Information Security

A comprehensive study about the current challenges in securing AI systems

Diploma Thesis

For attainment of the academic degree of

Diplom-Ingenieur/in

submitted by

Bernadette Jilch

is191816

in the

University Course Information Security at St. Pölten University of Applied Sciences

The interior of this work has been composed in LaTeX.

Supervision

Advisor: Dipl.-Ing. Peter Kieseberg

St. Pölten, September 1, 2021 ____________________ ____________________

(Signature author) (Signature advisor)

St. Pölten University of Applied Sciences, Matthias Corvinus-Straße 15, A-3100 St. Pölten,T: +43 (2742) 313 228, F: +43 (2742) 313 228-339, E:office@fhstp.ac.at, I:www.fhstp.ac.at

Declaration

I declare that to the best of my knowledge and belief

- This thesis is my own, original work composed entirely by myself.
- I have made no use of sources, materials or assistance other than those which habe been acknowledged.
- This work has not previously been published, accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Date

Signature

Kurzfassung

Die Anwendung von Systemen, die mit einer künstlichen Intelligenz (AI) zusammenarbeiten beziehungsweise auf dieser basieren, ist heutzutage keine Seltenheit mehr und stellt die Informationssicherheit vor neue Herausforderungen in Bezug auf Risikomanagment und Threat-Modelling.

Vorhersagen gehen davon aus, dass etwa 30% aller Bedrohung für AI-Systeme auf der Manipulation von Trainingsdaten, dem Diebstahl von AI-Modellen oder feindlichen Angriffen beruhen werden. Diese Tatsache betont die Wichtigkeit für eine umfassende Darstellung aller Angriffsmöglichkeiten, um Mitigationsstrategien für AI-Systeme zu entwickeln. In einer umfassenden Studie wurden 157 Risiken und Bedrohungen identifiziert und nach ihrem Angriffsziel und ihrem Auftreten im Lebenszyklus des AI-Systems sortiert, welche die folgenden sind:

1. Rohdaten
2. Datensatzzusammenstellung
3. Datensätze
4. Lernalgorithmus
5. Auswertung
6. Eingabe
7. Modell
8. Inferenz-Algorithmus
9. Ausgabe
10. Systemweite und allgemeine Belange

Außerdem wurden die aktuellen Risiken von AI-Systemen in der realen Welt skizziert, hierfür wurden Angriffe auf AI-basierte Systeme aus dem autonomen Fahren sowie auf Anwendungen, die ihren Einsatz in der Cybersecurity Domäne finden, als Beispiele herangezogen. Die AI-Prinzipien von Unternehmen und Forschung über AI-Systeme analysiert, wobei die Erkenntnis gezogen wurde, dass die Risiken und Bedrohungen durch AI-Systeme in diesen zu wenig oder gar nicht repräsentiert werden.

Abstract

Working with systems, that are based on Artificial Intelligence (AI) is a normal thing today, but what are known threats and risks for these systems? Predictions are that about 30% of all AI attacks will be based on training data poisoning, AI model theft or adversarial examples.

Therefore, threat and risk modelling for these systems will become more and more important, this is why a comprehensive study about these was conducted, which resulted in 157 identified risks and threats for AI systems, these were sorted based on their attack goal and appearance in the lifecycle of AI systems, which are following:

1. Raw data
2. Dataset assembly
3. Datasets
4. Learning algorithm
5. Evaluation
6. Input
7. Model
8. Inference algorithm
9. Output
10. System-wide and broad concerns

In addition, the current risks of AI systems in the real world, using the example of autonomous driving and AI-based monitoring systems for the cybersecurity domain, were outlined and summarised. Moreover, principles of companies and research about trustworthy AI were analysed, which resulted in the finding, that risks and threats ofAI are not or too little represented in these.

Contents

1 Introduction

According to Microsoft there has been a notable increase in attacks on commercial Machine Learning (ML) systems in the last four years [1]. The report from Gartner's Top 10 Strategic Technology Trends for 2020 [2] predicted, that about 30% of all AI attacks will be based on training data poisoning, model theft or adversarial examples. Now you may think, why does this concern me or our industry?
Over the years there was a continuous progress in AI systems and an increased use of AI-based applications and systems. These systems are found in everyday life - in robot vacuum cleaners in order to prevent hitting obstacles and learning from the floor plan, in smart online translators for getting the best translations, in customized google searches in order to improve the individual shopping experience, in the search assistant on smartphones, in google maps in order to predict traffic jams, and AI can even be integrated into smoke detectors, to name a few examples. Moreover, AI systems are used in the public and enterprise sector for automated network traffic analyses, intrusion detection, malware detection, recognition tasks, medical image analyses - in fact they are used data and prediction analyses in general.

Although these technologies are widely utilized in different ways and across many industries, the main concerns regarding privacy loss, amplification of bias and social remain. There are a number of mechanisms to secure the privacy and the model of these AI systems, but standardization, which is the foundation of secure AI development, is still in progress. [3]–[5].

1.1 Thesis Outline

This thesis is organized with chapter 1 introducing the current problem of securing AI systems. In chapter 2 prerequisites and fundamental knowledge to understand the terms and techniques of AI systems are described. It also describes terms from the cybersecurity domain, that are necessary to understand the further AI systems. In chapter 3 the research background and questions are defined, whereas in chapter 4 the threats and risks regarding AI systems are listed. In chapter 5 adversarial attacks in the wild for autonomous driving and AI-based mechanisms in the cybersecurity domain are outlined. In chapter 6 the phenomena of trustworthy AI are analysed regarding AI principles from research and companies, and chapter 7 concludes the topic with a brief summary and an outlook into future work possibilities.

2 Prerequisites

First of all, even though colloquially AI, ML and Deep Learning (DL) are sometimes used as synonyms, they are not the same. Inernational Business Machines Corporation (IBM) [6] used the concept of Russian dolls - as can be seen in Figure 2.1 - to showcase that they are subfields of each other.

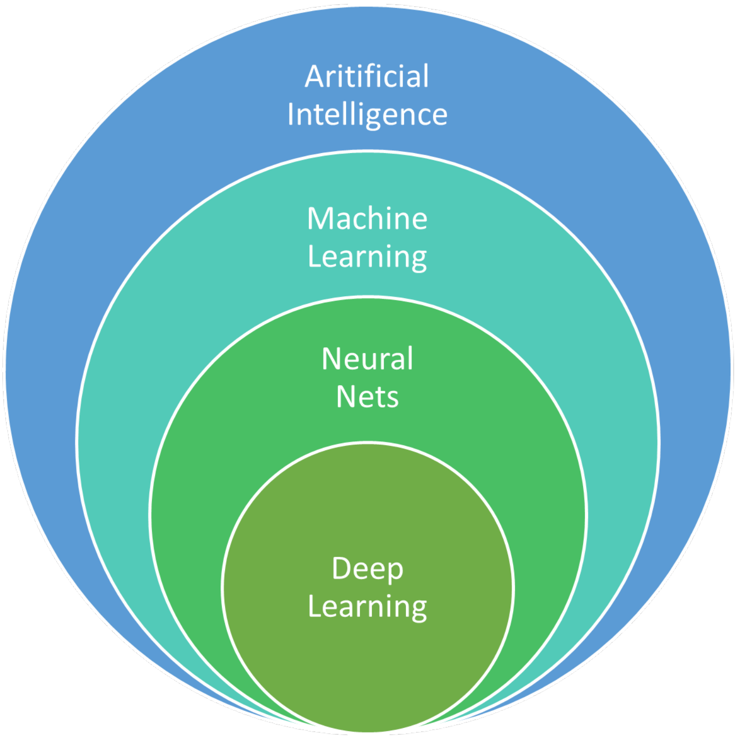

Figure 2.1: Euler diagram, which symbolises the interdependence between the domains (AI, ML, Neural Network (NN), DL

2.1 Artificial Intelligence

The general term for this branch of science is AI. AI therefore should be used in the same way the terms *biology* and *physics* are used in their respective scientific fields. AI's focus is to build intelligent programs and machines, that are able to solve problems. It can also refer to a system, which is capable of performing tasks, that require a sort of intelligence, these tasks are often being learned from data and/or experience. Generally the term AI is the broader concept and often used to describe their subfields (ML, DL). The application field of these systems is wide-ranging from picture classification to intrusion detection systems [7], [8]. In the context of security and safety, the impact of AI attacks on critical machines is very high and successful attacks have been demonstrated in the research, as well as reported in the wild [9].

2.1.1 Machine Learning

Machine Learning is a subset of AI, which refers to a system, that has the ability to learn automatically and therefore improve through experience, without being manually adapted or explicitly reprogrammed. This learning process is possible due to a provided dataset, which is used by the system to adapt its identification / classification algorithm, because it learns the differing characteristics of e.g. objects [7], [8], [10].

2.1.2 Deep Learning

Deep Learning is a subset of ML, which is based on NN in order to analyse different factors. It is inspired by the NN of human brains and is often referred as the *next evolution of ML*. DL systems are taught to classify various types of information and identify patterns similar to human brains. The difference between ML and DL is, that ML needs to be manually provided with the classification features, whereas DL automatically discovers these features. However, to complete this task DL needs comparable more training data and better computing power to deliver accurate results [7], [8]. DL is split into many subcategories, which are:

- Artificial Neural Network (ANN) [11], [12]
- Deep Neural Network (DNN) [12]–[14]
- Convolutional Neural Network (CNN) [11], [12], [15]
- Feedforward network [12]
- Recurrent Neural Network (RNN) [12]

Artificial Neural Network

Artificial Neural Network (ANN) is inspired by the biological NN of human brains and is based on perceptrons called neurons. Simplified, a perceptron is an algorithm, which maps a set of inputs to an output using an activation function. In the learning phase the weightings and activation function are trained to correctly classify the outputs. Artificial Neural Network (ANN) can be supervised and feedforwarded, like inConvolutional Neural Network (CNN) and Deep Neural Network (DNN), or it can work unsupervised (e.g. self-organising maps) with learning rules [11].

Deep Neural Network

A Deep Neural Network (DNN) is a large neural network, which works with multiple hidden layers clustered in neurons. These neurons are like individual computing units and are connected by links with different weightings and biases, neurons transport the inputs. A DNN mimics the neural network in the human brain, which learns and builds knowledge from the dataset. Therefore, these networks are often used for complex problems, that cannot be described as linear or non-linear problem. They extract the features from the unprocessed input and enhance the performance of the system by finding hidden (latent) structures in the unstructured, unlabelled data. Moreover, they are based on continuous real-valued vector representations, which are beneficial for handling data in various formats, like images, text, video and audio. A DNN is capable of non-linear mapping from original high-dimensional data points to lower dimensional space. However, the explainability of these systems is not given, they work as black-boxes. This means, that the learned classification of the neurons is difficult to retrace, which is one reason why the robustness of these systems can hardly be evaluated [11], [15]. Therefore, researchers used small, imperceptible perturbations to assess the robustness of DNN and concluded that these networks are not robust against these perturbation attacks. These attack inputs were named adversarial samples [13], [14].

Convolutional Neural Network

A Convolutional Neural Network (CNN) can consist of one ore more convolutional or sub-sampling layers, which is followed by one or more fully connected layers in order to share weightings and reduce the number of parameters in the model. The convolutional layer utilizes convolution operations in order to extract meaningful local patterns of input. The architecture of the system is designed for 2D input structures, like images or other computer vision tasks, since it is order-sensitive. The CNN creates a feature map, which is down-sampled, that means the dimensionality of the feature map is reduced, but stays robust to small distortions by retaining the most important informations. The task of the last fully connected layer is to classify the data by using the feature matrix formed by previous layers. The main purpose of CNN is feature extraction, but it is also used for image recognition tasks by data processing applications [11], [15]

2.1.3 Feedforward Neural Network

These networks have only connections in one directions - forward. They form an acyclic graph with designated input and output nodes. Each node computes a function of the input and passes the results on to the next node. There are no loops in the system [12].

2.1.4 Recurrent Neural Network

Recurrent Neural Network (RNN), is different to feed-forward networks, because they allow cycles in the computation graph [12].

2.1.5 Artificial Intelligence lifecycle

One of the most valuable assets of AI is data. It is continuously transformed during the different phases of the lifecycle. From the raw dataset during the ingestion phase, to exploration, pre-processing and feature selection in order to structure the dataset, all the way to the divided datasets to train, test and evaluate data. Besides data, there are also different lifecycle actors, that influence the AI systems and their threat vectors:

- An **AI (application) designer**, who creates the concept of the systems.
- An **AI developer**, that builds the software and algorithms.
- A **data scientists**, who interprets data, designs and develops AI models.
- A **data engineer** that extracts, collates and standardizes data from different sources.
- A **data owner**, who owns the datasets (often businesses).
- Moreover, there is a **data provider/broker**, who monetizes data.
- A **model provider**, who provides already trained systems, who can be cloud provider.
- There are **third-party providers**, that provide software frameworks and libraries.
- Finally, there are **end users**, like service consumers and model users [16].

Artificial Intelligence lifecycle phases

The AI lifecycle is divided into twelve different phases, which will be briefly listed and are graphically illustrated in Figure 2.2 [16]:

1. **Business goal definition** - In the first phase the business purpose of the AI system needs to be identified, the goal is, that the AI model supports the business purpose.
2. **Data ingestion** - Data is (dynamically) collected from multiple sources with corresponding context metadata, prepared according to the requirements of the AI system and imported into the system.
3. **Data exploration** - During this phase the data is verified to fit known statistical distribution and the corresponding statistical parameters are estimated.
4. **Data pre-processing** - Data is cleaned, integrated and transformed in this phase. For example, data is converted to a metric format, missing values are replaced by interpolation, outliers are filtered, data is anonymized/pseudonymized and augmented.

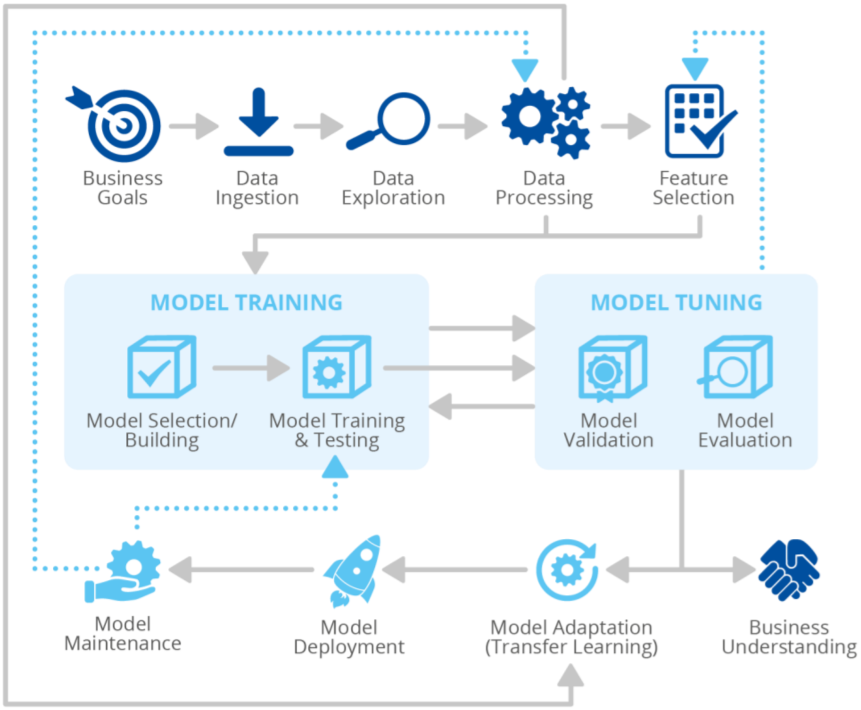

Figure 2.2: Reference model of a generic AI lifecycle - from identifying of business goals to model adaption [16]

5. **Feature selection** - Features are identified in accordance to the dataset, taking into account global parameters of the dataset, e.g. the overall variance of labels. The data is set along these dimensions, discarding others.
6. **Model selection/Building** - The AI model most suitable for the application is selected. Data input vectors are encoded to match the required input format.
7. **Model training** - In this phase the selected training algorithm is applied with appropriate parameters, which modifies the chosen model to fit the provided training data. Moreover, the model training phase is validated through a validation dataset, this strategy is called cross validation.
8. **Model tuning** - In this phase the hyper-parameters are validated through the validation datasets and adapted if necessary.
9. **Transfer learning** - A pre-trained AI model in the same application domain is started and trained, in order to improve its accuracy through transfer learning.
10. **Model deployment** - In the deployment process the trained model is made available to the users, through software, firmware or hardware.
11. **Model maintenance** - The inference results of the model and the received input data should be mon-

itored, in order to identify changes of the model and retrain the model, if needed.

12. **Business understanding** - In this phase the added value of the AI system should be assessed.

Another generic AI system is described as followed. Processes are represented as ovals in figure 2.3, whereas collections of data information are represented in rectangles [17].

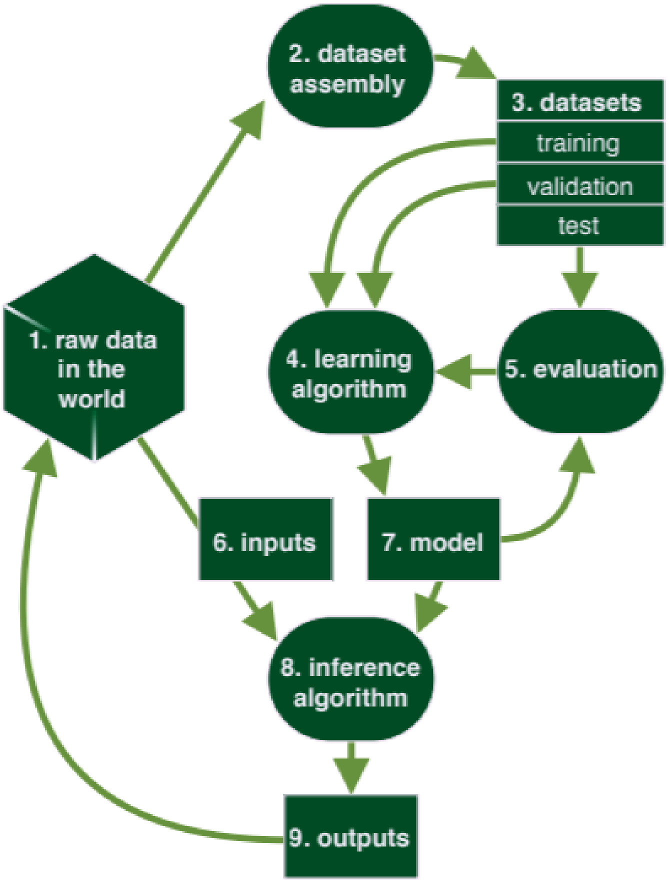

Figure 2.3: Components of a generic ML application with information flow represented as arrows [17].

2.2 ML algorithms

2.2.1 Supervised learning

By supervised learning a supervisor assists the AI system throughout the training process. Moreover, the training dataset includes labelled data.

For example, the system should be able to classify objects as triangles, circles and rectangles. In the first step the data is presented with a label, then the program runs in a validation loop, that checks, if the function can classify the objects correctly. The system makes allegations and is corrected by the supervisor, if the classification is wrong. This training process ends, when the desired accuracy level is reached on the training data. This learning method is often used for regression and classification, like language detection, spam filtering or computer vision in general. Algorithm examples are [7]:

- Naive Bayes
- Support Vector Machine
- Decision Tree
- K-Nearest Neighbours
- Logistic Regression
- Linear and Polynomial regressions

2.2.2 Unsupervised learning

For unsupervised learning systems no assistance, labels or features are provided, which allows the system to search for patterns independently.

For example, sorting the cutlery by separating into different categories, such as forks, spoons and knives. This would be a typical task for unsupervised learning, which is based on clustering. It is possible due to the similarity of objects of one category. Furthermore, unsupervised learning can be useful in data analytics, e.g. detecting fraudulent transactions, analyse customer preference based on their search history, anomaly detection, risk management, fake image analysis. Algorithm examples are [7]:

- K-means clustering
- DBSCAN
- Mean-Shift
- Singular Value Decomposition (SVD)
- Principal Component Analysis (PCA)
- Latent Dirichlet allocation (LDA)

- Latent Semantic Analysis, FP-growth

2.2.3 Semi-supervised learning

By semi-supervised learning, the training data consists of a small amount of labelled data and a larger amount of unlabelled data. The system has to find patterns to structure the data and make decisions based on it [7].

2.2.4 Reinforcement learning

In reinforcement learning a trial and error loop exists. Thus it is similar to the human learning process. It does not need to be supervised constantly, however, reinforcement signals in response to the actions are necessary.
A classic example is a child, who touches a hot stove top once and learns from the pain not to do it again. This concept enables the system to work with dynamic data and not only static datasets, which means it learns in the real world, in *noisy environments*. A good reinforcement learning setting are games, because their game scores are a good signal to train behaviours, moreover, they are used for self-driving cars, robots and resource management. Algorithm examples are [7]:

- Q-Learning
- Genetic algorithm
- SARSA
- DQN
- A3C

2.3 Attacks

2.3.1 Perturbation

Perturbation describes the state of being disturbed [18], in the context of an attack it means, that small noises are added to an original input, with the aim to influence the system. The modified input is called perturbed input. The changes (added noise) have to be specially crafted in order to be under a certain value. Stealthy perturbation attacks are not perceptible to humans and lead to misclassification of the input by the system [15], [19].

Adversarial Input

These perturbed inputs were named adversarial examples and this notation is now used as a general denotation of all kinds of perturbation samples [13]–[15]. These samples can also be called adversarial inputs. The added noise leads to a classification boundary jump for the original input, which means the adversarial sample leads to a misclassification [9].

2.3.2 Transferability

Transferability means, that the same attack can be used on different models. It describes the generalization of the attack method. *Cross-model-generalization* describes the transferability of an adversarial example, which was generated for one DNN and one dataset, between neural networks, while *cross-data generalization* describes the transferability of models with different datasets. Transferability is mostly reached by black-box attacks, because the information of the system are unknown and therefore does not affect the attack method as much [15]. Transferability is categorized into three levels in DNN [15]:

- the same architectures with different datasets
- different architectures with the same datasets
- different architectures with different datasets

2.3.3 Attack Types

Attack types are split into three different categories, which describe the amount of knowledge, that the attacker has about the targeted ML model. These categories are defined as followed:

- **White-box** - in these attacks the attackers knows everything, including learned weights, parameters, information about the training phase, the architecture about the model and so forth. These attacks are especially interesting to evaluate the extent of exposure to internal adversaries. However, these scenarios are usually not realistic and therefore rarely occur in reality.
- **Black-box** - by these attacks the knowledge of the attacker about the targeted system is limited to none existing. The usual attack starts by querying the model and observing the labels or confidence scores.
- **Gray-box** - in these scenarios the attacker's information lies between black- and white-box scenarios.

Thus, an inverse relationship between the information of the adversaries and the attack difficulty exists, and a direct relationship between the attack difficulty and the attack complexity [9], [19].

2.4 Assets

Assets are anything, that has a value for a stakeholder, therefore they range from information, software, hardware and other physical assets, services, people and their areas of expertise, as well as intangibles like reputation and public image [20].

2.5 Threat Modelling

Adam Shostack describes threat modelling as something everybody does, like sneaking something through security at the airport or stealing away from an event with the risk of being caught [21]. In general two types of models are made, one describing what is being build and one describing what can go wrong, the threats. A threat model defines the possible attack vectors of a specific system. It takes resources into account, which might be available to potential attackers, and the system specifications. Different cyber attack models were introduced over time, the most established frameworks are [19], [21]–[23]:

- **Attack trees** - introduced by Bruce Schneier in 1999. Attacks against a system are presented in a tree structure, where the goal is the leaf node to be achieved through multiple leaf nodes.
- **STRIDE** - is a mnemonic and stands for spoofing, tampering, repudiation, information disclosure, denial of service and elevation of privilege, which describes threats, that are not beneficial in secure settings.
- **Cyber kill chain** - was introduced by Lockheed Martin analysts in 2011, which describes the structure of an attack, from target identification to its destruction.
- **MITRE's adversarial tactics, techniques and common knowledge (ATT&CK)** - a collection of pre- and post-compromise techniques.

The key essential questions of all frameworks are:

- What are you building?
- What can go wrong with it, once it is built?
- What should you do about those things, that can go wrong?
- Did you do a decent job of analysis?

The following adversarial attack categorization was introduced by Huang et al. [24] in 2011:

- **Influence** - describes if the training process is influenced by the training data (causative) or if the data discovery is the goal (exploratory).
- **Security violation** - integrity attacks resulting in false negatives, availability attacks causes classification errors and privacy violations are a matter of confidentiality.

- **Specificity** - describes the goal of the attacker, is it targeted, then only some target points are interesting, or is it indiscriminate, then the goal is more flexible.

Including threat modelling already in the development lifecycle phase of software can mitigate future threats, security issues and additional costs [19], [21], [22].

2.6 Malware and obfuscation

The word malware is composed of the word malicious and software. The name is very descriptive, because malware is often software with unwanted behaviour, that can cause intentional damage to systems or a company. For example, it can infect computer systems, which endangers the confidentiality and integrity of the systems. It can even be capable of replicating, propagating and executing itself. In order to avoid detection of these malicious software, attackers can use polymorphic or metamorphic techniques to make the identification of malicious software more difficult [25]. Obfuscation can be used to protect the intellectual property against theft, but it also can be used to hide the functionality of a malicious software. The goal of obfuscation is to make the recovery of the internal software logic as as hard as possible. This can be achieved by implementing redundancies into original programs, however, the semantics of the original program have to stay the same [26].

2.7 Spam and Ham

The distribution of unsolicited messages is called spamming. They are often sent in bulk using email, while emails, that are sent for legitimate reasons, are known as Ham [27], [28]. Spam mails are used for multiple purposes, like marketing, but also for malicious reasons (reputational damage, financial disruption, ...). Moreover, emails are a popular choice for scammers to deliver malware, therefore, financial gain is the main motivation for spammers. An estimation from 2018 says, that annually about 3.5 million USD are gained by spamming [28], [29].

2.8 Malicious use of AI

The main focus of Brundage et. al. [10] is the use and the duality of AI systems. A good example for AI's duality is the offensive and defensive side of security. The intended use can differ from the actual purpose, which is the phenomena of the dual-use.

The focus of the described threat landscape is kept very general - expanding existing threats, introducing new threats and altering the typical character of threats. In Brundage et. al. scenarios of misuse are outlined in different security categories:

- **Digital security** - automated tasks carried out by AI systems, like automated social engineering attacks, automated vulnerability discovery, human-like denial-of-service, prioritizing targets for cyber attacks using ML, data poisoning attacks and model extraction.
- **Physical security** - the misuse of AI systems, e.g. the usage of drones, autonomous vehicles to deliver explosives or causing accidents.
- **Political security** - misuse of AI from the governmental side, e.g. the creating fake news reports with highly realistic videos or audio (deep fakes).

Moreover, Brundage et al. [10] proposed four high-level recommendations:

- The collaboration and cooperation between political decision-makers and researchers with the aim of preventing, mitigating and investigating the misuse of AI systems.
- Consideration of the dual-use nature by researchers and engineers in order to influence research priorities and norms when the misuse of AI systems is foreseeable.
- Best practices for dealing with dual-use concerns should be established for the AI research.
- Expand the stakeholders and domain experts in the discussion of these challenges.

3 Approach

The aim of this thesis is to outline the current risks of AI systems in the real world on the example of autonomous driving and AI-based technologies used in the cybersecurity domain, and to aggregate specific and general threats and risks regarding AI systems. Moreover, trustworthy AI and its definition in principles is discussed.

3.0.1 Research questions

My research questions for this thesis are:

- How are AI threats and risks described in research?
- What are current AI attacks in the wild, on what threat are they based on?
- What is the current state of trustworthy AI principles in the research as well as in the industry?

3.0.2 Method

In order to answer this research questions, a comprehensive literature review was conducted and the results are outlined in this thesis.

4 Threats and Risks

4.1 ENISA

European Union Agency for Cybersecurity (ENISA)'s goal is to strengthen the security in the EU. It published two guides called *cybersecurity challenges* regarding AI, one summarizing the threat landscape of AI and the other describes the the challenges behind autonomous driving.

4.1.1 AI Challenges

The main chapters in ENISA's publication are the lifecycle, assets and threats regarding AI. ENISA describes the importance of trustworthy AI deployment and highlights black-box model of AI systems, that can be influenced by attackers. The main points according to ENISA in securing AI are [16]:

- **Assets** - Understanding what needs to be secured
- **Data** Understanding of the related data governance models
- **Shared models and taxonomies** - Managing threats in multi-party ecosystems
- **Controls** - Developing controls to ensure a secure AI AI itself

4.1.2 AI threats

A distinction between the threat actors should be made, because they vary in intention, motivation, financial support, knowledge and skill sets. Potential threat actors, that need to be considered are cybercriminals, insiders, malicious, non-malicious (accident), nation state actors, terrorists, hacktivists, script kiddies and competitors. According to ENISA firstly, the security-relevant properties of the systems need to be identified and secondly, the communication paths and dependencies of the AI application need to be defined. After these steps the critical assets, that need to be secured must be identified and associated threats derived from these objectives. Finally, the vulnerabilities from the system must be identified.

The security properties, that were considered, were confidentiality, integrity, availability, authenticity, authorization and non-repudiation, which are fundamental characteristics in the security field. Moreover, AI specific attributes are added including: robustness, trustworthiness, safety, transparency, explainability, accountability and data protection.

ENISA works with the following threat taxonomy and differentiates between them:

- nefarious activities / abuse (NAA)
- eavesdropping / interception / hijacking (EIH)
- physical attacks (PA)
- unintentional damage (UD)
- failures / malfunctions (FM)
- outages (OUT)
- disaster (DIS)
- legal actions (LEG).

This taxonomy is later used for section 4.7.

4.2 ISO/IEC TR 24028:2020

ISO is an independent, non-governmental international organisation with the goal to develop market relevant international standards, that support innovation as well as provide solutions [30].

New challenges arise based on AI usage like accountability, new security threats, privacy threats, improper specifications, inadequate implementation, improper use and various sources of bias, these are the current problems of AI system. New standards were and will be designed with new controls and mitigations, that need to be implemented in order to protect the system. For AI systems these include, according to ISO/IEC TR 24028:2020 [20]:

- transparency
- security controls
- privacy controls
- robustness
- resilience
- choice of ML algorithms
- configuration of ML algorithm
- data considerations (e.g. biases)
- system controllability considerations

Moreover, the risks of ML were outlined, which will be included in the following matrix section 4.7.

4.3 Berryville Institute of Machine Learning

The Berryville Institute of Machine Learning (BIML) was founded in 2019 with the mission to explore security implications built into ML systems. Therefore, the institute worked on an architectural risk analysis of ML. The researchers pursue the architectural risk approach, which analyses the system's design and therefore, encounters systematic risks, which can be mitigated during the design phase of software models.

4.3.1 Top ten ML security risks

BIML identified the following system-wide, top ten ML security risks, which can be intentional actions of attacker or even design flaws [17].

- **Adversarial examples** - are a common attack vector for AI, which provides malicious input (e.g. perturbations), that forces the AI system to make a false prediction or categorization.
- **Data poisoning** - happens when an attacker can manipulate the data of the system. Therefore, the possible effect on training data from outsiders has to be considered while designing the AI system.
- **Online system manipulation** - a system is online, when it continues to learn during its use. Attackers can manipulate these systems by retraining. The data provenance, algorithm choice and system operations play a significant role in the risk of online system manipulation and therefore, have to be considered.
- **Transfer learning attack** - ML systems, which are based on existing, already trained models, can be compromised, as these base models can already be manipulated.
- **Data confidentiality** - the challenge of protecting sensitive or confidential data in the ML context is important. Extraction attacks need to be considered.
- **Data trustworthiness** - data quality, provenance and integrity are essential for ML systems, public data sources and online models can bear a risk for the trustworthiness of data.
- **Reproducibility** - when systems and results cannot be reproduced, the classification of these systems cannot be trusted.
- **Overfitting** - overfit models can be easily manipulated, because the system has only learned the training set and is not able to generalise to new data.
- **Encoding integrity** - issues with the encoding integrity can bias a model.
- **Output integrity** - can be attacked, when an attacker interposes between the AI system and the user, therefore, output manipulation is possible.

4.4 Microsoft - AI and ML security

Microsoft differentiates between intentional failures (IF) by active adversary and unintentional failures (UF) when the ML system produces a correct, but unsafe outcome. The Microsoft team rated malicious/intentional attack classes with their security severity rating system (low, moderate, important, critical), which describes the severity of these attack vectors and their impact [31]–[34].

4.5 MITRE - Adversarial Threat Matrix

The Massachusetts Institute of Technology Research and Engineering (MITRE) along with 17 other business such as Microsoft, IBM, NVIDIA, Airbus, contributed to create a knowledge base of attack vectors for ML systems. In the threat model proposed by MITRE, attacks are distinguished during the learning phase of the system and after deployment, which is called *inference time*. Moreover, ML attacks are clustered into five categories, which are following [1], [35]:

Type	Attack	Description
Inference	Model evasion	In these attacks the adversary modifies the query in order to get the desired output. Model evasion is possible by iteratively querying and analysing the model's output.
Inference	Functional extraction	By iteratively querying the model, the attackers can retrain a substitute model.
Inference	Model inversion	In this scenario the attacker can extract features used to train the ML system, which can result in Membership inference attacks.
Train	Model Poisoning	By these attacks the attacker contaminates the ML model during the training phase, in order to manipulate predictions of new data at inference time. The access to the training data could lead to data extraction.
Both	Traditional attacks	Describe attacks with known tactics, techniques and procedures to attain the attacker's goal.

Table 4.1: The table summarizes the different threat categories identified by MITRE.

4.6 NIST - adversarial ML

The National Institute of Standards and Technology (NIST) was founded in 1901 and is part of the U.S. Department of Commerce. The aim of the institute is promote U.S. innovation and industrial competitiveness [36].

The report from NIST describes three different aspects of adversarial ML - attacks, defences and consequences. ML attacks can be targets of attacks using different techniques and knowledge about the system, which will be described in more detail in the following. The consequences are split into the following groups: integrity, availability and confidentiality violations [37].

The targets of adversarial attacks, according to NIST [37], depend on the ML stages, starting by the physical domain with input and output sensors, the digital representation with the pre-processing or the ML model itself.

4.7 Aggregated threats and risks collection

In this thesis an aggregated threat and risk collection for AI systems is introduced, which is subdivided in ten categories, in order to cluster these threats/risks by their appearance in the AI lifecycle.

4.7.1 Raw data

Threat Category	NAA, UD
Threat	Compromising AI inference's correctness - data
Description	The inference's correctness can be influenced by data manipulation, selection bias of raw data, modification of labels, deletion or omission of labelled data items
Potential Impact	Integrity
Source	ENISA, NIST

Table 4.2: Compromising AI inference's correctness - data

Threat Category	NAA
Threat	Data tampering
Description	Manipulation of data due to multiple causes, can introduce bias.

Potential Impact	Availability, integrity
Source	ENISA, NIST

Table 4.3: Data tampering

Threat Category	NAA
Threat	Introduction of selection bias
Description	Selection bias can be introduced through intentional selection bias in published raw data, which affects results adversely.
Potential Impact	Integrity, availability
Source	ENISA, BIML

Table 4.4: Introduction of selection bias

Threat Category	UD, FM
Threat	Lack of sufficient representation in data
Description	Raw datasets can always be not representative enough, for example the image is compressed in a lossy manner, therefore, assessments of data representatives is crucial in retrospect, in order to prevent biased data.
Potential Impact	Availability
Source	ENISA, BIML

Table 4.5: Lack of sufficient representation in data

Threat Category	UD
Threat	Erroneous encoding
Description	The data is not representative of the problem, that the AI is trying to solve. This can lead to biased model, because of the encoding process.
Potential Impact	Integrity
Source	BIML

Table 4.6: Erroneous encoding

Threat Category	UD
Threat	Wrong text encoding
Description	The system cannot process non-expected encoding, for example the system processes ASCII text, but the data is in Unicode.
Potential Impact	Integrity
Source	BIML

Table 4.7: Wrong text encoding

Threat Category	NAA, UD
Threat	Looping issues
Description	The output of the system is later used as input data and retrains the model, therefore, a retraining with adversarial examples can lead to misclassification.
Potential Impact	Integrity
Source	BIML

Table 4.8: Looping issues

Threat Category	UD
Threat	Data entanglement
Description	If the data is entangled, this should also be represented in the datasets in order to maintain information.
Potential Impact	Integrity
Source	BIML

Table 4.9: Data entanglement

Threat Category	NAA, UD
Threat	Metadata
Description	Metadata should be included in the raw data and protected from manipulation.

Potential Impact	Integrity
Source	BIML

Table 4.10: Metadata

Threat Category	NAA, FM
Threat	Blinded or calibrated sensors
Description	Input sensors can be blinded, moreover, sensors require for consistent feature identification human calibration.
Potential Impact	Integrity
Source	BIML

Table 4.11: Blinded and calibrated sensors

4.7.2 Dataset assembly

Threat Category	NAA, FM
Threat	Compromising ML pre-processing
Description	Through flaws or defects of the (meta-) data schemata, which influences the analysis of the data.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.12: Compromising ML pre-processing

Threat Category	NAA, UD
Threat	Reducing data accuracy
Description	Due to modified data or mixing of datasets with different data quality.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.13: Reducing data accuracy

Threat Category	NAA, FM
Threat	Scarce data
Description	The threat can be exploited deliberate or unintentionally by data scarcity, that can compromise AI viability or limit its results.
Potential Impact	Availability
Source	ENISA

Table 4.14: Scarce data

Threat Category	UD
Threat	Mishandling of statistical data
Description	Lack of minorities and bias representation for statistical significance.
Potential Impact	Availability, confidentiality
Source	ENISA

Table 4.15: Mishandling of statistical data

Threat Category	LEG
Threat	Profiling of end users
Description	Labelling can lead to profiling and therefore, threaten anonymity and privacy.
Potential Impact	Confidentiality
Source	ENISA

Table 4.16: Profiling of end users

Threat Category	NAA, UD, FM
Threat	Label manipulation or weak labelling
Description	Threats through wrong or imprecise data labels by supervised learning systems.
Potential Impact	Availability, confidentiality, integrity
Source	ENISA

Table 4.17: Label manipulation or weak labelling

Threat Category	NAA, UD
Threat	Manipulation of labelled data
Description	Labels and data is deleted / omitted, spuriously labelled or modified influencing model training and inference.
Potential Impact	Integrity
Source	ENISA

Table 4.18: Manipulation of labelled data

Threat Category	NAA
Threat	Overloading / confusing labelled dataset
Description	Adding random samples to the training set in order to disturb meaningful inference.
Potential Impact	Availability
Source	ENISA

Table 4.19: Overloading / confusing labelled dataset

Threat Category	FM
Threat	Inadequate / absent data quality checks
Description	The importance of data and quality markers (e.g. sample size, variances, applied data collection methodologies) are not present, which leads to inadequate data and poor performance.
Potential Impact	Availability, confidentiality
Source	ENISA

Table 4.20: Inadequate / absent data quality checks

Threat Category	FM, UD

Threat	Data acquisition
Description	In the data acquisition phase it is important to collect large, rich quality data from different sources, which can be difficult through various data minimization principles and data protection mechanisms.
Potential Impact	Integrity
Source	ISO

Table 4.21: Data acquisition

Threat Category	FM, UD
Threat	Bias
Description	Bias is defined as favouritism towards someone or something, which means someone or something is neglected. Biases arise from the data, which can include human cognitive, societal and statistical bias or even technical errors. Bias can manifest in different lifecycle phases of the AI system, which can affect labels, training datasets or lead to missing features/labels, data processing issues or even architectural issues.
Potential Impact	Integrity
Source	ISO

Table 4.22: Bias

Threat Category	FM, EIH, NAA
Threat	Transfer learning attack / Attacking the supply chain
Description	AI system, that are built on existing, trained models and are only fine-tuned for the specific use-case, may already be compromised and manipulated in advance.
Potential Impact	Integrity, confidentiality, availability
Source	BIML, Microsoft, MITRE

Table 4.23: Transfer learning attack / Attacking the supply chain

Threat Category	UD
Threat	Untrustworthy data sources
Description	The data quality, provenance and integrity is essential for machine learning systems, public data sources and online models can bear a risk for the trustworthiness of data.
Potential Impact	Integrity
Source	BIML

Table 4.24: Untrustworthy data sources

Threat Category	UD
Threat	Erroneous encoding
Description	Encoding problems can be introduced or aggravated. Bias in raw data processing can impact moral and ethical implications.
Potential Impact	Integrity
Source	BIML

Table 4.25: Erroneous encoding

Threat Category	NAA
Threat	Manipulated annotations
Description	Annotation tags can be manipulated to influence classification.
Potential Impact	Integrity
Source	BIML

Table 4.26: Manipulated annotations

Threat Category	UD
Threat	Overly normalizing
Description	Through removing outliers the raw data is changed, which can result in exceedingly biased datasets and possibly destroying a feature of interest.
Potential Impact	Integrity

Source	BIML

Table 4.27: Overly normalizing

Threat Category	UD
Threat	Fusion risk
Description	Data from multiple sensors can make AI more robust, but it also introduces the risk of data fusion.
Potential Impact	Integrity
Source	BIML

Table 4.28: Fusion risk

Threat Category	UD, FM
Threat	Filter information leakage
Description	Information of raw data filtration schemes can help attackers in later stages.
Potential Impact	Confidentiality, integrity
Source	BIML

Table 4.29: Filter information leakage

4.7.3 Datasets

Threat Category	FM, UD
Threat	Biased partitioning
Description	When partitioning data into training, validation and testing data care must be taken to not create biased datasets.
Potential Impact	Integrity
Source	BIML

Table 4.30: Biased partitioning

Threat Category	NAA

Threat	Adversarial partitions
Description	Attackers can control the partitioning of the dataset into training and evaluation data. This can misrepresent the reality, manipulate the system and classification.
Potential Impact	Integrity
Source	BIML

Table 4.31: Adversarial partitions

Threat Category	NAA
Threat	Backdoor / insert attacks on training datasets
Description	Special trigger patterns are part of the training dataset and therefore, targeted misclassification can take place.
Potential Impact	Integrity
Source	ENISA

Table 4.32: Backdoor / insert attacks on training datasets

Threat Category	NAA, UD
Threat	Compromising ML training - augmented data
Description	Threats regarding augmented data due to inconsistency within the training set.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.33: Compromising ML training - augmented data

Threat Category	NAA
Threat	Compromising ML training - validation data
Description	Referring to shortening of the ML model training, due to an adapted validation set.
Potential Impact	Integrity, availability

Source	ENISA

Table 4.34: Compromising ML training - validation data

Threat Category	NAA
Threat	Data poisoning
Description	By data poisoning attacks, adversaries inject erroneous / tampered / wrong data into the training or validation data in order to influence the model. The aim of this attack is to train / retrain a bad model, which is not able to classify correctly. These attacks are especially vulnerable, if the model is exposed to the Internet, like online models, that re-evaluate their model constantly. It’s important to identify possible attack vectors for poisoning attacks, following should be considered: • Tainting data from acquisition / label corruption - adversaries alter labels in a training set causing misclassification of the model. • Tainting data from open source supply chains - attackers add their own data to an open source dataset in order to manipulate the classification. • Tainting data from acquisition / chaff data - adding noise to datasets can lower the prediction accuracy of the model and cause misclassification. • Tainting data in training / label corruption - adversaries can manipulate training labels in order to misclassify malicious inputs.
Potential Impact	Integrity, availability
Source	ENISA, ISO, BIML, Microsoft, MITRE, NIST

Table 4.35: Data poisoning

Threat Category	NAA
Threat	Access Control List manipulation

Description	Group-based ACLs for datasets may fail, which can lead to privilege elevation attacks
Potential Impact	Integrity
Source	ENISA

Table 4.36: Access Control List manipulation

Threat Category	NAA
Threat	Manipulation of datasets and data transfer process
Description	Data is manipulated or tampered while being stored or processed in third-party infrastructure.
Potential Impact	Confidentiality, integrity
Source	ENISA

Table 4.37: Manipulation of datasets and data transfer process

Threat Category	NAA
Threat	Unauthorized access to datasets and data transfer process
Description	Data can be accessed unauthorized while being stored or processed in third-party infrastructure.
Potential Impact	Confidentiality, integrity
Source	ENISA

Table 4.38: Unauthorized access to datasets and data transfer process

Threat Category	FM, EIH
Threat	Stream interruption
Description	The interruption of data streams during processes, like data ingestion and training, can cause failures in AI systems.
Potential Impact	Confidentiality, integrity, availability
Source	ENISA

Table 4.39: Stream interruption

Threat Category	UD, NAA
Threat	Data leakage
Description	Data is leaking during training phase, which means adversaries can gather data and retrain a substitute model.
Potential Impact	Confidentiality, integrity
Source	ISO

Table 4.40: Data leakage

Threat Category	UD
Threat	Dissimilarity risk
Description	If the data integrity, trustworthiness and statistical distribution of the different datasets are not similar, the ML system cannot be trained properly
Potential Impact	Integrity
Source	BIML

Table 4.41: Dissimilarity risk

Threat Category	UD
Threat	Insufficient datasets
Description	Weak representation in data leads to problems in categorization, generalization and susceptibility to adverse input behaviour.
Potential Impact	Integrity
Source	BIML

Table 4.42: Insufficien datasets

4.7.4 Learning algorithm

Threat Category	NAA
Threat	Manipulation of optimization algorithm
Description	Erroneous tuning of models due to nefarious use of optimization algorithms.

Potential Impact	Availability
Source	ENISA, NIST

Table 4.43: Manipulation of optimization algorithm

Threat Category	UD
Threat	Compromising feature selection
Description	Performance degradation of feature selection algorithms by providing feature sets, that are strongly predictive only for only classes and neglect other features. Moreover, the use of single features, that predominantly contribute to predictions, can lead to inaccurate results .
Potential Impact	Integrity, availability
Source	ENISA, ISO, NIST

Table 4.44: Compromising feature selection

Threat Category	NAA, UD
Threat	Compromise of data brokers / providers
Description	Manipulation of the ML algorithm due to compromised data from brokers / providers.
Potential Impact	Integrity, Confidentiality, Availability
Source	ENISA

Table 4.45: Compromise of data brokers / providers

Threat Category	UD
Threat	Overfitting
Description	The model learned from data with too many details including noise, therefore, these models can be manipulated easily, because the system only learned to reproduce / classify the training set and is not able to generalize these to new data. This can be caused by oversensitive hyper-parameters.
Potential Impact	Integrity

Source	ISO, BIML

Table 4.46: Overfitting

Threat Category	UD
Threat	Underfitting
Description	The model learned too little details and therefore, its model predictions of the inputs are not correctly.
Potential Impact	Integrity
Source	ISO, BIML

Table 4.47: Underfitting

Threat Category	UD
Threat	Exploit-vs-explore
Description	The search space of the system has to be identified and understood, and therefore, the fitting model architecture has to be chosen.
Potential Impact	Integrity
Source	BIML

Table 4.48: Exploit-vs-explore

Threat Category	UD
Threat	Blind spots
Description	The implemented algorithm may have blind spots, that make the system vulnerable to attacks, such as adversarial examples.
Potential Impact	Integrity, confidentiality
Source	BIML

Table 4.49: Blind spots

Threat Category	UD

Threat	Oscillation
Description	The AI model can oscillate and not converge properly if, for example, it uses gradient descent in a space where the gradient is misleading.
Potential Impact	Integrity
Source	BIML

Table 4.50: Oscillation

Threat Category	NAA, UD
Threat	Hyper-parameter optimization
Description	Is the risk of selecting incorrect hyper-parameters, which influences the learning process directly and therefore, leads to prediction failures. The settings of the hyper-parameters are not always understandable, therefore, attackers can influence hyper-parameters without detection.
Potential Impact	Integrity, confidentiality
Source	ISO, BIML

Table 4.51: Hyper-parameter optimization

Threat Category	NAA, UD
Threat	Oversensitive hyper-parameters
Description	Oversensitive hyper-parameters can result in overfitting, these can happen through wrong evaluation and exploration or through an insider threat.
Potential Impact	Integrity
Source	BIML

Table 4.52: Oversensitive hyper-parameters

Threat Category	NAA, UD
Threat	Manipulated parameters
Description	Parameters can be attacked, if the aggressor manipulates the settings in a public model.

Potential Impact	Integrity
Source	BIML

Table 4.53: Manipulated parameters

4.7.5 Evaluation

Threat Category	NAA
Threat	Reduce effectiveness of AI/ML results
Description	Due to erroneous usage, there is no model maintenance and a lack of retraining procedures.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.54: Reduce effectiveness of AI/ML results

Threat Category	UD, NAA
Threat	Wrong model validation / evaluation
Description	Influenced validation of the trained model with a wrong / influenced (not representative) dataset. Or bad evaluation data can mislead researchers to think the system is not working correctly. Or the evaluation dataset can be too small or too similar.
Potential Impact	Integrity
Source	ISO, BIML

Table 4.55: Wrong model validation / evaluation

Threat Category	UD,
Threat	Problems by model updates
Description	Problems by the update process for newly acquired data, monitoring the accuracy of the model in order to correct through retraining/updating, moreover, auditing and version controls by updates are important.
Potential Impact	Integrity, confidentiality, availability

Source	ISO

Table 4.56: Problems by model updates

Threat Category	UD
Threat	Catastrophic forgetting
Description	Catastrophic forgetting happens when the system has too much overlapping information and cannot classify correctly (mainly online systems).
Potential Impact	Integrity
Source	BIML

Table 4.57: Catastrophic forgetting

Threat Category	UD
Threat	Choking on big data
Description	Choking on big data happens, when the algorithm of the online system cannot improve its performance beyond what it has learned.
Potential Impact	Integrity
Source	BIML

Table 4.58: Choking on big data

Threat Category	UD
Threat	Incomplete testing in realistic conditions
Description	Incomplete testing in realistic conditions happens when the ML system is only tested in constructed environments, which leads to a wrongly trained system, that cannot work with real environmental conditions.
Potential Impact	Integrity
Source	Microsoft

Table 4.59: Incomplete testing in realistic conditions

4.7.6 Input

Threat Category	NAA
Threat	Adversarial examples
Description	An adversarial attack is an input, that includes perturbation, which is e.g. a slightly modified image, that affects the system and can lead to misclassification.
Potential Impact	Integrity, availability
Source	ENISA, ISO, BIML, Microsoft, NIST

Table 4.60: Adversarial examples

Threat Category	NAA, UD
Threat	Dirty input
Description	The input data has not been sanitized in a controlled manner, in order to prevent adding noise to the data.
Potential Impact	Integrity
Source	BIML

Table 4.61: Dirty input

Threat Category	EIH
Threat	Controlled input stream
Description	The trained system gets manipulated input data from the attacker.
Potential Impact	Integrity
Source	BIML

Table 4.62: Controlled input stream

Threat Category	NAA, EIH
Threat	Looped input and pre-processing

Description	The model can be influenced by manipulated input, therefore, the pre-processing steps have to be replicated and input data for retraining have to be chosen with care.
Potential Impact	Integrity
Source	BIML

Table 4.63: Looped input and pre-processing

Threat Category	NAA
Threat	Reprogramming DNN
Description	Attackers are able to reprogram the ML system to perform a task, that deviates from the original intent with special crafted queries.
Potential Impact	Integrity, Availability
Source	Microsoft, MITRE

Table 4.64: Reprogramming deep neural nets

Threat Category	UA
Threat	Natural adversarial examples
Description	Natural adversarial examples are caused by inputs, that are incorrectly recognized by the system using hard negative mining.
Potential Impact	UD
Source	Microsoft

Table 4.65: Natural adversarial examples

Threat Category	NAA, UD
Threat	Common corruption
Description	Common corruption and perturbations cannot be handled by the system, like tilting, zooming, adding noise, using synonyms, adjustments to brightness and contrast.
Potential Impact	Integrity

Source	Microsoft, NIST

Table 4.66: Common corruption

Threat Category	NAA
Threat	Model evasion
Description	By these attacks the query is modified by the attacker to get the desired output. This manipulation is possible by iteratively querying and observing the model's output.
Potential Impact	Integrity, Confidentiality
Source	MITRE

Table 4.67: Model evasion

4.7.7 Model

Threat Category	NAA
Threat	ML model confidentiality
Description	Attackers are exploiting the ML model to gain information.
Potential Impact	Confidentially
Source	ENISA, BIML

Table 4.68: ML model confidentiality

Threat Category	NAA
Threat	ML model integrity manipulation
Description	Manipulation of the ML model due to manipulated parameters or bias.
Potential Impact	Integrity
Source	ENISA

Table 4.69: ML model integrity manipulation

Threat Category	NAA, EIH

Threat	Model backdoor
Description	Backdoor threats, that expose inner working, impact operation or degrade / cancel performance, for example from an AI provider. By these attacks the model execution has an unintended effect and can lead to persistent access to systems. The execution of unsafe ML models is possible due to the use of pre-trained models, which source is compromised or by pickle embedding, pickles are used in (de-)serializing a Python object structure. ML models can sometimes be stored and shared as pickles, which can contain malicious payloads.
Potential Impact	Integrity, availability, confidentiality
Source	ENISA, Microsoft, MITRE

Table 4.70: Model backdoor

Threat Category	NAA
Threat	Model poisoning
Description	The model is poisoned by the attacker to perform in a specific way, for example, an input containing a trigger causes an inference error or the legitimate model file is replaced by poisoned model file.
Potential Impact	Integrity, availability
Source	ENISA, Microsoft, MITRE

Table 4.71: Model poisoning

Threat Category	NAA, PA
Threat	Model sabotage
Description	Exploitation or physical damage of libraries and ML platforms in order to sabotage the system.
Potential Impact	Availability, integrity
Source	ENISA

Table 4.72: Model sabotage

Threat Category	NAA
Threat	Unauthorized access to models' code
Description	ML libraries and platforms are manipulated by malicious code injection in order to gain access to datasets.
Potential Impact	Confidentiality, Integrity
Source	ENISA

Table 4.73: Unauthorized access to models' code

Threat Category	UD
Threat	Erroneous configuration of models
Description	Usage of model without considering contextual factors, which can result in biases and discriminations or bad performance.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.74: Erroneous configuration of models

Threat Category	NAA, UD, EIH
Threat	Model extraction / inversion
Description	Through input/output observation classification information or data of the AI system can be stolen.
Potential Impact	Confidentiality
Source	BIML, Microsoft, MITRE, NIST

Table 4.75: Model extraction / inversion

Threat Category	UD, FM
Threat	Improper model re-use
Description	The model is re-used, but for a different purpose, and only the classification of the scheme is adapted, however, problems such as non-representative data for the use-case may occur.

Potential Impact	Integrity
Source	BIML

Table 4.76: Improper model re-use

Threat Category	NAA
Threat	Trojan
Description	A transferred model may contain a Trojan or be otherwise damaged.
Potential Impact	Confidentiality, integrity, availability
Source	BIML

Table 4.77: Trojan

Threat Category	NAA
Threat	Model inversion
Description	In this scenario the attacker can extract features, which were used to train the ML system, which can result in Membership inference attacks.
Potential Impact	Integrity, confidentiality
Source	MITRE, NIST

Table 4.78: Model inversion

4.7.8 Inference algorithm

Threat Category	NAA, UD
Threat	Compromising ML inference’s correctness - algorithms
Description	Threats that affect the accuracy and availability of the ML training algorithm.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.79: Compromising ML inference’s correctness - algorithms

Threat Category	NAA
Threat	Misclassification based on adversarial examples
Description	Manipulation of model parameters or the use of adversarial examples during inference phase to force misclassification of prediction results. Access to models and datasets is necessary.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.80: Misclassification based on adversarial examples

Threat Category	NAA, UD
Threat	Online system manipulation
Description	A system is online when it continues to learn during its use. Attackers can manipulate these systems by retraining, therefore, to mitigate this risk: data provenance, algorithm choice and system operations has to be considered. Or replace the model through a backdoor in the system.
Potential Impact	Confidentiality, integrity
Source	ENISA, BIML

Table 4.81: Online system manipulation

Threat Category	NAA, UD
Threat	Hyper-parameter optimization
Description	Is the risk of selecting incorrect hyper-parameters, which influence the learning process directly and therefore, lead to prediction failures. The settings of the hyper-parameters are not always understandable and therefore, can be influenced by attackers.
Potential Impact	Integrity
Source	ISO, BIML

Table 4.82: Hyper-parameter optimization

Threat Category	UD, NAA
Threat	Confidence scores
Description	Information about the confidence score can help an attacker to tweak the system or extract the model itself through the given feedback.
Potential Impact	Confidentiality
Source	BIML

Table 4.83: Confidence scores

Threat Category	UD
Threat	Distributional shifts
Description	Distributional shifts mean, that the system is unable to adapt to changes in the environment and therefore, predicts incorrectly.
Potential Impact	Integrity
Source	Microsoft

Table 4.84: Distributional shifts

4.7.9 Output

Threat Category	NAA
Threat	White-box, targeted or non-targeted
Description	Describes the misclassification to a specific target class or different class in general.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.85: White-box, targeted or non-targeted

Threat Category	NAA
Threat	Model stealing

Description	This threat is affecting privacy as well as security, because adversaries are able to steal the model and therefore, can replicate the inner function of it. Model stealing can be done by querying a system and using the response to train a new model.
Potential Impact	Integrity
Source	ISO, Microsoft, MITRE

Table 4.86: Model stealing

Threat Category	EIH
Threat	Output manipulation
Description	The output integrity can be attacked when an attacker interposes between the AI system and the user, therefore, output manipulation is possible.
Potential Impact	Integrity
Source	BIML

Table 4.87: Output manipulation

Threat Category	UD
Threat	Miscategorization
Description	For example adversarial examples can lead to deceptive output or overfitted models.
Potential Impact	Integrity
Source	BIML

Table 4.88: Miscategorization

Threat Category	UD, NAA
Threat	Looped output
Description	The model can be adapted, if subtle feedback loops are allowed. The output data from the model is later used as training data.
Potential Impact	Integrity

Source	BIML

Table 4.89: Looped output

Threat Category	NAA
Threat	Membership inference attack
Description	Membership inference attacks allow attackers to determine whether a particular record was part of the model's training dataset or not.
Potential Impact	Confidentiality
Source	Microsoft, MITRE, NIST

Table 4.90: Membership inference attack

4.7.10 System-wide and broad concerns

Threat Category	NAA
Threat	Hosting
Description	Hosted services have to be protected against AI-related attacks
Potential Impact	Integrity, confidentiality, availability
Source	BIML

Table 4.91: Hosting

Threat Category	UD
Threat	Exposed interests of users
Description	Accessing an AI system can expose the interests of the user to the system owner.
Potential Impact	Confidentiality
Source	BIML

Table 4.92: Exposed interests of users

Threat Category	NAA, UD

Threat	No randomness
Description	If the randomness of a system is manipulated or not given, it can influence the results of the ML systems negatively
Potential Impact	Integrity
Source	BIML

Table 4.93: No randomness

Threat Category	NAA, UD
Threat	Compromise and limit AI results
Description	This threat emerges due to involuntary or unintentionally actions, like data hiding or lack of experience.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.94: Compromise and limit AI results

Threat Category	NAA, UD, FM
Threat	Compromise of model frameworks
Description	Failure of the model framework due to misconfiguration or vulnerabilities in software, firmware and hardware.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.95: Compromise of model frameworks

Threat Category	NAA, LEG, FM
Threat	Corruption of data indexes
Description	Content of data indexes are corrupted, multiple causes are possible.
Potential Impact	Integrity, availability, confidentiality
Source	ENISA

Table 4.96: Corruption of data indexes

Threat Category	FM, OUT
Threat	(Distributed) denial of service attacks
Description	The goal of the adversaries is to reduce the availability of the system.
Potential Impact	Availability
Source	ENISA, BIML, MITRE

Table 4.97: (Distributed) denial of service attacks

Threat Category	NAA
Threat	Elevation-of-privilege
Description	Exploitation in order to get access to the AI system.
Potential Impact	Confidentiality, integrity
Source	ENISA

Table 4.98: Elevation-of-privilege

Threat Category	NAA
Threat	Insider threat
Description	The data or model is exposed on purpose, because of malicious insiders, or for example if the supervisor is malicious the algorithm can be incorrectly trained.
Potential Impact	Confidentiality, integrity, availability
Source	ENISA, BIML

Table 4.99: Insider threat

Threat Category	NAA, PA
Threat	Sabotage
Description	Involving intentionally destroying or maliciously affecting the IT system, that supports the AI application.
Potential Impact	Availability
Source	ENISA

Table 4.100: Sabotage

Threat Category	NAA
Threat	Transferability of adversarial attacks
Description	Describes adversarial examples, that can be transferred to other AI/ML applications and environments / tools like libraries and platforms.
Potential Impact	Integrity
Source	ENISA

Table 4.101: Transferability of adversarial attacks

Threat Category	UD, NAA
Threat	Bias introduced by someone in the AI lifecycle
Description	Person can be biased or their information, this can affect the trustworthiness of the AI system.
Potential Impact	Integrity
Source	ENISA

Table 4.102: Bias introduced by someone in the AI lifecycle

Threat Category	UD, LEG
Threat	Compromise privacy during data operations
Description	Data modification or erroneous handling can lead to data breaches or legal concerns.
Potential Impact	Confidentiality
Source	ENISA, ISO, BIML

Table 4.103: Compromise privacy during data operations

Threat Category	NAA
Threat	Model querying

Description	Model querying can lead to a stolen model, these kind of attacks impact the confidentiality of the program and can expose sensitive information. Model querying attacks are not phase specific, they can happen throughout the whole AI lifecycle. Moreover, the gained information can be used for profiling, sorting or classifying individuals.
Potential Impact	Confidentiality
Source	ISO, BIML, MITRE

Table 4.104: Model querying

Threat Category	UD, LEG
Threat	Disclosure of personal information
Description	Personal information can be disclosed, due to e.g. lack of data randomization, pseudonymisation etc.
Potential Impact	Confidentiality
Source	ENISA, BIML

Table 4.105: Disclosure of personal information

Threat Category	UD
Threat	Misconfiguration or mishandling of AI systems
Description	Unintentional exposed data or models impact trustworthiness and confidentiality of the AI system.
Potential Impact	Confidentiality, integrity
Source	ENISA

Table 4.106: Misconfiguration or mishandling of AI systems

Threat Category	UD, FM
Threat	ML model performance degradation

Description	The performance of an AI's system may degrade the model (reasons: governance policy, by omission or by corruption, system crashes, loss of network connectivity).
Potential Impact	Availability, confidentiality
Source	ENISA

Table 4.107: ML model performance degradation

Threat Category	LEG
Threat	Lack of data governance policies
Description	No implementation of a data governance policies although personal data are processed (GDPR).
Potential Impact	Integrity, confidentiality
Source	ENISA, BIML

Table 4.108: Lack of data governance policies

Threat Category	FM
Threat	Weak governance policies
Description	Implemented data governance policies with defective data metrics, absence of documentation and lack of adaptability.
Potential Impact	Confidentiality, integrity
Source	ENISA

Table 4.109: Weak governance policies

Threat Category	LEG
Threat	Lack of data protection compliance of third-parties
Description	Lack of data protection compliance in applications, libraries, models, etc. of third-parties.
Potential Impact	Confidentiality
Source	ENISA

Table 4.110: Lack of data protection compliance of third-parties

Threat Category	LEG
Threat	SLA breach
Description	Service Level Agreement (SLA) breaches can lead to degradation of performance or unavailability of AI systems.
Potential Impact	Availability
Source	ENISA

Table 4.111: SLA breach

Threat Category	LEG
Threat	Vendor lock-in
Description	Is the risk on relying on only one third party provider without alternatives / backup.
Potential Impact	Availability
Source	ENISA

Table 4.112: Vendor lock-in

Threat Category	FM, PA
Threat	Errors or timely restrictions due to non-reliable data infrastructure
Description	Describes the data and computational exposure and/or inadequate capacity, that may expose data and compromise privacy preservation.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.113: Errors or timely restrictions due to non-reliable data infrastructure

Threat Category	FM
Threat	Lack of documentation

Description	Documentation of the algorithm parameter choice, why alternatives where discarded, moreover, model under- and overfitting, parameter and design choices should be explained.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.114: Lack of documentation

Threat Category	FM
Threat	Poor resource planning
Description	Lack of adequate computational resources (storage capacity, transmission speed, computational power) can compromise proper functioning of AI systems.
Potential Impact	Integrity, availability
Source	ENISA

Table 4.115: Poor resource planning

Threat Category	EIH
Threat	Data inference
Description	This can be exploited by data and model providers and lead to inference of data.
Potential Impact	Confidentiality
Source	ENISA

Table 4.116: Data inference

Threat Category	FM, OUT
Threat	Third-party provider failure
Description	Failures of third-party providers may lead to unavailability of AI systems or improper / delayed operation.
Potential Impact	Availability, Confidentiality

Source	ENISA

Table 4.117: Third-party provider failure

Threat Category	NAA, EIH
Threat	Malicious third-party
Description	By this attack scenario the training process can be fully or partially outsourced to a malicious party, which provides a backdoored component. Therefore, the ML system results can be modified by the adversary party.
Potential Impact	Confidentiality, Integrity, Availability
Source	Microsoft

Table 4.118: Malicious third-party

Threat Category	EIH, NAA
Threat	Data theft
Description	This threat may manifest during the transportation of data, during data ingestion and while accessing data storage, data may be intercepted and stole.
Potential Impact	Confidentiality, integrity
Source	ENISA

Table 4.119: Data theft

Threat Category	EIH
Threat	Model disclosure
Description	Leaking information about trained / tuned models, internal parameters or other model settings.
Potential Impact	Confidentiality
Source	ENISA

Table 4.120: Model disclosure

Threat Category	EIH
Threat	Weak encryption
Description	This threat refers to potential eavesdropping of data or hijacking of communications in the case of data transfers/storage/processing. This can expose datasets and sensitive information.
Potential Impact	Confidentiality, integrity
Source	ENISA, BIML

Table 4.121: Weak encryption

Threat Category	PA
Threat	Communication networks tampering
Description	Tampered communication networks can lead to unavailable AI systems, moreover, through side-channel attacks sensitive information can be exposed.
Potential Impact	Confidentiality, availability
Source	ENISA

Table 4.122: Communication networks tampering

Threat Category	PA
Threat	Infrastructure / system physical attacks
Description	Physical attacks against infrastructure, that support AI systems can lead to performance issues and unavailability.
Potential Impact	Availability
Source	ENISA

Table 4.123: Infrastructure / system physical attacks

Threat Category	OUT
Threat	Communication networks outages

Description	This outages may adversely influence the performance and operation of AI systems.
Potential Impact	Availability
Source	ENISA, BIML

Table 4.124: Communication networks outages

Threat Category	OUT
Threat	Infrastructure / system outages
Description	Outages of infrastructure may lead to performance issues and unavailability of the AI system and its critical functions.
Potential Impact	Availability
Source	ENISA, BIML

Table 4.125: Infrastructure / system outages

Threat Category	DIS
Threat	Environmental phenomena (heating, cooling, climate change)
Description	These phenomena may adversely influence the operation of IT infrastructure and hardware systems, that support AI systems.
Potential Impact	Availability
Source	ENISA

Table 4.126: Environmental phenomena (heating, cooling, climate change)

Threat Category	DIS
Threat	Natural disasters (earthquake, flood, fire, etc.)
Description	These disasters may lead to unavailability or destruction of the IT infrastructures and hardware supporting AI systems.
Potential Impact	Availability
Source	ENISA

Table 4.127: Natural disasters (earthquake, flood, fire, etc.)

Threat Category	NAA, UD
Threat	Software bugs / attacks
Description	By these attacks adversaries manipulate traditional software vulnerabilities, such as buffer overflows, erroneous memory access / inputs / outputs, data and control flows, race conditions, deadlocks and so on.
Potential Impact	Confidentiality, Integrity, Availability
Source	ISO, MITRE, Microsoft

Table 4.128: Software bugs / attacks

Threat Category	UD
Threat	Reward hacking
Description	Reward hacking can result in data discrepancies between the specified reward and the true intended reward due to reinforcement learning systems.
Potential Impact	Integrity
Source	Microsoft

Table 4.129: Reward hacking

Threat Category	UD
Threat	Side effects
Description	The reinforcement learning system interferes with the environment, while trying to achieve its goal. For example a cleaning robot, which should vacuum all edges, knocks over an unstable table with a vase, because it is trying to clean everything.
Potential Impact	Integrity, Availability, Confidentiality
Source	Microsoft

Table 4.130: Side effects

Threat Category	FM
Threat	Hardware bugs

Description	Different attacks can affect the integrity and confidentiality of data, therefore, traditional mechanisms like memory integrity and trusted platform modules are necessary. However, also new techniques have to be implemented in order to secure AI systems, like enforcing the program-intended logic by runtime, e.g. a control-flow attack can circumvent AI model inference and therefore, cause invalid trained AI systems. Moreover, mechanisms to prevent memory safety bugs are important, logic flaws in programs can lead to buffer overflows or use-after-free can lead to faulty operations in AI systems. Therefore, threats to complex device models, like hardware accelerators, device spoofing, runtime memory remapping and man-in-the middle attacks, need to be considered.
Potential Impact	Integrity, confidentiality, availability
Source	ISO, MITRE

Table 4.131: Hardware bugs

Threat Category	LEG
Threat	Legal issues with data
Description	Data can be protected by copyright or even contain illegal data, additionally in order to fulfil the General Data Protection Regulation (GDPR) requirements a deletion process for data has to be considered.
Potential Impact	Confidentiality, integrity
Source	ENISA, BIML, ISO

Table 4.132: Legal issues with data

Threat Category	FM, UD
Threat	Insecure storage
Description	By these attacks adversaries exploit insecure storage mechanisms in order to extract data of the ML system, therefore, data access should be managed.
Potential Impact	Integrity, confidentiality
Source	BIML, MITRE, NIST

Table 4.133: Insecure storage

Threat Category	UD
Threat	Overconfidence
Description	The user may rely too much on the AI system's results, even though the system's data comes from an AI application with flawed behaviour and is treated as high confidence data. The overconfidence in ML systems is related to the fact, that these systems are still insufficiently understood and described.
Potential Impact	Integrity
Source	BIML

Table 4.134: Overconfidence

Threat Category	UD
Threat	Cry wolf
Description	When the ML subsystem generates too many alarms within a larger system, the subsystem may be ignored. Regular false alarms can cause error messages to be ignored.
Potential Impact	Integrity, confidentiality
Source	BIML

Table 4.135: Cry wolf

Threat Category	UD
Threat	API encoding
Description	The error handling of incorrect encoded data for Application Programming Interface (API) system is faulty. General security measures for APIs have to be implemented in order to secure them.
Potential Impact	Integrity, Confidentiality
Source	BIML

Table 4.136: API encoding

Threat Category	NAA
Threat	Acquire information
Description	Attackers can leverage information, that is publicly available or open source intelligence (OSINT) about the organisation and how AI is used in a system, which helps adversaries to tailor the attacks to the target.
Potential Impact	Confidentiality
Source	MITRE

Table 4.137: Acquire information

Threat Category	NAA
Threat	ML model discovery
Description	Adversaries attempt to identify information from the ML model, that may be useful, e.g., for further acquisition, exfiltration, disruption or tailored attacks targeting the AI system. It is split into two categories • Reveal ML ontology - specific components are already known, mostly white-box or grey-box based. • Reveal ML model family - specifics are unknown, analysing query-results or publicly available information, that leak for example the underlying algorithm, black-box based.
Potential Impact	Confidentiality
Source	MITRE

Table 4.138: ML model discovery

Threat Category	NAA
Threat	Gathering datasets

Description	Attackers can collect datasets similar to the one used for training the ML system. Therefore, adversaries can replicate functions or enable other attacks like evasion attacks.
Potential Impact	Integrity, confidentiality
Source	MITRE

Table 4.139: Gathering datasets

Threat Category	NAA
Threat	Model replication
Description	Model replication may be possible by training a shadow model, by leveraging pre-trained weights or by exploiting the ML system's API.
Potential Impact	Confidentiality
Source	MITRE

Table 4.140: Model replication

Threat Category	NAA, EIH
Threat	User accounts
Description	Valid accounts can be obtained by adversaries for gaining access to ML systems, which can be hard to detect. Attackers manipulate accounts to gain persistence to ML systems.
Potential Impact	Confidentiality, integrity, availability
Source	MITRE

Table 4.141: User accounts

Threat Category	NAA, EIH
Threat	Phishing
Description	Phishing can be used to gain access or sensitive information to ML systems.
Potential Impact	Integrity, confidentiality
Source	MITRE

Table 4.142: Phishing

Threat Category	NAA, EIH
Threat	Trusted relationship attack
Description	Trusted relationship attacks describe the access from attackers due to trusted third party relationship, because of an unsecure connection or elevated permission due to maintenance work.
Potential Impact	Integrity, confidentiality
Source	MITRE

Table 4.143: Trusted relationship attack

Threat Category	NAA, EIH, FM
Threat	Execution via API
Description	There are three possible interactions: building an offline copy of the model (model stealing / replication), online attacks (model inversion, online evasion, membership inference) or tainting the training data of the model via feedback loop.
Potential Impact	Integrity
Source	MITRE

Table 4.144: Execution via API

Threat Category	FM, NAA
Threat	Implant container imager
Description	By these attacks the cloud container images contain malicious code, that grants attackers access.
Potential Impact	Integrity, confidentiality, availability
Source	MITRE

Table 4.145: Implant container imager

Threat Category	NAA
Threat	Evasion attack
Description	By evasion attacks the ML classifiers are not able to identify the data sample. By offline evasion, the attacker has a copy of the online model and can test the classification of the data input, by online evasion, the live ML model is tested.
Potential Impact	Integrity, confidentiality
Source	MITRE

Table 4.146: Evasion attack

Threat Category	NAA
Threat	Exfiltrate training data
Description	Adversaries are able to exfiltrate private information from the ML models. For example by the membership inference attack the adversary queries the inference API of the ML system and obtains private information. By the ML model inversion the training data of the system can be reconstructed by analysing the confidence scores of the inference API.
Potential Impact	Confidentiality, integrity
Source	MITRE, NIST

Table 4.147: Exfiltrate training data

Threat Category	NAA
Threat	Defacement
Description	Adversaries (re)train the ML system with data inputs for fun. The cause of these attacks lies in the feedback loop of the ML system, which is exploited.
Potential Impact	Confidentiality
Source	MITRE

Table 4.148: Defacement

Threat Category	LEG
Threat	Stolen intellectual property
Description	Intellectual property may be stolen by model replication or model stealing attacks.
Potential Impact	Confidentiality
Source	MITRE

Table 4.149: Stolen intellectual property

Threat Category	NAA
Threat	Data encrypted for impact defacement
Description	Ransomware can encrypt the ML system or make its data inaccessible permanently.
Potential Impact	Availability, integrity, confidentiality
Source	MITRE

Table 4.150: Data encrypted for impact defacement

Threat Category	NAA, OUT
Threat	Stop system - shutdown / reboot
Description	Adversaries may stop the system through a shutdown or reboot in order to interrupt access or destruct the ML system.
Potential Impact	Availability
Source	MITRE

Table 4.151: Stop system - shutdown / reboot

Threat Category	LEG, FM
Threat	Weak requirements analysis
Description	AI requirements may fail, if they are considered without context (isolated), therefore, they need to consider the circumstances and adapt where necessary.

Potential Impact	Availability, Integrity, Confidentiality
Source	ENISA

Table 4.152: Weak requirements analysis

Threat Category	FM
Threat	Compromising AI application viability
Description	Referring to a lack of understanding of what AI/ML are and how to succeed with the business models.
Potential Impact	Availability
Source	ENISA

Table 4.153: Compromising AI application viability

Threat Category	UD
Threat	Opaqueness
Description	Opaqueness means, that the transparency of the decision is not given, which can have the following causes: • the AI system itself, which its decision making • if data sources or data are not transparent • AI system based on organizational context, which is not comprehensible by externals
Potential Impact	Integrity
Source	ISO

Table 4.154: Opaqueness

Threat Category	UD, FM
Threat	Unpredictability

Description	The predictability of an AI system is an important factor for its acceptability. Because trust in the technology is often based on the predictability. However, scenarios, that differ a lot from the test environment can lead to unpredictable behaviour, without override mechanisms, like fail-safes.
Potential Impact	Integrity
Source	ISO, BIML

Table 4.155: Unpredictability

Threat Category	UD, FM
Threat	Irreproducibility
Description	When ML models and results cannot be reproduced, the classification of these systems cannot be trusted.
Potential Impact	Integrity
Source	BIML

Table 4.156: Irreproducibility

Threat Category	UD, FM
Threat	Transparency
Description	The decisions by the systems should be transparent, otherwise attacks can stay undetected easily.
Potential Impact	Integrity
Source	BIML

Table 4.157: Transparency

Threat Category	UD, NAA, FM
Threat	Loss of public image

Description	Because some ML systems predict incorrectly / biased (racist, xenophobic, etc.), the confidence in all kind of ML systems is lost. Although these systems operate much more effectively and may be capable of a decision process beyond human capability. One misbehaving system can erode the trust in all AI systems.
Potential Impact	Integrity
Source	BIML

Table 4.158: Loss of public image

4.8 Likelihood

Moreover, the likelihood of the attacks has to be considered, which is, for example, influenced by services, that are remotely available and application, that are public facing. However, also the attackers' motivation plays a critical part and influences the likelihood of an attack.

5 Adversarial attacks in the wild

In this section real world threats for AI are summarized by representative examples. The main focus is on *real* world examples, because most attacks proposed in the AI research do not work there. The main target of these attacks is to fool the sensors and the model behind the systems. The physical attacks will be outlined using adversarial attacks on road traffic, because its robustness is currently highly discussed, especially by media outlets [38]–[40]. Moreover, the context of attacks in the cyber security domain will be discussed, as this is an interesting aspect, since there are already adversaries, who attack protection mechanisms, that rely on AI [41].

5.1 Adversarial attacks in road traffic

The perceptional systems in autonomous cars combine inputs from multiple sensors in order to make decisions based on them. This can result in a relatively easy attack vector, because sensors can be deceived and are therefore, the main target to exploit these systems. Moreover, they are the main component in controlling autonomous vehicles [42] and the surrounding/setup (the input) can be influenced comparably easily by attackers in a real environment [43].

5.1.1 Phantom Attacks of the Advanced Driver Assistance Systems

Autonomous cars were products of science fiction movies, but after years of development and research these Advanced Driver Assistance Systems (ADAS) can now be used to support or even replace manual driving due to AI models [44], [45]. These AI models process data from multiple sensors in real-time to steer the vehicle, recognize obstacles and traffic signs and trigger alerts, when the car is off track. These systems rely mainly on image input and therefore, their robustness against adversarial ML attacks has been tested in various research [46]–[51]. Study results have increased our collective attention on AI systems and improved the understanding of their limits. Moreover, the research highlighted the importance of robust models, as unsafe systems in this context can endanger human lives.

Split-second phantom attacks are investigated, which is an adversarial attack, that causes Advanced Driver Assistance Systems (ADAS) to treat depthless objects, which appear for a few milliseconds, as real obstacles/objects. The ADAS investigated are from the Tesla Model X (HW 2.5 and HW 3) and Mobileye 630. The split-second phantom attacks can be projected on a digital billboard or created by a projector by remote attackers. These images are 3D objects, for example of road signs, which causes the Tesla to halt the car in the middle of the road and Mobileye to issue false notifications. Moreover, the projection can cause Tesla's autopilot to stop, when a pedestrian is projected on the road. The attack is possible due to perceptual weakness in the dataset and its inexperience to check for fake obstacles, due to the training with its primary focus on recognizing and detecting obstacles. Moreover, these algorithms do not consider following perceptual aspects:

- **Context** - the location of the object and its local context within the frame are not considered.
- **Colour** - the colour of the object is not considered. For instance the grey background is not alarming to the algorithm.
- **Texture** - the texture of the object is also not taken into account, for example is the projected traffic sign composed of 25% of the pixels from an original sign.

Furthermore, there is a disagreement between the video camera sensor and other depth sensors, because pedestrian detection via radar becomes unreliable after a range of 10 meters in real scenarios, due to reflections from other objects [52]. At some angles the car can only rely on input from the cameras, because other sensors do not cover these areas. Therefore, the algorithm trusts more in video cameras for pedestrian detection, as it has no possibility to validate the images from the cameras. Moreover, it is a common practice, that stationary objects identified by radar are ignored by the car, because it focuses mainly on moving objects [53]. For example to keep the distance between vehicles. These hurdles result in a known problem for ADAS, which has to be solved in real-time, therefore, the *safety first* approach is implemented in autonomous driving [54] and these phantoms are treated as real obstacles.

This means split-second phantoms can be created by using projectors on drones remotely or by embedding objects into digital advertisement alongside the road. They leave no physical forensic evidence about the attack and the attacker, these attacks do not require complex manipulation of the AI systems and they target the ADAS obstacle detection system.

The researches propose a countermeasure: *GhostBusters*, which is a lightweight CNN, that assess the authenticity of objects based on their context, depth, surface and light [55].

This attack is an example for **adversarial examples** Table 4.60, **lack of sufficient representation in data** Table 4.5 and **blinded and calibrated sensors** Table 4.11.

5.1.2 Automated Lane Centering

This technology automatically steers the vehicle, in order to keep it in the centre of the road. The Automated Lane Centering (ALC) enhances the convenience and driving experience, however the system is critical for safety and security, because it influences the steering decision of the human driver. If the system makes incorrect decisions and the driver's reaction time is too slow, incidents such as driving off the road or colliding with other vehicles can occur. Therefore, the most critical performance of an Automated Lane Centering (ALC) system is the lane detection, which is performed by the front camera of the car. For this task DNN scores the highest accuracy for lane detection. This technology is nowadays used by Tesla's Autopilot. However, recent research shows, that DNN can be exploited through physical adversarial attacks, such as malicious stickers on traffic signs [46], [47], [51], [56]. Therefore, in the experiment stickers for the road itself were crafted and tested in secure settings, since such experiments are illegal on public streets.

It is important, that the attack is faster than the reaction time of the driver, multiple camera frames are attacked or otherwise the impact to the driving lane would be too little. Therefore, the goal of this setting is to influence the shape of the lane line curves, with similar work, that also tries to change the final steering angle decisions [57]–[60].

These challenges led to the design of the *dirty road patch*, which is an example of an **adversarial example** Table 4.60, which can appear legitimate. The deployment is easy, due to its design and standards procedures of repairing roads with such patches, and the appearance resembles normal roads for the human eye. In addition, the polluted roads are not classified as scenarios, that the ALC cannot process.

Motion model based input generation is a difficulty, that has to be overcome for the success of the attack. The frames of the road have to be dynamically updated and in order to create these updates as efficiently as possible, vehicle motion models and perspective transformation need to create these camera frame updates synthetically. An optimization-based approach similar to the work of Eykholt et al. [47] and Szegedy et al. [14] was used in order to generate these malicious and effective patches.

Due to ethical factors the attack was mainly tested in an environmental setup, nevertheless, the results show, that this attack is highly effective with a 97.5% success rate and a success time of 0.903 seconds, which is considerably lower than an average driver reaction time with 2.5 seconds [61]. Moreover, the attack is robust according to different model designs, stealthiness and lighting conditions.

However, the safety impact was also tested with a real vehicle, a Toyota 2019 Camry, with OpenPilot 0.7.4, which supports different assistance features such as Lane Departure Warnings (LDW), Adaptive Cruise Control (ACC), Forward Collision Warning (FCW) and Automatic Emergency Braking (AEB), OpenPilot is open source [62]. The test environment was a rarely-used dead-end road with a double-yellow line in the middle, which can only be used for U-turns. The driving speed was 45 km/h (28 mph), which is the minimum speed for using OpenPilot on the vehicle. Carbon boxes as obstacles were placed outside the current lane in order to simulate barriers. The attack was tested ten times and concluded with a collusion rate of 100%, where front and side collisions were considered equally. The other assistance features (LDW, ACC, FCW, AEB) were not able to prevent the *accidents* caused by the attack on the ALC system. LDW, ACC and AEB were not triggered at all. The FCW was triggered five times out of ten, however, it only warned the driver and could not prevent the collision. The warning itself occurred on average only 0.46 seconds before the crash, which is not enough time for the driver's reaction time. In conclusion, it is highly possible, that this attack can be used in the wild [63].

5.1.3 Lidar perception modules in vehicles

In Yang et al. [42] an attack vector on Lidar sensor and their DL models in autonomous vehicles are analysed. Small roadside objects are sometimes incorrectly classified as vehicles through the on-board AI. Inspired by this phenomena the researchers attacked these sensors, as well as the detection algorithm. The researchers tested the DL algorithm in white-box and black-box scenarios, which describe the existing knowledge of possibly adversaries. In the white-box scenario the attacker has access to the DL model of the target and in the black-box attack a genetic-evolving algorithm was used in order to generate adversarial objects. The approach was to create adversarial point clouds in simulated environments and to attack the autonomous vehicles in the physical world and influence the decision making of theAI.

The camera sensors and Lidar sensors provide input for the decision process, however, the camera's high-resolution is limited to two dimensional images, which provides only limited information about the exact distance and location of surrounding objects. Lidar sensors capture three dimensional information and present it as coordinates in point clouds. This information are obtained through infra-red lasers, whose return time is measured in order to derive the distance of objects. Moreover, various DL algorithms can be applied to analyse the obtained 3D point data captured by the sensor [64]–[66].

An attack algorithm was introduced, which can generate manufacturable 3D objects, that can influence Lidar's learning models by creating an adversarial point cloud. Its first steps are to obtain purely digital adversarial point clouds. Then the adversaries create an adversarial watertight mesh from these point clouds. The impact is measured through a second simulated Lidar, that checks, if the derived point clouds fulfil the attackers' goals. Subsequently, a 3D print of this mesh is created, in order to prove, that the concept of this attack is feasible in both the physical and digital domains. In the physical world the printed, physical objects is used in road tests and in an experimental setup indoors. Moreover, these adversarial objects are imported into a virtual map to evaluate the impact on autonomous vehicle control in this environment. An Apollo controlled vehicle is impacted by the adversarial object, which illustrates the existing risk of placing these objects on the roadside. The misclassification rate of this attacks lies between 83.7 - 88.17 and in indoor scenarios between 86.6 - 92.6%, which shows, that the success rate of this object is high.

Even when 3D adversarial example defences are employed to the systems, the attack remains effective. Therefore, the researchers introduce a detection mechanism, that can prevent this type of attacks by inspecting and considering the different physical properties between the adversarial and normal points.

This attack is an example for **insider threat** Table 4.99 and **adversarial examples** Table 4.60 and symbolizes a possible risk, when the **model extraction** Table 4.75 was successful.

5.1.4 Traffic sign

One of the most well known adversarial attacks, is the attack on the classification of traffic signs by intelligent transportation systems. DNNs are vulnerable to **adversarial examples** Table 4.60, even small-sized perturbations added to the input can disturb the classification process of the algorithm. These systems are frequently used, e.g. by autonomous cars, driving assistance, and are a safety-critical part in the decision process of the DNNs, which means, that adversarial attacks are therefore able to mislead systems and endanger human lives. The researchers introduced an attack algorithm called *Robust Physical Perturbations (RP2)*, which generates visual adversarial perturbations, that are robust under different physical situations. The proposed algorithm achieves a misclassification rate of 84.8% in real-world cases against standard road sign classifiers under various environmental conditions. Moreover, the researchers propose a two-staged testing method for robust physical adversarial examples consisting of lab and field tests.

The perturbation itself is done in form of black and white stickers, that are placed on a real stop sign, which was selected because of its importance in the road traffic [67].

5.2 Adversarial attacks on cybersecurity systems

AI-based systems find more and more use in the cyber security domain, from malware detection to Network Intrusion Detection System (NIDS) and much more. However, even systems, whose main purpose is to secure systems, are vulnerable to adversarial attacks. The cyber security domain seems more attractive to adversarial attackers, because the attackers already exist and other domains do not have real adversaries, with few exceptions like e.g. terrorists. On the other hand, in the cyber security context attackers exist with targets and goals, like ransomware developers, which goal is to evade anti-malware tools or phishing mail senders, that are trying to evade classification as phishing [68].

5.2.1 Malware Detection and Classification

Next Generation Antivirus (NGAV), like CrowdStrike, SentinelOne and Microsoft ATP, are based on DL and ML models additionally to signatures and heuristics, which are using an algorithm, that detects *malicious* behaviour in run-time. However, these features come with the disadvantages of being vulnerable to adversarial attacks.

The malware classification can either take place with static features gathered by analysing the program itself or with dynamic features by collecting data during code execution. The disadvantage of static code analysis is, that the code can change during execution, e.g. through various obfuscation techniques, polymorphisms and packing. Moreover, static analyses do not identify file-less attacks, e.g. through code injection and process hollowing. That is where dynamic analysis with NGAV products can detect malicious code and attacks. One dynamic feature, that can only be analysed and collected during execution are API / system calls, if they are made to the operation system itself. These system calls are harder to obfuscate during runtime without affecting the functionality and typical behaviour of the software [41].

DNN Malware Classifiers

Rosenberg et al. [41], [69] research focused on bypassing NGAV products without reversing them and generating Portable Executable (PE) malware, which is able to bypass NGAV products. The adversarial sample was generated through an explainability algorithm, which identifies all important features for the specific sample. Thereafter, those features (like checksums and timestamps) are changed feature-by-feature. Moreover, the attack is transferable, which means the attack [41] can be applied to NGAV with unknown classifiers. Although NGAV products use different classifiers with different feature subsets and are trained on different datasets, they still identify/classify a similar subset of important features. Based on this concept, Rosenberg et al. conducted an attack on a publicly available classifier and generated a malicious PE file, which evades classifiers and therefore, also commercial NGAV solutions.

In the research, the EMBER dataset [70] and PE structural features were used to train a substitute DNN model. Moreover, an explainability machine learning algorithm was used to identify features with high impact on the malware classification, that could be easily modified without harming the functionality of the program. These malicious programs were afterwards tested in a grey-box inference attack and they bypassed the substituted model and the target GBDT classifier, which was based on a different model. Therefore, these attacks are likely to succeed with actual, real-life NGAV. Nevertheless, their main limitation is, that this attack cannot be fully automated, due to the fact that the feature selection has to be conducted manually. The attack consists of **model extraction** Table 4.75, which makes the identification of the classification features possible, moreover the attack is **transferable** Table 4.101.

5.2.2 Network Intrusion Detection System

A common intrusion detection system working with AI is NIDS, which monitors the network traffic from a strategic point of the network and can be either a device or software. When this system detects malicious activities, events / alarms are generated.

Kuppa et al. [71] proposed a new black-box attack on deep anomaly detectors, which uses a manifold approximation algorithm [72] for query reduction and generates **adversarial samples** Table 4.60 using spherical local subspaces. In other words the researchers find **various data distributions in the labelled data** Table 4.19. The attack step itself limits the input distortion and KL divergence.

The research group evaluated seven state-of-the-art anomaly detectors, which are based on different underlying systems / models:

- Deep Autoencoding Gaussian Mixture Model [73]
- Autoencoder, a plain autoencoder, which was trained by the researchers with five fully-connected layers
- AnoGAN [74]
- Adversarially Learned Anomaly Detection [75]
- Deep Support Vector Data Description [76]
- One Class Support Vector Machines [77]
- Isolation Forests [78]

Therefore, it can be concluded, that the new attack is transferable to different NIDS, because these models are based on different classifiers. However, all classifiers were trained on similar dataset [79]. The threat of **transferability** is described in Table 4.101.

5.2.3 Spam Filters

Spam filters are widely used, because they classify incoming messages into legitimate (also called ham) or unsolicited (also called spam) messages. These filters were one of the first security protection mechanisms, that used AI and also one of the first to be targeted by attackers. Some attacks will be described in the following.

A real life example is the e-mail filter protection from Proofpoint, which was reverse engineered through a **model leakage problem** Table 4.120. The e-mail **header scores** Table 4.83 were used to identify the classifiers, that were making the decisions [43], [80].

Kuleshov et al. [81] implemented a generalized black-box inference attack, which was effective against several natural language processing classifiers. The notation of an adversarial example in this setting was formalized by the researchers. Moreover, algorithms, that construct such examples, were described. These adversarial perturbations can be applied in different domains, like spam filtering, fake news detection and sentiment analysis, and models, like CNN and Recurrent Neural Network (RNN) and linear classifiers. In order to avoid classification of the spam mail 10-30% of words in a sentence are replaced by synonyms without changing the meaning, which is called the greedy attack. Through the modification up to 90% of adversarial examples were not classified as spam / fake news in the conducted research. For example, the Naive Bayes classifiers detected 96 % of the original fake news examples and 0% of the modified examples, Long short-term memory (LSTM) and deep character-level convolutional networks (DCNN) were also evaluated. Furthermore, these perturbations are transferable across models to a certain degree.

Lei et al. [82] proposed a similar attack, which preserves the original syntax and semantics of the original message by using a joint sentence and word paraphrasing technique. The researchers tested LSTM and 1DCNN, which were trained on the same datasets used by Kuleshov et al. [81]

These attacks are possible through **common corruptions** Table 4.66 by using synonyms, which can also be **adversarial examples** Table 4.60 and are **transferable to other spam filter programs** Table 4.101

6 Trustworthy Artificial Intelligence

In reality it is not feasible to protect a system from every possible attack. Therefore, prioritization and guidelines for secure development and operation are necessary. One approach for the design and development of secure systems, is to use guiding principles, which lead to an improved security against unknown future attacks and mitigates the *attack-of-the-day* problem [17]. Guiding principles for AI systems can be found under the term *trustworthy AI*, however, there are many approaches, which describe different aspects. Some of them are outlined and compared in the following subchapters.

6.1 NIST - Four Principles of explainable AI

The National Institute of Standards and Technology (NIST) is a federal agency of the United States and is responsible for standardisation processes in technology. NIST [83] defined five principles in order to secure AI systems:

- **Explainability** - Systems should be able to provide evidence/justification for all outputs.
- **Understandability** - These explanations should be understandable for all users.
- **Accuracy** - The explanations of AI systems should be correctly (accurate).
- **Knowledge limits** - Systems should be operated only when there is sufficient confidence in their results, and they should only be used for their intended purpose.

6.2 High-level expert group on AI

The High-level expert group (HLEG) group of the European commission defined three principles, that an AI system should fulfil throughout the entire lifecycle [84]:

- **Lawful** - Compliance with all applicable regulations and laws should be ensured.
- **Ethical** - Compliance with all ethical principles and values should be ensured.
- **Robust** - Robustness from a technical and social perspective should be ensured, because AI systems can cause unintentional damage.

Moreover, the expert group [84] stated seven points in order to realise trustworthy AI, which are:

1. **Human agency and oversight** - AI systems should not restrict human autonomy, but empower people. They should enable social equality by supporting human agencies and fundamental rights. Proper oversight mechanisms through humans should be ensured.
2. **Technical robustness and safety** - AI systems should be secure and resilient to attacks, errors and inconsistencies during all lifecycle phases. A fall back plan and the general safety should be ensured, predictions of the system should be accurate, reliable and reproducible.
3. **Privacy and data governance** - AI systems should respect the privacy of individuals, therefore, adequate data governance mechanisms should be in place in order to ensure the quality, integrity, and confidentiality of datasets.
4. **Transparency** - AI systems' predictions should be transparent, which can be achieved through traceability and explainability. Interactions with AI systems should be visible to users and not hidden.
5. **Diversity, non-discrimination and fairness** - Procedures to avoid unfair bias should be established. The usability and universal design of AI systems should be ensured. Stakeholders should participate and support the AI system throughout its lifecycle.
6. **Societal and environmental well-being** - AI systems should be sustainable and environmentally friendly. Furthermore, they should only be used for beneficial social change and their effects on institutions as well as democracy should be regularly assessed.
7. **Accountability** - Mechanisms should be implemented to ensure the responsibility and accountability of AI systems and their outcome. AI systems should be auditable, moreover, negative impact should be reported and minimised, trade-offs should be made in a rational and methodical manner, and adequate redress should be ensured.

6.3 Trustworthy AI Development

The main focus of trustworthy AI from Brundage et al. [4] is based on the main components of AI systems and their development process, which are:

- **Institutional mechanisms** - play a key role in the verification of AI development claims, since the development of AI itself is still human-driven. The main focus points are **third party auditing**, in order to substitute self-assessments, **red teaming exercises** to consider misuse cases. **Bias and safety bounties** to appeal people to review and report errors in AI systems. Additionally **sharing of AI incidents**, in order to improve these AI systems and the understanding of their behaviour during incidents.

- **Software mechanisms** - can be integrated to foster a better understanding and overview of specific AI systems and their properties. The big bullet points are **audit trails** in order to enable accountability by capturing critical information about the deployment and development process, **interpretability** to enable an understanding and scrutiny of characteristics of AI systems', and to implement **privacy-preserving ML**.
- **Hardware mechanisms** - play a fundamental role for privacy and security claims, creating transparency about the usage of organization's resources. The main focus points of these mechanisms are the **usage of secure hardware for ML**, in order to ensure privacy and security requirements, **high-precision compute measurements**, to compare claims about usage of computing power, and **supporting research** in order to enable the academics outside the public sector to asses claims about large-scale AI systems.

6.3.1 BIML - Security Principles and Machine Learning

BIML was founded in 2019 with the mission to explore security implications, which are built into ML systems. BIML has derived its principles for the development of AI from the principles for the development of secure software. The focus on the main objectives in order to design secure AI systems. These ten principles provide a guideline for the implementation of secure ML systems, but due to the complexity they are neither complete nor comprehensive. In the following the BIML principles are described in detail [17]:

1. **Secure the Weakest Link** - The security of the systems is like a chain, if one component can be easily targeted, then the whole system can be vulnerable. Like in all other security related areas it is important to secure the weakest link in the chain. For this reason, risk and threat analyses are important to identify weak links.
2. **Practice defence in depth** - The concept of defence in depth consists of multi-layered defence strategies to manage risks, in order to secure systems even if one layer fails.
3. **Fail securely** - Failures should always be planned for, as they are unavoidable. Therefore, if a system fails, it should fail securely.
4. **Follow the principle of least privilege** - This principle describes permission systems, where all users have only the minimum access required to perform their daily operations or where specific permissions are only granted for the minimum period of time.
5. **Compartmentalize** - In this context this means to divide the system into its components and to enforce security measures individually. By this concept the individual components can be logically

separated and structured, for example, the principle of least privilege can be enforced for each component.

6. **Keep it simple** - Due to the complexity of the software, there is a risk, that it is inadequately implemented or poorly designed. Therefore, it is important to keep the software as simple as possible to avoid problems, this concept is called *keep it simple, stupid* (short KISS).
7. **Promote privacy** - Protecting privacy can be difficult, because attackers can try to extract data from the system. A possible solution to improve privacy in AI settings is to use differential privacy, this concept sets privacy restrictions, meaning, e.g., an individual's patient record never has too much influence on the dataset, which is used to train the system. If the data of the ML system is sensitive, a protection concept should be in place.
8. **Remember that hiding secrets is hard** - This principle describes the difficulty of keeping secrets in order to maintain security, for example, keys, personal data, algorithms, model, hyper-parameters and configuration values need to be protected. However, keeping information secret should not be the only strategy to secure the system.
9. **Be reluctant to trust** - AI systems rely on external sources, e.g., for data and their computation. But some questions arise about data collection: Can the data fed to the system be trusted? Is the collector trustworthy? Blind trust would make the ML systems vulnerable to security risks and should be avoided. Moreover, external tools and functions should be evaluated regarding security requirements.
10. **Use community resources** - trust to these communities is necessary.

6.4 IEEE - General Principles on AI

Institute of Electrical and Electronics Engineers (IEEE) is a standard association, which defined eight principles, that should be followed in order to add value to human life and to ensure trustworthiness of systems, they are the following [85]:

1. **Human rights** - Systems should be created and operated to respect, protect and promote internationally recognized human rights.
2. **Well-being** - The primary success criterion for the development of the system should be the added value for human well-being.
3. **Data agency** - Individuals should be empowered by the AI creators with the ability to access and securely share their data, in order to maintain the control over their identity.
4. **Effectiveness** - System creators and operators should provide evidence about the effectiveness and

appropriateness of the AI system.

5. **Transparency** - The predictions/decisions of the systems should always be traceable.
6. **Accountability** - The AI system's decisions should always have an unambiguous rationale.
7. **Awareness of misuse** - AI creators should always protect the system against potential misuses and risks of the systems.
8. **Competence** - System creators should specify the skills and knowledge required for safe and effective use of the system and operators should fulfil them.

6.5 OECD - Recommendations on AI

Organisation for Economic Co-operation and Development (OECD) is an European intergovernmental organisation, which identified five principles for trustworthy AI, which are following [86]:

- **Inclusive growth, sustainable development and well-being** - Stakeholders should engage in the pursuit of trustworthy AI. In addition AI systems should benefit human-beings and the planet, reduce inequality and protect natural environments.
- **Human-centred values and fairness** - AI systems should respect and protect laws, human rights and democratic values throughout the whole AI lifecycle. These include freedom, dignity, autonomy, privacy, non-discrimination, equality, diversity, fairness, social justice and internationally recognised labour rights. Safeguards to protect these values should be implemented.
- **Transparency and explainability** - AI creators should ensure transparency and responsible disclosure for their AI systems.
- **Robustness, security and safety** - AI systems should be robust, safe and secure during the whole AI lifecycle, including the foreseeable use or misuse. Moreover, the traceability of predictions should be ensured and a systematic risk assessment should be conducted regularly to ensure the safety.
- **Accountability** - AI actors should be accountable for the AI systems.

In addition five recommendations for policy makers for trustworthy AI were defined [86]:

- Investing in AI research and development
- Fostering a digital ecosystem for AI
- Shaping an enabling policy environment for AI
- Building human capacity and preparing for labour market transformation
- International cooperation for trustworthy AI

6.6 Asilomar - AI principles

The Asilomar conference was organized by the *future of life institute* in 2017, which focused on defining following AI principles [87]:

1. **Research goal** - The research goal should be beneficial to society.
2. **Research funding** - The beneficial use of AI systems should be ensured through investments/funding.
3. **Science-policy link** - There should be good and constructive exchange between researchers and policy-makers.
4. **Research culture** - Between researchers and developers of AI systems there should be collaboration, trust and transparency.
5. **Race avoidance** - Active cooperation should exist between teams in order to secure AI systems and avoid cutting back on them.
6. **Safety** - The safety of the systems should be secured during their whole lifecycles.
7. **Failure transparency** - Reasons for faulty behaviour of AI systems should be transparent.
8. **Judicial transparency** - Systems used for judicial decisions should be explainable and their decisions traceable. Thereby they should be auditable and reviewable by human authority.
9. **Responsibility** - Designers and builders of AI systems are responsible for their systems and their moral implications. e.g. their use and misuse.
10. **Value alignment** - The goals and behaviours of AI systems should be aligned with human values during all stages.
11. **Human values** - Human values should not be constrained by AI systems, e.g. human dignity, rights, freedom and cultural diversity.
12. **Personal privacy** - Individuals should be empowered to access, manage and control their data.
13. **Liberty and privacy** - The freedom of an individual should not be restricted by AI applications.
14. **Shared Benefit** - As many people as possible should benefit and be empowered through AI systems.
15. **Shared Prosperity** - All human-beings should benefit from economic prosperity created by AI.
16. **Human Control** - Humans should be able to control, when they want to delegate decisions to AI systems.
17. **Non-subversion** - AI systems should respect and improve the social and civic process, rather than subvert them.
18. **AI Arms Race** - AI systems should not be used for lethal autonomous weapons.
19. **Capability Caution** - Upper limits of future AI capabilities cannot be defined and therefore, assump-

tions should be avoided.

20. **Importance** - Advanced AI systems could change everything and should be planned and managed with care.
21. **Risks** - Risks should be planned and mitigated regarding their expected impact.
22. **Recursive Self-Improvement** - Self-learning systems should be secured through safety and control measures.
23. **Common Good** - Super-intelligence should only be used for the benefit of all humanity and should be using widely shared ethical ideas.

6.7 Artificial intelligence at Google: our principles

Google develops and invests a lot in AI research, their team is very involved in policy making and research, as such they established principles to lay a foundation for their company and their future development of AI [88]:

1. **Social benefit** - AI systems have an impact on society and therefore, Google commits to consider social and economic factors and weigh up the associated foreseeable risks and disadvantages. In addition, Google obliges itself to respect legal, social and cultural norms in the countries of operation.
2. **Prevent Bias** - Because algorithms and datasets of AI systems can reflect, reinforce or reduce unfair biases, Google seeks to avoid unfair effects for human, e.g. related to race, ethnicity, religion.
3. **Ensure Safety** - Google commits to the continuous development of security measures for AI systems.
4. **Accountability** - Google obliges to design AI systems, which are able to provide feedback, explanations and objections. Moreover, AI systems are under adequate human direction and control.
5. **Privacy** - Privacy principles have to be implemented in AI technology.
6. **Standards of scientific excellence** - Google commits to standards of scientific excellence, to share AI knowledge by publishing research, best practices and other materials.
7. **Availability** - Because the usage of AI systems can be diverse, it is important for Google to limit potential harmful or abusive uses and therefore, likely utilizations are evaluated.

6.7.1 AI applications, that will not be pursued

In addition to all of the above principles and guides, Google has defined application areas, where development or implementation should not be pursued, which are systems [88]:

1. that cause harm or are likely to cause it. If a risk is identified, only systems, that outweigh these risks

with their benefits are developed.

2. with the primary purpose to harm people, e.g. weapons.
3. that violate internationally recognised norms by collecting information or using information for surveillance.
4. that violate international law and human rights.

6.8 Microsoft - AI principles

Microsoft puts its AI principles into practice through the *Office of Responsible AI*, which puts rules and governance processes into place. The *AI, Ethics, and Effects in Engineering and Research (Aether) Committee*, which advises the leader board on new opportunities, and the *Responsible AI Strategy in Engineering (RAISE)*, which enables the implementation of Microsoft responsible AI rules across engineering groups. Microsoft principles are [89]:

- **Fairness** - People should be treated fairly by AI systems.
- **Reliability and safety** - Reliable and safe operation of the systems should be ensured.
- **Privacy and security** - The systems should be secure and ensure privacy.
- **Inclusiveness** - AI Systems should empower and engage everyone.
- **Transparency** - The systems should be developed to be understandable.
- **Accountability** - Someone should be accountable for the AI systems.

6.9 IBM - foundational properties for AI ethics

IBM defined five foundational pillars for their AI ethics, which are [90]:

- **Explainability** - The transparency and explainability of AI systems should be a given.
- **Fairness** - The decision of AI system should assist humans in making fairer choices.
- **Robustness** - Because AI systems make crucial decisions, they have to be secure and robust.
- **Transparency** - The trust in AI systems should be reinforced through transparency and the best way is through disclosure.
- **Privacy** - The prioritization and safeguard of consumers' privacy and data rights have to be ensured.

6.10 Facebook - Five pillars of Responsible AI

AI is part of the systems at Facebook and therefore, these five pillars were introduced [91]:

- **Privacy and security** - The privacy and security of the individuals' data should be secured, which is ensured through a Privacy Review process.
- **Fairness and inclusion** - The products should treat everyone fairly and should be usable for everyone.
- **Robustness and safety** - AI systems should meet standards and are therefore tested to ensure their safety and behaviour. Moreover, an AI red team was established, that tests the robustness of the systems.
- **Transparency and control** - There should be more transparency and control about data collection and usage. The explainability of decisions is important.
- **Accountability and governance** - The AI systems have to meet internal and external standards/best practices, where necessary or appropriate humans should be in the loop to monitor the decisions of the systems and are able to intervene.

6.11 Telia - Guiding Principles on trusted AI ethics

Telia Company is the leading telecommunications group and mobile network operator in Finland, Sweden and Lithuania. They work with AI for advanced analytics and increased efficiency, therefore, trust in these AI systems is mandatory and nine guiding principles were proposed [92]:

1. **Responsible and value centric** - Value should be added for the customers and other stakeholders. AI systems should be developed and operated with responsibility and care, which considers opportunities as well as potential risks.
2. **Human centric** - AI systems should simplify and enhance the customers' life. The AI should complement and extend the interaction and human abilities in a sustainable way.
3. **Rights respecting** - Respected international rights and anti-discrimination approaches should be preserved. Moreover, the collection of data should always be transparent and purposeful.
4. **Control** - AI systems should be permanently monitored and intervention should be possible in order to improve the system or prevent/reduce damage.
5. **Accountable** - Telia should remain responsible for its systems and responsibilities should be clearly defined.
6. **Safe and secure** - Misuse and risk of compromise should be reduced/prevented through regular testing the AI systems.
7. **Transparent and explainable** - Transparency of predictions should be ensured and users informed about the use of AI systems.

8. **Fair and equal** - Fairness and equality should be ensured in datasets and algorithms during all AI lifecycle phases.
9. **Continuous review and dialogue** - The best practices of relevant industries should be reviewed and adaptions should be planned when necessary.

6.12 Principle comparison table

The following table aggregates the principles, listed before and maps them to different categories. It seems that, the more mentions a category has, the more important it is for the institutions and the private sector. However, it also shows, that misuse and risks of AI systems are still under-represented in these principles.

Principle	NIST	HLEG	Brund.	BIML	IEEE	OECD	Asil.	Google	MS	IBM	FB	Telia
Explainability	x					x				x	x	x
Traceability	x	x							x			
Accuracy	x											
Knowledge limits	x											
Human agency and oversight		x					x				x	x
Technical robustness and safety		x	x	x		x	x	x	x	x	x	x
Privacy and (data) governance		x	x	x		x	x	x	x	x	x	x
Transparency		x			x	x			x	x	x	x
Diversity, non-discrimination, bias, fairness		x	x			x	x	x	x	x		x
Accountability		x			x	x	x	x	x		x	x

Principle	NIST	HLEG	Brund.	BIML	IEEE	OECD	Asil.	Google	MS	IBM	FB	Telia
Societal & environmental well-being		x			x	x	x	x				x
Third party auditing			x									
Red teaming			x								x	
Bounties			x									
Sharing of incidents			x									
Audit trails			x									
Interpretability			x									
Research funding			x				x					
Sharing resources				x								
Data agency					x							
Effectiveness					x							
Awareness of misuse					x		x					x
Competence					x							
Continuous review							x	x				x
Availability								x				
Inclusion									x		x	
Data control											x	
Research culture							x	x				x
Risks							x					

Table 6.1: Principle comparison table

7 Conclusion

In this thesis, a general introduction to AI systems, their lifecycle, their use and their different implementations was given. Since AI systems are becoming more popular in order to ease time-consuming tasks, the misuse of these systems is also rising. To secure these systems an aggregated risk and threat table was created, which defined 157 possible risks and threats for AI systems. These risks and threats are AI-specific, but are also based on their generic components. Moreover, these identified risks and threats were sorted, based on their attack goal and appearance in the lifecycle of AI systems. The categories are as followed:

1. Raw data
2. Dataset assembly
3. Datasets
4. Learning algorithm
5. Evaluation
6. Input
7. Model
8. Inference algorithm
9. Output
10. System-wide and broad concerns

The threat and risk level is influenced by the likelihood of the exploit, which in turn can be influenced by different factors, like remote availability, public facing applications and the attackers' motivation. These threats and risks can help categorise adversary attacks and make systems more secure, because specific misuse cases can be mitigated.

Another aspect reviewed by this thesis, was the current state of trustworthy AI principles, which concluded that risks and threats are not being examined in great detail right now.

7.1 Future Work

Future work can be based on the aggregated risk and threat table. This table can be further developed and used for threat modelling of AI systems. Mitigations for the risks and threats of these aggregated list can be elaborated. The generic risks and threats for AI systems can be used to classify new attacks.
Moreover, the overview of principles for trustworthy AI systems can help future development of such guidelines and principles, and inspire to include the threat and risk landscape.

List of Figures

List of Tables

Glossary

ADAS	Advanced Driver Assistance Systems
AI	Artificial Intelligence
ALC	Automated Lane Centering
ANN	Artificial Neural Network
API	Application Programming Interface
BIML	Berryville Institute of Machine Learning
CNN	Convolutional Neural Network
DL	Deep Learning
DNN	Deep Neural Network
e.g.	exempli gratia
ENISA	European Union Agency for Cybersecurity
GDPR	General Data Protection Regulation
HLEG	High-level expert group
IBM	Inernational Business Machines Corporation
IEEE	Institute of Electrical and Electronics Engineers
Malware	Malicious computer software

MITRE	Massachusetts Institute of Technology Research and Engineering
ML	Machine Learning
NGAV	Next Generation Antivirus
NIDS	Network Intrusion Detection System
NIST	National Institute of Standards and Technology
NN	Neural Network
OECD	Organisation for Economic Co-operation and Development
PE	Portable Executable
Ransomware	Malware, preventing user from accessing system or files, demanding money in exchange for files
RNN	Recurrent Neural Network
SLA	Service Level Agreement
U.S.	United States

Bibliography

[1] Ram Shankar Siva Kumar and Anna Johnson, *Cyberattacks against machine learning systems are more common than you think*, en-US, Microsoft, Oct. 2020. [Online]. Available: `https://www.microsoft.com/security/blog/2020/10/22/cyberattacks-against-machine-learning-systems-are-more-common-than-you-think/` (visited on 04/14/2021).

[2] David Cearley, Brian Burke, David Smith, Nick Jones, Arun Chandrasekaran, and CK Lu, *Top 10 Strategic Technology Trends for 2020*, en. [Online]. Available: `https://www.gartner.com/en/doc/432920-top-10-strategic-technology-trends-for-2020` (visited on 07/27/2021).

[3] Andreas Holzinger, Peter Kieseberg, Edgar Weippl, and A. Min Tjoa, *Current Advances, Trends and Challenges of Machine Learning and Knowledge Extraction: From Machine Learning to Explainable AI*, Andreas Holzinger, Peter Kieseberg, A Min Tjoa, and Edgar Weippl, Eds. Springer International Publishing, 2018, pp. 1–8.

[4] Miles Brundage, Shahar Avin, Jasmine Wang, Haydn Belfield, Gretchen Krueger, Gillian Hadfield, Heidy Khlaaf, Jingying Yang, Helen Toner, Ruth Fong, Tegan Maharaj, Pang Wei Koh, Sara Hooker, Jade Leung, Andrew Trask, Emma Bluemke, Jonathan Lebensold, Cullen O'Keefe, Mark Koren, Théo Ryffel, JB Rubinovitz, Tamay Besiroglu, Federica Carugati, Jack Clark, Peter Eckersley, Sarah de Haas, Maritza Johnson, Ben Laurie, Alex Ingerman, Igor Krawczuk, Amanda Askell, Rosario Cammarota, Andrew Lohn, David Krueger, Charlotte Stix, Peter Henderson, Logan Graham, Carina Prunkl, Bianca Martin, Elizabeth Seger, Noa Zilberman, Seán Ó hÉigeartaigh, Frens Kroeger, Girish Sastry, Rebecca Kagan, Adrian Weller, Brian Tse, Elizabeth Barnes, Allan Dafoe, Paul Scharre, Ariel Herbert-Voss, Martijn Rasser, Shagun Sodhani, Carrick Flynn, Thomas Krendl Gilbert, Lisa Dyer, Saif Khan, Yoshua Bengio, and Markus Anderljung, "Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims," *arXiv:2004.07213 [cs]*, Apr. 2020, arXiv: 2004.07213. [Online]. Available: `http://arxiv.org/abs/2004.07213`.

[5] *ISO - ISO/IEC JTC 1/SC 42 - Artificial intelligence*. [Online]. Available: https://www.iso.org/committee/6794475/x/catalogue/p/0/u/1/w/0/d/0 (visited on 07/27/2021).

[6] *AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What's the Difference?* en-us, May 2020. [Online]. Available: https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks (visited on 07/21/2021).

[7] Yulia Gavrilova, *AI vs. ML vs. DL: What's the Difference*, en, 2020. [Online]. Available: https://serokell.io/blog/ai-ml-dl-difference (visited on 07/21/2021).

[8] Dr Michael J. Garbade, *Clearing the Confusion: AI vs Machine Learning vs Deep Learning Differences*, en, 2018. [Online]. Available: https://towardsdatascience.com/clearing-the-confusion-ai-vs-machine-learning-vs-deep-learning-differences-fce69b21d5eb (visited on 07/21/2021).

[9] Collin Payne and Edward J. Glantz, "Teaching adversarial machine learning: Educating the next generation of technical and security professionals," in *Proceedings of the 21st Annual Conference on Information Technology Education*, ser. SIGITE '20, Virtual Event, USA: Association for Computing Machinery, 2020, pp. 7–12, ISBN: 9781450370455. DOI: 10.1145/3368308.3415381. [Online]. Available: https://doi.org/10.1145/3368308.3415381.

[10] Miles Brundage, Shahar Avin, Jack Clark, Helen Toner, Peter Eckersley, Ben Garfinkel, Allan Dafoe, Paul Scharre, Thomas Zeitzoff, Bobby Filar, Hyrum Anderson, Heather Roff, Gregory C. Allen, Jacob Steinhardt, Carrick Flynn, Seán Ó hÉigeartaigh, Simon Beard, Haydn Belfield, Sebastian Farquhar, Clare Lyle, Rebecca Crootof, Owain Evans, Michael Page, Joanna Bryson, Roman Yampolskiy, and Dario Amodei, "The malicious use of artificial intelligence: Forecasting, prevention, and mitigation," *CoRR*, vol. abs/1802.07228, 2018. [Online]. Available: http://arxiv.org/abs/1802.07228.

[11] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay, *Adversarial attacks and defences: A survey*, 2018. arXiv: 1810.00069 [cs.LG].

[12] Stuart Russell and Peter Norvig, "Artificial intelligence: A modern approach, global edition 4th," *Foundations*, vol. 19, p. 23, 2021.

[13] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy, *Explaining and harnessing adversarial examples*, 2015. arXiv: 1412.6572 [stat.ML].

[14] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus, *Intriguing properties of neural networks*, 2014. arXiv: 1312.6199 [cs.CV].

[15] Wei Emma Zhang, Quan Z. Sheng, Ahoud Alhazmi, and Chenliang Li, "Adversarial attacks on deep-learning models in natural language processing: A survey," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 3, Apr. 2020, ISSN: 2157-6904. DOI: 10.1145/3374217. [Online]. Available: https://doi.org/10.1145/3374217.

[16] ENISA, *Artificial Intelligence Cybersecurity Challenges*, en-gb, Report/Study, Dec. 2020. [Online]. Available: https://www.enisa.europa.eu/publications/artificial-intelligence-cybersecurity-challenges (visited on 04/06/2021).

[17] Gary McGraw, Harold Figueroa, Shepardson Victor, and Richie Bonett, *An architectural risk analysis of machine learning systems*, Jan. 2020. [Online]. Available: https://berryvilleiml.com/docs/ara.pdf.

[18] *Definition of PERTURBATION*, en. [Online]. Available: https://www.merriam-webster.com/dictionary/perturbation (visited on 07/19/2021).

[19] Siddhant Bhambri, Sumanyu Muku, Avinash Tulasi, and Arun Balaji Buduru, *A survey of black-box adversarial attacks on computer vision models*, 2020. arXiv: 1912.01667 [cs.LG].

[20] International Organization for Standardization [ISO], *Iso/iec tr 24028:2020 information technology — artificial intelligence — overview of trustworthiness in artificial intelligence*. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso-iec:tr:24028:ed-1:v1:en (visited on 07/22/2021).

[21] Adam Shostack, *Threat modeling: Designing for security*. John Wiley & Sons, 2014.

[22] Nektaria Kaloudi and Jingyue Li, "The ai-based cyber threat landscape: A survey," *ACM Comput. Surv.*, vol. 53, no. 1, Feb. 2020, ISSN: 0360-0300. DOI: 10.1145/3372823. [Online]. Available: https://doi.org/10.1145/3372823.

[23] Suvda Myagmar, Adam J Lee, and William Yurcik, "Threat modeling as a basis for security requirements," in *Symposium on requirements engineering for information security (SREIS)*, Citeseer, vol. 2005, 2005, pp. 1–8.

[24] Ling Huang, Anthony D. Joseph, Blaine Nelson, Benjamin I.P. Rubinstein, and J. D. Tygar, "Adversarial machine learning," in *Proceedings of the 4th ACM Workshop on Security and Artificial Intelligence*, ser. AISec '11, Chicago, Illinois, USA: Association for Computing Machinery, 2011,

pp. 43–58, ISBN: 9781450310031. DOI: 10.1145/2046684.2046692. [Online]. Available: https://doi.org/10.1145/2046684.2046692.

[25] Imtithal A Saeed, Ali Selamat, and Ali MA Abuagoub, "A survey on malware and malware detection systems," *International Journal of Computer Applications*, vol. 67, no. 16, 2013.

[26] Hui Xu, Yangfan Zhou, and Michael Lyu, "N-version obfuscation," in *Proceedings of the 2nd ACM International Workshop on Cyber-Physical System Security*, ser. CPSS '16, Xi'an, China: Association for Computing Machinery, 2016, pp. 22–33, ISBN: 9781450342889. DOI: 10.1145/2899015.2899026. [Online]. Available: https://doi.org/10.1145/2899015.2899026.

[27] Omar Saad, Ashraf Darwish, and Ramadan Faraj, "A survey of machine learning techniques for spam filtering," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 12, no. 2, p. 66, 2012.

[28] Asif Karim, Sami Azam, Bharanidharan Shanmugam, and Krishnan Kannoorpatti, "Efficient clustering of emails into spam and ham: The foundational study of a comprehensive unsupervised framework," *IEEE Access*, vol. 8, pp. 154 759–154 788, 2020. DOI: 10.1109/ACCESS.2020.3017082.

[29] Emily Bauer, *15 Outrageous Email Spam Statistics that Still Ring True in 2018*, Feb. 2018. [Online]. Available: https://www.propellercrm.com/blog/email-spam-statistics (visited on 07/21/2021).

[30] *ISO - About us*, en. [Online]. Available: https://www.iso.org/about-us.html (visited on 07/26/2021).

[31] Ram Shankar Siva Kumar, David O'Brien, Jeffrey Snover, Kendra Albert, and Salome Viljoen, *Failure Modes in Machine Learning - Security documentation*, Microsoft, Nov. 2019. [Online]. Available: https://docs.microsoft.com/en-us/security/engineering/failure-modes-in-machine-learning (visited on 04/28/2021).

[32] Andrew Marshall, Jugal Parikh, Emre Kiciman, and Ram Shankar Siva Kumar, *AI/ML Pivots to the Security Development Lifecycle Bug Bar - Security documentation*, en-us, Microsoft, Nov. 2019. [Online]. Available: https://docs.microsoft.com/en-us/security/engineering/bug-bar-aiml (visited on 04/28/2021).

[33] ——, *Threat Modeling AI/ML Systems and Dependencies - Security documentation*, en-us, Microsoft, Nov. 2019. [Online]. Available: https://docs.microsoft.com/en-us/security/engineering/threat-modeling-aiml (visited on 04/14/2021).

[34] Andrew Marshall, Raul Rojas, Jay Stokes, and Donald Brinkman, *Securing the Future of AI and ML at Microsoft - Security documentation*, en-us, Microsoft, Apr. 2018. [Online]. Available: `https://docs.microsoft.com/en-us/security/engineering/securing-artificial-intelligence-machine-learning` (visited on 04/28/2021).

[35] MITRE, Microsoft, Bosch, IBM, NVIDIA, Airbus, PricewaterhousCoopers, Deep Instinct, Two Six Labs, University of Toronto, Cardiff University, Software Engineering Institute/Carnegie Mellon University, Berryville Institute of Machine Learning, Citadel AI, McAfee, Unaffiliated, Ant Group, and Palo Alto Networks, *Adversarial Threat Matrix*. [Online]. Available: `https://github.com/mitre/advmlthreatmatrix` (visited on 04/28/2021).

[36] robin.materese@nist.gov, *About NIST*, en, text, Last Modified: 2017-06-14T16:04-04:00, Jan. 2015. [Online]. Available: `https://www.nist.gov/about-nist` (visited on 07/26/2021).

[37] Elham Tabassi, Kevin J. Burns, Michael Hadjimichael, Andres D. Molina-Markham, and Julian T. Sexton, "A taxonomy and terminology of adversarial machine learning," en, preprint, Oct. 2019. DOI: `10.6028/NIST.IR.8269-draft`. [Online]. Available: `https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8269-draft.pdf` (visited on 04/08/2021).

[38] ENISA, *Cybersecurity Challenges in the Uptake of Artificial Intelligence in Autonomous Driving*, en-gb, Report/Study. [Online]. Available: `https://www.enisa.europa.eu/publications/enisa-jrc-cybersecurity-challenges-in-the-uptake-of-artificial-intelligence-in-autonomous-driving` (visited on 04/06/2021).

[39] Joseph Early, *Your Car May Not Know When to Stop — Adversarial Attacks Against Autonomous Vehicles*, en, Sep. 2019. [Online]. Available: `https://towardsdatascience.com/your-car-may-not-know-when-to-stop-adversarial-attacks-against-autonomous-vehicles-a16df91511f4` (visited on 07/11/2021).

[40] James Rundle and John McCormick, "Bosch Deploys AI to Prevent Attacks on Cars' Electronic Systems," en-US, *Wall Street Journal*, Jan. 2020, ISSN: 0099-9660. [Online]. Available: `https://www.wsj.com/articles/bosch-deploys-ai-to-prevent-attacks-on-cars-electronic-systems-11578306600` (visited on 07/11/2021).

[41] Ishai Rosenberg and Shai Meir, "Bypassing NGAV for Fun and Profit," en, p. 25, [Online]. Available: `https://i.blackhat.com/eu-20/Thursday/eu-20-Rosenberg-Bypassing-NGAV-For-Fun-And-Profit-wp.pdf` (visited on 07/16/2021).

[42] Kaichen Yang, Tzungyu Tsai, Honggang Yu, Max Panoff, Tsung-Yi Ho, and Yier Jin, "Robust roadside physical adversarial attack against deep learning in lidar perception modules," in *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, ser. ASIA CCS '21, Virtual Event, Hong Kong: Association for Computing Machinery, 2021, pp. 349–362. DOI: `10.1145/3433210.3453106`. [Online]. Available: `https://doi.org/10.1145/3433210.3453106`.

[43] Ariel Herbert-Voss, *Practical Defenses Against Adversarial Machine Learning - black hat 2020*. [Online]. Available: `https://www.youtube.com/watch?v=RdHYZJ2S_Zk&t=1152s` (visited on 07/16/2021).

[44] Timothy B. Lee, *Waymo tells riders to get ready for fully driverless rides*, en-us, Oct. 2019. [Online]. Available: `https://arstechnica.com/cars/2019/10/waymo-starts-offering-driverless-rides-to-ordinary-riders-in-phoenix/` (visited on 07/13/2021).

[45] Anjali Berdia Simona Shemer, *Self-Driving Spin: Riding In An Autonomous Vehicle Around Tel Aviv*, en-US, Jun. 2019. [Online]. Available: `https://nocamels.com/2019/06/autonomous-vehicle-yandex-tech/` (visited on 07/13/2021).

[46] Shang-Tse Chen, Cory Cornelius, Jason Martin, and Duen Horng Chau, "Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector," *Lecture Notes in Computer Science*, pp. 52–68, 2019, ISSN: 1611-3349. DOI: `10.1007/978-3-030-10925-7_4`. [Online]. Available: `http://dx.doi.org/10.1007/978-3-030-10925-7_4`.

[47] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song, *Robust physical-world attacks on deep learning models*, 2018. arXiv: `1707.08945 [cs.CR]`.

[48] *Tencent Keen Security Lab: Experimental Security Research of Tesla Autopilot | Keen Security Lab Blog*, 2019. [Online]. Available: `https://keenlab.tencent.com/en/2019/03/29/Tencent-Keen-Security-Lab-Experimental-Security-Research-of-Tesla-Autopilot/` (visited on 07/13/2021).

[49] Nir Morgulis, Alexander Kreines, Shachar Mendelowitz, and Yuval Weisglass, *Fooling a real car with adversarial traffic signs*, 2019. arXiv: `1907.00374 [cs.CR]`.

[50] Chawin Sitawarin, Arjun Nitin Bhagoji, Arsalan Mosenia, Mung Chiang, and Prateek Mittal, *Darts: Deceiving autonomous cars with toxic signs*, 2018. arXiv: `1802.06430 [cs.CR]`.

[51] Yue Zhao, Hong Zhu, Ruigang Liang, Qintao Shen, Shengzhi Zhang, and Kai Chen, *Seeing isn't believing: Practical adversarial attack against object detectors*, 2019. arXiv: `1812.10217 [cs.CV]`.

[52] David Gerónimo, Antonio M. López, Angel D. Sappa, and Thorsten Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1239–1258, 2010. DOI: `10.1109/TPAMI.2009.122`.

[53] Jack Stewart, "Why Tesla's Autopilot Can't See a Stopped Firetruck," en-US, *Wired*, 2018, ISSN: 1059-1028. [Online]. Available: `https://www.wired.com/story/tesla-autopilot-why-crash-radar/` (visited on 07/13/2021).

[54] "Safety First for Automated Driving," en, p. 157, 2019. [Online]. Available: `https://www.daimler.com/documents/innovation/other/safety-first-for-automated-driving.pdf`.

[55] Ben Nassi, Yisroel Mirsky, Dudi Nassi, Raz Ben-Netanel, Oleg Drokin, and Yuval Elovici, "Phantom of the adas: Securing advanced driver-assistance systems from split-second phantom attacks," in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '20, Virtual Event, USA: Association for Computing Machinery, 2020, pp. 293–308, ISBN: 9781450370899. DOI: `10.1145/3372297.3423359`. [Online]. Available: `https://doi.org/10.1145/3372297.3423359`.

[56] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Florian Tramer, Atul Prakash, Tadayoshi Kohno, and Dawn Song, *Physical adversarial examples for object detectors*, 2018. arXiv: `1807.07769 [cs.CR]`.

[57] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana, "Deepxplore: Automated whitebox testing of deep learning systems," *Proceedings of the 26th Symposium on Operating Systems Principles*, Oct. 2017. DOI: `10.1145/3132747.3132785`. [Online]. Available: `http://dx.doi.org/10.1145/3132747.3132785`.

[58] Yuchi Tian, Kexin Pei, Suman Jana, and Baishakhi Ray, *Deeptest: Automated testing of deep-neural-network-driven autonomous cars*, 2018. arXiv: `1708.08559 [cs.SE]`.

[59] Alesia Chernikova, Alina Oprea, Cristina Nita-Rotaru, and BaekGyu Kim, *Are self-driving cars secure? evasion attacks against deep neural networks for steering angle prediction*, 2019. arXiv: `1904.07370 [cs.LG]`.

[60] Husheng Zhou, Wei Li, Zelun Kong, Junfeng Guo, Yuqun Zhang, Bei Yu, Lingming Zhang, and Cong Liu, "Deepbillboard: Systematic physical-world testing of autonomous driving systems," in *2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE)*, 2020, pp. 347–358.

[61] Dan Strollo, *The Three Second Rule*, en-US. [Online]. Available: `https://www.driveincontrol.org/drivingtips/the-three-second-rule` (visited on 07/13/2021).

[62] *Commaai/openpilot: Openpilot is an open source driver assistance system. openpilot performs the functions of Automated Lane Centering and Adaptive Cruise Control for over 100 supported car makes and models.* [Online]. Available: `https://github.com/commaai/openpilot` (visited on 07/13/2021).

[63] Takami Sato, Junjie Shen, Ningfei Wang, Yunhan Jack Jia, Xue Lin, and Qi Alfred Chen, *Dirty road can attack: Security of deep learning based automated lane centering under physical-world attack*, 2021. arXiv: `2009.06701 [cs.CR]`.

[64] Alex H. Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 689–12 697. DOI: `10.1109/CVPR.2019.01298`.

[65] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 526–10 535. DOI: `10.1109/CVPR42600.2020.01054`.

[66] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li, *Pointrcnn: 3d object proposal generation and detection from point cloud*, 2019. arXiv: `1812.04244 [cs.CV]`.

[67] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song, "Robust physical-world attacks on deep learning visual classification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1625–1634. DOI: `10.1109/CVPR.2018.00175`.

[68] Ishai Rosenberg, Asaf Shabtai, Yuval Elovici, and Lior Rokach, "Adversarial machine learning attacks and defense methods in the cyber security domain," *ACM Comput. Surv.*, vol. 54, no. 5, May 2021, ISSN: 0360-0300. DOI: `10.1145/3453158`. [Online]. Available: `https://doi.org/10.1145/3453158`.

[69] Ishai Rosenberg, Shai Meir, Jonathan Berrebi, Ilay Gordon, Guillaume Sicard, Eli, and David, *Generating end-to-end adversarial examples for malware classifiers using explainability*, 2020. arXiv: `2009.13243 [cs.CR]`.

[70] Hyrum S. Anderson and Phil Roth, *Ember: An open dataset for training static pe malware machine learning models*, 2018. arXiv: `1804.04637 [cs.CR]`.

[71] Aditya Kuppa, Slawomir Grzonkowski, Muhammad Rizwan Asghar, and Nhien-An Le-Khac, “Black box attacks on deep anomaly detectors,” in *Proceedings of the 14th International Conference on Availability, Reliability and Security*, ser. ARES ’19, Canterbury, CA, United Kingdom: Association for Computing Machinery, 2019, ISBN: 9781450371643. DOI: `10.1145/3339252.3339266`. [Online]. Available: `https://doi.org/10.1145/3339252.3339266`.

[72] Didong Li, Minerva Mukhopadhyay, and David B. Dunson, *Efficient manifold and subspace approximations with spherelets*, 2019. arXiv: `1706.08263 [stat.ML]`.

[73] Bo Zong, Qi Song, Martin Renqiang Min, Wei Cheng, Cristian Lumezanu, Daeki Cho, and Haifeng Chen, “Deep autoencoding gaussian mixture model for unsupervised anomaly detection,” in *International Conference on Learning Representations*, 2018. [Online]. Available: `https://openreview.net/forum?id=BJJLHbb0-`.

[74] Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs, *Unsupervised anomaly detection with generative adversarial networks to guide marker discovery*, 2017. arXiv: `1703.05921 [cs.CV]`.

[75] Houssam Zenati, Manon Romain, Chuan Sheng Foo, Bruno Lecouat, and Vijay Ramaseshan Chandrasekhar, *Adversarially learned anomaly detection*, 2018. arXiv: `1812.02288 [cs.LG]`.

[76] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft, “Deep one-class classification,” in *Proceedings of the 35th International Conference on Machine Learning*, Jennifer Dy and Andreas Krause, Eds., ser. Proceedings of Machine Learning Research, vol. 80, PMLR, Oct. 2018, pp. 4393–4402. [Online]. Available: `http://proceedings.mlr.press/v80/ruff18a.html`.

[77] Bernhard Schölkopf, Robert Williamson, Alex Smola, John Shawe-Taylor, and John Platt, “Support vector method for novelty detection,” vol. 12, Jan. 1999, pp. 582–588.

[78] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou, "Isolation forest," in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ser. ICDM '08, USA: IEEE Computer Society, 2008, pp. 413–422, ISBN: 9780769535029. DOI: `10.1109/ICDM.2008.17`. [Online]. Available: `https://doi.org/10.1109/ICDM.2008.17`.

[79] *A Realistic Cyber Defense Dataset (CSE-CIC-IDS2018) - Registry of Open Data on AWS*. [Online]. Available: `https://registry.opendata.aws/cse-cic-ids2018/` (visited on 07/16/2021).

[80] NIST, *NVD - CVE-2019-20634*. [Online]. Available: `https://nvd.nist.gov/vuln/detail/CVE-2019-20634` (visited on 08/08/2021).

[81] Volodymyr Kuleshov, Shantanu Thakoor, Tingfung Lau, and Stefano Ermon, *Adversarial examples for natural language classification problems*, 2018. [Online]. Available: `https://openreview.net/forum?id=r1QZ3zbAZ`.

[82] Qi Lei, Lingfei Wu, Pin-Yu Chen, Alexandros G. Dimakis, Inderjit S. Dhillon, and Michael Witbrock, *Discrete adversarial attacks and submodular optimization with applications to text classification*, 2019. arXiv: `1812.00151 [cs.LG]`.

[83] P. Jonathon Phillips, Carina A. Hahn, Peter C. Fontana, David A. Broniatowski, and Mark A. Przybocki, "Four Principles of Explainable Artificial Intelligence," en, preprint, Aug. 2020. DOI: `10.6028/NIST.IR.8312-draft`. [Online]. Available: `https://nvlpubs.nist.gov/nistpubs/ir/2020/NIST.IR.8312-draft.pdf` (visited on 04/08/2021).

[84] HLEG, *HIGH-LEVEL EXPERT GROUP ON ARTIFICIAL INTELLIGENCE - ETHICS GUIDELINES FOR TRUSTWORTHY AI*, Apr. 2019. [Online]. Available: `https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai` (visited on 07/30/2021).

[85] Raja Chatila and John C. Havens, "The IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems," en, in *Robotics and Well-Being*, Maria Isabel Aldinhas Ferreira, João Silva Sequeira, Gurvinder Singh Virk, Mohammad Osman Tokhi, and Endre E. Kadar, Eds., vol. 95, Series Title: Intelligent Systems, Control and Automation: Science and Engineering, Cham: Springer International Publishing, 2019, pp. 11–16, ISBN: 978-3-030-12523-3 978-3-030-12524-0. DOI: `10.1007/978-3-030-12524-0_2`. [Online]. Available: `http://link.springer.com/10.1007/978-3-030-12524-0_2` (visited on 07/30/2021).

[86] OECD Council, "Recommendation of the council on artificial intelligence," May 2019. [Online]. Available: https://legalinstruments.oecd.org/api/print?ids=648 (visited on 07/30/2021).

[87] Asilomar Conference, *AI Principles*, en-US, Jan. 2017. [Online]. Available: https://futureoflife.org/ai-principles/ (visited on 07/30/2021).

[88] *Artificial Intelligence at Google: Our Principles*, en. [Online]. Available: https://ai.google/principles/ (visited on 05/30/2021).

[89] Microsoft, *Responsible AI principles from Microsoft*, en-us. [Online]. Available: https://www.microsoft.com/en-us/ai/responsible-ai (visited on 07/30/2021).

[90] IBM, *AI Ethics*, en-us. [Online]. Available: https://www.ibm.com/artificial-intelligence/ethics (visited on 07/30/2021).

[91] Jerome Pesenti, *Facebook's five pillars of Responsible AI*, de, Jun. 2021. [Online]. Available: https://ai.facebook.com/blog/facebooks-five-pillars-of-responsible-ai/ (visited on 07/30/2021).

[92] Telia, *AI ethics*, en. [Online]. Available: https://www.teliacompany.com/sv/om-foretaget/public-policy/ai-ethics/ (visited on 07/30/2021).