ADVERSARIAL MACHINE LEARNING- TYPES OF ATTACKS & DEFENSES

What Is Adversarial Machine Learning?

- Adversarial Machine Learning is a branch of AI and machine learning that studies how attackers can deliberately exploit machine learning systems and how to defend against such attacks.
- Key points:
- Adversarial attacks involve intentionally modifying input data to trick a machine learning model into making incorrect predictions or revealing sensitive information.
- AML research focuses on:
  - Developing and understanding attack techniques.
  - Designing defence mechanisms that make models more robust and secure.
- Security is not only about the model itself but also about the surrounding system in which the model operates.
- System-level protections, such as limiting the number of queries a user can make or restricting the types of questions that can be asked, can reduce the risk of attacks.
- For example, attempting to extract confidential model information by repeatedly querying a chatbot becomes more difficult when query restrictions are in place.
- Although system-level security is important, most AML research concentrates on attacks targeting machine learning models and their training data.

Types of Adversarial Attacks

- The effectiveness and design of adversarial attacks often depend on how much information an attacker has about the target model.
- Two common categories are
- white-box attacks and
- black-box attacks.

1. White-Box Attacks

- In a white-box attack, the attacker has complete knowledge of the machine learning model, including:
- Model architecture
- Parameters and weights
- Training data
- Learning algorithms
- Example: An open-source large language model such as Llama 3.1, where the model details are publicly available.
- Advantages for attackers:
- Easier to analyse the model and identify weaknesses.
- More precise adversarial examples can be crafted.

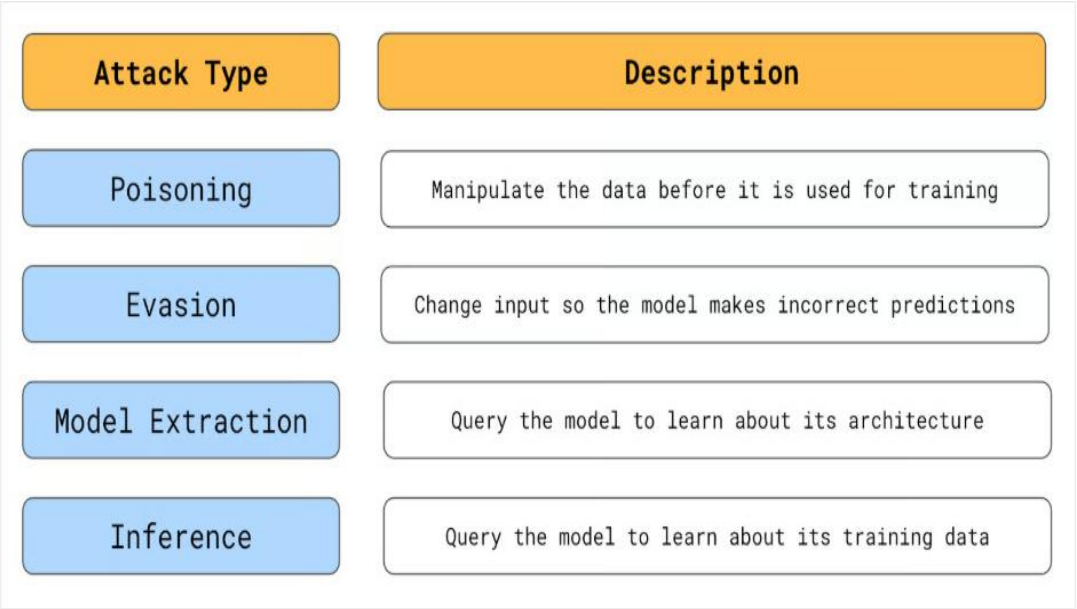
Advantages for defenders:

- The broader research community can inspect the model.

- Vulnerabilities may be discovered and fixed before attackers exploit them.

2. Black-Box Attacks

- In a black-box attack, the attacker has very limited knowledge of the model. They cannot see:
- Internal architecture
- Parameters or weights
- Training data
- Instead, they can only:
- Submit inputs (queries)
- Observe outputs (predictions or responses)
- Example: Commercial models such as GPT-4o mini, where users interact through an API or chat interface without access to the internal model.
- Challenges for attackers:
- Less information about the model's inner workings.
- More experimentation is required.



Poising attacks

- A poisoning attack is an adversarial attack in which an attacker deliberately manipulates the training data used to build a machine learning model. The goal is to corrupt the model's learning process so that it makes incorrect predictions after deployment.

How Poisoning Attacks Work

- An attacker can:
- Modify existing training data.
- Inject malicious data into the training dataset.
- Introduce incorrect labels for specific examples.

- As a result, the model learns incorrect patterns and behaves as the attacker intends

Example

- Consider a fraud detection system:
- Legitimate labels:
- Fraudulent transaction → Fraud
- Genuine transaction → Not Fraud
- An attacker could intentionally relabel certain fraudulent transactions as Not Fraud in the training data.
- After training, the model may learn that transactions with similar characteristics are legitimate, allowing the attacker to carry out fraud without being detected.

Factors That Increase Poisoning Risk

- 1. Public Data Sources
- Models that continuously learn from user-generated content are more vulnerable because attackers can influence the incoming data.
- 2. Frequent Model Updates

Defences Against Poisoning Attacks

- Common mitigation strategies include:
-
- Data validation and sanitisation.
- Human review of training data.
- Outlier and anomaly detection.
- Monitoring data sources for suspicious behaviour.
- Restricting direct learning from untrusted public inputs.
- Robust training techniques designed to withstand corrupted data.

Evasion attacks

- Evasion attacks target a machine learning model during the prediction (inference) phase, rather than during training.
- The attacker modifies the input data so that it appears legitimate but causes the model to make an incorrect prediction.

How Evasion Attacks Work

- An attacker alters the data submitted to the model while keeping the changes subtle enough that they appear normal.

For example:
- A loan approval system uses the applicant's country as a risk factor.
- An attacker from a high-risk country uses a VPN to conceal their true location.

- The model receives misleading information and may approve a loan that would otherwise be rejected.
- In this case:
- Training data remains unchanged.
- The attack occurs only when the model is making predictions.

Model extraction attacks

- A model stealing or model extraction attack is an attack in which an adversary attempts to discover how a machine learning model works by repeatedly interacting with it. The ultimate goal is often to create a copy of the original model or learn enough about it to launch more effective attacks.
- Objectives of Model Extraction Attacks
- Attackers may pursue model extraction for several reasons:
- Financial Gain
  - A valuable proprietary model can be copied and used without paying for its development.
  - For example, a sophisticated stock trading model could be replicated and used to make investment decisions.
- Preparation for Future Attacks
  - Understanding a model's behaviour makes it easier to craft adversarial examples, conduct evasion attacks, or identify vulnerabilities.

How Model Extraction Works

- In most cases, attackers do not directly access the model's internal parameters. Instead, they:
- Submit many carefully chosen inputs.
- Observe the outputs returned by the model.
- Analyse the relationship between inputs and outputs.
- Build a substitute model that mimics the behaviour of the target model.
- Over time, the attacker can develop a surprisingly accurate approximation of the original model.

Inference attacks

- Inference attacks aim to extract sensitive information from a machine learning model, particularly information that was present in its training data. Unlike model extraction attacks, where the goal is to copy the entire model, inference attacks focus on recovering specific confidential details.
- How Inference Attacks Work
- Attackers interact with a trained model by submitting carefully designed queries and analysing the responses. Through this process, they may:
  - Directly obtain memorised information.
  - Infer confidential details from the model's outputs.
  - Determine whether specific data was used during training.

- The target information might include:
  - Personal information
  - Private conversations
  - Confidential business data
  - Passwords or authentication-related information
  - Medical or financial records

Defending Against Adversarial Attacks

1. Adversarial Training

- Adversarial training strengthens a model by including adversarial examples in the training dataset.

How it works
- Generate adversarial examples using known attack techniques.
- Add these examples to the training data.
- Retrain the model on both normal and adversarial inputs.
- Benefit
- The model learns to recognise and correctly handle manipulated inputs, making it more resistant to evasion attacks.

2. Defensive Distillation

- Defensive distillation is a technique that trains one model (the student) to imitate another model (the teacher).

- Process
- Train a teacher model on the original dataset.
- Use the teacher's output probabilities (soft labels) instead of hard class labels.
- Train a student model using these soft labels.
- Benefit
- Produces smoother decision boundaries.
- Makes the model less sensitive to small adversarial perturbations.
- Improves robustness against some attack methods.

2. Gradient Masking

- Many adversarial attacks rely on gradient information to determine how inputs should be modified.
- Gradient masking attempts to hide or obscure these gradients.
-
- Techniques
- Non-differentiable layers
- Binary activation functions
- Randomisation methods
- Stochastic network behaviour
-
- Benefit

- Attackers find it harder to calculate how to manipulate inputs effectively.

4. Intrinsically Interpretable Models

- A powerful but often overlooked defence is choosing simpler models when possible.
-
- Examples include:
- Linear Regression
- Logistic Regression
- Decision Trees (in many cases)

- Why They Are Safer
- These models are intrinsically interpretable, meaning humans can easily understand how predictions are made.
- Benefits include:
- Easier vulnerability identification
- Easier auditing
- Greater transparency
- Reduced hidden behaviours
- Complex deep neural networks often contain thousands or millions of parameters, making weaknesses difficult to detect.

5. System-Level Defences

- AML is not only about protecting the model itself. The surrounding environment is equally important.
- Data Validation and Sanitisation
- Before training:
- Verify data integrity.
- Remove suspicious samples.
- Check labels for inconsistencies.
- Filter malicious inputs.
- This helps prevent poisoning attacks.

- Anomaly Detection
- Specialised anomaly detection systems can:
- Identify unusual inputs.
- Detect potential adversarial examples.
- Stop suspicious data before it reaches the main model.

- Access Controls
- Organisations may also:
- Limit model queries.
- Restrict API access.
- Monitor suspicious usage patterns.
- Rate-limit users.
- These measures reduce the effectiveness of model extraction and inference attacks.