, and side effects of the solution.

## 4 Conclusion

In this work, we presented a framework to establish a comprehensive knowledge base of attacks on AI/ML systems with appropriate mitigation techniques and defensive and offensive tools. To develop the presented meta model of the framework, we performed a systematic literature review and analyzed selected papers from various perspectives. The presented meta model (framework's skeleton) thoroughly describes all information required to know about various attacks, mitigation techniques, and tools and describes the relationship between them. The paper presents initial attributes for each component, which will be extended in the immediate future. Even though the presented framework is in its initial phase, it is already capable of covering a very broad spectrum of attacks on AI/ML algorithms in various application domains.

[4] www.design.se.rit.edu/programs/ai-ml-framework

## References

MITRE ATT&CK. MITRE ATT&CK. https://attack.mitre.org/, 2021.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Support vector machines under adversarial label noise. In *Asian conference on machine learning*, pp. 97–112. PMLR, 2011.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks, 2017.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel. On the (statistical) detection of adversarial examples, 2017.

Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. M. Leung. A survey on security threats and defensive techniques of machine learning: A data driven view. *IEEE Access*, 6:12103–12117, 2018. doi: 10.1109/ACCESS.2018.2805680.

MITRE. Common weakness enumeration. https://cwe.mitre.org/, 2021.

S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. Deepfool: A simple and accurate method to fool deep neural networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2574–2582, 2016. doi: 10.1109/CVPR.2016.282.

Luis Muñoz González, Battista Biggio, Ambra Demontis, Andrea Paudice, Vasin Wongrassamee, Emil C. Lupu, and Fabio Roli. Towards poisoning of deep learning algorithms with back-gradient optimization. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, AISec '17, pp. 27–38, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450352024. doi: 10.1145/3128572.3140451. URL https://doi.org/10.1145/3128572.3140451.

N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami. Distillation as a defense to adversarial perturbations against deep neural networks. In *2016 IEEE Symposium on Security and Privacy (SP)*, pp. 582–597, 2016. doi: 10.1109/SP.2016.41.

Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z. Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European Symposium on Security and Privacy (EuroS P)*, pp. 372–387, 2016. doi: 10.1109/EuroSP.2016.36.

Nikolaos Pitropakis, Emmanouil Panaousis, Thanassis Giannetsos, Eleftherios Anastasiadis, and George Loukas. A taxonomy and survey of attacks against machine learning. *Computer Science Review*, 34:100199, 2019.

R. Shokri, M. Stronati, C. Song, and V. Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 3–18, 2017. doi: 10.1109/SP.2017.41.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks, 2014.

Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction apis. In *25th USENIX Security Symposium (USENIX Security 16)*, pp. 601–618, Austin, TX, August 2016. USENIX Association. ISBN 978-1-931971-32-4. URL https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/tramer.

Y. Xiang, Y. Xu, Y. Li, W. Ma, Q. Xuan, and Y. Liu. Side-channel gray-box attack for dnns. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 68(1):501–505, 2021. doi: 10.1109/TCSII.2020.3012005.

Han Xu, Yao Ma, Hao-Chen Liu, Debayan Deb, Hui Liu, Ji-Liang Tang, and Anil K Jain. Adversarial attacks and defenses in images, graphs and text: A review. *International Journal of Automation and Computing*, 17(2):151–178, 2020.

Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE transactions on neural networks and learning systems*, 30(9):2805–2824, 2019.