January 2023

Towards Specifying And Evaluating The Trustworthiness Of An AI-Enabled System

Mark Arinaitwe

TOWARDS SPECIFYING AND EVALUATING THE TRUSTWORTHINESS OF AN AI-ENABLED SYSTEM

by

Mark Arinaitwe
Master of Science in Computer Science, University of North Dakota 2023

A Thesis

Submitted to the Graduate Faculty

of the

University of North Dakota

in partial fulfillment of the requirements

for the degree of

Master of Science

Grand Forks, North Dakota

May
2023

Name: Mark Arinaitwe

Degree: Master of Science

This document, submitted in partial fulfillment of the requirements for the degree from the University of North Dakota, has been read by the Faculty Advisory Committee under whom the work has been done and is hereby approved.

DocuSigned by:
Hassan Reza
7DBBFCC1BD6F4BE...
Hassan Reza

DocuSigned by:
Eunjin Kim
ECEA0B2E147147E...
Eunjin Kim

DocuSigned by:
Wen-Chen Hu
EE71D886B1C04D0...
Wen-Chen Hu

This document is being submitted by the appointed advisory committee as having met all the requirements of the School of Graduate Studies at the University of North Dakota and is hereby approved.

DocuSigned by:
Chris Nelson
2E0AF088C7334O3...
Chris Nelson
Dean of the School of Graduate Studies

2/27/2023
Date

PERMISSION

Title Towards Specifying and Evaluating the Trustworthiness of an AI-enabled System

Department School of Electrical Engineering and Computer Science.

Degree Master of Computer Science

In presenting this thesis in partial fulfillment of the requirements for a graduate degree from the University of North Dakota, I agree that the library of this University shall make it freely available for inspection. I further agree that permission for extensive copying for scholarly purposes may be granted by the professor who supervised my thesis work. It is understood that any copying or publication or other use of this thesis or part thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of North Dakota in any scholarly use which may be made of any material in my thesis.

Mark Arinaitwe

March 3, 2023

ACKNOWLEDGMENTS

I wish to express my sincere appreciation to the members of my advisory Committee for their guidance and support during my time in the master's program at the University of North Dakota

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Applied AI has shown promise in the data processing of key industries and government agencies to extract actionable information used to make important strategical decisions. One of the core features of AI-enabled systems is the trustworthiness of these systems which has an important implication for the robustness and full acceptance of these systems. In this paper, we explain what trustworthiness in AI-enabled systems means, and the key technical challenges of specifying, and verifying trustworthiness. Toward solving these technical challenges, we propose a method to specify and evaluate the trustworthiness of AI-based systems using quality-attribute scenarios and design tactics. Using our trustworthiness scenarios and design tactics, we can analyze the architectural design of AI-enabled systems to ensure that trustworthiness has been properly expressed and achieved.

The contributions of the thesis include (i) the identification of the trustworthiness sub-attributes that affect the trustworthiness of AI systems (ii) the proposal of trustworthiness scenarios to specify trustworthiness in an AI system (iii) a design checklist to support the analysis of the trustworthiness of AI systems and (iv) the identification of design tactics that can be used to achieve trustworthiness in an AI system.

Index Terms - Trustworthiness, Trustworthy AI, Utility Tree, Trust, Software Architecture, Quality Attribute Scenarios, Architectural Tactics

CHAPTER 1

INTRODUCTION

This chapter describes the problem, motivations, and contributions of the thesis. Section 1.1 provides the problem definition. Section 1.2 outlines the scope of the work. Section 1.3 describes the motivation of the paper. Section 1.4 describes the approach taken to solve the problem. Section 1.5 provides the expected outcomes, and section 1.6 describes the structure of the paper.

1.1 Problem Definition

Artificial Intelligence (AI) is the theory and development of computer systems capable of intelligent behavior such as learning, pattern recognition, visual perception, and problem-solving. Its goals include planning, social perception, natural language processing, and image recognition [1]. Due to advances in AI techniques, computing architectures, and digital data in recent years major developments in the field of AI are being made [2]. By using machine learning (ML) algorithms, such as neural networks, AI systems are automatically able to learn and improve from experience without explicit programming. The use of AI brings about numerous benefits. Some of these benefits include a lower rate of human error, the capability of completing tasks in hostile environments, the completion of repetitive tasks more efficiently and quickly than humans, and the ability to work tirelessly twenty-four hours a day. As a result, the use of AI is becoming increasingly widespread and popular in fields such as healthcare and finance.

Due to the black-box nature of AI, trust is a key factor in whether people are going to adopt AI systems and continue to use them. The increasing popularity of AI has brought about an awareness

of the lack of regulations and norms in academia and industry to produce trustworthy AI [3]. The importance of trustworthy AI is now being recognized [4], and the need for trustworthiness as an important quality attribute has been seen in various industries [5] [6]. This is because a higher level of trustworthiness leads to greater level of trust. However, there are several technical challenges in specifying and verifying the trustworthiness. Trustworthiness is subjective in nature, leading to a lack of consensus of what trustworthiness means in an AI system. Furthermore, AI systems vary in nature and different AI systems require different trustworthiness qualities to be considered trustworthy. For trustworthiness to be properly achieved there must be a way to specify and evaluate what it means for a particular AI system to be trustworthy. Once trustworthiness has been achieved there must then be a way of achieving it. One way to do this is to evaluate the trustworthiness of the system during the design of its software architecture. This allows an AI system manifest trustworthiness and allows for the development of a system with trustworthiness in mind. We propose the use of trustworthiness scenarios to specify an AI systems trustworthiness and trustworthiness tactics to achieve it.

1.2 Scope of Work

The scope of the work falls under the design of trustworthy AI system. We focus on the initial stages of the development of an AI system's software architecture and how trustworthiness, as a quality attribute, can be achieved through the design of a system's software architecture.

1.3 Motivation

Due to the size and complexity of AI systems, the prevention of faults and failures is a challenging task and can be catastrophic leading to loss of life [7] [8]. Furthermore, the automaticity and black-

box nature of AI create uncertainty as to how a decision was made. The rise of information technology such as communication has led to an increase in the amount of data being generated all over the world and such data is subject to existing biases based on religion, race, and gender. This data is an important aspect of an AI system, and the accuracy and correctness of decisions made by AI are highly dependent on this data [9].

The result of all these issues can cause a lack of trust in systems that use AI. This makes trustworthiness a necessary quality for AI systems. A lack of trustworthiness leads to a lack of trust, and the abandonment of such AI systems, forgoing the many benefits that AI provides. Achieving trustworthiness in AI is, therefore, critical to ensure the acceptance and successful adoption of services and products that integrate AI into their systems [10]. Analyzing the trustworthiness of an AI system early in its development provides several benefits. These include enabling a system's trustworthiness to more easily manifest, allowing for the design of an AI system with trustworthiness in mind, and allowing stakeholders to have a greater understanding of what trustworthiness means in a system.

1.4 Approach

We propose the use of trustworthiness scenarios to specify trustworthiness and the use of tactics to achieve trustworthiness in an AI system. We begin with the identification of various trustworthiness sub-attributes that can affect AI systems, then the identification of design decisions such as tactics that can be used to achieve trustworthiness. We then outline a design checklist for trustworthiness to support the design and analysis process, and the generation of trustworthiness scenarios, which can be used in conjunction with methods such as the Architecture Tradeoff

Analysis Method (ATAM) [11] to assess an AI system's software architecture for trustworthiness. The steps taken to do so are described in more detail in the methodology section.

1.5 Thesis Contributions

The purpose of the thesis is to propose a way to specify and evaluate the trustworthiness of an AI system and how to achieve it. The expected outcome is to demonstrate the feasibility of analyzing and designing an AI system for trustworthiness, using trustworthiness scenarios, and achieving trustworthiness using design tactics.

The contributions of the thesis include the following:

- An AI trustworthiness general scenario
- A design checklist for the trustworthiness of AI systems
- Design tactics to achieve trustworthy AI
- A sample use case to validate the feasibility of the proposal

1.6 Thesis Structure

The thesis is structured as follows. Chapter II introduces background information of various concepts used in the research of the thesis. It presents concepts such as trust, trustworthiness, utility tree, and scenarios and challenges faced in the trustworthiness of AI. Chapter III is a review of works done related to the thesis in terms of the trustworthiness of an AI system and attempts to model and measure it. Chapter IV includes the methodology; chapter V discusses the results of the work, with an analysis using ATAM, and chapter VI concludes the thesis with suggestions for future work needed.

CHAPTER 2

BACKGROUND

In this chapter, we provide background information on trust and trustworthiness in an AI system and discuss the various challenges being faced in the trustworthiness of AI. We also explore concepts used in the thesis such as tactics, scenarios, and utility trees and their use in the development and design of a computer system's software architecture. We also discuss the artificial tradeoff analysis method which is used in the validation of this thesis.

2.1 Trust

Before we can understand what makes an entity trustworthy, we must understand what trust is. Trust has been defined in several ways. Schoorman et al. [12] define trust as the willingness of a trustor to be placed in a position of vulnerability to the action of another party (trustee), based on expectations that the other party will perform or behave accordingly, regardless of the ability of the trusting party to monitor and control the trustee. Lee et al. [13] define trust as the attitude that an agent will assist in achieving an individual's goals in a situation of uncertainty and vulnerability. The distinction between the two definitions is that in Schoorman's definition trust involves a willing party while Lee's definition involves an attitude. In Schoorman's definition, there is an action of being vulnerable to a trustee while Lee's involves a belief regardless of whether they interact with a trustee or not. However, both definitions show that trust between a trustor and a trustee is dependent on uncertainty and risk in the act. Therefore, this is an important aspect that should be considered in the relationship between humans and AI.

Concerning AI systems, we can therefore define trust to be defined as the willingness to rely on a complex system that cannot be completely understood or explained. It is the latter definition that we will be using in the paper.

2.2 Trustworthiness and Trustworthy AI

Trustworthiness refers to an attribute that someone or something possess. It is an inherent characteristic, and AI system that exhibits this attribute can be considered 'trustworthy.' It is a critical requirement for the success of AI systems and a deciding factor on whether AI systems are accepted, and their benefits are taken advantage of.

However, defining what factors affect trustworthiness and measuring their influences can be challenging. An example of this is a scale of trust in the field of human factors created by Jian et al. [15], which considers six attributes of trust i.e., fidelity, loyalty, reliability, security, familiarity, and integrity. Some of these attributes are difficult to apply to a machine. For example, the idea that a system can have loyalty cannot be expressed.

There are several attempts to define what factors affect the trustworthiness of AI. Examples of these factors include technical, social, and ethical aspects of trustworthiness [16]. The European Commission formed the High-level Expert Group on Artificial Intelligence (AI-HLEG). The AI-HLEG produced seven key technical, societal, and individual requirements that AI systems should have to be trustworthy [17]: human agency and oversight, technical robustness and safety, privacy and data governance, transparency, diversity, non-discrimination and fairness, societal and environmental wellbeing, and accountability. Madsen et al [18], considered five factors: reliability, technical competence, faith, personal attachment, and understandability. Balfe et al.

[19] considered factors such as reliability, competence, dependability, faith, predictability, personal attachment, understandability, feedback, and robustness.

The current literature also considers different requirements for trustworthiness that dependent on the type of stakeholders involved i.e., developers i.e., those who research, develop, and design the AI system, deployers i.e. public and private groups who offer services and products to others and use AI systems in their businesses, and end-users i.e. those who use AI systems either directly or indirectly.

Trustworthiness is not isolated and is a broad concept that includes attributes such as fairness, robustness, transparency, accountability, explainability, and value alignment. It builds upon the notion of dependability by adding concepts such as providing explanations for the decisions of AI, discovering who is accountable for the decisions of AI, and making sure decisions made by AI are ethically or morally correct. According to Avižienis et al [20], a computer system can be characterized by five fundamental properties: functionality, usability, performance, cost, and dependability. They describe the dependability of a system as the ability of a system to deliver services that can be justifiably trusted. It can be described as the ability of a system to avoid failures or outages that are more severe, more frequent, or have longer durations than is acceptable for a user or users. By extending the definition of dependability, trustworthiness can also include the ability of a system to hide or manage failures, provide accountability for failures as well as determine whether an outcome is a failure or not.

The trustworthiness of AI is also related to security. For example, an adversarial attack is a perturbation in the inputs of a model and can cause it to output incorrect predictions [18]. It is also closely related to safety which deals with preventing a system from causing harm to its users or

environment. It is also related to performance since a higher-performing system is more likely to be trusted by an AI.

Trustworthiness in AI is also about availability by enabling a user to feel confident using an AI system by reducing the possibility of unexpected failures in the system. Due to the nature of AI, one of the most important tasks in artificial intelligence is reducing failures in an AI system and to do so one must understand how failures may arise in such systems.

2.3 Challenges faced in trustworthy AI

A. Subjective

Unlike other quality attributes such as availability and performance which have accepted definitions and have been studied for years, the notion of what constitutes the trustworthiness of AI is unclear. It is subjective to individual interpretations and preferences. For example, an end-user in one field may be concerned with the usability of the system while an end-user in another field may be concerned with the security of the system.

Furthermore, trustworthiness can be seen from different viewpoints i.e., society, ethical, law, and technical points of view. In other words, the issue with trustworthiness is that it stems from trust, which individuals experience in diverse ways. This lack of a precise definition hinders the specification and verification of AI-based systems.

B. Difficult to Assess

There have been many attempts at measuring trust, and many of them are based on arbitrary ideas on what factors influence trust. Adams et al. [21] attempted to measure the trust in automated systems by considering the factors such as reliability, transparency, level of automation, usability,

security, appearance, predictability, susceptibility, and reputation of the system designer. Cahour et al. [22] considered factors such as predictability, reliability, and efficiency, and Muir [23] considered factors such as persistence, performance, and responsibility. This lack of consensus on the sub-attributes that affect the trustworthiness of the system makes it complex and difficult to measure the trustworthiness of an AI system.

C. AI Bias

There is a growing awareness of bias in AI systems, and its effects on results [24]. This bias also known as algorithmic bias is experienced when a systematically incorrect result is produced by a machine learning algorithm. An example of such bias includes an instance where judges used an AI system to set paroles and the algorithm showed a bias toward Black defendants in terms of the likelihood of committing an offense [25]. Another example includes a study that showed how gender biases can be embedded in text [26]. Just as various articles and journals reflect their writers' biases, a machine learning algorithm can reflect the biases of their creators. Bias in an AI system reflects the data algorithms, data blending methods, model construction practices, interpretation, and application of results that their developers chose to use and are driven by human judgments [27]. This bias leads to a lack of trust in the decisions made by AI systems.

D. Responsibility and Accountability

The potential harm that can be caused by a decision made by an AI system increases with the increase in the adoption of AI systems. For example, in health care, the current practices used in moral accountability and safety practices have not adapted to the introduction of AI-based clinical tools [28]. There is, therefore, a growing concern about who is accountable and responsible for the results of a decision made by an AI system. Ananny et al. [29] attempt to focus on ways to make

algorithms transparent and explainable enough to properly pinpoint the accountability for the harm caused by an algorithm. Furthermore, it is difficult to determine what level of control given to an AI system operator is enough to make him/her accountable. Assistive AI machines can provide recommendations; however, a user can never be certain that the system's conclusions are in line with their intention.

E. Lack of explainability

To achieve their remarkable performances, various AI systems are based on training models using large data sets or reinforced learning methods. However, due to the use of such models, it is difficult to understand what underlying processes and principles were used by the system to come to a decision [30]. These models are, therefore, deemed black boxes, as it is not clear what made them arrive at a decision This makes it difficult to verify the AI systems decisions and brings about uncertainty about the decision made by the AI system. This makes it difficult for a user to trust an AI system. There is currently work being done in the field of explainable AI to address this concern. For example, Fuji et al. [31] attempt to provide explainability using knowledge graphs, and Baehrens et al. [32] attempt to use sensitivity analysis to provide explainability.

F. Value Alignment

Value alignment refers to making sure the behavior and decisions of AI systems are properly aligned with the values of humans. The issues in value alignment can be separated into two parts; the first part deals with the technical aspect of how an AI system can properly be constrained or trained to follow the principle and values that a user deems ethical and proper. The second part deals with the variety of cultures, societal perspectives, and individual preferences, and it is difficult to decide which preferences the AI system should be aligned towards [33].

2.4 Quality Attributes

A software system can be specified by functional and non-functional requirements. Functional requirements describe what a system must do while non-functional requirements describe how the system behaves and exhibits quality attributes [34]. For example a non-functional requirement for the quality attribute, performance, could be that a system should respond to a user's input within 0.2 seconds. A quality attribute is a property of a system and characterize what a system has, for example, availability is the property that a particular software is there and ready to complete a task when needed, and security is the system's ability to prevent unauthorized access to data while providing access to authorized entities [35]. Quality attributes are system-wide properties and therefore are determined by a systems software architecture [36]. They are not simply just achieved but satisfied within a context of a given scenario [37]. Often, they come into conflict with each other, and tradeoffs between quality attributes must be determined.

2.5 Quality attribute Scenarios

Quality attributes can be difficult to describe and evaluate as there have been numerous definitions and taxonomies used to describe them. They tend to be vague and there are no universal or simple measurements for most attributes. However, this issue can be solved with the use of scenarios that provide a more concrete description and specification of what the quality attribute means in the system. They allow stakeholders to view quality attributes in a more specific manner, through the context of system use [38] and enable developers to analyze a software architecture in terms of how close a scenario, and therefore a quality attribute, is satisfied.

Scenarios can vary in breadth and scope. However, Bass et al [35] consider six characteristics that are considered important in a scenario for specifying quality attributes. These characteristics include stimulus, stimulus source, response, response measure, environment, and artifact.

Stimulus- This refers to the event that occurs and causes the scenario to occur. For example, a stimulus for usability could be a user with the desire to learn how to use the system, and a stimulus for security could be an attack on the system.

Stimulus source - The stimulus source is where the stimulus comes from and depending on the source the system may respond differently. for example, the system may have different security measures to an external entity than from an internal entity. An example, stimulus source for availability can be the hardware or software.

Response - The response refers to how the system responds to a specific stimulus. These responses can involve run-time responsibilities or development-time responsibilities that should be performed when a stimulus occurs. For example, in a useability scenario, the stimulus would be a user's desire to use the system efficiently, and the response would be the system providing features necessary to do so.

Response measure - This is used to determine whether a response to the stimulus is satisfactory or not i.e., whether the response has satisfied the requirement. For example, for usability, it could be the length of time it takes for a user to learn how to use the system.

Environment - The environment refers to the context or the state of the system in which the scenario is taking place. This is important because there are situations where the response to a stimulus can change depending on the circumstances that the system is currently in. For example, in availability, the first failure of a system may be treated differently from the fifth consecutive failure.

Artifact - This refers to the part of the system involved in the scenario. Most of the time it involves the entire system but during certain scenarios, only certain portions of the system may be involved. For example, a failure in one part of the system may be more critical than other parts of the system and would have to be treated differently. Table 2.1 below summarizes the parts of a scenario.

Table 2.1 Parts of a scenario

Parts of scenario	**Description**
Stimulus	The event that requires a response
Source of stimulus	The entity in the scenario that brings about stimulus
Environment	Context and conditions of the scenario
Artifact	Part of the system that participates in the scenario
Response	The response that is given by the system in the scenario
Response Measure	The measure of response to test whether the requirement is met

2.6 Design Tactics

Tactics are architectural design decisions that can be used to achieve a desired quality attribute by influencing the response of the system to a particular stimulus [35]. They are used to transform the software architecture of the system and are a means to bring the measure of a quality attribute

closer to a desired goal. Tactics come in various forms and are used to describe solutions for a range of quality attribute concerns. For example, availability tactics provide solutions for detecting faults, and performance tactics provide solutions for managing resources. In a tactic, tradeoffs with other quality attributes are not considered, and instead, it is the responsibility of the designer to consider and control the tradeoffs between other quality attributes. A tactic aims to improve a particular element of a quality attribute. For example, one concern in modifiability is to reduce coupling between modules, and a tactic that can be used to achieve it is encapsulation. Tactics are realized as a part of a pattern along with other structures in the system's software architecture. Like architectural patterns, tactics can shape the software architecture of a system, but they are concerned with only a single quality attribute while patterns are concerned with multiple attributes. This allows a software architect to have more control over the design [35].

2.7 Utility Trees

A utility tree is a top-down approach to help prioritize and make the quality goals of a system more definitive. It represents the decomposition of the stakeholders' goals for the architecture of the system. A utility tree tends to consist of a main root labeled "Utility." Its second level then consists of the main quality attributes which be broken down into subcategories on the third level. The leaves of the utility tree consist of scenarios. Figure 2.1 below shows an example utility tree.

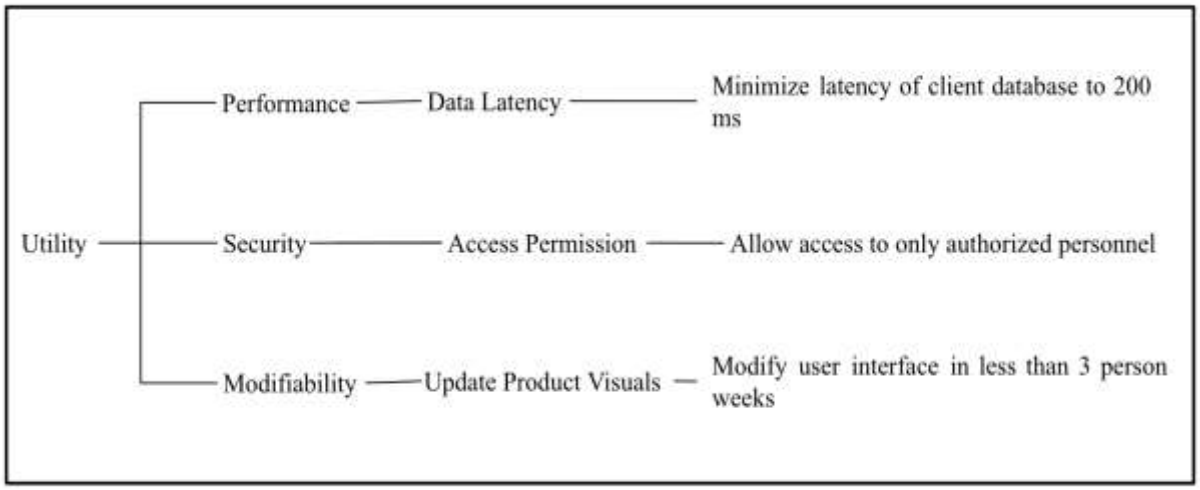

Figure 2.1 Sample utility tree

2.8 Architecture Tradeoff Analysis Method

The Architecture Tradeoff Analysis Method (ATAM) is a method used to analyze the software architecture of a system [11]. It provides insight into how well a software architecture supports certain quality attribute goals such as availability, performance, and security, and provides insights into how these goals relate to each other. The goal of ATAM is to understand how a system's architectural designs affect its quality attribute requirements. It provides several benefits that include raising awareness of various aspects of the architecture, identification of risk in the architecture, increase in the amount of documentation, insight into tradeoffs between attributes, eliciting the main attribute goals of the architecture, and refinement of architectural design decisions.

There are 9 steps to ATAM, though, it should be noted that in most cases the steps of ATAM are not strictly followed, and it is the type of system, the customer’s needs, time constraints, and the phase of the development process for the system that determines what steps are followed and to what extent they are followed [11].

The steps of ATAM are as follows:

- Present ATAM
- Present Business Drivers
- Present Architecture
- Identify Architectural Approaches
- Generate Quality Attribute Tree
- Analyze Architectural Approaches
- Brainstorm and Prioritize Scenarios
- Analyze Architectural Approaches
- Present Results

Present ATAM

In this step, ATAM is introduced to the stakeholders, which include the customers, the architectural team, managers, developers, and testers. The stakeholders are introduced to the steps of ATAM and told what is to be expected during the process.

Present Business Drivers

In this step, the business goals and the main architectural drivers of the system are introduced to the team. This includes informing the team about the time to production, the level of security of the system, the budget constraints, etc.

Present Architecture

In this step, the proposed architecture is introduced to the stakeholder. It is described, and stakeholders are shown how it affects the business goals of the system.

Identify Architectural Approaches

In this step, the different architectural approaches are introduced but no analysis takes place.

Generate Quality Attribute Trees

In this step, a utility tree that identifies the system's driving quality attributes is generated. These attributes are refined into scenarios with stimuli and responses and prioritized.

Analyze Architectural Approaches

In this step, architectural approaches are discussed and analyzed based on how they can support the driving quality attributes identified in the previous steps. Furthermore, various architectural risks, factors, and tradeoffs that affect the goals of the system are discussed.

Brainstorm and Prioritize Scenarios

In this step, the scenarios generated in the utility tree are prioritized as well as discussed. The scenarios can then be prioritized using a vote among all the stakeholders.

Analyze Architectural Approaches

This step follows the same processes as the preceding analysis step, but the scenarios identified and prioritized in the previous steps are considered in the analysis. The scenarios are used to identify other architectural approaches, risks, tradeoffs, and crucial factors that may affect the system's goals.

Present Results

In this step, the results of the ATAM are presented to the stakeholders and depending on the requirement a report can be made that records the information of the ATAM. The results can include the scenarios, the resulting software architecture, various concerns that have been identified, the utility tree, tradeoffs, artifacts developed, etc.

CHAPTER 3

LITERATURE REVIEW

In this chapter, we discuss previous work that has been done in the field of trust and trustworthiness of AI systems. We discuss the goals of the papers and their contributions to the field. Furthermore, we discuss the differences in their work in comparison to this paper and any concerns, drawbacks in the papers, or areas where the papers may be lacking.

3.1 The relationship between trust in AI and trustworthy machine learning technologies

In this paper [16], the authors relate the idea of trust from the view of social sciences to the technologies proposed for the trustworthiness of AI-related services and products. They do this by providing a systematic approach that is based on the Ability, Benevolence, Integrity, Predictability (ABI+) framework and mapping the framework with related technology qualities that support trustworthiness. The ABI framework considers that the assessment of the trustworthiness of a party is affected by ability, benevolence, and integrity. The authors aim to identify how the latter trustworthiness technologies affect trust in AI. They also show a relationship between trustworthiness technologies and ethical or similarly related research areas.

The authors also introduce the concept of the "chain of trust" which defines trust as having a connection between various stages of the machine learning pipeline. This includes the data, preparation, feature extraction, training, testing and inference, and expansion. The chain of trust gives an insight into how deep and far-reaching qualities of trust can affect the trustworthiness of an AI system. The authors considered four trustworthiness factors: fairness, explainability, auditability, and safety (FEAS) and attempted align the discussion of trustworthiness factors with principled AI frameworks in the literature, however, they stress that these factors will not make

the system completely trusted as they are not the only factors that affect trust. Unlike this thesis, which focuses on the initial stages of the development of an AI system, the authors considered trust from a temporal point of view i.e., that trust consists of initial trust and continuous trust and draws from techniques in the social sciences to support trust in AI systems.

3.2 Building a Trustworthy Explainable AI in Healthcare

In this paper [10], the authors attempt to deal with the issue of how the lack of transparency in the decision-making of AI algorithms may cause a lack of acceptance, accountability, and trust in AI in healthcare. They stress how important these issues are, as the lives of patients are dependent on decisions that the AI makes. In order, to support the trustworthiness and explainability of AI, they propose a framework that would support these qualities in the sector. The motivation for their proposal is that there is still a large amount of distrust in AI technologies in healthcare among not only medical professionals but also the public. This distrust stems from the lack of understanding of how AI technologies work. The framework they propose consists of two components: explanation characteristics and human-machine trust which is divided into cognitive-based trust and affective-based trust. The focus of this paper has a narrower scope of trustworthiness by focusing on explainability, in comparison to this thesis, which has a more general view of trustworthiness. However, the paper provides useful insight into how different fields will have different trustworthiness requirements.

Furthermore, the authors stress the importance of explainability, which is associated with AI systems due to their black box nature and provide a breakdown of what characteristics constitute a useful explanation. For example, they show that explainability may be broken down further i.e.,

explanations are contrastive, domain-dependent, social, truthful, thorough, and general. The paper does however fall short of the usefulness of the framework, which remains to be seen.

3.3 A Metric Model for Trustworthiness of Software

In this paper [39], the authors establish a metric model for the trustworthiness of a software system. The authors state that trustworthiness is a description of the behavior of a software system when completing a task and that it consists of several attributes. With the latter in mind, their metric model is based upon the idea that trustworthiness consists of other attributes of the system and is given a value in the range of 1 to 10. Furthermore, these attributes are divided into two distinct categories namely: critical and non-critical attributes. Critical attributes are deemed to be more important than non-critical attributes. This is done based on the idea that the effect on the trustworthiness of a software system differs depending on the attribute.

The metric model proposed satisfies four criteria that are based on common properties of software attributes. These criteria include monotonicity, acceleration, sensitivity, and substitutivity. Monotonicity is based on the property that as the level of one attribute goes up the trustworthiness also increases. Acceleration is based on the property that the effect of an increase of an attribute on the trustworthiness decreases as the attribute increases. Sensitivity describes the level of effect on the trustworthiness of a system when the associated attribute increases. Substitutivity describes how, given two attributes, if one attribute replaces the other the trustworthiness does not change. The metric model is shown below:

$$T = \frac{10}{11} \min_{1 \le i \le m} \left\{ \left(\frac{y_i}{10} \right)^{\in} \right\} y_1^{\alpha\alpha_1} y_2^{\alpha\alpha_2} \dots y_m^{\alpha\alpha_m} + \frac{10}{11} y_{m+1}^{\beta\beta_{m+1}} y_{m+2}^{\beta\beta_{m+2}} \dots y_{m+s}^{\beta\beta_{m+s}} \quad (1)$$

Where T is the trustworthiness level, α represents the portion of critical attributes and β represents the portion of non-critical attributes, m represents the number of critical attributes, s represents the

number of non-critical attributes, $y_1 \ldots y_m$ represent the critical attributes, and $y_{m+1} \ldots y_{m+s}$ represent the non-critical attributes. The metric model provides a level or rank of trustworthiness. Like this thesis, the paper views trustworthiness as consisting of more than one factor. It provides insights into how trustworthiness may be measured, albeit with a single metric. However, the paper fails to describe how critical and non-critical attributes are obtained and divided into their respective categories. As such, its practical usefulness remains to be seen in future work.

3.4 Trustworthiness Attributes and Metrics for Engineering Trusted Internet-Based Software Systems

This paper [40] aims to provide an extensive list of software quality attributes that contribute to the trustworthiness of internet-based software systems. Furthermore, they propose methods to obtain metrics to measure the trustworthiness of the system. In their paper, they state that most literature approaches trustworthiness from a security point of view. However, security is not the only characteristic of a trustworthy system. They describe trustworthiness as a multidimensional construct consisting of a broad spectrum of characteristics such as reliability, security, performance, etc., and that imbalances between a user's level of trust and the trustworthiness of a system can cause various issues. Their detailed survey revealed a set of more than 15 attributes and sub-attributes.

The authors describe trust from both a sociological and security perspective and then define trustworthiness and how it relates to trust. In their paper, the authors define trust as a bet about the future contingent action of other individuals or groups. Furthermore, the authors also propose using the Goal Question Metric Approach to develop metrics for the trustworthiness of the system. However, they fail to properly elaborate on how this would be done. The positions taken by the

authors of the paper are similar in nature to how this thesis approached the idea of trustworthy AI, however, the scope of their paper is on the trustworthiness of internet-based systems. Similarly, we consider that the trustworthiness of an AI system is dependent on more than just security, but dependent on several characteristics or attributes and do a detailed review to discover those attributes. The authors in the paper fail, however, at providing a more detailed description of how this should be done for a particular system.

CHAPTER 4

METHODOLOGY

In this chapter, we illustrate and describe the sources and criteria for the identification of the sub-attributes that affect the trustworthiness of AI systems, the identification of the general characteristics of trustworthiness scenarios, and the identification of the tactics that can be used to achieve trustworthiness goals.

4.1 Sources

The identification of trustworthiness sub-attributes, characteristics of the trustworthiness general scenario, tactics involved the examination of research papers from reputable sources to find papers that fell under the scope of trustworthy AI and trustworthiness. The research obtained comes from reputable sources of archival research such as the IEEE, ACM digital library, Science Direct, Springer, Wiley Interscience, and IBM. A selection process is then applied to narrow the results to only relevant papers.

4.2 Selection criteria for trustworthiness sub-attributes

Quality attributes can be composed of other sub-attributes, for example portability can be broken down into adaptability, installability, replaceability [41]. Sub-attributes contribute and combine to achieve a particular quality attribute goal. As shown in chapter 2, the trustworthiness of an AI system can have various sub-attributes depending on the context and AI system being developed. The goal of this section was to identify the various sub-attributes that can trustworthiness of AI systems can be composed of in the literature.

The selection of papers for trustworthiness sub-attributes process involved the following criteria:

- Selection of peer-reviewed and published papers that relate to the trustworthiness of AI.

- Selection of papers that defined trustworthy AI and where it stands.
- Selection of papers that related to the sub-attributes of trustworthy AI.
- Selection of papers that relate to trust in AI.
- Papers published in 2000 -2022.

4.3 Selection criteria for trustworthiness general scenarios for AI systems

Once the sub-attributes of trustworthiness were identified, we did a query of the literature for various scenarios that related to the trustworthiness of AI. We did this to obtain a general scenario for AI trustworthiness cases. General scenarios allow for abstracting various situations of concern in software projects and provide good coverage of scenario instances. Although due to the nature of abstraction, details can be lost, the abstraction is needed to cover most, if not all, scenarios. General scenarios can be used as guidelines or checklists for creating utility trees in methods such as ATAM by helping in the identification of scenarios that may specify a particular quality attribute and reducing the chance that a scenario is overlooked [42]. They can also be applied to most systems because various systems tend to have similar components to them [43]. Software architecture evaluation methods such as the architecture tradeoff analysis method (ATAM) rely heavily on building scenarios to compare how and to what extent various quality attributes are met by candidate architectures and these scenarios tend to be instances of general scenarios. The selection process involved the following criteria in addition to the criteria in the previous section:

- Selection of papers that related to the sub-attributes of trustworthy AI identified in the search for trustworthiness sub-attributes.
- Selection of peer-reviewed and published papers that relate to the use of AI systems.

4.4 Selection criteria for trustworthiness tactics for AI systems

For a human to have trust in an AI system the system must show some qualities that make it trustworthy, and this can be achieved using tactics. Tactics are architectural design decisions that can be used to achieve a desired quality attribute by influencing the response of the system to a particular stimulus [35]. They are used to transform the software architecture of the system to bring the measure of a quality attribute closer to a desired value, and they affect various aspects of a system's software architecture, such as its properties and structure. A primary element of trust is uncertainty [44][14] and in AI systems this uncertainty stems from the black-box nature of most AI. For example, there is uncertainty as to what the AI values, uncertainty in how a decision was made by the AI, uncertainty in what biases the AI has, and uncertainty in how susceptible to outside influence the AI system is. This can be applied to trust in AI, and therefore the tactics for trustworthiness should deal with reducing this uncertainty. One way to think about tactics for the trustworthiness of an AI system would be to think about how one could get a human to trust another human. Trustworthy humans have certain characteristics such as being honest, fair, dependable, and difficult to manipulate, being clear about what is going on, behaving the way people want, and not hiding any relevant information. In addition to criteria in the previous sections, the selection process involved the following additional criteria.

- Selection of peer-reviewed and published papers that relate to methods of achieving trustworthiness in AI system.
- Selection of papers that related to achieving the sub-attributes of trustworthy AI identified in section 4.2.

CHAPTER 5

RESULTS

In this chapter, we show the results of our query of the literature. Section 5.1 shows the trustworthiness sub-attributes from the literature, section 5.2 shows a general scenario for the trustworthiness of an AI system, section 5.3 shows the trustworthiness tactics from the literature, section 5.4 provides a design checklist for trustworthiness.

5.1 Trustworthiness sub-attributes

In this section, we show the resulting trustworthiness sub-attributes identified in the literature. We show the number and variety of sub-attributes that can affect the trustworthiness of a system, as well as the papers that indicate a particular sub-attribute of trustworthiness. Table 5.1 below shows the sub-attributes of the trustworthiness of AI systems according to a review of the literature. The papers can be found in Appendix I. In table 5.1, the papers are represented by [PS-#]. For example, [PS-1], represents the first paper. These papers can be found in Appendix II.

Table 5.1 Identified sub-attributes of trustworthiness from the literature.

Attribute	**Paper**
Interpretability	[PS-1], [PS-2], [PS-5], [PS-6], [PS-20], [PS-15], [PS-20], [PS-33]
Security	[PS-1], [PS-2], [PS-8], [PS-11], [PS-17], [PS-23], [PS-27], [PS-29], [PS-30], [PS-31], [PS-32], [PS-33], [PS-35], [PS-37], [PS-40], [PS-42]

Human control	[PS-2], [PS-19]
Robustness	[PS-2], [PS-8], [PS-10], [PS-11], [PS-12], [PS-13], [PS-23], [PS-25], [PS-27], [PS-29], [PS-31], [PS-33], [PS-35], [PS-39]
Reliability	[PS-2], [PS-11], [PS-19], [PS-22], [PS-24], [PS-25], [PS-33], [PS-35], [PS-40], [PS-42], [PS-43], [PS-45]
Explainability	[PS-2], [PS-3], [PS-4], [PS-6], [PS-7], [PS-12], [PS-13], [PS-15], [PS-20], [PS-21], [PS-22], [PS-23], [PS-24], [PS-26], [PS-27], [PS-28], [PS-29], [PS-30], [PS-34], [PS-36], [PS-37], [PS-38], [PS-41]
Fairness	[PS-2], [PS-4], [PS-6], [PS-8], [PS-9], [PS-10], [PS-11], [PS-12], [PS-13], [PS-15], [PS-16], [PS-18], [PS-23], [PS-25], [PS-26], [PS-27], [PS-29], [PS-30], [PS-33], [PS-37], [PS-41]
Auditability	[PS-4], [PS-24], [PS-30], [PS-40]
Safety	[PS-4], [PS-6], [PS-10], [PS-16], [PS-17], [PS-18], [PS-25], [PS-31], [PS-33], [PS-35]
Accuracy	[PS-4], [PS-6], [PS-13], [PS-19], [PS-23]
Efficiency	[PS-4], [PS-43]

Performance	[PS-4], [PS-6], [PS-35], [PS-44]
Traceability	[PS-5], [PS-37]
Transparency	[PS-5], [PS-6], [PS-7], [PS-9], [PS-10], [PS-11], [PS-12], [PS-13], [PS-23], [PS-24], [PS-26], [PS-27], [PS-29], [PS-33], [PS-34], [PS-37], [PS-39], [PS-42]
Provenance	[PS-6]
Explicability	[PS-8], [PS-11], [PS-16], [PS-18], [PS-31], [PS-34]
Accountability	[PS-8], [PS-10], [PS-12], [PS-13], [PS-15], [PS-23], [PS-26], [PS-27], [PS-29], [PS-39], [PS-41]
Equality	[PS-8]
Responsibility	[PS-8], [PS-19], [PS-44]
Liability	[PS-8]
Human autonomy	[PS-8], [PS-16], [PS-18], [PS-32]
Human agency	[PS-10]

Human oversight	[PS-10], [PS-11]
Privacy	[PS-2], [PS-10], [PS-35]
Data governance	[PS-10]
Societal wellness	[PS-10]
Lawful	[PS-11], [PS-14], [PS-27]
Ethical	[PS-11], [PS-13], [PS-14], [PS-15], [PS-26], [PS-29]
Unbiased	[PS-11], [PS-26], [PS-27]
Verifiability	[PS-12], [PS-29]
Sustainability	[PS-12], [PS-29]
Ease of use	[PS-15], [PS-22], [PS-35], [PS-40], [PS-42]
Correctness	[PS-17]
Dependability	[PS-19]

Predictability	[PS-19], [PS-40], [PS-42], [PS-43]
Fidelity	[PS-20]
Accessibility	[PS-22]
Reproducibility	[PS-24]
Learnability	[PS-24]
Generalization	[PS-29]
trusted for human-machine interaction	[PS-31]
Beneficence	[PS-32]
Resilience	[PS-39]
Validity	[PS-40]
Compatibility	[PS-40]
Trialability	[PS-40]
Representation	[PS-40]

Perception	[PS-40]
Sociability	[PS-40]
Collaboration	[PS-40]
Level of automation	[PS-42]
Appearance	[PS-42]
Susceptibility	[PS-42]
Reputation	[PS-40], [PS-42]
Persistence	[PS-44]
Understandability	[PS-45]
Faith	[PS-45]
Personal attachment	[PS-45]
Competence	[PS-45]

The table shows 56 terms from the literature to describe the sub-attributes of the trustworthiness of an AI system. These are numerous and varied and different AI systems will have varying sub-attributes that would be needed to be considered trustworthy. Moreover, some of these sub-attributes can refer to the same thing, albeit with a different name perception and appearance. Even still, these sub-attributes are useful in that they allow one to see the potential factors that can affect the trustworthiness of an AI system, and it is up to the stakeholders to determine what sub-attributes are needed for a particular AI system to exhibit trustworthiness.

5.2 Trustworthiness general scenarios for AI

In this section, we introduce a general scenario for trustworthiness and describe its characteristics and possible values for those characteristics. As stated in the previous sections, general scenarios allow for the abstraction of scenarios of a particular quality attribute. Following Bass et al [35], we considered six characteristics that are important in a general trustworthiness scenario for specifying quality attributes.

A. Stimulus for trustworthiness scenarios

The stimulus describes what causes the trustworthiness scenario to occur. There can be multiple stimuli for a trustworthiness scenario. These include the following:

- An adversarial attack on the AI system to manipulate the decision-making process of the AI.
- A user wants to understand why the AI made a certain decision
- Bias being detected within the data used to model the AI
- A user wants to know the most principal factors for a decision made by the AI

B. Source of Stimulus for trustworthiness scenarios

Sources of stimulus for trustworthiness can either be human or another machine. They can be both internal sources or external sources in the case of an attack on the system, for example, in an adversarial attack, the attacker may be known or unknown and can also be a human from within the organization. In the case of a human user wanting to know how to use the system, the stimulus would be the human.

C. Environment for trustworthiness scenarios

The environment for a trustworthiness scenario describes the state that the AI system would be in. For example, the state of the AI system would be whether the system is currently operating as normal, currently going through an adversarial attack, or in a state after an adversarial attack.

D. Artifact

The artifact relates to the portion of the AI system to which the scenario applies. The artifact could be the entire system; however, this is not always the case. For example, in a trustworthiness scenario, an adversarial attack on the data model is treated differently than an attack on the GUI of the system. Furthermore, the response to the attack may be prioritized differently based on the portion of the system being affected.

E. Response

The response is how the AI system responds to the stimulus in the scenario. Much like other systems the response can consist of the responsibilities of the AI system during runtime or developers during development time. For example, a response to the user who wants to see how a

decision was made for classifying an image would be to show the pixels or areas on the image that were the significant factors in a decision.

F. Response measure

The response measure is what is used to determine whether a requirement is fulfilled. In the case of a trustworthiness scenario, the response measure could be the amount of time it takes for a user to understand the explanation for a decision the AI system made. For an adversarial attack, the response measure could be the time it takes to detect an attack or correct the data model of the AI after an attack.

Table 5.2 below shows the trustworthiness general scenario, and table 5.3 shows an example of a complete trustworthiness scenario that is an instance of the general trustworthiness scenario.

Table 5.2 Trustworthiness general scenario

GENERAL SCENARIO
Source of stimulus: Internal; human, machine, software; external; human, machine, software. *Stimulus*: Attack where an external source provides deceptive inputs for the data model, the user looks for reasons for the AI decision, the user is uncertain about the decision. *Artifact:* Data model, complete system, vulnerable components of the system, part of the system being interacted with. *Environment:* Adversarial attack on the system has been detected, normal operation, a user request for explanation is sent, an attack has occurred, the system is currently being attacked *Response:* The data model is corrected to a state before the attack, the system provides appropriate information to the user, an attack is detected, an attack is prevented.

Response Measure: The data model is restored to a previous percentage accuracy, time taken for the user to understand AI output, time taken for the system to recover from an attack, time taken to detect an attack.

Table 5.3 An example trustworthiness scenario

SCENARIO
Scenario: The AI system has undergone an adversarial attack and the data model is restored *Source*: External machine *Stimulus*: External machine provides deceptive inputs for data model *Artifact*: Data model *Environment*: Adversarial attack on system has been detected *Response*: Data model is corrected to state before attack *Response* Measure: Data model is restored to previous percentage accuracy

5.3 Trustworthiness Tactics for AI

5.3.1 Identified Trustworthiness Tactics for AI

Based on the literature, the goal of trustworthiness tactics is to increase the trust a user has in the system by reducing their uncertainty about it. Uncertainty about an AI system occurs when a user is unsure of the decision-making of the AI system. Therefore, trustworthiness tactics are designed to address this uncertainty in the system so that a user is more willing to trust the AI system and its decisions. The tactics in this thesis reduce uncertainties in the system by providing a reason that reduces that uncertainty. Figure 5.1 below illustrates this goal.

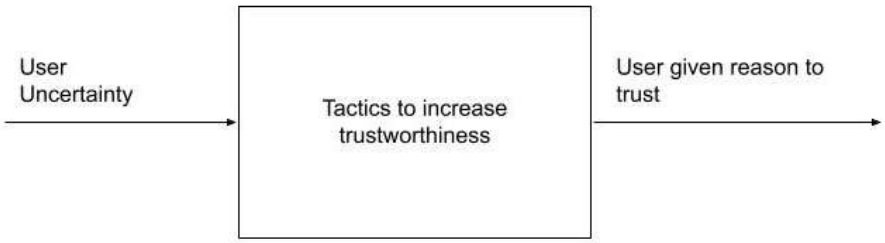

Figure 5.1 Goal of trustworthiness tactics

We can categorize trustworthiness tactics for AI systems as follows; support user understanding, align behavior, reduce bias, and robustness against attacks**.**

Table 5.4 below shows the papers used in the identification of trustworthiness tactics. They are categorized by the type of tactics used or suggested. In table 5.4, the papers are represented by [S-#]. For example, [S-1], represents the first paper. These papers can be found in Appendix II. These papers are categorized into the trustworthiness tactic category they fall under and then are further separated into the tactic used or suggested in them. For example, paper [S-13] falls into the tactic category of supporting user understanding. In the paper, they address this by suggesting the AI system provides the user with a reason as to why a decision was made.

Table 5.4 Identified categories for trustworthiness tactics

Tactic	**Paper**
Support user understanding	
Show user reasons for decision	[S-3], [S-13], [S-17], [S-2], [S-4], [S-6], [S-5], [S-10], [S-11], [S-9], [S-12], [S-16]
Human understandable model	[S-18], [S-7], [S-8], [S-14], [S-15], [S-31]

Align behavior	
Detect user preferences	[S-1], [S-2], [S-82], [S-85], [S-78], [S-83]
Model user preferences	[S-79], [S-81], [S-41], [S-42]
Verify alignment	[S-80], [S-44]
Reduce bias	
Detect bias	[S-26], [S-27], [S-40], [S-46], [S-47], [S-84]
Remove bias	[S-23], [S-24], [S-29], [S-30],[S-25],[S-19], [S-20], [S-21], [S-26],[S-27], [S-40], [S-46],[S-48] [S-49], [S-50], [S-86], [S-87], [S-88], [S-42], [S-43], [S-45]
Select less biased model	[S-22], [S-28]
Robustness against attacks	
Mitigate attack	[S-52], [S-53], [S-54], [S-55],[S-56], [S-57], [S-58], [S-60] [S-62], [S-63], [S-64], [S-65], [S-66], [S-67], [S-68], [S-70] [S-71], [S-72],[S-73], [S-75], [S-76]
Detect attack	[S-32], [S-33], [S-34], [S-35], [S-36], [S-37], [S-38], [S-39], [S-51], [S-59], [S-61], [S-74], [S-77]
Recover from attack	[S-69]

5.3.2 Trustworthiness Tactics

The trustworthiness tactics are summarized in figure 5.2 below which shows the categories of trustworthiness tactics and their corresponding tactics.

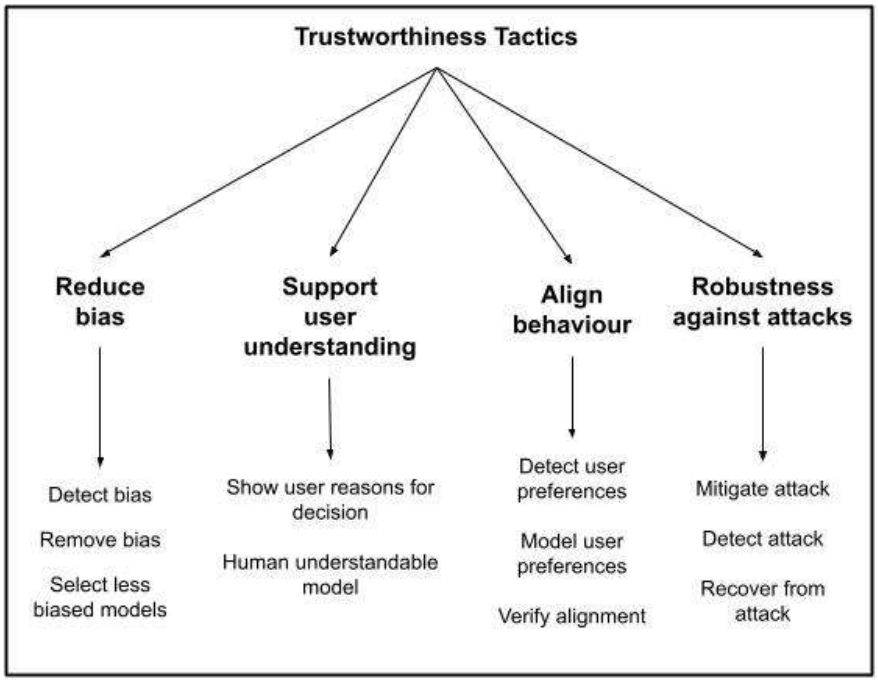

Figure 5.2 Categories of trustworthiness tactics and their tactics

Figure 5.2 shows the four categories of trustworthiness tactics namely: reduce bias, supporting using understanding, align behavior. The tactics for reducing bias include detect bias, remove bias and selection of less biased models. The tactics for support user understanding include show user reason for decision and human understandable models. The tactics for alignment of behavior include detect user preferences, model user preferences, and verify alignment. The tactics for robustness against attacks include mitigate attack, detect attack, and recover from attack. The following sections explain figure 5.2 in more detail.

5.3.1 Reduce bias

To be trusted the AI system should be fair and unbiased in its decision-making processes. The tactics that apply here are related to fairness and deal with making sure an AI system's decisions are fair and unbiased. Tactics that fall under making unbiased decisions include the following:

I. Detect bias
II. Remove bias
III. Select less biased models

Detect bias: This is the detection of data that could cause a model to become biased in its decisions. This data includes attributes such as race, gender, caste, religion, etc. The latter data are removed to reduce the impact of the attributes in the model and thus produce a more unbiased model. Some common ways to do this are to look for bias in datasets that are used to train the model [45] [46] or detect bias in the model after training [47] [48].

Remove bias: This is the removal of bias in an AI system to prevent biased decision-making by the system. The removal of bias can be separated into three categories: post-processing, pre-processing, and in-processing.

Pre-processing involves the removal of bias in the dataset used to train an AI model, for example, having constraints on sensitive attributes such as race and religion [49]. Another approach is the removal of commonly known biases that tend to show up repeatedly [50]. In-processing is the manipulation of a model to reduce biased decisions, for example, training a model to be less sensitive to some attributes for example gender or ethnicity [51]

Post-processing is the removal of bias in the output of the model, for example using other models to process a model's output [52].

Select less biased models: This is the process of selecting a less biased model. For example, after the creation of models from the data, the data models are compared, and the least biased model can be selected and enhanced [53].

5.3.2 Support user understanding

The capability of showing the user why an AI made a certain decision is an important part of increasing a user's trust in the AI's decision-making process, and making a user feel confident with a decision. Therefore, the tactics here focus on helping a user understand why an AI came to a decision. This can be done by showing a user the main factors that influenced a decision and making it easier for a user to understand the reasons why a decision was made. The tactics that fall under here include:

I. Show user reasons for decision
II. Human-understandable models

Show user reasons for decisions

For a user to understand why an AI system came to a decision, the AI system should show the user the reasons why it came to a particular decision. In this way, the user is better able to determine whether the AI system came to the correct conclusion and better able to determine whether the AI system is basing its decisions on the correct data. These explanations should be user understandable. This is because giving the user information that cannot be understood by the user has no use. Saliency maps [54], which show the most significant areas on an image that leads to an AIs decision, would, for example, allow a user to determine whether the AI came to the correct decision or at least see what it was about an image that made the AI come to a particular decision. Categorization of data can be used to allow users to understand complex data more easily. An

example of this is used in multi-level knowledge-guided Attention networks [55]. Furthermore, much like in human interactions, showing confidence in your decisions makes others believe that you are correct. Therefore, showing the level of confidence or the margin of error for a decision made by an AI system, for example with a percentage, can lead to a higher level of trust in the AI system.

Many AI models find patterns in data to make decisions. By showing the user what pattern was used to come to a decision, a user can then be aware and more confident in what and why the system made the decisions it did. One way to show patterns to a user is to use graphs to visualize them [56].

Contrastive explanations are used in various fields such as the health industry and criminology and can also be applied to AI. By showing the user the differences in data a user is much more able to understand and see how the AI system came to a certain decision. In the case of an image classifier, a user can be shown what should be absent to classify what it is [57].

Considering user knowledge is important because different users can have various levels of understanding and knowledge of certain fields and by considering a user's knowledge an explanation can be made more understandable. An embedding approach can be used to augment training data to include explanations from domain users [58].

Human understandable models

Most AI systems are black box systems that are complex and provide no insight into how certain inputs lead to a specific decision. A tactic to solve this complexity and black-box nature is to use more transparent and interpretable models i.e., models that are more understandable for humans. An example of this creating a more transparent model from a black box model i.e., creating a decision tree from a black box model [59].

5.3.3 Align behavior

Every human in the world has his or her preferences and an important part of getting a user to trust the AI system's decisions is whether an AI system behaves according to a user's preferences and ethics. These preferences may not only lie at the individual level but also societal level. The tactics under this category deal with aligning the decisions made by the AI based on the user's preferences. They are as follows:

I. Detect user preferences
II. Model user preferences
III. Verify alignment

Detect user preferences

In this tactic, user preferences are captured and learned by the AI system. When the user behaves in a certain way, the AI system should capture the behavior, and learn the user's preferred manners and type of decision-making. Techniques to do so include reinforcement learning, which allows an AI system to learn about a user's preference through observation of the user's behavior to learn certain constraints that are not shown in the data it was trained with [60]. Other ways to detect user preferences is to monitor users' behavior such as search history, or simply asking what their preferences are in a decision.

Model user preferences

In this tactic, a user's preferences and values are captured in the AI model, so that the AI systems decisions align with the user. For example, conditional preference networks (CP-nets) provide a way to model preferences and elicit optimized reasoning. In addition, they allow modeling priorities as well as optimization criteria [61].

Verify alignment

When a user's preferences and values have been learned or detected the system should be able to verify that the behavior of the AI system is aligned with the user's preferences and values [62]. Tactics used in detecting bias can also be used to detect and verify this.

5.3.4 Robustness against attacks

Adversarial attacks are algorithms that generate noisy data or manipulate inputs that cause an AI model to produce an incorrect result or decision [63]. The tactics here deal with handling such attacks and preventing the AI from being susceptible to such attacks. These tactics include:

I. Mitigate attack
II. Detect attack
III. Recover from attack

Mitigate Attack

These tactics are used to reduce the effects of an attack on the AI system. The most used is adversarial training. This is when an AI model is trained using adversarial examples or data and the resulting model is less susceptible to adversarial attacks. This type of training requires inserting noise or generating data that can trick the model into making the wrong decisions [64].

Detect Attack

These tactics are used to detect whether any of the datasets used to train the model have been manipulated. For example, the scanning of anomalous patterns in the data [65] and denoising with a denoiser, which involves the detection and removal of noise, such as Gaussian noise, in the data. A system can also have an ensemble of diverse denoisers that can be used since the noise added by the attacker is unknown to the defender [66].

Recover from Attack

Once an attack has been detected and removed, the system needs to recover. Attacks can lead to failures of the system and therefore tactics for recovery can be used for recovering from an attack [35]. These include having redundancy of the model, having, and keeping track of rollbacks, etc. Other tactics include using mode connectivity [67].

5.4 Trustworthiness Design Checklist

Following in the footsteps of Bass et al [35], we provide a design checklist to support the design and analysis of trustworthiness in a system.

A. Allocation of responsibilities

Determine what kind of trustworthiness responsibilities are applicable and may be needed for the AI system. Determine the parts of the system that require the highest levels of trustworthiness by determining parts of the system that the user may have uncertainty about, determining what parts of the AI system may be most vulnerable to attack, and determining the parts of the system where a user is blind to its internal workings.

Allocate system responsibilities that increase the trust that a user has in the system. These can include:

- Assisting the user in understanding an AI system's decisions
- Explaining an AI systems decisions
- Showing the user the ongoings of the AI system
- Detection of bias
- Reduction of bias in data
- Prevention of adversarial attacks

- Detection of adversarial entities
- Recovery from adversarial attacks

B. Coordination model

Determine what kind of trustworthiness responsibilities are applicable and may be needed for the AI system. Establish whether the systems coordination elements and mechanisms support those responsibilities. For example, determine whether the system allows for the delivery of an explanation after a decision is made by the AI system. Determine if the coordination system supports the prevention of attackers accessing vulnerable components. Determine whether the system coordinate supports the detection and quick recovery from an attack in a reasonable amount of time. Determine whether the coordination system itself is vulnerable to attacks or brings about new vulnerabilities to other parts of the system.

C. Data model

Determine the data portions that are applicable and are needed for the AI system its trustworthiness responsibility and in those portions determine the level of abstraction and appropriate level of access to data to support trustworthiness responsibilities, for example, determine how much detail in the user-perceivable data the user needs to know to feel confident in an explanation given and what data needs and should be to be accessed during training of the system or during run-time.

Establish that the data model is kept secure to prevent adversarial attacks. Ensure access to data is limited to authorized entities. Ensure data is stored in such a way that it can be restored after an attack, for example, by having a backup data model in case the data model is compromised.

D. Mapping among architectural elements

Determine which artifacts participate in the trustworthiness responsibility of the system, these responsibilities can include:

- Explaining an AI system’s decisions
- Showing the user the ongoings of the system
- Handling bias in the data
- Handling of attacks
- Determining how the extent of mapping and revealing certain elements of the architecture increases the vulnerability of the system to attacks.
- Determine how the mapping of the system affects the ease at which a user can understand what the AI system is doing.

E. Resource Management

Determine what necessary resources are needed to support the trustworthiness responsibilities of the system. Ensure that the level of resources does not reduce a user's level of trust in the AI system for example:

- Ensure there are sufficient resources for the system to recover from attacks in a reasonable amount of time.
- Ensure there are sufficient resources for the system to detect attacks in a reasonable amount of time.
- Ensure there are sufficient resources for the system to output an explanation in a reasonable amount of time.
- Ensure there are other resources left for other critical responsibilities of the system other than trustworthiness responsibilities.

F. Binding time

Determine what kind of trustworthiness responsibilities are applicable and may be needed for the AI system. Ensure that the strategies of how and when architectural elements are bound do not hinder the trustworthiness responsibilities and determine the appropriate strategies that support the trustworthiness responsibilities. For example, determine what parts of the system can be accessed during runtime and parts that do not need to be accessed to ensure safety against attacks. For example, the data model may not need to be changed during runtime and usage of the model and may only be accessed during certain periods such as when it is being trained. Determine how much and when a user or entity has control over the AI system during various times of the system's life cycle.

G. Choice of Technology

Determine the technologies that can help in achieving the trustworthiness responsibilities of the system. Make sure they support the trustworthiness scenarios of the system. Determine the technologies that can help detect, prevent, or recover from adversarial attacks. Determine the technologies that help a user in understanding an explanation for a decision made by the system. Ensure the technologies align with the trustworthiness responsibilities. Determine the trustworthiness of the technologies themselves. Make sure they do not hinder the trustworthiness scenarios of the system, for example, by bringing in more vulnerabilities to attacks, having a detrimental effect on the user's understanding of a decision, or affecting the response time of the AI.

CHAPTER 6

ANALYSIS OF TRUSTWORTHINESS WITH ATAM

In this section, we present an example of the design and analysis of the trustworthiness of a software architecture of a system using ATAM. The system used in the example is based on the AI system of a precision pollination robot [68]. The purpose of the robot is to automate the pollination of individual flowers in a greenhouse environment. In this example, we illustrate an overview of how the trustworthiness of an AI system can be analyzed using ATAM.

6.1 Phase 1

The first phase in the analysis is the introduction of ATAM to the stakeholders. In this step, ATAM is presented to the stakeholders, and they are told the purpose of ATAM, the expected information that is to be collected, the steps, and the expected outputs of the process.

6.2 Phase 2

The second phase involves the creation of a trustworthiness utility tree, the collection of information for architectural approaches, the prioritization of trustworthiness scenarios, and designing the architecture to support these scenarios.

6.2.1 Architectural documents

The architectural documents in the second phase consist of various views of the system. Figure 5.3 below shows a high-level view of the proposed software architecture for the system.

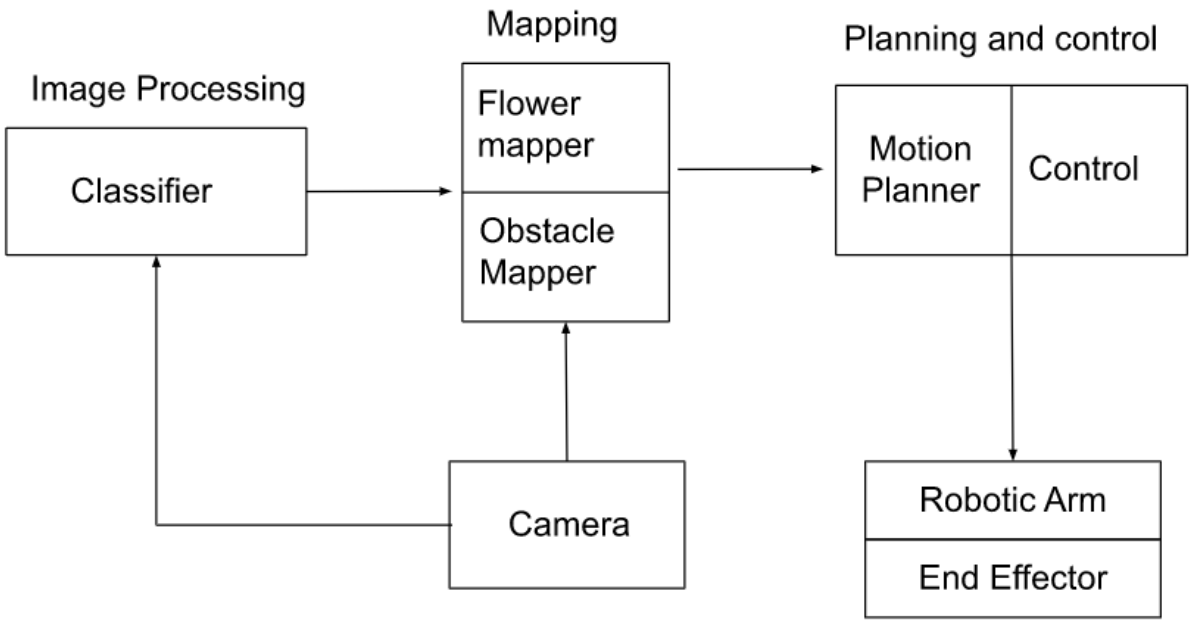

Figure 5.3 Proposed software architecture for robotics system

Figure 5.3 shows that the system is separated into four different components: image processing, mapping, planning and control, and manipulation. It also shows the communication paths between the components and hardware.

Image processing

The pollination robot should be able to detect and identify the post of each flower it comes across. This can be done through image processing in which images obtained from the depth camera are classified. A classifier is used to process the images and detect and separate flowers and non-flower parches. In addition, the classifier should be able to estimate the postures of the identified flowers.

Mapping

Once the images are processed a mapping is created from the processed images. Mapping consists of obstacle mapping and flower mapping. Obstacle mapping is done by the obstacle mapper and

involves mapping various obstacles and flowers to avoid collisions when moving around. Flower mapping is used to create a map of the postures of the flowers which is used when the robot is pollinating the flowers.

Planning and Control

After the creation of the obstacle map and flower map, a route is formed for the robot to follow and pollinate every flower. This route depends on the position of the flowers and obstacles as well as the posture of the flowers. The route should be the most efficient route that the robot needs to travel and allow for the most efficient motion that the robot arm needs to undergo for the end-effector to pollinate the flowers.

6.2.2 Architectural approaches

Based on the presented architecture, the approaches for the system are elicited by the stakeholders concerning trustworthiness scenarios and how best these approaches would support the trustworthiness scenarios. This is where various trustworthiness tactics can be considered to support the system's trustworthiness goals, as well as the design checklist to support the analysis.

6.2.3 Trustworthiness Utility Tree

A trustworthiness utility tree represents the overall trustworthiness of the AI system. The AI system’s architecture can then be analyzed in the next steps for its trustworthiness by considering how much it allows the trustworthiness scenarios to be possible. Based on the concerns of the stakeholders and what they consider a priority for the trustworthiness of the system, a trustworthiness utility tree can be created. Figure 5.4 below is an example portion of the utility tree for the subsystem showing the trustworthiness aspect. The stakeholders should ensure that the scenarios have all the characteristics of a scenario such as a response, stimulus, and environment.

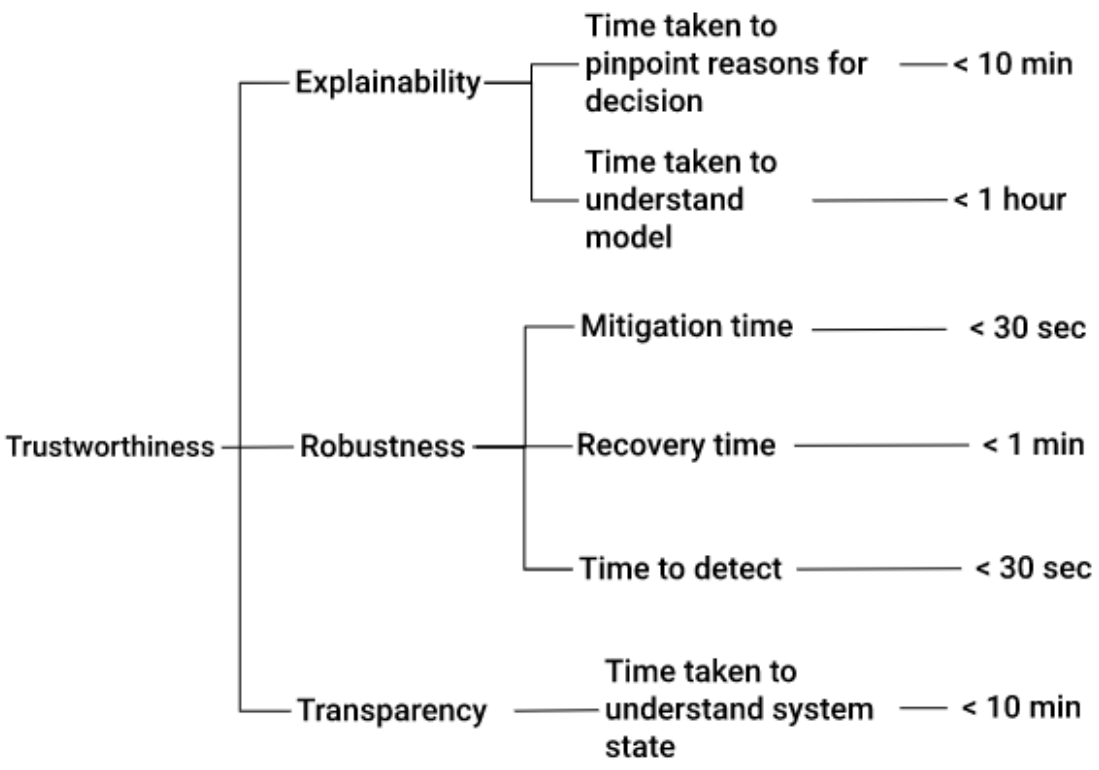

Figure 5.4 Trustworthiness Utility Tree

The trustworthiness utility tree in figure 5.4 shows the trustworthiness concerns of the stakeholders. These include explainability, robustness, and transparency. We can see the scenarios as leaves of the tree and the response measures showing what achieving these scenarios would mean. For example, one of the factors for the system to be considered trustworthy is that it must have robustness, which is achieved when the recovery time of the system is less than one minute, the time to detect an attack is less than 30 seconds, and the mitigation of the attack should take less than 30 seconds.

6.2.4 Analysis of architectural approaches

In this step, the stakeholders analyze the various trustworthiness architectural approaches. The stakeholders provide additional information about the various suggested approaches and their effects based on the utility tree. This is where various trustworthiness tactics can be considered to support the system's trustworthiness goals, as well as the design checklist to support the analysis. The stakeholders should discuss the risks of the approaches and compare the approaches for the benefits they provide and the tradeoff they bring about. Questions about the approaches will allow stakeholders to examine in greater detail how effective the approach is. These questions can include:

- How is an adversarial attack detected?
- How long does it take for the system to detect an attack?
- How can the system recover from an adversarial attack?

Following in the footsteps of [11], we can use the utility tree to see that the main attributes that affect the trustworthiness of the system are explainability, robustness against attacks, and transparency.

6.2.5 Trustworthiness Analytic Model

In this step, a model of the trustworthiness of the system as guided by the utility tree can be created as follows.

$$Qt = f(Qe, Qr, Qt) \quad (1)$$

Where the system's trustworthiness (Qt) is a function of explainability (Qe), robustness (Qr), and transparency (Qt). Using this model, the key characterization can be determined. For example, Qe could be refined into:

$$Qe = f(a1, a2, a3) \quad (2)$$

Where a1, a2, and a3 are factors that affect the explainability of the system. This modeling and analysis allow the stakeholders to discover the nature of any potential risks and benefits that could occur when changing a system's components and architecture. For example, one can see that the system's architecture currently does not show any trustworthiness qualities and considerations for alternate architecture should be made. These considerations can include using tactics such as:

- Creating a backup model to allow for recovery after an attack
- A graphical UI component to allow the user to understand the AI decision making
- A component to detect any adversarial entities

However, adding more components could increase the overhead of the system which can be detrimental to performance. As such, tradeoffs should be discussed and made as decided by the stakeholders and based on other modeling and analysis of other quality attributes of the system.

6.2.6 Analysis using scenarios

Scenarios allow more concrete description and specification of what the quality attribute means in the system and allow stakeholders to understand how various architectural approaches affect or achieve various quality attribute requirements. The figures below show example scenarios for the system. Using the general trustworthiness scenarios as a guide and methods such as a round-robin

and brainstorming can be used to elicit various trustworthiness scenarios. Table 5.5 below shows a sample trustworthiness scenario for the system with additional examples in appendix III.

Table 5.5 sample scenario

SCENARIO
Scenario 1: User tries to understand the explanation given by the AI system *Source*: User *Stimulus*: User request explanation of results *Artifact*: System graphical user interface (GUI) *Environment*: Normal operation *Response*: The system GUI provides an explanation *Response* Measure: User understands the explanation within 10 min

6.2.7 Prioritizing scenarios and analysis

Once scenarios have been elicited, they are prioritized in order of their importance through discussion with the stakeholders. One way to prioritize scenarios is to undergo a vote to see which scenarios stakeholder view as most important. Stakeholders could be given a limited number of votes in which to vote for the importance of the scenarios. Furthermore, discussions should be made if some scenarios that have been overlooked should be included even after the voting process if some stakeholders are adamant that a particular scenario should be a priority.

Once the scenarios are chosen and prioritized, they should be used as test cases for the architectural approaches that were discussed in previous steps. Discussions on how the architecture can be changed to fulfill a scenario should occur. For example, a discussion of what components need to be removed, added, or changed, as well as the various communication channels and interfaces between components that would need to be changed should take place. Furthermore, discussions on how these changes are detrimental to the architecture or beneficial to the architecture are

required between the stakeholders. The figure below shows a possible resulting software architecture for the system.

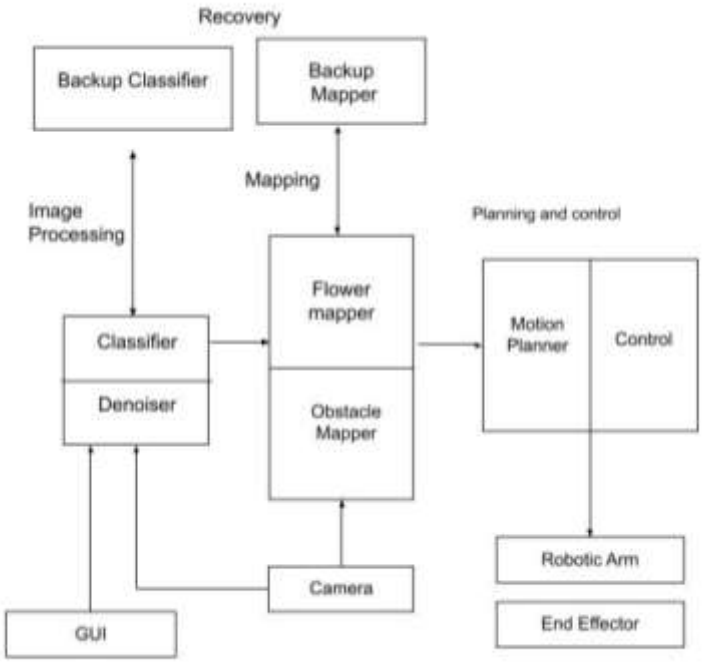

Figure 5.5 Proposed architecture to fulfill scenarios.

In figure 5.5 we can see newly added components to the architecture. In the proposed software architecture, a GUI is added which would provide information for the user to see the state of the system and show why a particular decision such as the motion of the robot is being made. Furthermore, backups of the classifier and mapper are added. This is in case of an attack and allows for faster recovery when the classifier model or map generated is corrupted, whereby the corrupted classifier or map can be quickly replaced. A denoiser is also added to detect and remove biased information that could come from an attack.

6.3 Discussion

Our analysis of the pollination robot shows that the system's architecture was lacking in its trustworthiness qualities. Through ATAM, a higher level of architectural documentation is achieved which allows for a more thorough analysis of the system, in terms of its trustworthiness and tradeoffs with other attributes. Furthermore, the system's trustworthiness requirements were be identified through the process. In the case of the pollination robot, this included explainability, robustness against attacks, and transparency. The analysis opened discussions for tradeoffs between quality attributes of the system, discussions of what are the most key factors for the system, any weaknesses the architecture may have, and discussion of various architectural approaches that could be used. The process of analyzing the trustworthiness of the system increased stakeholder's awareness of the issues of the trustworthiness of the software architecture of the system, provided insight on the major factors that affected it, and revealed what approaches can be used to achieve the trustworthiness requirements.

CHAPTER 7

CONCLUSIONS & FUTURE WORK

7.1 Conclusions

We have established that the trustworthiness of an AI system is a critical attribute needed for the acceptance of AI systems by humans and have proposed the use of trustworthiness scenarios to determine what trustworthiness means in an AI system and how to specify it. We have also described design tactics that can be used to achieve an AI system's trustworthiness goals and a design checklist to guide the analysis for trustworthiness. In our sample analysis of the trustworthiness of a pollination robot, the use of trustworthiness scenarios revealed the system's architecture's lack of trustworthiness and allowed for the specification of trustworthiness through sub-attributes such as explainability, robustness against attacks, and transparency. The analysis also opened discussions of the important factors that could affect the trustworthiness of the AI system, the weaknesses the architecture may have, and a discussion of various architectural approaches that could be used to achieve trustworthiness. This thesis answers several questions. First, how to determine what sub-attributes an AI system should have to be considered trustworthy, second, how to evaluate the trustworthiness of the system, and third, how to achieve trustworthiness in an AI system. In our sample analysis, a trustworthiness utility tree generated also showed how the trustworthiness goals and attributes of an AI system can be represented. With scenarios to provide a more concrete idea of what trustworthiness means to the stakeholders, a trustworthiness design checklist, and tactics to achieve trustworthiness the software architecture of an AI system can be designed with trustworthiness in mind.

7.2 Future Work

To further verify and validate the practicality of the paper's proposals, future work will include more concrete use cases. These use cases would also call to attention any drawbacks and adjustments that may have to be done in real-time. For example, an alternative to the round robin method would be the use of the analytical hierarchy process (AHP) [69] for the prioritization of trustworthiness attributes; other options could be used, such as the use of Triage [70], which is a process for determining the relative priorities. Furthermore, other architectural analysis processes such as the software architecture analysis method (SAAM) could be used to design and analyze the trustworthiness of an AI system. Lastly, there will undoubtedly be progress made in the field of the trustworthiness of AI, which will include the discovery of more trustworthiness design tactics, and these would need to be reflected in the research.

APPENDIX I

Primary studies for trustworthiness sub-attributes

[PS-1] Brundage, Miles, et al. "Toward trustworthy AI development: mechanisms for supporting verifiable claims."

[PS-2] Ahuja, Mohit Kumar, et al. "Opening the Software Engineering Toolbox for the Assessment of Trustworthy AI."

[PS-3] Nassar, Mohamed, et al. "Blockchain for explainable and trustworthy artificial intelligence." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 10.1 (2020).

[PS-4] Toreini, Ehsan, et al. "The relationship between trust in AI and trustworthy machine learning technologies." Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency. 2020.

[PS-5] Rai, Arun. "Explainable AI: From black box to glass box." Journal of the Academy of Marketing Science 48.1 (2020): 137-141.

[PS-6] Arnold, Matthew, et al. "FactSheets: Increasing trust in AI services through supplier's declarations of conformity." IBM Journal of Research and Development 63.4/5 (2019): 6-1.

[PS-7] Holzinger, Andreas, et al. "What do we need to build explainable AI systems for the medical domain?"

[PS-8] Kumar, Abhishek, et al. "Trustworthy AI in the Age of Pervasive Computing and Big Data."

[PS-9] Thelisson, Eva, Kirtan Padh, and L. Elisa Celis. "Regulatory mechanisms and algorithms towards trust in AI/ML." Proceedings of the IJCAI 2017 workshop on explainable artificial intelligence (XAI), Melbourne, Australia. 2017.

[PS-10] Floridi, Luciano. "Establishing the rules for building trustworthy AI." Nature Machine Intelligence 1.6 (2019): 261-262.

[PS-11] S. Jain, M. Luthra, S. Sharma and M. Fatima, "Trustworthiness of Artificial Intelligence," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2020, pp. 907-912, DOI: 10.1109/ICACCS48705.2020.9074237.

[PS-12] Wickramasinghe, Chathurika S., et al. "Trustworthy AI Development Guidelines for Human System Interaction." 2020 13th International Conference on Human System Interaction (HSI). IEEE, 2020.

[PS-13] Wing, Jeannette M. "Trustworthy AI."

[PS-14] Fhaolain, Labhaoise Ni, and Andrew Hines. "Could regulating the creators deliver trustworthy AI?"

[PS-15] Smith, Carol J. "Designing Trustworthy AI: A Human-Machine Teaming Framework to Guide Development."

[PS-16] Mittelstadt, Brent. "AI Ethics–Too principled to fail."

[PS-17] Bride, Hadrien, et al. "Towards Trustworthy AI for Autonomous Systems." International Conference on Formal Engineering Methods. Springer, Cham, 2018.

[PS-18] Antonov, Alexander, and Tanel Kerikmäe. "Trustworthy AI as a Future Driver for Competitiveness and Social Change in the EU." The EU in the 21st Century. Springer, Cham, 2020. 135-154.

[PS-19] Söllner, Matthias, et al. "Towards a Theory of Explanation and Prediction for the Formation of Trust in IT Artifacts." (2011).

[PS-20] Markus, Aniek F., Jan A. Kors, and Peter R. Rijnbeek. "The role of explainability in creating trustworthy artificial intelligence for health care: a comprehensive survey of the terminology, design choices, and evaluation strategies." arXiv preprint arXiv:2007.15911 (2020).

[PS-21] Glomsrud, Jon Arne, et al. "Trustworthy versus explainable ai in autonomous vessels." Proceedings of the International Seminar on Safety and Security of Autonomous Vessels (ISSAV) and European STAMP Workshop and Conference (ESWC) 2019. Sciendo, 2020.

[PS-22] Drobotowicz, Karolina. "Guidelines for Designing Trustworthy AI Services in the Public Sector." (2020).

[PS-23] Kaur, Davinder, Suleyman Uslu, and Arjan Durresi. "Requirements for Trustworthy Artificial Intelligence–A Review." International Conference on Network-Based Information Systems. Springer, Cham, 2020.

[PS-24] Di Maio, Paola. "Neurosymbolic Knowledge Representation for Explainable and Trustworthy AI." (2020).

[PS-25] Cortés, Ulises, Atia Cortés, and Cristian Barrué. "Trustworthy AI. The AI4EU approach." (2019).

[PS-26] Robert Jr, Lionel P., Gaurav Bansal, and Christoph Lütge. "ICIS 2019 SIGHCI Workshop Panel Report: Human–Computer Interaction Challenges and Opportunities for Fair, Trustworthy and Ethical Artificial Intelligence." AIS Transactions on Human-Computer Interaction 12.2 (2020): 96-108.

[PS-27] Vincent-Lancrin, Stéphan, and Reyer van der Vlies. "Trustworthy artificial intelligence (AI) in education: Promises and challenges." (2020).

[PS-28] Hayes, Bradley, and Michael Moniz. "Trustworthy Human-Centered Automation Through Explainable AI and High-Fidelity Simulation." International Conference on Applied Human Factors and Ergonomics. Springer, Cham, 2020.

[PS-29] Marino, Daniel L., et al. "AI Augmentation for Trustworthy AI: Augmented Robot Teleoperation."

[PS-30] Toreini, Ehsan, et al. "Technologies for Trustworthy Machine Learning: A Survey in a Socio-Technical Context." arXiv preprint arXiv:2007.08911 (2020).

[PS-31] He, Hongmei, et al. "The Challenges and Opportunities of Artificial Intelligence for Trustworthy Robots and Autonomous Systems." 2020 3rd International Conference on Intelligent Robotic and Control Engineering (IRCE). IEEE, 2020.

[PS-32] Floridi, Luciano, et al. "AI4People—an ethical framework for a good AI society: opportunities, risks, principles, and recommendations." Minds and Machines 28.4 (2018): 689-707.

[PS-33] Varshney, Kush R. "Trustworthy machine learning and artificial intelligence." XRDS: Crossroads, The ACM Magazine for Students 25.3 (2019): 26-29.

[PS-34] Larasati, Retno and De Liddo, Anna Building a Trustworthy Explainable AI in Healthcare. In: INTERACT 2019/ 17th IFIP: International Conference of Human Computer Interaction. Workshop: Human(s) in the loop -Bringing AI & HCI together, 2-6 Sep 2019, Cyprus, Cardiff, and Ubiquity Press.

[PS-35] He, Hongmei & Cangelosi, Angelo & Mcginnity, T & Mehnen, Jorn & John Gray, & Meng, Qinggang. (2020). The Challenges and Opportunities of Artificial Intelligence in Implementing Trustworthy Robotics and Autonomous Systems.

[PS-36] Li, Chen, et al. "Trustworthy Deep Learning in 6G Enabled Mass Autonomy: from Concept to Quality-of-Trust KPIs."

[PS-37] RP, Jagadeesh Chandra Bose, et al. "Framework for Trustworthy Software Development." 2019 34th IEEE/ACM International Conference on Automated Software Engineering Workshop (ASEW). IEEE, 2019.

[PS-38] Adadi, Amina, and Mohammed Berrada. "Peeking inside the black-box: A survey on Explainable Artificial Intelligence (XAI)." IEEE Access 6 (2018): 52138-52160.

[PS-39] Kaul Shiva. "Speed and accuracy are not enough! Trustworthy machine learning." Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. 2018.

[PS-40] Siau Keng, and Weiyu Wang. "Building trust in artificial intelligence, machine learning, and robotics." Cutter Business Technology Journal 31.2 (2018): 47-53.

[PS-41] Arrieta, Alejandro Barredo, et al. "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI." Information Fusion 58 (2020): 82-115.

[PS-42] Adams, Barbara D., et al. "Trust in automated systems." Ministry of National Defense (2003).

[PS-43] Cahour Béatrice, and Jean-François Forzy. "Does projection into use improve trust and exploration? An example with a cruise control system." Safety science 47.9 (2009): 1260-1270.

[PS-44] Muir, Bonnie M. "Trust in automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems." Ergonomics 37.11 (1994): 1905-1922.

[PS-45] Madsen, Maria, and Shirley Gregor. "Measuring human-computer trust." 11th australasian conference on information systems. Vol. 53. 2000.

APPENDIX II

Primary studies for AI trustworthiness tactics

[S-1] Arora, Saurabh, and Prashant Doshi. "A survey of inverse reinforcement learning: Challenges, methods and progress." *Artificial Intelligence* 297 (2021): 103500.

[S-2] Idé, Tsuyoshi, et al. "Anomaly attribution with likelihood compensation." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. No. 5. 2021.

[S-3] Fung, Yi, et al. "Infosurgeon: Cross-media fine-grained information consistency checking for fake news detection." *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2021.

[S-4] Nguyen, Quoc Phong, et al. "Gee: A gradient-based explainable variational autoencoder for network anomaly detection." *2019 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2019.

[S-5] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "" Why should I trust you?" Explaining the predictions of any classifier." *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016.

[S-6] Aas, Kjersti, Martin Jullum, and Anders Løland. "Explaining individual predictions when features are dependent: More accurate approximations to Shapley values." *Artificial Intelligence* 298 (2021): 103502.

[S-7] Pedapati, Tejaswini, et al. "Learning global transparent models consistent with local contrastive explanations." *Advances in neural information processing systems* 33 (2020): 3592-3602.

[S-8] Dhurandhar, Amit, Karthikeyan Shanmugam, and Ronny Luss. "Enhancing simple models by exploiting what they already know." *International Conference on Machine Learning*. PMLR, 2020.

[S-9] Luss, Ronny, et al. "Leveraging latent features for local explanations." *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2021.

[a10] [S-10] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one 10, 7 (2015), e0130140

[S-11] Ganesan, Balaji, et al. "Link prediction using graph neural networks for master data management." *arXiv preprint arXiv:2003.04732* (2020).

[S-12] Singh, Anjali, and Balaji Ganesan. "Reimagining GNN Explanations with ideas from Tabular Data." *arXiv preprint arXiv:2106.12665* (2021).

[S-13] Hind, Michael, Dennis Wei, and Yunfeng Zhang. "Consumer-Driven Explanations for Machine Learning Decisions: An Empirical Study of Robustness." arXiv preprint arXiv:2001.05573 (2020).

[S-14] Dhurandhar, Amit, Karthikeyan Shanmugam, and Ronny Luss. "Leveraging Simple Model Predictions for Enhancing its Performance." (2019).

[S-15] Oberst, Michael, et al. "Characterization of overlap in observational studies." *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020.

[S-16] Sreedharan, Sarath, et al. "Why can't you do that hal? explaining unsolvability of planning tasks." *International Joint Conference on Artificial Intelligence*. 2019.

[S-17] Alkan, Oznur, et al. "Making Business Partner Recommendation More Effective: Impacts of Combining Recommenders and Explanations through User Feedback." *IUI Workshops*. 2021.

[S-18] Wollenstein-Betech, Salomón, et al. "Explainability of intelligent transportation systems using knowledge compilation: a traffic light controller case." *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2020.

[S-19] C. Jung, M. Kearns, S. Neel, A. Roth, L. Stapleton, and Z. S. Wu, "An algorithmic framework for fairness elicitation," arXiv preprint arXiv:1905.10660, 2019.

[S-20] M. Yurochkin, A. Bower, and Y. Sun, "Training individually fair ML models with sensitive subspace robustness," in International Conference on Learning Representations (ICLR), 2020.

[S-21] M. Yurochkin and Y. Sun, "Sensei: Sensitive set invariance for enforcing individual fairness," in International Conference on Learning Representations (ICLR), 2021.

[S-22] A. Vargo, F. Zhang, M. Yurochkin, and Y. Sun, "Individually fair gradient boosting," in International Conference on Learning Representations (ICLR), 2021.

[S-23] Petersen, Felix, et al. "Post-processing for individual fairness." *Advances in Neural Information Processing Systems* 34 (2021): 25944-25955.

[S-24] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, "Fairness through awareness," in Proceedings of the 3rd innovations in theoretical computer science conference, 2012, pp. 214–226.

[S-25] M. P. Kim, A. Ghorbani, and J. Zou, "Multiaccuracy: Black-box post-processing for fairness in classification," in Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, 2019.

[S-26] P. K. Lohia, K. N. Ramamurthy, M. Bhide, D. Saha, K. R. Varshney, and R. Puri, "Bias mitigation post-processing for individual and group fairness," in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2019, pp. 2847–2851.

[S-27] P. Lohia, “Priority-based post-processing bias mitigation for individual and group fairness,” arXiv preprint arXiv:2102.00417, 2021

[S-28] D. Wei, K. N. Ramamurthy, and F. d. P. Calmon, “Optimized score transformation for fair classification,” in International Conference on Artificial Intelligence and Statistics (AISTATS), 2020.

[S-29] J. Kang, J. He, R. Maciejewski, and H. Tong, “Inform: Individual fairness on graph mining,” in Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020, pp. 379–389.

[S-30] P. Lahoti, K. P. Gummadi, and G. Weikum, “Ifair: Learning individually fair data representations for algorithmic decision making,” in 2019 ieee 35th international conference on data engineering (icde), IEEE, 2019, pp. 1334–1345

[S-31] Biggs, Max, Wei Sun, and Markus Ettl. "Model distillation for revenue optimization: Interpretable personalized pricing." *International Conference on Machine Learning*. PMLR, 2021.

[S-32] Hsu, Chih-Chung, Yi-Xiu Zhuang, and Chia-Yen Lee. "Deep fake image detection based on pairwise learning." *Applied Sciences* 10.1 (2020): 370.

[S-33] McFowland, Edward, Skyler Speakman, and Daniel B. Neill. "Fast generalized subset scan for anomalous pattern detection." *The Journal of Machine Learning Research* 14.1 (2013): 1533-1561.

[S-34] Neill, Daniel B. "Fast subset scan for spatial pattern detection." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 74.2 (2012): 337-360.

[S-35] Zhang, Xu, Svebor Karaman, and Shih-Fu Chang. "Detecting and simulating artifacts in gan fake images." *2019 IEEE international workshop on information forensics and security (WIFS)*. IEEE, 2019.

[S-36] Victor Akinwande, Celia Cintas, Skyler Speakman, and Srihari Sridharan. 2020. Identifying Audio Adversarial Examples via Anomalous Pattern Detection. In Workshop on Adversarial Learning Methods for Machine Learning and Data Mining, KDD'20.

[S-37] Celia Cintas, Skyler Speakman, Victor Akinwande, William Ogallo, Komminist Weldemariam, Srihari Sridharan, and Edward McFowland. 2020. Detecting Adversarial Attacks via Subset Scanning of Autoencoder Activations and Reconstruction Error. In IJCAI 2020.

[S-38] Celia Cintas, Skyler Speakman, Girmaw Abebe Tadesse, Victor Akinwande, Edward McFowland III, and Komminist Weldemariam. Pattern detection in the activation space for identifying synthesized content. *Pattern Recognition Letters* 153 (2022): 207-213.

[S-39] Run Wang, Felix Juefei-Xu, Lei Ma, Xiaofei Xie, Yihao Huang, Jian Wang, and Yang Liu. 2020. FakeSpotter: A Simple yet Robust Baseline for Spotting AI-Synthesized Fake Faces. (2020).

[S-40] Maity S, Xue S, Yurochkin M, Sun Y. "Statistical inference for individual fairness." *arXiv preprint arXiv:2103.16714* (2021).

[S-41] A. Bower, H. Eftekhari, M. Yurochkin, and Y. Sun. Individually fair ranking. ICLR, 2021.

[S-42] Mroueh, Youssef. "Fair Mixup: Fairness via Interpolation." *International Conference on Learning Representations*. 2021.

[S-43] Mukherjee, Debarghya, Mikhail Yurochkin, Moulinath Banerjee, and Yuekai Sun. "Two simple ways to learn individual fairness metrics from data." In *International Conference on Machine Learning*, pp. 7097-7107. PMLR, 2020

[S-44] Sharma, Shubham, Yunfeng Zhang, Jesús M. Ríos Aliaga, Djallel Bouneffouf, Vinod Muthusamy, and Kush R. Varshney. "Data augmentation for discrimination prevention and bias disambiguation." In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 358-364. 2020.

[S-45] Martin Hirzel, Kiran Kate, and Parikshit Ram. "Engineering fair machine learning pipelines." *target* 73, no. 2.2 (2021): 1-028.

[S-46] Pandey, Akshat, and Aylin Caliskan. "Disparate Impact of Artificial Intelligence Bias in Ridehailing Economy's Price Discrimination Algorithms." In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 822-833. 2021.

[S-47] John, Philips George, Deepak Vijaykeerthy, and Diptikalyan Saha. "Verifying individual fairness in machine learning models." In *Conference on Uncertainty in Artificial Intelligence*, pp. 749-758. PMLR, 2020.

[S-48] Calders T, Kamiran F, Pechenizkiy M. Building Classifiers With Independency Constraints. ICDM Workshops - IEEE International Conference on Data Mining. 2009:13-18. August 6-9, 2009; Miami, Florida.

[S-49] Kamishima T, Akaho S, Asoh H, Sakuma J. Fairness-aware classifier with prejudice remover regularizer. In: Flach PA, De Bie T, Cristianini N, eds. Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2012. Lecture Notes in Computer Science, vol 7524. Springer;2012

[S-50] Obermeyer, Ziad, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. "Dissecting racial bias in an algorithm used to manage the health of populations." *Science* 366, no. 6464 (2019): 447-453.

[S-51] Fung, Clement, Chris JM Yoon, and Ivan Beschastnikh. "The limitations of federated learning in sybil settings." In *23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020)*, pp. 301-316. 2020.

[S-52] Fu, Shuhao, Chulin Xie, Bo Li, and Qifeng Chen. "Attack-resistant federated learning with residual-based reweighting." *arXiv preprint arXiv:1912.11464* (2019).

[S-53] Pillutla, Krishna, Sham M. Kakade, and Zaid Harchaoui. "Robust aggregation for federated learning." *IEEE Transactions on Signal Processing* 70 (2022): 1142-1154.

[S-54] Blanchard, Peva, El Mahdi El Mhamdi, Rachid Guerraoui, and Julien Stainer. "Machine learning with adversaries: Byzantine tolerant gradient descent." *Advances in Neural Information Processing Systems* 30 (2017).

[S-55] Guerraoui, Rachid, and Sébastien Rouault. "The hidden vulnerability of distributed learning in byzantium." In *International Conference on Machine Learning*, pp. 3521-3530. PMLR, 2018.

[S-56] Chen, Yudong, Lili Su, and Jiaming Xu. "Distributed statistical machine learning in adversarial settings: Byzantine gradient descent." *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1, no. 2 (2017): 1-25.

[S-57] Chen, Yudong, Lili Su, and Jiaming Xu. "Distributed statistical machine learning in adversarial settings: Byzantine gradient descent." *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1, no. 2 (2017): 1-25.

[S-58] Yin, Dong, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. "Byzantine-robust distributed learning: Towards optimal statistical rates." In *International Conference on Machine Learning*, pp. 5650-5659. PMLR, 2018.

[S-59] Andreina, Sebastien, Giorgia Azzurra Marson, Helen Möllering, and Ghassan Karame. "Baffle: Backdoor detection via feedback-based federated learning." In *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*, pp. 852-863. IEEE, 2021.

[S-60] Wang, Siyue, Xiao Wang, Pin-Yu Chen, Pu Zhao, and Xue Lin. "Characteristic Examples: High-Robustness, Low-Transferability Fingerprinting of Neural Networks." In *IJCAI*, pp. 575-582. 2021.

[S-61] Zang, Xiao, Yi Xie, Jie Chen, and Bo Yuan. "Graph universal adversarial attacks: A few bad actors ruin graph learning models." *arXiv preprint arXiv:2002.04784* (2020).

[S-62] Wang, Ren, Kaidi Xu, Sijia Liu, Pin-Yu Chen, Tsui-Wei Weng, Chuang Gan, and Meng Wang. "On fast adversarial robustness adaptation in model-agnostic meta-learning." *arXiv preprint arXiv:2102.10454* (2021).

[S-63] Wang, Shiqi, Kevin Eykholt, Taesung Lee, Jiyong Jang, and Ian Molloy. "Adaptive Verifiable Training Using Pairwise Class Similarity." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 11, pp. 10201-10209. 2021.

[S-64] Boopathy, Akhilan, Lily Weng, Sijia Liu, Pin-Yu Chen, Gaoyuan Zhang, and Luca Daniel. "Fast training of provably robust neural networks by singleprop." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 8, pp. 6803-6811. 2021.

[S-65] Cheng, Minhao, Pin-Yu Chen, Sijia Liu, Shiyu Chang, Cho-Jui Hsieh, and Payel Das. "Self-progressing robust training." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 8, pp. 7107-7115. 2021.

[S-66] Dapello, Joel, Tiago Marques, Martin Schrimpf, Franziska Geiger, David Cox, and James J. DiCarlo. "Simulating a primary visual cortex at the front of CNNs improves robustness to image perturbations." *Advances in Neural Information Processing Systems* 33 (2020): 13073-13087.

[S-67] Mohapatra, Jeet, Ching-Yun Ko, Tsui-Wei Weng, Pin-Yu Chen, Sijia Liu, and Luca Daniel. "Higher-order certification for randomized smoothing." *Advances in Neural Information Processing Systems* 33 (2020): 4501-4511.

[S-68] Boopathy, Akhilan, Sijia Liu, Gaoyuan Zhang, Cynthia Liu, Pin-Yu Chen, Shiyu Chang, and Luca Daniel. "Proper network interpretability helps adversarial robustness in classification." In *International Conference on Machine Learning*, pp. 1014-1023. PMLR, 2020.

[S-69] Zhao, Pu, Pin-Yu Chen, Payel Das, Karthikeyan Natesan Ramamurthy, and Xue Lin. "Bridging mode connectivity in loss landscapes and adversarial robustness." *arXiv preprint arXiv:2005.00060* (2020).

[S-70] Zhao, Xin, Zeru Zhang, Zijie Zhang, Lingfei Wu, Jiayin Jin, Yang Zhou, Ruoming Jin, Dejing Dou, and Da Yan. "Expressive 1-lipschitz neural networks for robust multiple graph learning against adversarial attacks." In *International Conference on Machine Learning*, pp. 12719-12735. PMLR, 2021.

[S-71] Wang, Siyue, Xiao Wang, Pin-Yu Chen, Pu Zhao, and Xue Lin. "High-Robustness, Low-Transferability Fingerprinting of Neural Networks." *arXiv preprint arXiv:2105.07078* (2021).

[S-72] Yang, Chao-Han Huck, I. Hung, Te Danny, Yi Ouyang, and Pin-Yu Chen. "Causal inference q-network: Toward resilient reinforcement learning." *arXiv preprint arXiv:2102.09677* (2021).

[S-73] Chen, Tianlong, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. "Adversarial robustness: From self-supervised pre-training to fine-tuning." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 699-708. 2020.

[S-74] Mohapatra, Jeet, Tsui-Wei Weng, Pin-Yu Chen, Sijia Liu, and Luca Daniel. "Towards verifying robustness of neural networks against a family of semantic perturbations." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 244-252. 2020.

[S-75] Xu, Kaidi, Sijia Liu, Pin-Yu Chen, Mengshu Sun, Caiwen Ding, Bhavya Kailkhura, and Xue Lin. "Towards an efficient and general framework of robust training for graph neural networks." In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 8479-8483. IEEE, 2020.

[S-76] Wang, Xiao, Siyue Wang, Pin-Yu Chen, Xue Lin, and Peter Chin. "Advms: A multi-source multi-cost defense against adversarial attacks." In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 2902-2906. IEEE, 2020.

[S-77] Liu, Sijia, Pin-Yu Chen, Bhavya Kailkhura, Gaoyuan Zhang, Alfred O. Hero III, and Pramod K. Varshney. "A primer on zeroth-order optimization in signal processing and machine learning: Principals, recent advances, and applications." *IEEE Signal Processing Magazine* 37, no. 5 (2020): 43-54.

[S-78] Riedl, Mark O., and Brent Harrison. "Using stories to teach human values to artificial agents." In *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*. 2016.

[S-79] Loreggia, Andrea, Nicholas Mattei, Francesca Rossi, and K. Brent Venable. "Value alignment via tractable preference distance." In *Artificial Intelligence Safety and Security*, pp. 249-261. Chapman and Hall/CRC, 2018.

[S-80] Brown, Daniel S., Jordan Schneider, Anca Dragan, and Scott Niekum. "Value alignment verification." In *International Conference on Machine Learning*, pp. 1105-1115. PMLR, 2021.

[S-81] Yuan, Luyao, Xiaofeng Gao, Zilong Zheng, Mark Edmonds, Ying Nian Wu, Federico Rossano, Hongjing Lu, Yixin Zhu, and Song-Chun Zhu. "In situ bidirectional human-robot value alignment." *Science Robotics* 7, no. 68 (2022): eabm4183.

[S-82] Fisac, Jaime F., Monica A. Gates, Jessica B. Hamrick, Chang Liu, Dylan Hadfield-Menell, Malayandi Palaniappan, Dhruv Malik, S. Shankar Sastry, Thomas L. Griffiths, and Anca D. Dragan. "Pragmatic-pedagogic value alignment." In *Robotics Research*, pp. 49-57. Springer, Cham, 2020.

[S-83] Arnold, Thomas, and Daniel Kasenberg. "Value Alignment or Misalignment “What Will Keep Systems Accountable?" In *AAAI Workshop on AI, Ethics, and Society*. 2017.

[S-84] Feldman, Michael, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. "Certifying and removing disparate impact." In *proceedings of the 21st ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 259-268. 2015.

[S-85] Noothigattu, Ritesh, et al. "Interpretable multi-objective reinforcement learning through policy orchestration." *arXiv preprint arXiv:1809.08343* (2018)

[S-86] Sattigeri, Prasanna, et al. "Fairness GAN: Generating datasets with fairness properties using a generative adversarial network." IBM Journal of Research and Development 63.4/5 (2019): 3-1.

[S-87] Backurs, Arturs, et al. "Scalable fair clustering." International Conference on Machine Learning. PMLR, 2019.

[S-88] Sen, Procheta, and Debasis Ganguly. "Towards socially responsible ai: Cognitive bias-aware multi-objective learning." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 03. 2020.

APPENDIX III

Trustworthiness scenarios for pollination robot

Table 1: Scenario 1

SCENARIO 1
Scenario: user tries to understand AI model *Source*: user *Stimulus*: user request explanation of results *Artifact*: system graphical user interface *Environment*: normal operation *Response*: system provides explanation *Response* Measure: user understands explanation within 30 min

Table 2: Scenario 2

SCENARIO 2
Scenario: user tries to pinpoint the factors for a decision *Source*: user *Stimulus*: user request because a certain result was given *Artifact*: system graphical user interface *Environment*: normal operation *Response*: system provides significant factors for decision *Response* Measure: user knows significant factors of a result given within 5 minutes

Table 3: Scenario 3

SCENARIO 3
Scenario: system AI model has been compromised by attack *Source*: external system *Stimulus*: machine detects deceptive inputs for data model *Artifact*: model *Environment*: normal operation *Response*: data model restored *Response* measure: data model is restored to previous percentage accuracy

Table 4: Scenario 4

SCENARIO 4
Scenario: external system attempts an adversarial attack on model *Source*: external system *Stimulus*: AI has been given deceptive inputs *Artifact*: data model *Environment*: normal operation *Response*: attack on data model detected *Response* Measure: attack on data model detected within 1 minute

Table 5: Scenario 5

SCENARIO 5
Scenario: system dataset has undergone an adversarial attack *Source*: external system *Stimulus*: machine detects deceptive inputs for dataset *Artifact*: data model *Environment*: normal operation *Response*: dataset restored *Response* measure: dataset is restored within 1 minute

Table 6: Scenario 6

SCENARIO 6
Scenario: user requests information on state of machine *Source*: user *Stimulus*: user requests to see status of system *Artifact*: system graphical user interface *Environment*: normal operation *Response*: system shows user state of system *Response* Measure: user should be able to tell what the state of the system is in less than 10 minutes

REFERENCES

[1] W. Hasselbring and R. Reussner, "Toward trustworthy software systems," in Computer, vol. 39, no. 4, pp. 91-92, April 2006.

[2] Yeasmin, Samira. "Benefits of Artificial Intelligence in Medicine." *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)*. IEEE, 2019.

[3] Brundage, Miles, Shahar Avin, Jasmine Wang, Haydn Belfield, Gretchen Krueger, Gillian Hadfield, Heidy Khlaaf et al. "Toward trustworthy AI development: mechanisms for supporting verifiable claims." *arXiv preprint arXiv:2004.07213* (2020).

[4] Antonov, Alexander, and Tanel Kerikmäe. "Trustworthy AI as a Future Driver for Competitiveness and Social Change in the EU." *The EU in the 21st Century*. Springer, Cham, 2020. 135-154.

[5] Guan, Jian. "Artificial intelligence in healthcare and medicine: promises, ethical challenges and governance." *Chinese Medical Sciences Journal* 34.2 (2019): 76-83.

[6] Wickramasinghe, Chathurika S., Daniel L. Marino, Javier Grandio, and Milos Manic. "Trustworthy AI development guidelines for human system interaction." In 2020 13th International Conference on Human System Interaction (HSI), pp. 130-136. IEEE, 2020.

[7] Self-driving uber kills arizona woman in first fatal crash involving pedestrian, February 2018, [online]
Available:https://www.theguardian.com/technology/2018/mar/19/uber-self-driving-car-kills-woman-arizona-tempe.

[8] F. Lambert, "Understanding the fatal tesla accident on autopilot and the nhtsa probe", Electrek, 2016

[9] Kumar, Abhishek, Tristan Braud, Sasu Tarkoma, and Pan Hui. "Trustworthy AI in the age of pervasive computing and big data." In 2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), pp. 1-6. IEEE, 2020.

[10] Larasati, Retno, and Anna De Liddo. "Building a Trustworthy Explainable AI in Healthcare."

[11] Kazman, Rick, Mark Klein, and Paul Clements. ATAM: Method for architecture evaluation. Carnegie-Mellon Univ Pittsburgh PA Software Engineering Inst, 2000.

[12] Schoorman, F. David, Roger C. Mayer, and James H. Davis. "An integrative model of organizational trust: Past, present, and future." (2007): 344-354.

[13] Lee, John D., and Katrina A. See. "Trust in automation: Designing for appropriate reliance." *Human factors* 46.1 (2004): 50-80.

[14] Hinde, Robert Aubrey, Robert A. Hinde, and Jo Groebel, eds. Cooperation and prosocial behaviour. Cambridge University Press, 1991.

[15] Jian, Jiun-Yin, Ann M. Bisantz, and Colin G. Drury. "Foundations for an empirically determined scale of trust in automated systems." *International journal of cognitive ergonomics* 4.1 (2000): 53-71.

[16] Toreini, Ehsan, Mhairi Aitken, Kovila Coopamootoo, Karen Elliott, Carlos Gonzalez Zelaya, and Aad Van Moorsel. "The relationship between trust in AI and trustworthy machine learning technologies." In Proceedings of the 2020 conference on fairness, accountability, and transparency, pp. 272-283. 2020.

[17] High-Level Expert Group on Artificial Intelligence, 'Ethics Guidelines for Trustworthy AI', Technical report, European Commission, (2019).

[18] Madsen, Maria, and Shirley Gregor. "Measuring human-computer trust." *11th australasian conference on information systems*. Vol. 53. 2000.
[19] Balfe, Nora, Sarah Sharples, and John R. Wilson. "Understanding is key: An analysis of factors pertaining to trust in a real-world automation system." *Human factors* 60.4 (2018): 477-495.
[20] Avizienis, Algirdas, Jean-Claude Laprie, and Brian Randell. Fundamental concepts of dependability. University of Newcastle upon Tyne, Computing Science, 2001.
[21] Adams, Barbara D., Lora E. Bruyn, Sébastien Houde, Paul Angelopoulos, Kim Iwasa-Madge, and Carol McCann. "Trust in automated systems." Ministry of National Defence (2003).
[22] Cahour, Béatrice, and Jean-François Forzy. "Does projection into use improve trust and exploration? An example with a cruise control system." *Safety science* 47.9 (2009): 1260-1270.
[23] Muir, Bonnie M. "Trust in automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems." Ergonomics 37.11 (1994): 1905-1922.
[24] Yu, Kun-Hsing, and Isaac S. Kohane. "Framing the challenges of artificial intelligence in medicine." *BMJ quality & safety* 28.3 (2019): 238-241.
[25] Ali, Omar, Ilias Flaounas, Tijl De Bie, Nick Mosdell, Justin Lewis, and Nello Cristianini. "Automating news content analysis: An application to gender bias and readability." In Proceedings of the first workshop on applications of pattern analysis, pp. 36-43. PMLR, 2010.
[26] Bolukbasi, Tolga, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam T. Kalai. "Man is to computer programmer as woman is to homemaker? debiasing word embeddings." Advances in neural information processing systems 29 (2016).
[27] Nelson, Gregory S. "Bias in artificial intelligence." *North Carolina medical journal* 80.4 (2019): 220-222.
[28] Habli, Ibrahim, Tom Lawton, and Zoe Porter. "Artificial intelligence in health care: accountability and safety." *Bulletin of the World Health Organization* 98.4 (2020): 251.
[29] Ananny, Mike, and Kate Crawford. "Seeing without knowing: Limitations of the transparency ideal and its application to algorithmic accountability." *new media & society* 20.3 (2018): 973-989.
[30] Holzinger, A., Biemann, C., Pattichis, C. S., & Kell, D. B. (2017). What do we need to build explainable AI systems for the medical domain? *arXiv preprint arXiv:1712.09923*.
[31] Fuji, Masaru, Katsuhito Nakazawa, and Hiroaki Yoshida. "“Trustworthy and Explainable AI” Achieved Through Knowledge Graphs and Social Implementation." *FUJITSU SCIENTIFIC & TECHNICAL JOURNAL* 56.1 (2020): 39-45.
[32] Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., & Mueller, K. R. (2009). How to explain individual classification decisions. *arXiv preprint arXiv:0912.1128*.
[33] Gabriel, Iason. "Artificial intelligence, values, and alignment." *Minds and machines* 30.3 (2020): 411-437.
[34] Rosa, Nelson S., George RR Justo, and Paulo RF Cunha. "A framework for building non-functional software architectures." *Proceedings of the 2001 ACM symposium on Applied computing*. 2001.
[35] Bass, Len, Paul Clements, and Rick Kazman. *Software architecture in practice*. Addison-Wesley Professional, 2003.
[36] Harrison, Neil B., and Paris Avgeriou. "Leveraging architecture patterns to satisfy quality attributes." European conference on software architecture. Springer, Berlin, Heidelberg, 2007.
[37] Harrison, Neil B., and Paris Avgeriou. "How do architecture patterns and tactics interact? A model and annotation." Journal of Systems and Software 83.10 (2010): 1735-1758.

[38] Kazman, Rick, Gregory Abowd, Len Bass, and Paul Clements. "Scenario-based analysis of software architecture." IEEE software 13, no. 6 (1996): 47-55.
[39] Tao, Hongwei, and Yixiang Chen. "A metric model for trustworthiness of softwares." 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology. Vol. 3. IEEE, 2009.
[40] Gol Mohammadi, Nazila, Sachar Paulus, Mohamed Bishr, Andreas Metzger, Holger Könnecke, Sandro Hartenstein, Thorsten Weyer, and Klaus Pohl. "Trustworthiness attributes and metrics for engineering trusted internet-based software systems." In International Conference on Cloud Computing and Services Science, pp. 19-35. Springer, Cham, 2013.
[41] Gupta, Deepak, Anil Ahlawat, and Kalpna Sagar. "A critical analysis of a hierarchy based Usability Model." *2014 international conference on contemporary computing and informatics (IC3I)*. IEEE, 2014.
[42] Bass, Len, and Gabriel Moreno. Applicability of general scenarios to the architecture tradeoff analysis method. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, 2001.
[43] Bass, Len, Mark Klein, and Felix Bachmann. *Quality attribute design primitives*. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, 2000.
[44] Bauer, Paul C. "Conceptualizing trust and trustworthiness." (2019).
[45] Pandey, Akshat, and Aylin Caliskan. "Disparate Impact of Artificial Intelligence Bias in Ridehailing Economy's Price Discrimination Algorithms." In Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society, pp. 822-833. 2021.
[46] Feldman, Michael, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. "Certifying and removing disparate impact." In proceedings of the 21st ACM SIGKDD international conference on knowledge discovery and data mining, pp. 259-268. 2015.
[47] Maity S, Xue S, Yurochkin M, Sun Y. "Statistical inference for individual fairness." *arXiv preprint arXiv:2103.16714* (2021).
[48] John, Philips George, Deepak Vijaykeerthy, and Diptikalyan Saha. "Verifying individual fairness in machine learning models." In Conference on Uncertainty in Artificial Intelligence, pp. 749-758. PMLR, 2020.
[49] A. Bower, H. Eftekhari, M. Yurochkin, and Y. Sun. Individually fair ranking. ICLR, 2021.
[50] Sharma, Shubham, Yunfeng Zhang, Jesús M. Ríos Aliaga, Djallel Bouneffouf, Vinod Muthusamy, and Kush R. Varshney. "Data augmentation for discrimination prevention and bias disambiguation." In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, pp. 358-364. 2020.
[51] M. Yurochkin, A. Bower, and Y. Sun, “Training individually fair ML models with sensitive subspace robustness,” in International Conference on Learning Representations (ICLR), 2020.
[52] M. P. Kim, A. Ghorbani, and J. Zou, “Multi Accuracy: Black-box post-processing for fairness in classification,” in Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, 2019.

[53] Vasconcelos, Marisa, Carlos Cardonha, and Bernardo Gonçalves. "Modeling epistemological principles for bias mitigation in AI systems: an illustration in hiring decisions." Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. 2018.
[54] Tomsett, Richard, et al. "Sanity checks for saliency metrics." *Proceedings of the AAAI conference on artificial intelligence*. Vol. 34. No. 04. 2020.
[55] Hong, Shenda, Cao Xiao, Tengfei Ma, Hongyan Li, and Jimeng Sun. "MINA: multilevel knowledge-guided attention for modeling electrocardiography signals." arXiv preprint arXiv:1905.11333 (2019).
[56] Weber, Mark, Giacomo Domeniconi, Jie Chen, Daniel Karl I. Weidele, Claudio Bellei, Tom Robinson, and Charles E. Leiserson. "Anti-money laundering in bitcoin: Experimenting with graph convolutional networks for financial forensics." arXiv preprint arXiv:1908.02591 (2019).
[57] Dhurandhar, Amit, Pin-Yu Chen, Ronny Luss, Chun-Chen Tu, Paishun Ting, Karthikeyan Shanmugam, and Payel Das. "Explanations based on the missing: Towards contrastive explanations with pertinent negatives." Advances in neural information processing systems 31 (2018).
[58] Codella, Noel CF, Michael Hind, Karthikeyan Natesan Ramamurthy, Murray Campbell, Amit Dhurandhar, Kush R. Varshney, Dennis Wei, and Aleksandra Mojsilovic. "Teaching meaningful explanations." arXiv preprint arXiv:1805.11648 (2018).
[59] Pedapati, Tejaswini, Avinash Balakrishnan, Karthikeyan Shanmugam, and Amit Dhurandhar. "Learning global transparent models consistent with local contrastive explanations." Advances in neural information processing systems 33 (2020): 3592-3602.
[60] Noothigattu, Ritesh, Djallel Bouneffouf, Nicholas Mattei, Rachita Chandra, Piyush Madan, Kush R. Varshney, Murray Campbell, Moninder Singh, and Francesca Rossi. "Teaching AI agents ethical values using reinforcement learning and policy orchestration." IBM Journal of Research and Development 63, no. 4/5 (2019): 2-1.
[61] Loreggia, Andrea, Nicholas Mattei, Francesca Rossi, and K. Brent Venable. "A notion of distance between cp-nets." In Proc. of AAMAS, pp. 955-963. 2018.
[62] Brown, Daniel S., Jordan Schneider, Anca Dragan, and Scott Niekum. "Value alignment verification." In International Conference on Machine Learning, pp. 1105-1115. PMLR, 2021.
[63] Chang, Chih-Ling, Jui-Lung Hung, Chin-Wei Tien, Chia-Wei Tien, and Sy-Yen Kuo. "Evaluating robustness of ai models against adversarial attacks." In Proceedings of the 1st ACM Workshop on Security and Privacy on Artificial Intelligence, pp. 47-54. 2020.
[64] Shafahi, Ali, Mahyar Najibi, Mohammad Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S. Davis, Gavin Taylor, and Tom Goldstein. "Adversarial training for free!" Advances in Neural Information Processing Systems 32 (2019).
[65] McFowland, Edward, Skyler Speakman, and Daniel B. Neill. "Fast generalized subset scan for anomalous pattern detection." The Journal of Machine Learning Research 14.1 (2013): 1533-1561.

[66] Chow, Ka-Ho, Wenqi Wei, Yanzhao Wu, and Ling Liu. "Denoising and verification cross-layer ensemble against black-box adversarial attacks." In 2019 IEEE International Conference on Big Data (Big Data), pp. 1282-1291. IEEE, 2019.
[67] Zhao, Pu, Pin-Yu Chen, Payel Das, Karthikeyan Natesan Ramamurthy, and Xue Lin. "Bridging mode connectivity in loss landscapes and adversarial robustness." arXiv preprint arXiv:2005.00060 (2020).
[68] Strader, Jared, Jennifer Nguyen, Christopher Tatsch, Yixin Du, Kyle Lassak, Benjamin Buzzo, Ryan Watson et al. "Flower interaction subsystem for a precision pollination robot." In 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5534-5541. IEEE, 2019.
[69] Saaty, Roseanna W. "The analytic hierarchy process—what it is and how it is used." Mathematical modeling 9.3-5 (1987): 161-176.
[70] Davis, Alan M. "The art of requirements triage." *Computer* 36.3 (2003): 42-49.