DISSERTATIONES
INFORMATICAE
UNIVERSITATIS
TARTUENSIS
71

ABDUL-RASHEED O. OTTUN

Practical Trustworthy Artificial Intelligence with Human Oversight

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

71

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS
71

ABDUL-RASHEED OLATUNJI OTTUN

Practical Trustworthy Artificial Intelligence with Human Oversight

UNIVERSITY OF TARTU
Press
1632

Institute of Computer Science, Faculty of Science and Technology, University of Tartu, Estonia.

Dissertation has been accepted for the commencement of the degree of Doctor of Philosophy (PhD) in Computer Science on September 30, 2025 by the Council of the Institute of Computer Science, University of Tartu.

Supervisor

Assoc. Prof. PhD — Huber Raul Flores Macario
Institute of Computer Science
University of Tartu, Estonia

Opponents

Prof. PhD — Christian Becker
Institute of Parallel and Distributed Systems
University of Stuttgart, Germany

PhD — Alexandre Da Silva Veith
Software and Data Systems Research Lab
Nokia Bell Labs, Belgium

The public defense will take place on November 7, 2025 at 12:15 in Narva Rd. 18-1021.

The publication of this dissertation was financed by the Institute of Computer Science, University of Tartu.

ISSN 2613-5906 (print) ISSN 2806-2345 (pdf)
ISBN 978-9908-57-031-0 (print) ISBN 978-9908-57-032-7 (pdf)

University of Tartu Press
http://www.tyk.ee/

To my family and friends

ABSTRACT

Modern applications increasingly rely on machine and deep learning, or artificial intelligence (AI) to boost performance, enhance perception, and deliver a more reliable user experience. Despite their advanced reasoning capabilities, AI models are often opaque, creating safety concerns and reducing trust. Regulatory frameworks emphasize trustworthy AI, which builds on trustworthy computing with added principles like transparency and human oversight. However, integrating human-in-the-loop mechanisms in distributed AI systems remains a challenge. The core research question of this thesis is: **How can human oversight approaches be integrated into AI-enabled applications to monitor and contribute to their trustworthiness?**

To address these challenges, we propose three contributions, each targeting a specific technical problem and corresponding stage of the machine learning pipeline. First, since data quality is critical for AI decision-making, we introduce Social-Aware Federated Learning (SAFL) for distributed machine learning. SAFL adopts a collaborative approach that leverages social dynamics and task delegation to guide data selection for model training while incentivizing human participation. Through a rigorous user study and a proof-of-concept implementation, we demonstrate that SAFL enhances both data quality and model performance by integrating meaningful human input.

Second, as applications increasingly incorporate AI components, we present a solution to monitor their trustworthy properties. By examining the evolution of system architectures, we systematically explore how trustworthiness mechanisms can be embedded into modern systems. We introduce SPATIAL, a proof-of-concept architecture that integrates trustworthiness metrics into AI-enabled applications. SPATIAL features a user-facing dashboard that communicates these metrics clearly, enabling human experts to monitor AI inference logic effectively. Empirical evaluations demonstrate its effectiveness while highlighting the challenges of integrating mechanisms to measure and maintain trustworthiness in real-world applications.

Third, human oversight is also essential in monitoring deployed applications, particularly those operating autonomously at scale. To this end, we propose AntiVenom, an efficient, domain-agnostic technique for detecting anomalies in distributed AI deployments. AntiVenom leverages device-level performance metrics to identify irregularities and flag them for human review. Comparative analysis against initial examination with explainable AI (XAI) methods shows AntiVenom's potential for fast and proactive monitoring when compared with traditional and more complex methods.

Together, these contributions highlight both the potential and the complexity of embedding human oversight into increasingly autonomous systems, advancing the development of trustworthy AI.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ORIGINAL PUBLICATIONS

Publications included in the thesis

1. **Abdul-Rasheed Ottun**, Marasinghe Rasinthe, Elemosho Toluwani ... and Huber Flores., "The SPATIAL Architecture: Design and Development Experiences from Gauging and Monitoring the AI Inference Capabilities of Modern Applications." In: *Proceedings of the 44th IEEE International Conference on Distributed Computing Systems.* 2024, pp. 947-959. DOI: 10.1109/ICDCS60910.2024.00092.
2. **Abdul-Rasheed Ottun**, Marasinghe Rasinthe, Elemosho Toluwani, Liyanage Mohan, Ashfaq Hussain Ahmed ... and Huber Flores., "SPATIAL: Practical AI Trustworthiness with Human Oversight." *In Proceedings of the 44th IEEE International Conference on Distributed Computing Systems.* 2024, pp. 947-959. DOI: 10.1109/ICDCS60910.2024.00138.
3. **Abdul-Rasheed Ottun**, Mehrdad Asadi, Michell Boerger, Nikolay Tcholtchev, João Gonçalves, Duşan Borovčanin, Bartlomiej Siniarsk and Huber Flores."One to Rule Them All: A Study on Requirement Management Tools for the Development of Modern AI-based Software." In: *Proceedings of the IEEE International Conference on Big Data.* 2023, pp. 947-959. DOI: 10.1109/BigData59044.2023.10386926.
4. **Abdul-Rasheed Ottun**, Pramod C. Mane, Zhigang Yin, Souvik Paul, Mohan Liyanage, Jason Pridmore, Aaron Yi Ding, Rajesh Sharma, Petteri Nurmi and Huber Flores. "Social-Aware Federated Learning: Challenges and Opportunities in Collaborative Data Training." In: *IEEE Internet Computing*, vol. 27, no. 2, pp. 36-44, 1 March-April 2023. DOI: 10.1109/MIC.2022.3219263.
5. **Abdul-Rasheed Ottun**, Adeyinka Akintola, Mohan Liyanage, Michell Boerger, Pan Hui, Sasu Tarkoma, Nikolay Tcholtchev, Petteri Nurmi and Huber Flores. "AI Robustness Against Attacks in City-Scale Autonomous Drone Deployments." In: *IEEE Computer,* vol. 57, no. 12, pp. 47-57, December 2024. DOI: 10.1109/MC.2024.3461841.

Publications not included in the thesis

6. **Abdul-Rasheed Ottun** and Huber Flores. "Trustworthy AI in Practice: A Comprehensive Review of Human Oversight and Human-in-the-Loop Approaches" (under review)
   - The full version of this survey paper is included in Chapter 2 to further reflect the scope and depth of the author's contribution beyond the main chapters.

Other published work of the author

7. Dar, Farooq Ayoub, Mayowa Olapade, **Abdul-rasheed Ottun**, Zhigang Yin, Mohan Liyanage, Ulrich Norbisrath, Marko Radeta et al. "TOAD: Profiling and Evaluating 3D Printed IoT Rapid Prototype Designs." In: *ACM Transactions on Internet of Things, 2025.* DOI: 10.1145/3724128
8. Dar, Farooq, Mohan Liyanage, Mayowa Olapade, Zhigang Yin, **Abdul-Rasheed Ottun**, Adeyinka Akintola, Francisco Airton Silva, and Huber Flores. "Demo Abstract: PRINCE: Device Energy Estimation with a Single Photo." *In Proceedings of the 9th IEEE/ACM International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pp. 231-232, IEEE, 2024. DOI: 10.1109/IoTDI61053.2024.00031
9. Dar, Farooq, Mayowa Olapade, **Abdul-Rasheed Ottun**, Zhigang Yin, Mohan Liyanage, Agustin Zuniga, Monica Passanantti, Sasu Tarkoma, Petteri Nurmi, and Huber Flores. "LIZARD: Pervasive sensing for autonomous plastic litter monitoring." In: *Proceedings of the 9th IEEE/ACM International Conference on Internet-of-Things Design and Implementation,* pp. 37-48, IEEE, 2024, DOI: 10.1109/IoTDI61053.2024.00008
10. Olapade, Mayowa, **Abdul-Rasheed Ottun**, Zhigang Yin, Mohan Liyanage, Adeyinka Akintola, and Huber Flores. "Seamless Integration of Nano-Drones and Sensors for Agriculture Monitoring at Scale." In: *Proceedings of the 13th ACM International Conference on the Internet of Things,* pp. 189-192, 2023. DOI: 10.1145/3627050.3630733
11. Yin, Zhigang, Mohan Liyanage, **Abdul-Rasheed Ottun**, Farooq Dar, Mayowa Olapade, and Huber Flores. "Demo Abstract: A Smart Ring Monitoring Your Health using Hand-grip Strength." In: *Proceedings of the 21st ACM Conference on Embedded Networked Sensor Systems,* pp. 486-487. 2023. DOI: 10.1145/3625687.3628395
12. Olapade, Mayowa, **Abdul-Rasheed Ottun**, Zhigang Yin, Mohan Liyanage, Aleksandr Makarov, and Huber Flores. "Low-cost produce quality monitoring at scale: A practical re-purposing framework for pervasive agriculture." In: *Proceedings of the 13th ACM International Conference on the Internet of Things,* pp. 129-137, 2023. DOI: 10.1145/3627050.3627055
13. Yin, Zhigang, Mohan Liyanage, **Abdul-Rasheed Ottun**, Souvik Paul, Agustin Zuniga, Petteri Nurmi, and Huber Flores. "Hippo: Pervasive hand-grip estimation from everyday interactions." In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies,* 6, no. 4 (2023): 1-30, DOI: 10.1145/3570344
14. Olapade, Mayowa, Tarlan Hasanli, **Abdul-Rasheed Ottun**, Akintola Adeyinka, Mohan Liyanage and Huber Flores. Pervasive chatbots: Investigating chatbot interventions for multi-Device applications. In: *Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization,* pp.

290-300, 2024, DOI: 10.1145/3627043.3659570

15. Yin, Zhigang, Marko Radeta, Mohan Liyanage, Mayowa Olapade, **Abdul-Rasheed Ottun**, Agustin Zuniga, Pan Hui, Petteri Nurmi, and Huber Flores. "SNAKE: Harnessing Human Touch for Produce Quality Estimation to Foster Sustainable Retail Practices.", ACM Transactions on Sensor Networks, 2025, DOI: 10.1145/3733720.
16. Yin, Zhigang, Marko Radeta, Kevin Post, Mohan Liyanage, Mayowa Olapade, Reo Kuchida, **Abdul-Rasheed Ottun**, Adeyinka Akintola, Petteri Nurmi, and Huber Flores. "- BEE: Opportunistic Heat-Based Bio-Sensing for Produce Quality Monitoring.", ACM Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT), 2025.

Author's contribution to the publications

The author of this thesis is the lead author of Publications 1–5 listed in the original publications section. As lead author, he made significant contributions to the development of the research idea, its implementation, and experimental validation. In Publications 1 and 2, he was responsible for experimental design, conceptual development, and presenting the research at a conference, including a demo. In Publication 3, he led the experimental design, developed the analysis criteria, and deployed the tools, as well as presentation of the results at a conference. In Publications 4 and 5, he contributed to the research idea, experimental work, and played a major role in writing the articles.

Furthermore, the author's publications have contributed to the peer-reviewed deliverables of the EU SPATIAL Horizon 2020 project, for which the author also assisted in preparing the drafts. These deliverables are referenced as follows.

1. D.3.1 - Detection mechanisms to identify data biases and explanatory studies about different data quality trade-offs for AI-based systems.
2. D.3.2 - An explanatory platform that accounts AI systems based on its quantified quality.
3. D.3.3 - Automated diagnosis and mechanisms for tuning AI-based systems and trusted execution environments for accountable and resilient AI.
4. D.3.4 - Performance evaluation in controlled environments and guidelines to build the pilot studies in real testbeds.
5. D5.1 - Description of use-case, design, testbed, experimentation of pilots.

1. INTRODUCTION

Distributed systems have been fundamental to the development of nearly all digital applications we access ubiquitously. From online Web services to large-scale computations, they provide the interconnected infrastructure that powers and sustains the internet. The rapid advancement of artificial intelligence technologies also relies heavily on distributed computing, enabling the training of large-scale models such as ChatGPT, Gemini, and DeepSeek.

Thanks to the robust and accurate performance of machine and deep learning techniques, AI models, ranging from lightweight to large-scale, are now embedded in nearly every system, digital and data-driven application. From the curated news in our social feeds to music playlists that match our mood, and navigation apps that guide us home, AI shapes our everyday experiences. AI models have become a fundamental component of modern system architectures.

Modern applications depend on AI to enhance their features and drive digital activities. The reach of AI is undeniable and everywhere, and the potential it brings to our society is immense. However, this is not without caution or control. As we increasingly adopt these technologies, their failures are manifesting with severe consequences without accountability and responsible usage. These issues underscore the need for human monitoring to oversee their development process and operations so as to address anomalies and assure users of the safety of the technology. Understanding the balance between the potential of AI and our ability to govern its development with human oversight mechanisms is critical for the reliable deployment and maintenance of AI trust in our society. This research work explores the intersection between distributed systems and artificial intelligence (Figure 1).

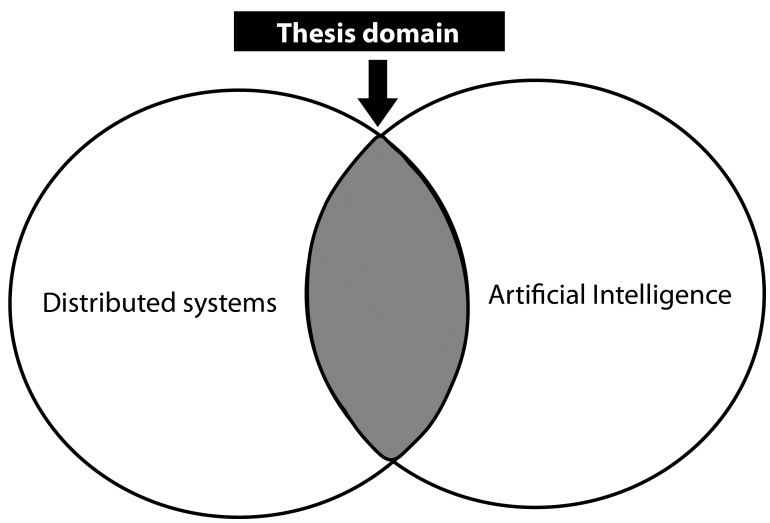


Figure 1: The field of research explored

1.1. Distributed systems

The evolution of human society has been closely intertwined with technological progress, much of it enabled by distributed systems [121, 269]. At its most abstract definition, *a distributed system is composed of interconnected components that interact with each other to achieve a common goal.* The 1940s marked a major milestone in computing history with the emergence of the first set of programmable electronic computation machines, setting the stage for other innovations in computation, communication technologies, and information processing. Over time, as computational capabilities increase, computing technologies become more pervasive and sophisticated, accelerating the digitization of society. These achievements progressed our society through different periods, from mainframe computing to personal computing, then mobile computing, and eventually cloud computing. Today, the backbone infrastructure that supports the Internet is highly distributed and scalable, enabling the creation of a vast array of innovative digital applications.

Distributed systems technologies have played a key role in the development of artificial intelligence [318]. Current capabilities of distributed systems allow us to fully explore the potential of machine and deep learning at scale. As hardware limitations are increasingly surpassed, distributed systems are becoming more powerful, opening the door to new approaches for building artificial intelligence, such as the emerging use of genetic algorithms [177, 403, 355]. The culmination of these advancements has brought about a data-driven era, marked by a significant shift in computing capabilities. As a result, computers are now better equipped to harness data and adapt to dynamic contexts, signaling a new phase in the evolution of intelligent systems within our society [292].

Paradoxically, the very intelligence that has emerged from distributed systems is now being harnessed to enhance these systems themselves. AI is playing a crucial role in optimizing the self-organization of distributed networks, enabling systems to autonomously detect and address failures [16], design more efficient topologies and deployments [338], and ultimately refine the user experience [227]. Through continuous learning and adaptation, AI helps these systems become more resilient, efficient, and responsive, improving both their performance and the way users interact with them.

1.2. Artificial intelligence meets distributed systems

AI models are integral parts of larger systems, where they are implemented as components based on machine and deep learning techniques (Figure 2). These techniques employ advanced pipelines to convert raw data into actionable insights. These pipelines basically represent a series of processes and steps, discussed in Section 1.2.2, for constructing and managing AI models. They leverage advanced algorithms to detect patterns, learn from data, and continuously refine their understanding. As a result, the models can generate accurate predictions, adapting and

improving as they process more information. These pipelines can be seamlessly integrated or augmented within larger systems, giving rise to AI-based applications. Recent advancements in distributed systems have further enhanced this process, allowing the training of models to be distributed across multiple devices or enabling different parts of a model to run in parallel across various systems. Distributed machine learning is now powering a range of applications designed to enhance user experience, such as keyword suggestions on smartphones, which are supported through federated learning.

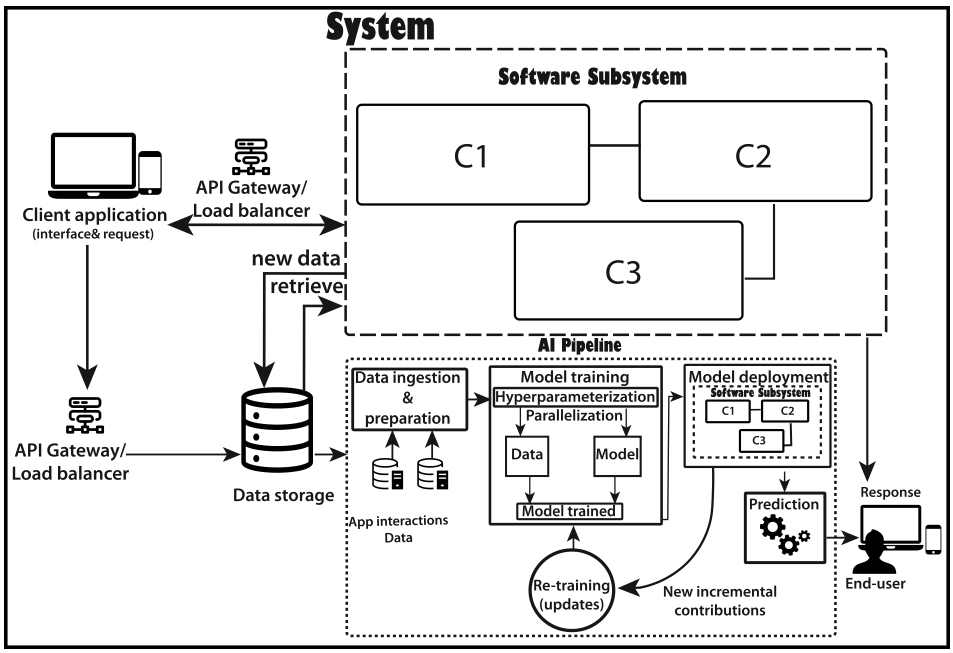


Figure 2: AI pipeline as a component of a (distributed) system

From the user's perspective, AI-based applications are designed to assist in human decision-making. These applications have the ability to perceive their operational environment, process the collected data, and recommend the best course of action to achieve optimal outcomes. The primary goal of AI-based applications is to enhance decision-making by achieving objectives that are defined by humans during the development process [198]. As such, the performance of these applications is closely linked to the quality of the data (i.e., the dataset) available for learning. The more data available, the greater the likelihood that the model can infer the optimal decision. This also means that large datasets require significant computational resources to process and train the models effectively (larger distributed systems). As challenges like data scarcity and quality variations are overcome, increasingly sophisticated AI models are emerging. Furthermore, recent advancements in data distillation techniques are enabling data purification, allowing AI models to create more accurate representations, which in turn improve their learning capabilities and decision-making performance [53, 328].

1.2.1. Evolution of system architectures

We now describe the evolution of system architectures, tracing their progression from classical client–server designs to the integration of distributed machine learning and sophisticated AI pipelines.

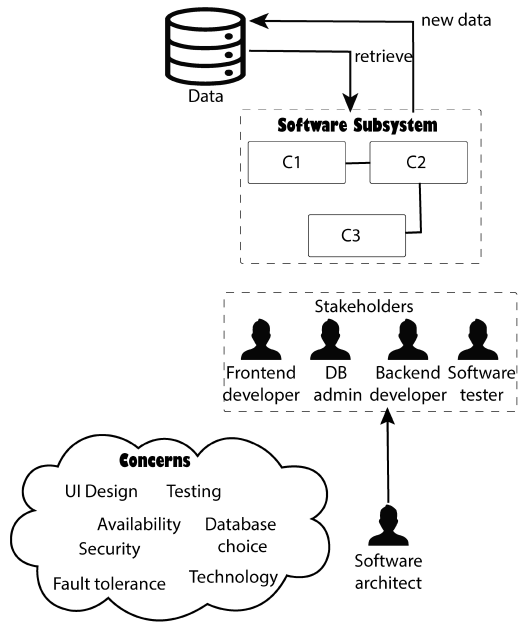


Figure 3: Client-server architecture [310]

Modern architectures: Modern applications have evolved considerably from its fundamental client-server architecture. At the same time, increasing attention has been given to designing and developing systems that are more intelligent. In early developments, in a basic client-server architecture illustrated by Figure 3, end devices acting as clients send requests to the server. At the server, the request is then processed, and a response is sent back to the client. At the core of this architecture is the data component that stores data and from which information is retrieved to process client requests, the software subsystems (C1, C2, C3) handling distinct functional tasks, and various stakeholders with varying concerns and responsibilities. Client–server architectures are traditionally shaped by priorities concerns such as functionality, performance, rigorous testing, and fault tolerance, ensuring reliability and efficiency in system operation.

Over time, more advanced architectures are designed to collect data in a centralized manner (at the server) from users interacting with applications. This data is then used to train machine learning models to improve certain functionality over time. Figure 4 illustrates the machine learning architecture where the basic client-server architecture is extended with a machine learning learning subsystem and more stakeholders. This new sub-system integrates data, algorithms, and

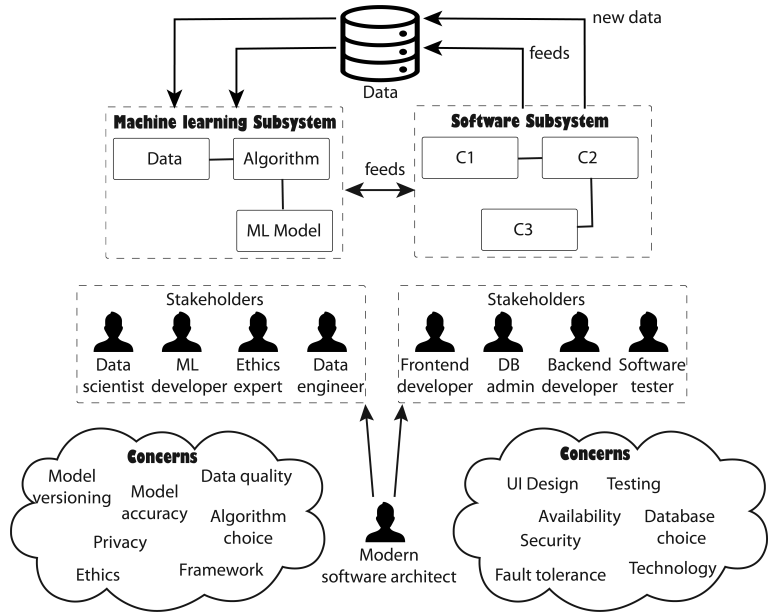


Figure 4: Machine learning architecture [310]

model training pipelines in Figure 6(a) to produce models that can improve the functionality of the software subsystem to offer specialized services to the clients. Unlike client–server architectures, machine learning–based systems prioritize concerns such as model versioning, accuracy, algorithm selection, data privacy, ethical considerations, and data quality.

Further developments have made these architectures capable of collecting data from clients in a distributed manner, such that more robust datasets can be used to train models. Currently, a global model is trained by data contributions of clients collected in a privacy-preserving manner, e.g., using federated learning. Once trained, this model is then propagated to all the end devices. Figure 5 extends the ML architecture presented in [310] to depict the latest advances of distributed training. Federated learning and other distributed learning architectures prioritize concerns such as privacy preservation, managing data heterogeneity across clients, and effective strategies for client selection, ensuring both security and fairness in the learning process.

1.2.2. Standard machine learning pipeline for AI model construction

Systems and applications equipped with AI models implement machine or deep learning (ML/DL) pipelines that facilitate their construction and incremental improvement over time. The standard pipeline for building an AI model can be summarized in Figure 6(a). Applications implement these typical steps to update models continuously as new data contributions are obtained. In the first step (data collection), available data is cleaned and prepared using common methods to enhance its quality, e.g., missing data, removing duplicates, and data augmenta-

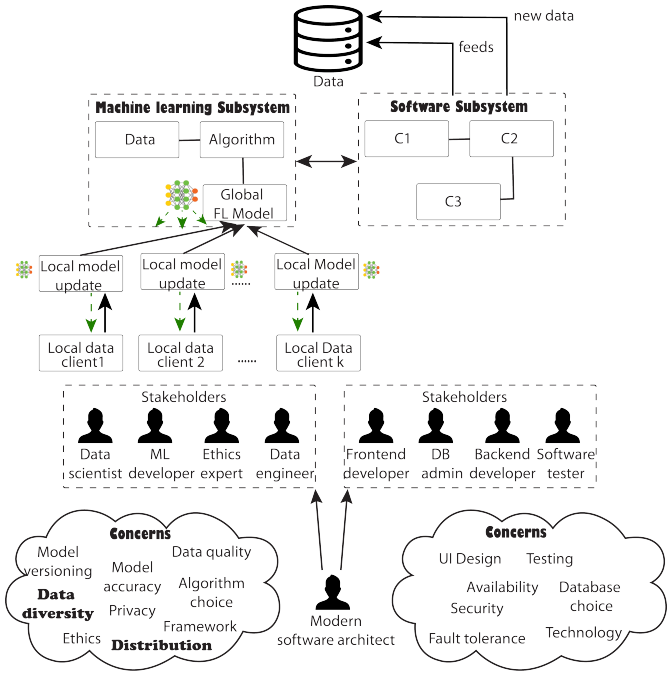


Figure 5: Federated learning architecture

tion [143]. After this step, data is transformed into a suitable input for the AI algorithm, meaning data is labeled, e.g., using human annotators. Next, the training process takes place. Here, an algorithm is selected, e.g., Random Forrest, Support Vector Machine; then the training process is decided, e.g., data parallelization or model partition [219], and the model is evaluated, e.g., using cross-validation [429]. Lastly, the model is deployed, and the performance is evaluated within applications. In classical architectures, models require re-training and re-deploying as new data contributions are obtained. In newer paradigms, such as federated learning, the model is achieved through a global aggregator that combines contributions from clients, such that the resulting model is propagated back to all the contributors.

1.2.3. Advanced AI pipelines: augmenting intelligence

Standard machine learning pipelines can be easily augmented to increase functional capabilities. However, the implementation of the AI pipeline is not that straightforward in newer paradigms. Indeed, there are some nuances in the operationalization of the pipeline for model construction in the various distributed paradigms, such as federated learning, large language models (LLMs), and retrieval augmented generation (RAG). These challenges stem from their distributed architecture, orchestration, and specific requirements [28, 222]. Below is an examination of the distinct operational complexities inherent to federated learning, large language models, and retrieval augmented generation pipelines among several other approaches.

Federated learning pipeline: In federated learning, the pipeline involves: i) client (device) selection based on resource availability, ii) weight sharing to devices from a central coordinator, iii) on-device (local) model training with data, and iv) training update transfer from devices to the central coordinator for model aggregation. Some complexities of the workflow include communication complexity, which results from FL's need to coordinate parameter exchanges across potentially hundreds or thousands of distributed nodes. This challenge pertains to the rate of parameter exchanges within the FL framework and its subsequent effect on system efficiency. The communication overhead becomes particularly problematic when integrating FL with other AI pipeline components, as each additional integration point multiplies the coordination requirements. Similarly, heterogeneity in data and the resources of participating devices presents a major challenge that can compromise the training operation and performance of the trained model. Because FL involves different clients (devices) using local data, the training data is very diverse in quality and quantity, which makes the operation susceptible to data. As such, the performance of the aggregated model can be compromised by a client with poor data contribution in the operation.

Large language model pipeline: The large language models are foundational models whose construction is achieved by leveraging a vast and diverse amount of datasets, from the entire web corpora, code repositories, and other multimodal sources, spanning billions of tokens that are continuously running on several large-scale distributed GPUs and TPUs [470]. The pipeline involves several computationally expensive phases of pre-training, fine-tuning, and alignment with human feedback and rigorous evaluation across benchmarks [330]. Commissioning this multi-stage pipeline presents significant overheads, particularly in computational scaling, where training requires coordinating thousands of accelerators across distributed clusters with sophisticated parallelization strategies, including data, model, and pipeline parallelism [470]. The computational overhead becomes especially problematic when integrating various optimization techniques such as gradient synchronization, memory management, and fault tolerance mechanisms across heterogeneous hardware configurations. Similarly, data quality and consistency challenges arise from processing web-scale corpora that contain biased, duplicate, and toxic content, requiring extensive preprocessing pipelines involving deduplication, filtering, and tokenization at unprecedented scales.LLM hyperparameters further complicate the challenges as the process is highly demanding in terms of time and financial resources. Achieving optimal hyperparameter tuning, learning rate scheduling, and checkpoint management across training runs can span weeks or months, while simultaneously managing the enormous financial costs that can reach hundreds of millions of dollars for state-of-the-art models.

Retrieval augmented generation (RAG) pipeline: The retrieval augmented generation, a general-purpose fine-tuning framework for enhancing pre-trained models [259], is yet another modern paradigm with its unique challenges. RAG pipeline

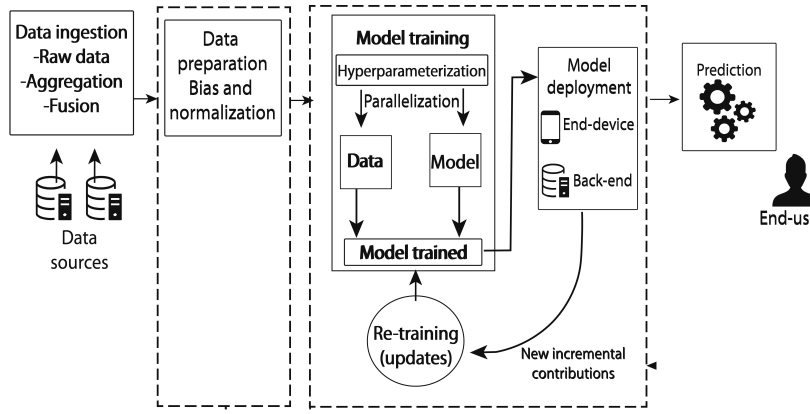


Figure 6: Standard pipeline to construct machine learning models

integrates steps involving i) document ingestion in segments where raw documents are preprocessed, chunked, and converted into vector embeddings stored in vector databases, ii) query processing and retrieval where user queries are embedded and matched against the database using hybrid search to retrieve relevant passages, iii) retrieved context augmentation where passages are re-ranked and integrated with the user query through prompt engineering, and iv) grounded generation where the LLM generates responses based on training knowledge and retrieved context, producing accurate, source-attributable answers while minimizing hallucinations. Deploying the pipeline can be complicated because orchestrating RAG systems requires complex coordination between multiple components(document processing, vector databases, LLMs, re-ranking stages) for synchronization. Besides orchestration, scaling RAG for ensuring performance is very resource-intensive as it requires both large storage infrastructure and heavy computing clusters [68]. Privacy and security considerations introduce additional layers of complexity. RAG systems accessing proprietary knowledge bases must implement fine-grained access controls at the retrieval level, ensuring users only receive content they are authorized to view. The expanded attack surface includes risks of data poisoning through malicious document injection and potential leakage of sensitive information in generated outputs [167].

1.2.4. Benefits of AI in systems and applications

The transformative potential of AI is unprecedented and enormous, which is driving rapid adoption. AI is increasingly embedded into modern applications to improve the experience, perception, and interaction between users and digital applications [75], providing functionality that facilitates application usage and value to users. Examples of this include advanced personal virtual assistance (Siri, Alexa, Bixby, Google assistance), Chatagent (ChatGPT, Gemini, Deepseek, Cluade), for e-commerce recommendations [455, 243], optimal route planning for practical drone delivery [161, 156]. Besides impacting modern application development,

AI has transformed critical sectors and infrastructures such as healthcare [341], transportation [345], finance [466], education [47], energy [380], agriculture [276], aviation [168], and military [354] to mention some.

However, AI is not immune to errors, which can lead to AI incidents. These failures in decision-making raise significant concerns about the safety and reliability of its operations. More importantly, in critical sectors that are considered to be high-risk domains where their failure can be catastrophic. AI often operates as "black boxes" with a limited or no understanding of the decision-making process, raising worries when it is responsible for making consequential decisions that can affect human lives. In addition, security vulnerabilities, like those related to AI infrastructure arising from architectural design (e.g., insecure APIs allowing unauthorized model access, inadequate input validation leading to prompt injection, poorly secured data pipelines enabling data poisoning.), data and other sources present exploitation channel for malicious actors, potentially compromising sensitive data or system integrity. These and additional ethical concerns necessitate the regulation of AI and the consideration of trustworthy computing principles for laying the foundation for establishing the trustworthiness of AI in line with regulatory requirements.

1.3. Trustworthy computing systems

Trustworthy computing principles and practices are the building blocks for developing computing systems that are inherently secure, reliable, and trustworthy. This initiative was pioneered by Microsoft in the early 2000s to address the incessant system failures caused by the increasing exploitation of system vulnerabilities by threats that compromised security and integrity [270]. This effort shifts computing priority from performance-centric to prioritizing users' trust in systems by emphasizing some foundational principles, such as *reliability, security, privacy, and integrity attributes,* to ensure systems behave predictably and safeguard the interests of users [209]. Consequently, stakeholders have to develop systems that are available when needed, exhibit expected behavior, protect data integrity and confidentiality, and operate safely without creating harm to users or the environment.

These foundational principles associated with the attributes of trustworthy computing systems have broadened over time as the field has matured. More attributes have been considered, such as **transparency, availability, and accountability**. Altogether, the comprehensive set provided the basis for the formal methods used in research and industry for designing, verifying, and certifying systems for deployment, especially in application contexts that involve sensitive data and operations. In addition, they are not merely relevant for the security of traditional systems but are fundamentally essential for addressing the contemporary challenges of AI Safety. Since AI systems run on computing infrastructure (hardware, software, networks), the principles inherent in trustworthy computing are essential prerequisites

for Trustworthy AI.

However, unlike traditional computing systems that are deterministic and predictable in behavior, AI systems are complex computing systems driven by probabilistic models and demonstrate the potential to cause unintended consequences at scale [345]. This complexity results from the fact that they generate inferences based on complex statistical relationships rather than following explicit program rules. This raises the need for establishing mechanisms to observe and verify the trustworthiness of the generated outcomes from AI. Moreover, their complexity stems from the data (input) and architectural requirements for their functionality. Consequently, trust verification methods for traditional systems such as simulation or testing are inadequate to capture the trustworthiness of AI systems [449]. This necessitates conscious reconsideration and extension of trustworthy computing principles to develop frameworks designed to ensure AI is not only technically robust but ethical, transparent, and secure.

1.4. Trustworthy artificial intelligence

Trustworthy artificial intelligence is concerned with establishing trust throughout the entire life cycle of an AI system for the lawful, ethical, and robust use of AI systems in any context. Existing solutions analyze AI models through post-defacto verification [260], meaning only after the model has been developed and deployed. As a result, current AI deployments in applications face substantial challenges that undermine their trust in society. For instance, several incidents of self-driving cars endangering other road users and pedestrians have been documented [435]. Similarly, the inherent biases from their training data can result in discriminatory outcomes during deployment, as seen in the Amazon gender bias recruitment system [118] and the racial bias judicial system [125]. The reality of AI's tendency to cause harm to users and society at large necessitates that the design, development, and deployment process must be transparent, safe, and reliable.

Governance initiatives from UNESCO [426], ISO [215], OECD [325], and countries like the US and regions such as the EU provide needed guidelines for developing trustworthy AI. These initiatives have extended the principles of trustworthy computing by incorporating legal and ethical considerations as part of the requirements for the trustworthiness of AI. The purpose is to institutionalize these considerations for the ethical development of AI so that human rights and values are not undermined by AI when deployed into society. The EU AI Act [108] is the first comprehensive governance framework for AI and regulates the commissioning of AI within the Union. It adopts a product and a risk-based approach by classifying AI-based products into levels according to their risk impact, such as *"High-risk AI systems", "Limited-risk AI systems", and "Minimal-risk AI systems"*. It further enumerates requirements and obligations to be fulfilled when designing each category of AI system to deem the system trustworthy. These requirements are the properties (attributes) for AI trustworthiness, and they include

the following: **safety and technical robustness, human agency and oversight, explainability, fairness and non-discrimination, privacy and data governance, transparency, and accountability**.

The high-level expert group on artificial intelligence (HLEG AI) in the EU grouped the attributes into legal, ethical, and technical, to provide the three foundational tenets of the EU AI Act [198]. Under the technical category, the robustness attribute entails the resilience of the system to both accidental failures and intentional attacks on the system. It is closely related to explainability, which demands transparency and the ability to provide clear, interpretable justifications for the decisions of AI systems. On the ethical front, fairness and non-discrimination address substantive and procedural biases that can arise in data and algorithms. Additionally, privacy and data governance are critical attributes, ensuring data is handled securely and responsibly in line with regulations. Attributes under the legal category are primarily concerned with ensuring accountability by establishing clear lines of responsibility for AI outcomes.

To ensure the safe and responsible use of AI, emerging regulations increasingly mandate that the expected attributes of AI systems be addressed from the earliest stages of development, including and even before requirements elicitation. However, the current landscape for developing AI-enabled software and analyzing its properties remains underdeveloped across the entire software life cycle. For example, existing requirements management tools often lack the capabilities needed to effectively capture, track, and assess the specific implications of integrating AI into software systems. Indeed, in a tool evaluation conducted within the context of an AI-focused project, only one out of five selected tools, chosen based on their potential to support AI-specific requirements, was able to partially meet the defined evaluation criteria (see appendix B). This highlights that the specification and management of AI-related requirements remain in a nascent stage, particularly regarding trustworthiness and regulatory compliance.

Categorizing the effects of trustworthy properties from different perspectives is crucial, as there is often no direct one-to-one mapping between technologies, socio-technical requirements, and human perceptions. From a legal standpoint, trustworthiness in AI requires privacy and accountability [198]. Privacy ensures that AI systems protect user information, respect user autonomy, and operate with transparency, while accountability establishes mechanisms for audit and redress if adverse outcomes occur. For AI systems to be deemed legitimate within the EU governance framework, they must meet these criteria.

From an ethical perspective, AI trustworthiness is closely linked to fairness and human oversight. The fairness property demands that AI systems operate without bias, discrimination, or rights violations, ensuring equality for all users. Human oversight, on the other hand, guarantees meaningful human involvement in the AI development and deployment process, safeguarding human autonomy and values. These two properties serve as the foundation for ethical considerations throughout the entire life cycle of AI systems and must be upheld by all stakeholders involved.

Finally, from a technical perspective, trustworthy AI focuses on ensuring safe, responsible, and effective deployment. This encompasses the analysis of accuracy, robustness, explainability, and transparency. Accuracy refers to the AI system's ability to make precise judgments and predictions, while robustness ensures that the system operates securely and reliably across different contexts. Explainability allows AI systems to clearly communicate their decision-making processes, and transparency provides insight into their internal mechanisms and capabilities. These properties are essential to ensuring that the AI system delivers on its value proposition, meeting expectations without compromising on performance, technical integrity, or safety.

Collectively, the requirements and other guidelines of the Act form a comprehensive framework that addresses the multifaceted aspects of AI trustworthiness, balancing legal compliance, ethical considerations, and technical performance to ensure responsible AI development and deployment.

1.5. Human oversight and AI regulations

Human oversight is the strategic involvement of humans in the development and deployment of AI systems, to leverage their expertise and judgment for monitoring and intervening in AI processes and operations [405]. Human oversight implementation encompasses a range of mechanisms for monitoring, understanding, influencing, evaluating, and mitigating AI systems risks [405].

1.5.1. The need for AI regulations

The severe consequences of AI failures and errors compel the obligation to define responsible stakeholders for accountability. Meaning that, when AI systems apply incorrect medical diagnoses for patient treatment, or when self-driving vehicles strike pedestrians, or when chatbots misinform the public, someone needs to take responsibility. This is because such errors can cause significant harm to humans, sometimes leading to death or financial loss as seen in the case of Uber's self-driving car testing incident [48, 401], Tesla's autopilot failure [123] and Air Canada's chatbot misinformation incident [461]. Establishing responsible stakeholders for these consequences necessitates human intervention to ensure that specific individuals or entities are held liable and accountable for the failures of AI systems. For instance, the test driver in the Uber self-driving incident was found liable for negligence [48, 401], and Air Canada was held accountable for the wrong information provided to customers by its chatbot [461]. Besides the need for accountability, AI systems often learn and amplify biases present in their training data, leading to discriminatory outcomes. Examples include biased hiring algorithms disadvantaged certain demographic groups and predictive policing tools disproportionately targeting minority communities. Despite the sophisticated pattern recognition capabilities of AI, it lacks the understanding of societal stereotypes and implications. Human oversight is therefore crucial to

identify, mitigate, and correct such biases before they cause real-world harm and erode societal trust.

Category	Risk Impact Level	Provision of EU AI Act	Human oversight requirements
High-risk AI System	Critical/Significant	Robust human oversight and stringent compliance	Proportionate and effective control measures.
			Built-in decision intervention mechanisms
			Real-time monitoring and control capabilities
			Training and awareness framework
			Risk assessment and mitigation
			Documentation and logging of oversight activities
			Trustworthiness, ethics and legal compliance
Limited-risk AI System	Low risk	Ethical compliance	Transparency obligations
			Disclosure requirements
Minimal-risk AI System	No risk/Negligible risk	No regulation and human oversight requirement	Voluntary codes of conduct

Table 1: Human oversight risk categorization, provisions and requirements for various AI system categories according to the EU AI Act

Significant emphasis has been put on the implementation of human oversight to provide assurance of safety by addressing the potential risks associated with AI systems and to ensure transparency in the decisions of AI systems [405]. The implementation responsibilities of human oversight depend on the type of AI system that is being built [7]. Table 1 highlights the human oversight responsibilities for various classes of AI systems according to the EU AI Act. *"Minimal-risk AI systems"* are systems considered to pose negligible risks to individuals or society, such as AI-enabled recommender systems, grammar checkers or spam filters. The AI Act does not mandate specific human oversight requirements for these systems but encourages the development of ethical guidelines and voluntary codes of conduct. *"Limited-risk AI systems"* are systems that operate with minimal decision-making influence and low potential for harm to users, the environment, and society. Typically, they are employed for procedural tasks like data transformation, document classification, and duplicate detection, which require primarily transparency-focused oversight rather than strict controls. The AI Act mandates clear disclosure mechanisms that inform users about the AI-driven nature of the system, its purpose, decision processes, and limitations, fostering trust while preventing potential harm. Notable examples in this category include chatbots, conversational agents, deepfakes, emotion recognition systems, and AI-generated content. The **"High-risk AI systems"** are the main focus of the AI Act. They require robust human oversight due to their potential impact on health, safety, and fundamental rights in critical sectors like healthcare, transportation, and law enforcement. These systems must implement comprehensive risk management protocols alongside specific human control mechanisms (HITL, HOTL, HIC), while adhering to strict requirements for data governance, technical documentation, and cybersecurity. Additionally, they must incorporate monitoring systems

capable of detecting anomalies and enabling intervention, with clearly defined roles and responsibilities distributed among various stakeholders to ensure proper supervision.

The implementation of human control mechanisms in the development process and operations of AI systems, for necessary intervention, can be achieved through varying levels of human involvement using these approaches. human-in-the-loop (HITL), human-on-the-loop (HOTL), and human-in-command (HIC). HITL entails human intervention in every decision cycle of AI systems. Under this approach, human approval is needed for any system action. An example of the implementation of HITL includes subjecting every prediction generated by health monitoring AI systems about patients to healthcare providers (human experts) to make final assessments and intervention decisions. The HOTL approach engages humans in supervisory capacity during the design cycle and for ongoing monitoring of the operation of AI systems. Human control is relaxed to enable AI to perform tasks, make decisions and take actions under human supervision. Common examples include the remote monitoring of self-driving cars and delivery robots for human intervention when they encounter challenging situations, or having fraud analysts review potential transactions flagged for fraud by a fraud system. The HIC approach avails humans with ultimate control over the AI system. Humans decide when, where, and how to deploy AI systems according to application situations.

1.5.2. Our position in the state-of-the-art: human oversight and human-in-the-loop

As mentioned earlier, all regulatory and economic frameworks have recognized the need for trustworthiness in AI. As a result, several initiatives, projects, and efforts are ongoing to define how to verify it. EU projects, such as EU TRUST-AI (https://trustai.eu/), EU SPATIAL (https://spatial-h2020.eu/), and EU TAILOR (https://tailor-network.eu/), have proposed principles and guidelines to ensure trustworthiness in AI development practices. Likewise, leading technological vendors have proposed frameworks to achieve AI trustworthiness, including IBM's AI Fairness 360, the what-if tool, and ML Fairness Gym of Google, Microsoft's Fairlearn, LinkedIn Fairness Toolkit (LIFT), AT&T Software System to Integrate Fairness Transparently (SIFT), and Fat Forensic. Other initiatives also include the PwC AI trust index, AI trust and transparency of Microsoft, and the AI Impact Assessment of Open AI. In parallel to this, development toolkits have also been released by private vendors and open-source communities. For instance, Google's model card toolkit measures transparency in AI models. Other development initiatives to verify the integrity and robustness of AI include open-source SHAPASH [377], IBM AI explainability 360 toolkit [37], Microsoft Interprete ML, and IBM Adversarial Robustness 360 toolkit. While there is a clear overlap between all these works [431, 432], a key challenge that remains unexplored is identifying essential and general requirements of trustworthiness.

Regulatory trustworthiness mandates human oversight in AI developments. While multiple frameworks have been developed to measure different trustworthy properties [231, 217], it is still unclear the role that humans play in the monitoring and supervision [249, 244]. XAI methods are the most common method to communicate the logic of AI models to users via (optimized) explanations, numerical values, visual diagrams, and so on [34]. At the machine and deep learning levels, several tools and frameworks are available to tune the inference process of AI models. For instance, TensorLeap (https://tensorleap.ai/), Neptune AI (https://neptune.ai/), and Comet ML (https://www.comet.com/site/).

Human oversight methods enable individuals to monitor and interact with AI inference behavior, with interventions possible at any stage of the machine learning pipeline (Chapter 2 provides a detailed and systematic survey). However, allowing humans to meaningfully influence model inference and make adjustments requires advanced human-in-the-loop mechanisms and interfaces that can both capture human input and present relevant insights about model behavior within the context of the application. Trustworthy AI encompasses many properties, but their importance often depends on the application domain. For example, Figure 7 illustrates use cases such as drones for environmental monitoring and autonomous vehicles. While both systems embed mechanisms to characterize and quantify trustworthiness, the priorities differ from the user's perspective. For drones, the ability to detect surrounding humans across different demographics highlights the importance of AI fairness. In contrast, for autonomous vehicles, user trust may rely more heavily on accurate navigation and parking assistance, emphasizing AI performance. These limitations indicate a research gap, leading to the following research question: ***"How can human oversight approaches be integrated into AI-enabled applications to monitor and contribute to their trustworthiness?"***. This is the primary question guiding this research.

Figure 7: Human operators monitoring the behavior of AI-based systems

1.6. Research goal and contributions

This research investigates the development of new methods and mechanisms for human oversight and human-in-the-loop approaches to monitor AI systems throughout their critical steps in the pipeline. Figure 8 maps how each contribution presented in this work is mapped to the AI pipeline. By integrating human oversight mechanisms into the pipeline, humans can monitor AI model inference, contribute high-quality data for model development, and respond quickly in case of AI failures. Each contribution addresses the following objective in its human oversight design.

1. (Contribution 1) To incentivize human participation through social dynamics, encouraging contributions of high-quality data for training AI models.
2. (Contribution 2) To monitor the trustworthiness of AI models during inference when they are deployed within system architectures.
3. (Contribution 3) To monitor runtime abnormalities during the training or execution of AI models, enabling human operators to intervene proactively

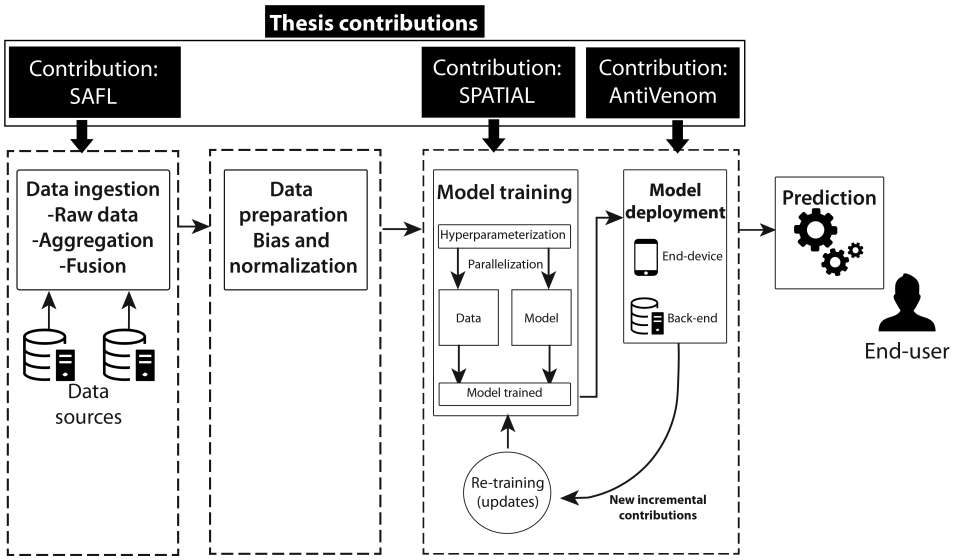


Figure 8: An overview of how each research contribution fits into the stages of the AI pipeline

Each contribution is then presented as follows:

1.6.1. SAFL for model training with human-in-the-loop

In Chapter 3, we introduce the concept of social-aware federated learning (SAFL), which leverages social connections to enhance training contributions in federated learning (FL). SAFL serves as a data collection mechanism that leverages human participation to enhance the quality of data used for training AI models. We address a key limitation in FL where devices may lack the capabilities or incentives to contribute effectively to model training. By enabling task delegation to trusted social contacts and implementing collaborative incentive mechanisms, SAFL increases both the quantity and quality of contributions to the global model. Through a controlled user study involving 30 participants under two compensation-sharing conditions, our work demonstrates that individuals are willing to collaborate and outsource tasks, especially when incentives are structured to benefit both the initiator and the delegate. Beyond establishing a functional prototype and validating the approach experimentally, the study also lays out a research road map identifying the challenges and opportunities of integrating social mechanisms into FL ecosystems, offering a foundation for scalable, trust-enhanced, and socially motivated federated learning systems.

1.6.2. The SPATIAL architecture for AI trustworthiness monitoring

In Chapter 4, we designed and developed SPATIAL architecture, a proof-of-concept system designed to augment modern applications with the capability to gauge and monitor the trustworthiness of AI inference capabilities in a human-in-the-loop

mechanism. SPATIAL supports both the training and inference of AI models, enabling trustworthy metrics to generate quantifiable indicators that can be visualized by humans through a dashboard. Our work examines the evolution of system architectures and enhances modern designs with capabilities to assess AI trustworthiness. These assessments generate quantifiable insights into AI inference performance. Through the dashboard, operators can interpret these insights, identify potential issues, and implement necessary corrective actions. Through comprehensive benchmarks and real-world industrial application experiments, we demonstrate SPATIAL's effectiveness in augmenting applications with trustworthiness metrics while simultaneously acknowledging the increased complexity of developing and maintaining such AI-integrated systems. SPATIAL enables the verification of AI systems for potential audits and ensures compliance with accountability regulations, thereby addressing the critical need for transparency and oversight in AI development and deployment.

1.6.3. AntiVenom for proactive human oversight

The increasing use of AI is enabling more autonomous applications, such as service robots and delivery drones. While the trustworthiness of these systems can be analyzed using standard mechanisms, such processes are often time-consuming, limiting humans' ability to take proactive action in critical scenarios. For example, an autonomous drone delivering a package could become the target of an attack. Assessing its trustworthiness might require recalling the drone to a laboratory, rather than allowing humans to make on-site decisions. This underscores the need for proactive monitoring of AI robustness. Thus, in Chapter 5, we present two complementary approaches to address this deployment challenge: XAI methods for detecting abnormalities in AI models, and performance-based methods—implemented in the AntiVenom framework—for rapid detection of potential attacks. AntiVenom is a lightweight, scalable solution for detecting data poisoning without requiring modifications to the AI model. It is compatible with any distributed machine learning paradigm, including federated learning (FL) and split learning (SL). By leveraging device-level metrics such as CPU and memory usage, AntiVenom detects anomalies linked to poisoned model updates without direct access to the AI model itself. This capability is particularly valuable for autonomous drones operating in dynamic environments, where distributed learning enables continuous model updates. The strength of AntiVenom lies in its independence, ability to classify different types of attacks, and minimal interference with device resources, making it ideal for real-world, large-scale drone operations. When combined, AntiVenom and XAI-based approaches form a robust defense mechanism: AntiVenom ensures efficient, real-time anomaly detection at the edge, while XAI provides interpretability and transparency for human oversight. Together, they significantly enhance AI robustness in autonomous drones, offering a powerful solution for trustworthy, resilient, and scalable city-wide deployments.

1.7. Scope of the thesis

This research advances trustworthy artificial intelligence (AI) with a focus on human oversight, developing mechanisms that allow humans to monitor and remain involved in AI processes.

This thesis is deliberately scoped. First, it focuses on evaluation and instrumentation mechanisms for AI trustworthiness rather than developing new models or algorithms, using existing AI models as assessment targets. Second, it emphasizes modular, microservice-based architectures in cloud–fog–edge environments, validated on use cases such as emergency communication, 5G/6G networks, and IoT infrastructures. Third, human oversight is explored through human-in-the-loop and human-on-the-loop paradigms, supported by metrics, dashboards, and trust monitoring systems, rather than fully autonomous frameworks.

The research acknowledges its limitations. Evaluation scenarios are representative but not exhaustive, and large-scale industrial deployment is outside the scope. Validation uses selected datasets and models without claiming full generalizability. Trade-offs between trust properties (e.g., fairness vs. accuracy, privacy vs. utility) are not fully resolved. Federated learning experiments in SAFL are limited to specific simulations. Engagement with regulatory frameworks such as the EU AI Act and GDPR focuses on technical alignment, not comprehensive legal or economic analysis.

In summary, this thesis provides practical, modular, and technically sound mechanisms for evaluating and enhancing AI trustworthiness with human oversight, while recognizing that full-scale deployment, exhaustive coverage of trust properties, and broader interdisciplinary considerations remain outside its scope.

2. TRUSTWORTHY AI IN PRACTICE: A COMPREHENSIVE REVIEW OF HUMAN OVERSIGHT AND HUMAN-IN-THE-LOOP APPROACHES

In Chapter 1, we outlined the research motivation, the goal, and contributions of this thesis in advancing practical trustworthy AI with human oversight. In this Chapter, we establish the foundation for the contributions of this thesis by presenting a systematic survey of the state of the art on trustworthy AI development. Our methodology leverages a structured literature review approach that enabled the comprehensive collection and selection of publications based on defined criteria. The review covers oversight mechanisms, technical methods, and supporting tools, with a particular emphasis on how human oversight mechanisms and trust dimensions can be embedded across different phases of the AI lifecycle. By synthesizing the acquired insights, this Chapter not only provides conceptual clarity on the dimensions of trustworthy AI but also lays the conceptual foundation upon which the subsequent technical contributions of this thesis are built.

2.1. Introduction

Artificial Intelligence (AI) has demonstrated transformative potential across diverse societal domains, from medicine to education, finance, and security. However, the opacity and unpredictability of AI have raised significant societal concerns about its safety and responsible usage. Documented failures, such as biased recruitment systems disproportionately disadvantaging marginalized groups [118, 359, 49] and predictive policing tools unfairly targeting minority communities [125], are a few examples of how AI can reinforce existing bias in society. Reports of more than 13,000 incidents of AI between 2018 and 2024 indicate the rationale for the growing public demand for transparency and accountability [316].

In response to these risks, international organizations and national governments have advanced comprehensive governance frameworks for trustworthy AI. UNESCO [426], ISO [215], and the OECD [325] have issued global principles, while major economies, including the EU, US, UK, Canada, Japan, China, Korea, and Brazil, have established their regulatory instruments. The EU has pioneered the foremost comprehensive AI ACT, which introduces binding obligations with compliance deadlines beginning in 2025 [198]. Similarly, the United States has published executive orders and the NIST AI Risk Management Framework [402], and other jurisdictions have issued their national legislation and sectoral guidelines of AI development and usage. A common position across these initiatives is the recognition that unregulated AI autonomy poses systemic risks, and that human control is critical for achieving trustworthy development and deployment. Thus, making human oversight an indispensable requirement for ensuring AI trustworthiness.

Establishing trust in the AI lifecycle is not a trivial task. This is because foundational principles of trustworthy computing that relate to the computing system must be fulfilled alongside emerging AI-specific regulatory requirements. The EU AI Act explicitly identifies human oversight as both a standalone obligation and a crucial enabler of other trust attributes such as robustness, transparency, fairness, and accountability [140]. For high-risk AI systems, the Act mandates comprehensive oversight through varying capacities and programs, ranging from technical interfaces to monitoring dashboards, and training programs [138]. Furthermore, the Act proposes three complementary approaches for establishing human oversight: Human-in-the-Loop (HITL), where humans remain actively involved in every decision cycle; Human-on-the-Loop (HOTL), where humans supervise and intervene selectively during design and operation; and Human-in-Command (HIC), where humans retain ultimate authority to govern system objectives and intervene in critical contexts [140].

While existing research has examined individual trustworthiness requirements and their implementation, the interdependencies between human oversight and other attributes remain underexplored. Human oversight is a meta-requirement, shaping how transparency is realized, how bias is detected and mitigated, and how robustness is validated in practice. However, systematic analysis of these interconnections and their practical implications is still underexplored. In this Chapter, we systematically examine the integration of human oversight into AI design and development, underscoring its significance for achieving trustworthy AI alongside technical safeguards. While governance frameworks consistently emphasize trustworthy framework implementation, they are difficult to translate into practical implementation. To address this gap, we provide a comprehensive review that analyzes how oversight is operationalized through Human-in-the-Loop (HITL), Human-on-the-Loop (HOTL), and Human-in-Command (HIC) mechanisms across the AI life cycle and in relation to key trustworthiness attributes.

Our analysis positions human oversight requirement for trustworthy AI as a meta-requirement that enables the establishment and validation of other trust dimensions such as fairness, robustness, privacy, transparency, and explainability. Grounded in the regulatory context of the EU AI Act, this review systematically maps oversight provisions to concrete control mechanisms, tools, and implementation strategies. By synthesizing insights from literature, governance frameworks, and development practices, we bridge the gap between compliance expectations and technical realization. This regulation-based and life cycle-oriented perspective demonstrates that human oversight constitutes a foundational pillar for ensuring trustworthy AI systems and sets the stage for the structured analysis presented in the following sections.

2.2. Survey scope and methodology

In recent years, publications on trustworthy AI have rapidly grown, indicating increasing interest in the field. Bibliometric analysis from major academic databases shows this growing trend. A Google Scholar search for the term "trustworthy artificial intelligence" returned approximately 10,700 results, with nearly 80% of these publications dated between 2014 and 2024. The annual output shows a steady upward trend, with a notable surge from 2022 to 2024, as illustrated in Figure 9. This surge can be attributed to significant developments during this period, such as the widespread deployment of large language models and generative AI systems, growing awareness of AI's societal implications, and the maturation of various regulatory frameworks aimed at AI governance around the world.

Furthermore, a more focused query that combined the terms "trustworthy AI" + "human oversight" returned 1,420 results within the same time frame, with more than 70% published in 2024. This result suggests that research on the trustworthiness of AI from a human oversight perspective is starting to gain momentum. The majority of the research effort on AI trustworthiness focuses on legal, ethical, or compliance considerations [231, 148], while others analyze individual trustworthiness requirements without addressing their interplay [265]. Despite these efforts, the understanding of the interplay between human oversight and each requirement towards making AI safe and trusted remains unclear.

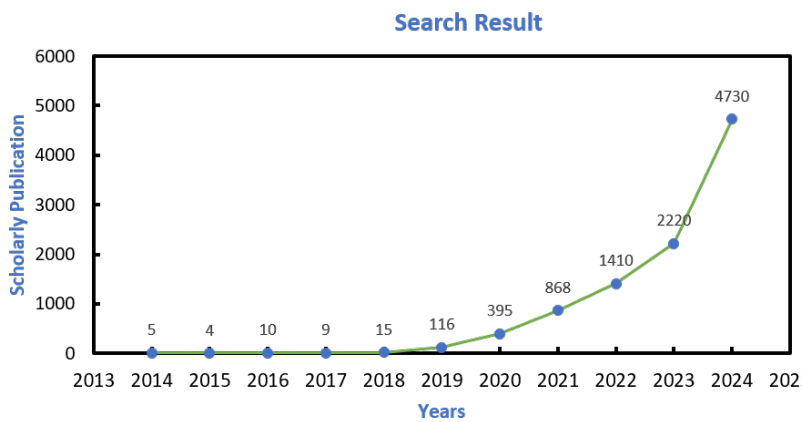


Figure 9: Trustworthy AI search results.

2.2.1. Related survey

Surveys on trustworthy artificial intelligence mostly examine AI trustworthiness by considering its dimensions or proposing implementation frameworks. On the other hand, a small number of surveys have evaluated the domain considering system interaction and trust verification. Table 2 summarizes existing related surveys. Studies that focused on trustworthy attributes examined core dimensions

of trust that are fundamental to building trust in AI systems. For instance, some studies delve into the requirements and explore risk mitigation methods, analyzing security, privacy, and robustness, addressing threats, and detailing advanced defense techniques. Most surveys in this area largely link trustworthiness to three key dimensions: explainability, robustness, and security. These works show that a trustworthy AI system requires clear model interpretation capabilities alongside resilience and protection from threats.

Surveys focusing on frameworks for trustworthy AI examine the implementation of trust across different phases of the AI systems' life cycle, particularly in the domains of connected and autonomous mobility, smart city robotics, and autonomous vehicle scenarios. Some works proposed the standardization of a trustworthy AI framework and guidelines for building trust in AI systems. These frameworks extend beyond the centralized learning environment to include distributed paradigms like federated learning environments for integrating trust requirements during model learning. Overall, these publications demonstrate the relevance of trust framework development and standardization for implementing trustworthiness in diverse AI applications.

Furthermore, the AI system interaction surveys provide insights into the operational dynamics of trustworthy AI systems. Though fewer in number, they extensively cover human-AI collaboration principles, examining trust development between AI systems and their users. Specifically, [321] explores taxonomies of human roles in human-in-loop cyber-physical systems (CPS), offering structured approaches to understanding human involvement in AI operations in CPS. The application domain includes industrial automation, cybersecurity, NLP, and autonomous systems. Lastly, the evaluation methodology survey category focused on verification and validation of trust. This category analyzed existing efforts directed towards establishing an approach for measuring and assessing trust in AI systems during development and deployment. Suggesting the need for robust evaluation methods that can verify trustworthiness across different domains.

While existing surveys explored varying aspects of the trustworthiness of AI, it is of interest to understand the intricacies of achieving human oversight through appropriate mechanisms, as well as the importance of human oversight as an overarching requirement that facilitates the effective implementation of other characteristics of trustworthiness. Our survey addresses this gap by systematically reviewing the regulatory provisions for human oversight according to the EU AI Act. Then we investigate the various components and mechanisms of human control and analyze the connection between human oversight and other trustworthiness qualities. Unlike the existing related surveys, our survey stands out by positioning human oversight not simply as a requirement of AI trustworthiness but as a foundational and enabling requirement. We systematically analyzed oversight mechanisms, particularly human-in-the-loop, human-on-the-loop, and human-in-command, as stipulated by the EU AI Act. By mapping regulatory expectations to practical implementations across the AI life cycle, we demonstrate the connection

Topic area	Total survey	Survey focus	Application domain
Trustworthy attributes	6	Comprehensive analysis of various trustworthy AI requirements and risk mitigation methods [231, 90], analysis of security, privacy and robustness, threats and defense techniques [482], trustworthiness of AI systems through the lens of explainability and robustness [91, 445], interdependencies between dimensions of trustworthiness [172],	federated learning, recommender systems
Frameworks	6	Trustworthiness in each phase of the AI lifecycle [260], methodology for achieving trustworthy AI system in the context of connected and autonomous mobility [225], standardization framework [231], ethical framework for embedding trust in AI system [246], framework for Integrating trustworthiness in variable autonomy (VA) [298], trusted federated learning (TFL) framework [481]	connected and autonomous mobility, smart city, robotics, federated learning
Interactions	3	Human-AI collaboration principles [231], fostering trust between AI systems and humans [447], requirements and taxonomy of human roles regarding human-in-the-loop for cyber-physical systems (CSP) [321]	industrial automation, NLP, cybersecurity, autonomous systems
Evaluation Methods	4	Technologies and methodologies for achieving trustworthiness in real-world application [274], verification and validation of trust [231, 260, 328]	healthcare, autonomous systems, social networks, and environmental systems.
Current survey		This survey contributes by mapping human oversight regulatory expectations to practical implementations across the AI lifecycle by analyzing how human oversight contributes to and enables the achievement of other trustworthiness requirements.	Any domain

Table 2: A summary of existing related survey

between oversight mechanisms and specific trust attributes and implementation tools. This survey fills a critical gap in translating regulatory oversight requirements into concrete, actionable design principles.

2.2.2. Paper collection methodology

To establish a comprehensive list of relevant literature to review, we systematically crafted our search terms through rigorous analyses of definitions and descriptions of AI trustworthiness and human oversight concepts from varying authoritative sources, such as the NIST glossary of trustworthy AI [40], EU AI Act [138], and NIST risk management framework [12] for identification of keywords. Further-

more, we considered each trustworthy requirement and generated representative keywords for each. These keywords were also included in the collection of the search terms, see table 3. We selected Google Scholar, IEEE Digital Library, and the ACM Digital Library for gathering publications for our survey due to their broad and specialized coverage of computer science publications and research resources.

Group Name	**Keywords**
Fundamental	trustworthy, human oversight
Scope	artificial intelligence, machine learning
Context	trustworthiness, transparency, robustness, safety, resilience, privacy, accuracy, performance, explainability, interpretability, accountability, human monitoring, human supervision, human-in-the-loop, human-AI interaction, societal well-being,

Table 3: Group names and keywords

2.2.3. Selection of articles

Our query was strategically executed. We initialized the search with "trustworthy artificial intelligence" between 2014 and 2024 on each database. In total, we got 26,140 initial publications from all databases. Next, we combined the fundamental keywords using the "AND" operator to help refine the result and establish the intersection between "trustworthy AI" and "human oversight". The search term was further extended to incorporate terms representing each trustworthy requirement to ensure the result covers explicitly relevant publications on various aspects of AI's trustworthiness. Additionally, we meticulously employed filtering procedures that excluded demos, workshop papers, tutorials, posters, short papers, and duplicates to focus on relevant research publications. The remaining publications were screened considering peer review status, with only journal articles and conference proceedings from established venues being retained. Each abstract was then evaluated for discussion of both trustworthy AI and human oversight components, along with the assessment of methodology and findings. At the end of the process, our approach resulted in the selection of 203 publications that comprehensively cover the intersection of trustworthy AI and human oversight for review.

Next, we leverage this publication to examine the activities conducted across various phases of the AI life cycle, the various learning paradigms of AI, and the foundational elements of AI regulation necessary for understanding the importance of AI governance.

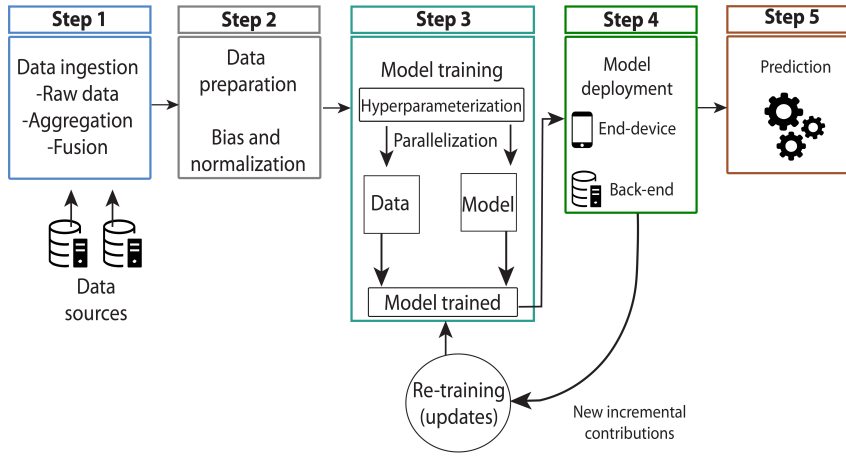


Figure 10: Standard machine learning pipeline for building AI models.

2.3. AI and trustworthy AI

To understand how trustworthiness can be embedded into AI systems, it is necessary to examine the development lifecycle of AI, the stages through which these systems are conceived, developed, and deployed. It provides a structured view of the phases involved, from data collection and model training to deployment and monitoring, each of which introduces distinct trust considerations. The following subsection outlines this life cycle as a basis for analyzing how human oversight and other trustworthiness mechanisms can be systematically integrated across different phases.

2.3.1. AI development life cycle

The steps and processes involved in building and deploying an AI model are illustrated in Figure 10, which represents the standard pipeline for AI model learning and deployment using machine or deep learning algorithms [328, 153]. These steps and processes are summarized as follows:

(a) Data collection (data ingestion): Data is the backbone for AI model development [213]. Data is acquired from several sources and vendors using different methods. It can be acquired via crowdsourcing/crowdsensing methods [362], discovery methods [44, 187], or augmentation [390]. Enough data must be collected so that it is possible to have available data for training, validation, and adaptation of the model. Data can be collected automatically by sensors or participatory methods that request end users for data contributions. For instance, a user may be asked by a 3D map service to take a picture of the street in exchange for compensation [348].

(b) Data preparation: Several data preparation (data pre-processing) steps are required to transform collected data into suitable input for AI algorithms. Data preparation accounts for 80% of the time that experts spend on processes within the standard pipeline [311]. Data scientists are usually faced with datasets that may

include missing or invalid data, resulting in low accuracy and performance during the learning process. Hence, data preparation, or data pre-processing, cleans the data and prepares it for learning. Data preparation encompasses several stages: discovery, cleaning, and labeling are the most common.

(b1.1) Data cleaning and discovery: Several methods to clean data are available for this step, as discussed in the previous section. In real scenarios, however, there is a lack of sufficient data for model construction. Methods to discover data to improve AI models' performance issues have been investigated. Commonly, data discovery methods are categorized into attribute/tuple level and table level discovery [88]. The basic idea of tuple-level data discovery is finding an appropriate external data source with similar table properties and overlapping schemes compared to the in-hand dataset, such that it is possible to fill the missing data with those aggregated contributions [81, 336]. Similarly, table-level data discovery methods provide interfaces that allow users and data scientists to explore the data lake using keywords and extract the related datasets [187]. Analysis of co-founding factors that relate different variables of datasets can also be applied to augment datasets with additional data.

(b1.2) Data labeling: After that is passively prepared, the next step is to assign correct labels, which can be used by AI models to perform correct predictions. The most common method to perform labeling of data with high confidence is human inspection, e.g., Captcha [104]. Other methods for crowdsourcing labeling tasks have also been adopted to improve the veracity and provenance aspects of data, e.g., Mechanical Turk [43]. While these methods can provide data with high confidence in AI usage, they can be biased depending on the human annotator group that performed the task. As a result, autonomous solutions have also been investigated [200].

(c) Model training: Once proper data input is in place for building the model, the training process is configured. The training process can be parallelized over the underlying computing resources based on the data or model partitions [219]. After the model is trained, it must also be tested using data. Cross-validation is a technique that can be used to achieve this. Different variations of cross-validation methods are used based on the amount of data available. Among them, leave-one-out cross-validation (LOOCV) and k-fold cross-validation can be mentioned. LOOCV splits the dataset of size n into two parts (1 and n-1). One-part (1) is utilized to test the model, whereas the (n-1) parts are considered to build the model. This process is iteratively performed n-1 times, where a different data item is used over time for the evaluation. In parallel to this, k-fold cross-validation can also be applied for model evaluation. K-fold involves randomly dividing the data into folds of equal size. The first fold is selected for evaluation, while the rest of k-1 are used for model training. This process is also repeated k-1 times, and each time, the testing fold is changed. In addition, bootstrapping can also be used for model validation, and it is typically used in situations where the dataset is small.

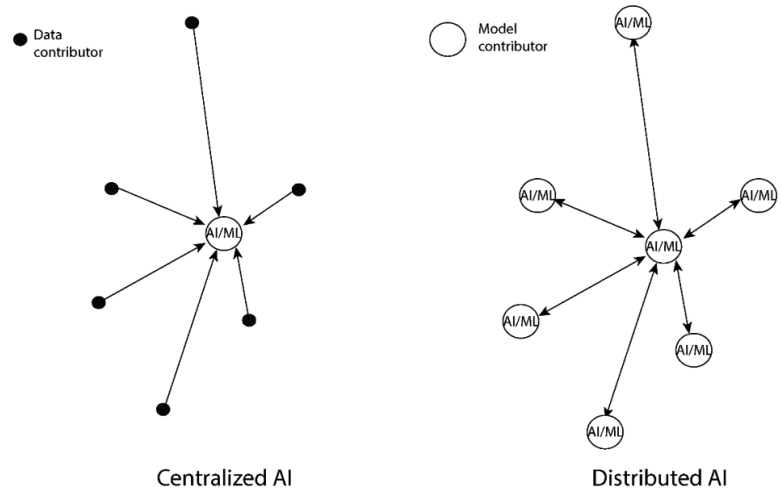


Figure 11: Centralized and distributed machine learning flavors

(d) Model deployment, inference, and incremental training: After evaluating the trained model, it is then deployed within systems and applications. In classical architectures, models are retrained and deployed with them as more data is collected. In newer paradigms, such as federated learning (explained in detail in further sections), the model is retrained by an aggregator, which then propagates a copy of the model to all the contributors.

2.3.2. Approaches for building AI models

AI models construction can be approached using centralized or distributed techniques. The centralized learning approach aggregates data in a single location for model training, while the distributed learning approach leverages multiple devices to accelerate the process. Distributed techniques enable models to learn from diverse, heterogeneous datasets across different sources, enhancing robustness and adaptability to real-world scenarios. As Figure 11 illustrates, each approach requires contributions. However, a key distinction between these approaches is that, in distributed learning, contributors use raw data to learn models locally and provide model updates (weights), rather than raw data. This improves the overall AI system and addresses privacy and scalability challenges inherent in centralized training.

(a) Centralized machine learning: Traditional machine learning architectures rely on a centralized approach, where all training data is aggregated at a single location to develop an AI model. While effective in controlled environments, this architecture presents significant limitations in scalability and privacy. A centralized system is inherently constrained by the volume of data it can process, as well as the communication costs associated with transferring large datasets. Moreover, when models learn from sensitive personal data, the need to copy and store user

information on centralized servers introduces privacy risks and potential regulatory concerns. These challenges highlight the need for decentralized and privacy-preserving alternatives, such as federated and distributed learning, to overcome the limitations of traditional centralized machine learning.

(b) Distributed machine learning: There are several variants in distributed machine learning architectures, although they share common properties regarding data distribution and selection of decentralized devices. Model inference can be distributed across multiple devices using collaborative processing and distributed computing methods [430]. Similarly, model training relies on distributed devices that intercommunicate to share and update parameters, addressing scalability by dividing the training load across multiple machines. There are two main approaches to distributed machine learning: data parallelism, where models are built with different subsets of the data, and model parallelism, where the model is split into parts and distributed across machines, such as training different layers on separate nodes. In addition to this, federated and split learning paradigms enable the building of AI models in a privacy-preserving manner and help overcome issues like data heterogeneity.

There are a number of variants in distributed machine learning architecture, although they share common properties. These architectures are tailored for two particular tasks: training and inference of AI models. Model inference load can be distributed between multiple using collaborative processing and distributed computing methods [430]. Likewise, model training relies on distributed devices to build AI models, and in this process, devices intercommunicate with each other to share and update parameters. Distributed machine learning addresses the scalability issue as the training load is divided across multiple machines. There are two main approaches to distributed machine learning: first, data parallelism, where the models are built with different subsets of the data; second, model parallelism, where the model is divided into multiple parts and distributed across the machines. For example, different layers may be trained on different nodes. Another method to train machine learning models is federated learning, which builds AI models in a privacy-preserving manner and can overcome the problem of data heterogeneity. Due to the large adoption and ease of integration of the approach, our experiments rely on federated learning.

2.3.3. Trustworthy computing and AI

Trustworthy computing refers to a set of properties or characteristics that software must exhibit to be considered reliable and secure. The concept originated when Bill Gates sent an internal memo to Microsoft, outlining key principles that their products should uphold to ensure trustworthiness [450]. This concept has evolved into a distinct domain, defining the trustworthy properties required for computing programs to be deemed trustworthy. Many of these properties overlap with those identified by policy instruments in the pursuit of trustworthy AI.

Developing safe AI requires addressing both technical and socio-technical factors guided by core principles. AI inference capabilities and performance can be assessed through various trustworthy properties. AI trustworthiness extends the properties of trustworthy computing software, incorporating new considerations that account for the probabilistic and opaque nature of AI algorithms and the quality of training data [449]. Trustworthy AI is defined by being valid, reliable, safe, fair, free of biases, secure, robust, resilient, privacy-preserving, accountable, transparent, explainable, and interpretable [260]. These properties are closely interconnected, and it has been demonstrated that multiple trade-offs can emerge between them [86, 442]. For instance, improving explainability might impact accuracy. However, it is important to note that AI trustworthiness is an ongoing process, with its definition continuously evolving through collaboration among technologists, developers, scientists, policymakers, ethicists, and other stakeholders. The mapping of ethical and legal requirements to technical solutions remains unclear. While human oversight is emphasized by AI regulations and policy instruments, its practical application seems ambiguous across AI application domains. Examining different provisions of various AI regulations can help to understand expectations, which can guide oversight initiatives.

Initiatives at the national and international levels on AI-based technologies seek to ensure that their adoption is trustworthy and the implementation aligns with existing ideals. Currently, over 1880 AI policy initiatives have been designed for establishing ethical guidelines, regulatory frameworks, and governance mechanisms across regions and countries around the world [131]. These initiatives translate trustworthy AI principles into actionable governance frameworks to guide AI development and deployment. Table 4 shows leading countries in AI regulation based on the number of policy instruments formulated to govern AI. This demonstrates the global landscape of AI governance, highlighting the proactive efforts of various nations and regions to establish ethical guidelines, regulatory frameworks, and governance mechanisms. Considering the top countries/regions according to their regulatory activities, the United States (US) leads in terms of the number of policy instruments, followed by the European Union (EU) and the United Kingdom (UK). The US regulates AI development and usage through the US Executive Order 13859/13960 [106, 54]. Similarly, countries like the UK, Japan, Australia, Colombia, Canada, China, and Brazil have emerging guidelines and established national policies that govern AI. While these regulations seek to drive these countries' nationalistic agenda, the EU AI Act is globally considered the foremost and most comprehensive legal framework on AI, which serves as the benchmark for other countries. It strategically outlines the plan for the adoption of AI across all member states within the EU and the approach for managing AI technologies risks.

Several research projects and programs, such as SPATIAL, TRUST-AI, AI4EU, TAILOR, ROBUST 6G, MUHAI, have been commissioned in the EU to advance the development of AI technologies that align with EU values and promote in-

SN	Country	Number of AI Policy Instruments	Main Guideline | Legislation | Regulation
1.	United States	33	Executive Order on the Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence
2.	European Union	25	**EU Artificial Intelligence Act ***
3.	United Kingdom	25	The Artificial Intelligence (Regulation) Bill
4.	Japan	19	AI Governance Guidelines for Implementation of AI Principles
5.	Australia	16	AI Ethics Principles
6.	Colombia	13	National Artificial Intelligence Policy
7.	Canada	13	Artificial Intelligence and Data Act
8.	China	10	Interim Measures for the Management of Generative Artificial Intelligence Services
*The EU AI Act is the regulation that is pertinent to this work.			

Table 4: Leading countries in artificial intelligence regulations based on the number of policy instruments formulated to governing AI

novations. In the rest of this survey, we consider the principles and provisions of the EU AI Act for establishing adequate human oversight towards achieving AI trustworthiness. Building upon this context of global AI policy and the EU's leading role, the remainder of this survey will specifically examine the principles and provisions of the EU AI Act, with a particular focus on establishing adequate human oversight to achieve AI trustworthiness.

2.4. Trustworthy AI and human oversight

Human oversight is a critical consideration for making AI systems trustworthy [231] and a crucial instrument for AI governance [255]. AI Regulatory frameworks recognize it as essential for embedding human values throughout the AI life cycle. It encompasses mechanisms, processes, control measures, and capabilities designed to empower human experts to monitor, interact with, and regulate AI systems. Human oversight aims to ensure AI operates within human autonomy, preserving critical decision authority with humans in high-stakes and complicated situations.

In practice, human oversight revolves around various clearly defined roles that humans occupy throughout the AI life cycle, spanning from the initial stages of design through development and deployment. The implementation requires deliberate considerations, such that oversight measures can be proportionate to AI system risks, effective in minimizing or preventing them, exercised by competent persons, and permit documentation and auditability. Providing a robust basis for human experts to effectively monitor, interact, and intervene in AI system operations for ensuring safety, accountability, and transparency in the development and deployment of AI systems [72].

Significantly, the EU AI Act mandates comprehensive human monitoring and intervention in AI processes to preserve human rights and societal expectations in the use of AI systems [198]. Under this regulatory framework, human responsibilities include setting requirements for performance, system decision checks,

supervising operations, and implementing interventions to avert harmful consequences. In addition, the Act further adopted a structured, risk-based approach for categorizing AI systems, which influenced and determined the extent and approach for implementing human oversight. AI systems are categorized into four distinct categories according to their potential harm and stipulated tiered oversight requirements for each category. The categories include **unacceptable-risk AI, high-risk AI, limited-risk AI, and minimal-risk AI**. This approach ensures that the stipulated oversight measures and risk initiatives are implemented proportionately with the level of risk exposure associated with each class of AI system. It prioritizes the safety and trustworthiness of the AI systems in line with governance requirements, according to potential harm from their applications.

Consequently, complying with human oversight requirements confers differential obligations upon developers, providers, deployers, operators, and any stakeholders involved with AI systems. For instance, the provider must design and develop AI systems with oversight capabilities to enable oversight functions and provide technical documentation, while the deployer is mainly responsible for the implementation of oversight measures when deploying and using the system [139]. Similarly, the provisions for oversight vary among categories of AI systems. Unacceptable-risk AI systems do not have human oversight provisions, while other classes of AI systems have human oversight provisions. This is because the unacceptable-risk AI systems are prohibited and not allowed to be developed, while the rest (high-risk, limited-risk, and minimal-risk) are permissible for development and deployment. Among the permissible categories of AI systems for development, the high-risk AI systems category has mandatory and stricter human oversight provisions relative to the rest.

2.4.1. Human oversight requirement for AI-system category

Determining and enumerating the human oversight specifications outlined by the Act for each category necessitates an understanding of the nature of each category from a regulatory standpoint.

(a) Minimal-risk AI system: These are AI-based systems or applications that pose a negligible risk of potential harm. As such, they are considered "no-risk systems". These systems include an AI system used in an AI-enabled recommender system, a spam filter, and grammar and spelling checkers. The Act does not mandate human oversight requirements for this category. However, it demands that they are developed ethically following voluntary codes of conduct [108].

(b) Limited-risk AI system: These are AI-based systems and applications whose deployment does not pose any significant risk, as they have no influence on the outcome of decision-making in the domain of application. The impact of their potential harm on the users, the environment, and society at large is low. They are often used in narrow procedural tasks such as data transformation during data processing, document classification, duplicate detection, etc. [425]. The human

oversight requirements for this category are mainly transparency obligations and disclosure mechanisms that clearly inform users that they are engaged with an AI-driven system, its purpose, decision process, and limitations to prevent harm and promote trust. AI systems in this category include: chatbot and conversational agents, deepfake, emotion recognition, and AI-generated content [6]. Table 5 highlights the main human oversight requirements for the limited-risk AI system. For this class of AI systems, the risk impact level is considered low, and only a provision for ethical compliance is mandated to fulfill the human oversight requirements.

(c) High-risk AI system: This category of AI systems is the most addressed by the AI Act and is at the centre of human oversight provisions of the Act. High-risk AI systems are systems that pose critical risks because they can cause potential harm that can jeopardize the health, safety, or fundamental rights of individuals or groups and society at large. Product-wise, they are considered either as a safety component of any regulated product, such as medical devices, machinery, toys, elevators, personal protective equipment, and radio equipment, or a standalone complete system themselves [7]. In terms of deployment, they are intended for deployment in contexts that are considered to be significantly risky due to the severity of the harm [5]. This context includes the use of AI systems for biometric identification and categorization, management of critical infrastructures, education or vocational training, law enforcement, migration and border control management, and administration of justice and democratic process. The Act mandates and emphasizes stringent human oversight requirements for high-risk AI systems in order to mitigate the consequences of their decisions or outcomes and ensure their trustworthiness and safety. The obligation to comply with human oversight requirements and carry out all the responsibilities delineated by the Act is mandatory for all stakeholders involved in the design, development, and deployment of high-risk AI systems. The allocation of responsibilities to stakeholders aims to foster the trustworthiness of AI by ensuring that all actors within the ecosystem of AI are conscious of the potential risks of AI technology and take appropriate measures that are proportionate to their influence and control to mitigate the risks. For instance, the developers of high-risk AI systems are primarily responsible for establishing a risk management system throughout the life cycle, achieving an appropriate level of accuracy, robustness, and security, enforcing data governance, and drawing up technical documentation and instruction manual [424]. Similarly, the deployers are responsible for using the system according to instructions, assigning human oversight, monitoring operations, operation log keeping, informing developers of incidents, etc. [139]. Besides the developers and deployers, the responsibilities of other stakeholders (importers and distributors) are largely connected to the value chain of AI.

The oversight requirements stipulated by the Act for high-risk systems as summarized in Table 5. As shown in the table, the risk impact directly influences compliance obligations and oversight requirements. In the case of high-risk systems

where the risk impact is considered critical, they must strictly comply with the provision for robust human oversight and stringent compliance measures towards implementing all the human oversight requirements for this class of AI systems.

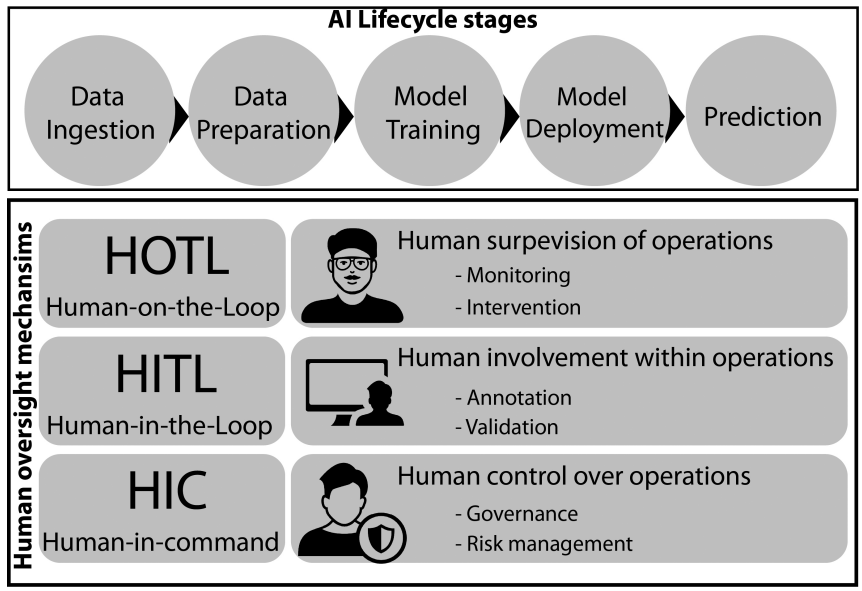


Figure 12: Visual summary of human oversight mechanisms across the AI life cycle. The diagram illustrates the integration of human-in-the-loop (HITL), human-on-the-loop (HOTL), and human-in-command (HIC) roles at different life cycle stages, from data ingestion to prediction, along with their primary control functions

2.4.2. Human oversight control mechanisms/ human intervention approach

Human oversight control mechanisms are the various forms of measures and tools through which humans exercise control, monitor, and intervene in AI decision-making processes, operations, and outcomes for maintaining human oversight. Technically, the inference process of an AI system is a continuous loop of inter-related activities. At each decision point within this loop, several measures are implemented to ensure transparency of the logic involved, accountability, and human autonomy. These mechanisms are crucial for enabling humans to oversee the loop, interpret the decision, and intervene to address or override the decision to prevent unintended outcomes.

Generally, the control mechanisms are designed to put all forms of system automation (autonomy) under human check. There are several levels of human autonomy or control [388] that form the basis of the approach for implementing human oversight for high-risk AI systems. These approaches include human-on-the-loop (HOTL), human-on-the-loop (HITL), and human-in-command (HIC) [198]. Each approach caters to different phases of the AI system life cycle and varies in

Category	Risk Impact Level	Provision of AI Act	Human oversight requirements
High-risk AI System	Critical/Significant	Robust human oversight and stringent compliance	Proportionate and effective control measures.
			Built-in decision intervention mechanisms
			Real-time monitoring and control capabilities
			Training and awareness framework
			Risk assessment and mitigation
			Documentation and logging of oversight activities
			Trustworthiness, ethics and legal compliance
Limited-risk AI System	Low risk	Ethical compliance	Transparency obligations
			Disclosure requirements
Minimal-risk AI System	No risk/Negligible risk	No regulation and human oversight requirement	Voluntary codes of conduct

Table 5: Human oversight requirement for various AI system categories

level of human involvement. The means for implementing control can take several forms, such as human-system interactions interfaces, decision support tools, alert systems, Periodic review protocol, etc. Control mechanisms implementation for oversight must be sufficiently robust to mitigate the inherent risk of the use of the AI system in any context to achieve oversight goals. This implies that the extent of implementation of control depends on the risk class of the AI system that is developed and commissioned for use.

(a) Human-on-the-loop: The human-on-the-loop (HOTL) approach for implementing human oversight consists of measures that enable active human supervision capabilities during the design and operation phase of AI systems [198]. It practically involves the integration of measures for active monitoring, supervision, and intervention during the design of high-risk AI systems. HOTL mechanisms are simply means for humans to supervise, observe, and monitor the systems using tools such as dashboards, metric trackers, and alert systems. These tools trigger intervention when predefined limits are reached, or operational rules are violated [87]. HOTL implementations have mainly been exhibited in the context of automation efficiency and human judgments. Some typical practical implementations include autopiloting of autonomous vehicles, where human drivers monitor the driving and context decisions of autonomous driving systems during the operation of autonomous vehicles so that drivers can take control in situations where the system encounters unfamiliar situations [123], vehicle steering and performance improvement [439, 204, 275, 437], fraud analyst monitor notifications form the fraud systems [124], custom officer supervises import duty system for exploration of triggers for systems to cope better with frauds from new importers [238], etc. It is essential to mention that the effectiveness of HOTL depends on the human handling system supervision and the functional capacity of the monitoring tools [335]. Several technical platforms and tools are increasingly being developed and utilized in industry for achieving HOTL oversight responsibilities. These HOTL enabling platforms facilitate oversight through real-time dashboards, alert systems,

and explainability layers. Platforms such as FiddlerAI, Arize AI, and WhyLabs enable human operators to monitor deployed models by tracking key metrics like drift, confidence scores, fairness indicators, and generating alerts for abnormal behavior. These systems support post-decision intervention, such as reviewing flagged transactions in financial fraud detection or supervising autonomous vehicle behavior using simulation environments like CARLA.

(b) Human-in-the-loop: The human-in-the-loop (HITL) approach for implementing human oversight engages humans at every decision point during the operation of an AI system. In other words, governance is attained by humans actively participating in the system decision cycle. This approach is particularly crucial in high-stakes domains where AI decisions can have significant consequences, such as healthcare diagnostics, judicial systems, and critical infrastructure management. The EU AI Act requires explicitly that HITL implementations in high-risk systems must include features that prevent AI decisions from taking effect before human review and validation [108]. In many HITL implementations, humans refine input to provide quality instances that AI models can learn from to improve their performance. Then, the decisions of the system using the model are subject to expert review and validation during operation. Many AI systems that are regulated with this approach are often deployed as decision support tools that offer recommendations or preliminary analyses that human experts validate before being adapted to use. For instance, in medical image analysis, AI systems are used to segment medical images or flag potential anomalies while domain experts like radiologists or doctors review and confirm the system result before they are applied to other medical procedures [73, 436], Pathologists utilize an AI-powered image retrieval system with interactive refinement tools, enabling them to dynamically guide the algorithm and improve diagnostic effectiveness and confidence in disease detection [82]. Similarly, a human expert is an analyst in a loop system for anomaly detection and management [3]. Other examples of human-in-the-loop anomaly detection cases include fraud detection, where fraud analysts examine all suspicious transactions flagged by the system to validate or reverse decisions of the system [124, 391], real-time human assessment of errors flagged by an AI system in manufacturing processes [406], etc. HITL approach simply ensures that human judgment remains central to critical decision-making processes while leveraging AI capabilities for enhanced accuracy and efficiency. In Human-in-the-Loop (HITL) systems, tools such as Label Studio, Prodigy, and Amazon SageMaker Ground Truth are widely used to support human annotation, validation, and correction of model outputs, particularly during training and evaluation stages. HITL is also integrated into decision-support workflows through platforms like RadiAnt in radiology and H2O.ai Driverless AI, which provide visual interfaces for expert review and confirmation.

(c) Human-in-command: The human-in-command (HIC) approach is the highest form of control for implementing human oversight. It seeks to address a broad

range of potential impacts, i.e., economic, social, legal, and other aspects of AI systems operations. It encompasses measures that empower humans with strategic authority and control to enforce ultimate discretion over the operation and deployment of AI systems [198]. HIC, unlike HOTL and HIC approaches, extends beyond operational consideration to include socio-technical aspects of the system, such as legal, economic, ethical, and societal considerations when determining the usage policies and appropriate context of deployment. Humans maintain strategic governance and ultimate discretion over the system to ensure that its usage is safe, reliable, and trustworthy. Some examples of the implementation of HIC include the management decision to decommission the use of the discriminatory AI-based hiring system in Amazon [118], reevaluation or withdrawal of the risk assessment system for prediction of chances of recidivism in judicial systems due to the reinforcement of racial bias in the historical criminal data [125], state reassessment of AI-driven grade prediction system in education as a result of unfair outcomes [220, 22, 46], cessation of the deployment of aggressive and bias content recommendation algorithm on social media platforms [43]. Enterprise-grade platforms like AI Fairness 360, IBM Watson OpenScale, and Microsoft Responsible AI Dashboard offer capabilities for documenting design decisions, risk management, conducting fairness and bias audits, and maintaining traceability for regulatory compliance. In high-risk domains, these platforms are often used in combination with internal governance protocols and policy review boards to enable multi-level accountability.

2.4.3. Human oversight and AI system lifecycle

AI development involves several iterative activities within a structured workflow designed to produce, refine, deploy, and adapt trained models to specific application contexts, (see the pipeline illustration in Figure 10). Each stage comprises different stakeholders who are interacting to collectively drive the AI development, establish and implement control measures, and oversee the entire AI development operations.

Human oversight mechanisms are control measures for ensuring ethical principles, legal compliance, and social accountability throughout the AI lifecycle. As shown in Figure 10, the AI lifecycle consists of five interconnected steps: data ingestion (Step 1), data preparation (Step 2), model training (Step 3), model deployment (Step 4), and prediction (Step 5). Each stage requires careful human involvement through various oversight mechanisms to mitigate risks and enforce alignment with ethical standards, regulatory demands, and organizational objectives.

Oversight is established following three primary approaches as illustrated in Table 6, human-on-the-loop (HOTL), where experts supervise automated processes and intervene when necessary; human-in-the-loop (HITL), where humans actively participate in decision-making processes; and human-in-command (HIC), where humans maintain ultimate authority over strategic directions and governance frame-

Control Mechanism	Machine Learning pipeline steps	Purpose	Key Component
Human-on-the-loop	Steps 1 & 2	Data collection, labeling, and data quality improvement for training models [490, 261, 185, 478, 441, 174, 271, 239, 29]	Visual interactive interface: Match scoring, Dashboard, User configurable settings, Context-based user interface, and Decision logging
	Step 3	Model training optimization and fine-tuning [185, 261, 256, 174, 453, 194, 446, 192]	Interactive user interface: Web-based interactive visualization, control panel, click-based user interface
		Safety and model inference enhancement [299, 4, 410]	VR-based interactive environment, User interface, Real-time alert system, motion-based signalling
	Steps 4 & 5	AI system safety, monitoring, efficiency, and collaboration [299, 319, 420, 196, 223, 258]	VR-based interactive environment, Interactive interface, Decision rule feedback, Graphical user interface, and Real-time feedback mechanism
Human-in-the-loop	Steps 1 & 2	Data annotation, labeling, preprocessing, preparation, and analytics [378, 479, 237, 80, 9, 257, 297, 291, 239]	Visual interactive interface: Web-based annotation interface, Mapping, Entity recognizer, AI-assisted annotation suggestions
	Step 3	Model selection and fine-tuning [174, 477, 190]	User control model selection
		Model validation and optimization [395, 329]	Interactive user interface
	Steps4 & 5	Model deployment and decision support [372, 367]	Visual interface
Human-in-command	1 & 2	Data quality enhancement [87, 173, 368]	Interactive interface, Documentation audit
	Step 3	Model training process transparency [301]	Documentation
	Steps 4 &5	Bias reduction and algorithm accountability [87, 353]	Interactive interface, Documentation, and Audit

Table 6: Overview of control mechanisms in AI lifecycle

works. These mechanisms can be embedded in AI systems via a range of control tools, such as interactive dashboards, audit trails, decision documentation platforms, and scenario-testing environments. The choice of oversight approach and its implementation depend on the risk profile of the application, the complexity of the decision context, and the stage of the lifecycle in which the AI system operates.

In Table 6, we provide a stage-by-stage analysis of how these oversight mechanisms are integrated across the AI life cycle to summarize human involvement in various AI development activities (process). Furthermore, it presents a structured comparison of human oversight approaches to integrating human involvement in the standard Machine Learning (ML) pipeline. Implementation of the approaches at each stage in the pipeline is further discussed in the paragraphs that follow:

(a) Human oversight in data ingestion stage: Data is a predominant element in the development of AI systems. Stage 1 is the data ingestion stage. The aim of this phase is to acquire a diverse volume of data to enrich the training and validation of the model from various sources like databases, sensors and external APIs. Human oversight in this phase focuses on data collection, aggregation, fusion, and data management activities. Experts supervise the acquisition of raw data from multiple sources, ensuring data quality, representativeness, and ethical considerations in data gathering processes. This initial oversight is crucial as the quality of ingested data

fundamentally determines the downstream capabilities and limitations of the AI system. While data ingestion establishes the foundation for AI system development through proper oversight of data ingestion operations, step 2 in the life cycle phase (Data preparation phase) builds upon this foundation with its own critical oversight mechanisms.

(b) Human oversight in data preparation stage: During Step 2 (data preparation), human oversight shifts to addressing data quality issues to prevent potential compromise in the performance of models. Human involvement in this operation helps to identify data bias, inaccuracy, incompleteness, and inconsistencies to improve the overall quality of data. Through visual interactive interfaces and annotation tools, human experts can identify and rectify potential biases, inconsistencies, or inaccuracies before data enters the model training. thereby establishing a foundation for fair and reliable AI systems. Oversight approach implementation differs throughout this phase. The human-on-the-loop approach mainly focuses on having various human experts supervise data preparation processes and activities within the data pipeline [490, 261, 185, 478, 441, 174, 271, 239, 29]. In the human-in-the-loop (HITL) approach, humans are more directly involved in the data process. They contribute expertise and contextual understanding to ensure that data are correctly interpreted according to the requirements of the use. For instance, experts can review complex road use images and provide bounding boxes on relevant objects in the image to help train object recognition models for autonomous vehicle [378, 479, 237, 80, 9, 257, 297, 291, 239]. For the human-in-command oversight approach, implementation concerns include ensuring compliance with data governance and other regulatory policies. Its implementation enforces ultimate authority over data decisions, ensuring compliance with regulations and organizational standards while maintaining transparency throughout the data lifecycle.

(c) Human oversight in model training stage: In stage 3, where the model is trained, several effort goes into configuring the model. This phase requires repetitive experimentation, tuning, and evaluation to build models. Human oversight becomes particularly technical, focusing on hyper-parameterization, parallelization strategies, algorithm selection, and performance validation. Several rounds of training, validation, and testing occur during this step of the pipeline. Through interactive visualization tools and control panels, experts can monitor training processes, adjust parameters, and ensure the model develops according to performance expectations while maintaining alignment with trustworthiness requirements. In this stage, the HOLT approach requires experts to enforce control measures that oversee the design of model operations [185, 261, 256, 174, 453, 194, 446, 192]. The HILT approach leverages the domain knowledge of technical experts by involving them in training optimization and fine-tuning processes relevant to algorithms and model architectures based on the model application domain and deployment context.

(d) Human oversight in model deployment and prediction stages: Steps 4

and 5 are the model deployment and prediction stages. The trained model is integrated into systems and applications for use in the operational domain. The deployment stage involves oversight mechanisms that facilitate the safe integration of trained models into operational environments. Using real-time alert systems and monitoring interfaces, human supervisors can observe model behavior in both back-end systems and end devices, enabling intervention when deployment challenges arise. Lastly, in Step 5 of the pipeline, human oversight concentrates on operational performance monitoring and ongoing evaluation. As the AI system makes predictions in real-world scenarios, experts utilize feedback mechanisms and graphical interfaces to assess output quality, identifying potential drift or unexpected behaviors that might require model updates or retraining.

2.4.4. Key components of human oversight

The AI Act establishes a tiered control mechanism framework that encompasses several critical components. These components are the medium for human intervention through which the control mechanisms can be implemented to achieve effective supervision and control of the operations of high-risk AI systems. They enable real-time monitoring and notification about potential operational risks during the operations of AI systems and operate to ensure that the autonomous operation of AI systems does not surpass human judgment and control. These components include technical interfaces for generating visualization and control, alert systems for prompting intervention, training programs, and decisive intervention capabilities for overriding systems. The technical interfaces are the primary points of interaction with the system, offering real-time system state information and actionable controls, while training frameworks ensure operators understand both system capabilities and limitations. Monitoring capabilities provide continuous assessment of system performance and behavior, enabling early detection of potential issues or anomalies. Decision intervention mechanisms complete this framework by providing operators with the tools to adjust parameters, override decisions, or implement emergency stops when necessary. Together, these means ensure AI systems remain aligned with human values and organizational objectives while maintaining safety and reliability in their operations

We further examined how these approaches translate into real-world practice by considering empirical cases to provide concrete evidence for demonstrating human oversight implementation across different domains. This helps to highlight how oversight configurations are adapted to specific technical and regulatory demands. Table 7 summarizes some applications across sectors such as healthcare, finance, manufacturing, mobility, criminal justice, and recruitment. Each case illustrates how context-sensitive combinations of human-in-the-loop (HITL), human-on-the-loop (HOTL), and human-in-command (HIC) are deployed to address life cycle-specific risks, from validating medical diagnoses and detecting financial fraud to ensuring fairness in hiring and accountability in judicial decisions. These

Domain / Application	Oversight Mechanism	Empirical Insight
Medicine [73]	Human-in-the-Loop (HITL)	Clinicians validate AI-generated anomalies; improves accuracy and trust.
Finance [124]	Human-on-the-Loop (HOTL)	Human analysts validate AI alerts; reduces false positives and detects new patterns.
Manufacturing [406]	Human-in-the-Loop (HITL)	Operators review flagged anomalies in production lines; enhances reliability and safety.
Autonomous Vehicles [123]	Human-on-the-Loop (HOTL)	Drivers supervise AI decisions and intervene; ensures safety in uncertain environments.
Judiciary [125]	Human-in-Command (HIC)	Judges review and sometimes override AI recommendations to prevent bias.
Human resources [118]	Human-in-Command (HIC)	HR managers retain authority over hiring decisions; oversight ensures fairness compliance.

Table 7: Empirical applications of human oversight across AI lifecycle phases

examples reinforce the need for a tailored and dynamic application of oversight mechanisms, guided not only by life cycle phase but also by domain-specific risk profiles, operational constraints, and societal impact. All together, these empirical insights demonstrate that effective human oversight is not monolithic, but rather an adaptive governance responsibility that must be integrated with AI system design, regulatory expectations, and real-world operational contexts for AI trustworthiness.

(a) Interactive interface: Interactive interfaces serve as the primary conduit for human-AI interaction across the AI life cycle phases. In data management, visual interactive interfaces facilitate critical tasks such as match scoring, dashboard monitoring, and user-configurable settings, enabling domain experts to assess and improve data quality [8, 26, 27, 37]. Web-based annotation interfaces, mapping tools, and entity recognizers support human-in-the-loop data annotation and labeling processes [5, 12, 36, 37], while documentation interfaces enable comprehensive auditing for human-in-command oversight [13, 25]. During model development, web-based interactive interfaces provide essential visualization capabilities, control panels, and click-based interaction mechanisms [26-29, 39, 43], allowing specialists to monitor training processes effectively. Similarly, user control model selection interfaces [26, 79] enable direct human involvement in algorithm choice, while documentation interfaces support transparency in the development process [51]. For testing and validation, VR-based interactive environments create immersive spaces for safety assessment [1, 7, 50], complemented by specialized user interfaces that facilitate comprehensive model validation [58, 67]. In deployment scenarios, interactive environments and interfaces extend beyond visualization to incorporate decision rule feedback mechanisms and graphical user interfaces [30, 41, 50], enabling real-time system monitoring and intervention when necessary. Across all life cycle phases, these interactive interfaces are designed with varying degrees of human involvement in mind, from occasional oversight in

human-on-the-loop approaches to continuous participation in human-in-the-loop systems and governance-focused interaction in human-in-command frameworks. The sophistication of these interfaces directly impacts the effectiveness of human oversight, making them fundamental components in ensuring AI systems remain under meaningful human control.

(b) Decision support: Decision support components provide the technological infrastructure necessary for human judgment to guide AI system behavior effectively. In human-on-the-loop approaches, these components include decision logging systems for data management [8, 26, 27, 37] and real-time alert systems during model testing [1, 7, 50], ensuring human experts can track decision rationales and respond promptly to potential issues. For human-in-the-loop frameworks, AI-assisted notation suggestions [5, 12, 36, 37] augment human decision-making during data annotation and labeling, while specialized decision support interfaces [62, 64] enable direct human involvement in deployment decisions. These components enhance human cognitive capabilities without replacing human judgment, creating a collaborative relationship between human expertise and AI capabilities. Human-in-command oversight relies heavily on comprehensive documentation and auditing tools [13, 25, 63] across life cycle phases, with specialized accountability mechanisms [13, 60] during deployment. These components ensure decisions are traceable, explainable, and aligned with governance requirements, maintaining ultimate human authority over AI systems. Decision support components are particularly critical during high-stakes operations, where they must balance providing sufficient information for informed human judgment with avoiding information overload that could impede effective decision-making. When properly implemented, these components create a complementary relationship between human and artificial intelligence, leveraging the strengths of each while mitigating their respective limitations.

(c) Other commonly used components: Beyond interactive interfaces and decision support systems, several other components play crucial roles in enabling effective human oversight. Context-based user interfaces [8, 26, 27, 37] adapt presentation and interaction mechanisms to specific operational situations, enhancing user understanding and efficiency across life cycle phases. Motion-based signaling systems [1, 7, 50] provide intuitive, potentially non-visual means of alerting operators to important events or anomalies during testing and deployment. Real-time feedback mechanisms [30, 41, 50, 53, 68] represent another essential component category, enabling continuous communication between AI systems and human operators. These mechanisms range from straightforward status updates to sophisticated performance analytics, providing operators with the information necessary for timely interventions when required. Entity recognizers and AI-assisted suggestions [5, 12, 36, 37] serve as augmentation tools that enhance human capabilities during data management tasks, while also preserving human judgment in final decisions. Documentation and auditing components [13, 25, 51, 60, 63]

Category	Trustworthy property	Description of the property
Category 1 (Legal requirements)	Privacy	Privacy reflects the ability of AI systems to safeguard user's information, respect the autonomy of users, and use information transparently at all times.
	Accountability	This property requires AI system to possess mechanisms that can enable audit and redress of adverse outcomes.
Category 2 (Ethical requirements)	Fairness	This attribute ensures that AI systems operate in a manner that is free from bias, inequality, discrimination, injustice, and abuse of rights.
	Human oversight	It entails the involvement of humans in the development, deployment, and usage of AI systems to ensure that human autonomy and values are established.
Category 3 (Technical requirements)	Accuracy	Accuracy relates to the ability of the AI system to use data or models to make correct judgements, classifications, predictions, recommendations, or decisions.
	Robustness	It is the ability of the AI system to function in a safe, secure, and reliable manner at all times in its context of use.
	Explainability	It is the ability of the AI system to explain the process involved in generating decisions in an understandable manner.
	Transparency	It is the ability of the AI system to provide information regarding its internal processes, capabilities, and purpose.

Table 8: Description of AI properties from regulatory frameworks

establish accountability frameworks that span multiple life cycle phases, ensuring transparency and traceability in AI system development and operation. These diverse components work in concert to create comprehensive oversight ecosystems tailored to specific AI applications, organizational contexts, and risk profiles. Their effectiveness depends not only on technical sophistication but also on thoughtful integration with human workflows, appropriate training programs, and organizational policies that clarify roles and responsibilities in human-AI collaboration. Together with interactive interfaces and decision support systems, these components form the technological foundation for implementing the tiered control mechanisms framework established by the AI Act.

2.5. Integration of human oversight and trustworthy requirements

AI systems are increasingly deployed across high-stakes sectors, including healthcare, finance, education, and the judiciary. When these systems fail or behave unexpectedly, they can be very detrimental to society at large. Consequently, the implementation of controls has become essential to ensure that their behavior during operation aligns with ethical and regulatory considerations. Human oversight is a fundamental control requirement that is vital to the trustworthiness of AI. It serves as a foundational requirement that enables the establishment and fulfillment of other trustworthy AI dimensions (properties or requirements), such as fairness, accuracy, robustness, transparency and explainability, privacy, and

Category	Trustworthy property	Description of the property
Category 1 (Legal requirements)	Privacy	**Privacy and fairness trade-off:** Sensitive identifiers may be required to make AI systems fair, which could increase exposure to privacy risks.
	Accountability	**Accountability and privacy trade-off:** Enhancing accountability necessitates collecting and monitoring data that can infringe privacy.
Category 2 (Ethical requirements)	Fairness	**Fairness and explainability trade-off:** Prioritising fairness constraints during model training can increase model complexity, making models less explainable.
	Human oversight	No trade-off.
Category 3 (Technical requirements)	Accuracy	**Accuracy and fairness trade-off:** Accuracy measured by performance can be compromised when attempting to maintain fairness in the AI system.
	Robustness	**Robustness and fairness trade-off:** Increasing robustness to specific adversarial examples or classes during training can cause disparity in treatment across classes. **Robustness and accuracy trade-off:** Adversarial training for robustness can compromise model performance on clean data.
	Explainability	**Explainability and robustness trade-off:** Generating explanations becomes significantly challenging as AI systems become highly robust and complex.
	Transparency	**Transparency and privacy trade-off:** Increasing transparency about how AI systems process data may inadvertently expose sensitive personal information.

Table 9: AI properties and associated trade-offs.

accountability. The involvement of humans in AI systems design, development, and deployment goes beyond the mere implementation of an additional layer of control; it is the mechanism for continuous improvement and collaboration for providing assurance regarding the design and development of AI systems. When oversight is adequately implemented, it potentially functions as a meta-requirement that reinforces and enhances other dimensions of trustworthiness. This is because it provides the basis for evaluating the technical implementation of individual requirements and continuously evaluating them for improvement towards achieving their implementation objective [231]. For instance, fairness requirements could be technically implemented; however, human experts are required to evaluate the contextual effectiveness of the implementation. So, when the system exhibits bias or discriminatory treatment, humans intervene to mitigate the biases and provide appropriate context to adapt the system. Similarly, when accuracy is considered, human validators can demystify complicated data input relevant to improving the learning process of the model, evaluate ground truth, and provide expertise that can improve the model's performance.

Table 8 summarizes the trustworthy properties (requirements) for AI systems, grouped into three categories, as described within the EU AI Act and related policy

instruments, while Table 9 complements it by presenting the trade-offs associated with these properties from existing literature. The categorization of these properties aligns with the proposal of the independent high-level expert group on AI (HLEG) for trustworthy AI [198]. Category 1 encompasses the legal properties that provide the legal considerations to be fulfilled during AI system design. This includes: Privacy and accountability properties. Category 2 addresses ethical considerations and includes fairness and human oversight. Our work considers Category 3, which relates to the technical properties that are required to make AI systems trustworthy. This includes robustness, explainability, and transparency. However, this survey separates human autonomy and oversight and considers human oversight as part of the technical properties because it governs all other properties.

In the following sub-sections, we explored how human oversight supports and enhances explainability, fairness, robustness, and privacy requirements throughout the life cycle of AI systems.

2.5.1. Transparency and explainability

The black-box nature of many AI systems is a central challenge for their trust and adoption in high-risk domains. The logic behind the operations of most complex AI models is opaque. Stakeholders often lack visibility into how specific inputs are processed to generate outputs, making it challenging to verify or validate the output. The common sentiment is that access to more information regarding internal operations of the underlying models of the technology can address the trust gap resulting from low confidence in AI operation and improve the safety of AI [240].

Transparency and explainability are related and complementary dimensions of trustworthy AI. However, they are conceptually distinct [347, 8, 34]. While transparency encompasses access to information about the AI model functionality, explainability focuses on deriving meaningful interpretations from the information for human understanding. Together, they provide they allow stakeholders to critically assess whether AI outcomes are valid and aligned with regulatory expectations. Human oversight is a critical enabler of the establishment of these trust dimensions. It operationalizes mechanisms, such as system monitoring protocols and structured procedural review, that support their fulfillment beyond mere declarations of compliance intents [23]. In addition, human oversight channels can be leveraged for demystification and easy presentation of the complexities of AI models' behavior to promote transparency and further enrich explanations provided by the system. We next examine transparency and explainability in line with governance and regulatory provisions.

(a) Transparency: is formally recognized as a legal obligation in major regulatory instruments, including the GDPR and the EU Ethical Guidelines [356, 198]. These frameworks mandate that autonomous system development and use must allow traceability and explainability. Thus, requiring that information about the system

be disclosed across the life cycle and made accessible to diverse stakeholders [304]. The specific content and depth of disclosure, however, vary depending on the stakeholder's role and level of interaction with the system. For instance, end-users interacting with in-vehicle facial expression recognition technologies desire to know and understand how their face and facial emotions are processed by the vehicle to adjust driving conditions to drivers' state for safe driving on the road [247, 41, 122, 373, 18].On the other hand, regulators require comprehensive disclosure and documentation about data governance, model functionality, and operational parameters to evaluate legal compliance and assess risk. For practitioners, the obligation to achieve transparency depends on the development context and can encompass multiple practices. For instance autonomous vehicle context, the practice may include explicit labeling of synthetic outputs across modalities such as video, image, text, and audio, as well as disclosure of information on data provenance and model behavior [6]. To support these activities, a variety of instruments are available in practice, such as *transparency-check questionnaires* [374], regulatory *frameworks* [135], and specialized *toolkits* [471, 397]. Importantly, the information disclosed through these mechanisms is not self-sufficient. It must be reviewed and validated by humans within the development process. This emphasizes the role of human oversight in ensuring that transparency obligations are meaningfully implemented and that disclosures are both accurate and actionable.

(b) Explainability: refers to the capacity of the AI system to generate the reasoning behind its decision and output to enable stakeholders to understand and interpret the internal processes of the system [399, 198]. While AI systems have been highly efficient in performing routine tasks and accomplishing complex undertakings that humans find challenging, the underlying logic that dictates input processing for output generation is vague. Consequently, regulatory demand for explainability confers on system developers and providers the responsibility to ensure traceability and understandability of algorithmic operations. In practice, this expectation ensures that affected stakeholders are duly informed of the existence of AI interactions, the rationale behind specific outcomes, and their rights in relation to the system's decisions [182, 313]. Which ultimately can build stakeholders' trust and confidence, especially the end-user, in the system outputs. Beyond compliance, explainability enhances AI systems. Studies have shown that it promotes interactivity by allowing stakeholders to query and engage with AI decisions [245, 190], enhance other trust dimensions like fairness and privacy awareness [254, 379], and support system evaluation. Explanations can be generated using a range of tools and techniques developed by researchers and AI practitioners in industry. Some techniques evaluate causality [279], others and enhance informativeness during system evaluation [214]. While some enable explanation of models across domains [360, 280, 85]. Collectively, these techniques demonstrate ongoing efforts to ascertain the explainability of AI systems. The following subsection examines how these objectives are operationalized through specific explainability methods and approaches.

(c) Explainability methods: Several approaches are applied in practice and research to derive insightful information from opaque AI models to make them understandable to various stakeholders. These approaches can be categorized based on some considerations such as *model complexity, scope of explanation, stage of explanation, and model knowledge* [8, 423]. It is important to mention that several explainability artificial intelligence (XAI) techniques would fit into one or more approaches and overlap.

Explainability techniques within the model complexity category focus on the relationship between the model complexity and explanations derivable from the model. This is based on the principle that model complexity is related to the ease of deriving explanations. For example, linear regression or decision trees present the relationships between input features and predictions that can be easily communicated. Thus, one way to derive explanations would be to rely on an inherently and intrinsically interpretable model [8]. Other inherently transparent models include k-nearest neighbors (KNN) and Rule-Based Learners. However, this approach presents a significant tradeoff in terms of model accuracy. Simple and easily interpretable models relatively exhibit less accuracy than complex models [370]. For instance, a random forest with hundreds of trees may achieve significantly higher performance than a single decision tree, but the random forest becomes large and more challenging to interpret. Alternatively, the more complex the model, like deep neural networks with millions of parameters, the more accurate but less explainable it is. Thus, a more sophisticated approach is adopted after training to derive an interpretation using external models to approximate the behavior of complex models.

The second category is based on the scope of explanation, which has to do with the level of detail of the interpretation derivable. Two primary methods can be discussed: global explanation and local explanation. Global explanation methods deal with explaining the entire behavior of the AI system, providing an understanding of the overall decision-making process of the model and feature importance across all instances. Prominent methods include permutation feature importance [70] and partial dependence plots [163], which demonstrate the impact of all features on predictions. Local explanation methods, conversely, deal with generating instance-level explanations regarding each specific inference of the AI system [115, 303]. For instance, when a loan application is rejected, local methods can explain precisely which factors in that particular application led to the rejection, providing transparency for each loan application decision.

The third category is based on the stage of implementing the explanation. Ante-hoc methods entail efforts taken before model training that are intended to make the model intrinsically transparent and understandable. These methods include designing self-explanatory algorithms like rule-based models, decision trees, etc., or incorporating interpretability constraints during training, such as regularization techniques [241, 202], and managing the depth of trees [384]. These ante-hoc approaches are commonly used for developing models that are inherently transparent

from their inception rather than attempting to explain black-box models after they have been trained [423]. By contrast, post-hoc methods are applied after model training to extract explanations from already trained models, which is especially useful for complex models like deep neural networks, where interpretability was not considered at the inception. This approach is the most adopted approach for generating explanations with XAI methods, and it utilizes diverse techniques, such as text explanations, visual representations, local interpretation methods, example-based clarifications, simplification techniques, and feature relevance analyses for presenting explanations to stakeholders [34].

The fourth category considered is based on model knowledge. Under this category, there are two approaches: model-agnostic and model-specific approaches. Model-agnostic methods are applicable to any model, regardless of the underlying architecture of the model. The methods require no access to the internal structure of the model but focus on understanding the relationship between input and output. This approach is flexible and can be applied across different types of models. Some examples of model-agnostic XAI techniques include LIME, an XAI method that approximates complex models with simpler and interpretable ones, and the SHAP method, which uses Shapley values to attribute feature importance. Model-specific methods, in contrast, consider the internal structure of the particular model types. These approaches leverage the unique characteristics of models to generate detailed explanations of the behavior of the model. For instance, an explanation of neural networks can be generated relying on techniques such as activation maximization, which highlights patterns that maximize neuron activations, or gradient-based attribution methods like Grad-CAM, which highlights the most influential regions in input data attributable to prediction.

(d) Human involvement in enhancing explanation: Human involvement in enhancing users' understanding of AI decisions cannot be overstated across the AI life cycle. Explanations emerge not only from algorithms but also through the interaction of stakeholders who interpret, validate, and communicate model behavior. Several stakeholders, technical and non-technical, interact during the development of AI for organizations. The technical stakeholders are the AI practitioners or model builders. They are mainly responsible for model design, development, testing, and integration of models into data infrastructure and products [205]. The non-technical stakeholders comprise the model breakers and model consumers. The model breakers are the domain experts, product managers, auditors, etc, who possess the knowledge to verify that the developed models sufficiently fulfill development requirements and objectives. The model consumers are the intended users that the models serve with information and decisions [205]. The interaction of the group of stakeholders provides complementary perspectives that make explanations meaningful and contextually relevant.

Generating meaningful interpretations and explanations of model behavior is a continuous process of interaction and engagement between several stakeholders to validate and refine the models throughout the lifespan of the model. These

interactions and collaborations enable them to continuously validate and improve the model, enhance confidence, and gain insights from model behavior and the trust of the users of the model. The responsibilities of these stakeholders vary depending on the context of the use of the AI system. AI practitioners and domain experts need to evaluate the stakeholders' needs for explanations and assess the risks (consequences) associated with AI decisions to determine appropriate human intervention and oversight approaches to deploy in different stages of the AI development process [205].

Table 10 provides an overview of different human oversight approaches and activities at different stages of AI development for enhancing model explanation and interpretation. Establishing human oversight in the explainability process encompasses deploying the oversight approach for implementing controls. The human-on-the-loop approach entails several practitioners supervising the development activities occurring in different stages. During data management, the intervention of model builders and domain experts enables the implementation of data preparation strategies [394], documentation [173, 301, 350], transparency and accountability [301, 350], interactive data inspection and exploration, and generation of insights from data before use through various activities, which their responsibilities encompass. This includes considering interpretability from the initial stages of model conceptualization, engineering features to extract meaningful features for training, collaborating with domain experts to ensure features are meaningful from a domain perspective, understanding root causes of problems surfaced by monitoring systems, risk management, and compliance, etc.

Human oversight approach	**Machine Learning pipeline steps**	**Human intervention and activities**	**Human role and intervention**
Human-on-the-loop	Steps 1& 2	Data preparation strategies and feature engineering [394]	Assist in preprocessing
		Documentation, transparency and accountability [173, 199, 301, 350, 33]	Ensure data integrity
		Interactive data inspection and exploration [394]	Inspect and validate
		Visualization of high-dimensional data [394]	Analyze data distributions
		Generating insight from data before use [394]	Extract key insights
	Step 3	Transparency [394]	Oversee model interpretability
	Steps 4 & 5	Monitoring and maintenance [195]	Track model performance
Human-in-the-loop	Steps 1 & 2	Data selection [32]	Curate datasets
		Representation [32]	Define data representations
		Visual exploration and interpretation [195]	Assess feature importance
	Steps 4 & 5	Gain insight into model behavior [32]	Interpret model decisions
		Model design, Parameter selection and tuning [32, 101]	Adjust model hyperparameters
		Documentation [301]	Maintain model records
	Steps 3	Model evaluation and refinement [32, 393]	Conduct validation tests
		Generating meaningful interpretation [393, 252, 360, 308]	Provide explainability feedback
	Steps 4 & 5	Monitoring strategies development [32]	Define monitoring policies
Human in command	Steps 4 & 5	Measuring trust and explanation [293, 289]	Validate and audit trustworthiness

Table 10: Human inputs towards enhancing explanation and interpretation of models

In the case of the human-in-the-loop approach, practitioners and model breakers actively participate in all the stages of the development process. During the data

phase, model builders define data representations [32] by selecting appropriate feature encodings and transformations that enhance interpretability. During model development, they participate in model design, parameter selection, tuning, and adjusting hyperparameters to balance performance with explainability [32, 101]. Furthermore, collaborations between the model breakers, more importantly, the domain experts and AI practitioners on review and interpretation of model performance and behavior provide insight into model behavior in order to interpret model decisions in context [195, 393, 252, 360, 308]. Create proper documentation for training efforts so as to maintain comprehensive details relating to choice-making in the model-building process, assumptions, and limitations. In the model testing phase, they are actively involved in model evaluation and refinement [32, 393]. They conduct validation tests to verify that models behave as expected and generate meaningful interpretations, providing explainability feedback that helps refine explanation methods. Other activities include testing the suitability of the models for the application context, comparing multiple models for progressive builds, fixing issues identified during model development, experimenting with explanation methods, verifying that the model meets expected behavior, providing feedback about areas needing improvement, addressing mismatches between human understanding and model behavior, etc.

The human-in-command approach places ultimate authority with human operators, which is particularly important in high-risk domains where accountability is paramount. During deployment, this approach focuses on measuring trust and explanation [293, 289], where humans validate and audit the trustworthiness of model explanations and document procedures such as feature engineering processes to ensure they meet regulatory requirements and ethical standards.

(e) Explanation format: The diversity of explanation methods, visual, textual, numerical, and feature importance, highlighted in the table, demonstrates the multifaceted nature of AI explainability. Visual analytics provide intuitive representations of model behavior, textual explanations offer narrative clarity, numeric metrics quantify confidence levels, and feature importance analyses reveal decision factors. This variety enables practitioners to tailor explanation approaches to different stakeholder needs and technical contexts.

Effective implementation of human oversight requires careful consideration of several factors, including stakeholder expertise, resource allocation, workflow integration, and continuous improvement. Different oversight approaches require varying levels of technical knowledge, and organizations must ensure that individuals involved possess appropriate expertise for their roles. Human oversight introduces resource requirements that must be balanced against operational constraints. Organizations should strategically allocate human attention to critical points in the AI life cycle. Human oversight mechanisms should be seamlessly integrated into existing workflows to avoid creating friction or bottlenecks in the development process. As AI systems evolve, human oversight approaches should adapt accordingly. Regular reassessment ensures that explainability mechanisms

remain effective as models and use cases change. Human oversight serves as a critical bridge between complex AI systems and the stakeholders who need to understand them.

(f) Human oversight mechanisms for enhancing explainability: Explanations of model behavior can not be fully realized through code implementations alone. They require the complementary involvement of human expertise to review, interpret, and contextualize technical outputs. Several mechanisms can be commissioned to leverage human input for enhancing explainability during model development. Some of these mechanisms are described below.

(f.1.1)Interpretable model reports: Human experts play a crucial role in creating and validating interpretable model reports that translate technical details into accessible explanations. These reports typically include feature importance analyses, counterfactual explanations, and decision boundary visualizations [52]. When prepared by cross-functional teams, including both technical experts and domain specialists, these reports effectively demystify AI decision-making processes for various stakeholders [205].

(f.1.2) Interactive explanation sessions: Human mediators often facilitate interactive sessions where AI developers explain system behavior to users or regulators. These sessions employ scenario-based demonstrations, what-if analyses, and real-time interrogation of the system to increase understanding [267]. The human presence in these sessions enables adaptive explanations tailored to the audience's technical background and specific concerns, significantly enhancing comprehension beyond what automated explanations can achieve [132].

(f.1.3) Decision review committees: For high-stakes AI applications, human oversight is often implemented through dedicated review committees. These cross-disciplinary teams evaluate contested or complex decisions, providing additional human judgment and explanation for stakeholders. Such committees are particularly valuable in sectors like healthcare and criminal justice, where understanding the rationale behind AI recommendations is essential for maintaining public trust.

By thoughtfully implementing these human oversight approaches throughout the AI life cycle, organizations can significantly enhance the explainability of their AI systems, fostering greater trust among stakeholders and enabling more responsible deployment of AI technologies.

2.5.2. Fairness

The pervasive deployment of AI technologies has raised ethical concerns about their bias and discriminatory tendencies. Unfair applications of AI technologies in domains such as recruitment, healthcare, judiciary, finance, and the criminal justice system can accentuate existing inequality and further marginalize groups when their decisions amplify societal bias and prejudices. Some examples of AI discrimination affecting society include gender discrimination and bias in hiring decisions [253, 36, 118], policing and profiling premised on biased decision,

discriminatory recommendations by recommender system [375], systemic bias in facial recognition system that is discriminative towards certain population and subgroups [352] and unfair prediction of crime tendency targeting a race group [208], unfair presentation and ranking of outcomes from search and advertising placement algorithms [208].

This growing number of documented AI bias and discrimination cases has become an ethical and societal concern, as the incidents are eroding the trust in AI technologies and simultaneously making AI more risky. Consequently, stakeholders now consider fairness attributes a critical requirement that must be systematically integrated throughout the entire life-cycle phase of AI systems. Several kinds of biases are inherent in the activities occurring in various stages of the AI life cycle, and each has the potential to make AI cause harm by discriminating directly or indirectly against individuals or groups based on one or more sensitive attributes that it considers [296]. Bias can creep into AI development through the input and learning algorithms and can be exhibited post-development when deployed into the environment. At the data management stages of AI development, different forms of ***data bias*** may occur during data collection, selection, and preparation operations. The occurrence may be due to unconscious errors or the nature of the data itself. Some data biases include measurement bias which occurs when data collection methods systematically distort the features and labels representing the construct of interest, leading to incorrect measurement, representation bias arises when the data samples are not truly representative of the population set as they do not accurately reflect nuances and diversity of the population [125], historical bias manifests when data exhibits or reinforces existing societal bias or stereotypes [409]. A common example is word embeddings showcasing societal bias about gender and profession, as it associates "nurse, homemaker" with "female gender" and "programmer, engineer" with "male gender", reinforcing existing bias at the time of data collection [66, 171].

After the prepared data is passed on for algorithms to learn the model, errors can also seep into the model development activities due to the way the algorithms draw associations, learn patterns, or treat data samples, causing biases to occur at the phase of the AI life-cycle. These kinds of biases are referred to as ***model bias***. For instance, learning bias occurs when the model training priorities(e.g., maximizing overall accuracy, minimizing specific types of prediction error, maximizing generalizability, etc) cause the trained model to consistently exhibit varying performance across different instances in the data such that the model significantly performs better on some data points than others [323]. Similarly, evaluation bias can occur in this phase during model evaluation, especially when the adopted benchmark data for evaluating a model does not represent the real-world use population, or the choice of evaluation metrics fails to disclose the poor performance of the model on certain subgroups. Popular examples include unequal facial recognition services based on race attribute [79, 312]. Model bias can also emanate as an aggregation bias when all groups or subgroups within a dataset are consistently modeled in

one way by assuming a similar relationship between inputs and labels across all data subsets, whereas every group or subgroup is peculiar [296]. An example of such an occurrence is when analyzing the text content on a social media platform of a community or group (e.g., gangs) with a general model. This can lead to misclassifications because slang, emojis, tags, or hashtag meanings can be peculiar to every group that a general model trained on all data from a specific platform would miss [342].

Bias also manifests during the deployment of a model. Deployment bias occurs when a model is deployed in the real world for a purpose for which it was not designed and developed [296]. For instance, a natural language processing model trained on academic text might be deployed to analyze social media conversations, leading to misinterpretations of colloquial language, slang, and cultural references [408].

Aside from datasets and algorithms-related biases, there are other types of bias and discrimination that cause unfairness to AI, see [296, 409] for details. This reality makes the conception of fairness in the AI context dependent and complex to ascribe a general description of fairness. The AI Act recognizes fairness as a crucial ethical principle for trustworthy AI that is connected to fundamental rights [108, 198]. It considers two interpretational dimensions for fairness and emphasizes that the development, deployment, and use of AI systems must be fair. The substantive dimension of fairness in the context of trustworthy AI aims for equitable and just outcomes by focusing on the distribution of benefits and costs, the avoidance of unfair bias and discrimination, equal opportunity, and respect for autonomy and proportionality, while the procedural dimension ensures the ability to contest AI system decisions through accountability and explicability.

(a) Role of human oversight according to regulation: Given the inherent limitations of purely technical approaches, human oversight emerges as a crucial mechanism in ensuring fairness in AI systems. Human intervention provides unique capabilities that complement algorithmic methods and address the nuances of fairness that algorithms alone cannot capture. The general objective of human oversight requirements is to minimize any risks that violate fundamental rights [138], which includes a tendency for discriminatory and unfair treatments or outcomes. The implementation of human oversight mechanisms enables experts to monitor anomalies and unexpected performance, which allows the detection of potential biases that could lead to unfair outcomes. In addition, the capability for humans to intervene by reviewing or overriding AI treatment of data instances and final output helps safeguard against potentially biased algorithmic decisions. Moreover, the Act stipulates obligations for providers and deployers to identify and mitigate harmful bias that can lead to discrimination [137]. This obligation complements other provisions for the deployment of fair AI systems as it enables further detection and correction of any residual biases potentially missed during the development phase.

To address fairness throughout the life-cycle of AI system, the Act recommends

some practices, such as i) removing identifiable and discriminatory bias during data collection; ii) evaluating datasets to understand inherent limitations before training operation; iii) implementing oversight processes to analyze and address the system's purpose, constraints, requirements, and decisions in a clear and transparent manner; iv) testing and monitoring bias, using technical tools to understand data, model and performance; v) Maintaining a diverse team of experts; vi) establishing a mechanism that allows flagging issues related to bias, discrimination, or poor performance; vii) Defining and measuring fairness using quantifiable metrics.

(b) Human intervention for enhancing fairness: In practice, human interventions towards achieving fairness demand having human experts at critical stages of the AI life cycle, from the point of data collection and annotation to model training, validation of output, and ongoing feedback. Experts are engaged throughout the development operation to monitor and enhance model learning, providing insights that incorporate fairness and ethical considerations. This ensures regular refinement and adaptation of models to information and changing contexts. Human interventions occur via different approaches to enable comprehensive oversight and compliance with regulatory requirements. At times, these interventions can take the form of periodic supervisory reviews of operations and implementation of corrections when needed. That form of intervention has human-in-the-loop (HITL). The main aim of HOTL is to ensure the inference generation and outcomes of AI systems are trustworthy at all times. A common use case of HOTL is in fraud systems where an analyst reviews transactions flagged by the AI system to determine the correctness of AI decisions and provide the system with feedback to refine decisions [221, 391]. In other situations, human experts are required to be actively involved in direct algorithm governance, audit, and the examination of development operations to identify and mitigate potential fairness concerns. This form of approach has human-in-the-loop (HITL). Experts are engaged in the system input generation and preparation activities to oversee data collection and annotation, select attributes, evaluate the performance and fairness of the model, etc. Depending on specific contexts and requirements, human intervention can also surpass expert capabilities. It can take the form of management assuming authoritative capacity over the entire life cycle and responsibility for operations so as to ensure that ethical standards and fairness are not compromised. [290]

(c) Processes for detecting discriminatory and bias: Discrimination is another major source of unfairness. The principle of fairness for the trustworthiness of AI systems includes the absence of discrimination [296], which connects discrimination to human prejudice and stereotyping based on sensitive attributes. Bias, on the other hand, is linked to unfairness that arises from data management activities such as collection, sampling, measurements, etc. Discrimination may be consciously or unconsciously exhibited, which then causes unfairness. When discrimination is intentional (conscious), differential treatment is directed towards an individual based on their protected attributes such as gender preference, race group, religious

Human oversight approach	Lifecycle phase	Human intervention
Human-on-the-loop	Data Management	Estblishment of data requirements and monitoring data related activities for identification of potential issues. [484]
	Model development	Monitoring abnormal and unexpected performance [484, 159, 35]
Human-in-the-loop	Data management	Evaluation of dataset for limitation [144, 284, 78]
	Model development	Feedback provenance for learning and guarding against errors [295, 129, 175, 159, 134, 487, 361]
	Model Testing	Algorithm assessment, evaluation and monitoring bias. [62, 400, 189, 188, 324]
Human-in-command	Deployment	Model audit and [434, 448]

Table 11: Various human oversight function for ensuring fairness

affiliation, etc. [296, 346], causing unfavourable outcomes for the individual. This particular form of direct treatment is referred to as ***disparate treatment***. Alternatively, when an unintended discriminatory treatment occurs against a protected group due to the implicit effects of protected attributes. This is considered to be indirect or unintentional discrimination and is referred to as ***disparate impact***. Generally, the processes for detecting discrimination and bias are fairness-enhancing mechanisms integrated into various stages of the AI development life cycle (see [231, 230, 296, 346]). Their implementation addresses unfairness in AI systems by targeting biases connected to input, algorithm, and outputs, and is systematically executed under the following approaches.

(c.1.1) Pre-processing approach: This approach caters to the pre-training phase of the AI life cycle. The aim of this approach during this phase is to mitigate bias and discriminatory patterns in training data before actual model training. Several data bias mitigation initiatives are deployed under this technique for transforming data to prevent unfair representations and ensure more equitable outcomes across different groups. Such initiatives include: data pre-processing, data sampling, and reweighting techniques to adjust the representation of different groups [78], and high-level feature transformation [295].

(c.1.2) In-processing approach: Bias can stem from the algorithm during the model training phase. The in-process approach addresses fairness during model training by refining the algorithm through the incorporation of constraints to enable the algorithm to cater for discrimination and bias [473, 474]. This approach is commonly demonstrated in most fairness research [295]

(c.1.3) Post-processing approach: This approach is deployed after the model training phase, especially when the model is trained without explicit fairness constraints. Irrespective of the underlying algorithm, it involves adjusting the output scores or decisions of a trained classifier to enhance its fairness. Some strategies adopted include threshold adjustment [112], learning separate classifiers [129], etc. Understanding the appropriate deployment and implementation of various

initiatives of these approaches is vital. Details can be found in [346].

2.5.3. Robustness

Robustness is the ability of an AI system and its components to function reliably and accurately under unexpected circumstances [265, 198]. Robustness consideration is crucial during AI system development. Robustness as a system attribute relates to the ability of a system to consistently maintain stability and reliable performance when experiencing unprecedented alteration in operating conditions, circumstances, or environment [69]. This denotes that a robust system is resilient and adaptive when situations demand, so as to preserve core functionalities, maintain operational integrity, and adapt to the change in situation. Unlike the traditional system, modern systems like AI systems possess additional capabilities that allow them to learn a model using data from their environment. Robustness in this context is the capability of AI systems to sustain operations and maintain correct inference in the face of deliberate efforts to undermine their operations or cause variations in expectations of their components. Thus, AI robustness encompasses the resilience of the AI system and any of its component units to threats such as perturbation in input data, malicious interactions, situational changes within the operational environment, and failures.

The rationale for robustness in AI systems and technologies stems from several reasons. Foremost is the fact that it is a requirement for their trustworthiness. The EU AI Act considers robust AI to be one of the cornerstones of trustworthy AI, and it is inextricably linked to the notion. Technical robustness is a prerequisite for making AI safe and trustworthy. As AI systems deployments have become pervasive across critical sectors like healthcare, transport, energy, education, etc., they must be resilient, highly reliable, and stable in their operations at all times to ensure safety and trust. This is because their failures can have consequential implications on societal and economic lives, infrastructures, and systems. For instance, a cyber attack on a power grid in Ukraine caused a massive power outage [128], an attack on an image recognition system used in healthcare diagnostics caused the system to give inaccurate diagnostics for patients [150], a Sensor failure caused Tesla autopilot to crash into public infrastructure and other road users, causing loss of property [123].

In addition, AI systems must be robust to manage attacks, errors, and circumstances that can compromise their safety and operational integrity. Attacks and threats like adversarial and evasion attacks that exploit AI models and cyberattacks that exploit the networking and communications infrastructures of AI [457] make AI unsafe. Ultimately, the robustness requirement is crucial during AI design and development for ensuring that AI systems can withstand these attacks and threats, with the aspiration to safeguard individuals and the use of critical infrastructures. By guaranteeing AI's operational integrity during such adverse situations, public confidence in AI systems could be further improved. Furthermore, robustness

enhances some other trustworthy AI, such as transparency, explainability, and privacy. Robust AI systems exhibit explainability by communicating explanations for system behavior at all times, even in the face of an attack, to enable comprehension of the state of the system. Also, it prevents data breaches and unauthorized access to the system, thereby enhancing privacy requirements [91].

(a) Adversarial attacks: AI systems have vulnerabilities that can be exploited by adversaries, making them susceptible to various types of attacks that can compromise their stability and reliability. The AI life cycle encompasses several activities that are executed sequentially in a pipeline for transforming vast amounts of data into models that are utilized in various application contexts. The series of activities flowing through the pipeline is a potential point of vulnerability that adversaries can exploit to harm the system. Mostly, data, the model, and the system itself are often the targets of adversarial attacks. Adversarial attacks scenario belongs to the AI and Machine Learning (ML) security domain, and it is better examined using a *threat model* [440, 458, 45, 92]. Threat modeling approach provides a comprehensive framework for understanding adversarial attacks from a system security and engineering perspective by systematically analyzing the attack surface, the attributes of the adversary (role, knowledge, capability, goal, and strategy), and attack vectors across the AI pipeline.

(a.1.1) Attack surface: The surface of attacks is the potential points of attack within the AI pipeline that an adversary can exploit to attack the system. It encompasses all workflows inside the pipeline, which include data collection, preprocessing, feature extraction, model training and testing, prediction, and model re-training.

(a.1.2) Attacker's goal: The motivations of the adversary to attack the system are usually to violate the system's security, which includes all three security concerns: system confidentiality, system integrity, and system availability.

(a.1.3) Attacker's knowledge: The knowledge of the adversary is defined by the level of confidential and sensitive information about the system that the adversary possesses about the targeted system. Generally, the attacker uses the level of knowledge at the attacker's disposal to plan and execute some kinds of attacks on the AI system. For instance, an adversary can leverage knowledge about datasets, feature sets, learning algorithms, model architecture, objective function, and training parameters [365] of the system to orchestrate a ***white-box attack*** when the adversary has a significant level of information *perfect knowledge*. Similarly, the adversary can leverage a limited amount of information at their disposal to orchestrate a ***grey-box attack***. Lastly, without any information, an adversary can orchestrate a ***black-box attack***.

(a.1.4) Attacker's capability: The capability of an adversary is the extent to which the adversary can access and manipulate input samples, training data, or observe the output of the trained model. The capability can be manifested in the ability to read, inject, modify, or logically corrupt input samples and training data [458]

(a.1.5) Attacker's strategy: The strategy of the attacker is the approach deployed by the adversary for carrying out the attack. This is determined by the amount of information available to the adversary about the target system.

(b) Adversarial attack vector: AI model can be easily hampered by induced and non-induced changes in any step of its construction [328, 392]. Induced changes are ***adversarial attacks***; they are deliberate manipulations of input with cleverly crafted perturbations to exploit the AI model so as to compromise inference ability or control the process. Non-induced changes occur due to situational events, e.g., environment, data quality, and failures of devices. Adversarial attack vectors comprise several attack methods that an adversary can explore at some specific phase within the AI system life cycle (pipeline). These attacks are discussed subsequently.

(b.1.1) Poisoning attack: Poisoning attacks are a significant issue as they target and contaminate the training process or data to harm the model. This flavour of attack exploits the AI systems during the training phase of the pipeline. The objective of the adversary is to violate the security of the system by compromising **integrity** of training data and model, and **availability** of inference [294, 59]. During training operations, the adversary can poison training data by injecting some malicious instances into the training set to cause misclassifications. Methods for inducing training data for attacking the model include label-flipping attacks, where the label of existing training data is altered for adverse purposes; data injection attacks, where perturbed data are injected directly into the training dataset; and backdoor and trojan attacks.

(b.1.2) Evasion attack: Evasion attacks target the inference phase of the model life cycle. They are the predominant threats in the threat vector against the AI system. They violate the security of the system by compromising the integrity of the inference capability using malicious examples. The success of the adversary strategy lies in the level of information that the adversary possesses about the system. Common example is the use of modification of network packets to bypass a network intrusion detection system [294]

(b.1.3) Model stealing: Model stealing (model extraction) attacks compromise model confidentiality during model inference. This attack targets models that are deployed as part of a service platform, e.g., Machine Learning as a Service (MLaaS) or through an Application Programming Interface (API). The adversary aims to craft queries that can exploit the model to acquire sensitive training information, which can be used to reproduce the same model.

(c) Robustness to adversarial attacks: Several techniques are used to enhance the robustness of AI models against adversarial attacks. Some of these techniques include:

(c.1.1) Data augmentation and pre-processing: AI systems largely rely on high-quality and heterogeneous data to train models for generalizability and robustness across diverse operational contexts. However, the scarcity of data inhibits this

purpose. Data augmentation and preprocessing techniques are fundamental strategies to address the scarcity of data and robust preparation of data for training robust models. Data augmentation encompasses the systematic transformation and generation of training instances with diverse coverage from existing data [42]. This helps to provide diverse and verse amounts of training data upon which the model can be trained to make the model resilient against both natural variations and adversarial manipulations. Data augmentation involves applying subtle geometric and color space transformation techniques on existing data, such as using methods like flipping, rotation, noise injection, cropping, and translation [51, 83, 93]. More sophisticated augmentation methods like CutMix and Mix [472, 332] combine different samples, while Generative Adversarial Network (GAN) methods generate useful synthetic samples for robust training. In addition, robust preprocessing pipelines that incorporate mechanisms that enable comprehensive feature engineering, scaling, normalization, outlier detection, and adversarial filtering can significantly mitigate vulnerability surfaces by standardizing inputs and eliminating potentially malicious patterns before they reach the model's core inference components [390].

(c.1.2) Adversarial training: This is the foremost approach towards enhancing AI model robustness against adversarial attacks. It seeks model robustness by training models with both original and adversarial training examples. By exposing the model to these malicious inputs during training, it learns to recognize and defend against such manipulations. Common adversarial attack methods, such as the Fast Gradient Sign Method (FGSM) [418], Universal Attack Approach (UAP) [305], Deepfool [306], and Projected Gradient Descent (PGD), are often used to generate these adversarial training examples. Advanced adversarial training techniques, such as TRADES, aim to find a better balance between robustness and accuracy on clean data.

(c.1.3) Gradient making for model robustness: Gradient masking enhances the robustness of the AI model by strategically modifying gradients of parameters such as input data, loss functions, or activation functions to thwart adversarial attacks [59]. This defence mechanism protects against various attacks, including L-BFGS and FGSM, by penalizing loss function gradients and minimizing loss over adversarial samples during model updates [285]. It also defends against C&W attacks through noise injection at the logit output level. Complementary gradient regularization techniques further enhance robustness by controlling variation in training data and enforcing Lipschitz continuity, creating smoother decision boundaries less vulnerable to manipulation. Recent research demonstrates that combining gradient masking with other defensive approaches like adversarial training provides more comprehensive protection, with adaptive methods emerging that balance robustness against performance by dynamically adjusting masking levels based on detected attack patterns.

(d) Human oversight and AI robustness: The process of making AI sufficiently

robust to adversarial attacks and bad actors cannot be attained without human involvement. While AI has demonstrated outstanding achievement in numerous tasks across different domains, absolute reliance on autonomous procedures for ensuring robustness is largely inadequate. More importantly, the consequential implications of AI exploits and ethical considerations for AI trustworthiness. AI learns patterns for real-world datasets. However, regardless of the amount of the dataset, it cannot absolutely reflect real-world complexities. Consequently, AI may not be able to effectively generalize to unfolding realities and can struggle to be resilient against the ambiguities and biases in human society. Human intervention mechanism integration is critical for addressing these critical limitations, especially in high-stakes domains. This involves incorporating human expertise, intuition, context understanding, and ethical judgment into the AI life-cycle so that it is more resilient to failures that could emanate from real-world complex scenarios.

Human oversight mechanisms can be implemented following the human-on-the-loop (HOTL), human-in-the-loop (HITL), and human-in-command (HIC) approaches. The HOTL approach is based on the notion that humans play a supervisory role and interaction is limited to need-based monitoring of system activities. The integration of the HITL approach requires humans to actively intervene in every system decision-making process. The HIC approach is a high-level control approach where human maintains a high level and absolute control over the AI system. This approach of integrating human judgment into AI processes and operations enhances AI system robustness and ensures that AI systems are responsibly deployed in any situation and domain.

Human oversight enhances the robustness of data operations. Experts monitor noise, anomalies, and data inconsistencies that undermine model reliability and stability during data collection and processing. They organize and implement comprehensive procedures for data collection and processing towards improving model robustness by applying strategies to manage the spectrum of data preparation and transformation issues, such as data missingness, outliers, data duplicity, data standardization and normalization, etc. In addition, domain experts analyze data samples to uncover errors, biases, and subtle complexities that automated preprocessing might miss. They review data samples to verify, validate, establish data requirements, refine labels and attributes to ensure that data is of high quality and relevant for the training and testing of robust models for deployment within AI system application contexts. Such intervention addresses data quality issues that could otherwise undermine model robustness.

2.5.4. Privacy

Privacy refers to the ability of the AI system to safeguard users' information, respect the autonomy of users, and use information transparently at all times. The efficiency of AI-based systems is fundamentally linked to the vast amount of data available for their operations. The daily rate of data explosion is exponentially high. The

estimation is put at 463*exabyte* every day in 2025 by the World Economic Forum [452], indicating the availability of diverse and enormous data repositories for AI-based systems to utilize. However, there are significant concerns, particularly regarding privacy. The volume of generation comes with the tendency of potential misuse and abuse of data, especially personal data. This also exacerbates the risk that several entities, like corporate organisations, the state, and individual actors, can exploit personal information. The scale of privacy breaches among corporations and public institutions has become alarming. For instance, Cambridge Analytica harvested the personal data of Facebook users in different countries inappropriately, which the company weaponized for influencing voters and election outcomes in those countries [212], a group of state-sponsored hackers took over Sony Entertain system and gained access to customers personal data, Capital One data breach incident [235], Targets lost customers cards information to hackers in 2013 data breach [322, 180]. Consequently, the scale of data breaches has prompted regulators to develop regulatory frameworks and legislation like the general data protection regulation (GDPR) [356] to govern the use of personal data and protect the rights of owners of personal data.

Privacy is among the required attributes for the trustworthiness of AI systems and technologies [198]. In the context of the trustworthiness of AI, privacy requirement is connected with data governance because several aspects of data governance and the use of data by AI systems have direct implications for user privacy management. The requirement entails the protection of the personal information of users throughout the entire life cycle of the AI system. This means that information provided by users must be protected from unauthorized access, use, and disclosure on the one hand. Similarly, the information generated about users during interaction with the AI system must also be protected. So, there must be no chance for personal information about individuals to be inferred from data and models at any time, even when it does not directly contain Personally Identifiable Information. This includes information that relates to the behavior, preferences, and decision-making patterns of individuals. Furthermore, considering that AI systems and technologies have the ability to infer sensitive personal information, the AI Act stipulates the procedure for ensuring privacy from data collection to deployment and interaction. This procedure includes establishing mechanisms to flag issues related to privacy or data protection in data collection and processing, assessing the type and scope of data, considering the use of less sensitive and personal data for AI system development, establishing measures to enhance privacy, and providing data owners with a feedback mechanism.

(a) Methods for enhancing data privacy: Data utilized for model training and testing undergo different processes from generation (collection) to destruction. After generation, they are transferred, stored, used, archived, and sometimes shared or published for legitimate purposes [95]. However, handling data to fulfill necessary obligations is not without some risks. Several techniques can be deployed to enhance privacy and address the risk associated with data privacy at

the different stages of the AI life cycle. Some of these methods manage the risk and enhance privacy in different ways. The approach includes data obfuscation and encryption techniques.

(a.1)Obfuscation techniques: Privacy-enhancing methods using this technique alter data to obscure sensitive attributes that can be used for identifying data subjects. This approach aims to achieve de-identification - the process of dissociating personally identifiable information (PII) of subject matter from the data subject to eliminate the possibility of re-identification of the subject in any form. PII could be direct or indirect. The direct PII are identifiers that can be directly used to identify or trace a data subject (person), e.g., date of birth, name, identification number, etc. While indirect PII are identifiers that can not be directly linked to a data subject unless complemented with other information, e.g., Zip code, sex, etc. [169]. The following are ways of obfuscating data to remove connections between identifiers and data subjects.

(a.1.1) Data anonymization: This technique replaces identifying information with artificial identifiers or pseudonyms, maintaining a separation between the identity of data subjects and the data itself. Unlike complete anonymization, pseudonymization allows for re-identification through additional information kept separately and securely. It serves as an important risk-reduction measure that complies with data protection regulations while preserving the analytical value of personal data [74, 288].

(a.1.2) Pseudonymisation: This involves replacing identifiers with artificial identifiers or pseudonyms, maintaining a separation between the identity of data subjects and the data itself. Unlike complete anonymization, pseudonymization allows for re-identification through additional information kept separately and securely. It serves as an important risk-reduction measure that complies with data protection regulations while preserving the analytical value of personal data [169].

(a.1.3) Data aggregation: This approach of data obfuscation employs statistical measures to aid data analysis. It combines data from multiple data subjects and then aggregates to derive group summary statistics, thereby obscuring individual-level information. By presenting information at a group level (e.g., averages, ranges, or frequencies), aggregation prevents the disclosure of information about specific individuals while still allowing for meaningful analysis of population-level patterns and trends [307]

(a.2) Encryption techniques: This privacy-enhancing technique employs mathematics and cryptography to control unauthorized access to data. Encryption of data involves employing algorithms for the transformation of data into an encoded format that can only be decrypted by using the appropriate decryption keys [462]. Modern encryption methods include symmetric encryption, where the same key is used for encryption and decryption, and asymmetric encryption, which uses public-private key pairs. For enhancing privacy, encryption can be applied at different phases within the data life cycle. It can be applied to data when it is

stored, when in transit (i.e., data is being transferred), and when in use (i.e., being processed). There are different types of data encryption for data privacy purposes.

(a.2.1) Identity-based encryption (IBE): This encryption approach leverages the identity information of the recipient (like an email address) directly as their public key, eliminating the need for certificate verification before encryption [67].

(a.2.1) Attribute-based encryption (ABE): This encryption approach leverages the attribute of users rather than their identities as the medium for providing fine-grained access control to data [366].

(a.2.3) Homomorphic encryption (HE): The encryption approach is more advanced than the other approaches. It is a computation-based technique for privacy risk mitigation. It allows computations on encrypted data without decryption, enabling privacy-preserving data analysis [462]

(a.3) Privacy preserving technique: Under this technique, sensitive data are directly processed at the source of the data to mitigate potential privacy risks inherent in data transfer. The methods used in the privacy-preserving technique do not require data collection in a centralized repository to utilize data. Rather, data is directly utilized from their various sources without compromising the privacy of the data providers. The approaches under privacy-preserving techniques for enhancing privacy are: differential privacy and federated learning. Differential privacy introduces carefully calibrated noise to outputs for enhancing privacy, while federated learning keeps data localized by bringing algorithms to the data instead of consolidating data in central repositories.

(a.3.1) Differential privacy: Differential privacy works by deliberately adding calibrated noise to query results, ensuring that the presence or absence of any single individual in the dataset does not significantly affect the output [130]. Privacy is measured with a metric, and the level of privacy protection is inversely related to the metric value. Differential privacy has been widely employed in different tasks [206, 480]. Organizations like the US Census Bureau, Apple, and Google have adopted it for collecting sensitive user data while preserving privacy. Local differential privacy, where noise is added to the user's device before data collection, provides even stronger guarantees by not requiring trust in a central data collector [228]. Recent advances include privacy budget management systems that track privacy loss across multiple queries to prevent excessive information leakage [1].

(a.3.2) Federated learning: Federated learning is a machine learning method for collaborative training models in a decentralized, distributed, and privacy-preserving fashion without the exchange of local training data. The learning approach addresses data privacy concerns using cryptography and differential privacy mechanisms. Unlike training in centralized settings, it protects personal data from being collected and processed during model learning. Rather, it allows for local training weights to be exchanged between centralized coordinating servers and all training nodes (parties) involved in the federation. Using the privacy mechanisms, models can be trained to learn from sensitive data without having to compromise any data

subjects [265, 327].

(b) Human oversight and privacy: Human interventions and monitoring activities contribute significantly to protecting personal information throughout the AI life cycle. Human oversight can ensure that AI systems adhere to the principle of data minimization by reviewing data collection and usage practices. Human review can verify that AI systems collect and utilize only the data that is strictly necessary for their intended purpose and that this is done with appropriate consent and transparency, aligning with established privacy policies and regulations. Human analysts can also play a crucial role in monitoring AI systems for potential data leakage and security vulnerabilities. By identifying anomalous activities or patterns, human experts can detect potential data breaches, unauthorized access attempts, or unintentional disclosures of sensitive information that automated systems might overlook.

Furthermore, human oversight is essential in ensuring the effectiveness of data anonymization and pseudonymization techniques applied to data used in AI training and operation. Human review can assess whether these techniques truly prevent the re-identification of individuals within the dataset, thereby safeguarding privacy. Auditing AI models for potential privacy risks is another critical function of human oversight. Human experts can employ various techniques, including simulating adversarial attacks, to identify potential privacy vulnerabilities within the model, such as the ability to infer sensitive information from model parameters or outputs.

Human oversight also provides a vital channel for addressing user concerns and grievances related to privacy. By establishing clear processes for users to report privacy concerns and ensuring that these concerns are addressed effectively and with empathy, organizations can build trust and demonstrate their commitment to protecting user privacy. Moreover, human oversight is indispensable for the implementation and ongoing monitoring of data protection measures within the context of AI. This includes ensuring that measures such as encryption, access controls, and data retention policies are not only implemented correctly but also remain effective over time in safeguarding personal data.

2.6. Challenges in human oversight for AI systems

In section 2.4, we have reviewed human oversight provisions of the EU AI Act and outlined the building blocks for implementing oversight and analyzed its connection to other trustworthiness attributes in section 2.5. The meaningful and effective implementation of human oversight in the development life cycle of trustworthy development AI faces several challenges. Some challenges are directly linked to the regulatory framework, and others are connected to vulnerabilities of AI. In this section, we examine key gaps in the regulatory framing, compare the trade-offs of common oversight mechanisms, and related challenges limiting the implementation of oversight.

(a) Operational ambiguity of framework: While the EU AI has provided the foundation framework for integrating human oversight mechanisms into high-risk systems, the implementation specifications stipulated in the Act are unclear, which makes the framework suffer from significant operational ambiguity that undermines its practical effectiveness. The operationalization of the broad oversight mechanisms, human-in-the-loop (HITL), human-on-the-loop (HOTL), and human-in-command (HIC), is vague because the necessary implementation specification is not stipulated. For instance, the regulation mandates capabilities for real-time human intervention in high-risk systems, but leaves critical questions unanswered: What constitutes sufficient intervention? How should alert and control interfaces be designed for optimal human response? What measures ensure operator readiness and accuracy under time constraints? Moreover, the Act inadequately addresses responsibility and liability distribution among developers, deployers, and users, particularly in complex autonomous environments. These substantial gaps between high-level principles and practical implementation guidance create compliance uncertainty and place an excessive interpretive burden on organizations implementing these systems.

(b) Complex trade-off: Establishing oversight mechanisms reveals complexities and trade-offs between human control and system autonomy, and operational efficiency. HITL approaches offer the highest level of human involvement and are typically adopted in high-stakes domains such as medical diagnostics or criminal justice. While they enable thorough validation of AI decisions, they also risk introducing latency, decision fatigue, or workflow disruption when applied to tasks that are time sensitive. HOTL mechanisms—employed in applications like autonomous vehicles or financial anomaly detection—strike a balance between automation and supervision by enabling post-hoc intervention. However, their effectiveness is constrained by the quality of monitoring tools and the operator's situational awareness. HIC mechanisms, on the other hand, are well-suited for strategic governance, policy setting, and life cycle risk management. Yet, they often lack immediacy, making them insufficient alone for real-time system oversight. These comparative limitations point to the need for adaptive, risk-sensitive oversight configurations that can combine mechanisms across system layers and life cycle stages. Unfortunately, such hybrid models are not currently addressed in regulatory provisions, leaving a significant implementation and auditing gap for practitioners and policymakers.

(c) Dynamic and tailored explanations: Explanations for users have garnered significant attention, with methods developed to effectively communicate the reasoning behind AI models' decisions. Various approaches have shown strong performance in extracting and generating explanations from model outputs. However, explanations are dynamic and context-dependent, often requiring adjustments based on the expertise of the human operator or end-user interacting with the model. For instance, explanations in critical scenarios versus explanations in more flexible

scenarios. As a result, creating a one-size-fits-all approach remains challenging. On the other hand, LLMs have demonstrated exceptional capabilities in interacting with users through natural language. Additionally, LLMs can clearly and easily explain the reasoning behind decisions at various levels of detail using advanced prompting techniques. This suggests that LLMs could be valuable tools for generating explanations. However, their inherent limitations and lack of trustworthy characteristics make it difficult to adapt them for explaining the inference processes of other AI models. It is conceivable, though, that once LLMs achieve a certain level of trustworthiness, they could be used to explain the inference processes of smaller, specialized AI models.

(d) Adversarial manipulation: AI models are vulnerable to adversarial attacks that exploit weaknesses at various stages of the machine learning pipeline, influencing the inference process and degrading reliability. This issue is further exacerbated by the rise of distributed learning paradigms such as federated and split learning, where decentralized structures introduce additional security risks. While explainable AI (XAI) methods aim to enhance transparency by revealing decision logic, these mechanisms themselves can be manipulated—e.g., scaffolding attacks, which obscure the true interactions between AI models and human users. As a result, human oversight becomes an extremely complex task, requiring continuous monitoring across multiple layers of AI decision-making. A potential solution is to shift oversight from individual monitoring to collective human consensus, where multiple experts validate AI behavior and execution. While this approach is feasible during AI development and life cycle management, it is difficult to implement in real-time for end-users, particularly in co-pilot applications that provide personalized, task-specific insights. The challenge, therefore, lies in balancing real-time oversight with practical usability, ensuring that AI-driven decisions remain transparent, trustworthy, and resistant to adversarial interference.

(e)Scalability and human cognitive load: As AI models achieve high performance and accuracy, they are increasingly integrated into a wide range of software applications. While this enhances user experience and efficiency, it also introduces a critical challenge: tracking and monitoring the inference capabilities and decision logic of numerous AI-driven applications becomes overwhelming. Although solutions such as AI dashboards and chain-of-thought explanations (e.g., LLM-generated insights) can improve transparency, their scalability remains uncertain. Requiring end-users to actively monitor, interpret, and validate multiple AI decisions across different applications creates a significant cognitive burden. The key challenge, therefore, is to develop AI monitoring mechanisms that are both scalable across applications and manageable for individual users. Without such solutions, users may struggle to trust, control, and effectively interact with AI, ultimately hindering widespread and responsible adoption.

(f) Real-time human interventions: The inference process of AI involves complex, high-volume computations, making it difficult for humans to follow or intervene

in real-time. Solutions like chain-of-thought descriptions can provide insights into how a decision is reached, but they do not necessarily enable timely human intervention. While users can review AI-generated reasoning, understanding the logic requires time and cognitive effort, delaying human feedback. As a result, intervention typically occurs only after the final decision is made, making it difficult to correct errors before they influence outcomes. This slows down AI adaptation and tuning, as multiple user validations may also be required to adjust (and verify) the inference process. The key challenge is to develop mechanisms that allow real-time human feedback and intervention, ensuring that AI can be corrected during inference to prevent undesirable or erroneous outcomes. Without such systems, AI models remain reactive rather than proactively aligned with human oversight, limiting their reliability and trustworthiness.

(g) Accountability gaps: The rapid adoption of AI co-pilots and advanced recommender systems is transforming decision-making and problem-solving in daily life. These systems are designed to support human reasoning, offering insights and recommendations that enhance efficiency and accuracy. However, a critical challenge emerges when individuals become overly influenced by AI-generated suggestions, leading to behavioral shifts that may be counterproductive or even harmful. When users rely too heavily on AI, they may lose critical thinking skills, make uninformed decisions, or pass responsibility onto AI systems. This raises concerns about accountability, as individuals may defer blame to AI-generated recommendations rather than taking ownership of their choices. Moreover, AI models—if not carefully designed—can subtly shape user behavior, potentially reinforcing biases or promoting actions that are legally or socially problematic. To mitigate these risks, AI models must be designed to provide neutral, non-coercive advice that does not unduly influence human decision-making. This is particularly crucial in domains where AI-generated recommendations have legal, ethical, or societal implications. Without proper safeguards, AI systems could become scapegoats for poor decisions, complicating issues of liability, ethics, and user autonomy. The challenge, therefore, is to develop AI that enhances human judgment without replacing or distorting it, ensuring that users remain responsible, accountable, and critically engaged in their decision-making processes.

(h) Regulatory uncertainty: Economic and regulatory bodies worldwide have expressed significant concerns regarding the deployment of AI, given its profound impact on both economic and societal development. In response, various countries, including the EU, USA, and China, have introduced acts and executive orders to address these challenges. However, the rapid acceleration of AI advancements, particularly with the release of LLMs, has exposed gaps in existing regulations, necessitating updates and revisions. While the EU AI Act primarily focuses on classical machine learning and deep learning methods to ensure accountability, transparency, and resilience, the rise of foundational models has prompted regulatory bodies to reconsider their approach, leading to new legislative considerations.

As AI technologies continue to evolve, including emerging mechanisms like genetic algorithms, there is an ongoing need to revisit and potentially rewrite existing regulations. This creates regulatory uncertainty regarding the use of AI in products. A key challenge, therefore, is ensuring that these regulations are flexible and forward-looking, capable of adapting to new AI developments with minimal disruption and ensuring effective oversight.

(i) Ethical dilemmas: As AI models become increasingly embedded in daily applications, determining whether a particular application truly requires AI becomes a challenging task. The integration of AI into applications necessitates careful fine-tuning of models to ensure that they operate in a fair, balanced, and neutral manner. This process includes not only optimizing the model's performance but also ensuring that it is adaptable to diverse user needs without bias. While human oversight plays a critical role in ensuring that AI models function as expected, a significant challenge arises in maintaining neutrality during the model tuning process. AI models should not favor any particular group, political interest, or ideology. The risk of unintentional bias creeping into these models can lead to unintended consequences, such as reinforcing existing inequalities or promoting controversial viewpoints. Therefore, a key challenge is ensuring that AI models are tuned and refined in a way that promotes fairness and neutrality, and human oversight must be vigilant to detect and address any potential biases that might emerge throughout the development and deployment stages. This requires both ethical considerations and technical solutions to balance model performance and fairness, which may not always align.

(j) Legal and technical trustworthiness: Defining regulatory trustworthiness in AI differs significantly from its practical implementation. Measuring and characterizing trustworthiness is an ongoing challenge as highlighted in our survey, with various methods developed to assess aspects like explainability (e.g., LIME, SHAP, Grad-CAM), fairness, resilience, and so on. Despite this progress, there is a clear gap between legal/ethical and technical requirements. The EU and US AI Acts outline trustworthiness requirements, while international initiatives like SHAPASH, PwC's AI Trust Index, Microsoft's AI Trust and Transparency, IBM's AI Fairness 360, and OpenAI's AI Impact Assessment have all contributed to defining trustworthiness. EU projects like TRUST-AI, SPATIAL, and TAILOR have also set principles for trustworthy AI development. However, a key challenge remains, which is identifying the essential requirements of trustworthiness. While the EU regulatory framework is expected to ensure AI trustworthiness, its solutions must be interoperable and adaptable to different legal and economic environments. Mapping legal/ethical requirements to technical standards is critical to understanding the limitations and implications of trustworthiness in practice.

(k) Sustainable AI: In addition to trustworthiness, a major requirement in AI systems lies in the significant energy consumption required by the underlying hardware during model training. As the demand for trustworthy AI grows, it is

crucial that AI models not only maintain their reliability but also minimize their computational and energy consumption. This requirement can be addressed in two main ways: either through innovative breakthroughs in computing technology, such as quantum computing, or by optimizing existing methods to maximize the efficiency of currently available hardware. Recent advancements in LLMs have already begun to mitigate the high computational costs and energy demands traditionally associated with these systems. For example, models like DeepSeek have been designed to run efficiently on constrained devices, demonstrating how optimization can reduce the environmental footprint of deploying AI models [53]. These advancements showcase the potential for improving the sustainability of AI systems while still maintaining their performance and scalability.

(l) Distillation and autonomous agents: Distillation methods have shown the potential to gradually improve the inference process of AI. This suggests that distillation can generate clear, rich, and comprehensive datasets for more accurate AI training. Simultaneously, autonomous agents capable of learning and interacting with each other are emerging, paving the way for autonomous training. This could significantly enhance AI inference capabilities in the near future, potentially leading to exponential growth in artificial intelligence. However, this scenario also implies that the need for human oversight may diminish, necessitating the implementation of additional safety mechanisms to prevent uncontrolled AI learning. In other words, while autonomous advances in learning may accelerate, they must be verified by humans to ensure AI trustworthiness. Furthermore, it remains unclear whether trustworthy AI can be inherently passed to distillation models. Despite these models being built from trustworthy AI systems, it is uncertain whether new distillation-based AI models are inherently trustworthy by default.

(m) AI regulations - a barrier to innovation or a key to competitiveness: AI regulations have subjected the development of AI models to rigorous scrutiny, requiring a more detailed examination of decision-making processes. Achieving trustworthiness in AI may necessitate a shift from conventional methods, creating new approaches that ensure thorough analysis of AI decisions. While this could offer a competitive advantage, as models developed with these mechanisms would be both accurate and trustworthy, in practice, the development of new methods to challenge existing ones may take significant time. Building effective AI systems has historically required years, if not decades, of research. This could potentially hinder AI innovation, as less restrictive approaches might already be capitalizing on AI's potential. From an economic perspective, countries that do not impose stringent AI regulations may gain a competitive edge, making it difficult for others to catch up. While EU regulations emphasize that they apply only to end-products, and not research and development, early adoption of AI technologies might become a barrier for economies striving to foster local dominance in AI solutions.

(n) Harmonization of governance frameworks and regulation: AI global governance and regulation landscape is becoming complex as many jurisdictions are

coming up with various principles, frameworks, and expectations. Consequently, coordinating operations and compliance across jurisdictions is a major challenge. In addition, the tendency of disparity in governance frameworks and non-alignment exists, which could discourage collaboration across countries and regions. A harmonized approach to AI governance that recognizes regional values and priorities can facilitate cross-border AI innovations, development, collaborations, and easy navigation of compliance operations for large corporations developing AI in different regions.

(o) Trustworthy properties measurement standardization: The standardization of methods and metrics for quantifying trustworthiness can formalize the measurements of the various properties of trustworthiness, as it is in trustworthy computing. This pursuit is a critical direction for advancing research in trustworthy AI because developing generalized metrics and evaluation methods provides the basis for comparative analysis of AI systems across diverse contexts. Such standardization could transform trustworthiness frameworks from mere compliance obligations into evidence-based assessment standards applicable to any AI system. Currently, methods for evaluating various trustworthiness properties are diverse, each with specific considerations and contexts dependent, making validation across domains challenging due to contextual misalignments. By standardizing these methods and measures, we can facilitate multi-dimensional validation of results and enable consistent interpretation of trustworthiness properties.

(p) Addressing paradoxical performance of AI systems: Artificial intelligent models are generally efficient with complex tasks. However, LLMs fumble on seemingly simple ones. This performance paradox presents a significant gap that has trust implications, as there is skepticism about the reliability of these models. Users are naturally perplexed about how LLM demonstrates proficiency in providing correct outcomes on complex questions and yet underperforms on basic reasoning questions (common sense questions) or factual accuracy. This inconsistency undermines confidence in LLMs and affects user trust. Addressing this paradox requires novel approaches that align model capabilities with human expectations, such that mastery of complex tasks should inherently encompass proficiency in simpler ones. This alignment is essential for developing AI systems that users can trust consistently across varying contexts and complexity levels.

(q) Unlearning AI systems for increased safety: AI systems acquire knowledge from diverse training datasets. These datasets often contain biases and inaccurate information that undermine the ethical and legal expectations of the AI system. Model unlearning enables the selective removal and discard of knowledge from trained systems to address wrong information in the knowledge base to improve the accuracy and ethical compliance of the system. This is an emerging frontier that addresses a spectrum of regulatory concerns relating to training data, from data quality to data obsolescence, sensitivity, etc. More importantly, it will enable capabilities to address aspects of AI trustworthiness like privacy and fairness,

facilitate correction, and support adaptation to evolving ethical standards. While machine unlearning can promote responsible development and AI safety, it must be considered by regulators as a relevant approach for AI governance during the development of AI models for it to be considered a valuable component of existing governance frameworks.

2.7. Summary and conclusions

This Chapter examined the critical role of human oversight in fostering the development and deployment of trustworthy AI systems. Drawing on diverse literature, regulatory frameworks, and industry practices, it provides a consolidated assessment of how human involvement enhances AI trustworthiness, with particular attention to the regulatory landscape shaped by the EU AI Act.

The survey contributes in three main ways. First, it systematically evaluates how different oversight mechanisms—human-in-the-loop (HITL), human-on-the-loop (HOTL), and human-in-command (HIC)—interact with key trustworthiness attributes such as transparency, robustness, fairness, and accountability across the AI lifecycle. Second, it develops a structured taxonomy that bridges the gap between regulatory provisions and technical practice by mapping oversight requirements to concrete implementation strategies, levels of human engagement, and specific phases of AI development. Third, it highlights persistent challenges in embedding oversight effectively and identifies research directions to align AI development more closely with governance expectations and societal needs.

Overall, the Chapter underscores both the diversity of existing mechanisms and the practical challenges of operationalizing them in real-world systems. These insights establish a conceptual foundation for the technical contributions that follow. In particular, they motivate the need for novel approaches that integrate oversight directly into AI pipelines. Building on this foundation, the next Chapter introduces Social-Aware Federated Learning (SAFL), which addresses one of the most critical phases of the pipeline, data collection and model training, by leveraging social dynamics and human participation to improve data quality and strengthen trust in collaborative learning environments.

3. SOCIAL-AWARE FEDERATED LEARNING: COLLABORATIVE DATA TRAINING WITH HUMAN-IN-THE-LOOP

We begin by focusing on the initial phase of the AI pipeline: data collection. This Chapter emphasizes improving data quality for AI model training. Training data forms the foundation upon which AI learning and inference capabilities are built. While various strategies can support training during system development, poor-quality data can undermine both the training process and overall model performance. The first contribution of this thesis addresses challenges related to data availability and quality in distributed machine learning, with a particular focus on federated learning. To this end, we propose Social-Aware Federated Learning (SAFL), a novel approach designed to enhance participation and trust in the data contribution process for AI training. In this Chapter, we detail the SAFL framework, describe its mechanisms for improving data quality and trust, and present evaluation results demonstrating its effectiveness.

3.1. Introduction

Federated learning (FL) has emerged as a potent mechanism for training powerful AI models in a decentralized and privacy-preserving way [265]. Federated learning is particularly powerful for emerging smartphone and smart device scenarios, such as healthcare or smart cities [463], as it enables individuals to take advantage of their devices to collect data and to train the model without needing to release data from the devices. Another benefit of federated learning is that it can take advantage of parallelization and decentralization to minimize the resource demands of individual devices.

Limitation of FL models: The performance of FL models is intrinsically linked with the availability of training contributions from individuals which in turn depends on the data they can collect. Indeed, if the data used for training is limited, the final model may suffer from poor generality as it fails to capture the true distribution of the data [349]. In the worst case, the model may even fail to converge if the data is too heterogeneous [364]. Given the heterogeneity of smart device ecosystems, the risk of failing to access sufficient amounts of the right data is significant as the devices may lack the right capabilities or may produce sub-standard contributions due to device limitations. Additionally, contributors may attempt to act as *free-riders* by refusing to spend resources on training the model. Instead, they try to benefit solely from the contributions of other users [160]. While many federated learning models demand a sufficient level of participation to training, even this approach is not sufficient as the free-riders may simply send random parameter updates, which can actually harm the overall model [286]. Ensuring sufficient quality for the federated learning model and overcoming these limitations

require new ways to boost the contributions of individuals while preserving the quality of the data. At the same time, the anonymous nature of the contributions can give malicious entities an opportunity to hamper the global AI model. Thus, an additional layer of social trust can improve the resilience of the system and help to overcome misuse.

Contributions: We contribute *social-aware federated learning*, the use of social connections to boost the training contributions in FL. These connections can either be known people, which implies a trusted relationship with the contributing person, or opportunistic contacts that are within the range of device-to-device connections [154]. Social-aware FL can simultaneously prevent situations where a client seeks to obtain benefits without contributing (i.e., free-rider problem), introduce trust in the training process through social connections [176], and mitigate the risk of malicious contributions. Our work harnesses social connections for fostering participation in the learning process, whereas previous works have been limited to using social connections to improve security and to avoid malicious attacks by determining with whom to share model updates or who to use as the aggregator [234]. In terms of improving the rate of FL contributions, the main alternatives for social-aware FL are to crowdsource the contributions and to rely on a centralized authority to coordinate model updates or to offer incentives to motivate individuals to contribute [469, 233]. We reflect on the state-of-the-art to identify key challenges and open issues, provide ways to overcome these challenges, and establish a research roadmap with the aim of acting as a catalyst for further research.

To understand the potential and the limitations of social-aware FL, we conduct an experiment with $N = 30$ participants to investigate the willingness of users to outsource tasks to other users. The results of our study show that individuals are interested in delegating tasks to others, and that the users are willing to execute the tasks for other users, provided that suitable incentive mechanisms are in place. Our work paves the way towards innovative social mechanisms for boosting contributions to training FL models and enables users to benefit from the FL model even when they lack the necessary capability to contribute to the model training themselves.

Summary of contributions:

- **Social-Aware Federated Learning (SAFL)** is a novel approach for model training in the Federated Learning paradigm that leverages social relationships for data selection and model training tasks.
- **Enhancing contribution diversity and overcoming client limitations** by enabling users to outsource tasks to their social connections, the system can gather data from a broader range of sources and devices. This helps to overcome the limitations of individual devices that may lack specific sensors or capabilities.
- **Proposes a roadmap** for future research in this direction .

3.2. Social-aware federated learning

Social contacts can support federated learning by enhancing data collection over time. We next discuss its feasibility

Model and assumptions: We consider a federated learning scenario where people with smartphones or other IoT devices collect data and use that to train a local model, which is then shared with an aggregator in exchange for a global model that can be used to improve services on the local device. We assume the FL application can access the social contacts of the participant's friends or other social contacts that have the same application installed. Each participant is expected to contribute a certain number of updates to the model, and a separate coordinator is responsible for requesting these updates. The coordinator is typically provided by a centralized authority, but it is also possible to choose the coordinator in a decentralized way using social voting. Each time the device sends valid updates, it receives compensation from an incentive mechanism used by the FL algorithm. Social-aware FL extends the basic FL by allowing the device to delegate the request to one of its social contacts. In this case, the device serves as the initiator and the social contact as a delegate. When tasks are delegated, the initiator is assumed to share all or part of the compensation they receive from the incentive mechanism with the delegate. Whenever a new user downloads the application, they are shared the current model parameters to ensure their training contributions are most useful for the current model.

Implementations: As shown in Figure 13, there are different ways to implement social-aware federated learning in practice, and the specifics depend on the overall implementation of the federated learning system. This also determines which information needs to be exchanged between devices. In a fully distributed case, the initiator needs to share their local model with the delegate, who then needs to load the model into memory and update it before sending the model parameters back to the initiator, which then sends them to the aggregator. Depending on the nature of the contracts, this type of system may require a separate reputation system to ensure the delegate gets compensated by the initiator. Alternatively, the initiator and delegate can establish a contract that is verified by the device(s) being responsible for aggregation, and the compensation can then be handled in line with this contract. In case a centralized aggregator exists, as is common in federated learning scenarios, the compensation scheme would be coordinated through the coordinator. The initiator can then either inform the delegate of the task descriptor and its own id or send the full model parameters to the delegate, as in the distributed case. The delegate can then send the updated parameters directly to the aggregator and, if needed, share them with the initiator.

Communications: The communication between the devices depends on the characteristics of the federated learning task. Most FL tasks correspond to *horizontal FL*, where each device shares the same feature space but has access to different samples. In this case, the model is trivial to share as all devices have exactly the

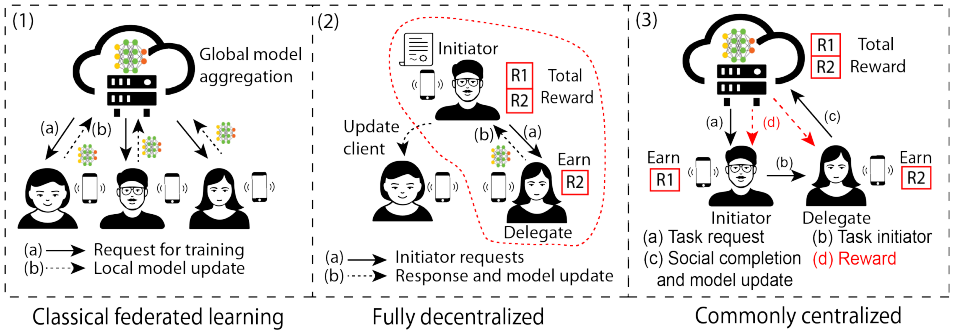


Figure 13: Design alternatives to extend federated learning with social-aware capabilities, (1) Classical federated learning, (2) Social-aware over a decentralized FL architecture, (3) Social-aware over a centralized FL architecture

same structure. In many smart device scenarios, including our experiments, devices have access to different sensors that need to be integrated to learn a common model. Thus, the feature space of the devices is different. This is known as *vertical FL*, which typically requires a different architecture [463]. One possibility is to rely on a hierarchical model where each sensor (type) has a separate convolutional structure, and a secondary convolutional structure maps the contributions of individual sensors into a unified format [465]. Finally, regardless of the implementation, the communications between the initiator and delegate should naturally be secured. This can be accomplished using a secure association mechanism to establish the communication channel and to encrypt the communications that take place. Social-aware FL assumes a prior trust relationship between the delegate and initiator, but the security of the mechanism can be further improved by integrating a mechanism on the initiator to detect possible malicious updates, e.g., by examining prediction performance before and after the update [98].

Peer-to-peer system: SAFL operations differ from traditional peer-to-peer (P2P) systems, even though it can also be implemented in a fully decentralized setup as illustrated in Figure 13(b). In P2P systems, collaboration is typically assumed to emerge naturally through direct communication between devices without centralized coordination, which can reduce overheads. However, it introduces significant challenges of security and privacy risks because interaction and exchanges may occur directly across potentially untrusted peers. In contrast, SAFL builds collaboration on top of social connections and incentives, thereby introducing an additional social layer absent in P2P systems. By leveraging existing trust relationships and reciprocity among participants, SAFL fosters willingness to share tasks by delegating to social connections when resources are insufficient, so as to ensure data contributions.

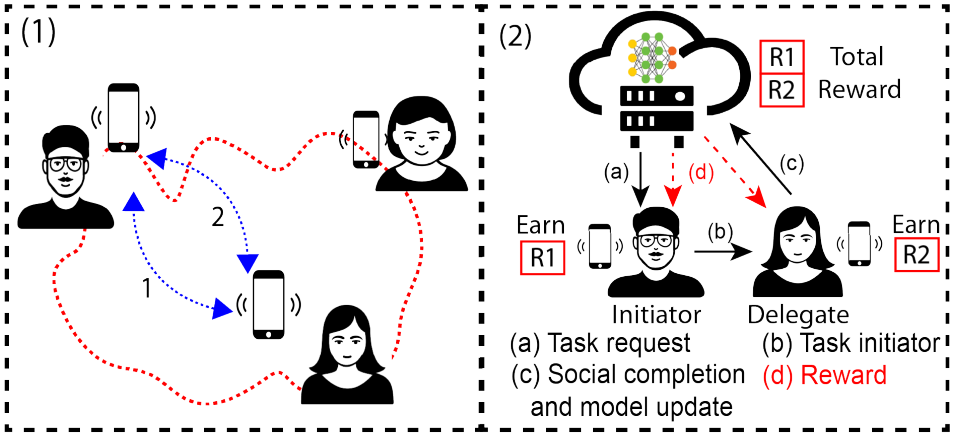


Figure 14: Comparing peer-to-peer (P2P) to socially aware federated learning, (1) Peer-to-peer (P2P) system, (2) Socially aware federated learning in a commonly centralized architecture

3.3. Experimental setup

We study the potential of social collaboration in federated learning by conducting an experiment that evaluates the users' perceptions of collaboration, their willingness to hand out or execute tasks, and the valuations that the users place on different tasks. We focus on tasks that involve training on diverse sensor measurements, as sensor data is an important source of data for emerging AI models, and as the data they provide is highly heterogeneous

Experimental design and methodology: Our study consists of two parts. The first part uses a Vickrey auction [339] to prime the perception of participants regarding the valuation of micro-task and establish a baseline for how individuals are highly likely to assign monetary worth to the micro-tasks. This auction procedure revealed interesting variation in valuation while minimizing strategic bidding, as participants were incentivized to reveal their true consideration for the micro-task rather than just hiking their bids. The second part is designed following a between-subjects methodology with two conditions: detached and attached. In the detached condition, the initiator hands out tasks to another user who is then given a fixed compensation (1€).This value was derived directly from the auction priming stage, where participants' bids converged toward modest, representative valuations for micro-tasks once unrealistic outliers had been excluded, see the bid outcome in 17(a). The attached condition is otherwise the same, but the initiator can freely choose to keep a fraction of the compensation, and the remainder is distributed to the person carrying out the task. In both conditions, the initiator is always the same, and the only difference is how the compensation from the incentive mechanism is shared with the social contact acting as a delegate. The monetary compensation is given once the execution of the task is completed. Tasks that are handed out to

others can be accepted or declined by the receiving party. If the task is rejected, it goes back to the initiator so that it can be handed out to others. Tasks that are accepted cannot be handed out again to avoid recurring tasks.

Application prototype: We implemented a mobile application for the study that allows the user to perform tasks or to distribute them to other users. The app is designed as a web application, implemented using the Adalo platform, and can be executed on any smartphone. The app uses notifications and alarms to make the participants aware of tasks received from other users. The app also integrates a database that captures the interactions of users with the tasks, e.g., rejecting a task, accepting a task, and task execution time.

Apparatus and task: We used three different smartphone models. Each smartphone is assigned to a specific task depending on the sensors it has: iPhone (HDR camera), Caterpillar CAT S61 (thermal camera), and Redmi Note 8 (air quality). We consider both generic tasks that can be performed on any device and specific tasks that can only be performed on devices having the appropriate sensor or other instrumentation. As a generic task, we consider GPS location, and as specific tasks, we consider high-resolution imaging, thermal imaging, and air quality monitoring, in line with the devices considered in the experiment. All tasks are listed on each smartphone, and participants use the application to complete them individually or by handing out the task to another participant. The nature of the tasks effectively corresponds to a vertical federated learning scenario [463] as the devices do not necessarily share the same feature space. We chose this type of scenario as it is representative of FL scenarios for smart devices, and as it is a scenario that benefits from collaboration.

Participants: We recruited 30 participants for the experiment. The participants were divided into groups of three to test social-aware collaboration. The participants are of different nationalities and were recruited through mailing lists and social media posts. As the study was designed as a between-subjects study, this means that 15 users (5 groups of 3) were allocated to both experimental conditions. To ensure the experiment involved social connections, we mentioned that the experiment was a group experiment that required three participants who share relationships to jointly test a social application, and the participants were encouraged to bring along another person they know with whom they would pair for the testing. Most participants came with friends, acquaintances, colleagues, or flatmates. When participants were unknown to each other, trust was ensured firstly by not using the personal device of the individual but rather one provided by the researcher, and secondly, by stating early during the experiment the reward obtained by participating, which has been used in other studies to encourage engagement in collaborative activities [65].

Procedure: The study procedure is summarized in Figure 15 and is split into two parts. Prior to starting the study, the recruited participants are assigned to a group. Each group is allocated to either one of the two experimental conditions (detached

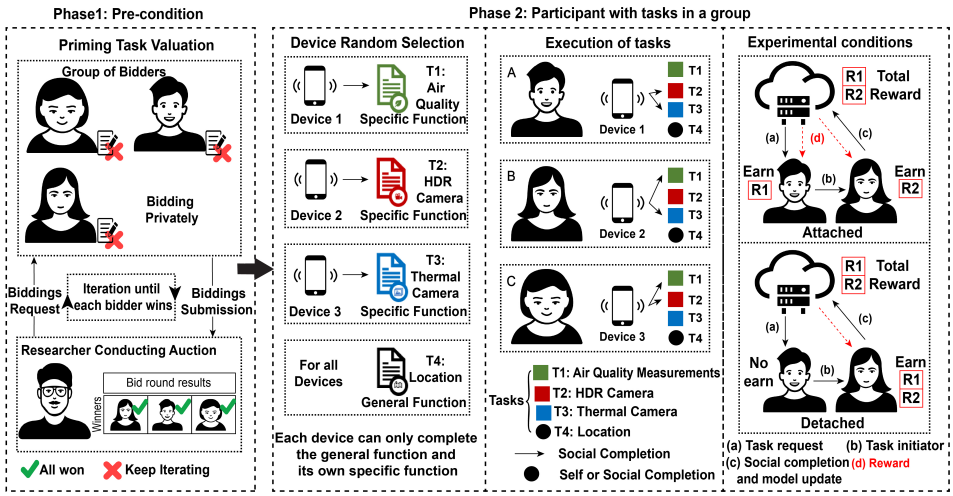


Figure 15: Overview of the experimental procedure (phase 1 and phase 2)

or attached) following a counterbalanced design. In the first part of the study (Phase-1), participants are pre-conditioned (or primed) to calibrate their valuations of different sensor data. This is done to ensure that people have a reasonable understanding of the costs and valuations associated with the different data, which forms the basis for deciding how to split the payments in the detached condition. The priming was achieved using a second-price Vickrey auction. We relied on this type of auction as it captures the most realistic perceptions of valuations from users over time [339]. After participants understood the auction type and signed a consent form, they were presented with a list of 20 distinct sensing tasks. Each sensing task focused on a different sensor to allow people to understand the potential of the task and to establish a valuation for tasks with diverse requirements. The sensors that were considered in the task were: camera, Bluetooth, microphone, GPS, humidity sensor, thermal camera, temperature sensor, touch sensor, and WiFi. Bids were elicited using questions that linked the sensor with a specific application. As an example, in one task, the participants were asked *"how much would you take to perform a task requiring the use of your phone's microphone to record a five-second sound clip to measure noise level in your current room"*. Participants then wrote their bids privately on a piece of paper. A researcher collected the bids and announced the winner. The first phase of the experiment concluded once every participant in the group had won the auction at least once.

The second part (Phase-2) introduces the participants to the mobile application in Figure 16. The researcher responsible for conducting the study explained the mobile application's functionality to the participants, who were given time to familiarize themselves with the application. Next, the three smartphones were distributed among the people in a group. Participants were then presented with tasks they needed to perform, and they were given 5 minutes to perform them. We presented four tasks, one for each of the four sensors (HDR camera, thermal

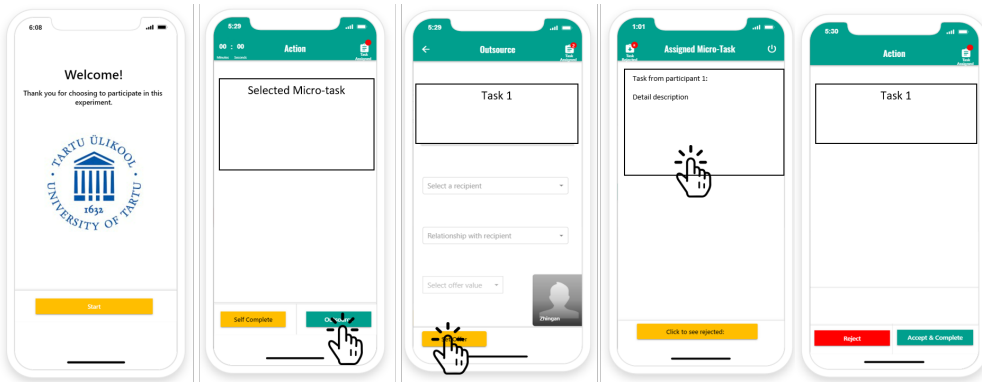


Figure 16: SAFL mobile application prototype

camera, air quality sensor, and GPS), to the participants and asked them to complete the task or to distribute it to another user. Given the differences in functionality and the design of the experiment, participants were able to carry out two tasks by themselves (GPS and the one task for which they had the right sensor on their device), and the two other tasks always required delegating the task to others. Once the experiment was finished, the smartphone was collected back and participants were compensated with a monetary reward for performing the tasks. To get the compensation, we stated at the beginning of the experiment that at least three tasks should be completed. If the participants did not finish the tasks, no compensation was given. The overall experiment lasted around 30 minutes. Additionally, a tea/coffee mug was given to each participant at the end of the experiment.

3.4. Results

3.4.1. Results of priming experiment

In the priming phase, in total 100 bidding rounds were executed and 297 bids were received. Figure 17(a) shows the distribution of the bid values. We applied a multivariate outlier detection (based on Mahalanobis distance) to remove bid values at both extremes (i.e., among the smallest and largest). Figure 17(a)-1 shows the overall distribution of the bids, and 17(a)-2 shows the resulting distribution after the outliers are removed. The results show that initially, many users placed higher bids, but rapidly recalibrated and started to accept lower valued bids as they were exposed to other users' bids.

We also separately assessed how privacy considerations factor into the users' valuations. Previous studies have shown that privacy implications of sensors affect users' perceptions [207] and thus we would expect to see these also reflected in the bids. However, other factors have also been shown to affect users, e.g., resource consumption is an important determinant. To isolate the effects of privacy, we chose three sensors that have similarly high resource consumption but different privacy implications: GPS (Personal), Camera (Public), and Microphone (Social).

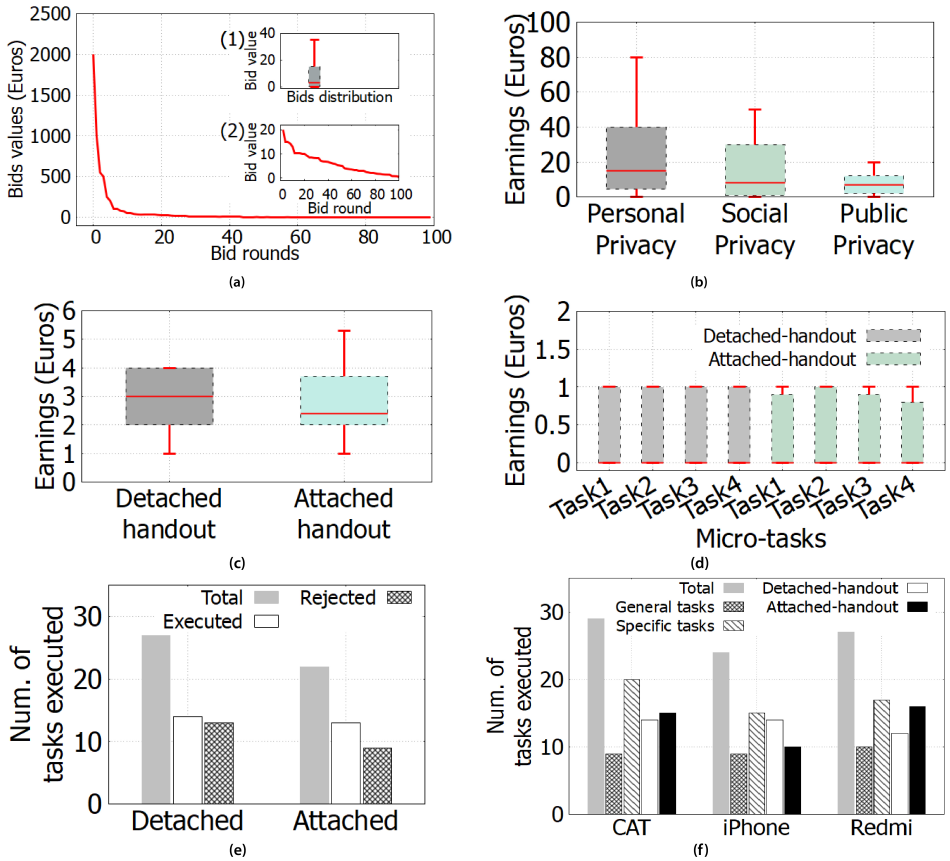


Figure 17: [a-b] Priming results of phase 1, a) Auction priming and bidding performed by participants, b) Quantifiable value of tasks based on sensor type and privacy data considerations. [c-f] Results of handout tasks using our prototype application in phase 2, c) Distribution of earnings in both conditions, d) Earnings obtained per task in the experiment, both conditions, e) Dissected actions of outsourced tasks, and f) Influence of device usage when performing tasks

Figure 17(b) shows the overall distribution of the task valuations for these sensors. The valuations reflect differences in privacy implications, which can be observed both from the mean valuations and the variance of the bids. The average valuations are 10.00€ for GPS, 5.00€ for camera, and 5.00€ for microphone. As the relative ordering reflects the differences in privacy, the results of the bids are in line with previous findings, suggesting that the valuations resulting from the priming experiment are realistic.

3.4.2. Results from application use

Earnings in the two conditions: We first assess the earnings of the participants across the two conditions: attached and detached. As the attached condition

resulted in the payment being split between the initiator and the task executor, we would expect the detached condition to result in a higher overall payment. The results also confirm this, but only show a marginal difference. Specifically, Figure 17(c) shows that the difference is merely 0.60€ between the two conditions (detached: mean = 3.00€, SD = 1.18, attached: mean = 2.40€, SD = 1.21). A Mann-Whitney U-test confirmed that no significance was found between the conditions ($U = 114, p > 0.05$). This result thus shows that the experimental design worked as intended and that splitting the payments resulted in a marginal loss of compensation. The payment differences were dependent on the role that the person was acting in.

We also compared the similarity of the payment distributions across the two conditions and roles (the values correspond to the test statistic of the Anderson-Darling test, which measures similarity of distributions). When the user acts as initiator, the payments are higher for self completion, suggesting that users are willing to carry out the tasks themselves, at least in exchange for compensation. No differences are found across the two conditions, suggesting that whether people can keep parts of the other user's compensation or otherwise does not affect the user's willingness to carry out the task themselves. In contrast, when the user acts as task executor, the distributions of payments depend heavily on the condition, with the user more likely to take advantage of social collaboration whenever they can keep some of the payment (detached: 13.10, attached: 4.52). Overall, the results thus show that the compensation mechanism has a desired impact on the payment structure, and that the payment structure affects how willing people are to take advantage of social collaboration, with the best results obtained when both the task initiator and task executor can be compensated for their effort.

Task completion: Figure 17(d) shows the earnings per task for the two conditions. The total earnings in the attached condition are smaller for most tasks, also for generic tasks that could be executed on any device, suggesting that the compensation structure also had some influence on the users' willingness to execute tasks. Figure 17(e), in turn, compares the total number of tasks that were outsourced and the number of tasks that were accepted for execution or rejected by the person receiving them. The detached condition resulted in a higher number of tasks being outsourced. Of these tasks, roughly the same number were accepted as in the attached condition, indicating that task executors were more likely to reject the task when they only received partial payment. The results thus show, on one hand, that giving compensation to the initiator is necessary to ensure as many tasks as possible are outsourced. On the other hand, the results show that the payment structure has to be carefully designed to motivate those receiving the tasks to accept and execute the tasks.

Device usage: Next, we analyze whether the device type influences the outsourcing of tasks in the two conditions. Figure 17(f) shows that for the generic tasks, not much differences can be observed. In contrast, for the specific tasks, a clear

difference can be observed depending on the task, device, and condition. On the CAT S61 and the Redmi smartphones, users were willing to execute more tasks than on the iPhone. This can potentially be explained by privacy considerations (see below), and the differences in task completions also support this view. Specifically, in line with the valuations in the priming experiment, tasks with higher impact on personal privacy were less likely to be executed (i.e., the generic GPS task) than tasks involving social or public privacy spheres. The differences may also result from perceptions of the devices and the data. For example, thermal images often appear less privacy intrusive than HDR images. As for the conditions, for the thermal imaging task (i.e., CAT S61), no differences could be observed. For the air quality task (i.e., Redmi), a higher number of tasks were executed in the attached than in the detached condition, whereas for the HDR imaging, the result was reversed. The differences in the HDR imaging task mirror the differences in the number of tasks that were outsourced in the two conditions, and thus, the difference is likely simply a result of a higher number of tasks being possible to execute. In contrast, the result for the air quality task suggests that people were likely more prone to rejecting air quality monitoring tasks that were outsourced with partial compensation.

Client participation and model accuracy: Finally, we considered the impact of participation on the training process by demonstrating how leveraging social connections for data contribution and collaborative task delegation increases the volume of usable samples and ultimately improves the overall accuracy of federated learning models. Figure 18 depicts how model accuracy improves as the amount of data each client contributes increases. The results clearly indicate that the number of participating clients directly affects model accuracy. As shown in the figure, increasing the number of clients (from 5 to 50) consistently raises the accuracy of the federated model. This highlights the importance of ensuring broad participation in socially-aware federated learning (SAFL) environments, where social ties and trust mechanisms can help recruit more participants. Moreover, the results suggest that larger sample contributions per client further enhance performance, reinforcing the need for SAFL to encourage not only wider engagement but also deeper individual contributions. Taken together with our earlier findings on incentives and task outsourcing, these results underline how socially-aware mechanisms can create the necessary conditions for achieving both sufficient scale and quality in federated learning.

Client participation and training efficiency: Figure 19 illustrates the effect of client participation on the number of training rounds required for convergence. The figure shows that when each client contributes a larger number of samples (e.g., 16 or 32), the model reaches convergence in significantly fewer rounds compared to scenarios where each client contributes fewer samples (e.g., 4 or 6). For instance, at 50 clients, a configuration with only 4 samples per client requires more than 80 rounds, whereas the same setup with 32 samples per client converges in fewer than

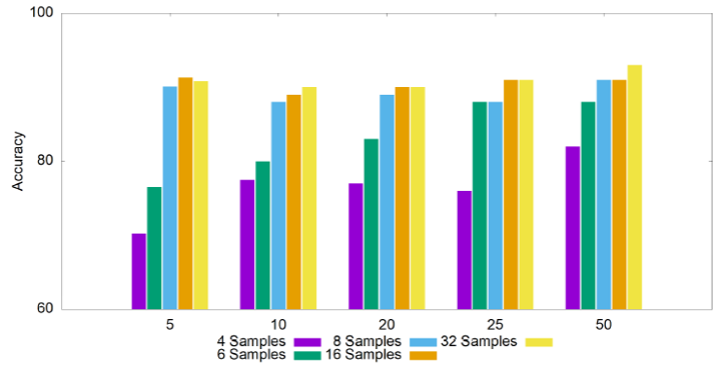


Figure 18: Impact of participation on training efficiency

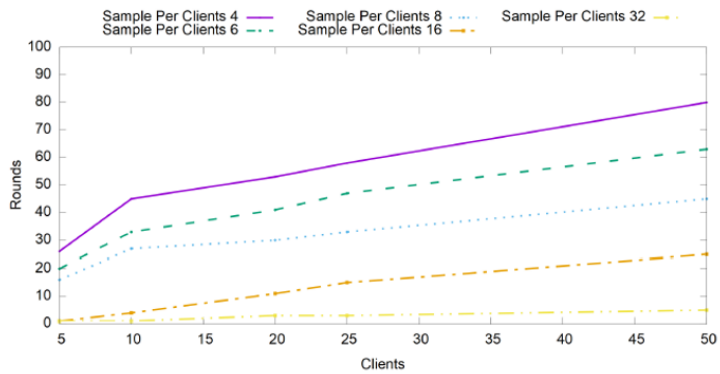


Figure 19: Impact of participation on model accuracy

10 rounds. This result underscores that both the breadth of participation (more clients) and the depth of contribution (more samples per client) jointly determine the efficiency and scalability of the federated training process. From a SAFL perspective, this provides empirical evidence that socially-aware mechanisms can improve not only model accuracy but also training efficiency, thereby making federated learning more sustainable in practice.

3.5. Challenges and opportunities

Collaborative compensation: Social links alone are not sufficient for supporting FL as people are prone to *churn*, i.e., their willingness to contribute wanes over time. Incentives are a potential way to overcome or at least mitigate this issue [469, 233]. Incentives for FL need to account for the complexity of the contributions as they affect the overall ecosystem in improving the *global model* instead of benefiting the initiator directly. The incentives should take these roles into consideration and potentially compensate both the person executing the task and the person serving as intermediary. At the same time, the compensation may be drawn from other

users of the FL system, as they all potentially benefit from the contribution to the model.

Data poisoning: Robust training of FL models requires multiple individuals to contribute aggregated data. This, however, can be exploited by malicious actors who exploit the system or compromise it through other forms of misuse. For example, so-called data poisoning can be used to hamper the AI inference process. Despite the several methods for detecting data poisoning [417] or other attacks, enforcing them with each model update is difficult. Incorporating an additional layer of trust based on social connections can reduce the possibility of aggregating poisoned updates to the global model. While social links are expected to increase the level of trust in the data providers, social links can also become a source of attacks when digital identities are stolen. For example, smartphones can have exploits (e.g., malware) without the users noticing them. These vulnerabilities can be used to poison the data that contributes to the global model or even the model itself. Overcoming this issue requires solutions that analyze the influence of individual contributions to the global model. For malicious actors, reputation mechanisms can offer a way to disqualify users who poison the data, e.g., by offering a way to rank the users based on the quality of their contributions.

Training moments: Ensuring high accuracy for an FL model requires multiple training rounds, at least until the model starts to converge. The processing time that is needed for these rounds can be significant and hamper the normal functionality of the device. As the key benefit of FL is avoiding data disclosure, this process cannot even be offloaded. Ensuring the training does not hamper the user's everyday activities requires a mechanism that allows suspending and later resuming the training on individual devices. Alternatively, methods that quantify the duration of time in which users can dedicate time and resources for a training task can be adopted, e.g., it is possible to quantify and predict the stability of a user's stay at a given location [154]. Probing times can also be considered when outsourcing an FL task to social connections to guarantee that tasks are not rejected by the delegate [224].

Recurring issues: There are also challenges that recur in all kinds of social-aware systems [364]. For example, besides participation and engagement, the distribution opportunities are governed by social contacts, which are limited. This may require solutions where further distribution is allowed, i.e., the social contact further propagates the request to one of his/her own contacts until the task can be executed. There are also recurring issues that affect the implementation of the FL model itself. For example, data can be highly heterogeneous, relying on sensors that only some devices have, or on multi-modal data that requires contributions from different sources. This requires the underlying FL model to be generic so that it can aggregate the different types of data. The data may also contain dependencies and come from different distributions (i.e., the data is non-I.I.D.), which requires separate mechanisms, such as using data augmentation or local tuning, to ensure

convergence [489]. Another recurring issue is privacy. Even if FL itself has been designed to be privacy preserving, there are attacks that can violate the user's privacy (e.g., data or model inversion). At the same time, users may not be fully aware of the privacy protection they are offered – especially as they may not be aware of what happens to the data once it has been collected. Improving the user's level of trust requires methods that foster transparency, e.g., by offering explanations of the inner functions of the FL pipeline and providing insights into the processing that is happening in the background [25]. Trustworthiness of FL models is difficult to achieve as a successful attack can be easily propagated to all the involved devices [417]. This then requires instrumenting each client and having them troubleshoot the model. Thus, further mechanisms and architectures that prevent attacks are also required.

3.6. Discussion

Stakeholders and adoption: Our experiments demonstrated that social mechanisms can improve the acceptance rate from the user's perception. Our method can be generalized not just to FL contexts, e.g., it can also be used to improve data collection in crowdsensing and crowdsourcing platforms. Our social mechanism to divide the compensation to execute a task (from finding the best suitor to task execution) can supplement existing platforms and provide new opportunities to augment the scope of data collection.

Room for improvement: We demonstrated how social connections can be harnessed to increase the rate of contributions in federated learning. This is particularly useful in scenarios where users lack a specific sensor or where they temporarily run low on resources. Our core focus was on exploring user perceptions; further work is needed to quantify the effects on the overall performance of the FL models. This would require running the experiment for several rounds, as the performance of federated learning depends on the number of samples that are provided and their capability to represent the overall distribution of data. When multiple clients provide data, a smaller number of rounds is usually sufficient, as the inclusion of data from different clients improves coverage of the overall data distribution. We are also interested in performing performance analysis focused on extracting system-oriented metrics such as the amount of data transfer, training payload, training time per device, and the number of rounds for model convergence.

Autonomous social agents: Social relations can be exploited by agents to automate the outsourcing of a task to social connections. Digital agents assigning tasks on behalf of a user can be useful to make optimal decisions for assigning tasks to social connections. It is also possible that the execution of a task can be performed solely by agents interacting through social connections. For instance, one agent can request another agent to measure the air quality of a room while the users are unaware of this interaction. Naturally, this can also open back doors to possible

cyber-attacks if the digital agent of a user is compromised. To mitigate this problem, blockchain and smart contract solutions can be adopted.

Multiple participation mechanism: SAFL engages multiple dimensions, social connections that build on trust, task delegation, and monetary incentives to encourage participation. This multiple integration of mechanisms increases the chances of sustainable client participation. Unlike Peer-to-peer (P2P) systems that remove reliance on a central coordinator, they often face severe participation bottlenecks, free-rider problems, and limited guarantees of contribution quality. SAFL addresses these shortcomings by embedding collaboration within existing social relationships, where trust and reciprocity naturally encourage clients to share tasks or contribute additional data.

Potential applications: Besides the large number of applications that can rely on FL support [463], it is also possible to envision new use cases and applications that require the social intervention of users to improve models. For instance, to speed up the convergence of models to accurately detect new illnesses, e.g., COVID-19, digital contact tracing applications can benefit from obtaining representative samples from infected individuals through social connections.

Threats to validity: The contributions of this work are subject to threats that may limit the validity and generalizability of the findings. A prominent threat is self-selection bias, which may have been introduced by the voluntary recruitment method. Since many participants were university students, some with related knowledge or technical backgrounds or a pre-existing interest in AI, their responses may not represent the broader perspectives of the general population. This limits the external validity and broader applicability of the SAFL framework's human feedback integration mechanisms. Furthermore, a threat to construct validity exists. The presentation and wording of the questions in the study's questionnaire could have imperfectly measured the intended constructs, potentially leading to an incomplete or inaccurate representation of participants' preferences. Finally, since the study has been conducted in controlled settings, it may not fully replicate the complexities of real-world AI deployment. We acknowledge the potential for a threat to ecological validity. Consequently, the results may not perfectly demonstrate the dynamics involved in performing human oversight in an AI development setting.

Client incentivization: Participation in federated learning is strongly influenced by the presence of appropriate incentive mechanisms, which can be both financial and non-financial [422]. While monetary rewards provide direct motivation, non-financial incentives, such as recognition, reciprocity, reputation, and gamification, can be leveraged for motivating client participation and contribution of data [102]. Social recognition can encourage clients to contribute actively, with tangible or symbolic rewards, such as digital badges or certificates, which can motivate clients. Reputation or credit points, though lacking direct monetary value, can serve as markers of reliability and trustworthiness. These reputation scores can influence

client selection for FL tasks, ensuring that clients with higher reputations are prioritized, thereby reinforcing trustworthy contributions and discouraging free-riding behavior.

3.7. Summary and conclusion

In this contribution, we introduced social-aware federated learning, a novel approach designed to enhance training contributions in federated learning scenarios, particularly in situations where individual contributors may lack the necessary resources or capabilities. Our approach leverages social connections to enable the outsourcing of data collection tasks, addressing limitations in data availability and promoting increased participation. We identified key challenges and opportunities associated with social-aware FL, including collaborative compensation, data poisoning, and the impact of training demands on user devices. To evaluate the feasibility and effectiveness of our approach, we conducted a user study with 30 participants, comparing a detached and an attached condition to analyze the influence of incentive structures on user behavior. The results of our study demonstrate the potential of social collaboration to improve user engagement and increase the rate of task completion in federated learning. We observed that providing compensation to both task initiators and executors is crucial for maximizing participation and ensuring efficient task distribution and completion. Furthermore, our findings highlight the importance of carefully designing incentive mechanisms to balance the motivation of all participants involved.

4. THE SPATIAL ARCHITECTURE: DESIGN AND DEVELOPMENT EXPERIENCES FROM GAUGING AND MONITORING THE AI INFERENCE CAPABILITIES OF MODERN APPLICATIONS

In the previous Chapter, we introduced a mechanism to improve data collection. However, AI models deployed in systems and applications also require continuous monitoring, both during inference and when the models are updated. Since AI models are often black boxes, their use raises significant safety and trust concerns within applications. To address these challenges, we focus on a solution for assessing and monitoring the trustworthiness of AI systems at runtime. This Chapter presents the SPATIAL architecture, our second contribution of this thesis. SPATIAL is a novel system designed to augment modern applications with the capability to gauge and monitor the trustworthiness of AI inference through a human-in-the-loop mechanism, enabling human oversight and control over AI development processes. In this Chapter, we detail the design and implementation of the SPATIAL architecture, describe its key components and functionalities, and present evaluation results demonstrating its effectiveness in enhancing AI trustworthiness.

4.1. Introduction

The adoption of AI is imminent in everyday applications. The AI market value is expected to reach a valuation of two trillion USD by 2030 [404], emphasizing the impact of AI on current software practices and systems development. Machine and deep learning components (aka AI components) are part of larger systems that provide autonomous decision capabilities for modern applications. AI components implement machine/deep learning pipelines to build AI models. These models are improving the perception, experience, and interaction between users and digital applications [75], providing human-like and insightful functionality that facilitates application usage and provides added value to users. Examples of this include advanced Chat-bots (ChatGPT, Gemini, Ernie) for e-commerce recommendations [351], optimal route planning for practical drone delivery [161], and sophisticated diagnosis capabilities in healthcare applications [158], to mention some.

Limitations of AI adoption: A key limitation for the adoption of AI at scale is its inherent black-box characteristics [8]. Indeed, AI incomprehensible advanced performance caused distrust to humans when massively trained, leading to the release of an open global petition in March 2023 to slow down the development of AI for at least 6 months [268]. The probabilistic nature of AI decision-making cannot be dissected using existing methods to verify software [38]. Besides this, AI models can be easily hampered throughout their entire life cycle, making them

vulnerable and exposed to many threats, impacting their autonomous decisions. This is worrisome in cybersecurity situations, where AI models can be (purposely) attacked to perturb their inference process, which can cause life-critical consequences for people and society.

As recognized by all economic and regulatory frameworks, with a primary emphasis on the EU but also encompassing the US and China, artificial intelligence (AI) stands out as the pivotal focus for developing a trustworthy technology. Traditionally, trustworthy computing ensures that a piece of software is trustworthy to users by verifying several of its properties, e.g., robustness, reliability, resilience, accuracy, and so on. Audit and accountability compliance on trustworthy software is simpler as there is a quantifiable understanding of its performance sensitivity to drifts and errors. As trustworthy verification cannot be conducted directly with AI using traditional methods, there is a lack of transparency, accountability, and resilience towards AI technologies.

This has led Europe to impose strict regulations for the use of AI, becoming a benchmark at an international level. AI trustworthiness extends fundamental principles of trustworthy computing with additional properties that have been considered and some defined by regulatory entities (EU AI Act and US Executive Order 13960). Trustworthy AI is valid, reliable, safe, fair, free of biases, secure, robust, resilient, privacy-preserving, accountable, transparent, explainable, and interpretable [260]. Notice, however, that AI trustworthiness is an ongoing process whose definition is evolving continuously and involves collaboration among technologists, developers, scientists, policymakers, ethicists, and other stakeholders.

As emerging regulatory standards mandate increased human control and oversight of AI, this concurrently reshapes the development practices and responsibilities of individuals engaging with AI. Moreover, new methods and approaches that help to understand the behavior of AI are being investigated or have re-gained attention, e.g., Explainable AI (XAI) methods [166]. As applications equipped with AI continue to proliferate in every aspect of human life, new methods are required to gauge, adjust, and monitor the trustworthiness of AI inference capabilities.

Our contributions: We contribute SPATIAL, a proof-of-concept architecture that augments modern applications with capabilities to robustly gauge and monitor the trustworthiness of AI in a human-in-the-loop manner. To achieve this, SPATIAL uses an AI dashboard and instrument applications with AI sensors. Conceptually, an AI dashboard serves as a tool to provide insights to human operators, enabling them to monitor and adjust AI trustworthiness according to their preferences. Additionally, it facilitates the verification of AI systems for potential audits and ensures compliance with accountability regulations set by regulatory bodies. In parallel to this, AI sensors that monitor specific trustworthy properties are instrumented within applications. Simply put, an AI dashboard shows to users quantifiable metrics extracted by AI sensors [152].

To design SPATIAL, first, we investigate the sensitivity of machine learning pipelines to (induced/non-induced) changes, from input data to model deployment.

With this information, trustworthy metrics that can be instrumented as AI sensors are reviewed in the current state-of-the-art. For instance, a sensor for fairness can be instrumented to analyze raw input data as well as to characterize fairness in decision making after model deployment [114]. Notice that currently, there is a misalignment between regulatory (legal) and technical trustworthy requirements. Thus, relevant metrics are selected from a technical point of view. Naturally, as regulatory trustworthiness evolves, it is possible to replace technical metrics with alternatives that better adjust to legal requirements. To augment modern applications with AI dashboards and sensors, we develop SPATIAL following a micro-service pattern. The key idea of using this pattern is that each micro-service contributes with the specific functionality to monitor a trustworthy property, and this functionality is requested by an AI sensor instrumented in the application (like an API). This also provides flexibility to add more metrics dynamically.

Besides this, the pattern also helps analyze a specific set of trustworthy properties. Indeed, as demonstrated by previous work, trustworthy properties are not agnostic. Thus, the number of trustworthy properties that can be derived from an application depends on its inherent characteristics [260, 449]. Through rigorous analysis and benchmarks conducted in real industrial use cases, we evaluate the performance and scalability of SPATIAL. Our results indicate that to measure trustworthiness in AI, it is necessary to instrument every step of the AI pipelines with sensors. Moreover, our results also suggest that AI dashboards and sensors are useful to individuals to monitor AI inference capabilities, but they increase the complexity of developing and maintaining AI components in modern applications. Our work also highlights lessons learned from designing and developing SPATIAL and describes ongoing challenges that require attention to achieve a robust analysis of AI trustworthiness and greater engagement of human oversight.

Summary of contributions:

- **Proof-of-concept architecture** that augments modern applications with capabilities to robustly gauge and monitor the trustworthiness of AI with a human-in-the-loop approach.
- **Implementation of an AI dashboard** that provides a user interface for human oversight in the analysis of models and data, and presentation of quantifiable insights into the decision process of AI for stakeholders to understand and address potential issues affecting performance.
- **Demonstration of the effectiveness of SPATIAL** in augmenting applications with trustworthiness metric through comprehensive benchmarks and real-world industrial application experiments.

4.2. SPATIAL design: trustworthy computing requirements

AI models are trained using data contributions collected over time, with each contribution helping to refine their probabilistic behavior. Regulatory frameworks

define the properties and standards for verifying and validating the correct development and use of AI models. For instance, the General Data Protection Regulation (GDPR) provides guidelines for handling personal data within the European Union, emphasizing fairness, security, privacy, trust, transparency, and explainability in software and AI solution development. Similar principles are reflected in the US AI Act, and comparable regulations are being considered in countries such as China, Japan, Brazil, and Canada. In practice, these requirements demand that modern AI applications include mechanisms or tools that allow users to understand AI inference processes, which in turn requires inspecting the full lifecycle and construction of AI models.

The SPATIAL platform is designed and developed around key principles of trustworthy AI: fairness, resilience, and explainability, in alignment with the EU AI Act. Fairness focuses on mitigating unjustified bias or disparate treatment across protected groups. The SPATIAL fairness module evaluates whether system predictions avoid discriminatory outcomes with respect to sensitive attributes such as gender, race, or age, using statistical parity and opportunity-based metrics. Resilience refers to an AI system's ability to maintain reliable performance despite adversarial attacks, system stress, or environmental disruptions. SPATIAL measures resilience in three areas: robustness against evasion and poisoning attacks, stability under high load or distributed conditions, and recovery capacity following failures or degradations. Explainability covers both interpretability and transparency. SPATIAL operationalizes explainability using a suite of XAI techniques, including SHAP, LIME, and occlusion sensitivity, enabling stakeholders to understand model behavior and decisions effectively.

SPATIAL enhances modern system architectures with new components that enable monitoring of trustworthy properties. These mechanisms function as sensors, providing continuous measurements that can be observed by human operators through the SPATIAL dashboard. Figure 20 illustrates this system augmentation. SPATIAL makes this augmentation using a microservice architecture, which is particularly well-suited for modern system development because it enables modular, independent components that can be developed, deployed, and scaled separately. Unlike monolithic or layered architectures, microservices allow systems to be easily augmented with additional capabilities, such as trustworthy service metrics, without disrupting existing functionality. This flexibility makes it ideal for integrating monitoring, human-in-the-loop mechanisms, and other trust-enhancing features, ensuring that AI-enabled applications remain reliable, transparent, and resilient.

4.3. The SPATIAL architecture

We next describe how modern applications are augmented with SPATIAL, such that it is possible to gauge and monitor the trustworthiness of its AI components. To do this, first, we analyze how sensitive AI pipelines are to vulnerabilities that

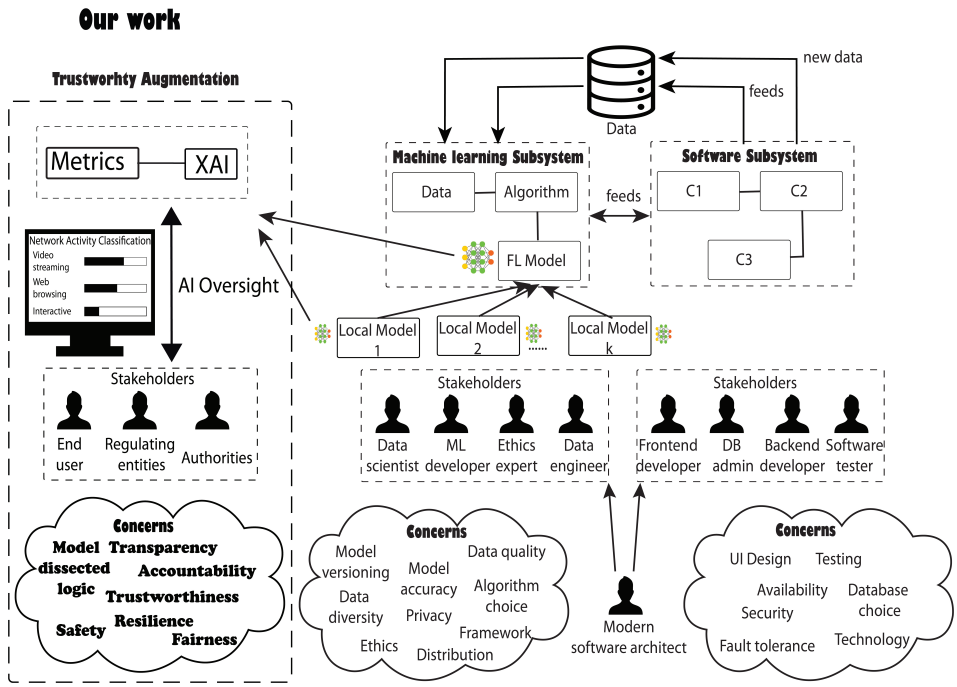


Figure 20: AI model construction: conceptual modern system architecture equipped with methods to monitor trustworthiness

can change the inference logic of AI models during their construction. After this, we introduce the concepts of AI dashboards and sensors, which encapsulate the complexity of the trustworthy analysis. With this information, we provide an overview of the SPATIAL system.

AI perturbations: Machine learning vulnerabilities can introduce perturbations throughout the AI pipeline, which may be exploited to alter the model's inference behavior. We enumerate the most common and critical vulnerabilities by relying on the CIA (confidentiality, integrity, and availability) approach. CIA provides a qualitative analysis to model the impact of vulnerabilities on AI models. Confidentiality depicts the level of access to AI models. Confidentiality is not limited to preventing access to a machine learning model but also to ensuring that its output predictions do not leak information that can be used to understand and reproduce its decision-making or reconstruct its training data. Similarly, integrity relates to preserving expected behavior, level of performance, and quality of predictions under any conditions, including attack. Likewise, availability refers to the idea that accurate predictions are produced, that reflect those seen in testing, and in a timely manner. Models are vulnerable throughout their construction life cycle pipeline. Table 12 summarizes these vulnerabilities together with associated security attributes that can lead to compromise. This suggests that metrics that quantify trustworthiness are required to be instrumented in different steps of the AI pipelines.

Vulnerability	Machine learning lifecycle phase	Threat against training data	Threat against machine learning model	Threat against predictions
Model poisoning	Training	(Integrity)	Integrity	Integrity Availability
Model evasion	Inference			Integrity
Model stealing	Inference		Confidentiality	
Training data inference	Inference	Confidentiality		
Compromised machine learning library	Training + Inference	Confidentiality	Integrity	Integrity Availability
Compromised pre-trained model	Training		Integrity	Integrity Availability
Compromised serialization library	Training + Inference	Integrity Confidentiality	Integrity	Integrity
Compromised training platform	Training	Confidentiality Integrity Availability	Confidentiality Integrity Availability	-
Compromised deployment platform	Inference	-	Confidentiality Integrity Availability	Integrity Availability

Table 12: Vulnerabilities against machine learning systems

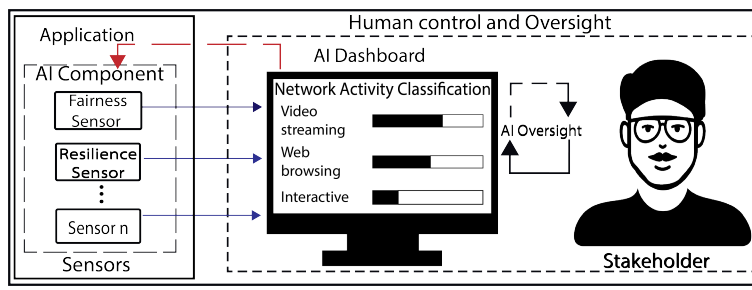


Figure 21: SPATIAL concept overview.

The SPATIAL architecture: SPATIAL augments the latest architectures by building upon the standard machine learning pipeline that constructs and updates AI models. Figure 21 shows the overall concept. Applications are instrumented with AI sensors (dedicated to evaluating each trustworthy property), and these sensors gauge and monitor the inference capabilities of AI models. At the architecture level, Figure 20 shows the system components augmented in modern applications. Notice that the architecture is easily integrated into any application as the trustworthy analysis is applied over the model and data. In practice, the trustworthy properties have to be monitored over time, as these can change as the AI model gets updated. Besides this, it has been demonstrated that trustworthy properties can be considered as trade-offs within applications [442], suggesting that modifying one property can impact others, e.g., robustness vs privacy, accuracy vs fairness, transparency vs security. Moreover, different types of applications have different predominant characteristics, influencing the extraction of a trust score and thus obstructing the adoption of a generic certification scale [26]. By using AI sensors, it is possible then to *quantify* the compliance of AI against available requirements. The main reason for abstracting trustworthy properties into sensors is that a sensor enables continuous monitoring of applications during runtime. AI sensors are software-based (aka virtual sensors) and are instrumented within the source code

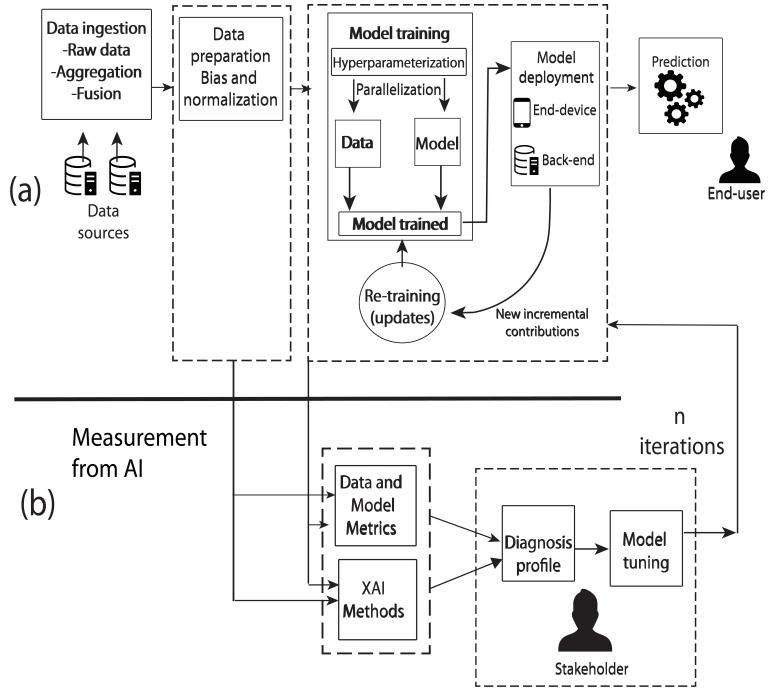


Figure 22: Augmented machine learning pipeline to analyze trustworthy trade-offs

of an application to monitor specific parts of its code execution, or can be instrumented as a concurrent process to monitor the behavior of the overall application. Thus, AI sensors can be considered APIs. Another reason for instrumenting and abstracting modern applications with AI sensors is to foster a correct-by-construction approach, such that standard trustworthy properties are considered from the early design and development phases of AI. Measurements obtained by the AI sensors are shown to human operators using the AI dashboard, such that human operators can aid in overseeing the development of AI models. Human feedback to change AI behavior is applied directly to the AI pipeline. Figure 22 shows the additional steps that are introduced. As any step can be easily hampered to change the model inference process, AI sensors are required to be instrumented across the pipeline. AI sensors are built using specific metrics to extract trustworthy properties, e.g., XAI methods, fairness metrics, and accuracy, among others.

4.4. Technological choices: implementation and deployment

System implementation: To demonstrate how modern applications can be augmented to gauge and monitor trustworthy properties from AI models. We design, develop and deploy a proof-of-concept system architecture. Our current implementation uses a micro-service API gateway to support various micro-services (see [110] for link to implementation code repository). These micro-services implement different metrics to analyze specific trustworthy properties. AI sensors are

instrumented within applications and request the functionality of a specific metric in an input/output manner. The main reason for using micro-services architecture is to add and replace metrics with ease. Indeed, currently, there is a misalignment between legal, regulatory and technical trustworthiness. Thus, technical metrics that fulfill and comply with regulatory requirements are meant to evolve over time. Another reason to rely on micro-service patterns is to dynamically augment the capacity of each individual metric to handle the workload. The source of this workload considers 1) several different applications requesting the metric, and 2) workload caused by continuous monitoring of the metric. To implement our API gateway, we rely on the open-source Kong technology. Kong can be easily extended through OpenAPI and configured to support continuous integration, facilitating re-deployment and managing versioning of our prototype. The API Gateway manages the communication flow, ensuring that each micro-service receives the necessary input, processes it, and returns the appropriate response. Micro-services connected to the API gateway rely on docker containerization to encapsulate each metric. In parallel to this, the front-end implementation (see [111] for link to implementation code repository) facilitates the analysis of AI models through SPATIAL using an AI dashboard. Humans can rely on the AI dashboard to obtain quantifiable, trustworthy characteristics of the AI model. SPATIAL front-end is implemented using React, providing users with an intuitive interface to seamlessly integrate with SPATIAL features. Node.js serves as the required runtime environment for React's development tools, including Babel and Webpack. The Bootstrap 5 framework is utilized for responsive design, while Tailwind CSS is employed for customized styling, resulting in visually appealing UI components. For dataset management and responsive chart visualization, we utilize D3.js, Chart.js, and Papaparse for parsing CSV data. Technical configurations are in subsection VI-B. The overall system is deployed in the computing infrastructure provided by the supercomputer LUMI at UT HPC data-center [427].

Trustworthy metrics for AI sensors: Micro-services implement different metrics to quantify specific trustworthy properties. Applications are instrumented with AI sensors requesting each metric functionality. Current micro-services implement metrics that can be used to support the resilience and accountability of AI models. Accountability metrics support the ability to explain the source causes that led to a decision. Thus, accountability is supported by implementing the XAI SHAP method. SHAP fosters transparency of inference capabilities of AI by highlighting the most important part of the data used for learning. Likewise, resilience metrics quantify the ability of models to resist and recover from an exploited machine learning vulnerability. Resilience insights are thus estimated by calculating complexity and impact metrics on model and data [358]. Complexity quantifies the effort required by an attacker to achieve a successful attack. The higher the complexity, the more difficult it is for the attack to hamper the model. Similarly, impact quantifies the extent of the attack's effect on the AI models within a system. The higher the impact, the more vulnerable the AI model becomes in that system. Besides this,

our architecture also implements a machine learning component, where several AI algorithms can be passed a dataset to create an AI model. This component also allows us to provide performance metrics about the AI model, e.g., accuracy and precision.

4.4.1. SPATIAL back-end and front-end overview

Figure 23 shows the deployment of our augmented software architecture. Next, we provide a detailed description of each component implementation.

Back-end implementation: SPATIAL follows a micro-service pattern to estimate AI trustworthiness based on combined metrics and services. The key idea is that each micro-service specializes in characterizing a specific, trustworthy property, e.g., micro-service for fairness, micro-service for privacy. Micro-service patterns enable easy replacement of metrics for quantifying trustworthiness. This is beneficial as, currently, there is a mismatch between legal and technical trustworthiness. Thus, metrics that align better with legal requirements can be easily updated in SPATIAL. Node.js serves as a foundational runtime environment in our architecture, preceding the API Gateway. It is employed for building scalable server-side applications, leveraging its asynchronous, event-driven programming model to handle concurrent requests efficiently. We rely on open-source Kong technology for our API gateway, which supports easy extensions through OpenAPI and configurations for continuous integration. The API Gateway orchestrates communication, ensuring each micro-service receives the necessary input, processes it, and delivers the correct response. We used NGINX to define Upstreams and API addresses in the configuration file to target particular URL paths to route to the corresponding micro-services. Metrics and services quantifying trustworthiness as micro-services are containerized (using Docker) and follow a request/response scheme. To aggregate metric/service in SPATIAL, a virtual machine is first created, followed by pushing Docker images encapsulating all the dependencies and configurations into the virtual machine. Deployment through Docker containers simplifies the procedure and provides a standardized, isolated environment, ensuring seamless deployment experiences across different instances. Our SPATIAL deployment is located in the High-Performance Computing (HPC) Center [427]. Affiliated with the University of Tartu, and part of the LUMI supercomputer.

Current micro-services include, XAI services (LIME, Occlusion sensitivity and SHAP), fairness metric over data using IBM AIF360 that quantifies demographic disparity, network traffic service applying impact and complexity metrics on AI models, differential privacy service obfuscating data, medical data analysis service implementing visualization methods for explanations, security diagnosis service implementation detection methods of model stealing and data poisoning attacks, LLM service implementing Llama LLM for adapting explanations to specific stakeholder terminology and the ML component implement traditional training functionality for different ML algorithms [328, 64].

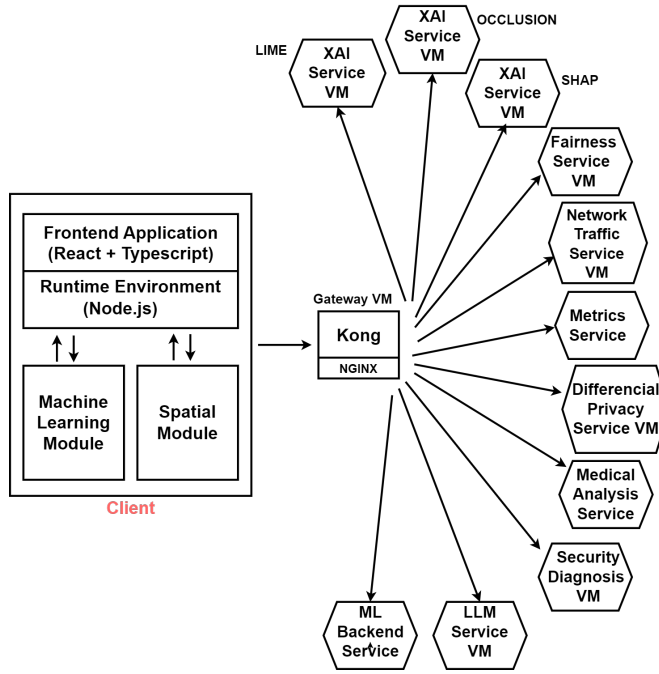


Figure 23: SPATIAL system deployment

Front-end implementation: SPATIAL frontend is implemented using React, providing users with an intuitive interface to seamlessly integrate with SPATIAL features. Node.js serves as the required runtime environment for React's development tools, including Babel and Webpack. The Bootstrap 5 framework is utilized for responsive design, while Tailwind CSS is employed for customized styling, resulting in visually appealing UI components. Additionally, the SPATIAL client integrates Okta for identity management, ensuring secure and robust authentication and authorization capabilities for access control. For dataset management and responsive chart visualization, we utilize D3.js, Chart.js, and Papaparse for parsing CSV data.

4.4.2. SPATIAL usage

Figure 24 shows the overall flow of usage of SPATIAL. First, a user (aka stakeholder) login into the SPATIAL. Here, the user can then select the type of stakeholders, such that the LLM component can adjust the generated explanations from SPATIAL metrics and services based on the expertise of the user. In step 1, after the user is logged in, the user can build AI model using the ML component, see 24.1. To do this, the user has to upload a dataset or provide a link to retrieve the dataset. Alternatively, it is possible for the user to upload its own serialized version of the AI model. Once an AI model is available, the AI model can be analyzed using SPATIAL back-end metrics and services. Thus, the AI model is passed to the AI dashboard. Next is step 2; at this point, both the AI model and the data will

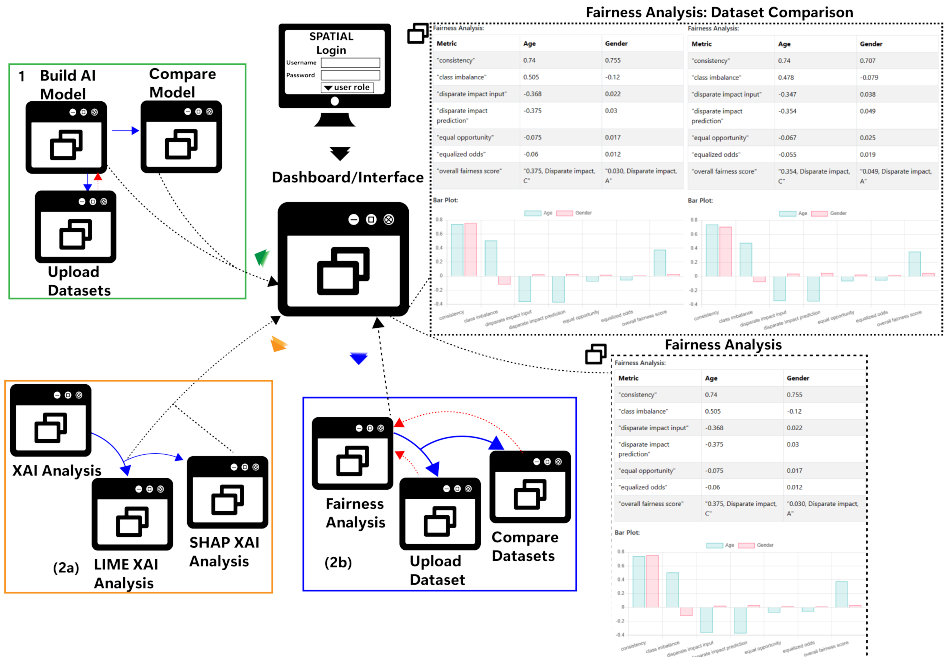


Figure 24: Overall flow of SPATIAL usage and its applicability (fairness only) over a use case application

be passed to the respective micro-services to characterize a specific, trustworthy property (Explainability and fairness properties are considered in this demo, see figure24.2(a) & figure24.2(b)). Each result provided by a micro-service will be visually presented in the AI dashboard, either as a diagram or a text explanation. The AI dashboard is used by the user to understand the quantifiable, trustworthy characteristics of the AI model. After this, the AI dashboard can be used to facilitate changes on either the AI model or data. In this process, a new version of the model or data is created, such that changes can be applied, and SPATIAL can be reapplied in the new versions. SPATIAL also provides a comparison tool feature, such that different trustworthy properties from different AI model or data version can be compared side by side.

4.5. The experiments

We conduct experiments to analyze the performance and scalability of SPATIAL as industrial modern applications are augmented with it. Two sets of experiments are conducted. The first focuses on gauging the trustworthiness properties of AI components of applications, whereas the second focuses on analyzing the capacity of the system to monitor applications and handle the workload of concurrent requests. In the following, we provide a detailed description of the experimental setup.

4.5.1. Monitoring performance.

We next evaluate how SPATIAL can gauge and monitor the inference capabilities of AI. To do this, we analyze how changes in AI models can be quantified and monitored over time. Monitoring the inference process is important to identify when models have been compromised. The first use case focuses on analyzing sensor data to trigger medical emergency support, whereas the second application depicts a network activity classification system, where network data is poisoned to disguise the classification model.

Use case 1: Medical e-calling application. It is a mobile application, part of an e-calling system, that uses accelerometer data to detect the falling of an elderly person. As the falling event is detected, the application triggers an emergency call to request medical assistance.

Dataset and model: The UniMiB SHAR dataset [300] was employed in training five different ML models, Logistic Regression (LR), Random Forest (RF), Multilayer Perceptron (MP), Deep Neural Network (DNN), and Decision Tree (DT). The UniMiB dataset is a benchmark dataset for human activity and fall detection comprising 11,771 acceleration samples from 30 subjects, 9 classes representing activities of daily living (ADL), and 8 classes representing falls.

Adversary model and assumptions: We assume a black-box attacker model where the attacker has only access to the training data but has no knowledge about the underlying structure of the utilized model. Furthermore, we expect the attacker to be capable of randomly poisoning the data up to a poisoning rate of p. Thereby, we expect that the attacker poisons the data by performing a random label-flipping attack.

Setup and procedure: The label flipping attack is performed systematically to different subsets of the dataset. Precisely, the attack is executed at varying poisoning rates p of 0% (baseline), 1%, 5%,10%, 20%, 30%, 40%, and 50%, respectively. Baseline results without data poisoning are also collected for reference purposes. Afterwards, the respective ML model (e.g., DNN, DT, RF, LR, or MLP) is trained on the poisoned training dataset and then evaluated with the retained clean test dataset based on the accuracy, precision, and recall evaluation metrics. In addition, we explore the impact of the attack on the model's explainability. More specifically, we also calculate the similarity of SHAP explanations of the DNN model for each of the varying poisoning rates. To realize this, we determine the five nearest neighbors regarding the Euclidean distance for each fall instance in the retained clean test set. We then measure the average distance of the corresponding SHAP explanations. Finally, we average the average distances of explanations, resulting in an average distance of explanations of similar instances across the test set w.r.t. the class "fall".

Use case 2: Network activity classification application. The second use case is a network monitoring application that examines IP and TCP/UDP data headers. The application is able to identify the type of activity an online user is performing. Three

common types of online activities are considered: Web browsing, Web interactions and video streaming. Network monitoring is important to design security policies, safeguard user privacy and efficient dynamic allocation of resources, particularly in 4G/5G networks.

Dataset and model: We set up a testbed to collect network data of user activities using our application. Network data depicts real online activities of users at [Annon. Vendor], a network data monitoring provider. We rely on Wireshark to create pcap files with a size of 2.15 GB that contain the activities of users captured through the network traffic. Our datasets comprise multiple network traces, linked to different users. The network traffic traces contain essential information such as the source and destination IP addresses, protocols, port numbers, packet timestamps, and packet size, to mention some. We clean the dataset using standard methods and select relevant features to identify the previously described activities. After applying filtering methods, the final dataset consists of 382 labeled traces across three traffic classes: Web, Interactive, and Video activities, with 304, 34, and 44 traces respectively. The processed CSV files derived from this dataset are used for the analysis and evaluation of our AI-based classification model. Feature extraction reveals 21 features categorized into five main categories: duration, protocol, uplink, downlink, and speed. We employ various machine learning classification algorithms, including Neural Networks (NN), LightGBM (LGBM), and XGBoost.

Adversary model and assumptions: We assume a white-box attack model, where the attacker has complete knowledge about the AI model structure. This type of attack depicts a common situation where the AI models are hampered from inside an organization. By injecting commonly use poisoning and evasion attacks, the attacker's objective is to compromise the integrity of our models leading to a significant degradation in the model's accuracy. Fast Gradient Sign Method (FGSM) is a technique used in adversarial ML to generate adversarial examples by adding a small amount in the direction of the gradient of the loss function with respect to the input. Resilience of models against an evasion attack is quantified based on impact and complexity metrics. Here, complexity is measured by characterizing the processing power required to generate evasion data points. Impact, on the other hand, is measured by counting each successful misclassification gained through those evasion data points. In parallel to this, a GAN-based poisoning attack is also performed, and the goal is to generate synthetic data that looks very similar to the real data. Random swapping labels attack chooses randomly two samples of the training dataset and swaps their labels. Target label flipping attack flips the labels of some samples from one class to the target class (e.g., Video class). Here, complexity and impact are also estimated based on different observations. Complexity is measured by quantifying the percentage of data that is poisoned out of all the data used for training the model. Similarly, impact is measured by using the drifts in any performance metric of the model, e.g., accuracy, F1-score.

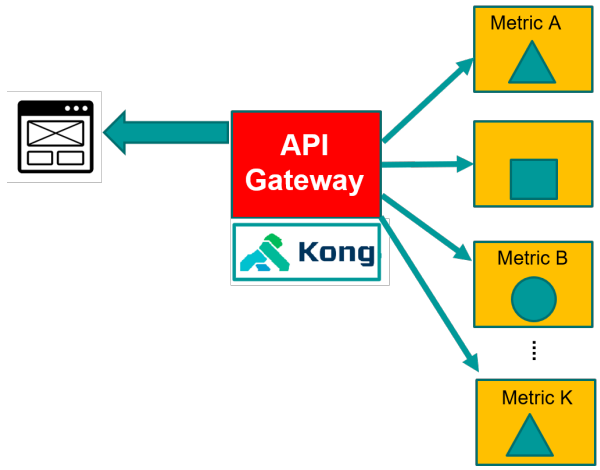


Figure 25: System deployment schema

Setup and procedure: We generated 103 adversarial samples from the 103 test data samples that were initially obtained. After this, the white-box FGSM evasion attack is launched. For the GAN-based attack, we use CTGAN [460] for modeling tabular data to generate 5000 synthetic samples. For other poisoning attacks, such as label flipping and random swapping labels attacks, the poisoning rates are 0% (baseline), 10%, 20%, 30%, 40%, 50%. Subsequently, the corresponding ML models (e.g., NN, LightGBM and XGBoost) are retrained using the manipulated training dataset and compared against the baseline to identify performance degradation based on accuracy, precision, and recall metrics.

4.5.2. Capacity-load performance

Experimental setup: To verify the performance and scalability of SPATIAL, SPATIAL is deployed following the setup shown in Figure 25. The system consists of six (6) different machines, one acting as the integration/API gateway, and others as back-end micro-services. The machine running the Kong Gateway consists of 32 vCPUs and 64 GB of RAM running Linux. The remaining machines host a specific service to extract a metric. Micro-services include a LIME micro-service (4 vCPUs and 4 GB RAM); a SHAP micro-service (4 vCPUs and 4 GB RAM), an Occlusion-sensitivity micro-service (4 vCPUs and 8 GB RAM), an impact resilience micro-service (computing instance with NVIDIA A4000 GPU, Intel Xeon 2.10 GB CPU, and 128 GB RAM running Ubuntu 20.04), and an AI pipeline micro-service that provides performance indicators (8 vCPUs and 8 GB RAM). All micro-services are accessible through the API gateway, and requests to micro-services are specified by the clients. The system is deployed in the computing infrastructure provided by LUMI.

Tools and metrics: Once the system is deployed and running, capacity-based testing is performed to evaluate the performance of individual requests and concurrent

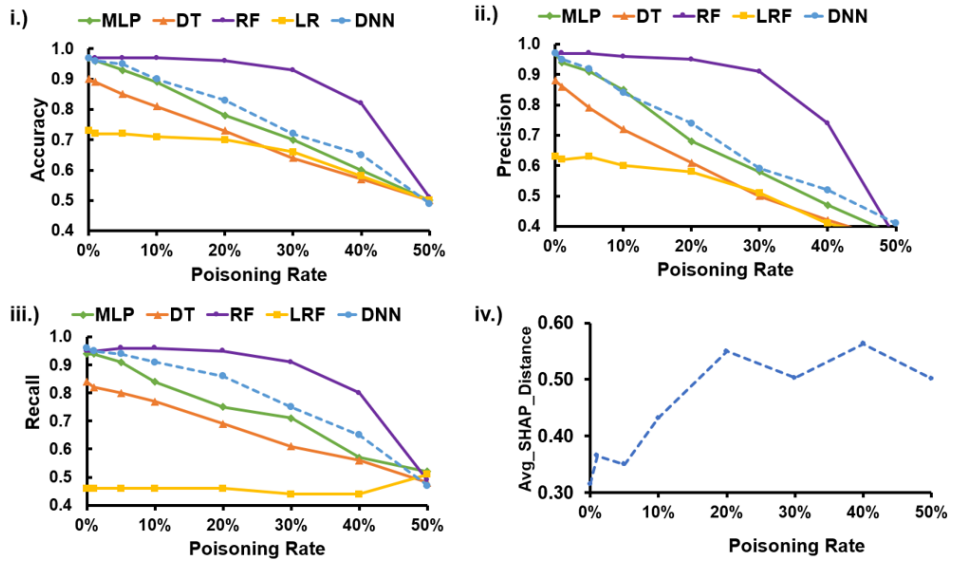


Figure 26: Use case 1 results (medical application); Effect of label flipping based on (i) accuracy, (ii) precision, (iii) recall; and (iv) poisoning quantification using SHAP dissimilarity

requests, handled by the system as its usage increases, depicting an in production environment. To generate stress capacity load, we rely on JMeter, deployed in a different machine, but running in the same network as the SPATIAL deployment. JMeter is installed in a Windows machine with an 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz CPU and 16 GB RAM.

Experiment 1: We evaluate perturbations in the AI model and derive the impact that poisoning attacks have on resilience. We also evaluate the analysis of SHAP and LIME values over model predictions. In the configuration process for the JMeter script, we create a test plan encompassing an ultimate thread group with a thread count set to 100 to simulate concurrent requests to the micro-services. To examine the performance of specific micro-services, an HTTP request sampler was added, specifying the server name, port, protocol, and endpoint path. Parameters or file uploads were configured as necessary. To gauge response times, the Response Times Over Active threads or the Summary Report listener was incorporated into the test plan. These listeners provided detailed metrics, including average response time, throughput, and error rate for each micro-service.

Experiment 2: We next evaluate the performance of the system when handling heavier loads induced by image inputs. In this case, when analyzing image-based samples, the analysis of methods, such as LIME, SHAP, and Occlusion sensitivity, increases. As a result, we analyze the extent to which these services impact the overall response time. Notice that the configuration presented in experiment 1 cannot be handled by these services when considering input images. As a result, with this setup, a different capacity load is generated. We select incremental

concurrent load from 5 to 25 requests. Requests are also set to be sent to services with a ramp-up period of 1*second* in parallel.

4.6. Results

Monitoring results on use case 1: Prior to poisoning the models, reference baselines of the models are established to measure performance deviation. Our performance evaluation indices, LR (73%), DNN (97%), RF (97%), DT (90%), and MLP (97%), respectively. Moreover, our results indicate that DNN, MLP, and RF models are best suited for fall detection when compared to others. It is also possible to observe from the results that DNN, MLP, and RF are able to attain 97% accuracy and precision in performing the binary classification task, but at slightly different recall rates, respectively. After this, models are poisoned. Figure 26 shows the results. From the figure, it is possible to observe that label flipping has a significant impact on model performance, with most metrics decreasing as the attack rate increased (Figure 26(a)-i shows accuracy, figure 26(a)-ii shows precision and 26(a)-iii shows recall). In line with this result, the average performance of all the models in accurately detecting falls before the data poisoning attack was 90%. However, this average performance starts to decline to 75% as the data is gradually poisoned from 1% to 50%. We calculated a metric based on SHAP values that addresses the similarity of SHAP explanations of similar data points. Figure 26(a)-iv illustrates the results of this metric relative to the poisoning rate of the model. As can be seen from this figure, the metric is higher at higher poisoning rates, suggesting its capability of indicating poisoning of the dataset. This result alone provides insights for detecting possible attacks on the model, requiring further monitoring of the model to apply corrective actions, e.g., Label sanitization methods. Besides this, analysis of the result indicated that the high-performing models (DNN, MLP, and RF) showed relatively small performance losses at low attack rates (1% and 5%), indicating some degree of robustness in maintaining their capabilities to still detect fall, up to 5% poisoning rate, but this is lost when the intensity exceeded 5%. Interestingly, the random forest (RF) model showed better resilience against the poisoning attack. Even at a 30% poisoning rate, the RF model maintained an accuracy of 93%, close to its baseline performance. Only at a poisoning rate of 40% did a significant performance decrease occur, rendering the model unusable. The RF recall and precision metrics were also relatively stable, up to a 30% poisoning rate, further highlighting its robustness.

Monitoring results on use case 2: A reference baseline about the performance of our models for user activity classification is estimated to be NN (96%), LightGBM (94%) and XGBoost (94%). After this, the (FGSM) evasion attack is performed over the models, degrading their performance to NN (71%), LightGBM (72%) and XGBoost (54%). We then use SHAP to observe differences as models get hampered. Figure 27(a) and (b) shows the results of SHAP when applied to NN, before and after the evasion attack. From the result, it is possible to observe that

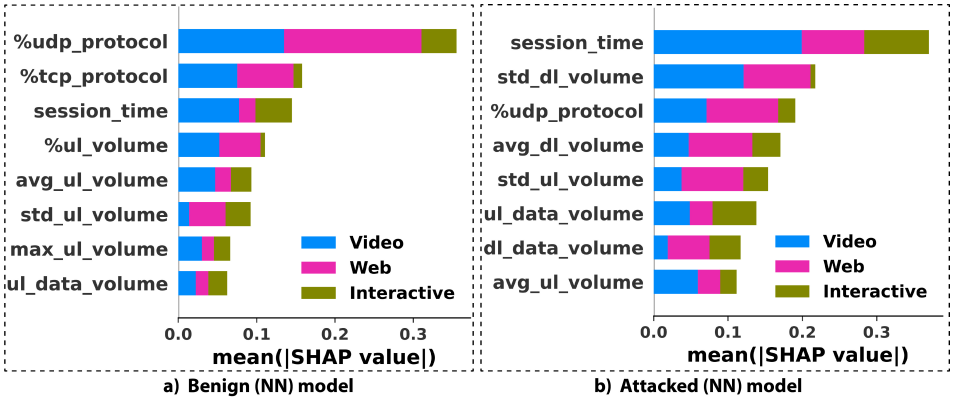


Figure 27: Use case 2 results (network activity monitoring); SHAP analysis for evasion attacks; a) Benign (NN) model, b) Attacked (NN) model

Shapley values for web activities have decreased around 16% for the udp_protocol, causing the feature to drop to the second place in ranking, while the importance of the tcp_protocol has almost doubled. This means that attacks on the model can easily induce misclassification of user activities. At the same time, it is possible to detect these changes with SHAP; however, the detection alone is insufficient to identify concrete causes or overall performance degradation of the model, requiring additional information to be computed. Thus, *complexity and impact metrics* are calculated from the models using the methods presented in [360].

For each model, impact and complexity are estimated, NN (Impact 29%, Complexity 37.86 μs), LightGBM (Impact 28%, Complexity 37.86 μs) and XGBoost (Impact 45%, Complexity 37.86 μs). The results of the metrics indicate that XGBoost is (17%) more vulnerable for the FGSM attack when compared with the other two models. Moreover, since the FGSM generation was done with only the NN model, the complexity of the attack was always constant at around 37 μs. In parallel to this, in the case of poisoning attacks, SHAP can provide valuable insights to detect changes in performance. For instance, after label flipping and GAN-based poisoning are performed in our models, it is possible to observe Shapley values for web activities have also changed significantly (tcp_protocol increases by 10% while udp_protocol decreased to half of its initial importance). To reinforce this detection further, we then calculate impact and complexity metrics to further analyze the impact of poisoning in our NN model. Figure 28 shows the results estimated by impact and complexity metrics. From the results, we can observe how metrics changed based on the level of poisoning applied. We can observe that there is an increasing relative trend between increased poisoning and drift in impact and complexity.

Capacity-load results: Experiment 1 results are shown in Figure 29(a) and Figure 29(b). The figures show capacity results when handling concurrent requests by the impact resilience micro-service and LIME/SHAP micro-services, respectively. From the results, it is possible to observe a lower response time for the

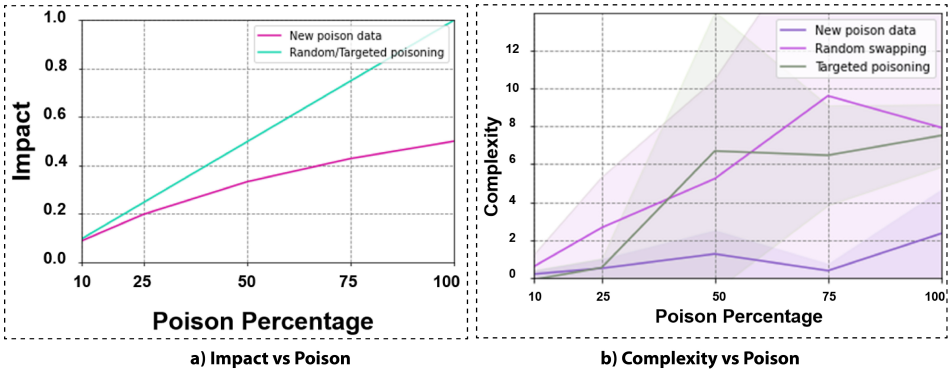


Figure 28: Poisoning attacks quantified by impact and complexity metrics; a) Impact vs Poison%, b) Complexity vs Poison%

evasion impact metric. Even with nearly 100 parallel requests, the numerical metric converges to an average of around 1600 milliseconds across the ramp-up time. Similarly, SHAP's and LIME's APIs under 100 requests are also presented in Figure 29(a). From this result, it is possible to observe that SHAP's and LIME's explanations require an average processing time of 228.6 and 243.4 milliseconds, respectively. In both cases, the response times depict latencies that are tolerable by end-users and can also be used for continuous monitoring. Notice, however, that XAI methods can also be used to analyze images, such that it is possible to obtain a representation regarding which parts of the images the model used to learn. Thus, we also evaluate LIME to handle resource-intensive workload (Experiment 2). Figure 29(c) shows the results of experiment 2. From the figure, it is possible to observe that LIME methods require a considerable amount of computation. As a result, when facing resource-intensive processing, XAI are not able to handle concurrent workload below 1s. In fact, we can observe a steady increase in response time that depends on the number of concurrent users accessing the service. This has direct implications for the types of models/datasets that can be analyzed with available XAI methods.

4.7. SPATIAL performance testing in the wild

We demonstrate the responsiveness and the capability of SPATIAL to handle workload or users' demand through an extensive load and performance evaluation within a controlled setup. This assessment is crucial to understand SPATIAL's operational capability for scalability, reliability and the feasibility of SPATIAL in a real deployment scenario where multiple users may concurrently interact with the system. Given SPATIAL's microservice architecture, each component was tested in isolation to identify potential bottlenecks and assess performance under increasing user loads, specifically targeting up to 100 concurrent users as specified during the requirement gathering phase. Besides understanding the capacity of SPATIAL, we also conducted performance testing to align our development with

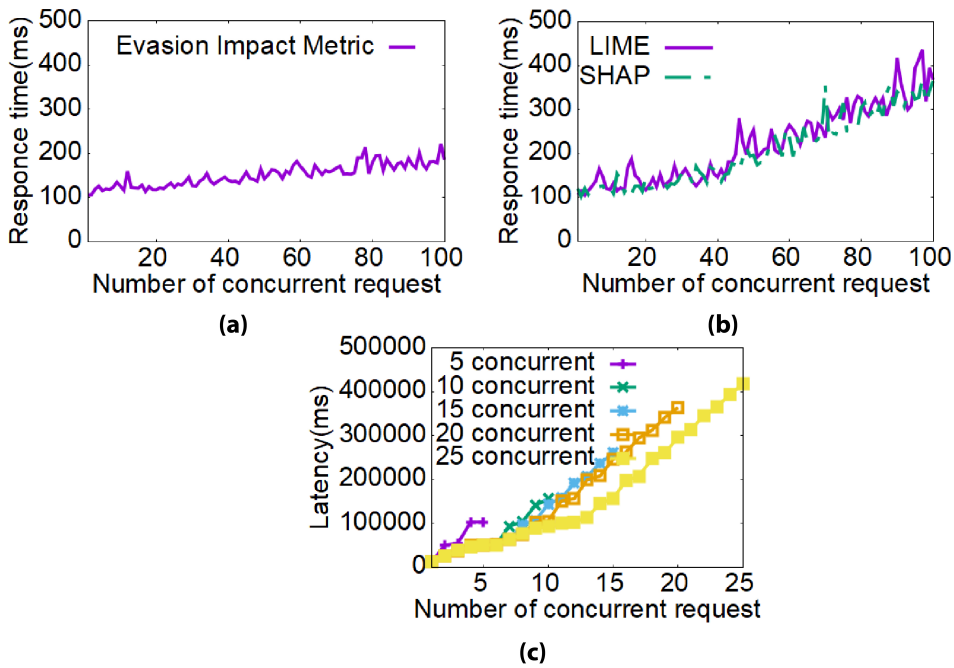


Figure 29: Capacity-load experiments, a) Load in impact metric; b) Load in LIME and SHAP; and c) Load in LIME when handling requests requiring heavy computations.

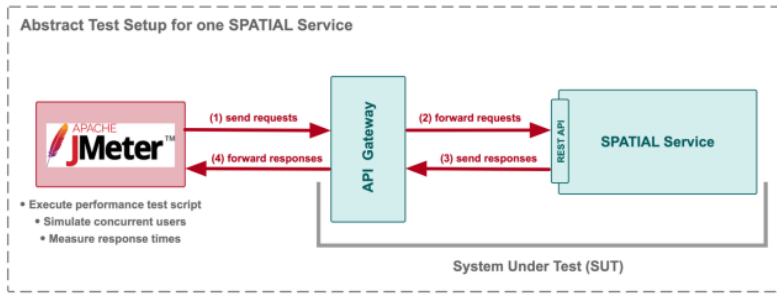


Figure 30: SPATIAL test setup

regulatory expectations by demonstrating with empirical evidence that services measuring trustworthy attributes like fairness, explainability, and privacy, etc, can operate efficiently without compromising responsiveness. This effort enables us to discover bottlenecks and initiate improvements at the component level to optimize resources, tune SPATIAL, and build confidence among stakeholders.

Testing setup: Figure 30 illustrates the SPATIAL test setup for achieving robust testing of its constituent services. The performance testing setup adopts an independent evaluation approach at the microservice level, such that each component (e.g. explainability, privacy, fairness) can be distinctly evaluated through an integrated API gateway, like in a real-world deployment. Load was generated using Apache JMeter, simulating up to 100 concurrent users by gradually ramping up and down over a 42-minute test duration. All requests were routed through the Gateway, which handled service orchestration and ensured consistent request flow across components. This setup enabled a reproducible and scalable testing environment to evaluate system responsiveness, detect bottlenecks, and ensure the operational robustness of each trustworthiness service in realistic conditions.

Performance testing results:

We present the load testing results for SPATIAL's microservices starting with the differential privacy service: Figure31 illustrates the workload capacity performance of SPATIAL's Differential Privacy (DP) component under simulated federated learning conditions, specifically assessing the response time when processing concurrent updates from configurations involving 10 and 100 clients. The performance tests spanned a total duration of 2.6 hours and investigated two scenarios: one where the client explicitly provided the noise multiplier parameter (sigma), and another where the component was responsible for computing sigma dynamically during each request. The results indicate that when sigma is predefined and passed as an input, the component maintains consistent and acceptable response times across both client configurations, demonstrating good scalability. In contrast, dynamically computing sigma for each request introduces substantial latency, particularly under high concurrency, where many clients are sending requests at the same time. This results in timeouts as the workload increases. To address this bottleneck, the component was enhanced to cache and reuse the sigma

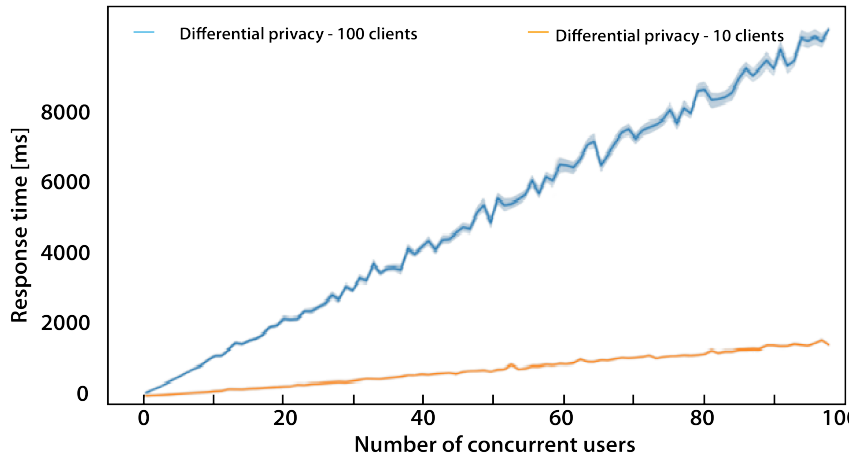


Figure 31: Privacy component load testing performance[1]

value computed during the initial request, which corresponds to the first round in a typical federated learning process, thereby preserving the required differential privacy guarantees while significantly improving performance under load.

Figure 32 presents the performance evaluation of the Fairness component under simulated concurrent usage conditions, specifically assessing the response time of its numerical and visual analysis endpoints as the number of users increases. The evaluation involved up to 100 concurrent users and measured the SPATIAL's responsiveness across six key endpoints responsible for computing individual and group fairness metrics. The results show that numerical endpoints such as disparate impact, equal opportunity, and consistency scores consistently maintain low response times, averaging around $1,600$ milliseconds, even under peak load. However, endpoints responsible for generating visual fairness explanations, including consistency and compacity plots, exhibit significantly higher response times, averaging approximately $16,900$ milliseconds. This increased latency is attributed to the computational overhead of rendering and exporting graphical outputs using visualization libraries like Matplotlib and Kaleido. Overall, the component demonstrates strong scalability and responsiveness for core fairness computations, while highlighting the need for optimization or asynchronous handling of visualization-heavy tasks in critical performance or large-scale deployment scenarios.

Next, we evaluated the performance of the XAI-resilience component of SPATIAL. Figure 33 illustrates the performance evaluation of the use of three distinct XAI methods within the resiliency component for understanding experimented attacks, see 4.5, under simulated concurrent user conditions, specifically measuring the response times of explanation generation and data poisoning endpoints as user load increases up to 100 concurrent threads. The evaluation distinguishes between lightweight GET requests for generating SHAP and LIME explanations, and heav-

[1]Original image in SPATIAL deliverable D.3.4

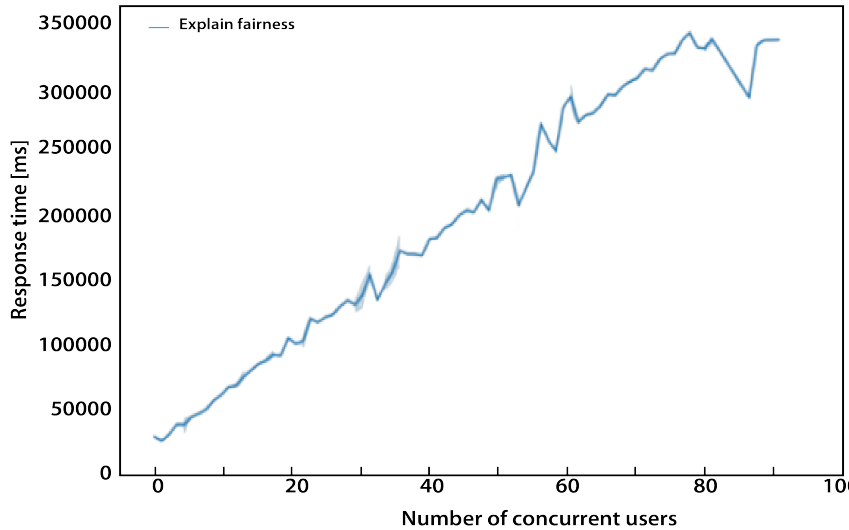


Figure 32: Fairness component load testing performance[2]

ier POST requests for executing poisoning attacks such as Random Swapping Labels (RSL). The results demonstrate that explanation endpoints maintain strong responsiveness under load, with average response times of approximately 228.6 milliseconds for SHAP and 243.4 milliseconds for LIME, even at peak concurrency levels. These findings confirm the component's efficiency and scalability in delivering real-time interpretability insights to users. In contrast, the RSL poisoning attack endpoint exhibits significantly higher latency, with an average response time of 18,041.4 milliseconds, due to the computational overhead involved in modifying training data and retraining models. Despite this, no request failures were observed, indicating that the component remains functionally robust under stress. Overall, the evaluation highlights that while the XAI for Resiliency service scales well for explanation tasks, high-cost operations like attack simulations should be managed with resource-aware strategies or offloaded asynchronously in performance-critical environments.

Lastly, we consider the implications of increasing demand on the metric service component. This component comprises six distinct metrics that are computed during a request. Figure 34 presents the performance evaluation of the metrics component under simulated concurrent user conditions, specifically measuring the response times of six key endpoints, comprising both numerical and visual trustworthiness metrics, under a load of up to 100 parallel users. The results reveal that numerical endpoints such as accuracy metric, consistency metric, compacity metric, and evasion impact metric consistently maintained low latency, with average response times around 1,600 milliseconds, even as the number of concurrent requests increased. In contrast, endpoints responsible for rendering visual outputs, consistency-metric-plot and compacity-metric-plot, exhibited significantly higher

[2]Original image in SPATIAL deliverable D.3.4

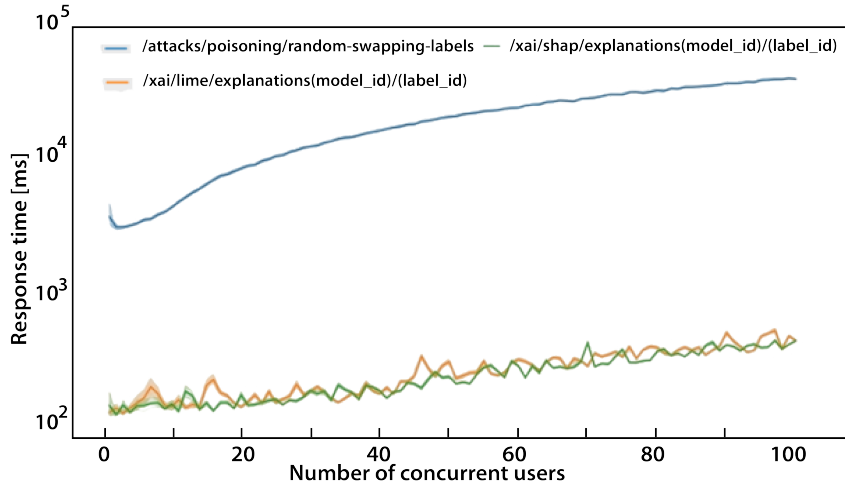


Figure 33: XAI component load testing performance[3]

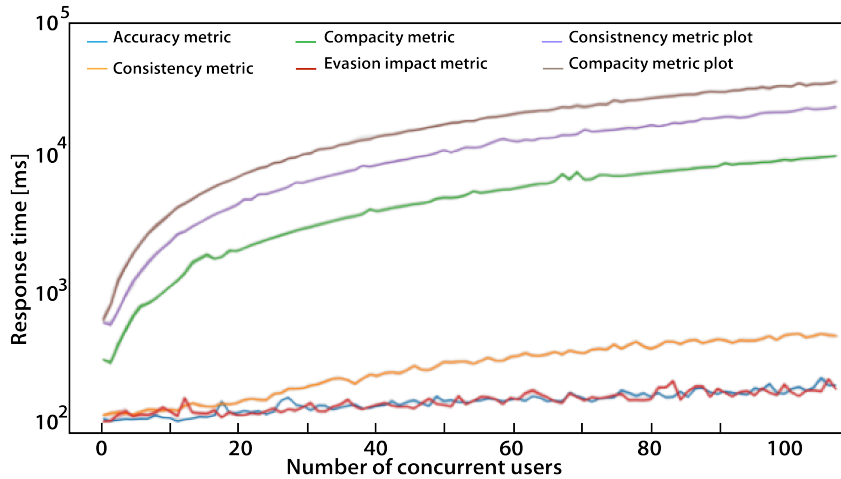


Figure 34: Metric component load testing performance[4]

response times, averaging approximately 16,900 milliseconds. This disparity stems from the additional computational overhead required to generate and format visual explanations using libraries such as Matplotlib, Shapash, and Kaleido. Despite the increase in response time for plotting endpoints, the system successfully handled the full concurrency range without request failures. These results confirm that the Metrics Component is well-optimized for interactive and large-scale use of numeric metrics, while also highlighting the need for performance-aware handling of visualization tasks in environments with high user load.

[3]Original image in SPATIAL deliverable D.3.4
[4]Original image in SPATIAL deliverable D.3.4

4.8. Challenges, outlook and experiences

While all regulatory frameworks agree on the strategic importance of AI trustworthiness, the development of trustworthy AI is an ongoing process. While principles, tools, guidelines and methods are available to aid in this matter, there is still a gap between regulations and technical requirements. Thus, there are several challenges that remain open for augmenting modern applications with AI trustworthiness capabilities. Based on our experiences, we next highlight technical challenges that require further attention for complying robustly with the trustworthy AI requirements.

AI trust score and AI sensors: AI trustworthiness involves the characterization of several properties [260], including technical (e.g., validity, accuracy, reliability, robustness, resilience, or security) as well as the socio-technical characteristics (explainability, interpretability, managing bias, privacy enhanced, safety). Each property can be obtained through specialized metrics, based on the nature of the area of application at hand. For instance, in a loan application, fairness can be applied to identify data biases in individual or specific groups (equitable), whereas fairness can also be calculated to estimate whether the decision process was fair to all the involved loaners (procedural). Similarly, in an object detection application, explainability can be generated using occlusion sensitivity to identify the most relevant area on an image contributing to the object detection. In turn, LIME divides the image into multiple section areas and ranks each accordingly to measure their contribution to the overall model prediction. Encapsulating all different properties into AI sensors is a key challenge to foster the easy integration of trustworthiness in current software development practices. AI sensors can provide general procedures and guidelines to instrument applications with trustworthy mechanisms. Another important challenge is to produce a coherent and comparable trust score from measurements obtained by AI sensors, such that trustworthiness can be understood as an overall feature of applications. While the development of a trust score has been explored by previous work [86], these solutions simplify the extraction of trustworthiness by considering all properties homogeneous and not considering its different inherent characteristics.

Human oversight and AI tuning: As part of the EU AI Act, humans play a critical role in overseeing the behavior of AI. AI dashboards can provide critical information about the AI inference capabilities to stakeholders. For example, the level of fairness, robustness, and resilience, to name a few. Through the dashboard inspection, individuals relying on AI models can be aware of the limitations and scope of the decision support provided by AI models. Ultimately, dashboards can support humans to decide whether or not to use AI for aiding with a particular task. Moreover, as the trustworthy properties are considered trade-offs that can be adjusted depending on the requirements of different stakeholders using the applications, it becomes critical to tune these properties over time. Existing methods can be used to perform hyper-parametrization on the way an AI algorithm

learns and thus adjust its resulting decision process [449]. As the tuning of models is an iterative process that involves a reinforced human-in-the-loop feedback rather than a single shot, a key challenge is to integrate such a process in the construction of AI models. To obtain significant feedback from stakeholders, it is important that explanations describing the overall trustworthiness of a model are tied to specific domain terminology of stakeholders, e.g., tailored explanations for end users and software developers. An extra layer of transformation is thus required to map understandable insights of a model to a specific target audience. A potential solution is to rely on large language models (ChatGPT-like preamble) or a meta-model that dynamically changes the explanations to a specific domain audience. Besides this, another key challenge is to determine what changes can be applied to the model by individuals. For instance, removing personal data from the training dataset or changing the machine learning algorithm. This is a critical challenge to overcome as AI models have to support the individual needs of users while preserving general values from groups and society. Otherwise, conflicts on AI usage may arise, halting everyday activities and human processes. Another remaining challenge is to develop AI dashboards that motivate users to be involved in the AI tuning process [337].

Adversarial threats over AI algorithms and data: As demonstrated in our experiments, the decision process of AI models can be changed abruptly. Induced changes (aka attacks) are of particular interest as proactive counter measurements have to be taken rapidly by human operators; otherwise, compromised applications can become a source of harm for citizens and urban infrastructure, e.g., attacks on drone delivery [392]. Other examples of this include adversarial generative patches that confuse AI models and poisoned data that can make devices drain energy at faster rates, e.g., sponge attacks in IoT devices. As there is a large plethora of attacks that can hamper AI functionality, a key challenge is to quantify the level of AI resilience to attacks by applying multiple detection methods and suggesting countermeasures to human operators. Naturally, the level of resilience depends on the available methods that attest whether the model/data has been compromised. Besides this, while some post-defacto verification methods could be applied to detect attacks over AI functionality, other methods require re-playing the overall training process, involving a more time-consuming analysis.

Privacy-preserving data and computations: Data is a key element in the machine and deep learning pipelines, building AI models. Regulatory guidelines in the use of data, e.g., EU GDPR, forbid the inclusion of private and sensitive data that can be used to identify specific individuals [136]. Thus, data is required to be obfuscated before it can be used within the AI pipelines. Existing solutions to aid in this matter include differential privacy and data anonymity techniques [181]. However, data removal degrades the decision-making process performance, requiring new methods to obfuscate sensitive information without reducing model performance levels, e.g., sparse coding and compressive sensing compensation models. At the

same time, since direct access to model and data are required to estimate different trustworthy properties, a key challenge is to guarantee that the analysis of these properties is conducted in a secure manner to avoid potential induced attacks over AI. Existing methods based on multi-party computation, homomorphic encryption and TEEs (Trusted Execution Environments) could be adopted in this matter. However, integrating these mechanisms within existing architectures requires managing extra computation overhead as well as solving several technological limitations to achieve scalable solutions. For instance, while TEEs currently support secure computation, they impose limitations on software runtime execution characteristics, e.g., programming language, dependency management, and storage requirements [264, 331],etc.

AI sensors: AI sensors are designed to be embedded at the code level of modern applications, enabling analysis of serialized AI models (e.g., JSON/YAML), datasets, and pipelines. Acting as APIs, they use standard technologies to support system integration and interoperability. A clear separation between the interface (client API) and functionality (back-end) allows lightweight instrumentation, reduces processing overhead, and enables functional upgrades, e.g., via microservice architectures, without modifying the end application. This is crucial for addressing the mismatch between technical and legal trustworthiness. Since multiple AI sensors may be needed to evaluate different trustworthiness properties, offloading functionality to remote infrastructure helps manage processing demands. Moreover, AI sensors are designed to interact and autonomously negotiate trust levels based on application context, which requires additional processing. This is especially valuable in dynamic, context-sensitive scenarios where data use may depend on user consent. In such cases, AI sensors can assist users by automating aspects of data handling and management, provided users are aware of and can configure their preferences within the application [152].

AI dashboards: An AI dashboard presents concise visual insights from AI sensors, enabling users to inspect, assess, and, where appropriate, tune AI behavior. While all trustworthiness properties can be visualized, their relevance and information displayed depend on the application domain. For example, fairness is critical in finance, healthcare, or employment contexts, but less so in autonomous systems like self-driving cars. This calls for adaptive content organization, using techniques such as hierarchical analysis or progressive disclosure. Tuning AI models is not a task for individual users but requires input from domain experts who interpret user feedback and adjust the models accordingly. AI dashboards support this by surfacing relevant inference insights to users while enabling experts to refine models using tools like Ray Tune, Optuna, Hyperopt, Vizer, Microsoft NNI, Keras Tuner, and SigOpt. Given the potential impact of model tuning, secure mechanisms are essential to prevent malicious or unintended tampering [152].

AI perturbations: Attacks on machine learning systems can be identified by threat modeling using frameworks like ENISA, MITRE, NIST, IBM, Microsoft.

Attack Type / Algorithm	DNNs	SVMs	Decision Trees	Random Forests	GBTs & XGBOOST	Bayesian Networks
Poisoning Attacks	[488] [381] [277] [263]	[55] [444]	[13] [20]	[412] [127]	[309] [55]	[19] [20]
Evasion Attacks	[334] [83] [100] [71]	[218] [97] [56] [57]	[315] [99] [13] [334] [96]	[315] [13] [226]	[13] [96] [226] [103]	–
Model Stealing Attacks	[193] [126]	[357] [281] [107]	[419]	[31] [283]	[273]	–
Data Interference Attacks	[389] [483] [178]	[272] [358] [21]	[421] [31] [162] [412]	[283]	[31]	–

Table 13: AI perturbations on algorithm and attack type

AI pipelines implement a set of steps to build AI models. These models can be hampered by induced and non-induced changes in any step of its construction [392]. Non-induced changes occur due to situational events, e.g., environment, data quality and failures of devices. Induced changes (aka adversarial attacks) are perpetrated by an attacker with the main intention to control/induce the inference process of AI models. Poisoning attacks are of a significant issue as contaminate the data used for model training [483, 488, 381, 277, 83, 444, 55, 315, 13, 127, 55, 19, 334, 99, 226, 193, 357, 107, 419, 31, 283, 273, 389, 178, 358, 21, 421, 31, 20, 272]. Adversarial attacks can also occur at the model level by changing internal structure and parameters of the model [334, 83, 100, 218, 97, 13, 226], e.g., model evasion, model stealing. A summary of attacks investigated in the relevant literature in the last years is shown in Figure 13. From the figure, it is possible to observe the type of attack that can be performed depending on the AI algorithm used for training. SPATIAL augments modern applications with functionality to gauge and monitor changes in AI inference capabilities such that human operators can visualize and react to them.

4.9. Discussion and implications

Legal vs technical trustworthiness: Our work presents the design and development experiences from augmenting modern applications with capabilities to gauge and monitor AI trustworthiness. The selected metrics of our prototype are considered from a technical point of view based on the most common methods currently adopted to analyze AI black-box characteristics. We are interested in replacing our metrics with others that align better with regulatory trustworthiness. This, however, requires conducting a legal analysis that considers all metrics available in the state-of-the-art to identify the most suitable. This analysis is out of the scope of this work.

Cost and complexity: SPATIAL not just augments modern applications with new regulatory functionality, but it also augments the amount of components and enlarges the underlying deployment of the overall system running the applications. This increases the complexity of developing and maintaining the applications.

Moreover, the cost of the deployment also increases as it is not possible to piggyback on already existing infrastructure due to the increased load required for computation. Indeed, as shown in our experiments, methods such as XAI can induce a heavy load in the overall system, requiring them to be deployed on their own dedicated machine.

Verifying vs embedding trustworthiness: Current practices to analyze trustworthiness of AI inference capabilities rely on post-defacto verification of models. The use of AI sensors can foster the embedding of mechanisms to gauge and monitor AI trustworthy properties from early development and design phases. This, however, requires /standard procedures on how to create AI sensors (like APIs) that encapsulate each trustworthiness property. Moreover, guidelines and best practices on how to instrument modern applications with AI sensors are also required to facilitate their adoption in software development practices.

Adaptive trustworthiness: In our work, we present the encapsulation of trustworthy properties into AI sensors. More advanced AI sensors are envisioned to provide adaptive trustworthiness. As these properties can be considered trade-offs, it is possible to establish interactions and negotiations between AI sensors to obtain a balanced level of trust (similar to Chatbot negotiations). Achieving this level of automation, however, requires developing further autonomy in AI sensors.

4.10. Summary and conclusions

In this contribution, we demonstrated a working implementation of SPATIAL, a proof-of-concept system that augments modern applications with capabilities to analyze and quantify trustworthy aspects of AI models. SPATIAL uses a micro-service and API gateway pattern to combine different methods and metrics for analyzing AI algorithms, its data, and the resulting AI model. SPATIAL also implements an AI dashboard to present the results of the analysis to users, introducing human-in-the-loop feedback that can be used to monitor and tune AI model behavior. Through rigorous benchmarks and analyses that consider a real-world industrial application, we demonstrated the performance and scalability of SPATIAL to perform practical AI trustworthiness.

Furthermore, we presented technical development information to support stakeholders with essential details about SPATIAL development. Our application of SPATIAL demonstrates the capabilities of SPATIAL in supporting trustworthy development. Through the configurable workflows SPATIAL can be utilized by users to build models, configure trustworthiness services such as SHAP-based explanations and fairness, simulate attacks, and compare pre- and post-attack behavior. This promotes interpretability, accountability, and resilience of AI systems in practice. Our demonstration of SPATIAL capabilities and practical application further validated the performance, flexibility, and scalability of SPATIAL in supporting trustworthy AI evaluation at scale. Our findings underscore both the potential and current limitations of trust analysis workflows, particularly in terms

of computational cost and complexity, highlighting the need for continued research into optimizing and operationalization trustworthiness tools. Ultimately, SPATIAL illustrates a promising path toward embedding transparent and reliable AI oversight mechanisms into practical applications.

5. ANTIVENOM: SAFEGUARDING AI ROBUSTNESS WITH PROACTIVE HUMAN OVERSIGHT

So far, we have contributed to multiple steps of the machine learning pipeline for constructing AI models. First, we improved data collection using SAFL (Chapter 3), and then we introduced the SPATIAL architecture, which augments modern applications with capabilities to measure and monitor AI trustworthiness (Chapter 4). While AI models enhance the accuracy and functionality of applications, their performance depends on the quality and quantity of training data and the effective execution of each pipeline step. Consequently, AI performance aims to optimize and maximize predictable outcomes, but it can be easily compromised by induced changes, such as attacks, or non-induced issues, like poor data quality, particularly during the deployment phase. Additionally, performance may degrade depending on situational or operational context. Therefore, beyond high performance, AI models must also be robust—able to maintain reliable inference under adversarial and contextual challenges. For human operators, this necessitates monitoring mechanisms that provide rapid insights into potential abnormalities, allowing timely intervention to prevent issues from propagating. While several approaches exist to analyze models and training data, these analyses are often time-consuming and unsuitable for continuous monitoring. To address this, we introduce AntiVenom, a novel and efficient method for detecting poisoning attacks in distributed machine learning systems. AntiVenom leverages device-level performance metrics to identify malicious manipulations, particularly in autonomous systems and applications. We evaluate AntiVenom using a practical use case of drone deployments in urban environments, with results demonstrating its potential to enhance the security, reliability, and robustness of AI systems in real-world settings.

5.1. Introduction

Autonomous drone technology has undergone significant advancements, encompassing autonomous ground vehicles (AGV) such as delivery robots, service robots, and unmanned aerial drones (UAVs) [302, 248]. These technological strides have enabled drones to be effectively utilized for delivering essential goods, including food and medicine, with further applications anticipated in environmental monitoring, urban surveillance, and related fields [161]. These diverse applications are made possible by sophisticated AI models that provide the capabilities for autonomous operations, such as navigation and localization support [164].

Limitations of Autonomous drone deployment: As autonomous drones operate in crowded urban settings, ensuring the robustness, resilience, and consistency of their AI models is crucial for safe and reliable operation. Adversarial attacks pose a significant threat to AI robustness, as they aim to induce unintended con-

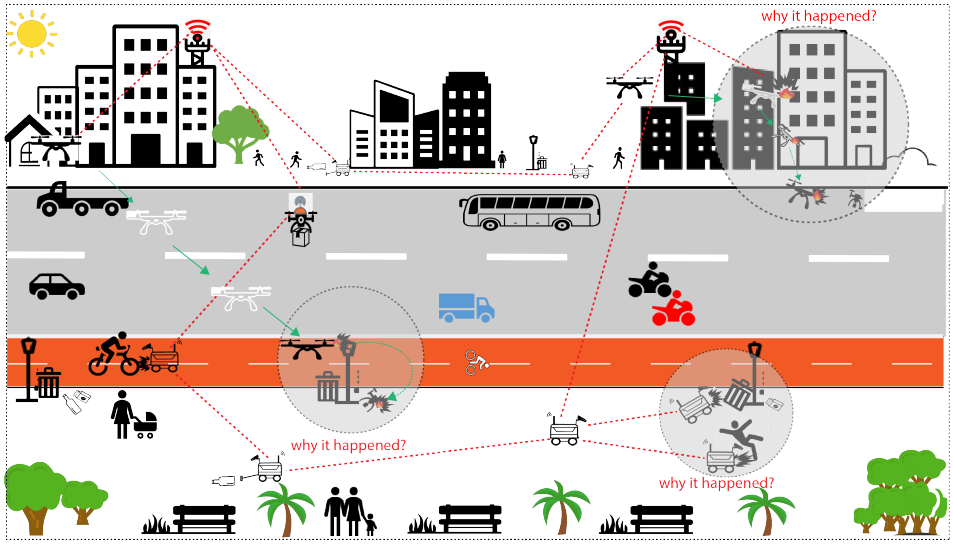


Figure 35: City-scale deployment of autonomous drones and how these can malfunction or misbehave in urban settings.

sequences in the AI models, potentially leading to damage to the city or harm to individuals (Fig. 35), for instance, by colliding with urban infrastructure. Of particular concern are data manipulation attacks, which can be executed by manipulating the environment without any direct access to the autonomous drone. For example, an attacker wearing an adversarial generative patch can deceive the drone, causing it to misinterpret its location and change course. To ensure safe and trustworthy operations of autonomous drones, it is essential to comprehensively understand how these attacks affect autonomous drones and develop effective strategies to mitigate their effects for safe and trustworthy operations.

Contributions: This contribution assesses the threats posed by adversarial attacks on AI robustness and the widespread deployment of autonomous drones in urban environments. The primary objective is to appraise the readiness of these deployments before they become a reality. A benign use case was employed to demonstrate the impact of adversarial data manipulation attacks on AI robustness. Following this, the potential of explainable AI (XAI) methods to enhance AI robustness and identify attacks is examined. XAI methods are techniques used to make the decisions and processes of AI models understandable to humans. Our findings indicate that XAI methods demonstrate high accuracy in identifying basic attacks by examining how the attack significantly alters the data features used for learning. However, identifying the root cause of the issue may be challenging. Additionally, XAI methods require prior knowledge of potential attacks, making it difficult to cover the full spectrum of potential data manipulations. We conclude by reflecting on the results and elaborating on the risks, opportunities, and research challenges that need to be addressed to enhance AI robustness. Potential technologies that could improve the robustness and resilience of AI models are

highlighted. This work lays the groundwork for the practical implementation of city-scale deployments of autonomous drones, envisioning a future where they become a common sight in our everyday environments.

Summary of contributions:

- **Assessment of the threats** that adversarial attacks pose on AI model robustness and large deployment of autonomous drones in urban environments.
- **Examination of the potential of XAI methods** to identify data manipulation attacks on AI models for enhancing AI robustness.
- **Foundation** for exploring potential approaches for detecting attacks and enhancing AI robustness in large scale deployment of autonomous drones.
- **AntiVenom,** a versatile and efficient method for detecting poisoning attacks in large-scale deployment of IoT and distributed machine learning systems. The core idea behind AntiVenom is to monitor device performance metrics, such as CPU frequency, memory usage, and temperature, and to correlate these with variations in distributed training.

5.2. The impact of attacks on autonomous drones

To illustrate the potential vulnerabilities of autonomous drones, we begin by demonstrating how abnormal model behavior and potential disruption of AI performance can be caused by external data poisoning attacks. We also present our threat model and application scenario.

Threat model: A generic threat model is considered, where the AI model in the autonomous drone is targeted to fail by the attacker. The attack can result in specific misbehavior, such as accidents caused by the failure of navigation support to recognize pedestrians or cars. Alternatively, it could be an attack that causes the AI to malfunction, for instance, a sponge attack that drains the autonomous drone's resources or a ransomware attack that prevents normal operations. The motivation for the attack could be to harm the citizens or the city, financial gain, or notoriety.

Application scenario: The use case involves litter recognition with autonomous drones, serving as a representative example of AI-driven operations. In this scenario, thermal images are analyzed in real-time to identify various litter objects and determine their materials. Specifically, the drone analyzes the dissipation of sunlight-induced thermal radiation that is captured by a thermal camera integrated onto the autonomous drone [468]. Attacks against the model can disrupt the operations of the autonomous drones or drain their resources. More serious attacks could target navigation, obstacle detection, or other functions that could directly result in harm to citizens or damage to the environment. While our use case presents a benign example to illustrate the risks of attacks without risking the citizens or the environment, our findings are applicable to any AI applications that rely on computer vision.

Experimental setup: Experiments were conducted using three common litter

objects with different materials: (A) Plastic bottle, (B) Face mask, and (C) Cardboard cup. Video footage of disposed litter was recorded by the autonomous drone, which was then pre-processed and analyzed to identify litter [468]. Data injection attacks were employed to manipulate the input data using blurring and steganography techniques [411, 266]. These attacks, while easy to implement, can have significant impacts, including the installation of backdoor triggers [266] to drain the autonomous drone's resources [392] or create unexpected behaviors. Notably, as we focus on data manipulation, the attacks do not require direct access to the vision system of the drone, as they can manipulate objects in the environment or use additional devices, such as lasers, to alter the data captured and analyzed by the sensors [165].

Results: The thermal dissipation times for the litter objects were measured and are as follows: plastic bottle 62.5*seconds*, cardboard cup 72.5*seconds*, and face mask 82*seconds*. The relative differences align with those reported in [468] for the same materials. However, the absolute values differ due to the varying intensity of the thermal source, the size of the material, and the total exposure time. To analyze poisoning, two levels of poisoning are considered: 10% (low) and 40% (high). Values higher than 40% result in poisoning taking over the model. For blurring, the dissipation times after poisoning are 51.5*seconds* (plastic bottle), 51.1*seconds* (cardboard cup), and 38.4*seconds* (face mask) for 10% poisoning, and 49.3*second*, 22.9*seconds*, and 40.5*seconds* when 40% is poisoned, respectively. The relative differences in the thermal dissipation values thus change significantly, breaking the AI model used for detecting litter materials. The resource drain on the autonomous drone also increased notably, highlighting the potential real-world implications of such attacks. In contrast to blurring, steganography attacks did not influence thermal dissipation times, indicating a varying response of the model to different attack types.

5.3. XAI as model diagnostics

Explainable AI (XAI) methods offer a potential solution for overcoming attacks by offering diagnostics that can identify when an attack occurs. In the following, the potential of different XAI methods to detect targeted poisoning attacks is analyzed, and their benefits and disadvantages are evaluated. The quantifiable values provided by XAI methods in benign cases are examined, and their performance against poisoned data is analyzed. Additionally, the impact of different processing techniques on the behavior of XAI methods when applied to the full image and the processed image with the background removed is investigated to understand how different processing techniques affect the behavior of XAI methods.

Experiment setup: The TrashNet litter classification dataset, comprising $2,527$ litter images [30], was utilized for the experiment. This dataset was chosen due to its large collection of real-world images, enabling the analysis of different environments and contexts for litter classification. A convolutional neural network

(CNN) model was trained because it had demonstrated strong performance with this data [30]. Images were resampled to 300×300 to have consistent input dimensionality. Data augmentation techniques, including horizontal and vertical flipping and rescaling, were applied to the training set, which consisted of $2,276$ images trained with a batch size of 32 for each epoch iteration. The remaining images were used for testing. A collection of 10 poisoned and non-poisoned images was separately considered to illustrate the performance of XAI methods. The experiment was conducted on the Google Colab platform using the latest version of the Keras library (2.8.0) with TensorFlow (v2.8.2).

XAI methods: Three model-agnostic XAI methods were considered for the analysis: LIME [360], SHAP [282], and Occlusion sensitivity. These methods do not rely on CNN gradients; instead, they use perturbations to interpret model behavior and derive versatile (global and local) insights that can be compared across different models. LIME segments images into superpixels, SHAP attributes importance to features, and Occlusion sensitivity [475] uses sensitivity heat maps to observe prediction impact. The selected XAI methods were applied separately to images with the background removed (i.e., only the litter object) and to the original input image. The object extraction process, depicted in Figure 36, involved applying a dynamic patch (determined using object detection) on the image to isolate it. From the final output of the XAI methods, a pixel percentage metric was calculated to capture the importance of a pixel.

Samples and poisoning: Six litter categories were considered: glass, paper, cardboard, trash, metal, and plastic. For poisoning the data, two attacks were considered: blurring and steganography. Blurring can cause autonomous drones to misidentify targets in urban areas, such as crossing signals and pedestrian sidewalks. Steganography introduces extra information in the binary information of the images, which can become resource-intensive for the autonomous drone as more processing power is required to extract relevant information, similar to a sponge attack. The level of poisoning was systematically assessed by poisoning the data in 10% increments from 10% to 40%.

5.4. Results

Model performance under poisoning: The performance of the CNN in classifying litter is 0.7 when no data is poisoned, but this performance gradually decreases as the data is poisoned. After the blurring attack, the model accuracy was decreased to 0.61 (10% poisoned); 0.53 (20% poisoned); 0.53 (30% poisoned), and 0.60 (40% poisoned). Similarly, the steganography attack decreased the model accuracy to 0.52 (10% poisoned); 0.52 (20% poisoned); 0.62 (30% poisoned) and 0.67 (40% poisoned). In both cases, a clear drop in accuracy was observed. Unlike the earlier experiment, the performance drop was higher for data poisoned with steganography than with blurring. This difference in results is simply due to differences in the sensors (RGB vs thermal camera) and the processing pipeline, highlighting how

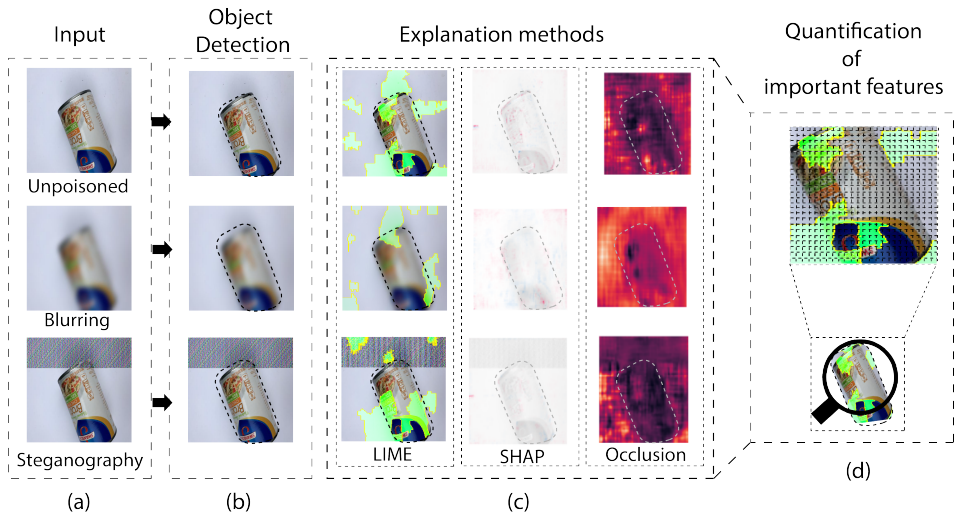


Figure 36: Data sample analysis using different XAI methods, a) Data samples (poisoned and unpoisoned), b) Object detection, c) XAI methods output over samples (LIME, SHAP, and Occlusion sensitivity), and d) Object extraction

	LIME					SHAP					Occlusion Sensitivity				
Poisoning Level	**0%**	**10%**	**20%**	**30%**	**40%**	**0%**	**10%**	**20%**	**30%**	**40%**	**0%**	**10%**	**20%**	**30%**	**40%**
Poisoning type								**Blurring**							
Cardboard	1	0.9	0.9	0.8	0.9	1	0.8	0.8	0.8	0.9	1	0.8	0.9	0.8	0.9
Glass	1	0.7	0.7	0.8	0.6	0.9	0.8	0.8	0.8	0.6	1	0.8	0.8	0.8	0.6
Metal	0.7	0.8	0.7	0.8	0.4	0.6	0.8	0.7	0.8	0.4	0.7	0.7	0.7	0.8	0.4
Paper	0.9	0.9	0.7	0.7	1	0.9	0.9	0.9	0.7	0.9	0.9	0.9	0.9	0.7	1
Plastic	0.8	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	0.7	0.8	0.7	0.7	0.7	0.7
Trash	0.9	0.6	0.6	0.8	0.7	0.9	0.6	0.6	0.8	0.6	0.9	0.6	0.6	0.8	0.7
Average	**0.9**	**0.8**	**0.7**	**0.8**	**0.7**	**0.9**	**0.8**	**0.7**	**0.8**	**0.7**	**0.9**	**0.8**	**0.8**	**0.8**	**0.7**
Poisoning type								**Steganography**							
Cardboard	1	1	1	0.8	0.8	1	1	1	0.8	0.8	1	1	1	0.8	0.8
Glass	1	0.7	0.6	1	1	1	0.6	0.6	1	0.9	1	0.7	0.6	1	0.9
Metal	0.7	0.6	0.6	0.7	0.7	0.7	0.6	0.6	0.7	0.8	0.7	0.6	0.6	0.7	0.8
Paper	0.9	1	1	1	1	0.9	1	1	0.9	0.9	0.9	1	1	0.9	0.9
Plastic	0.8	0.7	0.7	0.7	0.8	0.8	0.7	0.7	0.7	0.8	0.8	0.7	0.7	0.7	0.8
Trash	0.9	0.8	0.6	0.9	1	0.9	0.6	0.6	0.9	0.9	0.9	0.7	0.6	0.9	0.9
Average	**0.9**	**0.8**	**0.8**	**0.9**	**0.9**	**0.9**	**0.8**	**0.8**	**0.8**	**0.9**	**0.9**	**0.8**	**0.8**	**0.8**	**0.9**

Table 14: Individual performance of XAI methods on selected poisoned and unpoisoned samples

the effectiveness of the attack is influenced by the task and the specifics of the AI being used. The performance drop resulting from poisoning depends on how much the attack affects the patterns in the data. In general, as larger amounts of the data become poisoned, the inference process starts to be dominated by the poisoned patterns, while smaller amounts result in distortions that can confuse the model. This pattern is observed with both attacks, with the sole exception being blurring at a 10% rate, as a small level of blurring does not distort the patterns of the litter object sufficiently to impact the AI model.

Analysis of XAI methods: The effectiveness of XAI methods was analyzed considering 10 randomly chosen poisoned samples from each litter category to report the accuracy of estimating the correct class for each sample. Table 14 summarizes the results for the different XAI methods. The effect of poisoning

depends on the litter category and the extent of poisoning. Paper and cardboard objects with regular shapes are the easiest for the XAI methods, whereas classes containing irregular shapes (metal, plastic, trash) exhibit the highest variation in results. Similar to the results for the CNN model, a higher level of poisoning can result in a smaller drop, or in some cases even an increase, in performance. This pattern is more common for steganography, as the poisoned data starts to dominate the inference process once a higher fraction of the data is poisoned. While XAI methods can only help recognize poisoning without directly enhancing the performance of the classifiers, they can indirectly offer insights that can help improve the classifiers. For example, samples that are identified as poisoned can be used to develop data augmentation techniques, which can be incorporated into the model training process to improve the robustness of the classification models. To illustrate this point, blurring is already a commonly used data augmentation technique for improving the training of AI models. From our experiment, it was visually observed that the attack tends to impact the background more than the foreground. Thus, processing techniques that separate the object from the background are likely to improve performance.

Diagnosing objects with XAI: Lastly, the effect of data poisoning on the important features of the object when it is isolated from the background is examined. The coefficient of variation of the poisoned pixels, which depicts the ratio of the standard deviation to the mean, was considered. The higher the value of the coefficient, the higher the dispersion, and thus the better the method is at identifying poisoned data. The results for the 10 test samples of each class are shown in Figure 37. For the blurring attack, the average values of the XAI methods are 0.35 (LIME), 0.17 (SHAP), and 0.3 (Occlusion). For data poisoned with steganography, the corresponding values are 0.22 (LIME), 0.10 (SHAP), and 0.26 (Occlusion). One-way ANOVA between the three XAI methods indicates statistical significance ($F(2,1794)=118.4$, p-value < 0.001), indicating that there are differences in the applicability of the different XAI methods. The higher average values of LIME and Occlusion indicate that they are better at identifying the poisoned data. SHAP performs well for metal objects, which are the most irregular, but struggles with other categories. A one-way ANOVA test was also used to verify that the difference in variation across classes is significant across all XAI methods, poisoning attacks, and levels of poisoning ($F(5,1791)= 14.76$, p-value < 0.001). Across all XAI methods, the coefficients of variation are larger for steganography than for blurring, indicating that XAI methods can also provide clues about the nature of the error. The effect of attack type and data poisoning level was also investigated. A two-way ANOVA test between attack type and data poisoning level indicates a significant effect ($F(1,4)=3.396$, p-value < 0.01), i.e., the coefficients of variation depend not only on the attack type but also on the extent of poisoning. Taken together, these results show that XAI methods help to identify the important features of the image, even after data is poisoned. However, their effectiveness is affected by the object, the type of attack, and the extent of poisoning generated by the attack. In any

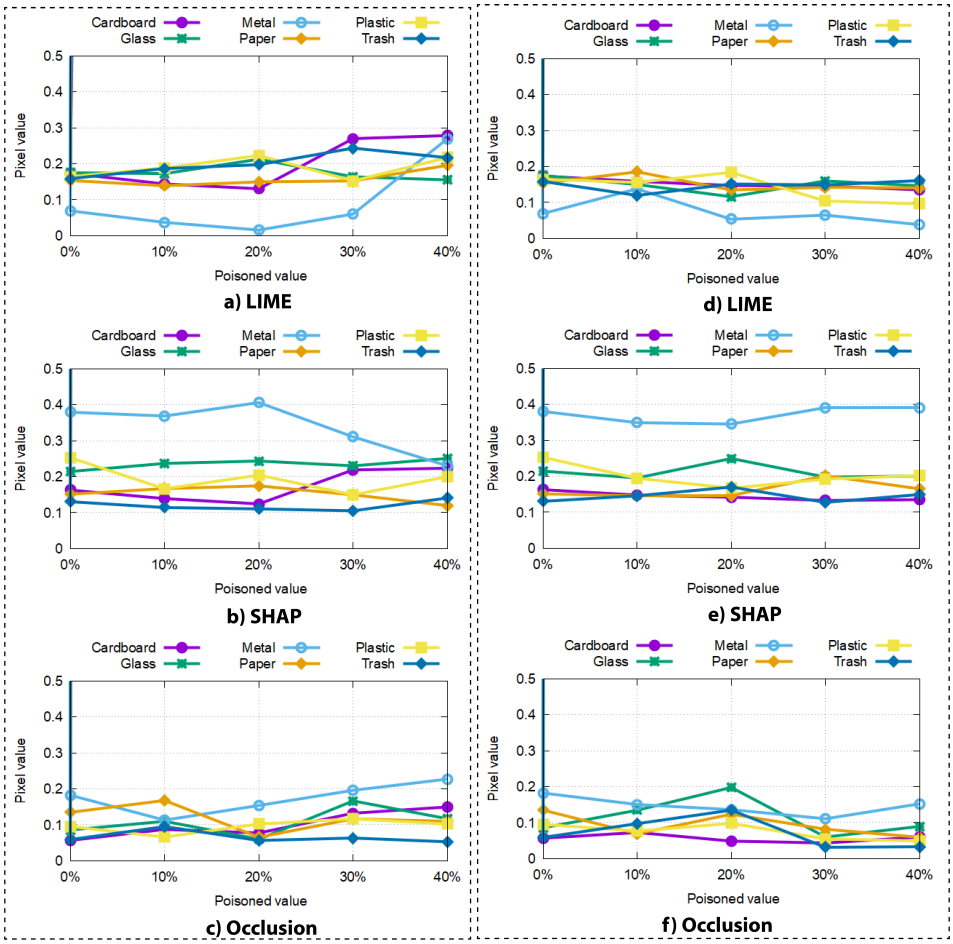


Figure 37: Object analysis with each XAI method as data is poisoned with, (a-c) Blurring and (d-f) Steganography

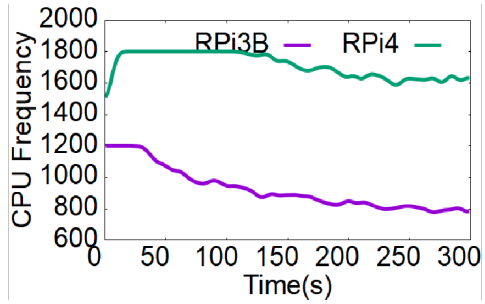


Figure 38: Equal processing load captured using CPU frequency in different devices (RPi4 and RPi3B)

case, even when the objects can be separated and analyzed, an elaborate processing pipeline is required for processing, which drains the resources of the autonomous drones faster and limits their operations.

5.5. AntiVenom: Safeguarding AI robustness against poisoning attacks with proactive human oversight

Distributed machine learning (DML) has emerged as a powerful mechanism for constructing advanced AI models across networked systems. By harnessing insights from multiple devices and federating the training process, paradigms such as split learning and federated learning have demonstrated impressive performance in tasks like environmental monitoring [117] and drone-based aerial surveillance [476]. However, with data and computing being distributed, ensuring the authenticity of devices and data becomes challenging, making the process susceptible to security threats.

Poisoning attacks, adversarial manipulations of the training data or the training process, are among the most significant security threats to DML as even a small amount of poisoned data can substantially degrade a model's performance [376]. As modern AI models grow in size and complexity, and as data becomes both commercially valuable and privacy-sensitive, the transition from centralized to federated training is unlikely to slow. In fact, we can expect an increasing number and variety of devices integrating DML, such as autonomous vehicles, drone swarms, and even home appliances. However, as the scale of deployments and the diversity of devices increase, the threats from poisoning are further exacerbated. For instance, poisoning attacks against autonomous drones can be executed relatively easily by altering the drone's operational environment. AI models may fail to recognize a person or face when data samples are collected from individuals wearing generative adversarial patches [17]. These attacks can disrupt critical functionalities, such as navigation or obstacle detection, potentially draining the drone's battery and leading to unexpected stops or, in the worst-case scenario, crashes that damage urban infrastructure.

To mitigate these risks, it is essential to develop mechanisms that can identify

poisoning attacks and safeguard the devices running DML. Current methods for identifying poisoning are ill-suited for the diversity and scale of emerging DML applications, assuming a sufficiently homogeneous deployment base or at least a sufficient degree of control over the DML process. The most common approach involves running anomaly detection in the cloud and isolating potentially contaminated devices [63, 467, 343]. This requires constant network connectivity, access to relevant model information, and knowledge of the models running on clients, making it impractical for deployments with diverse devices. For example, while XAI techniques can be employed, they are time-consuming and require the device to be recalled to the lab for analysis [326]. Furthermore, while these approaches can identify contaminated devices, they do not prevent them from operating, which means they can still cause damage. Although some solutions analyze local model updates, they require instrumentation or privileged access [94, 486, 317], which is often unavailable due to proprietary protections. Even if such access were granted, deploying these methods would be challenging due to the need to understand variations in models across heterogeneous clients. Consequently, there is an urgent need for a solution that can reliably detect poisoning without requiring information about the model itself or the devices that run it.

We contribute AntiVenom as a reliable and efficient mechanism for identifying poisoning attacks without requiring any access to the server or the model running on the clients. AntiVenom leverages the insight that contaminated data samples significantly impact model performance and internal parameters [459], which, in turn, affect the I/O operations of the device. These operational changes correlate with shifts in I/O performance, providing a basis for detecting poisoning. Building on this premise, AntiVenom monitors execution characteristics on the device and compares changes over time as new data or model updates are incorporated during training. By analyzing variations, both within a single device and across multiple devices, potentially malicious activity can be identified. The resulting devices can either be isolated and their operations stopped, or they can be flagged for further performance analysis if necessary. However, achieving robust detection is non-trivial as the performance characteristics also vary across devices. Figure 38 illustrates this point, showing how the CPU frequency of two different Raspberry PI models (RPi4 and RPi3B) varies with an equal processing load (induced by the *stress-ng* tool). AntiVenom has been designed to overcome this variation by using a novel *Change Point Index (DCPI)* metric to capture variations in metrics across different devices. As AntiVenom does not require any access to the server or the model running on the device, it can operate as a standalone solution that is easy to deploy, unlike existing methods. The monitoring of execution metrics relies on information already collected by the device, ensuring minimal impact on device performance from running AntiVenom. Through our experiments, we demonstrate that AntiVenom is a generic solution compatible with different DML approaches, specifically showing accurate and efficient detection for both Split Learning (SL) and Federated Learning (FL). Our work significantly improves the

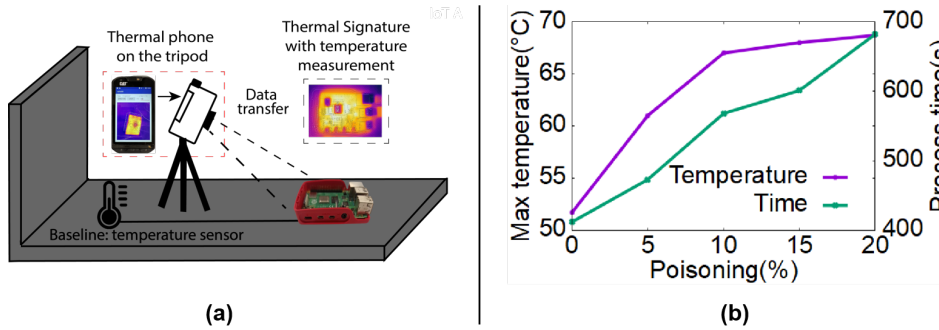


Figure 39: Detection of poisoning attacks: a) Experimental testbed, b) Insights using CPU temperature (RPi4)

safety of distributed machine learning and is particularly important for large-scale deployments harnessing security critical devices, such as drones, autonomous vehicles, and IoT household appliances.

5.6. Motivation

Autonomous drones exemplify scenarios that can benefit from distributed machine learning, and where the consequences of data poisoning can be catastrophic, causing harm to the citizens or the urban infrastructure. To further underscore the necessity for AntiVenom, we next examine the recently proposed ThermAware approach [170], one of the few methods capable of detecting poisoning attacks without requiring instrumentation of the device or access to the model. This technique employs a thermal camera to monitor temperature changes as a means to identify potential poisoning attacks. However, while promising, we demonstrate that this approach lacks scalability, a significant drawback for drone operations, as recalling large numbers of drones for troubleshooting after an attack is both daunting and labor-intensive.

Apparatus: We utilize a Raspberry Pi 4 Model B as our computing unit, which accurately represents the processing capabilities required for drone operations. The RPI features a 64-bit quad-core Cortex-A2 processor and 8GB RAM, providing sufficient processing power for training neural network models. Additionally, we employ a CAT smartphone equipped with FLIR thermal camera sensors to monitor the device's temperature during AI model training; see Figure 39(a). To further validate the temperature readings, we use the open-source tool Chronograf, which allows for external monitoring of the devices.

Experiment: For our experiments, we utilize a traffic sign dataset, detailed in Section 5.10, to train a traffic signal classification model on the RPi. The model undergoes multiple training rounds, with the dataset being incrementally poisoned using occlusion and label flipping attacks. Incremental poisoning varies from 5% to 20% of the dataset, with our main experiments later considering even higher

percentages. During each training round, we record temperature measurements from the device while also collecting data through Chronograf for comparison.
Results: Figure 39(b) illustrates a clear and strong correlation between increased levels of poisoning and elevated temperature changes in the device. These findings corroborate those reported by [170]. Furthermore, the figure highlights that the model requires additional training time, indicating that poisoning not only affects temperature but can also significantly prolong overall training duration. This observation is particularly relevant for the design of AntiVenom, as the thermal characteristics of a device are known to correlate with its execution behaviors [155].
Intuition: Our results suggest that poisoning detection may be feasible through thermal monitoring of device temperature. While this approach is flexible and user-friendly, it proves impractical for large-scale autonomous drone deployments or real-time threat mitigation as it requires access to an external thermal camera. Nevertheless, the experiments also provide evidence that poisoned data elevates temperatures, which reflects heavier processing. Intuitively, this can be understood by the poisoned data resulting in the gradient of the local model increasing, which requires larger updates to the model parameters than a gradient that mostly decreases over time. This finding provides the foundation for AntiVenom, motivating the use of performance characteristics to detect poisoning.

5.7. Attacks on distributed machine learning

The performance of distributed machine learning models can be easily hampered by *adversarial attacks* performed by third parties attempting to alter the model's behavior [39]. Adversarial attacks can be categorized into exploratory, poisoning, and evasion attacks. Exploratory attacks retrieve execution and behavioral information about the model but do not influence its training process [162]. Evasion attacks target the testing phase of the model, where attackers provide carefully crafted malicious inputs to force incorrect inferences, resulting in misclassifications [58, 333]. Finally, *poisoning attacks* compromise the model during the training process by contaminating data samples, causing the model to learn incorrect weights [320]. These attacks can be executed in either a white-box or black-box manner [89, 39]. In white-box attacks, the attacker has full knowledge of the model's internal configurations, such as architecture, parameters, layers, and defenses, allowing them to easily manipulate the model's behavior. In black-box attacks, the attacker has no knowledge of the model's internal workings and relies solely on observing the input-output behavior. Our work specifically focuses on developing a method to detect data poisoning attacks for both distributed machine learning paradigms, split and federated learning. Split learning divides the training of a neural network across multiple devices, sharing only intermediate representations to enhance privacy while collaboratively training a global (or shared) model; whereas Federated learning trains a global model across multiple devices by exchanging model updates instead of raw data, preserving privacy and reducing data transfer.

5.8. AntiVenom design and development

AntiVenom offers a lightweight, efficient, and easy-to-deploy mechanism for identifying poisoning attacks in distributed machine learning (DML) approaches, such as split learning (SL) and federated learning (FL). We next detail the main design goals of AntiVenom, the adversary model we consider, and the overall processing pipeline, followed by AntiVenom.

Design goals: AntiVenom is designed for seamless deployment directly on devices participating in DML without requiring instrumentation at the application level to collect data samples or analyze behavior on the server side. It operates using execution characteristics that can be easily gathered at the operating system level, ensuring hardware-agnostic functionality without accessing the data itself. Execution metrics such as CPU usage, memory consumption, and other performance indicators are inherently influenced by the model's structure and the nature of its training inputs. Changes in the model structure or data directly impact its execution [459], subsequently altering resource usage patterns [184]. More complex models amplify these effects due to interdependencies among layers. These execution perturbations are particularly evident when analyzing resource usage during inference or training tasks. As highlighted in Section 5.6, such perturbations are especially pronounced in CPU frequency and temperature. However, other I/O metrics can also reveal cues about these perturbations [344].

Adversary threat model and assumptions: We consider a data poisoning scenario where the adversary acts as a malicious participant (i.e., attacker) and has control over the input data used to train one or more compromised client devices participating in the DML training process. For instance, autonomous drones collecting data samples using their integrated cameras can be easily compromised without direct access to the captured samples. This could involve physically interfering with a device, such as smearing or obstructing a camera lens, or introducing false data into the environment to mislead the model during training. Changes in the environment, in turn, can be achieved using generative adversarial patches placed by the adversary in the surrounding infrastructure, which can confuse any device relying on computer vision [17], leading to undesired behaviors, such as moving a vehicle or a drone to a specific location or ceasing operations entirely. Other potential attacks include occluding part of the camera's field of view or applying overlapping filters [438], such as luminosity changes. It is also possible for the attacker to use chemical sprays or other compounds to obscure the camera lens or otherwise alter the captured view. While the adversary can poison the local training data of these compromised devices, we assume that they cannot modify the local models after training (i.e., model poisoning is excluded) nor tamper with the performance monitoring of the compromised devices. This assumption can be enforced, for example, by operating model training and performance monitoring within a trusted execution environment [133], which would send local models and performance metrics over a secure channel directly to the server coordinating

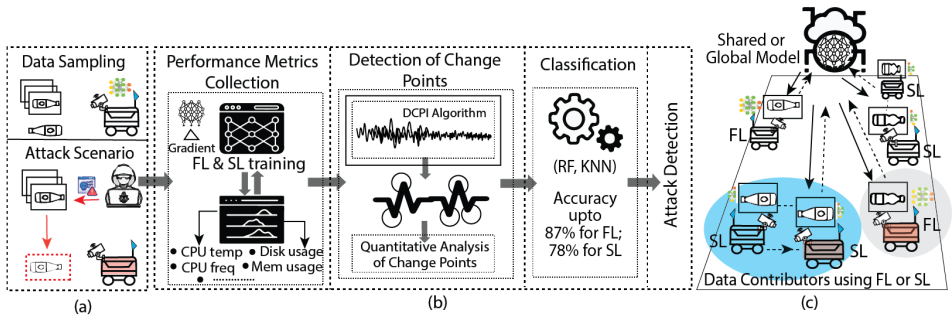


Figure 40: AntiVenom pipeline and deployment; a) Sampling and attack scenario (poisoned device - red autonomous drone), b) Pipeline phases for poisoning detection (including FL and SL training), c) AntiVenom deployment in the wild

DML execution. Thus, the only way for the adversary to poison the local model is by poisoning the data. Similarly, we assume that the FL and SL servers are not compromised and perform secure aggregation [146] to prevent analysis of local model updates, thereby protecting client privacy.

Overview: Figure 40 presents a high-level overview of the AntiVenom pipeline. Performance metrics are treated as time series data, recorded sequentially over time to capture fluctuations in resource utilization at specific intervals. AntiVenom leverages these patterns to identify periods of high resource usage associated with model activity, enabling the detection of anomalies linked to potential poisoning attacks. To achieve this, AntiVenom estimates a *Device Change Point Index (DCPI)* value, which captures trends in changing performance metrics from each client device. These metrics can be analyzed locally, shared with neighboring clients, or shared with a server alongside model parameters to offer information on potential poisoning attacks. As poisoned training data enters the SL or FL process, the server collects the corresponding poisoned parameters and performance metrics. In case the analysis is performed on the server, these values can be analyzed in depth to verify the poisoning attack before taking countermeasures. The diagnostics process of AntiVenom is carried out in three continuous phases, as illustrated in Figure 40(b). These phases work together to ensure robust identification and mitigation of data poisoning threats. Details on the calculation and application of the DCPI value are discussed next.

(i) Performance data collection: To characterize abnormal data contributions from model training, AntiVenom collects performance metrics (CPU utilization, memory usage, and other potential I/O parameters) from individual clients running a model. Since these samples are collected during runtime, AntiVenom does not require the model or source code to be instrumented. The performance metrics are analyzed either locally or sent to the server, where differences are analyzed. This process is repeated every round to ensure that the metrics are always up-to-date. To enhance the robustness and reliability of AntiVenom, samples are taken during periods when the training model has the highest execution priority in the system,

as this reduces variation from other processes.

(ii) Data modeling and analysis: To analyze the time-series performance metric data, denoted as Pf_k, we employ the Pruned Exact Linear Time (PELT) algorithm [236], a robust and computationally efficient technique for detecting change points in sequential data. The PELT algorithm minimizes a cost function that balances the goodness-of-fit of the model with its complexity, incorporating a penalty term, β, to regulate the number of change points identified. This configurable penalty ensures that the algorithm can adapt to the specific dynamics of IoT performance metrics, making it highly suitable for this domain. Key parameters of the algorithm, such as window size, jump length, and cost function, are systematically adjusted to align with the unique characteristics of the data. For this analysis, we adopt an ℓ_2-based cost function (`costL2`) to detect significant changes in the mean and variance, as these variations are crucial indicators of abnormal system behavior. Over time, PELT effectively captures the Change Point Index (CPI). Specifically, the algorithm's output is leveraged to compute $CPI(X(t))$, a normalized metric that quantifies the frequency of change points within the model training duration, $\mathscr{T}$. This metric is instrumental in characterizing system performance variations, as formally expressed in Equation 5.1:

$$CPI(X(t)) = \frac{1}{\mathscr{T}} \sum_{t=1}^{\mathscr{T}} CP(X(t)). \tag{5.1}$$

The second metric is the threshold for differences in change points, $DCPI_{thresh}$, defined in Equation 5.2. Indeed, $DCPI_{thresh}$ is the absolute difference between the CPI of the client's performance metrics in an idle state ($CPI(X(t))idle$) and $CPI(X(t))clean$, the CPI of the client's performance metrics when the device is training a model on clean (untainted) data:

$$DCPI_{thresh} = |CPI(X(t))_{idle} - CPI(X(t))_{clean}|. \tag{5.2}$$

Lastly, to capture an individual device's abnormal performance behavior, we calculate a device-specific difference in change point index, $DCPI_k$, by comparing the change point index of device k during training to the DCPI threshold value, as presented in Equation 5.3:

$$DCPI_k = |CPI(X(t))_k - DCPI_{thresh}|. \tag{5.3}$$

These three metrics characterize the distribution of change points in the data. The higher the frequency of changes in performance, the higher the value of CPI, and vice versa. Respectively, DCPI summarizes the average performance variation when executing FL on the device, while DCPI_k provides a device-specific characterization of performance variations. These computations for DCPI occur on the clients. Note that all of these metrics are straightforward to calculate and

can be implemented in constant run-time, which is essential to ensure AntiVenom has minimal impact on the device.

Algorithm 1 summarizes the overall workflow of AntiVenom for detecting data poisoning attacks. In the algorithm, the central server aggregates the global weights, and the $DCPI_k$ calculated on each client is evaluated to identify performance abnormalities concerning normal device state execution. The estimation of the $DCPI_{thresh}$ is crucial, especially when the device may be engaged in other processing activities that can influence the performance metrics (e.g., CPU utilization). In such scenarios, the absolute difference between $CPI(X(t))idle$ and $CPI(X(t))clean$ nullifies the impact of third-party activity, such as other software processes. $DCPI_k$ serves as the primary metric derived from the performance metrics on a specific device, Pf_k, to detect poisoning attacks.

Algorithm 1 DMLDetect(*ServerExecute*)

INPUT: Clients: k, Fraction of Clients per round: C
OUTPUT: Global weight w_{t+1}

for each round $t = 1$ to ∞ **do**
  $m \leftarrow max(C * K, 1)$
  $S_t \leftarrow$ set of m clients
  for each client $k \in S_t$ in parallel **do**
    $W_{t+1}^k \leftarrow ClientUpdate(k, w_t)$
  end for
  $w_{t+1} \leftarrow \sum_{k=1}^{\mathscr{K}} \frac{n_k}{n} w_{t+1}^k$
end for
for each client k **do**
  Detect CP_k in Pf_k of k
  Calculate the $CPI(X(t))_k$ of as in the Equation 1
  Calculate the $DCPI_{thresh}$ as in the Equation 2
  Calculate DCPI of k, $DCPI_k$ as in the Equation 3
  if $DCPI_k > DCPI_{thresh}$ **then**
    Calculate the amount of Poisoning in k using ML
  end if
end for

(iii) Identification and classification: After extracting the change points from the performance metrics, these data points are input into machine learning models to classify whether the device is poisoned or not. AntiVenom employs classical machine learning algorithms, such as Random Forest (RF) and k-Nearest Neighbors (KNN). Each device and the server have their own models for FL and SL tasks, and the server utilizes a separate model to detect poisoning attacks in the aggregated model.

5.9. The experiments

AntiVenom is evaluated on a physical hardware testbed using Raspberry Pi devices, widely used as core processing units in autonomous systems, service robots, and

drones. For example, the TurtleBot4 delivery robot employs a Raspberry Pi 4B (4 GB), while the indoor drone Romba RPi utilizes a Raspberry Pi Zero 2W. The detection performance of AntiVenom is analyzed by systematically evaluating its response to varying contamination rates from different poisoning attacks. We next describe the testbed and experiments in detail.

Physical testbed, apparatus and deployment: Our testbed comprises a network of 10 Raspberry Pi 4 Model B devices functioning as clients, connected to a laptop server running Ubuntu 22.04 LTS with a 4-core CPU. The devices are linked via LAN cables, maintaining a stable network latency with a Round-Trip Time (RTT) of 102.5 ms. Both the Raspberry Pi devices and the server operate in a temperature-controlled environment set at approximately 23 C, regulated by a thermostat.

Distributed software framework and data collection: We utilized the Flower framework, an open-source Python library supporting both Federated and Split Learning for distributed machine learning. To collect performance metrics from the devices, we employed Telegraf, a performance profiling tool that captures runtime performance indicators. The collected data is stored as time-series data in an InfluxDB database and queried using SQL.

System model and datasets: For our use case, we focus on the practical deployment of autonomous drones. Drone delivery is increasingly becoming a critical real-world application, supported by several manufacturers. Fully autonomous drones are envisioned to continuously learn and collaborate by building machine learning models from data collected across diverse environments, thereby enhancing their operational behavior and robustness over time. For the primary dataset, we use TrashNet for object detection, simulating the data collected by autonomous drones. These drones navigate pedestrian and urban roads, requiring the ability to overcome obstacles and identify surrounding objects. This dataset effectively represents the practical conditions faced by autonomous drones. Additionally, experiments with the Chinese Traffic Sign dataset are conducted in later sections to generalize our findings. Traffic sign data is used by autonomous drones when deciding to cross between roads.

Distribute machine learning model: We divided the dataset into equal-sized subsets, and each subset was randomly assigned to a different client, ensuring each client has balanced data contributions for model training. Based on our primary dataset, a convolutional neural network (CNN) was used, consisting of three convolutional layers (32, 64, and 32 filters, respectively, all with kernel size 3) followed by three fully connected layers (64, 32, and 6 neurons). Each convolutional layer uses a Leaky ReLU activation and max pooling. The flattened output is passed through fully connected layers with dropout (20%) for classification. The model has approximately two million parameters and a computational cost of 200 million FLOPS.

Poisoning attacks: We examine several poisoning attacks applicable to distributed

machine learning through autonomous drone collaboration, including blurring, occlusion, steganography, and label flipping. Blurring and occlusion can be carried out by smearing or physically damaging the drone's camera. Blurring is simulated using the PIL filter, while occlusion is implemented with torchvision's RandomErasing function to obscure parts of the image. Steganography, a more covert and sophisticated attack, embeds malicious patterns or payloads within the images captured by the drone. Unlike direct manipulations, steganographic attacks can incrementally contaminate the model without detection, as the embedded data remains visually imperceptible while subtly affecting the model's learning process over time. Label flipping introduces incorrect labels, causing objects to be misclassified—for example, a traffic sign mistaken for a person—resulting in unsafe or unintended behavior.

Experimental procedure: We evaluate distributed machine learning for autonomous drone collaboration using two paradigms: Federated and Split Learning. For the FL experiments, we simulate varying levels of data poisoning, increasing the poisoned fraction by 5% increments, ranging from 5% to 30%, to analyze the severity of poisoning. Additionally, we vary the number of poisoned clients, testing scenarios with 1, 4, 7, and 10 clients. Given the established effects of poisoning in distributed training [486], we chose this configuration to evaluate poisoning detection under varying distributed conditions. For each condition, a specified fraction of data (5% to 30%) is poisoned using one of the selected attack methods. This process is systematically repeated for all the attack types to ensure a comprehensive evaluation across multiple poisoning scenarios. Each FL training round consists of 5 epochs, and we employ early stopping with a patience of 5 rounds. Similarly, for the SL experiments, we use the same testing conditions and poisoning rates. However, since Split Learning requires the model to be distributed across clients, the split occurs after the convolutional layers and just before the fully connected layers. Each SL training round consists of 1 epoch, and we employ early stopping with a patience of 10 rounds. Furthermore, recognizing that autonomous drones operate in dynamic environments, we analyze the impact of background processes that could interfere with detection and alter the performance of our method. For the detection of poisoning on performance metrics, the Pruned Exact Linear Time (PELT) algorithm was applied to time-series data from client performance metrics. In the FL experiments, the PELT algorithm was configured with a window size of 2 and a skip value of 5 to accommodate the longer training round durations. For the SL experiments, where training rounds were shorter, the algorithm used a window size of 1 and a skip value of 1. The threshold for the DCPI in the detection process was determined by measuring the CPI idle from one hour of idle time-series data and the CPI clean from data collected over 10 training rounds. After analyzing the data, classical machine learning models are developed to identify attacks. Performance metrics from client devices are combined with metadata, such as device IDs, global loss and accuracy, poisoning method, and poisoning rate. The data is labeled as poisoned or not poisoned. Machine learning

models, including Random Forest and K-Nearest Neighbors, are trained on this data, with model validation performed using an 80:20 train-test split.

5.10. Results

Ground truth reference: We first verify that the TrashNet model achieves comparable performance in both federated learning and split learning paradigms. In the FL setup, the model attains an accuracy of 73.6% on our testbed, aligning with results from other benchmarking studies using centralized TrashNet training [119]. For simplicity and to make our results comparable, we also used the same testbed for SL. In the SL case, the model achieves 53.7% accuracy, though performance varies with the distributed configuration. Notably, SL achieves accuracy levels comparable to Federated Learning, reaching 71.8% when configured with a single client and server. This highlights the importance of selecting the right number of clients in SL, as an optimal split is crucial for effective training and avoiding time-outs. Given our primary goal of detecting poisoning in distributed training, the current model performance is sufficient for our purposes.

Impact of poisoning attacks on model performance: Table 15 and 16 present the impact of varying numbers of poisoned devices and poisoning rates on model performance. The results show that an uncontaminated model achieves an accuracy of 73.6% for FL and 53.7% for SL. As the number of poisoned devices increases, the accuracy declines sharply, indicating the sensitivity of model performance to data integrity. Higher poisoning rates further exacerbate this decline and significantly affect both FL and SL models. The Friedman test confirmed that the impact of poisoning rates on model performance is statistically significant for FL ($\chi^2(2) = 20.0$, $W = 0.22$, p $< .05$) and SL ($\chi^2(2) = 9.175$, $W = 0.20$, p $< .05$), highlighting the vulnerability of distributed learning deployments to such attacks. Similarly, we also verified that the number of poisoned devices has a significant effect on model performance for FL ($\chi^2(2) = 70.9$, $W = 0.66$, p $< .05$) and SL ($\chi^2(2) = 28.9$, $W = 0.48$, p $< .05$), demonstrating that even a single contaminated device can degrade overall performance. When comparing FL and SL, we can observe that the performance decline is higher in FL when compared with SL deployments.

Disruption of gradient updates by poisoning: We next demonstrate that performance degradation is linked to significant and quantifiable changes in model gradient information. Figure 41 illustrates these results. The figure shows that increasing poisoning rates cause abrupt changes in the model's training process. Gradient differences between poisoned and unpoisoned models are compared using root mean square error (RMSE) values. Figure 42 shows that highly poisoned training exhibits higher RMSE values. Additionally, Pearson correlation coefficients further verified the significant correlations ($p < .05$) between gradient distortions and poisoning in both convolutional layers *conv* and fully connected layers fc: conv1 ($r = 0.86$), conv2 ($r = 0.85$), conv3 ($r = 0.85$), fc1 ($r = 0.96$), fc2 ($r = 0.96$)

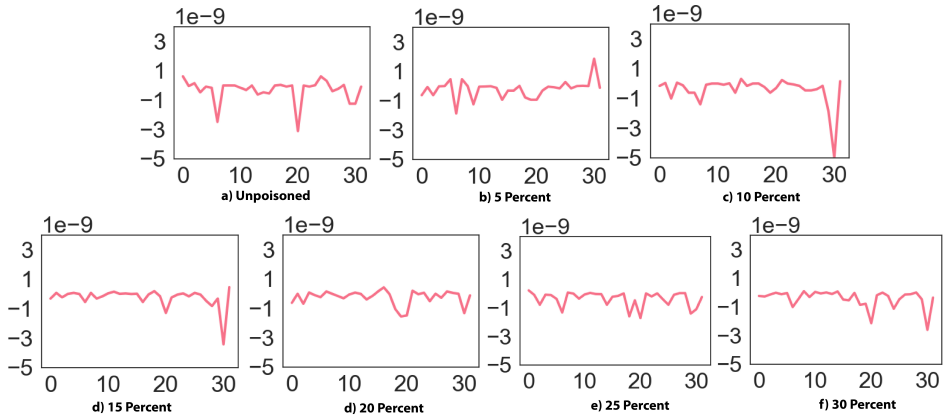


Figure 41: Gradient change of the last layer of the model during training with different levels of poisonings

and fc3 ($r = 0.95$). These results underscore the abrupt distortions in gradients caused by poisoned data during training, revealing that such variations can be directly linked to processing operations and are detectable in real time during the training process.

DCPI characterization performance: Figure 43 presents the average DCPI values for each device, derived from performance metrics collected over 20 rounds with 30% blurring. In this trial, seven devices (id1–id7) were poisoned, while three devices (id8–id10) remained unpoisoned. The results highlight DCPI's effectiveness as a metric for distinguishing poisoned devices from unpoisoned ones. Poisoned devices exhibit significantly higher variations in DCPI compared to unpoisoned devices. A Mann-Whitney U test confirmed these differences ($U = 12532, p < .05$), with variance also supporting this distinction: $4.55e-05$ for poisoned devices versus $3.97e-05$ for unpoisoned ones. Additionally, Wilcoxon-Bonferroni tests found no significant differences among unpoisoned devices ($p > .05$), indicating consistent behavior for normal devices. These findings demonstrate that AntiVenom reliably identifies abnormalities caused by poisoning while distinguishing them from normal operations.

AntiVenom poisoning classification performance in FL: Table 17 presents the classification results of different poisoning attacks using various classical machine learning algorithms. The average accuracy (76%) is high when only DCPI is considered in federated learning FL. This indicates that DCPI can capture the poisoning features accurately. The best performance is achieved for FL with an average accuracy of 80% when considering DCPI, accuracy, CPI, and loss as input features. The results show that classification accuracy improves as additional features are incorporated into the models.

AntiVenom robustness in real-conditions: Since performance metrics can fluctuate due to the processing load on a device, we next demonstrate that AntiVenom can detect poisoning attacks using DCPI even under concurrent background processes.

Poisoning(%)	1/10	4/10	7/10	10/10
un-poisoned	73.6 ± 0.5	73.6 ± 0.5	73.6 ± 0.5	73.6 ± 0.5
Poisoning Type	**Blurring**			
5	72.7 ± 1.9	63.7 ± 0.7	53.7 ± 0.7	46.3 ± 1.1
10	69.4 ± 1.0	65.0 ± 2.7	51.9 ± 1.1	46.3 ± 1.6
15	72.5 ± 1.9	68.1 ± 1.8	53.7 ± 0.7	43.3 ± 1.0
20	70.8 ± 1.6	65.2 ± 1.1	53.2 ± 0.5	46.4 ± 1.3
25	70.7 ± 1.2	62.9 ± 0.5	52.4 ± 1.7	47.2 ± 1.1
30	70.5 ± 1.8	62.5 ± 1.0	51.4 ± 0.3	41.4 ± 0.4
Poisoning Type	**Occlusion**			
5	70.5 ± 1.0	66.9 ± 0.9	60.1 ± 2.9	49.4 ± 1.7
10	71.0 ± 0.4	67.2 ± 1.9	54.2 ± 0.6	46.6 ± 1.0
15	71.2 ± 1.5	66.8 ± 2.0	60.3 ± 0.3	46.4 ± 1.1
20	69.6 ± 0.6	67.2 ± 1.9	56.5 ± 0.9	48.1 ± 0.4
25	71.6 ± 1.1	66.9 ± 3.0	55.0 ± 0.5	44.0 ± 0.4
30	70.1 ± 0.3	66.8 ± 0.3	51.0 ± 0.0	46.1 ± 1.8
Poisoning Type	**Steganography**			
5	71.5 ± 0.7	64.4 ± 0.6	55.9 ± 1.0	51.5 ± 0.7
10	72.9 ± 0.7	64.8 ± 0.2	56.3 ± 0.4	50.5 ± 1.0
15	71.9 ± 0.4	66.5 ± 0.4	53.7 ± 0.7	49.2 ± 1.7
20	72.0 ± 0.6	66.7 ± 1.1	54.5 ± 0.4	49.6 ± 1.7
25	72.1 ± 0.9	65.4 ± 0.6	54.3 ± 0.5	50.7 ± 0.6
30	70.8 ± 0.8	65.6 ± 0.3	53.9 ± 0.4	51.2 ± 1.0
Poisoning Type	**Label** $_{flipping}$			
5	70.2 ± 1.0	62.9 ± 1.5	53.2 ± 0.5	50.3 ± 0.9
10	71.9 ± 1.1	64.2 ± 1.0	55.7 ± 1.6	47.6 ± 0.4
15	72.2 ± 0.4	63.9 ± 1.2	50.7 ± 0.5	45.9 ± 0.4
20	71.2 ± 1.7	63.2 ± 1.3	47.7 ± 1.7	40.3 ± 0.5
25	71.0 ± 1.1	61.9 ± 0.6	45.4 ± 1.3	38.8 ± 1.2
30	70.1 ± 1.2	59.3 ± 0.5	43.1 ± 0.5	36.6 ± 0.3

Table 15: Model performance degradation as incremental data poisoning is introduced by (virtual) individual devices gradually under federated learning (FL)

Poisoning(%)	1/10	4/10	7/10	10/10
un-poisoned	53.7 ± 1.5	53.7 ± 1.5	53.7 ± 1.5	53.7 ± 1.5
Poisoning Type	**Blurring**			
5	48.2	45.9	43.9	35.6
10	48.4	41.0	33.3	33.9
15	50.4	38.6	35.9	37.5
20	49.6	42.4	38.2	32.5
25	51.0	31.6	38.8	34.9
30	46.9	43.3	38.8	34.7
Poisoning Type	**Occlusion**			
5	50.6	46.1	42.5	39.2
10	50.4	43.3	42.0	39.0
15	49.6	46.7	41.4	38.8
20	49.2	41.0	38.0	33.7
25	48.0	41.4	37.8	31.4
30	49.0	38.8	35.1	31.8
Poisoning Type	**Steganography**			
5	56.9	52.5	46.7	41.6
10	52.5	47.6	40.4	41.4
15	53.1	45.3	41.2	42.0
20	44.9	44.3	38.2	41.0
25	49.8	48.8	37.1	44.1
30	52.2	46.1	45.7	41.2
Poisoning Type	**Label flipping**			
5	52.4	43.1	49.8	45.7
10	48.6	45.5	46.9	43.3
15	55.5	47.1	48.8	44.1
20	45.7	46.3	46.9	38.6
25	48.4	42.7	38.8	41.8
30	53.4	48.0	49.8	43.7

Table 16: Model performance degradation as incremental data poisoning is introduced by (virtual) individual devices gradually under split learning (SL)

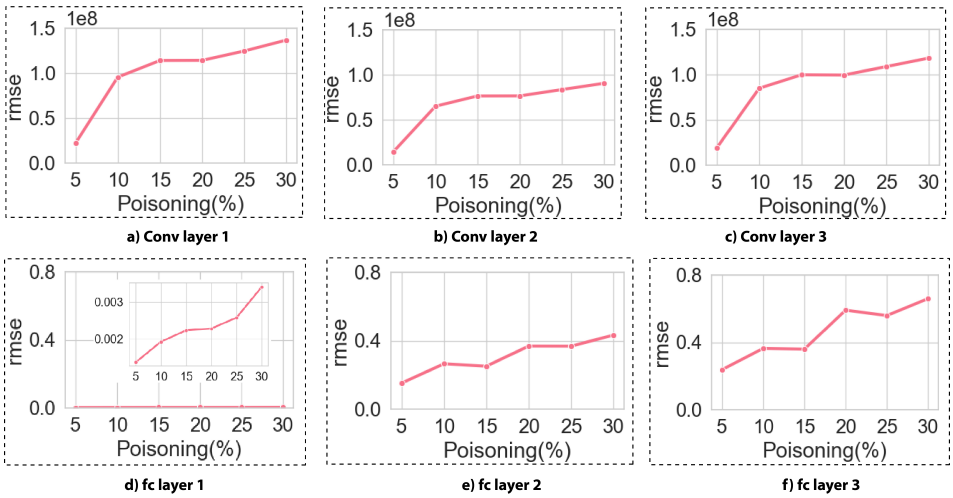


Figure 42: RSME of the gradient changes of the model during training with different levels of poisonings

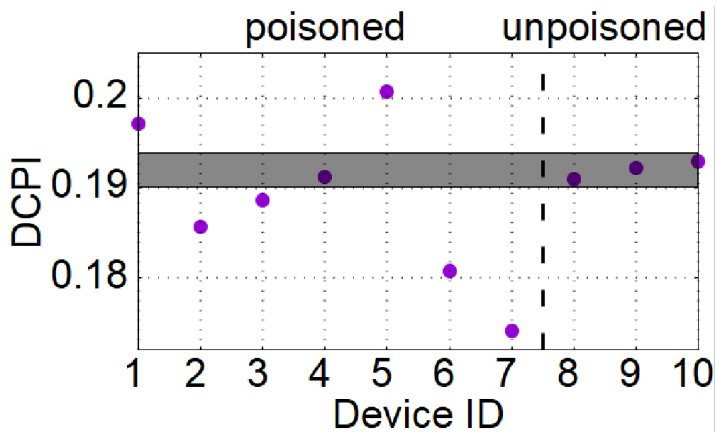
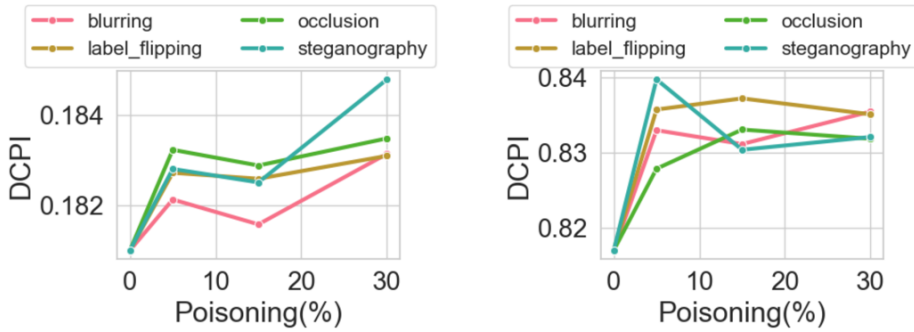


Figure 43: DCPI mean from each device (blurring=30%)

Test (Model data → Predicted)	RF		KNN		Average	
	FL	SL	FL	SL	FL	SL
(DCPI) → P	0.77	0.81	0.75	0.78	0.76	0.80
(DCPI,Accuracy) → P	0.77	0.72	0.70	0.66	0.74	0.69
(DCPI,CPI) → P	0.77	0.81	0.75	0.77	0.76	0.79
(DCPI,CPI,Accuracy) → P	0.83	0.76	0.76	0.71	0.80	0.74
(DCPI,CPI,Accuracy, Loss) → P	0.87	0.78	0.73	0.65	0.80	0.72

Table 17: Binary case "poisoned or not-poisoned", TrashNet classification accuracy (%) for predicting data poisoning attacks (P), Random forest (RF) and K-nearest neighbor (KNN)



(a) FL (b) SL

Figure 44: DCPI with different levels of poisonings

Figure 45(a) shows CPI metrics during idle operation and model training with unpoisoned data, where variations are minimal. This can be used as a reference to assess AntiVenom under different conditions. Figures 45(b-d) illustrate DCPI metrics under different poisoning attacks while background processes are active. We can still observe that the variations caused by poisoning remain detectable during normal operations and varying workloads. Mann-Whitney U tests also showed the significant difference between poisoned and unpoisoned devices: 4 processes ($U = 4.0$, $p < 0.05$), 8 processes ($U = 0.0$, $p < 0.05$), and 12 processes ($U = 4.0$, $p < 0.05$). However, some poisoning techniques do not perform consistently across different background processes, e.g., label flipping. Therefore, although AntiVenom can detect poisoning attacks accurately, the inconsistent pattern can still arise from the system dynamics under different background loads.

AntiVenom extended performance in SL deployments: Figure 44(b) presents the results of DCPI under different poisoning levels. We can still observe that DCPI effectively captures training disruptions induced by varying poisoning rates in SL deployments. Mann-Whitney U tests still confirmed a significant difference in DCPI values between poisoning and non-poisoning ($U = 0.0$, $p < 0.05$). Moreover, the DCPI values increase with a high poisoning level, which is similar to the results in FL deployments. This indicates the good generalization of AntiVenom in various training paradigms. Additionally, the classification performance for detecting poisoning attacks in SL scenarios is still high with an average of 80% when only DCPI is considered, see Table 17. More features cannot improve the accuracy of SL efficiently, which is different from FL paradigms. However, the server-side model in split learning focuses on deeper feature representations from shared activations, which could achieve similar accuracy as federated Learning with fewer features by effectively leveraging the compressed information. Overall, these findings reinforce the effectiveness of DCPI as a reliable indicator for detecting disruptions caused by poisoning across different attack methods and models.

AntiVenom generalizability: As our final step, to demonstrate the versatility of

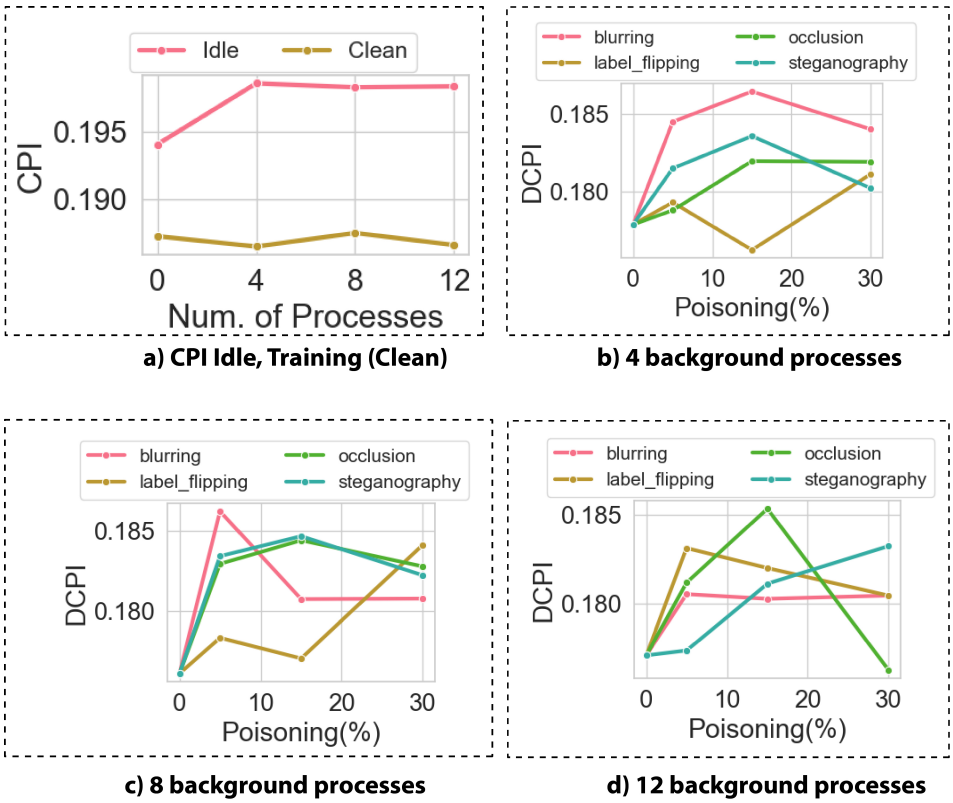


Figure 45: CPI and DCPI with different numbers of background processes

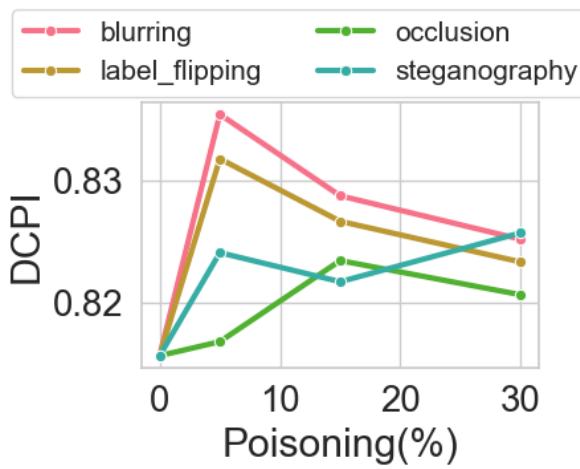


Figure 46: DCPI with different levels of poisonings on the additional dataset for generalization purposes

Test (Model data → Predicted)	RF	KNN	Average
(DCPI) → P	0.81	0.78	0.80
(DCPI,Accuracy) → P	0.69	0.67	0.68
(DCPI,CPI) → P	0.82	0.79	0.81
(DCPI,CPI,Accuracy) → P	0.78	0.74	0.76
(DCPI,CPI,Accuracy, Loss) → P	0.79	0.70	0.75

Table 18: Binary case "poisoned or not-poisoned", Chinese traffic sign dataset classification accuracy (%) for predicting data poisoning attacks (P), Random forest (RF) and K-nearest neighbor (KNN)

AntiVenom, we conduct an additional experiment using the Chinese traffic sign dataset. We use a pre-trained MobileNetV3-Small model for image classification for SL deployments. In the SL experiments, the model is split after the last bottleneck block of MobileNetV3-Small, with input dimensions of $7 \times 7 \times 96$ and output dimensions of $7 \times 7 \times 576$. Figure 46 demonstrates a consistent relationship between DCPI and poisoning rates when applied to a different dataset, confirming that AntiVenom is transferable to other datasets. Mann-Whitney U tests still revealed a significant difference in DCPI values between non-poisoning and poisoning ($U = 0.0, p < .05$). This also confirms that AntiVenom can capture poisoning patterns accurately using DCPI. We also applied the same classification models to predict the poisoning. Table 18 shows the results for all the classifiers. Similar to the TrashNet dataset performance, the accuracy is still high when only using the DCPI feature with an average of 80%, although a DCPI-CPI model can achieve the best performance with an average of 81%. Thus, AntiVenom can be generalized to different training paradigms, poisoning attacks and datasets.

5.11. Reducing failures and improving resilience

Practical drone deployments can be enabled through existing technologies, but the lack of resilience and accountability in AI models prevents these deployments from being robust. Latest regulatory requirements mandate that AI systems must exhibit robustness against various challenges and adversarial conditions. Next, important challenges and state-of-the-art approaches that can improve AI robustness are highlighted.

Immersive evaluation and remediation: Current AI model evaluation and testing processes are not extensive enough to assess performance in all possible situations in which AI models can fail. Evaluating AI models extensively within a short period without delay is a key challenge, especially for autonomous drones operating in urban scenarios. A promising synergy of technologies that can aid in this manner is digital twins and generative AI. Digital twin technology can provide the means to build digital representations of autonomous drones running AI models, whereas generative AI can provide an immersive experience to adjust the situational context of the autonomous drone in a large spectrum of different settings. For instance, the

urban context of the autonomous drone can be dynamically changed to evaluate navigation in different terrains, e.g., rural vs urban areas. Immersive evaluation can also be used to apply different counter-measurements in case autonomous drones are compromised or hampered. For instance, label sanitization methods can be applied to digital autonomous drones to select the most suitable for their physical counterparts.

Data bias and drift detection: Autonomous drones deployed in natural environments engage in ongoing learning processes, enhancing AI resilience via distributed model training like federated learning. However, data collected by these drones may include privacy-sensitive information (e.g., face, speech, or car registration plates). Ensuring privacy necessitates employing techniques like data obfuscation and privacy-preserving methods [411]. As AI models are vulnerable to data poisoning attacks that can disrupt their operations, there is a need for techniques that can quantify model resilience to erroneous updates before they impact the functionality of autonomous drones. A key challenge is the separation between non-intentional malfunctions (e.g., camera failure), intrinsic data biases, and targeted attacks. Existing methods largely target one type of issue (e.g., drift or poisoning) without being able to distinguish between the different causes. Another challenge is to ensure that the methods can operate at different temporal scales, i.e., they can identify problems even when biased or erroneous data is aggregated with valid data and when erroneous data arrives gradually.

Continuous model verification: Trustworthy autonomous drone operations also require easy-to-use solutions for continuous, on-site analysis and verification of decisions, especially for large-scale deployments like in cities [183]. Otherwise, the effort needed for verification can limit the scalability of deployments. Diagnosing AI models often involves accessing the internal structure of the model, which typically requires halting drone operations, modifying source code, and bypassing security features, often necessitating lab work. Explainable AI (XAI) methods provide valuable insights into AI behavior, but their effectiveness is limited by the need for access to data and the model structure, making them less viable as a comprehensive solution. Integrating XAI into security features, such as trusted execution environments (TEEs), could help. However, TEEs currently have significant computational limitations for such practical applications. Additionally, other formal verification methods for AI models encounter challenges, particularly when dealing with non-linear activation functions [449, 454].

Model interpretability and physical drone components: The performance of AI models is intrinsically linked to the resources and components of the autonomous drone [451]. Over time, these components need maintenance or may be upgraded to improve the operations of the drone. These changes can affect the model and result in unexpected behavior. For example, integrating a higher-resolution camera affects the dimensionality of the input data and may capture more detailed images. This can require replacing the model or at least retraining it. In terms of inter-

pretability, this requires linking model diagnostics with the physical components of the autonomous drone and being able to analyze and interpret the effects that individual physical components have on the model's decisions. It is important to note that these changes do not necessarily affect the input data. For example, autonomous drones can operate using different payloads, which affects their weight and resource consumption. This necessitates integrating physical configuration directly into the model diagnosis. This integration is essential for detecting unsafe operations, such as identifying unsafe payloads. Current methods are insufficient as they are unable to link model behavior with the physical characteristics of the operating environment.

Cooperative operations and failures: Effective cooperation between autonomous drones is important for balancing workload between them. This coordination results in dependencies between the AI models deployed on the different autonomous drones, and understanding potential errors or threats requires analyzing the combined logic of all autonomous drones working in tandem, e.g., swarm intelligence [27]. Current XAI and other model diagnostics techniques are tailored to analyzing individual models, and hence, they can only be used if the autonomous drones have a global model that integrates the decision logic of all autonomous drones working together. It is important to note that this task is more complex than analyzing the performance of individual autonomous drones, as attacks or errors can affect only some of the autonomous drones, yet have an influence on all of them by compromising the coordination of the drones through the network [387]. Understanding the effects of target errors on autonomous drones' coordination network and collaboration requires the use of improved diagnostic mechanisms for analyzing network formation groups, individual parts of the network (slices), and the model.

Human oversight and AI: The advanced human-like reasoning of AI models has led to concerns about trust and safety among human operators and developers. Consequently, human oversight has been established as a key requirement for ensuring that AI models are trustworthy. Human-in-the-loop is preferred over fully autonomous approaches, such as automation-in-the-loop agents, because it allows for human expertise and judgment to be applied throughout the life cycle of AI models. This ensures that the AI operates within defined boundaries, delivers desirable outcomes, and behaves as expected after deployment [231]. Moreover, human oversight is critical in practical drone deployments, as failures must be remediated rapidly to avoid halting operations. Human expertise and past experiences can be leveraged to improve AI robustness. Instead of performing a time-consuming remediation analysis, human experts can intuitively select near-optimal solutions to remediate issues. A key challenge, however, is communicating the dissected logic of AI models to human experts. Although regulations like the EU and US AI Acts, as well as China's regulations, require human-in-the-loop involvement in generative AI, best practices for achieving this are not yet

well-defined.

5.12. Discussion

Additional application areas: Our initial work, "*AI model robustness against attack in city-scale autonomous drone deployment*", utilized a litter classification model to demonstrate the impact of data poisoning on AI decision-making. These attacks, however, can be applied across various algorithms and input data types, with implications that vary by context. For instance, random spoofing may exploit different model characteristics when targeting time series data compared to image data.

Model safety assurance: Model diagnostics are essential for companies and organizations deploying autonomous drones in urban environments, particularly for delivery services in smart cities. These diagnostics provide critical safety assurances and performance monitoring capabilities. Similarly, municipalities and government authorities increasingly require the integration of comprehensive diagnostic systems into drones as a prerequisite for issuing operational permits, ensuring public safety and regulatory compliance in shared airspace.

Autonomous drones vulnerabilities: Attacks on AI models are not the sole threats to autonomous drones; hardware and software components are also vulnerable, leading to operational failures and misbehavior. Hence, it is crucial to develop methods that can operate on autonomous drones and distinguish between these vulnerabilities in real-time. For example, a sponge attack can drain the battery of an autonomous drone, while a jamming attack can disrupt its GPS localization. Similarly, software backdoors can expose autonomous drones to manipulation. Consequently, the robust deployment of autonomous drones requires security measures that extend beyond just the AI components, encompassing the entire system architecture from sensors and communication channels to physical hardware protection.

Applicability: AntiVenom is well-suited for real-world scenarios where cooperative and opportunistic devices collaboratively train robust AI models, leveraging performance metrics to detect data poisoning. By supporting cooperative and flexible distributed training, AntiVenom enables poisoning detection regardless of the specific distributed training paradigm that can be adopted based on available devices. While our experiments focus on a practical drone deployment use case, AntiVenom is broadly applicable across other scenarios, such as personalized models in wearables, e.g., Decentralized Federated Learning as a Service [229], IoT sensor nodes, and autonomous vehicles. Although AntiVenom is hardware-agnostic, its effectiveness is highest on single-purpose devices with dedicated functionality. On high-end devices, such as smartphones, separating the performance impact of model training from other activities may require isolating periods when training is the dominant process and analyzing sparser samples. Despite this,

our experiments demonstrate that AntiVenom excels in detecting poisoning attacks, particularly on devices with relatively static tasks, such as autonomous drones, where its performance is most robust.

Other attacks: A major challenge with any security measures is that they become subject to attacks themselves, and AntiVenom is no exception. For instance, data poisoning may be executed gradually or concealed over time to evade detection. AntiVenom can be adapted to these scenarios by incorporating further metrics, e.g., ones that monitor changes in data distribution over time. Beyond attacks on AntiVenom, real-world attacks may exhibit unknown patterns, and there can be so-called zero-day attacks for which no procedures exist. AntiVenom can be extended to these scenarios by replacing the classification models with explainable AI techniques like Shapley values, for identifying parts contributing to deviations in CPU trace. Thus, AntiVenom is not limited to studied attacks but can be extended to other input manipulations.

Practical limitations: In the context of our initial work on city-scale autonomous drone deployment, we explored XAI methods to understand the implications of data manipulation on model behavior. However, beyond the XAI methods we explored, several other approaches exist to dissect the learning and inference processes of AI models. Model-agnostic XAI methods were employed to ensure comparability across different models. Methods such as Guided Backpropagation and Grad-CAM can provide more detailed analyses of AI models, but they are not model-agnostic, requiring adaptation to the used AI model architecture. Notably, for autonomous drones operating in urban areas, XAI methods are critical for ensuring accountable operations. The choice of the XAI method for analyzing AI must meet not only technical criteria but also regulatory and legal requirements set by governmental entities.

Similarly, there are some practical limitations to our work on AntiVenom. Performance metrics are currently collected during the model training process. These metrics are analyzed on-device. However, a significant limitation arises from the security risks associated with transferring these metrics to a centralized server instead. Communication channels can be compromised, rendering the method ineffective. To address this, collaborative and opportunistic computing can be further leveraged, allowing devices involved in model sharing to also analyze the data locally. Multi-party computation techniques offer promising solutions to enhance trustworthiness when sharing data across environments. Additionally, Trusted Execution Environments (TEEs) could be employed to secure both the transfer and analysis of data. However, current TEEs face constraints in handling large data volumes, limiting their scalability [264]. Lastly, hardware failures can produce misleading results, necessitating systems to assess hardware health before collecting performance metrics. Moreover, batch processes can be scheduled by the operating system of the device, such that they are executed during periods of inactivity. When examining performance metrics over time to analyze the behavior

of models, this can become a source of noise, as it can be recognized as a possible exploit or attack. This can be mitigated by bootstrapping the performance analysis by collecting information on common operational routines and by comparing metrics across multiple devices simultaneously. Lastly, while we demonstrated that it is possible to detect poisoning even with background processes, AntiVenom is better suited for periods of low processing activities.

5.13. Related work

Distributed training and autonomous devices: Autonomous devices like drones, vehicles, and service robots can collaboratively improve their operations through DML [413, 211]. DML combines constrained resources of low-power devices to execute models over the underlying (fragmented) computing infrastructures [250]. While FL is widely adopted for such purpose [265], SL has sometimes emerged as a more secure and complementary approach [414, 28]. FL aggregates local updates for collaborative global model training, while SL partitions models between clients and a server to reduce computational overhead. Both paradigms have been successfully employed in autonomous systems to improve operational efficiency and preserve privacy [50, 415]. For instance, FL has been implemented in aerial drones for air quality monitoring [443, 105] and underwater vehicles for navigation [157]. Similarly, SL has been employed in autonomous drones to enable collaborative training while addressing bandwidth and energy constraints [464, 186]. However, both FL and SL face distinct challenges with autonomous systems. Since ground drones are more efficient when compared with other modalities [278], enabling ground drones for the execution of AI models is envisioned to facilitate cooperation between drones [15]. AntiVenom addresses these gaps by monitoring performance execution metrics to detect data poisoning, making it a robust solution for distributed training for autonomous drones

Distributed training and data poisoning: Both FL and SL are susceptible to data poisoning attacks, where malicious participants introduce contaminated data or updates to degrade model performance [485, 417]. In FL, adversaries can manipulate gradient updates to subtly alter global models, causing misclassifications or degraded performance [456]. Similarly, in SL, poisoned activations can compromise training by exploiting intermediate-layer communication [416, 340]. Existing detection methods for FL rely on server-side anomaly detection or client instrumentation, which increases communication overhead and latency [10, 386]. In contrast, SL often requires monitoring intermediate activations, making it vulnerable to adversarial manipulation during data exchange [428, 369]. Recent studies [210, 287] address poisoned updates; they often assume full access to client models, which is impractical in privacy-sensitive settings. AntiVenom, on the other hand, leverages performance metrics, such as CPU frequency and temperature, to detect poisoned devices during training for both FL and SL. This client-agnostic approach avoids intrusive instrumentation, allowing scalable deployments in autonomous

systems.

Autonomous devices meet AI: Recent years have witnessed significant advancements in autonomous devices, driven by breakthroughs in artificial intelligence. Research in robotics has seen remarkable progress in areas like navigation [50], obstacle avoidance [24], and manipulation [60], with AI algorithms enabling service robots to perceive their environment, plan actions, and execute tasks with increasing autonomy [398]. AI-powered path planning algorithms have significantly improved the efficiency and safety of drone operations [363], while deep learning models have enabled ground drones to perceive and interpret their environment with greater accuracy [24]. Building on these advancements, AntiVenom explores the design of a ground multi-drone system that implements two decentralized learning paradigms (federated and split learning) to address the critical issue of poisoning attacks without instrumenting the source code of the devices, ensuring the integrity and reliability of autonomous drone operations.

5.14. Summary and conclusions

First, we evaluated the threat posed by adversarial attacks on AI model robustness in city-scale drone deployments. Robust AI models are crucial for ensuring trustworthy operations of autonomous drones, especially in complex and dynamic environments such as cities. These models must exhibit accuracy and resilience; otherwise, they risk causing harm to citizens or damaging the environment. The current state of practical drone deployments and their trustworthiness was explored, with a focus on the importance of model diagnostics in light of data poisoning attacks that affect the robustness of AI models used in autonomous drones. Notably, these attacks do not require access to the device or AI model, as they are executed solely by manipulating the inputs used. Furthermore, the significance of explainable AI (XAI) methods in identifying data poisoning issues was demonstrated, acknowledging that XAI methods also have their limitations. Based on these findings, challenges and opportunities to enhance AI robustness were presented, alongside the identification of promising technologies and requirements to make autonomous drone operations safer.

Building on the limitations and findings in the previous work *"AI Robustness against attacks in city-scale autonomous drone deployments,"* we further explored methods for detecting poisoning attacks in our work "AntiVenom: Defending Split and Federated Learning Deployments from Poisoning Attacks". We contributed AntiVenom, a lightweight, scalable, and easily deployable method for detecting data poisoning without requiring instrumentation or deep expertise in the underlying AI models. AntiVenom is compatible with various distributed machine learning paradigms, enabling flexible detection of poisoning under different opportunistic conditions. By analyzing performance metrics, such as CPU temperature and frequency, AntiVenom identifies abnormalities and classifies them as benign outliers or potential attacks. Rigorous experiments with FL and SL paradigms

demonstrated that AntiVenom effectively detects poisoned data in the model training process. With an accuracy of 69%–80%, AntiVenom enables the separation of any potential contaminated device contributor from the collaborative network, safeguarding collaborative training environments from poisoning threats.

6. CONCLUSION AND FUTURE DIRECTIONS

6.1. Conclusions

This Chapter represents the culmination of our research effort. We present the conclusion of this thesis in this Chapter. We begin by revisiting the primary research question that has guided this investigation: **How can human oversight approaches be integrated into AI-enabled applications to monitor and contribute to their trustworthiness?** This question motivated our investigation, stemming from recognizing that the rising trend in integrating AI into everyday applications presents profound challenges concerning trust, safety, and reliability. The inherently opaque "black box" nature of many contemporary AI systems, coupled with their documented incidents, necessitates a shift toward development practices prioritizing transparency, accountability, and meaningful human intervention. Throughout this thesis, we have strove to address these challenges by examining practical mechanisms for incorporating effective human oversight across the AI life cycle, thereby establishing a foundation for creating and deploying genuinely trustworthy AI systems that align with human values and societal expectations.

We begin our investigation by examining and addressing the challenges of data availability and quality within the distributed learning paradigm. We proposed Socially Aware Federated Learning (SAFL), a novel approach that leverages social connections to enhance training contributions in a federated learning environment. SAFL addresses the issue of limited or low-quality data, which can significantly impair model performance and reliability, by leveraging social dynamics for task delegation to trusted social contacts and incorporating collaborative incentive mechanisms. Our evaluation demonstrates that this innovative approach not only substantially improves the quantity and quality of data available for model training but also establishes that contributors are willing to participate in collaborative training of AI models.

Having extensively examined the critical role of human-in-the-loop approaches for quality data contribution, we proceed with our investigation by examining system architectures, the backbone of modern AI applications, to understand how trustworthiness metrics and human oversight mechanisms can be integrated into a modern application and system. To this end, we develop the SPATIAL architecture, a proof-of-concept architecture that augments existing architectural paradigms with robust monitoring capabilities and embeds trustworthiness metrics within AI-based applications. Through the SPATIAL dashboard, measures and analyses characterizing these metrics are effectively communicated to human operators, providing quantifiable insights into the trustworthiness of AI-based applications and systems. This interface enables human-in-loop implementation by facilitating human monitoring of AI systems and timely and informed human intervention.

Since data serves as the cornerstone for AI models, we also examined the impact of data poisoning attacks on AI systems to address critical issues of AI

robustness, particularly within the context of sophisticated adversarial attacks. The research culminated in the development of AntiVenom, a novel technique for detecting anomalies in distributed AI deployments. AntiVenom leverages device-level performance metrics to identify irregularities indicative of poisoning attacks, offering a proactive and efficient solution for safeguarding autonomous systems. This contribution is particularly relevant in safety-critical applications, such as autonomous drones, where the consequences of AI failures can be severe.

In summary, this thesis represents a comprehensive effort to advance the field of trustworthy AI by providing practical solutions for integrating human oversight into AI systems. The contributions of this research, including the SPATIAL architecture, Socially Aware Federated Learning, and AntiVenom, offer valuable tools and methodologies for developing AI applications that are intelligent but also reliable, safe, and aligned with human values. The findings of this research have significant implications for the future of AI development and deployment, paving the way for a more trustworthy approach with human oversight.

6.2. Limitations and reflections

This thesis presents three contributions that enhance human oversight at different stages of the machine learning pipeline. These mechanisms have been shown to improve both human contributions and responses to AI models. Human-in-the-loop mechanisms can vary widely, relying on different senses and input channels, and our contributions demonstrate how humans can be effectively leveraged to improve and monitor AI behavior. While trustworthy AI emphasizes human oversight, it also encompasses other properties, such as fairness, robustness, and transparency, that contribute to trustworthiness. Our work addresses some of these properties through the design of human oversight mechanisms. Still, it does not cover all aspects, nor does it explore the trade-offs between different trust properties and human-in-the-loop considerations. It is important to note that trustworthy AI is an evolving field, involving developers and AI scientists and insights from multiple disciplines, reflecting the broader societal, ethical, and technical dimensions of AI deployment.

Besides this,in this thesis, we primarily investigated the trustworthy analysis of classical machine learning and deep learning models built through the standard machine learning pipeline. The trustworthy regulations applied to these pipelines draw on the broader principles of trustworthy computing, addressing aspects such as reliability, robustness, and fairness. However, with the advent of LLMs, generative AI, and foundation models, the scope of trustworthiness has expanded significantly. These models introduce new dimensions, including richer user interactions and additional attributes such as reasoning and hallucination, which must be systematically assessed. While the contributions of this thesis provide a solid foundation that can be extended to these emerging paradigms, further research is needed to adapt and refine the proposed approaches for these more complex scenarios.

6.3. Future directions

Sustainable AI: In addition to trustworthiness, a major requirement in AI systems lies in the significant energy consumption required by the underlying hardware during model training. As the demand for trustworthy AI grows, it is crucial that AI models maintain their reliability and minimize their computational and energy consumption. This requirement can be addressed in two main ways: through innovative breakthroughs in computing technology, such as quantum computing, or by optimizing existing methods to maximize the efficiency of currently available hardware. Recent advancements in LLMs have already begun to mitigate the high computational costs and energy demands traditionally associated with these systems. For example, models like DeepSeek have been designed to run efficiently on constrained devices, demonstrating how optimization can reduce the environmental footprint of deploying AI models [53]. These advancements showcase the potential for improving the sustainability of AI systems while still maintaining their performance and scalability.

Distillation and autonomous agents: Distillation methods have shown the potential to improve the inference process of AI gradually. This suggests that distillation can generate clear, rich, comprehensive datasets for more accurate AI training. Simultaneously, autonomous agents capable of learning and interacting with each other are emerging, paving the way for autonomous training. This could significantly enhance AI inference capabilities in the near future, potentially leading to exponential growth in artificial intelligence. However, this scenario also implies that the need for human oversight may diminish, necessitating the implementation of additional safety mechanisms to prevent uncontrolled AI learning. In other words, while autonomous advances in learning may accelerate, they must be verified by humans to ensure AI trustworthiness. Furthermore, whether trustworthy AI can be inherently passed to distillation models remains unclear. Despite these models being built from trustworthy AI systems, it is uncertain whether new distillation-based AI models are inherently trustworthy by default.

AI regulations- A barrier to innovation or a key to competitiveness: AI regulations have subjected the development of AI models to rigorous scrutiny, requiring a more detailed examination of decision-making processes. Achieving trustworthiness in AI may necessitate a shift from conventional methods, creating new approaches that ensure thorough analysis of AI decisions. While this could offer a competitive advantage, as models developed with these mechanisms would be both accurate and trustworthy, developing new methods to challenge existing ones may take significant time in practice. Building effective AI systems has historically required years, if not decades, of research. This could hinder AI innovation, as less restrictive approaches might already capitalize on AI's potential. From an economic perspective, countries that do not impose stringent AI regulations may gain

a competitive edge, making it difficult for others to catch up. While EU regulations emphasize that they apply only to end-products and not research and development, early adoption of AI technologies might become a barrier for economies striving to foster local dominance in AI solutions.

Harmonization of governance frameworks and regulation: The AI global governance and regulation landscape is becoming complex as many jurisdictions develop various principles, frameworks, and expectations. Consequently, coordinating operations and compliance across jurisdictions is a significant challenge. In addition, disparity in governance frameworks and non-alignment exist, which could discourage collaboration across countries and regions. A harmonized approach to AI governance that recognizes regional values and priorities can facilitate cross-border AI innovations, development, collaborations, and easy navigation of compliance operations for large corporations developing AI in different regions.

Trustworthy properties measurement standardization: The standardization of methods and metrics for quantifying trustworthiness can formalize the measurements of the various properties of trustworthiness, as is the case in trustworthy computing. This pursuit is a critical direction for advancing research in trustworthy AI because developing generalized metrics and evaluation methods provides the basis for comparative analysis of AI systems across diverse contexts. Such standardization could transform trustworthiness frameworks from mere compliance obligations into evidence-based assessment standards applicable to any AI system. Currently, methods for evaluating various trustworthiness properties are diverse, each with specific considerations and contexts dependent, making validation across domains challenging due to contextual misalignments. By standardizing these methods and measures, we can facilitate multi-dimensional validation of results and enable consistent interpretation of trustworthiness properties.

Addressing paradoxical performance of AI systems: Artificial intelligence models are generally efficient with complex tasks. However, LLMs fumble on seemingly simple ones. This performance paradox presents a significant gap with trust implications, as there is skepticism about the reliability of these models. Users are naturally perplexed about how LLM demonstrates proficiency in providing correct outcomes on complex questions and yet underperforms on basic reasoning questions (common sense questions) or factual accuracy. This inconsistency undermines confidence in LLMs and affects user trust. Addressing this paradox requires novel approaches that align model capabilities with human expectations, such that mastery of complex tasks should inherently encompass proficiency in simpler ones. This alignment is essential for developing AI systems that users can trust consistently across varying contexts and complexity levels.

BIBLIOGRAPHY

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. "Deep learning with differential privacy". In: *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 2016, pp. 308–318.

[2] Waseem Abbas, Wasi Haider Butt, et al. "Systematic Literature Review on Requirement Management Tools". In: *2022 International Conference on Emerging Trends in Smart Technologies (ICETST)*. IEEE. 2022, pp. 1–6.

[3] Naoki Abe, Bianca Zadrozny, and John Langford. "Outlier detection by active learning". In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2006, pp. 504–509.

[4] Sophia Abraham, Zachariah Carmichael, Sreya Banerjee, Rosaura Vidal-Mata, Ankit Agrawal, Md Nafee Al Islam, Walter Scheirer, and Jane Cleland-Huang. "Adaptive autonomy in human-on-the-loop vision-based robotics systems". In: *2021 IEEE/ACM 1st workshop on AI engineering-software engineering for AI (WAIN)*. IEEE. 2021, pp. 113–120.

[5] Artificial Intelligence Act. *Annex III – High-Risk AI Systems*. Accessed: 2025-02-19. 2024.
URL: `https://artificialintelligenceact.eu/annex/3/`

[6] Artificial Intelligence Act. *Article 50 – Measures in Support of Innovation*. Accessed: 2025-02-19. 2024.
URL: `https://artificialintelligenceact.eu/article/50/`.

[7] Artificial Intelligence Act. *Article 6 – Classification of High-Risk AI Systems*. Accessed: 2025-02-19. 2024.
URL: `https://artificialintelligenceact.eu/article/6/`.

[8] Amina Adadi and Mohammed Berrada. "Peeking inside the black-box: a survey on explainable artificial intelligence (XAI)". In: *IEEE access* 6 (2018), pp. 52138–52160.

[9] Bishwo Adhikari and Heikki Huttunen. "Iterative bounding box annotation for object detection". In: *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE. 2021, pp. 4040–4046.

[10] Shaashwat Agrawal, Sagnik Sarkar, Ons Aouedi, Gokul Yenduri, Kandaraj Piamrat, Mamoun Alazab, Sweta Bhattacharya, Praveen Kumar Reddy Maddikunta, and Thippa Reddy Gadekallu. "Federated learning for intrusion detection system: Concepts, challenges and future directions". In: *Computer Communications* 195 (2022), pp. 346–361.

[11] Khlood Ahmad et al. "What's up with requirements engineering for artificial intelligence systems?" In: *IEEE International Requirements Engineering Conference (RE)*. 2021, pp. 1–12.

[12] NIST AI. *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. 2023.

[13] Elie Alhajjar, Paul Maxwell, and Nathaniel Bastian. “Adversarial machine learning in network intrusion detection systems”. In: *Expert Systems with Applications* 186 (2021), p. 115782.

[14] Abdulrahman Alhazmi and Nalin AG Arachchilage. “A serious game design framework for software developers to put GDPR into practice”. In: *Proceedings of the 16th International Conference on Availability, Reliability and Security*. 2021, pp. 1–6.

[15] Ali Aliedani and Seng W Loke. “Cooperative autonomous vehicles: An investigation of the drop-off problem”. In: *IEEE T-IV* 3.3 (2018), pp. 310–316.

[16] Osianoh Glenn Aliu, Ali Imran, Muhammad Ali Imran, and Barry Evans. “A survey of self organisation in future cellular networks”. In: *IEEE Communications Surveys & Tutorials* 15.1 (2012), pp. 336–361.

[17] Yigit Alparslan et al. “Adversarial attacks on convolutional neural networks in facial recognition domain”. In: *arXiv preprint arXiv:2001.11137* (2020).

[18] Belal Alshaqaqi, Abdullah Salem Baquhaizel, Mohamed El Amine Ouis, Meriem Boumehed, Abdelaziz Ouamri, and Mokhtar Keche. “Driver drowsiness detection system”. In: *2013 8th international workshop on systems, signal processing and their applications (WoSSPA)*. IEEE. 2013, pp. 151–155.

[19] Emad Alsuwat, Hatim Alsuwat, John Rose, Marco Valtorta, and Csilla Farkas. “Detecting adversarial attacks in the context of bayesian networks”. In: *Data and Applications Security and Privacy XXXIII: 33rd Annual IFIP WG 11.3 Conference, DBSec 2019, Charleston, SC, USA, July 15–17, 2019, Proceedings 33*. Springer. 2019, pp. 3–22.

[20] Emad Alsuwat, Hatim Alsuwat, Marco Valtorta, and Csilla Farkas. “Adversarial data poisoning attacks against the PC learning algorithm”. In: *International Journal of General Systems* 49.1 (2020), pp. 3–31.

[21] Yasmeen Alufaisan, Murat Kantarcioglu, and Yan Zhou. “Robust transparency against model inversion attacks”. In: *IEEE transactions on dependable and secure computing* 18.5 (2020), pp. 2061–2073.

[22] AJ Alvero, Noah Arthurs, Anthony Lising Antonio, Benjamin W Domingue, Ben Gebre-Medhin, Sonia Giebel, and Mitchell L Stevens. “AI and holistic review: informing human reading in college admissions”. In: *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 2020, pp. 200–206.

[23] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N Bennett, Kori Inkpen, et al. “Guidelines for human-AI interaction”. In: *Proceedings of the 2019 chi conference on human factors in computing systems*. 2019, pp. 1–13.

[24] Umberto Andriolo et al. “Mapping marine litter on coastal dunes with unmanned aerial systems: A showcase on the Atlantic Coast”. In: *Science of the Total Environment* 736 (2020), p. 139632.

[25] Plamen P Angelov et al. “Explainable artificial intelligence: an analytical review”. In: *Data Mining and Knowledge Discovery* 11.5 (2021).

[26] Marco Anisetti, Claudio A Ardagna, Nicola Bena, and Ernesto Damiani. “Rethinking Certification for Trustworthy Machine Learning-Based Applications”. In: *IEEE Internet Computing* (2023).

[27] Mathias Anneken, Manjunatha Veerappa, and Nadia Burkart. “Anomaly Detection and XAI Concepts in Swarm Intelligence”. In: (2021).

[28] Huiqing Ao, Hui Tian, and Wanli Ni. “Federated split learning for distributed intelligence with resource-constrained devices”. In: *2024 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE. 2024, pp. 798–803.

[29] Zinat Ara, Hossein Salemi, Sungsoo Ray Hong, Yasas Senarath, Steve Peterson, Amanda Lee Hughes, and Hemant Purohit. “Closing the Knowledge Gap in Designing Data Annotation Interfaces for AI-powered Disaster Management Analytic Systems”. In: *Proceedings of the 29th International Conference on Intelligent User Interfaces*. 2024, pp. 405–418.

[30] Rahmi Arda Aral et al. “Classification of trashnet dataset based on deep learning models”. In: *IEEE BigData*. IEEE. 2018, pp. 2058–2062.

[31] Gonzalo Martínez Ruiz de Arcaute, José Alberto Hernández, and Pedro Reviriego. “Assessing the impact of membership inference attacks on classical machine learning algorithms”. In: *2022 18th International Conference on the Design of Reliable Communication Networks (DRCN)*. IEEE. 2022, pp. 1–4.

[32] Dustin L Arendt, Nasheen Nur, Zhuanyi Huang, Gabriel Fair, and Wenwen Dou. “Parallel embeddings: a visualization technique for contrasting learned representations”. In: *Proceedings of the 25th International Conference on Intelligent User Interfaces*. 2020, pp. 259–274.

[33] Matthew Arnold, Rachel KE Bellamy, Michael Hind, Stephanie Houde, Sameep Mehta, Aleksandra Mojsilović, Ravi Nair, K Natesan Ramamurthy, Alexandra Olteanu, David Piorkowski, et al. “FactSheets: Increasing trust in AI services through supplier’s declarations of conformity”. In: *IBM Journal of Research and Development* 63.4/5 (2019), pp. 6–1.

[34] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI”. In: *Information fusion* 58 (2020), pp. 82–115.

[35] Maria De-Arteaga, Riccardo Fogliato, and Alexandra Chouldechova. “A case for humans-in-the-loop: Decisions in the presence of erroneous algo-

rithmic scores". In: *Proceedings of the 2020 CHI conference on human factors in computing systems*. 2020, pp. 1–12.

[36] Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. "Bias in bios: A case study of semantic representation bias in a high-stakes setting". In: *proceedings of the Conference on Fairness, Accountability, and Transparency*. 2019, pp. 120–128.

[37] Vijay Arya, Rachel KE Bellamy, Pin-Yu Chen, Amit Dhurandhar, Michael Hind, Samuel C Hoffman, Stephanie Houde, Q Vera Liao, Ronny Luss, Aleksandra Mojsilović, et al. "AI Explainability 360 Toolkit". In: *Proceedings of the 3rd ACM India Joint International Conference on Data Science & Management of Data (8th ACM IKDD CODS & 26th COMAD)*. 2021, pp. 376–379.

[38] Aleksandre Asatiani, Pekka Malo, Per Rådberg Nagbøl, Esko Penttinen, Tapani Rinta-Kahila, and Antti Salovaara. "Challenges of explaining the behavior of black-box AI systems". In: *MIS Quarterly Executive* 19.4 (2020), pp. 259–278.

[39] Felix Assion et al. "The attack generator: A systematic approach towards constructing adversarial attacks". In: *Proceedings of the IEEE/CVF Workshops*. 2019, pp. 1370–1379.

[40] Daniel Atherton, Reva Schwartz, Peter C. Fontana, and Patrick Hall. *The Language of Trustworthy AI: An In-Depth Glossary of Terms*. Tech. rep. NIST Artificial Intelligence AI 100-3. Gaithersburg, MD: National Institute of Standards and Technology, 2023. DOI: 10.6028/NIST.AI.100-3. URL: https://doi.org/10.6028/NIST.AI.100-3.

[41] Jackie Ayoub, Lilit Avetisian, X Jessie Yang, and Feng Zhou. "Real-time trust prediction in conditionally automated driving using physiological measures". In: *IEEE Transactions on Intelligent Transportation Systems* 24.12 (2023), pp. 14642–14650.

[42] Ms Aayushi Bansal, Dr Rewa Sharma, and Dr Mamta Kathuria. "A systematic review on data scarcity problem in deep learning: solution and applications". In: *ACM Computing Surveys (Csur)* 54.10s (2022), pp. 1–29.

[43] Nathan Bartley and Kristina Lerman. "RTs!= Endorsements: Rethinking Exposure Fairness on Social Media Platforms". In: *arXiv preprint arXiv:2409.13237* (2024).

[44] Robert Baumgartner, Wolfgang Gatterbauer, and Georg Gottlob. "Web Data Extraction System." In: *Encyclopedia of database systems* 1 (2009).

[45] Firas Bayram and Bestoun S Ahmed. "Towards Trustworthy Machine Learning in Production: An Overview of the Robustness in MLOps Approach". In: *ACM Computing Surveys* 57.5 (2025), pp. 1–35.

[46] BBC News. “A-levels and GCSEs: How did the exam algorithm work?” In: *BBC News* (2020). Accessed: 2025-04-08. URL: https://www.bbc.com/news/education-53787203.

[47] Joseph Beck, Mia Stern, and Erik Haugsjaa. “Applications of AI in Education”. In: *XRDS: Crossroads, The ACM Magazine for Students* 3.1 (1996), pp. 11–15.

[48] Timothy Bella. *Backup driver in deadly Uber self-driving car crash pleads guilty*. Accessed: 2025-06-18. 2023. URL: https://www.washingtonpost.com/nation/2023/07/31/uber-self-driving-death-guilty/.

[49] Marianne Bertrand and Sendhil Mullainathan. “Are Emily and Greg more employable than Lakisha and Jamal? A field experiment on labor market discrimination”. In: *American economic review* 94.4 (2004), pp. 991–1013.

[50] Juan A. Besada et al. “Drones-as-a-service: A management architecture to provide mission planning, resource brokerage and operation support for fleets of drones”. In: *IEEE PerCom Workshops 2019*, pp. 931–936.

[51] Arjun Nitin Bhagoji, Daniel Cullina, Chawin Sitawarin, and Prateek Mittal. “Enhancing robustness of machine learning systems via data transformations”. In: *2018 52nd Annual Conference on Information Sciences and Systems (CISS)*. IEEE. 2018, pp. 1–5.

[52] Umang Bhatt, McKane Andrus, Adrian Weller, and Alice Xiang. “Machine learning explainability for external stakeholders”. In: *arXiv preprint arXiv:2007.05408* (2020).

[53] Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. “DeepSeek LLM: Scaling Open-Source Language Models with Longtermism”. In: *arXiv e-prints* (2024), arXiv–2401.

[54] Joseph R Biden. “Executive order on the safe, secure, and trustworthy development and use of artificial intelligence”. In: (2023).

[55] Battista Biggio, Igino Corona, Giorgio Fumera, Giorgio Giacinto, and Fabio Roli. “Bagging classifiers for fighting poisoning attacks in adversarial classification tasks”. In: *Multiple Classifier Systems: 10th International Workshop, MCS 2011, Naples, Italy, June 15-17, 2011. Proceedings 10*. Springer. 2011, pp. 350–359.

[56] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. “Evasion attacks against machine learning at test time”. In: *Joint European conference on machine learning and knowledge discovery in databases*. Springer. 2013, pp. 387–402.

[57] Battista Biggio, Igino Corona, Blaine Nelson, Benjamin IP Rubinstein, Davide Maiorca, Giorgio Fumera, Giorgio Giacinto, and Fabio Roli. “Security evaluation of support vector machines in adversarial environments”. In: *Support vector machines applications*. Springer, 2014, pp. 105–153.

[58] Battista Biggio, Blaine Nelson, and Pavel Laskov. “Poisoning attacks against support vector machines”. In: *arXiv preprint arXiv:1206.6389* (2012).

[59] Battista Biggio and Fabio Roli. “Wild patterns: Ten years after the rise of adversarial machine learning”. In: *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. 2018, pp. 2154–2156.

[60] Aude Billard and Danica Kragic. “Trends and challenges in robot manipulation”. In: *Science* 364.6446 (2019), eaat8414.

[61] Volodymyr Biryuk. *OpenReq*. `https://https://github.com/orgs/OpenReqEU/repositories?page=2&type=all`. 2019.

[62] Emily Black, Samuel Yeom, and Matt Fredrikson. “Fliptest: fairness testing via optimal transport”. In: *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 2020, pp. 111–121.

[63] Peva Blanchard et al. “Machine learning with adversaries: Byzantine tolerant gradient descent”. In: *31st NeurIPS*. 2017, pp. 118–128.

[64] Michell Boerger et al. *Deliverable (D3.4) - performance evaluation in controlled environments and guidelines to build the pilot studies in real testbeds*. Tech. rep. EU SPATIAL project, 2024. URL: `https://spatial-h2020.eu/`.

[65] Iris Bohnet. “Trust in experiments”. In: *Behavioural and Experimental Economics*. Springer, 2010, pp. 253–257.

[66] Tolga Bolukbasi, 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”. In: *Advances in neural information processing systems* 29 (2016).

[67] Dan Boneh and Matt Franklin. “Identity-based encryption from the Weil pairing”. In: *Annual international cryptology conference*. Springer. 2001, pp. 213–229.

[68] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. “Improving language models by retrieving from trillions of tokens”. In: *International conference on machine learning*. PMLR. 2022, pp. 2206–2240.

[69] Houssem Ben Braiek and Foutse Khomh. “Machine learning robustness: A primer”. In: *Trustworthy AI in Medical Imaging*. Elsevier, 2025, pp. 37–71.

[70] Leo Breiman. “Random forests”. In: *Machine learning* 45 (2001), pp. 5–32.

[71] Wieland Brendel, Jonas Rauber, and Matthias Bethge. “Decision-based adversarial attacks: Reliable attacks against black-box machine learning models”. In: *arXiv preprint arXiv:1712.04248* (2017).

[72] Kiel Brennan-Marquez, Karen Levy, and Daniel Susser. “Strange loops”. In: *Berkeley Technology Law Journal* 34.3 (2019), pp. 745–772.

[73] Margot Brereton, Aloha Hufana Ambe, David Lovell, Laurianne Sitbon, Tara Capel, Alessandro Soro, Yue Xu, Catarina Moreira, Benoit Favre, and Andrew Bradley. “Designing Interaction with AI for Human Learning: Towards Human-Machine Teaming in Radiology Training”. In: *Proceedings of the 35th Australian Computer-Human Interaction Conference*. 2023, pp. 639–647.

[74] Justin Brickell and Vitaly Shmatikov. “The cost of privacy: destruction of data-mining utility in anonymized data publishing”. In: *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2008, pp. 70–78.

[75] Thomas M Brill, Laura Munoz, and Richard J Miller. “Siri, Alexa, and other digital assistants: a study of customer satisfaction with artificial intelligence applications”. In: *The role of smart technologies in decision making*. Routledge, 2022, pp. 35–70.

[76] Jace Browning and Robert Adams. *Doorstop*. `https://github.com/doorstop-dev/doorstop`. 2014.

[77] Jace Browning and Robert Adams. “Doorstop: text-based requirements management using version control”. In: (2014).

[78] Marc-Etienne Brunet, Colleen Alkalay-Houlihan, Ashton Anderson, and Richard Zemel. “Understanding the origins of bias in word embeddings”. In: *International conference on machine learning*. PMLR. 2019, pp. 803–811.

[79] Joy Buolamwini and Timnit Gebru. “Gender shades: Intersectional accuracy disparities in commercial gender classification”. In: *Conference on fairness, accountability and transparency*. PMLR. 2018, pp. 77–91.

[80] Crystal Butler, Harriet Oster, and Julian Togelius. “Human-in-the-loop ai for analysis of free response facial expression label sets”. In: *Proceedings of the 20th ACM international conference on intelligent virtual agents*. 2020, pp. 1–8.

[81] Michael J Cafarella, Alon Halevy, and Nodira Khoussainova. “Data integration for the relational web”. In: *Proceedings of the VLDB Endowment* 2.1 (2009), pp. 1090–1101.

[82] Carrie J Cai, Emily Reif, Narayan Hegde, Jason Hipp, Been Kim, Daniel Smilkov, Martin Wattenberg, Fernanda Viegas, Greg S Corrado, Martin C Stumpe, et al. “Human-centered tools for coping with imperfect algorithms during medical decision-making”. In: *Proceedings of the 2019 chi conference on human factors in computing systems*. 2019, pp. 1–14.

[83] Nicholas Carlini and David Wagner. “Towards evaluating the robustness of neural networks”. In: *2017 ieee symposium on security and privacy (sp)*. Ieee. 2017, pp. 39–57.

[84] de Gea Juan M Carrillo et al. “Commonalities and differences between requirements engineering tools: A quantitative approach”. In: *Computer Science and Information Systems* 12.1 (2015), pp. 257–288.

[85] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. “Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission”. In: *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 2015, pp. 1721–1730.

[86] Alberto Huertas Celdran, Jan Kreischer, Melike Demirci, Joel Leupp, Pedro M Sanchez, Muriel Figueredo Franco, Gérôme Bovet, Gregorio Martinez Perez, and Burkhard Stiller. “A Framework Quantifying Trustworthiness of Supervised Machine and Deep Learning Models”. In: *SafeAI2023: The AAAI’s Workshop on Artificial Intelligence Safety*. 2023, pp. 2938–2948.

[87] Chengliang Chai and Guoliang Li. “Human-in-the-loop Techniques in Machine Learning.” In: *IEEE Data Eng. Bull.* 43.3 (2020), pp. 37–52.

[88] Chengliang Chai, Jiayi Wang, Yuyu Luo, Zeping Niu, and Guoliang Li. “Data management for machine learning: A survey”. In: *IEEE Transactions on Knowledge and Data Engineering* 35.5 (2022), pp. 4646–4667.

[89] Anirban Chakraborty et al. “Adversarial attacks and defences: A survey”. In: *arXiv preprint arXiv:1810.00069* (2018).

[90] Vinay Chamola, Vikas Hassija, A Razia Sulthana, Debshishu Ghosh, Divyansh Dhingra, and Biplab Sikdar. “A review of trustworthy and explainable artificial intelligence (xai)”. In: *IEEe Access* (2023).

[91] Bhanu Chander, Chinju John, Lekha Warrier, and Kumaravelan Gopalakrishnan. “Toward trustworthy artificial intelligence (TAI) in the context of explainability and robustness”. In: *ACM Computing Surveys* (2024).

[92] Bhanu Chander, Chinju John, Lekha Warrier, and Kumaravelan Gopalakrishnan. “Toward trustworthy artificial intelligence (TAI) in the context of explainability and robustness”. In: *ACM Computing Surveys* 57.6 (2025), pp. 1–49.

[93] Chun-Hao Chang, George Alexandru Adam, and Anna Goldenberg. “Towards robust classification model by counterfactual and invariant data generation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 15212–15221.

[94] Bryant Chen et al. “Detecting backdoor attacks on deep neural networks by activation clustering”. In: *arXiv preprint arXiv:1811.03728* (2018).

[95] Deyan Chen and Hong Zhao. “Data security and privacy protection issues in cloud computing”. In: *2012 international conference on computer science and electronics engineering*. Vol. 1. IEEE. 2012, pp. 647–651.

[96] Hongge Chen, Huan Zhang, Duane Boning, and Cho-Jui Hsieh. “Robust decision trees against adversarial examples”. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 1122–1131.

[97] Hongyi Chen, Jinshu Su, Linbo Qiao, and Qin Xin. “Malware collusion attack against SVM: Issues and countermeasures”. In: *Applied Sciences* 8.10 (2018), p. 1718.

[98] Jian Chen et al. “De-pois: An attack-agnostic defense against data poisoning attacks”. In: *TIFS* 16 (2021), pp. 3412–3425.

[99] Jianbo Chen, Michael I Jordan, and Martin J Wainwright. “Hopskipjumpattack: A query-efficient decision-based attack”. In: *2020 ieee symposium on security and privacy (sp)*. IEEE. 2020, pp. 1277–1294.

[100] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. “Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models”. In: *Proceedings of the 10th ACM workshop on artificial intelligence and security*. 2017, pp. 15–26.

[101] Tianqi Chen and Carlos Guestrin. “Xgboost: A scalable tree boosting system”. In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016, pp. 785–794.

[102] Zichen Chen, Zelei Liu, Kang Loon Ng, Han Yu, Yang Liu, and Qiang Yang. “A gamified research tool for incentive mechanism design in federated learning”. In: *Federated Learning: Privacy and Incentive*. Springer, 2020, pp. 168–175.

[103] Minhao Cheng, Thong Le, Pin-Yu Chen, Jinfeng Yi, Huan Zhang, and Cho-Jui Hsieh. “Query-efficient hard-label black-box attack: An optimization-based approach”. In: *arXiv preprint arXiv:1807.04457* (2018).

[104] Monica Chew and J Doug Tygar. “Image recognition captchas”. In: *International Conference on Information Security*. Springer. 2004, pp. 268–279.

[105] Prateek Chhikara et al. “Federated Learning and Autonomous UAVs for Hazardous Zone Detection and AQI Prediction in IoT Environment”. In: *IEEE IoT Journal* 8.20 (2021), pp. 15456–15467. DOI: 10.1109/JIOT.2021.3074523.

[106] *CIO.gov - Executive Order (EO) 13960*. URL: https://www.cio.gov/policies-and-priorities/Executive-Order-13960-AI-Use-Case-Inventories-Reference.

[107] Michael R Clark, Peter Swartz, Andrew Alten, and Raed M Salih. “Toward Black-box Image Extraction Attacks on RBF SVM Classification Model”. In: *2020 IEEE/ACM Symposium on Edge Computing (SEC)*. IEEE. 2020, pp. 394–399.

[108] European Commission. *EU Artificial Intelligence Act: Political Agreement Reached on Landmark Rules for Trustworthy AI*. Accessed: 2025-02-19. 2024. URL: https://ec.europa.eu/commission/presscorner/detail/ov/ip_24_4123.

[109] *Commonalities and differences between requirements engineering tools: A quantitative approach*. Accessed on 17 November 2023. URL: https://www.um.es/giisw/EN/re-tools-survey/part2.pdf.

[110] Mobile Cloud Computing. *Spatial Backend*. https://github.com/mobile-cloud-computing/spatial-backend. Accessed: 2025-06-25. 2021.

[111] Mobile Cloud Computing. *Spatial Backend*. https://github.com/mobile-cloud-computing/spartial-frontendv2. Accessed: 2025-06-25. 2021.

[112] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. "Algorithmic decision making and the cost of fairness". In: *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*. 2017, pp. 797–806.

[113] Luiz Marcio Cysneiros et al. "Software transparency as a key requirement for self-driving cars". In: *IEEE 26th international requirements engineering conference (RE)*. 2018, pp. 382–387.

[114] Alexander D'Amour, Hansa Srinivasan, James Atwood, Pallavi Baljekar, David Sculley, and Yoni Halpern. "Fairness is not static: deeper understanding of long term fairness via simulation studies". In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 2020, pp. 525–534.

[115] Hoa Khanh Dam, Truyen Tran, and Aditya Ghose. "Explainable software analytics". In: *Proceedings of the 40th international conference on software engineering: New ideas and emerging results*. 2018, pp. 53–56.

[116] Daniela Damian et al. "Awareness meets requirements management: awareness needs in global software development". In: *Proc. of the Int'l Workshop on Global Software Development, International Conference on Software Engineering (ICSE 2003)*. 2003.

[117] Farooq Dar et al. "LIZARD: Pervasive Sensing for Autonomous Plastic Litter Monitoring". In: *9th ACM/IEEE IOTDI*. 2025.

[118] Jeffrey Dastin. "Amazon scraps secret AI recruiting tool that showed bias against women". In: *Ethics of data and analytics*. Auerbach Publications, 2022, pp. 296–299.

[119] Berardina De Carolis, Francesco Ladogana, and Nicola Macchiarulo. "Yolo trashnet: Garbage detection in video streams". In: *2020 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS)*. IEEE. 2020, pp. 1–7.

[120] Juan M Carrillo De Gea, Joaquín Nicolás, José L Fernández Alemán, Ambrosio Toval, Christof Ebert, and Aurora Vizcaíno. "Requirements engineering tools: Capabilities, survey and assessment". In: *Information and Software Technology* 54.10 (2012), pp. 1142–1157.

[121] Atsushi Deguchi, Chiaki Hirai, Hideyuki Matsuoka, Taku Nakano, Kohei Oshima, Mitsuharu Tai, and Shigeyuki Tani. “What is society 5.0”. In: *Society* 5.0 (2020), pp. 1–24.

[122] Wanghua Deng and Ruoxue Wu. “Real-time driver-drowsiness detection system using facial features”. In: *Ieee Access* 7 (2019), pp. 118727–118738.

[123] Murat Dikmen and Catherine Burns. “Trust in autonomous vehicles: The case of Tesla Autopilot and Summon”. In: *2017 IEEE International conference on systems, man, and cybernetics (SMC)*. IEEE. 2017, pp. 1093–1098.

[124] Xueying Ding, Nikita Seleznev, Senthil Kumar, C Bayan Bruss, and Leman Akoglu. “From Detection to Action: a Human-in-the-loop Toolkit for Anomaly Reasoning and Management”. In: *Proceedings of the Fourth ACM International Conference on AI in Finance*. 2023, pp. 279–287.

[125] Julia Dressel and Hany Farid. “The accuracy, fairness, and limits of predicting recidivism”. In: *Science advances* 4.1 (2018), eaao5580.

[126] Vasisht Duddu, Debasis Samanta, D Vijay Rao, and Valentina E Balas. “Stealing neural networks via timing side channels”. In: *arXiv preprint arXiv:1812.11720* (2018).

[127] Corey Dunn, Nour Moustafa, and Benjamin Turnbull. “Robustness evaluations of sustainable machine learning models against data poisoning attacks in the internet of things”. In: *Sustainability* 12.16 (2020), p. 6434.

[128] Wenli Duo, MengChu Zhou, and Abdullah Abusorrah. “A survey of cyber attacks on cyber physical systems: Recent advances and challenges”. In: *IEEE/CAA Journal of Automatica Sinica* 9.5 (2022), pp. 784–800.

[129] Cynthia Dwork, Nicole Immorlica, Adam Tauman Kalai, and Max Leiserson. “Decoupled classifiers for group-fair and efficient machine learning”. In: *Conference on fairness, accountability and transparency*. PMLR. 2018, pp. 119–133.

[130] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. “Calibrating noise to sensitivity in private data analysis”. In: *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*. Springer. 2006, pp. 265–284.

[131] Organisation for Economic Co-operation and Development. *National AI policies & strategies*. `https://oecd.ai/en/dashboards/overview`. Accessed: 2025-04-07.

[132] Upol Ehsan, Philipp Wintersberger, Q Vera Liao, Elizabeth Anne Watkins, Carina Manger, Hal Daumé III, Andreas Riener, and Mark O Riedl. “Human-Centered Explainable AI (HCXAI): beyond opening the black-box of AI”. In: *CHI conference on human factors in computing systems extended abstracts*. 2022, pp. 1–7.

[133] Jan-Erik Ekberg et al. “The untapped potential of trusted execution environments on mobile devices”. In: *IEEE S&P* 12.4 (2014), pp. 29–37.

[134] Salma Elmalaki. “Fair-iot: Fairness-aware human-in-the-loop reinforcement learning for harnessing human variability in personalized iot”. In: *Proceedings of the International Conference on Internet-of-Things Design and Implementation*. 2021, pp. 119–132.

[135] European Parliamentary Research Service. *A Governance Framework for Algorithmic Accountability and Transparency*. Tech. rep. PE 624.262. European Parliament, 2019. URL: https://www.europarl.europa.eu/thinktank/en/document/EPRS_STU(2019)624262.

[136] European Union. *Article 9 GDPR - Processing of special categories of personal data*. https://gdpr-info.eu/art-9-gdpr/. Accessed: 2025-07-22. 2018.

[137] European Union. *Artificial Intelligence Act - Article 10*. Accessed: 2025-03-24. 2025. URL: https://artificialintelligenceact.eu/article/10/.

[138] European Union. *Artificial Intelligence Act: Article 14 – Human Oversight*. Accessed: 2025-01-08. 2024. URL: https://artificialintelligenceact.eu/article/14/.

[139] European Union. *Artificial Intelligence Act: Article 14 – Obligations of Deployers of High-Risk AI Systems*. Accessed: 2025-01-08. 2024. URL: https://artificialintelligenceact.eu/article/14/.

[140] European Union. *Artificial Intelligence Act: Chapter III, Section 2 – Requirements for High-Risk AI Systems*. Accessed: 2025-01-08. 2024. URL: https://artificialintelligenceact.eu/section/3-2/.

[141] Shamal Faily and Duncan Ki-Aries. “Usable and Secure Requirements Engineering with CAIRIS”. In: *2019 IEEE 27th International Requirements Engineering Conference (RE)*. IEEE. 2019, pp. 502–503.

[142] Shamal Faily and Shamal Faily. “Usable and Secure Software Design: The State-of-the-Art”. In: *Designing Usable and Secure Software with IRIS and CAIRIS* (2018), pp. 9–53.

[143] Wenfei Fan. “Data quality: From theory to practice”. In: *Acm Sigmod Record* 44.3 (2015), pp. 7–18.

[144] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. “Certifying and removing disparate impact”. In: *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*. 2015, pp. 259–268.

[145] Alexander Felfernig et al. “OpenReq: recommender systems in requirements engineering”. In: *Proceedings of the Workshop Papers of I-Know 2017: Co-Located with International Conference on Knowledge Technologies and Data-Driven Business 2017 (I-Know 2017): Graz, Austria, October 11-12, 2017*. 2017, pp. 1–4.

[146] Hossein Fereidooni, Samuel Marchal, et al. “SAFELearn: secure aggregation for private federated learning”. In: *IEEE SP Workshops 2021*, pp. 56–62.

[147] Carlos Bermejo Fernandez et al. “Implementing GDPR for mobile and ubiquitous computing”. In: *Proceedings of the 23rd Annual International Workshop on Mobile Computing Systems and Applications*. 2022, pp. 88–94.

[148] Clàudia Figueras, Harko Verhagen, and Teresa Cerratto Pargman. “Trustworthy AI for the People?” In: *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 2021, pp. 269–270.

[149] Anthony Finkelstein and Wolfgang Emmerich. “The future of requirements management tools”. In: Oesterreichische Computer Gesellschaft (Austrian Computer Society), 2000.

[150] Samuel G Finlayson, John D Bowers, Joichi Ito, Jonathan L Zittrain, Andrew L Beam, and Isaac S Kohane. “Adversarial attacks on medical machine learning”. In: *Science* 363.6433 (2019), pp. 1287–1289.

[151] Andreas Florath. *rmToo – Requirements Management Tool*. `https://github.com/florath/rmtoo`. 2020.

[152] Huber Flores. “AI Sensors and Dashboards”. In: *IEEE Computer Magazine* (2024).

[153] Huber Flores. “Ai sensors and dashboards”. In: *Computer* 57.8 (2024), pp. 55–64.

[154] Huber Flores et al. “Collaboration Stability: Quantifying the Success and Failure of Opportunistic Collaboration”. In: *IEEE Computer Magazine* (2021).

[155] Huber Flores et al. “Evaluating energy-efficiency using thermal imaging”. In: *Proceedings of the 20th HotMobile*. 2019, pp. 147–152.

[156] Huber Flores. “Opportunistic multi-drone networks: Filling the spatiotemporal holes of collaborative and distributed applications”. In: *IEEE Internet of Things Magazine* 7.2 (2024), pp. 94–100.

[157] Huber Flores et al. “PENGUIN: aquatic plastic pollution sensing using AUVs”. In: *Proceedings of the 6th DroNet*. 2020, pp. 1–6.

[158] Alexander L Fogel and Joseph C Kvedar. “Artificial intelligence powers digital medicine”. In: *NPJ digital medicine* 1.1 (2018), p. 5.

[159] Riccardo Fogliato, Maria De-Arteaga, and Alexandra Chouldechova. “A case for humans-in-the-loop: Decisions in the presence of misestimated algorithmic scores”. In: *Available at SSRN* (2022).

[160] Yann Fraboni et al. “Free-rider attacks on model aggregation in federated learning”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 1846–1854.

[161] Eitan Frachtenberg. “Practical drone delivery”. In: *Computer* 52.12 (2019), pp. 53–57.

[162] Matt Fredrikson et al. “Model inversion attacks that exploit confidence information and basic countermeasures”. In: *22nd ACM SIGSAC*. 2015, pp. 1322–1333.

[163] Jerome H Friedman. “Greedy function approximation: a gradient boosting machine”. In: *Annals of statistics* (2001), pp. 1189–1232.

[164] Yuchuan Fu et al. “A survey of driving safety with sensing, vehicular communications, and artificial intelligence-based collision avoidance”. In: *IEEE T-ITS* (2021).

[165] Zhangjie Fu, Yueyan Zhi, Shouling Ji, and Xingming Sun. “Remote Attacks on Drones Vision Sensors: An Empirical Study”. In: *IEEE Transactions on Dependable and Secure Computing* 19.5 (2021), pp. 3125–3135.

[166] Krishna Gade, Sahin Cem Geyik, Krishnaram Kenthapadi, Varun Mithal, and Ankur Taly. “Explainable AI in industry”. In: *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2019, pp. 3203–3204.

[167] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. “Retrieval-augmented generation for large language models: A survey”. In: *arXiv preprint arXiv:2312.10997* 2.1 (2023).

[168] Anna Baron Garcia, Radu F Babiceanu, and Remzi Seker. “Artificial intelligence and machine learning approaches for aviation cybersecurity: An overview”. In: *2021 integrated communications navigation and surveillance conference (ICNS)*. IEEE. 2021, pp. 1–8.

[169] Simson Garfinkel et al. *De-identification of Personal Information:*. US Department of Commerce, National Institute of Standards and Technology, 2015.

[170] Nakul Garg et al. “Thermware: Toward side-channel defense for tiny iot devices”. In: *24th Hotmobile*. 2023, pp. 81–88.

[171] Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. “Word embeddings quantify 100 years of gender and ethnic stereotypes”. In: *Proceedings of the National Academy of Sciences* 115.16 (2018), E3635–E3644.

[172] Yingqiang Ge, Shuchang Liu, Zuohui Fu, Juntao Tan, Zelong Li, Shuyuan Xu, Yunqi Li, Yikun Xian, and Yongfeng Zhang. “A survey on trustworthy recommender systems”. In: *ACM Transactions on Recommender Systems* 3.2 (2024), pp. 1–68.

[173] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. “Datasheets for datasets”. In: *Communications of the ACM* 64.12 (2021), pp. 86–92.

[174] Yolanda Gil, James Honaker, Shikhar Gupta, Yibo Ma, Vito D’Orazio, Daniel Garijo, Shruti Gadewar, Qifan Yang, and Neda Jahanshad. “Towards

human-guided machine learning". In: *Proceedings of the 24th international conference on intelligent user interfaces*. 2019, pp. 614–624.

[175] Naman Goel, Mohammad Yaghini, and Boi Faltings. "Non-discriminatory machine learning through convex fairness criteria". In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. 2018, pp. 116–116.

[176] Xiaowen Gong et al. "Incentivizing truthful data quality for quality-aware mobile data crowdsourcing". In: *In Proceedings of ACM MobiHoc*. 2018, pp. 161–170.

[177] Yue-Jiao Gong, Wei-Neng Chen, Zhi-Hui Zhan, Jun Zhang, Yun Li, Qingfu Zhang, and Jing-Jing Li. "Distributed evolutionary algorithms and their models: A survey of the state-of-the-art". In: *Applied Soft Computing* 34 (2015), pp. 286–300.

[178] Divya Gopinath, Hayes Converse, Corina Pasareanu, and Ankur Taly. "Property inference for deep neural networks". In: *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE. 2019, pp. 797–809.

[179] Orlena CZ Gotel and CW Finkelstein. "An analysis of the requirements traceability problem". In: *Proceedings of IEEE international conference on requirements engineering*. IEEE. 1994, pp. 94–101.

[180] Claire Greene and Joanna Stavins. "Did the Target data breach change consumer assessments of payment card security?" In: *Journal of Payments Strategy & Systems* 11.2 (2017), pp. 121–133.

[181] Venkat Gudivada, Amy Apon, and Junhua Ding. "Data quality considerations for big data and machine learning: Going beyond data cleaning and transformations". In: *International Journal on Advances in Software* 10.1 (2017), pp. 1–20.

[182] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. "A survey of methods for explaining black box models". In: *ACM computing surveys (CSUR)* 51.5 (2018), pp. 1–42.

[183] David Gunning and David Aha. "DARPA's explainable artificial intelligence (XAI) program". In: *AI magazine* 40.2 (2019), pp. 44–58.

[184] Suyog Gupta et al. "Model accuracy and runtime tradeoff in distributed deep learning: A systematic study". In: *ICDM*. IEEE. 2016, pp. 171–180.

[185] Sairam Gurajada, Lucian Popa, Kun Qian, and Prithviraj Sen. "Learning-based methods with human-in-the-loop for entity resolution". In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2019, pp. 2969–2970.

[186] Houda Hafi, Bouziane Brik, Miloud Bagaa, and Adlen Ksentini. "Impact of Neural Network Depth on Split Federated Learning Performance in

Low-Resource UAV Networks". In: *2024 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE. 2024, pp. 1290–1295.

[187] Alon Halevy, Flip Korn, Natalya F Noy, Christopher Olston, Neoklis Polyzotis, Sudip Roy, and Steven Euijong Whang. "Goods: Organizing google's datasets". In: *Proceedings of the 2016 International Conference on Management of Data*. 2016, pp. 795–806.

[188] Moritz Hardt, Eric Price, and Nati Srebro. "Equality of opportunity in supervised learning". In: *Advances in neural information processing systems* 29 (2016).

[189] Christopher G Harris. "Combining human-in-the-loop systems and AI fairness toolkits to reduce age bias in AI job hiring algorithms". In: *2024 IEEE International Conference on Big Data and Smart Computing (BigComp)*. IEEE. 2024, pp. 60–66.

[190] Helen Hastie, Francisco Javier Chiyah Garcia, David A Robb, Pedro Patron, and Atanas Laskov. "MIRIAM: a multimodal chat-based interface for autonomous systems". In: *Proceedings of the 19th ACM International Conference on Multimodal Interaction*. 2017, pp. 495–496.

[191] Ammar Haydari et al. "Deep reinforcement learning for intelligent transportation systems: A survey". In: *IEEE Trans. Intell. Transp. Syst* 23.1 (2020), pp. 11–32.

[192] Chen He, Vishnu Raj, Hans Moen, Tommi Gröhn, Chen Wang, Laura-Maria Peltonen, Saila Koivusalo, Pekka Marttinen, and Giulio Jacucci. "VMS: Interactive Visualization to Support the Sensemaking and Selection of Predictive Models". In: *Proceedings of the 29th International Conference on Intelligent User Interfaces*. 2024, pp. 229–244.

[193] Xinlei He, Jinyuan Jia, Michael Backes, Neil Zhenqiang Gong, and Yang Zhang. "Stealing links from graph neural networks". In: *30th USENIX Security Symposium (USENIX Security 21)*. 2021, pp. 2669–2686.

[194] Yi He, Xi Yang, Chia-Ming Chang, Haoran Xie, and Takeo Igarashi. "Efficient Human-in-the-loop System for Guiding DNNs Attention". In: *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 2023, pp. 294–306.

[195] Florian Heimerl, Christoph Kralj, Torsten Möller, and Michael Gleicher. "embcomp: Visual interactive comparison of vector embeddings". In: *IEEE Transactions on Visualization and Computer Graphics* 28.8 (2020), pp. 2953–2969.

[196] Patrick Hemmer, Monika Westphal, Max Schemmer, Sebastian Vetter, Michael Vössing, and Gerhard Satzger. "Human-AI collaboration: the effect of AI delegation on human task performance and task satisfaction". In: *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 2023, pp. 453–463.

[197] Hans-Martin Heyn, Eric Knauss, Amna Pir Muhammad, Olof Eriksson, Jennifer Linder, Padmini Subbiah, Shameer Kumar Pradhan, and Sagar Tungal. "Requirement engineering challenges for ai-intense systems development". In: *IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI (WAIN)*. 2021, pp. 89–96.

[198] High-Level Expert Group on Artificial Intelligence (AI HLEG). *Ethics Guidelines for Trustworthy AI*. European Commission, Digital Strategy Directorate. 2019. URL: `https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai`.

[199] Michael Hind, Stephanie Houde, Jacquelyn Martino, Aleksandra Mojsilovic, David Piorkowski, John Richards, and Kush R Varshney. "Experiences with improving the transparency of AI models and services". In: *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*. 2020, pp. 1–8.

[200] Matthias Hirth, Tobias Hoßfeld, and Phuoc Tran-Gia. "Anatomy of a crowdsourcing platform-using the example of microworkers. com". In: *2011 Fifth international conference on innovative mobile and internet services in ubiquitous computing*. IEEE. 2011, pp. 322–329.

[201] Kalle Hjerppe, Jukka Ruohonen, and Ville Leppänen. "The general data protection regulation: requirements, architectures, and constraints". In: *IEEE 27th International Requirements Engineering Conference (RE)*. 2019, pp. 265–275.

[202] Torsten Hoefler, Dan Alistarh, Tal Ben-Nun, Nikoli Dryden, and Alexandra Peste. "Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks". In: *Journal of Machine Learning Research* 22.241 (2021), pp. 1–124.

[203] Matthias Hoffmann et al. "Requirements for requirements management tools". In: *Proceedings 12th IEEE International Requirements Engineering Conference, 2004.* IEEE. 2004, pp. 301–308.

[204] Cheok Jun Hong and Vimal Rau Aparow. "System configuration of Human-in-the-loop Simulation for Level 3 Autonomous Vehicle using IPG CarMaker". In: *2021 IEEE international conference on internet of things and intelligence systems (IoTaIS)*. IEEE. 2021, pp. 215–221.

[205] Sungsoo Ray Hong, Jessica Hullman, and Enrico Bertini. "Human factors in model interpretability: Industry practices, challenges, and needs". In: *Proceedings of the ACM on Human-Computer Interaction* 4.CSCW1 (2020), pp. 1–26.

[206] Yuan Hong, Jaideep Vaidya, Haibing Lu, Panagiotis Karras, and Sanjay Goel. "Collaborative search log sanitization: Toward differential privacy and boosted utility". In: *IEEE Transactions on Dependable and Secure Computing* 12.5 (2014), pp. 504–518.

[207] Simo Hosio et al. “Monetary assessment of battery life on smartphones”. In: *In Proceedings of CHI*. 2016, pp. 1869–1880.

[208] Ayanna Howard and Jason Borenstein. “The ugly truth about ourselves and our robot creations: the problem of bias and social inequity”. In: *Science and engineering ethics* 24.5 (2018), pp. 1521–1536.

[209] Michael Howard and Steve Lipner. *The security development lifecycle*. Vol. 8. Microsoft Press Redmond, 2006.

[210] Kai Hu, Sheng Gong, Qi Zhang, Chaowen Seng, Min Xia, and Shanshan Jiang. “An overview of implementing security and privacy in federated learning”. In: *Artificial Intelligence Review* 57.8 (2024), p. 204.

[211] Shuyan Hu, Xiaojing Chen, Wei Ni, Ekram Hossain, and Xin Wang. “Distributed machine learning for wireless communication networks: Techniques, architectures, and applications”. In: *IEEE Communications Surveys & Tutorials* 23.3 (2021), pp. 1458–1493.

[212] Jing Hua and Ping Wang. “Security vulnerabilities in Facebook data breach”. In: *International Conference on Information Technology-New Generations*. Springer. 2024, pp. 159–166.

[213] Ben Hutchinson, Andrew Smart, Alex Hanna, Emily Denton, Christina Greer, Oddur Kjartansson, Parker Barnes, and Margaret Mitchell. “Towards accountability for machine learning datasets: Practices from software engineering and infrastructure”. In: *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*. 2021, pp. 560–575.

[214] Johan Huysmans, Karel Dejaeger, Christophe Mues, Jan Vanthienen, and Bart Baesens. “An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models”. In: *Decision Support Systems* 51.1 (2011), pp. 141–154.

[215] International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC). *Information Technology — Artificial Intelligence — Artificial Intelligence Concepts and Terminology*. ISO/IEC 22989:2022, Published in July 2022. 2022. URL: `https://www.iso.org/standard/77304.html`.

[216] ISO/IEC. *Information Technology – Systems and Software Engineering – Guide for Requirements Engineering tool Capabilities*. First ed. 2009. 2009.

[217] Alon Jacovi, Ana Marasović, Tim Miller, and Yoav Goldberg. “Formalizing trust in artificial intelligence: Prerequisites, causes and goals of human trust in AI”. In: *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*. 2021, pp. 624–635.

[218] Mikhiya James, M Mruthula, Vismaya Bhaskaran, S Asha, et al. “Evasion Attacks On Svm Classifier”. In: *2019 9th International Conference on Advances in Computing and Communication (ICACC)*. IEEE. 2019, pp. 125–129.

[219] Zhihao Jia, Sina Lin, Charles R Qi, and Alex Aiken. “Exploring Hidden Dimensions in Parallelizing Convolutional Neural Networks.” In: *ICML*. 2018, pp. 2279–2288.

[220] Weijie Jiang and Zachary A Pardos. “Towards equity and algorithmic fairness in student grade prediction”. In: *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 2021, pp. 608–617.

[221] Fatema Tuz Johora, Rakibul Hasan, Syeda Farjana Farabi, Mohammad Zahidul Alam, Md Imran Sarkar, and Md Abdullah Al Mahmud. “AI Advances: Enhancing Banking Security with Fraud Detection”. In: *2024 First International Conference on Technological Innovations and Advance Computing (TIACOMP)*. IEEE. 2024, pp. 289–294.

[222] Madhura Joshi, Ankit Pal, and Malaikannan Sankarasubbu. “Federated learning for healthcare domain-pipeline, applications and challenges”. In: *ACM Transactions on Computing for Healthcare* 3.4 (2022), pp. 1–36.

[223] Jeesu Jung, Hyein Seo, Sangkeun Jung, Riwoo Chung, Hwijung Ryu, and Du-Seong Chang. “Interactive User Interface for Dialogue Summarization”. In: *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 2023, pp. 934–957.

[224] Peter Kairouz et al. “Advances and open problems in federated learning”. In: *Foundations and Trends® in Machine Learning* 14.1–2 (2021), pp. 1–210.

[225] Alper Kanak, Salih Ergün, Ali Serdar Atalay, Stefano Persi, and Ahu Ece Hartavi Karcı. “A review and strategic approach for the transition towards third-wave trustworthy and explainable ai in connected, cooperative and automated mobility (CCAM)”. In: *2022 27th Asia Pacific Conference on Communications (APCC)*. IEEE. 2022, pp. 108–113.

[226] Alex Kantchelian, J Doug Tygar, and Anthony Joseph. “Evasion and hardening of tree ensemble classifiers”. In: *International conference on machine learning*. PMLR. 2016, pp. 2387–2396.

[227] Jan Kantert, Sven Tomforde, and Christian Mueller-Schloer. “Measuring self-organisation in distributed systems by external observation”. In: *ARCS 2015-The 28th International Conference on Architecture of Computing Systems. Proceedings*. VDE. 2015, pp. 1–8.

[228] Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. “What can we learn privately?” In: *SIAM Journal on Computing* 40.3 (2011), pp. 793–826.

[229] Kleomenis Katevas et al. “FLaaS-enabling practical federated learning on mobile environments”. In: *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services*. 2022, pp. 605–606.

[230] Davinder Kaur et al. “Requirements for trustworthy artificial intelligence–a review”. In: *Advances in Networked-Based Information Systems: The*

23rd International Conference on Network-Based Information Systems (NBiS-2020) 23. Springer. 2021, pp. 105–115.

[231] Davinder Kaur, Suleyman Uslu, Kaley J Rittichier, and Arjan Durresi. "Trustworthy artificial intelligence: a review". In: *ACM Computing Surveys (CSUR)* 55.2 (2022), pp. 1–38.

[232] Markus Kelanti, Jarkko Hyysalo, Pasi Kuvaja, Markku Oivo, and Antti Välimäki. "A case study of requirements management: Toward transparency in requirements management tools". In: *Proceedings of the eighth international conference on software engineering advances (ICSEA 2013)*. 2013, pp. 597–604.

[233] Latif U Khan et al. "Federated learning for edge networks: Resource optimization and incentive mechanism". In: *IEEE Communications Magazine* 58.10 (2020), pp. 88–93.

[234] Latif U Khan et al. "Socially-aware-clustering-enabled federated learning for edge networks". In: *IEEE TNSM* 18.3 (2021), pp. 2641–2658.

[235] Shaharyar Khan, Ilya Kabanov, Yunke Hua, and Stuart Madnick. "A systematic analysis of the capital one data breach: Critical lessons learned". In: *ACM Transactions on Privacy and Security* 26.1 (2022), pp. 1–29.

[236] Rebecca Killick, Paul Fearnhead, and Idris A Eckley. "Optimal detection of changepoints with a linear computational cost". In: *Journal of the American Statistical Association* 107.500 (2012), pp. 1590–1598.

[237] Bongjun Kim and Bryan Pardo. "A human-in-the-loop system for sound event detection and annotation". In: *ACM Transactions on Interactive Intelligent Systems (TiiS)* 8.2 (2018), pp. 1–23.

[238] Sundong Kim, Tung-Duong Mai, Sungwon Han, Sungwon Park, DK Thi Nguyen, Jaechan So, Karandeep Singh, and Meeyoung Cha. "Active learning for human-in-the-loop customs inspection". In: *IEEE Transactions on Knowledge and Data Engineering* 35.12 (2022), pp. 12039–12052.

[239] Yujin Kim, Eunyeoul Lee, Yunjung Lee, and Uran Oh. "Understanding Novice's Annotation Process For 3D Semantic Segmentation Task With Human-In-The-Loop". In: *Proceedings of the 29th International Conference on Intelligent User Interfaces*. 2024, pp. 444–454.

[240] Bran Knowles and John T Richards. "The sanction of authority: Promoting public trust in AI". In: *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*. 2021, pp. 262–271.

[241] Johnson Kolluri, Vinay Kumar Kotte, MSB Phridviraj, and Shaik Razia. "Reducing overfitting problem in machine learning using novel L1/4 regularization method". In: *2020 4th international conference on trends in electronics and informatics (ICOEI)(48184)*. IEEE. 2020, pp. 934–938.

[242] Gerald Kotonya and Ian Sommerville. *Requirements engineering: processes and techniques*. Wiley Publishing, 1998.

[243] Pigi Kouki, Ilias Fountalis, Nikolaos Vasiloglou, Nian Yan, Unaiza Ahsan, Khalifeh Al Jadda, and Huiming Qu. “Product collection recommendation in online retail”. In: *Proceedings of the 13th ACM conference on recommender systems*. 2019, pp. 486–490.

[244] Riikka Koulu. “Proceduralizing control and discretion: Human oversight in artificial intelligence policy”. In: *Maastricht Journal of European and Comparative Law* 27.6 (2020), pp. 720–735.

[245] Josua Krause, Adam Perer, and Kenney Ng. “Interacting with predictions: Visual inspection of black-box machine learning models”. In: *Proceedings of the 2016 CHI conference on human factors in computing systems*. 2016, pp. 5686–5697.

[246] Sushant Kumar, Sumit Datta, Vishakha Singh, Deepanwita Datta, Sanjay Kumar Singh, and Ritesh Sharma. “Applications, Challenges, and Future Directions of Human-in-the-Loop Learning”. In: *IEEE Access* (2024).

[247] Thomas Kundinger, Philipp Wintersberger, and Andreas Riener. “(Over) Trust in automated driving: The sleeping pill of tomorrow?” In: *Extended abstracts of the 2019 CHI conference on human factors in computing systems*. 2019, pp. 1–6.

[248] Kaya Kuru and Wasiq Khan. “A framework for the synergistic integration of fully autonomous ground vehicles with smart city”. In: *IEEE Access* 9 (2020), pp. 923–948.

[249] Kyriakos Kyriakou and Jahna Otterbacher. “In humans, we trust: Multidisciplinary perspectives on the requirements for human oversight in algorithmic processes”. In: *Discover Artificial Intelligence* 3.1 (2023), p. 44.

[250] Eemil Lagerspetz et al. “Pervasive Data Science on the Edge”. In: *IEEE Pervasive Computing* (2019).

[251] Francesca Lagioia et al. “The impact of the General Data Protection Regulation (GDPR) on artificial intelligence”. In: (2020).

[252] Himabindu Lakkaraju, Stephen H Bach, and Jure Leskovec. “Interpretable decision sets: A joint framework for description and prediction”. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1675–1684.

[253] Anja Lambrecht and Catherine Tucker. “Algorithmic bias? An empirical study of apparent gender-based discrimination in the display of STEM career ads”. In: *Management science* 65.7 (2019), pp. 2966–2981.

[254] Pat Langley, Ben Meadows, Mohan Sridharan, and Dongkyu Choi. “Explainable agency for intelligent autonomous systems”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 2. 2017, pp. 4762–4763.

[255] Johann Laux. “Institutionalised distrust and human oversight of artificial intelligence: towards a democratic design of AI governance under the European Union AI Act”. In: *AI & society* 39.6 (2024), pp. 2853–2866.

[256] Lane Lawley and Christopher Maclellan. “VAL: Interactive Task Learning with GPT Dialog Parsing”. In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–18.

[257] Trung-Nghia Le, Akihiro Sugimoto, Shintaro Ono, and Hiroshi Kawasaki. “Toward interactive self-annotation for video object bounding box: Recurrent self-learning and hierarchical annotation based framework”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2020, pp. 3231–3240.

[258] Stephan J Lemmer, Anhong Guo, and Jason J Corso. “Human-Centered Deferred Inference: Measuring User Interactions and Setting Deferral Criteria for Human-AI Teams”. In: *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 2023, pp. 681–694.

[259] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. “Retrieval-augmented generation for knowledge-intensive nlp tasks”. In: *Advances in neural information processing systems* 33 (2020), pp. 9459–9474.

[260] Bo Li, Peng Qi, Bo Liu, Shuai Di, Jingen Liu, Jiquan Pei, Jinfeng Yi, and Bowen Zhou. “Trustworthy AI: From principles to practices”. In: *ACM Computing Surveys* 55.9 (2023), pp. 1–46.

[261] Guoliang Li. “Human-in-the-loop data integration”. In: *Proceedings of the VLDB Endowment* 10.12 (2017), pp. 2006–2017.

[262] He Li, Lu Yu, and Wu He. *The impact of GDPR on global technology development*. 2019.

[263] Jiangnan Li, Yingyuan Yang, Jinyuan Stella Sun, Kevin Tomsovic, and Hairong Qi. “Towards adversarial-resilient deep neural networks for false data injection attack detection in power grids”. In: *2023 32nd International Conference on Computer Communications and Networks (ICCCN)*. IEEE. 2023, pp. 1–10.

[264] Mengyuan Li, Yuheng Yang, Guoxing Chen, Mengjia Yan, and Yinqian Zhang. “Sok: Understanding design choices and pitfalls of trusted execution environments”. In: *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*. 2024, pp. 1600–1616.

[265] Qinbin Li, Wen, et al. “A survey on federated learning systems: vision, hype and reality for data privacy and protection”. In: *IEEE TKDE* (2021).

[266] Shaofeng Li, Minhui Xue, Benjamin Zi Hao Zhao, Haojin Zhu, and Xinpeng Zhang. “Invisible backdoor attacks on deep neural networks via steganography and regularization”. In: *IEEE Transactions on Dependable and Secure Computing* 18.5 (2020), pp. 2088–2105.

[267] Q Vera Liao, Daniel Gruen, and Sarah Miller. "Questioning the AI: informing design practices for explainable AI user experiences". In: *Proceedings of the 2020 CHI conference on human factors in computing systems*. 2020, pp. 1–15.

[268] Future of life. *Pause Giant AI Experiments: An Open Letter*. Accessed Dec 31, 2023. URL: https://futureoflife.org/open-letter/pause-giant-ai-experiments/.

[269] Dominic Lindsay, Sukhpal Singh Gill, Daria Smirnova, and Peter Garraghan. "The evolution of distributed computing systems: from fundamental to new frontiers". In: *Computing* 103.8 (2021), pp. 1859–1878.

[270] Steve Lipner. "The trustworthy computing security development lifecycle". In: *20th Annual Computer Security Applications Conference*. IEEE. 2004, pp. 2–13.

[271] Chi Liu, Tianqing Zhu, Jun Zhang, and Wanlei Zhou. "Privacy intelligence: A survey on image privacy in online social networks". In: *ACM Computing Surveys* 55.8 (2022), pp. 1–35.

[272] Gaoyang Liu, Chen Wang, Kai Peng, Haojun Huang, Yutong Li, and Wenqing Cheng. "Socinf: Membership inference attacks on social media health data with machine learning". In: *IEEE Transactions on Computational Social Systems* 6.5 (2019), pp. 907–921.

[273] Gaoyang Liu, Shijie Wang, Borui Wan, Zekun Wang, and Chen Wang. "ML-Stealer: Stealing Prediction Functionality of Machine Learning Models with Mere Black-Box Access". In: *2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE. 2021, pp. 532–539.

[274] Haochen Liu, Yiqi Wang, Wenqi Fan, Xiaorui Liu, Yaxin Li, Shaili Jain, Yunhao Liu, Anil Jain, and Jiliang Tang. "Trustworthy ai: A computational perspective". In: *ACM Transactions on Intelligent Systems and Technology* 14.1 (2022), pp. 1–59.

[275] Jun Liu, Qikun Dai, Hongyan Guo, Jingzheng Guo, and Hong Chen. "Human-oriented online driving authority optimization for driver-automation shared steering control". In: *IEEE Transactions on Intelligent Vehicles* 7.4 (2022), pp. 863–872.

[276] Simon Y Liu. "Artificial intelligence (AI) in agriculture". In: *IT professional* 22.3 (2020), pp. 14–15.

[277] Yunfei Liu, Xingjun Ma, James Bailey, and Feng Lu. "Reflection backdoor: A natural backdoor attack on deep neural networks". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*. Springer. 2020, pp. 182–199.

[278] Mohan Liyanage et al. "GEESE: Edge computing enabled by UAVs". In: *PMC* 72 (2021), p. 101340.

[279] David Lopez-Paz, Robert Nishihara, Soumith Chintala, Bernhard Scholkopf, and Léon Bottou. "Discovering causal signals in images". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 6979–6987.

[280] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. "Causal effect inference with deep latent-variable models". In: *Advances in neural information processing systems* 30 (2017).

[281] Daniel Lowd and Christopher Meek. "Adversarial learning". In: *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. 2005, pp. 641–647.

[282] Scott M Lundberg and Su-In Lee. "A unified approach to interpreting model predictions". In: *Adv Neural Inf Process Syst* 30 (2017).

[283] Xinjian Luo, Yuncheng Wu, Xiaokui Xiao, and Beng Chin Ooi. "Feature inference attack on model predictions in vertical federated learning". In: *2021 IEEE 37th International Conference on Data Engineering (ICDE)*. IEEE. 2021, pp. 181–192.

[284] Binh Thanh Luong, Salvatore Ruggieri, and Franco Turini. "k-NN as an implementation of situation testing for discrimination discovery and prevention". In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2011, pp. 502–510.

[285] Chunchuan Lyu, Kaizhu Huang, and Hai-Ning Liang. "A unified gradient regularization family for adversarial examples". In: *2015 IEEE international conference on data mining*. IEEE. 2015, pp. 301–309.

[286] Lingjuan Lyu, Xinyi Xu, Qian Wang, and Han Yu. "Collaborative fairness in federated learning". In: *Federated Learning*. Springer, 2020, pp. 189–204.

[287] Zhuoran Ma, Jianfeng Ma, Yinbin Miao, Yingjiu Li, and Robert H Deng. "ShieldFL: Mitigating model poisoning attacks in privacy-preserving federated learning". In: *IEEE Transactions on Information Forensics and Security* 17 (2022), pp. 1639–1654.

[288] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. "l-diversity: Privacy beyond k-anonymity". In: *Acm transactions on knowledge discovery from data (tkdd)* 1.1 (2007), 3–es.

[289] Ram Machlev, Michael Perl, Juri Belikov, Kfir Yehuda Levy, and Yoash Levron. "Measuring explainability and trustworthiness of power quality disturbances classifiers using XAI—Explainable artificial intelligence". In: *IEEE Transactions on Industrial Informatics* 18.8 (2021), pp. 5127–5137.

[290] Michael A Madaio, Luke Stark, Jennifer Wortman Vaughan, and Hanna Wallach. "Co-designing checklists to understand organizational challenges and opportunities around fairness in AI". In: *Proceedings of the 2020 CHI conference on human factors in computing systems*. 2020, pp. 1–14.

[291] Koki Madono, Teppei Nakano, Tetsunori Kobayashi, and Tetsuji Ogawa. "Efficient human-in-the-loop object detection using bi-directional deep sort and annotation-free segment identification". In: *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE. 2020, pp. 1226–1233.

[292] Ioannis Magnisalis, Stavros Demetriadis, and Anastasios Karakostas. "Adaptive and intelligent systems for collaborative learning support: A review of the field". In: *IEEE transactions on Learning Technologies* 4.1 (2011), pp. 5–20.

[293] Georgios Makridis, Georgios Fatouros, Athanasios Kiourtis, Dimitrios Kotios, Vasileios Koukos, Dimosthenis Kyriazis, and Jonh Soldatos. "Towards a unified multidimensional explainability metric: evaluating trustworthiness in AI models". In: *2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*. IEEE. 2023, pp. 504–511.

[294] Samuel Marchal, Alexey Kirichenko, Andrew Patel, Michell Boerger, Nikolay Tcholtchev, Manh-Dung Nguyen, Vinh Hoa La, Ana Rosa Cavalli, Claudio Soriente, Nicolas Kourtellis, Diego Perino, Andra Lutu, Souneli Park, Prachi Bagave, Aaron Ding, Marcus Westberg, Madhusanka Liyanage, Shen Wang, Bartlomiej Siniarski, Chamara Sandeepa, and Thulitha Senevirathna. *Horizon 2020 SPATIAL Deliverable 1.2 Security Threat Modeling for AI-Based System Architecture*. Technical Report. Version 1.2. H2020-SU-DS-2020, Nov. 2022. URL: https://spatial-h2020.eu/d1-2-security-threat-modeling-for-ai-based-system-architecture/.

[295] Ninareh Mehrabi, Fred Morstatter, Nanyun Peng, and Aram Galstyan. "Debiasing community detection: the importance of lowly connected nodes". In: *Proceedings of the 2019 IEEE/ACM international conference on advances in social networks analysis and mining*. 2019, pp. 509–512.

[296] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. "A survey on bias and fairness in machine learning". In: *ACM computing surveys (CSUR)* 54.6 (2021), pp. 1–35.

[297] Qinghao Meng, Wenguan Wang, Tianfei Zhou, Jianbing Shen, Yunde Jia, and Luc Van Gool. "Towards a weakly supervised framework for 3D point cloud object detection and annotation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.8 (2021), pp. 4454–4468.

[298] Leila Methnani, Manolis Chiou, Virginia Dignum, and Andreas Theodorou. "Who's in charge here? A survey on Trustworthy AI in Variable Autonomy Robotic Systems". In: *ACM Computing Surveys* 56.7 (2024), pp. 1–32.

[299] Maximilian Metzner, Daniel Utsch, Matthias Walter, Christian Hofstetter, Christina Ramer, Andreas Blank, and Jörg Franke. "A system for human-in-the-loop simulation of industrial collaborative robot applications". In:

2020 IEEE 16th International Conference on Automation Science and Engineering (CASE). IEEE. 2020, pp. 1520–1525.

[300] Daniela Micucci, Marco Mobilio, and Paolo Napoletano. "UniMiB SHAR: A Dataset for Human Activity Recognition Using Acceleration Data from Smartphones". In: *Applied Sciences* 7.10 (2017). ISSN: 2076-3417. DOI: 10.3390/app7101101. URL: http://www.mdpi.com/2076-3417/7/10/1101.

[301] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. "Model cards for model reporting". In: *Proceedings of the conference on fairness, accountability, and transparency*. 2019, pp. 220–229.

[302] Nader Mohamed, Jameela Al-Jaroodi, Imad Jawhar, Ahmed Idries, and Farhan Mohammed. "Unmanned aerial vehicles applications in future smart cities". In: *Technological forecasting and social change* 153 (2020), p. 119293.

[303] W Nor Haizan W Mohamed, Mohd Najib Mohd Salleh, and Abdul Halim Omar. "A comparative study of reduced error pruning method in decision tree algorithms". In: *2012 IEEE International conference on control system, computing and engineering*. IEEE. 2012, pp. 392–397.

[304] Sina Mohseni, Niloofar Zarei, and Eric D Ragan. "A multidisciplinary survey and framework for design and evaluation of explainable AI systems". In: *ACM Transactions on Interactive Intelligent Systems (TiiS)* 11.3-4 (2021), pp. 1–45.

[305] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. "Universal adversarial perturbations". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1765–1773.

[306] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. "Deepfool: a simple and accurate method to fool deep neural networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2574–2582.

[307] Saleh Mosaddegh, Loic Simon, and Frédéric Jurie. "Photorealistic face de-identification by aggregating donors' face components". In: *Asian conference on computer vision*. Springer. 2014, pp. 159–174.

[308] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. "Explaining machine learning classifiers through diverse counterfactual explanations". In: *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 2020, pp. 607–617.

[309] Mehran Mozaffari-Kermani, Susmita Sur-Kolay, Anand Raghunathan, and Niraj K Jha. "Systematic poisoning attacks on and defenses for machine

learning in healthcare". In: *IEEE journal of biomedical and health informatics* 19.6 (2014), pp. 1893–1905.

[310] Henry Muccini et al. "Software architecture for ml-based systems: what exists and what lies ahead". In: *IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI (WAIN)*. IEEE. 2021, pp. 121–128.

[311] Aiswarya Raj Munappy, Jan Bosch, Helena Holmström Olsson, Anders Arpteg, and Björn Brinne. "Data management for production quality deep learning models: Challenges and solutions". In: *Journal of Systems and Software* 191 (2022), p. 111359.

[312] Vidya Muthukumar. "Color-theoretic experiments to understand unequal gender classification accuracy from face images". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 2019, pp. 0–0.

[313] Luca Nannini, Agathe Balayn, and Adam Leon Smith. "Explainability in AI policies: A critical review of communications, reports, regulations, and standards in the EU, US, and UK". In: *Proceedings of the 2023 ACM conference on fairness, accountability, and transparency*. 2023, pp. 1198–1212.

[314] Tomoko Nemoto and David Beglar. "Likert-scale questionnaires". In: *JALT 2013 conference proceedings*. 2014, pp. 1–8.

[315] AKM Iqtidar Newaz, Nur Imtiazul Haque, Amit Kumar Sikder, Mohammad Ashiqur Rahman, and A Selcuk Uluagac. "Adversarial attacks to machine learning-based smart healthcare systems". In: *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE. 2020, pp. 1–6.

[316] NBC News. *Senate bill proposes more transparency from AI developers*. Accessed: 2024-12-30. NBC News. 2024. URL: `https://www.nbcnews.com/tech/senate-bill-transparency-ai-developers-rcna181724`.

[317] Thien Duc Nguyen et al. "Poisoning attacks on federated learning-based IoT intrusion detection system". In: *Proc. Workshop DISS*. 2020, pp. 1–7.

[318] Bostrom Nick. *Superintelligence: Paths, dangers, strategies*. 2014.

[319] Alexander Nikitin and Samuel Kaski. "Human-in-the-loop large-scale predictive maintenance of workstations". In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2022, pp. 3682–3690.

[320] Rui Ning et al. "Invisible poison: A blackbox clean label backdoor attack to deep neural networks". In: *INFOCOM 2021*.

[321] David Sousa Nunes, Pei Zhang, and Jorge Sá Silva. "A survey on human-in-the-loop applications towards an internet of all". In: *IEEE Communications Surveys & Tutorials* 17.2 (2015), pp. 944–965.

[322] Fer O'Neil. "Target data breach: applying user-centered design principles to data breach notifications". In: *Proceedings of the 33rd Annual International Conference on the Design of Communication*. 2015, pp. 1–8.

[323] Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. "Dissecting racial bias in an algorithm used to manage the health of populations". In: *Science* 366.6464 (2019), pp. 447–453.

[324] Luca Oneto, Michele Doninini, Amon Elders, and Massimiliano Pontil. "Taking advantage of multitask learning for fair classification". In: *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. 2019, pp. 227–237.

[325] Organisation for Economic Co-operation and Development (OECD). *OECD AI Principles*. Adopted on May 22, 2019, by OECD member countries. 2019. URL: `https://oecd.ai/en/ai-principles`.

[326] Abdul-Rasheed Ottun, Adeyinka Akintola, Mohan Liyanage, Michell Boerger, Pan Hui, Sasu Tarkoma, Nikolay Tcholtchev, Petteri Nurmi, and Huber Flores. "AI Robustness Against Attacks in City-Scale Autonomous Drone Deployments". In: *Computer* 57.12 (2024), pp. 47–57.

[327] Abdul-Rasheed Ottun, Pramod C Mane, Zhigang Yin, Souvik Paul, Mohan Liyanage, Jason Pridmore, Aaron Yi Ding, Rajesh Sharma, Petteri Nurmi, and Huber Flores. "Social-aware federated learning: Challenges and opportunities in collaborative data training". In: *IEEE Internet Computing* (2022).

[328] Abdul-Rasheed Ottun, Rasinthe Marasinghe, Toluwani Elemosho, Mohan Liyanage, Mohamad Ragab, Prachi Bagave, Marcus Westberg, Mehrdad Asadi, Michell Boerger, Chamara Sandeepa, et al. "The SPATIAL architecture: Design and development experiences from gauging and monitoring the ai inference capabilities of modern applications". In: *2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS)*. IEEE. 2024, pp. 947–959.

[329] Changkun Ou, Sven Mayer, and Andreas Martin Butz. "The Impact of Expertise in the Loop for Exploring Machine Rationality". In: *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 2023, pp. 307–321.

[330] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. "Training language models to follow instructions with human feedback". In: *Advances in neural information processing systems* 35 (2022), pp. 27730–27744.

[331] Arttu Paju, Muhammad Owais Javed, Juha Nurmi, Juha Savimäki, Brian McGillion, and Billy Bob Brumley. "Sok: A systematic review of tee usage for developing trusted applications". In: *Proceedings of the 18th*

International Conference on Availability, Reliability and Security. 2023, pp. 1–15.

[332] Haolin Pan, Yong Guo, Mianjie Yu, and Jian Chen. "Enhanced long-tailed recognition with contrastive cutmix augmentation". In: *IEEE Transactions on Image Processing* (2024).

[333] Nicolas Papernot et al. "Distillation as a defense to adversarial perturbations against deep neural networks". In: *SP*. IEEE. 2016, pp. 582–597.

[334] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. "The limitations of deep learning in adversarial settings". In: *2016 IEEE European symposium on security and privacy (EuroS&P)*. IEEE. 2016, pp. 372–387.

[335] Raja Parasuraman, Thomas B Sheridan, and Christopher D Wickens. "A model for types and levels of human interaction with automation". In: *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans* 30.3 (2000), pp. 286–297.

[336] Hyunjung Park and Jennifer Widom. "Crowdfill: collecting structured data from the crowd". In: *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*. 2014, pp. 577–588.

[337] Seonwook Park, Christoph Gebhardt, Roman Rädle, Anna Maria Feit, Hana Vrzakova, Niraj Ramesh Dayama, Hui-Shyong Yeo, Clemens N Klokmose, Aaron Quigley, Antti Oulasvirta, et al. "Adam: Adapting multi-user interfaces for collaborative environments in real-time". In: *Proceedings of the 2018 CHI conference on human factors in computing systems*. 2018, pp. 1–14.

[338] Flavio Parodi, Mikko Kylvaja, Gordon Alford, Juan Li, and Jose Pradas. "An automatic procedure for neighbor cell list definition in cellular networks". In: *2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*. IEEE. 2007, pp. 1–6.

[339] Simon Parsons et al. "Auctions and bidding: A guide for computer scientists". In: *ACM Computing Surveys (CSUR)* 43.2 (2011), pp. 1–59.

[340] Dario Pasquini, Giuseppe Ateniese, and Massimo Bernaschi. "Unleashing the tiger: Inference attacks on split learning". In: *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. 2021, pp. 2113–2129.

[341] Bhavik N Patel, Louis Rosenberg, Gregg Willcox, David Baltaxe, Mimi Lyons, Jeremy Irvin, Pranav Rajpurkar, Timothy Amrhein, Rajan Gupta, Safwan Halabi, et al. "Human–machine partnership with artificial intelligence for chest radiograph diagnosis". In: *NPJ digital medicine* 2.1 (2019), p. 111.

[342] Desmond U Patton, William R Frey, Kyle A McGregor, Fei-Tzin Lee, Kathleen McKeown, and Emanuel Moss. "Contextual analysis of social media: The promise and challenge of eliciting context in social media

posts with natural language processing". In: *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 2020, pp. 337–342.

[343] Andrea Paudice et al. "Detection of adversarial training examples in poisoning attacks through anomaly detection". In: *arXiv preprint arXiv:1802.03041* (2018).

[344] Ella Peltonen et al. "Energy Modeling of System Settings: A Crowdsourced Approach". In: *PerComs*. Mar. 2015, pp. 37–45.

[345] Jon Perez-Cerrolaza, Jaume Abella, Markus Borg, Carlo Donzella, Jesús Cerquides, Francisco J Cazorla, Cristofer Englund, Markus Tauber, George Nikolakopoulos, and Jose Luis Flores. "Artificial intelligence for safety-critical systems in industrial and transportation domains: A survey". In: *ACM Computing Surveys* 56.7 (2024), pp. 1–40.

[346] Dana Pessach and Erez Shmueli. "A review on fairness in machine learning". In: *ACM Computing Surveys (CSUR)* 55.3 (2022), pp. 1–44.

[347] Forough Poursabzi-Sangdeh, Daniel G Goldstein, Jake M Hofman, Jennifer Wortman Wortman Vaughan, and Hanna Wallach. "Manipulating and measuring model interpretability". In: *Proceedings of the 2021 CHI conference on human factors in computing systems*. 2021, pp. 1–52.

[348] Premise. *Premise - Data Collection and Insights Platform*. Accessed: 2025-04-03. 2025. URL: `https://premise.com/`.

[349] Teemu Pulkkinen et al. "Understanding wifi cross-technology interference detection in the real world". In: *Proceedings of IEEE ICDCS 2020*. IEEE. 2020, pp. 954–964.

[350] Mahima Pushkarna, Andrew Zaldivar, and Oddur Kjartansson. "Data cards: Purposeful and transparent dataset documentation for responsible ai". In: *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. 2022, pp. 1776–1826.

[351] Lingyun Qiu and Izak Benbasat. "Evaluating anthropomorphic product recommendation agents: A social relationship perspective to designing information systems". In: *Journal of management information systems* 25.4 (2009), pp. 145–182.

[352] Inioluwa Deborah Raji and Joy Buolamwini. "Actionable auditing: Investigating the impact of publicly naming biased performance results of commercial ai products". In: *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. 2019, pp. 429–435.

[353] Inioluwa Deborah Raji, Andrew Smart, Rebecca N White, Margaret Mitchell, Timnit Gebru, Ben Hutchinson, Jamila Smith-Loud, Daniel Theron, and Parker Barnes. "Closing the AI accountability gap: Defining an end-to-end framework for internal algorithmic auditing". In: *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 2020, pp. 33–44.

[354] Robert Rasch, Alexander Kott, and Kenneth D Forbus. “Incorporating AI into military decision making: an experiment”. In: *IEEE Intelligent Systems* 18.4 (2003), pp. 18–26.

[355] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. “Regularized evolution for image classifier architecture search”. In: *Proceedings of the aaai conference on artificial intelligence*. Vol. 33. 01. 2019, pp. 4780–4789.

[356] *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*. `https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng`. Official Journal of the European Union, L 119, 4 May 2016, pp. 1–88. 2016.

[357] Robert Nikolai Reith, Thomas Schneider, and Oleksandr Tkachenko. “Efficiently stealing your machine learning models”. In: *Proceedings of the 18th ACM Workshop on Privacy in the Electronic Society*. 2019, pp. 198–210.

[358] Khondker Jahid Reza, Md Zahidul Islam, and Vladimir Estivill-Castro. “Privacy protection of online social network users, against attribute inference attacks, through the use of a set of exhaustive rules”. In: *Neural Computing and Applications* 33.19 (2021), pp. 12397–12427.

[359] Peter A Riach and Judith Rich. “Field experiments of discrimination in the market place”. In: *The economic journal* 112.483 (2002), F480–F518.

[360] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “" Why should i trust you?" Explaining the predictions of any classifier”. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1135–1144.

[361] Jake Robertson, Thorsten Schmidt, Frank Hutter, and Noor Awad. “A Human-in-the-Loop Fairness-Aware Model Selection Framework for Complex Fairness Objective Landscapes”. In: *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. Vol. 7. 2024, pp. 1231–1242.

[362] Yuji Roh, Geon Heo, and Steven Euijong Whang. “A survey on data collection for machine learning: a big data-ai integration perspective”. In: *IEEE Transactions on Knowledge and Data Engineering* 33.4 (2019), pp. 1328–1347.

[363] Arnau Rovira-Sugranes, Abolfazl Razi, Fatemeh Afghah, and Jacob Chakareski. “A review of AI-enabled routing protocols for UAV networks: Trends, challenges, and future outlook”. In: *Ad Hoc Networks* 130 (2022), p. 102790.

[364] Yichen Ruan et al. “Towards flexible device participation in federated learning”. In: *AISTATS*. PMLR. 2021, pp. 3403–3411.

[365] Koosha Sadeghi, Ayan Banerjee, and Sandeep KS Gupta. “A system-driven taxonomy of attacks and defenses in adversarial machine learning”. In:

IEEE transactions on emerging topics in computational intelligence 4.4 (2020), pp. 450–467.

[366] Amit Sahai and Brent Waters. "Fuzzy identity-based encryption". In: *Advances in Cryptology–EUROCRYPT 2005: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005. Proceedings 24*. Springer. 2005, pp. 457–473.

[367] Vildan Salikutluk, Janik Schöpper, Franziska Herbert, Katrin Scheuermann, Eric Frodl, Dirk Balfanz, Frank Jäkel, and Dorothea Koert. "An Evaluation of Situational Autonomy for Human-AI Collaboration in a Shared Workspace Setting". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–17.

[368] Nithya Sambasivan, Shivani Kapania, Hannah Highfill, Diana Akrong, Praveen Paritosh, and Lora M Aroyo. ""Everyone wants to do the model work, not the data work": Data Cascades in High-Stakes AI". In: *proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 2021, pp. 1–15.

[369] Eric Samikwa, Antonio Di Maio, and Torsten Braun. "Ares: Adaptive resource-aware split learning for internet of things". In: *Computer Networks* 218 (2022), p. 109380.

[370] Sanjoy Sarkar, Tillman Weyde, A d Garcez, Gregory G Slabaugh, Simo Dragicevic, and Chris Percy. "Accuracy and interpretability trade-offs in machine learning applied to safer gambling". In: *CEUR Workshop Proceedings*. Vol. 1773. CEUR Workshop Proceedings. 2016.

[371] Iqbal H Sarker et al. "Ai-driven cybersecurity: an overview, security intelligence modeling and research directions". In: *SN Computer Science* 2 (2021), pp. 1–18.

[372] Sheree May Saßmannshausen, Nazmun Nisat Ontika, Aparecido Fabiano Pinatti De Carvalho, Mark Rouncefield, and Volkmar Pipek. "Amplifying Human Capabilities in Prostate Cancer Diagnosis: An Empirical Study of Current Practices and AI Potentials in Radiology". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–20.

[373] K Satish, A Lalitesh, K Bhargavi, M Sishir Prem, and T Anjali. "Driver drowsiness detection". In: *2020 international conference on communication and signal processing (ICCSP)*. IEEE. 2020, pp. 0380–0384.

[374] Laura Schelenz, Avi Segal, Oduma Adelio, and Kobi Gal. "Transparencycheck: An instrument for the study and design of transparency in ai-based personalization systems". In: *ACM Journal on Responsible Computing* 1.1 (2024), pp. 1–18.

[375] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. "Recommendations as treatments: Debiasing learn-

ing and evaluation". In: *international conference on machine learning*. PMLR. 2016, pp. 1670–1679.

[376] Avi Schwarzschild, Micah Goldblum, Arjun Gupta, John P Dickerson, and Tom Goldstein. "Just how toxic is data poisoning? a unified benchmark for backdoor and data poisoning attacks". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 9389–9398.

[377] MAIF Data Scientists. *SHAPASH*. 2023. URL: ,%20https://github.com/MAIF/shapash.

[378] Jessica Zeitz Self, Radha Krishnan Vinayagam, James Thomas Fry, and Chris North. "Bridging the gap between user intention and model parameters for human-in-the-loop data analytics". In: *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*. 2016, pp. 1–6.

[379] Sungyong Seo, Jing Huang, Hao Yang, and Yan Liu. "Interpretable convolutional neural networks with dual local and global attention for review rating prediction". In: *Proceedings of the eleventh ACM conference on recommender systems*. 2017, pp. 297–305.

[380] Andreea Claudia Şerban and Miltiadis D Lytras. "Artificial intelligence for smart renewable energy sector in europe—smart energy infrastructures for next generation smart cities". In: *IEEE access* 8 (2020), pp. 77364–77377.

[381] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. "Poison frogs! targeted clean-label poisoning attacks on neural networks". In: *Advances in neural information processing systems* 31 (2018).

[382] Atif Shah et al. "An evaluation of software requirements tools". In: *Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)*. 2017, pp. 278–283.

[383] Rushabh Shah et al. "IoT and AI in healthcare: A systematic literature review." In: *Issues in Information Systems* 19.3 (2018).

[384] FM Javed Mehedi Shamrat, Sovon Chakraborty, Md Masum Billah, Protiva Das, Jannatun Naeem Muna, and Rumesh Ranjan. "A comprehensive study on pre-pruning and post-pruning methods of decision tree classification algorithm". In: *2021 5th International conference on trends in electronics and informatics (ICOEI)*. IEEE. 2021, pp. 1339–1345.

[385] Wenting Shao and Xi Wang. "Modeling Data Requirements for Machine Learning Systems". In: *IEEE 13th International Conference on Software Engineering and Service Science (ICSESS)*. 2022, pp. 97–100. DOI: 10.1109/ICSESS54813.2022.9930317.

[386] Anee Sharma and Ningrinla Marchang. "A review on client-server attacks and defenses in federated learning". In: *Computers & Security* (2024), p. 103801.

[387] Muhammad K Shehzad et al. “Artificial Intelligence for 6G Networks: Technology Advancement and Standardization”. In: *IEEE Vehicular Technology Magazine* (2022).

[388] Ben Shneiderman. “Human-centered artificial intelligence: Reliable, safe & trustworthy”. In: *International Journal of Human–Computer Interaction* 36.6 (2020), pp. 495–504.

[389] Reza Shokri et al. “Membership inference attacks against machine learning models”. In: *2017 IEEE symposium on security and privacy (SP)*. IEEE. 2017, pp. 3–18.

[390] Connor Shorten and Taghi M Khoshgoftaar. “A survey on image data augmentation for deep learning”. In: *Journal of big data* 6.1 (2019), pp. 1–48.

[391] Joseph Shu, LihChyun Shu, Wun-Yan Chang, and ChiaCheng Su. “Fraud Detection Models and their Explanations for a Buy-Now-Pay-Later Application”. In: *Proceedings of the 2024 9th International Conference on Intelligent Information Technology*. 2024, pp. 439–445.

[392] Ilia Shumailov, Yiren Zhao, Daniel Bates, Nicolas Papernot, Robert Mullins, and Ross Anderson. “Sponge examples: Energy-latency attacks on neural networks”. In: *2021 IEEE European symposium on security and privacy (EuroS&P)*. IEEE. 2021, pp. 212–231.

[393] Venkatesh Sivaraman, Yiwei Wu, and Adam Perer. “Emblaze: Illuminating machine learning representations through interactive comparison of embedding spaces”. In: *Proceedings of the 27th International Conference on Intelligent User Interfaces*. 2022, pp. 418–432.

[394] Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B Viégas, and Martin Wattenberg. “Embedding projector: Interactive visualization and interpretation of embeddings”. In: *arXiv preprint arXiv:1611.05469* (2016).

[395] Alison Smith, Varun Kumar, Jordan Boyd-Graber, Kevin Seppi, and Leah Findlater. “Closing the loop: User-centered design and evaluation of a human-in-the-loop topic modeling system”. In: *Proceedings of the 23rd International Conference on Intelligent User Interfaces*. 2018, pp. 293–304.

[396] Aron Smith. *Open Source Requirements Management Tool,* `https://github.com/osrmt/osrmt`. 2019.

[397] Kacper Sokol, Raul Santos-Rodriguez, and Peter Flach. “FAT Forensics: A Python Toolbox for Algorithmic Fairness, Accountability and Transparency”. In: *Software Impacts* (2022), p. 100406.

[398] Mohsen Soori, Behrooz Arezoo, and Roza Dastres. “Artificial intelligence, machine learning and deep learning in advanced robotics, a review”. In: *Cognitive Robotics* 3 (2023), pp. 54–70.

[399] Timo Speith and Jing Xu. "Explainability and Transparency in Practice: A Comparison Between". In: *Explainable and Transparent AI and Multi-Agent Systems: 6th International Workshop, EXTRAAMAS 2024, Auckland, New Zealand, May 6–10, 2024, Revised Selected Papers*. Vol. 14847. Springer Nature. 2024, p. 205.

[400] Biplav Srivastava and Francesca Rossi. "Towards composable bias rating of AI services". In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. 2018, pp. 284–289.

[401] Unspecified CNN Staff. *Uber self-driving car death: Backup driver pleads guilty*. Accessed: 2025-06-18. July 2023. URL: https://edition.cnn.com/2023/07/29/business/uber-self-driving-car-death-guilty.

[402] National Institute of Standards and Technology (NIST). *AI Risk Management Framework (AI RMF)*. Accessed: 2024-12-30. National Institute of Standards and Technology, 2024. URL: https://www.nist.gov/itl/ai-risk-management-framework.

[403] Kenneth O Stanley, Jeff Clune, Joel Lehman, and Risto Miikkulainen. "Designing neural networks through neuroevolution". In: *Nature Machine Intelligence* 1.1 (2019), pp. 24–35.

[404] Statista. *Artificial intelligence (AI) market size worldwide in 2021 with a forecast until 2030*. Accessed Dec 31, 2023. URL: https://www.statista.com/study/38609/artificial-intelligence-ai-statista-dossier/.

[405] Sarah Sterz, Kevin Baum, Sebastian Biewer, Holger Hermanns, Anne Lauber-Rönsberg, Philip Meinel, and Markus Langer. "On the Quest for Effectiveness in Human Oversight: Interdisciplinary Perspectives". In: *The 2024 ACM Conference on Fairness, Accountability, and Transparency*. 2024, pp. 2495–2507.

[406] Ljiljana Stojanovic, Marko Dinic, Nenad Stojanovic, and Aleksandar Stojadinovic. "Big-data-driven anomaly detection in industry (4.0): An approach and a case study". In: *2016 IEEE international conference on big data (big data)*. IEEE. 2016, pp. 1647–1652.

[407] Margaret-Anne D Storey et al. "On the use of visualization to support awareness of human activities in software development: a survey and a framework". In: *Proceedings of the ACM symposium on Software visualization*. 2005, pp. 193–202.

[408] Aishwarya Sundaram, Hema Subramaniam, Siti Hafizah Ab Hamid, and Azmawaty Mohamad Nor. "A systematic literature review on social media slang analytics in contemporary discourse". In: *IEEE Access* 11 (2023), pp. 132457–132471.

[409] Harini Suresh and John Guttag. "A framework for understanding sources of harm throughout the machine learning life cycle". In: *Proceedings of*

the 1st ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization. 2021, pp. 1–9.

[410] Ammar Al-Taie, Graham Wilson, Euan Freeman, Frank Pollick, and Stephen Anthony Brewster. "Light it Up: Evaluating Versatile Autonomous Vehicle-Cyclist External Human-Machine Interfaces". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–20.

[411] Afaf Taık, Hajar Moudoud, and Soumaya Cherkaoui. "Data-Quality Based Scheduling for Federated Edge Learning". In: *2021 IEEE LCN*. IEEE. 2021, pp. 17–23.

[412] Abdulrahman Takiddin, Muhammad Ismail, Usman Zafar, and Erchin Serpedin. "Robust electricity theft detection against data poisoning attacks in smart grids". In: *IEEE Transactions on Smart Grid* 12.3 (2020), pp. 2675–2684.

[413] R Thangamani, RK Suguna, and GK Kamalam. "Drones and Autonomous Robotics Incorporating Computational Intelligence". In: *Computational Intelligent Techniques in Mechatronics* (2024), pp. 243–296.

[414] Chandra Thapa, Pathum Chamikara Mahawaga Arachchige, Seyit Camtepe, and Lichao Sun. "Splitfed: When federated learning meets split learning". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 8. 2022, pp. 8485–8493.

[415] Chandra Thapa, Mahawaga Arachchige Pathum Chamikara, and Seyit A Camtepe. "Advancements of federated learning towards privacy preservation: from federated learning to split learning". In: *Federated Learning Systems: Towards Next-Generation AI* (2021), pp. 79–109.

[416] Zhiyi Tian et al. "A comprehensive survey on poisoning attacks and countermeasures in machine learning". In: *ACM Computing Surveys* 55.8 (2022), pp. 1–35.

[417] Vale Tolpegin et al. "Data poisoning attacks against federated learning systems". In: *ESORICS*. Springer. 2020, pp. 480–501.

[418] Florian Tramer, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. "ENSEMBLE ADVERSARIAL TRAINING: ATTACKS AND DEFENSES". In: *stat* 1050 (2020), p. 26.

[419] Florian Tramèr, Fan Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. "Stealing machine learning models via prediction {APIs}". In: *25th USENIX security symposium (USENIX Security 16)*. 2016, pp. 601–618.

[420] Tram Thi Minh Tran, Callum Parker, Marius Hoggenmüller, Yiyuan Wang, and Martin Tomitsch. "Exploring the Impact of Interconnected External Interfaces in Autonomous Vehicles on Pedestrian Safety and Experience". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–17.

[421] Stacey Truex, Ling Liu, Mehmet Emre Gursoy, Lei Yu, and Wenqi Wei. "Demystifying membership inference attacks in machine learning as a service". In: *IEEE Transactions on Services Computing* 14.6 (2019), pp. 2073–2089.

[422] Xuezhen Tu, Kun Zhu, Nguyen Cong Luong, Dusit Niyato, Yang Zhang, and Juan Li. "Incentive mechanisms for federated learning: From economic and game theoretic perspective". In: *IEEE transactions on cognitive communications and networking* 8.3 (2022), pp. 1566–1593.

[423] Naeem Ullah, Javed Ali Khan, Ivanoe De Falco, and Giovanna Sannino. "Explainable artificial intelligence: importance, use domains, stages, output shapes, and challenges". In: *ACM Computing Surveys* 57.4 (2024), pp. 1–36.

[424] European Union. *Article 16: Obligations of Providers of High-Risk AI Systems*. Accessed: 2025-02-20. 2024. URL: https://artificialintelligenceact.eu/article/16/.

[425] European Union. *Recital 53 - Artificial Intelligence Act*. Accessed: 2025-02-19. 2025. URL: https://artificialintelligenceact.eu/recital/53/.

[426] United Nations Educational, Scientific and Cultural Organization (UNESCO). *Recommendation on the Ethics of Artificial Intelligence*. Adopted on November 23, 2021, by UNESCO's General Conference. 2021. URL: https://unesdoc.unesco.org/ark:/48223/pf0000381137.

[427] University of Tartu. *UT Rocket*. 2018. DOI: 10.23673/PH6N-0144. URL: share.neic.no.

[428] Dmitrii Usynin, Alexander Ziller, Marcus Makowski, Rickmer Braren, Daniel Rueckert, Ben Glocker, Georgios Kaissis, and Jonathan Passerat-Palmbach. "Adversarial interference and its mitigations in privacy-preserving collaborative machine learning". In: *Nature Machine Intelligence* 3.9 (2021), pp. 749–758.

[429] Aki Vehtari, Andrew Gelman, and Jonah Gabry. "Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC". In: *Statistics and computing* 27 (2017), pp. 1413–1432.

[430] Joost Verbraeken et al. "A survey on distributed machine learning". In: *ACM Computing Surveys* 53.2 (2020), pp. 1–33.

[431] Oleksandra Vereschak, Fatemeh Alizadeh, Gilles Bailly, and Baptiste Caramiaux. "Trust in AI-assisted Decision Making: Perspectives from Those Behind the System and Those for Whom the Decision is Made". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–14.

[432] Oleksandra Vereschak, Gilles Bailly, and Baptiste Caramiaux. "How to evaluate trust in AI-assisted decision making? A survey of empirical

methodologies". In: *Proceedings of the ACM on Human-Computer Interaction* 5.CSCW2 (2021), pp. 1–39.

[433] Andreas Vogelsang et al. "Requirements engineering for machine learning: Perspectives from data scientists". In: *IEEE 27th International Requirements Engineering Conference Workshops (REW)*. IEEE. 2019, pp. 245–251.

[434] Sandra Wachter, Brent Mittelstadt, and Chris Russell. "Why fairness cannot be automated: Bridging the gap between EU non-discrimination law and AI". In: *Computer Law & Security Review* 41 (2021), p. 105567.

[435] Daisuke Wakabayashi. "Self-driving Uber car kills pedestrian in Arizona, where robots roam". In: *The New York Times* 19.03 (2018).

[436] Guotai Wang, Wenqi Li, Maria A Zuluaga, Rosalind Pratt, Premal A Patel, Michael Aertsen, Tom Doel, Anna L David, Jan Deprest, Sébastien Ourselin, et al. "Interactive medical image segmentation using deep learning with image-specific fine tuning". In: *IEEE transactions on medical imaging* 37.7 (2018), pp. 1562–1573.

[437] Qun Wang, Haoxuan Dong, Fei Ju, Weichao Zhuang, Chen Lv, Liangmo Wang, and Ziyou Song. "Adaptive leading cruise control in mixed traffic considering human behavioral diversity". In: *IEEE Transactions on Intelligent Transportation Systems* 25.6 (2023), pp. 5059–5070.

[438] Shujuan Wang et al. "Occluded Person Re-Identification via Defending Against Attacks From Obstacles". In: *IEEE Transactions on Information Forensics and Security* 18 (2022), pp. 147–161.

[439] Wenshuo Wang, Junqiang Xi, Chang Liu, and Xiaohan Li. "Human-centered feed-forward control of a vehicle steering system based on a driver's path-following characteristics". In: *IEEE transactions on intelligent transportation systems* 18.6 (2016), pp. 1440–1453.

[440] Xianmin Wang, Jing Li, Xiaohui Kuang, Yu-an Tan, and Jin Li. "The security of machine learning in an adversarial setting: A survey". In: *Journal of Parallel and Distributed Computing* 130 (2019), pp. 12–23.

[441] Xinru Wang, Hannah Kim, Sajjadur Rahman, Kushan Mitra, and Zhengjie Miao. "Human-LLM collaborative annotation through effective verification of LLM labels". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–21.

[442] Yifei Wang. "Balancing Trustworthiness and Efficiency in Artificial Intelligence Systems: An Analysis of Tradeoffs and Strategies". In: *IEEE Internet Computing* (2023).

[443] Yuntao Wang et al. "Learning in the Air: Secure Federated Learning for UAV-Assisted Crowdsensing". In: *IEEE Trans. Netw. Sci. Eng.* 8.2 (2021), pp. 1055–1069. DOI: 10.1109/TNSE.2020.3014385.

[444] Sandamal Weerasinghe, Tansu Alpcan, Sarah M Erfani, and Christopher Leckie. "Defending support vector machines against data poisoning at-

tacks". In: *IEEE Transactions on Information Forensics and Security* 16 (2021), pp. 2566–2578.

[445] Wenqi Wei and Ling Liu. "Trustworthy distributed ai systems: Robustness, privacy, and governance". In: *ACM Computing Surveys* (2024).

[446] Daniel Karl I Weidele, Shazia Afzal, Abel N Valente, Cole Makuch, Owen Cornec, Long Vu, Dharmashankar Subramanian, Werner Geyer, Rahul Nair, Inge Vejsbjerg, et al. "AutoDOViz: Human-Centered Automation for Decision Optimization". In: *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 2023, pp. 664–680.

[447] Chathurika S. Wickramasinghe, 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)*. 2020, pp. 130–136. DOI: 10.1109/HSI49210.2020.9142644.

[448] Christo Wilson, Avijit Ghosh, Shan Jiang, Alan Mislove, Lewis Baker, Janelle Szary, Kelly Trindel, and Frida Polli. "Building and auditing fair algorithms: A case study in candidate screening". In: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. 2021, pp. 666–677.

[449] Jeannette M Wing. "Trustworthy ai". In: *Communications of the ACM* 64.10 (2021), pp. 64–71.

[450] Jeannette M. Wing. "Trustworthy AI". In: *Commun. ACM* 64.10 (Sept. 2021), pp. 64–71. ISSN: 0001-0782. DOI: 10.1145/3448248. URL: https://doi.org/10.1145/3448248.

[451] Anna Wojciechowska et al. "Designing drones: Factors and characteristics influencing the perception of flying robots". In: *Proceedings of IMWUT 2019* 3.3 (), pp. 1–19.

[452] World Economic Forum. *How Much Data is Generated Each Day?* Accessed: 2025-04-02. 2019. URL: https://www.weforum.org/stories/2019/04/how-much-data-is-generated-each-day-cf4bddf29f/.

[453] Sherry Wu, Hua Shen, Daniel S Weld, Jeffrey Heer, and Marco Tulio Ribeiro. "Scattershot: Interactive in-context example curation for text transformation". In: *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 2023, pp. 353–367.

[454] Tingting Wu, Yunwei Dong, Zhiwei Dong, Aziz Singa, Xiong Chen, and Yu Zhang. "Testing Artificial Intelligence System Towards Safety and Robustness: State of the Art." In: *IAENG International Journal of Computer Science* 47.3 (2020).

[455] Bo Xiao and Izak Benbasat. "E-commerce product recommendation agents: Use, characteristics, and impact". In: *MIS quarterly* (2007), pp. 137–209.

[456] Xianghua Xie, Chen Hu, Hanchi Ren, and Jingjing Deng. “A survey on vulnerability of federated learning: A learning algorithm perspective”. In: *Neurocomputing* (2024), p. 127225.

[457] Wenpeng Xing, Minghao Li, Mohan Li, and Meng Han. “Towards Robust and Secure Embodied AI: A Survey on Vulnerabilities and Attacks”. In: *arXiv preprint arXiv:2502.13175* (2025).

[458] Pulei Xiong, Scott Buffett, Shahrear Iqbal, Philippe Lamontagne, Mohammad Mamun, and Heather Molyneaux. “Towards a robust and trustworthy machine learning system development: An engineering perspective”. In: *Journal of Information Security and Applications* 65 (2022), p. 103121. ISSN: 2214-2126. DOI: https://doi.org/10.1016/j.jisa.2022.103121. URL: https://www.sciencedirect.com/science/article/pii/S2214212622000138.

[459] Guowen Xu et al. “Secure and verifiable inference in deep neural networks”. In: *ACSAC*. 2020, pp. 784–797.

[460] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. “Modeling tabular data using conditional gan”. In: *Advances in neural information processing systems* 32 (2019).

[461] Maria Yagoda. *Airline held liable for its chatbot giving passenger bad advice – what this means for travellers*. Accessed: 2025-06-18. Feb. 2024. URL: https://www.bbc.com/travel/article/20240222-air-canada-chatbot-misinformation-what-travellers-should-know.

[462] Pan Yang, Naixue Xiong, and Jingli Ren. “Data security and privacy protection for cloud storage: A survey”. In: *Ieee Access* 8 (2020), pp. 131723–131740.

[463] Qiang Yang et al. “Federated machine learning: Concept and applications”. In: *ACM Trans. Intell. Syst. Technol* 10.2 (2019), pp. 1–19.

[464] Jingjing Yao. “Split learning for image classification in Internet of drones networks”. In: *2023 IEEE 24th International Conference on High Performance Switching and Routing (HPSR)*. IEEE. 2023, pp. 52–55.

[465] Shuochao Yao et al. “Deepsense: A unified deep learning framework for time-series mobile sensing data processing”. In: *Proceedings of the 26th WWW*. 2017, pp. 351–360.

[466] Anam Yasir, Alia Ahmad, Sagheer Abbas, Mohammad Inairat, Amer Hani Al-Kassem, and Atta Rasool. “How artificial intelligence is promoting financial inclusion? A study on barriers of financial inclusion”. In: *2022 International Conference on Business Analytics for Technology and Security (ICBATS)*. IEEE. 2022, pp. 1–6.

[467] Dong Yin et al. “Byzantine-robust distributed learning: Towards optimal statistical rates”. In: *ICML*. PMLR. 2018, pp. 5650–5659.

[468] Zhigang Yin, Mayowa Olapade, Mohan Liyanage, Farooq Dar, Agustin Zuniga, Naser Hossein Motlagh, Xiang Su, Sasu Tarkoma, Pan Hui, Petteri Nurmi, and Huber Flores. "Toward City-Scale Litter Monitoring Using Autonomous Ground Vehicles". In: *IEEE Pervasive Comput.* (2022).

[469] Han Yu et al. "A fairness-aware incentive scheme for federated learning". In: *Proceedings of the AAAI/ACM AIES*. 2020, pp. 393–399.

[470] Binhang Yuan, Yongjun He, Jared Davis, Tianyi Zhang, Tri Dao, Beidi Chen, Percy S Liang, Christopher Re, and Ce Zhang. "Decentralized training of foundation models in heterogeneous environments". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 25464–25477.

[471] Tonfi Yuki, Masayuki Okamoto, Hiraku Murayama, Kirihito Yajima, and Yuta Nishigaki. *AI Algorithm Transparency Toolkit: A Proposal for a Governance System to Enable Society to Accept and Benefit from AI-based Innovations*. Tech. rep. Graduate School of Public Policy, The University of Tokyo, 2023. URL: `https://www.pp.u-tokyo.ac.jp/cregg/assets/img/program/expert/report-document-20230407_EN.pdf`.

[472] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. "Cutmix: Regularization strategy to train strong classifiers with localizable features". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 6023–6032.

[473] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. "Fairness constraints: Mechanisms for fair classification". In: *Artificial intelligence and statistics*. PMLR. 2017, pp. 962–970.

[474] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. "Fairness constraints: Mechanisms for fair classification". In: *Artificial intelligence and statistics*. PMLR. 2017, pp. 962–970.

[475] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks". In: *ECCV*. Springer. 2014, pp. 818–833.

[476] Tengchan Zeng, Omid Semiari, Mohammad Mozaffari, Mingzhe Chen, Walid Saad, and Mehdi Bennis. "Federated learning in the sky: Joint power allocation and scheduling with UAV swarms". In: *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE. 2020, pp. 1–6.

[477] Xingchen Zeng, Ziyao Gao, Yilin Ye, and Wei Zeng. "IntentTuner: An Interactive Framework for Integrating Human Intentions in Fine-tuning Text-to-Image Generative Models". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–18.

[478] Dongping Zhang, Angelos Chatzimparmpas, Negar Kamali, and Jessica Hullman. "Evaluating the utility of conformal prediction sets for ai-advised image labeling". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–19.

[479] Shanshan Zhang, Lihong He, Eduard Dragut, and Slobodan Vucetic. “How to invest my time: Lessons from human-in-the-loop entity extraction”. In: *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2019, pp. 2305–2313.

[480] Sicong Zhang, Hui Yang, and Lisa Singh. “Anonymizing query logs by differential privacy”. In: *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 2016, pp. 753–756.

[481] Yifei Zhang, Dun Zeng, Jinglong Luo, Xinyu Fu, Guanzhong Chen, Zenglin Xu, and Irwin King. “A survey of trustworthy federated learning: Issues, solutions, and challenges”. In: *ACM Transactions on Intelligent Systems and Technology* 15.6 (2024), pp. 1–47.

[482] Yifei Zhang, Dun Zeng, Jinglong Luo, Zenglin Xu, and Irwin King. “A survey of trustworthy federated learning with perspectives on security, robustness and privacy”. In: *Companion Proceedings of the ACM Web Conference 2023*. 2023, pp. 1167–1176.

[483] Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. “The secret revealer: Generative model-inversion attacks against deep neural networks”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 253–261.

[484] Yunfeng Zhang, Rachel Bellamy, and Kush Varshney. “Joint optimization of AI fairness and utility: a human-centered approach”. In: *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 2020, pp. 400–406.

[485] Zaixi Zhang, Xiaoyu Cao, Jinyuan Jia, and Neil Zhenqiang Gong. “Fldetector: Defending federated learning against model poisoning attacks via detecting malicious clients”. In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2022, pp. 2545–2555.

[486] Lingchen Zhao et al. “Shielding collaborative learning: Mitigating poisoning attacks through client-side detection”. In: *IEEE Transactions on Dependable and Secure Computing* (2020).

[487] Tianyu Zhao, Mojtaba Taherisadr, and Salma Elmalaki. “Fairo: Fairness-aware sequential decision making for human-in-the-loop cps”. In: *2024 ACM/IEEE 15th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE. 2024, pp. 87–98.

[488] Chen Zhu, W Ronny Huang, Hengduo Li, Gavin Taylor, Christoph Studer, and Tom Goldstein. “Transferable clean-label poisoning attacks on deep neural nets”. In: *International Conference on Machine Learning*. PMLR. 2019, pp. 7614–7623.

[489] Hangyu Zhu et al. “Federated learning on non-IID data: A survey”. In: *Neurocomputing* 465 (2021), pp. 371–390.

[490] Yan Zhuang, Guoliang Li, Zhuojian Zhong, and Jianhua Feng. “Hike: A hybrid human-machine method for entity alignment in large-scale knowledge bases”. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 2017, pp. 1917–1926.

Appendix A. ONE TO RULE THEM ALL: A STUDY ON REQUIREMENT MANAGEMENT TOOLS FOR THE DEVELOPMENT OF MODERN AI-BASED SOFTWARE

A.1. Introduction

Modern system architectures are rapidly adopting AI-based functionality [310]. Indeed, advanced machine and deep learning models can improve the usability and performance of the applications that we use in our daily life activities. For instance, transportation systems can rely on AI for better navigation of autonomous vehicles, and healthcare can improve the accuracy of diagnosis for different diseases using AI [383, 371, 191]. A major challenge to integrating AI into the software development life cycle is that standard pipelines for building AI models can be easily hampered (through adversarial of situational changes) at any phase of the model construction [433, 11, 113], e.g, data poisoning during data collection. Besides this, the resulting AI model is black-box, meaning that the logic used to make a decision is obscured to users. Another important challenge is to make AI trustworthy for its adoption at scale [449]. Ensuring that AI is trustworthy requires taking into consideration several trustworthy properties as early as the conceptual designs defining the applications conceived. Trustworthy properties of AI are a set of characteristics that AI requires to be equipped with, for instance, explainability, interpretability, managing biases, data governance, privacy enhancement, and safety, to mention some. Tracking the fulfillment of these characteristics over time and their changes require the use of specialized management tools.

Regulations defined for the development of AI-based software, e.g., GDPR [251] and US AI Executive Order (EO) 13960 [106]; have introduced guiding principles imposing ethical, lawful, and robust considerations for the development and deployment of AI-based solutions. As early conceptual and blueprint decisions can influence the fulfillment of these regulations in practice, trustworthy properties are required to be considered as early as the requirements collection phase. Indeed, requirements can be greatly affected due to the sensitivity and privacy of data that is used to train/retrain AI models. For instance, the use of data in mobile applications becomes situational, requiring, in some cases, consent from surrounding individuals to use their data [147]. As requirements capture private related information from users, this should be reflected throughout the development process. Likewise, the type of AI algorithm selected can also influence the post-de facto verification of AI behavior, making it difficult to assess characteristics, such as transparency and resilience. Another example is tracking specifications regarding the resilience of AI models to the large spectrum of possible attacks that influence their decision process [385], e.g., model evasion and model stealing. As a result, the monitoring and tracking of AI requirements becomes critical for the auditability of accountability of developed AI-based software. While there

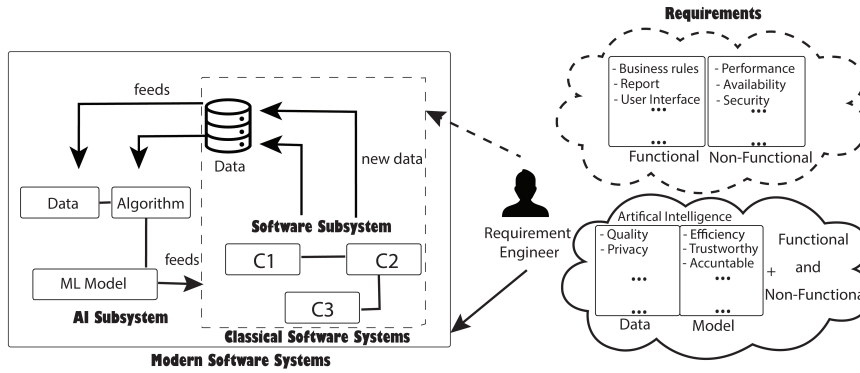


Figure 47: Conceptual modern software architectures implementing machine learning [310] and set of evolving AI requirements to be tracked and monitored.

is a large variety of (open source and off-the-shelf commodity) tools available for tracking requirements and monitoring them over time [433, 230], not all the tools provide flexibility and adaptability to cope with dynamic changes in data and model characteristics. More importantly, besides the established regulations, other stakeholder considerations also become important for the selection of the management tool as the construction process of AI may require to apply different internal policies affecting each involved stakeholder.

We contribute by analyzing the suitability of different requirement management tools. As large software applications are typically built in large teams, located in different areas, and belonging to different organizations. To evaluate the effectiveness of different management tools, we conduct the analysis in the context of an EU Horizon project formed by a consortium of industry and academic partners located across different EU and non-EU countries. This analysis is conducted using the Commercial Off-The-Shelf (COTS) methodology. By using COTS, we assessed whether popular and off-the-shelf commercial tools can be used easily to manage and track the requirements of AI-based software. In this analysis, five technical partners are selected, where one is appointed as a coordinator, defining the overall criteria for evaluating the tool. The coordinator is also responsible for assigning each partner a tool, such that each individual partner can evaluate the tool against the criteria. The results of our analysis demonstrate that new regulations imposed for the development of AI reduced the pool of options available for handling requirements. Besides this, our results demonstrate that while internal policies of organizations may also be a problem when selecting requirements management tools, it is much easier to bypass them when compared with rules imposed by regulatory entities. Lastly, we also share our lessons learned and experiences from selecting requirement tools that can be used in team-based consortium projects.

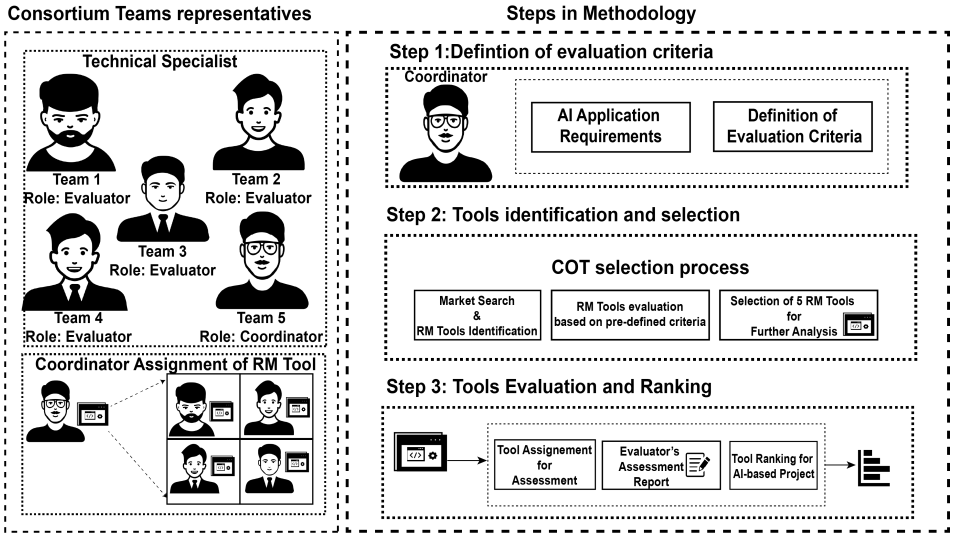


Figure 48: Evaluation methodology.

Requirement type	Feature	Description
Functional	Business rules	Transactional functions that must be performed in the end product based on user stories
	External interfaces	API design and accessibility to external APIs
	Access control	Authentication of end users
Non-functional	Performance	Response time of application
	Availability	Continuous availability of the system for users.
	Scaleability	Ability to perform as expected under load.
	Usability	user-friendly and easy to use system.
	Security	Authorization and resource access control

Table 19: Generic requirements of applications

A.2. Evolution in software requirements

Classical software requirements: Majority of classical software requirements focus on providing functional and non-functional features according to the client's needs. As shown in Table 19, these requirements ensure that the applications meet the behavior and contain features required during the software development process. Functional requirements range from use case business rules and transactional correctness to user interfaces that enable the client to achieve a higher user experience. On the other hand, non-functional requirements are quality attributes that determine how the system should behave after deployment. For instance, even if the functionality is as expected, having poor performance after deployment may lead to a negative user experience and jeopardize system safety. Table 19 describes the classical software requirements in detail.

GDPR and imposed regulations: Europe is leading the verification of AI-based solutions, such that AI is trustworthy to users. The General Data Protection Reg-

ulations (GDPR) stipulates the guidelines for dealing with personal data within the European Union (EU), putting forward fairness, security, privacy, trust, transparency, and explanation considerations during software and AI-based solution development. Practically, the guidelines have compliance implications on data and software architecture [201, 14] as they obligate these considerations for validating and verifying that the software solutions and systems are rightly and correctly developed [262, 197]. As a result, software engineers, requirement engineers, and other practitioners now have to consider a new set of requirements, like data traceability, minimization, rectification, and erasure, system security, and privacy, that have been imposed by the regulations as part of system requirements during the system design and development. Similarly, these principles are also described in the US AI ACT, and other countries have also considered similar regulations, for instance, China, Japan, Brazil, and Canada.

Modern software requirements: The inclusion of AI functionality in applications has changed significantly the elicitation of requirements when designing software [310]. As shown in Figure 47, the current software systems leverage AI (e.g., machine learning), leading to a revolution in intelligent software design. Since AI modules within software mainly rely on data, modern software requirements change accordingly. As shown in Table 20, in addition to current classical software requirements, AI-enabled systems raise a set of new requirements related to data and the internal model logic of AI. For instance, AI functionality highly relies upon the quantity and quality of data fed to the system. Moreover, keeping an individual's privacy during data collection should be considered by AI-driven software engineers.

Requirement type	Feature	Description
Data	Privacy	Keeping private data anonymous
	Quality	Ensure quality of data before model fitting
	Bias-free	Completeness of data
Model	Efficiency	To perform efficient in terms of energy / time
	Robustness	To be robust against adversaries
	Trustworthiness	Transparent, explainable, and accountable AI
System	Resilience	Perform as expected in every circumstance
	Reliability	The probability that AI-based system performs correctly for a time period.
	Safety	Minimize harmful consequences

Table 20: AI-based application requirements

A.3. Analysis of RM tools: methodology

Assumptions: The analysis of RM tools is conducted in the context of a project that aims to design and develop an AI-based solution. Our project brings together specialists from the academy and industry to build a software solution that can be used to analyze AI models running in modern applications. During the development

Criteria	Tools				
	OSRMT [396]	RMTOO [151]	**OpenReq [145]**	**Doorstop [77]**	**CAIRIS [141]**
Extensibility	✓	✓	✓	✓	✓
Local Installation	✓	✗	✗	✓	✓
Traceability	✓	✗	✗	✓	✓
GDPR Compliance	✗	✗	✗	✗	✓
Export Capability	✗	✗	✓	✓	✓

Table 21: Requirement management tools criteria

of our solution, we assume that each team can access the overall code and contribute to the development of the application. In addition to this, once a tool is selected for the joint development of the solution, we also assume that requirements are tuned over time solely by the coordinator of the analysis, such that issues about accessing data by other teams do not infringe our analysis criteria.

Methodology: Our analysis is conducted by relying on the principles of COTS methodology. Figure 48 describes the overall process. By using COTS, we selected a reasonable number of RM tools and applied clear steps to conduct individual evaluations of each tool by different teams. To use COTS, first, we defined specific criteria for establishing the baseline features for the RM tools to be considered. Subsequently, we searched to identify relevant RM tools that align with the pre-defined criteria. Right after, we then systematically assessed each tool by relying on individual evaluations conducted by individual teams over assigned tools. Next, we explain the step-by-step procedure in detail as follows:

A.3.1. Step 1: Definition of evaluation criteria

After identifying the essential functionalities of our AI-based application within the context of our project. We adopted the comprehensive and formal guideline provided by **ISO/IEC TR 24766:2009** [216] framework. The framework enabled us to understand the mapping of requirement management activities to the capabilities of any RM tools as the standard is the classification framework for evaluating relevant features of RM tools [120]. In addition, this framework availed us with crucial insights into the specific capabilities that candidate RM tools should possess. Following that, we established workshops and discussion sessions to elaborate on other criteria that RM tools must possess to serve our purpose for effective support of AI-based application development. Overall, we conducted six (6) workshops with a duration of forty (40) minutes. These criteria are defined as follows:

(a) Extensibility: RM tools that are open source and publicly accessible are relevant to our overall objective. This allows the development team to modify and extend the code as needed. Although these tools rely on community production and peer review, they are often more affordable and flexible and do not require additional production team costs to manage requirements. Moreover, this feature helps us to have an extensibility feature that enables us to adapt the RM tools and

extend them to meet the specific needs of different stakeholders. Furthermore, the criterion demands that the RM tools to be considered must have detailed and comprehensible documentation about object models, interfaces, and API such that data, requirement artifacts, and functions can be adapted and extended.

(b) Textual requirements: Textual requirements are text statements about the features or functionality of the application under development. The ability of users to use text is crucial for RM tools. The needs and expectations of stakeholders are easily captured by way of text. The captured information provides the basis for application design and development.

(c) Exportability: Exportability criterion relates to the ability to export data in different formats. This capability ensures that users can access and share requirements information outsides of the RM tool environment with third-party applications or relevant stakeholders without access to the tools.

(d) GDPR compatibility: AI-based applications use a lot of sensitive data as such, compliance with privacy regulations is crucial for RM tools deployed for managing the requirements of AI-based applications. RM tools must be designed to address privacy and data protection concerns and comply with GDPR and other relevant regulations throughout the requirement management process and application development lifecycle. RM tools to consider must possess functionalities that can guarantee the confidentiality and integrity of sensitive information. At the basic level, the tools must sufficiently pass the GDPR principles described in Table 22.

GDPR Principle	**Description**
Lawfulness, Fairness, and Transparent	Data with privacy properties are processed only if it's recognized as personal data. Use cases involving personal data are associated with a necessary goal or requirement.
Purpose Limitation	Use cases involving personal data are associated with a necessary goal concerned with that personal data.
Data Minimisation	Data with privacy properties are accounted for in processes.
Accuracy	Personal data has an Integrity security property.
Storage Limitation	Personal data in data stores is processed.
Integrity & Confidentiality	Personal information has confidentiality, integrity, and privacy properties that must be preserved.

Table 22: GDPR principles and descriptions [141]

(e) Local installation: Since it is important to keep the confidential data of the project on local servers, it is important to have the tools installed locally and not on outsourced and untrusted servers.

(f) Requirements traceability: Traceability of elements is vital for requirement monitoring. It encompasses the specification and the tracking of the relationships between system elements [179, 382].

A.3.2. Step 2: Tools identification and selection

In this step, we focused on tool identification and assessment. To identify a suitable set of RM tools, we conducted a search of available options in the market. We discovered that a variety of RM tools are available with different features and functionalities. However, considering the time limitations and the overwhelming number of options, it was impractical to assess all the identified tools. Hence, we adopted the COTS method alongside the criteria defined in Step 1 for selecting a cross-section of the identified tools for further evaluation of their capabilities in managing requirements in the context of AI-based system development.

A.3.3. Step 3: Tools evaluation

After several iterations of assessment of identified tools based on pre-defined criteria, we selected the five (5) tools that can be adapted for our AI-based application for further evaluation with stakeholders involved in the application development. Each of the selected tools was evaluated in detail against the criteria described in Step 2. We randomly distributed the tools among different project stakeholders and asked them to evaluate the assigned tool based on the proposed criteria. The criteria of the evaluation were also explained in detail to each of the teams assessing the tools. Table 21 shows the evaluation outcomes and demonstrates if the tools comply with the defined criteria.

A.3.4. Threats to validity

The criteria for the selection of the RM tools and evaluators is based on the consortium's requirement management needs and the stakeholders, the potential domain experts and practitioners, who are the users of the tool within the context of our EU Horizon project. There are quite a large number of RM tools available in the market, not all of which would fit the peculiarity of our project needs. We then relied on the expertise and experience of the consortium stakeholders to establish the criteria for the selection and evaluation of candidate tools to ensure that the necessary criteria required to meet the project demands were sufficiently satisfied. To minimize the potential reactivity threat, where one evaluator's judgment could influence another, each team installed the assigned tool locally and extensively assessed it over a period of three months. This allows for teams' independent judgment within a reasonable period of observation based on the established criteria. However, we acknowledge the limited generalizability of our findings due to the relatively small sample of RM tools considered, the number of evaluators involved, and the specific evaluation context. Despite these and other limitations, our evaluation provides valuable insights into our project's specific needs and contributes to the understanding of RM tools' capabilities in the context of AI-based application development.

A.4. Quantitative analysis of the tools

Next, a quantitative analysis is conducted to quantify each relative aspect that conforms to the overall analysis criteria.

Experimental setup: We conducted a survey that captures the evaluation of each team with its respective assigned tool. We utilized a questionnaire as the main instrument for systematically collecting evaluation information about the specified criteria. We designed our questionnaire following the standard requirement evaluation questionnaire as presented in [84, 109], which also follows the ISO/IEC TR 24766:2009 standard, making emphasis in the key aspects defined in our criteria. Five-point Likert-like Scale response (5: Strongly Agree, 4: Agree, 3: Cannot say, 2: Disagree, 1: Strongly Disagree [314] was provided to quantify the questions. Examples of representative questions include, *Can we extend the tool based on the documentation and guidelines?*; *Please rank the tool based on the "Ease of Local Deployment criterion."* and *Is it easy to deploy the tool on our local servers?*.

Results: Although most of the tools that we have considered were open-sourced, we can observe from Figure 50(a) that about 60% of tools have inadequate documentation or have poor guidelines that result in a low score of ease of extensibility. Moreover, as shown in Figure 50(b), Local Deployment has mostly received an agreement of 80% among partners, but only OpenReq did not comply with this feature, resulting in a Disagreement of 20%. It is also observable that traceability was among the features that were considered in most of the tools (40% Agree and 20% Strongly Agree, see Figure 50(c)). It is also worth noting that RM tools require more attention to clarify how the tool adheres to the GDPR principles since the evaluators were not able to analyze this aspect clearly (80% Neutral - see Figure 50(d)).

Figure 49(a) shows the distribution of scores by RM tools based on predefined criteria. The figure confirms the results of Figure 50 and shows how features are ranked for each tool. From Figure 49(b), it is also observable that CAIRIS achieved the highest scores of the analyzed criteria and, most notably, its compliance with GDPR because of its rich documentation and elaboration on GDPR principles. The tools that come next in rankings are Doorstop and RMTOO. The lower rate to RMTOO and Doorstop was mainly because of the partners' Disagreement with their ease of extensibility and GDPR compliance, resulting from poor documentation and interfaces. Accordingly, as shown in the figure, OpenReq is the tool that received the least score, and it is because of the lack of maintenance since 2019. It seems that the deployment of the tool on local premises was not easy because of outdated implementation and was not recommended for further usage in AI-based application requirement management.Taken together, our results suggest that CAIRIS provides the most optimal support for tracking and monitoring the requirements of AI-based software.

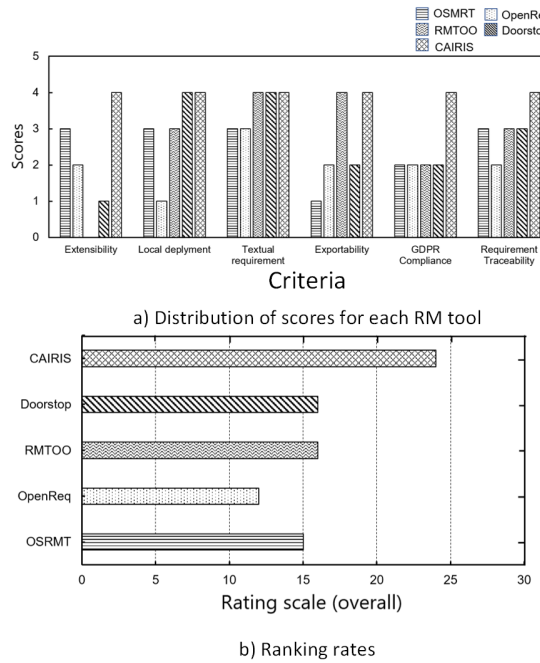


Figure 49: Evaluation of tools based on defined criteria

A.5. Qualitative analysis of the tools

Each team provided first a qualitative analysis of the assigned tool. This analysis contained a description of the tool and qualitative information regarding how the tool fulfilled the established criteria.

(a) OSRMT (Team 1): The open-source requirements management tool (OSRMT) is an open-source requirement management tool with GNU General Public License version 2.0 (GPLv2). It offers full capabilities for defining and managing requirements for software development. It can be installed as a single-user desktop application or a multiple-user application with a centralized server.

- **Extensibility:** OSMRT is available free and open to any user. The tool's license allows users to use the tool for any purpose, access the source code and modify it for specific use. This avails OSMRT with a large community of users to collaborate and contribute towards its development and potential for wider adoption.
- **Traceability:** The tool offers traceability capabilities that enable users to create relationships between artifacts and requirements, allowing for requirement analysis and change management. Stakeholders can visualize the traceability to aid their understanding and decision-making.
- **GDPR compliant:** Information about the tool's compliance with the General Data Protection Regulation (GDPR) is not available in its GitHub repository.

- **Textual requirement:** OSRMT supports textual requirements in various forms. Users have the ability to create and manage five fundamental artifacts as baseline entities: Features, Requirements, Design Modules, Implementations, and Test Cases. These artifacts serve as essential components within the requirements management process, enabling effective organization and traceability of the software development project.
- **Exportability:** OSMRT allows the export of artifacts to XML only. However, exporting to another format such as Excel/CSV, or PDF is not possible.

(b) RMTOO (Team 2): RMTOO is a requirement management tool that is implemented in Python and designed to fully support the Linux operating system. The tool utilizes a command line interface, which consolidates input and output operations within a single environment, thus enhancing the efficiency of requirements handling. While RMTOO primarily caters to Linux, its implementation in a computer-independent programming language also enables its usage on other operating systems. Local installation can be accomplished by employing a virtual machine along with the necessary dependencies, such as Python version 3.5 and above, Latex, grahpviz, and gnuplot [151].

- **Extensibility:** RMTOO is available to the public for use under the GNU General Public License version 3.0 (GPLv3) license, an improved version of GPLv2. This implies that the license is not restricted only to the codes but also extends to any hardware or related works that use the source code of RMTOO. However, the compatibility of RMTOO is relatively strict due to its license as GPLv3 adheres to stricter compatibility guidelines which have an implication on the extensibility of the tool's source code to other projects. This necessitates that the source code can only be combined with projects with similar licenses i.e. GPLv3 or another compatible license. Thereby restricting the options for combining RMTOO with other projects that are without compatible licenses.
- **Traceability:** Linkage between requirement artifacts is visually represented using various colors in RMTOO. In RMTOO, requirement artifacts are distinctly identified with unique names and connections are color-coded, providing a visual form of traceability. However, the specifics regarding how RMTOO implements and facilitates traceability cannot be found in its repository [151].
- **GDPR compliant:** The available documentation and repository for RMTOO do not provide explicit information concerning the privacy and security measures implemented to protect user data. As a result, the tool's compliance with the GDPR cannot be verified or confirmed due to the absence of specific details regarding data privacy practices.
- **Textual requirement:** RMTOO is a comprehensive text-based requirement management tool, enabling users to seamlessly create requirement artifacts using text prompts throughout the entire process. The tool empowers users

to efficiently generate requirement artifacts by utilizing text-based inputs, facilitating a streamlined and intuitive experience for requirement creation. Through its text-based interface, RMTOO ensures that users can effortlessly capture, organize, and manage requirements using textual representations, ensuring clarity and consistency throughout the requirement management lifecycle.

- **Exportability:** RMTOO offers versatile output capabilities, allowing users to generate output in multiple file formats, including PDF, HTML, and visualization formats. Users can leverage this flexibility to select the format that best suits their needs for sharing and presenting requirements information. Moreover, RMTOO facilitates seamless export of the generated output into any of the supported formats, allowing users to utilize the output in their preferred format for further analysis, distribution, or documentation purposes.

(c) OpenReq (Team 3): OpenReq requirement management tools aim to utilize modern recommender algorithms to develop intelligent recommendation and decision technologies that support different phases of requirement engineering such as elicitation, specification, analysis, management, and negotiation. It proposes to function as a support tool that relies heavily on artificial intelligence (AI) for driving innovative and efficient requirement management. However, the tool is at the moment still a prototype that is yet to evolve into a full fledge finished RM tool.

- **Extensibility:** OpenReq is an open-source platform, which means it is freely available and can be customized and extended to meet specific project needs. The open-source nature of the platform encourages community contributions, fosters innovation, and allows for continuous improvement over time.
- **Traceability:** Information about the traceability functionality of Open Req is not available in the existing repository and demo [61]
- **GDPR compliant:** The current prototype has no data management and protection information.
- **Textual requirement:** OpenReq possess detailed and well-developed textual capabilities that enable users to create, organize, and manage requirement as well as related artifacts. In addition, the textual capabilities of the tool support collaboration among stakeholders and enable the recommender algorithms model to learn and decision-making.
- **Exportability:** The tool supports the generation of PDF output and visualization. However, the exportability of output is unknown at the moment.

(d) Doorstop (Team 4): Doorstop is essentially a Python library that helps store and manage textual files with source code for requirement management control. It requires Python 3.7+ and a version control system for requirements storage. The library can be installed locally using the pip function in the terminal and could be accessed via the terminal or imported in Python script via the import command. The source code is available on GitHub [76].

- **Extensibility:** Doorstop is distributed under the GNU Lesser General Public License (LGPLv3), which implies a "weak copyleft" license. This designation indicates that users are permitted to utilize, distribute, and modify the library's code under this license. However, users are obligated to include a complete copy of the license and the original copyright notice. In the case of derivative works, the original source code must be included or made accessible, and the derived work must be licensed under the same LGPLv3 license and not previous versions [76].
- **Traceability:** Every Doorstop file is stored inside the version control repository. Every file is assigned a unique name sequentially numbered, allowing easier linking and historical review.
- **GDPR compliant:** Doorstop prioritizes security in its available documentation. However, the documentation did not provide details about compliance with GDPR.
- **Textual requirement:** Doorstop facilitates the use of human-readable text files, for instance, YAML files, which can be easily interpreted by individuals and accessed using standard text editors. Furthermore, these text files can be parsed effortlessly using Python libraries.
- **Exportability:** Requirement artifacts can be exported for editing and exchange with other systems in different formats such as YAML (.yml), Comma-separated Values (.csv), Tab-separated Values (.tsv), and Microsoft Excel file (.xlsx). It is also possible to create requirements using any of the formats.

(e) CAIRIS (Team 5): CAIRIS (Computer Aided Integration of Requirement and Information Security) is an open-source platform for eliciting, specifying, and validating secure and usable systems. CAIRIS was developed by Shamal Faily to aid designers in integrating security and usability requirements during application development [141]. CAIRIS is compatible with mainstream operating systems, Windows and Linux, and is installable through GitHub, Virtualbox, and Vagrant.

- **Extensibility:** CAIRIS is available under the GNU Affero General Public License (AGPL). Under this license, the tool is modifiable and distributable by users. CAIRIS source codes are freely accessible. Users can collaboratively contribute to development. Being an open-source tool, the source codes can be reviewed, modified, improved, and regularly updated by a large community of developers and users. Developers and users can build on the existing functionalities of CAIRIS and extend them to suit their distinct needs.
- **Local Installation:** CAIRIS can be installed on local servers, desktops, or laptops and supports various operating systems. Users can install CAIRIS locally through GitHub, Docker, and Vagrant. Multiple users can install CAIRIS locally in different locations and can collaborate on the same project.
- **Traceability:** In CAIRIS, traceability is automated. CAIRIS employs the

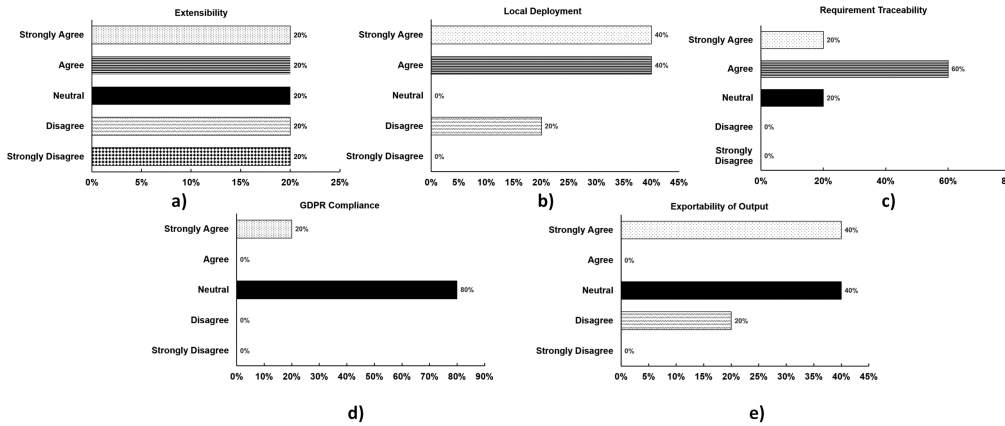


Figure 50: An overview of criteria evaluation among RM tools

IRIS meta-model to automate traceability and model relationships between elements in the requirement process. This automated traceability feature simplifies requirement management by allowing users to identify dependencies and assess the impact of changes on the application as a whole.

- **GDPR compliant:** Compliance with data protection law and privacy principles are important aspects of the CAIRIS requirement management tool. The developers of CAIRIS have incorporated model validation checks for GDPR compliance into CAIRIS. They defined three new types of roles to CAIRIS (Data Controller, Data Processor, and Data Subject) and steps for introducing personal data assets into a CAIRIS model.
- **Textual requirement:** In CAIRIS, text can be created, modified, and managed through the tool's interface. CAIRIS' text requirements are stored in the database, allowing for traceability between text requirements and other elements, which helps ensure that the application being developed meets identified needs and expectations.
- **Exportability:**
  CAIRIS has export capabilities that enable users to export project data in multiple formats like CSV, JSON, PDF, and XML. In addition, users can also export specific reports like risk assessments, threats, and vulnerabilities and share data with external stakeholders. The export capabilities also serve as a means for users to maintain a backup of their data in a format that can be easily imported into CAIRIS.

A.6. Guidelines and recommendations

In the light of our analysis, we next define some guidelines for the selection of tools and provide some general recommendations regarding the expected functionality that new versions of the tools must include for modern software.

Extendability: Most of the RM tools that comply with the guidelines and regulations about building AI software are also open source. While their source code is available online, extending the tool with certain functionality may require low-level programming instrumentation and re-compilation of the source code. This may be a difficult task that can potentially increase the elicitation process of requirements. Besides this, instrumenting code may also become a security breach in the development life cycle.

Supplementary features: Vulnerabilities are weaknesses that can be exploited to compromise the security of a system. AI-based systems are susceptible to different attacks. Hence, the prioritization of risk and vulnerability assessment features for RM tools dedicated to AI-based systems is required. The objective of vulnerability assessment during system development is to facilitate risk management. The process of addressing the vulnerabilities entails **identifying, assessing, and prioritizing potential security vulnerabilities** [142]. The process can be challenging for developers. However, deploying RM tools with risk and vulnerability assessment capabilities can strengthen and adequately manage the process as part of the requirement management process during development. For instance, RM tools such as CAIRIS may be equipped with in-built functionality that allows users to create a vulnerability form detailing the name, description, type of vulnerability, the likelihood of occurrence, impact level, and the exposed asset. Another key supplementary functionality that RM tools could improve is concurrent management [116, 407]. Indeed, RM tools for concurrent or real-time editing by multiple users working on the same can aid in tuning requirements in collaborative projects. High resilience to concurrent changes in requirements can ensure that any changes one user makes are automatically reflected and updated in real-time across all user interfaces, irrespective of geographical location.

Maintainability: Several open-source tools lack detailed documentation, increasing their learning curve. RM tools with minimal (or non-existent) documentation can delay the process of defining requirements, as the development team first needs to study in detail the functionality of the tool and assess whether it is suitable for the project or not. RM tools that handle the requirements of modern applications must clearly define the level of support to handle AI specifications.

Version control: AI-based applications and systems development processes differ significantly from classical application development due to a series of iterative activities and tuning involved in model training for achieving stable and optimal model performance. It is essential to capture and track changes in metadata, data, parameters, configuration, etc, that transpire throughout the development to monitor different versions of model performance that are observed. The inability to log, track and trace these changes using a requirement tool can result in costly consequences. Therefore, the availability of version control capability is crucial when selecting requirement management tools for any AI-based developmental projects.

Data management: Meta-data defines the attributes available in the dataset and it is expected that RM tools can handle such descriptor representation for the dataset. Considering that AI-based systems rely heavily on big amounts of data, it is crucial to consider also whether the changes on raw data, e.g., new data contributions, data format; can be also tracked by the RM tools. Dataset version has a direct link to model version and its performance.

Deployment: Besides having specific characteristics, it is also important to re-assess the functionalities of the RM tools once is deployed for its usage. Indeed, it is possible that vendors providing the underlying infrastructure to host the tool do not comply with certain regulations, invalidating completely the usage of the tool. For instance, a private cloud vendor could avoid specifying how the replication process is handle with their cloud technology. For instance, replication just in the same location (Europe) or to different locations (EU to US).

A.7. Discussion

Room for improvement: While our work provides several insights on selecting a RM tool for handling AI-based requirements, our work is limited in the context of a single project. Thus, we are interested on replicating our study to other projects, such that it is possible to generalize our findings. Besides this, there are several factors that may impact this type of studies. For instance, the number of teams in a project, and the amount of resources available to explore the suitability of a RM tool.

Pre-defined evaluation frameworks: As demonstrated in our work, selecting a RM tool for AI-based solution development can pose challenges due to the plethora of available options which consumes a significant amount of time. However, employing evaluation frameworks like ISO/IEC TR 24766:2009 and the International Council on Systems Engineering (INCOSE) list can potentially assist in navigating the selection process by offering a comprehensive list of features and facilitating the comparisons of these features across different tools. It is important to exercise caution when utilizing these frameworks, as they serve as a guide and the listed features should still be evaluated based on the specific requirements of the development project.

Stakeholder challenges: The complexities involved in fulfilling stakeholders' diverse expectations and preferences can make the selection of RM tools difficult. When developing an AI-based solution, there are different types of stakeholders to consider, each with unique needs and priorities. For instance, stakeholders' expectations could vary regarding the tool functionalities, user experiences, traceability, report generation, customization, resource sharing, etc. This diversity makes it difficult to align the capabilities of the various tools to consider with the stakeholders' expectations. However, effective engagement of stakeholders during the tool selection process can overcome the challenges.

Tool integration and compatibility: The requirement management process is not a standalone process in the AI-based product development processes and other activities. Integrating the RM tool into existing processes and systems is essential for effectively managing requirements. Analyzing the compatibility of the requirement management tool can ensure seamless integration of the tool into existing systems. Similarly, the requirement management process should integrate easily with existing procedures to facilitate efficient exchange of requirements data, artifacts, and information among systems and stakeholders. Hence, considering the fitness of tools in the light of their integration and compatibility with existing processes and systems can influence choice-making when selecting a requirement management tool.

Lessons learned: Several tools in the market are designed in line with the standard capabilities to handle **ISO/IEC TR 24766:2009** requirement definitions. Our assessment of RM tools for AI-based system development projects reveals that not all RM tools can effectively support such development due to some of the unique requirements of the development, for instance, data privacy and transparency requirements. Managing the requirements thus requires specific capabilities that some RM tools lack. Moreover, while standards and regulations exist to establish the standard capabilities of RM tools, the standard needs continuous review and update to adapt to the emerging reality of the integration of AI into development projects. In addition, projects involving different stakeholders, such as technical and socio-technical experts, require to consider selecting an RM tool that can handle the definition of requirements from specific domain angles and terminology.

A.8. Background and related work

Requirement management (RM) tools support many requirement engineering processes such as gathering, elicitation, analysis, and verification within the software development life-cycle [203]. These tools are typically used to track and monitor the evolution of requirements during the whole development process. At the same time, RM tools provide a way to document and quantify the building process of software applications [242]. Several studies have categorized different tools based on their functionalities and features [203, 120, 2]. As modern system architectures evolve, RM tools also provide over time new features [149] that are identified by different stakeholders involved in the development of applications [232], e.g., software developers and data scientists, among others.

On the other hand, regulatory entities also have introduced new aspects and properties that computing software has to consider when implementing AI-based functionality. For instance, as classical software architectures are augmented to consider new paradigms such as machine and federated learning [327], new features in RM tools are required that take into consideration the collection of data at scale. Besides this, systems and applications implementing AI are required to adopt trustworthy properties, requiring analysis of other dimensional aspects of

data used for training AI models, such as fairness and transparency [230]. Indeed, trustworthy AI depicts a set of trustworthy properties required for computing programs to be considered trustful [449] and several studies have identified new types of requirements to consider in these modern systems [230]. In this work, we revisit the list of requirements to be considered in AI-based systems, and unlike others, we analyze whether off-the-shelf RM tools can cope with these new considerations.

A.9. Summary and conclusions

In this paper, we presented a rigorous qualitative and quantitative analysis of different requirement management tools. The goal of the analysis was to evaluate whether existing requirement management tools can cope with the demands of tracking and monitoring requirements for AI-based applications, which are subject to specific characteristics imposed by regulatory entities. The analysis is conducted in the context of a consortium that consists of several academic and industry partners. By applying the COTS method, five different tools were assessed, each by one individual partner from the consortium. Our results suggest that new regulations make it difficult to find a requirement management tool for developing AI-based software. We also share our lessons learned and experiences from selecting requirement tools that can be used in team-based consortium projects.

ACKNOWLEDGEMENTS

All praise and special gratitude to **Allah**, the Almighty and Lord of creations, for His infinite mercy, guidance, and the fortitude **He** bestowed upon me towards the successful completion of my Ph.D. program.

"It is said that if you want to go far, go together". My sincere and profound appreciation extends to my supervisor, **Associate Professor Huber Flores**, for his relentless support throughout this journey. Without his mentorship, companionship, and diligent guidance, this journey would have been far more challenging. I am very grateful for the opportunity and the invaluable insights you provided to shape both my research work and aspirations.

To my esteemed colleagues and cherished friends, your thoughtful discussions, collaborative spirit, and constant encouragement have been instrumental to this achievement. The intellectual and camaraderie relationship we built together inspired me throughout this demanding journey. Thank you for enriching this experience in immeasurable ways.

Finally, my deepest gratitude belongs to my beloved family, whose unconditional support, endless patience, and persistent prayers sustained me through every triumph and challenge. Your belief in me never faltered, even when mine wavered. This achievement stands as a testament to your love and sacrifice; it belongs as much to you as it does to me. Words cannot adequately express the depth of my appreciation for your presence in my life and in this academic pursuit.

This research has been financed by the European Social Fund via the “ICT programme” measure and the Estonian Research Council grant (PRG 2725). This research is also part of the SPATIAL project, funded by the European Union’s Horizon 2020 research and innovation programme under grant agreement No.101021808.

All content in the thesis is original, and generative AI tools were used solely to help polish the text for improved readability.

SISUKOKKUVÕTE

Praktiline usaldusväärne tehisintellekt inimjärelevalvega

Kaasaegsed rakendused tuginevad üha enam masin- ja süvaõppele ehk tehisintellektile (AI, ingl k *artificial intelligence*), et suurendada jõudlust, parandada taju ja pakkuda usaldusväärsemat kasutajakogemust. Vaatamata täiustatud arutlusvõimele on AI-mudelid sageli läbipaistmatud, tekitades muret ohutuse osas ja vähendades usaldust. Regulatiivsed raamistikud rõhuvad usaldusväärsele tehisintellektile, mis on arendatud kooskõlas usaldusväärse andmetöötluse (ingl k *trustworthy computing*) printsiipidega: läbipaistvus ja inimjärelevalve. Paraku esineb praktikas olulisi takistusi, mis pärsivad inimjärelvalve ja AI-süsteemide integreerimist. Käesoleva doktoritöö peamiseks uurimisküsimuseks on: **Kuidas integreerida inimjärelvalve AI-põhistesse rakendustesse, et jälgida ja parandada nende usaldusväärsust?**

Püstitatud ülesande lahendamiseks esitame kolm tulemust, millest igaüks keskendub konkreetsele tehnilisele küsimusele vastavas masinõppe etapis. Esiteks, kuna andmete kvaliteet on tehisintellekti otsuste tegemisel kriitilise tähtsusega, toome sisse sotsiaalteadliku liitõppe (SAFL, ingl k *Socially Aware Federated Learning*) hajutatud masinõppe eesmärgil. SAFL juhib mudeli treenimiseks vajalikke andmete valikut koostööpõhiselt, kasutades ära sotsiaalseid dünaamikaid ja ülesannete delegeerimist viisil, mis soodustab inimeste osalust. Põhjaliku kasutajauuringu ja kontseptsiooni tõestava rakenduse tulemused näitavad, et SAFL-i abil saadud inimeste sisend parandab nii andmete kvaliteeti kui ka masinõppemudeli jõudlust.

Teiseks, kuna rakendused kaasavad üha enam tehisintellekti komponente, pakume lahenduse nende usaldusväärsuse omaduste jälgimiseks. Jälgides süsteemiarhitektuuride arengut uurime süstemaatiliselt, kuidas saab integreerida usaldusväärsuse mehhanisme kaasaegsetesse süsteemidesse. Kontseptsiooni tõestuseks loome SPATIAL-i – arhitektuuri, mis integreerib usaldusväärsuse mõõdikud AI-põhistesse rakendustesse. SPATIAL kuvab neid mõõdikuid lihtsasti arusaadava kasutajaliidesena, võimaldades vastavatel ekspertidel jälgida AI järeldusloogikat. Empiirilised hinnangud demonstreerivad SPATIAL-i efektiivsust, tuues samas esile usaldusväärsuse omaduste hindamise ja jälgimise keerukuse.

Kolmandaks rõhutame inimjärelvalve vajalikkust ka olemasolevate rakenduste jälgimiseks, eriti kui need rakendused toimivad autonoomselt ja laiaulatuslikult. Selleks pakume välja AntiVenom-i – tõhusa ja valdkonna-agnostilise meetodi anomaaliate tuvastamiseks hajutatud tehisintellekti rakendustes. AntiVenom kasutab seadme tasemel jõudlusmõõdikuid, et tuvastada ebakorrapärasusi ja märgistada need inimese poolt läbivaatamiseks. Võrdlus olemasolevate selgitatavate tehisintellekti (XAI, ingl k *explainable AI*) meetoditega näitab AntiVenomi potentsiaali kiire ja ennetava jälgimise jaoks võrreldes traditsiooniliste ja keerukamate meetoditega.

Kokkuvõttes panustavad toodud tulemused usaldusväärse AI arengusse, tuues esile nii inimosaluse potentsiaali kui ka keerukust selle rakendamisel järjest autonoomsemaks muutuvates süsteemides.

CURRICULUM VITAE

Personal data

Name:	Abdul-Rasheed Olatunji Ottun
Date of birth:	25.08.1988
Citizenship:	Nigeria
Contact:	ottunrasheed@gmail.com

Education

2022–2025	**University of Tartu, Estonia** Doctoral degree in computer science
2018–2020	**University of Tartu, Estonia** Masters' degree in innovation and technology management
2008–2013	**Obafemi Awolowo University, Nigeria** Bachelors' degree in economics

Employment

2022–2026	**University of Tartu, Estonia** Junior research fellow (EU SPATIAL Horizon)
2021–2022	**University of Tartu, Estonia** Scientific programmer

Teaching experience

- Teaching assistant for distributed systems course (LTAT.06.007)
- Co-supervision of master's students' course projects and theses

Scientific research area

Main scientific research area includes:

- Trustworthy and responsible artificial intelligence
- Federated learning within pervasive systems
- Human - AI interaction with focus on human oversight mechanisms

Volunteering experience and award

- Student volunteer, ACM International Conference on Mobile Systems, Applications, and Services (MobiSys 2023)
- Student grant award, IEEE International Conference on Distributed Computing Systems (ICDCS 2024)

ELULOOKIRJELDUS

Isikuandmed

Nimi:	Abdul-Rasheed Olatunji Ottun
Sünniaeg:	25.08.1988
Kodakondsus:	Nigeeria
Contact:	ottunrasheed@gmail.com

Haridus

2022–2025	**Tartu Ülikool, Eesti** Doktoriõpe, eriala: arvutiteadus
2018–2020	**Tartu Ülikool, Eesti** Magistriõpe, eriala: innovatsiooni- ja tehnoloogiajuhtimine
2008–2013	**Obafemi Awolowo Ülikool, Nigeeria** Bakalaureusekraad, eriala: majandusteadus

Teenistuskäik

2022–2026	**Tartu Ülikool, Eesti** Nooremteadur (EU SPATIAL Horizon)
2021–2022	**Tartu Ülikool, Eesti** Teaduslik programmeerija

Õpetamiskogemus

- Õppeassistent aines Hajussüsteemid (LTAT.06.007)
- Magistrantide kursusprojektide ja lõputööde kaasjuhendamine

Teadustegevus

- Usaldusväärne ja vastutustundlik tehisintellekt
- Liitõpe laussüsteemides (ingl *pervasive systems*)
- Inimese-tehisintellekti interaktsioon keskendudes inimjärelevalvele

Vabatahtlik töö ja uuringupreemiad

- Vabatahtlik, ACM International Conference on Mobile Systems, Applications, and Services (MobiSys 2023)
- Üliõpilaste reisitoetuse auhind, IEEE International Conference on Distributed Computing Systems (ICDCS 2024)

DISSERTATIONES INFORMATICAE PREVIOUSLY PUBLISHED IN DISSERTATIONES MATHEMATICAE UNIVERSITATIS TARTUENSIS

19. **Helger Lipmaa.** Secure and efficient time-stamping systems. Tartu, 1999, 56 p.
22. **Kaili Müürisep.** Eesti keele arvutigrammatika: süntaks. Tartu, 2000, 107 lk.
23. **Varmo Vene.** Categorical programming with inductive and coinductive types. Tartu, 2000, 116 p.
24. **Olga Sokratova.** Ω-rings, their flat and projective acts with some applications. Tartu, 2000, 120 p.
27. **Tiina Puolakainen.** Eesti keele arvutigrammatika: morfoloogiline ühestamine. Tartu, 2001, 138 lk.
29. **Jan Villemson.** Size-efficient interval time stamps. Tartu, 2002, 82 p.
45. **Kristo Heero.** Path planning and learning strategies for mobile robots in dynamic partially unknown environments. Tartu 2006, 123 p.
49. **Härmel Nestra.** Iteratively defined transfinite trace semantics and program slicing with respect to them. Tartu 2006, 116 p.
53. **Marina Issakova.** Solving of linear equations, linear inequalities and systems of linear equations in interactive learning environment. Tartu 2007, 170 p.
55. **Kaarel Kaljurand.** Attempto controlled English as a Semantic Web language. Tartu 2007, 162 p.
56. **Mart Anton.** Mechanical modeling of IPMC actuators at large deformations. Tartu 2008, 123 p.
59. **Reimo Palm.** Numerical Comparison of Regularization Algorithms for Solving Ill-Posed Problems. Tartu 2010, 105 p.
61. **Jüri Reimand.** Functional analysis of gene lists, networks and regulatory systems. Tartu 2010, 153 p.
62. **Ahti Peder.** Superpositional Graphs and Finding the Description of Structure by Counting Method. Tartu 2010, 87 p.
64. **Vesal Vojdani.** Static Data Race Analysis of Heap-Manipulating C Programs. Tartu 2010, 137 p.
66. **Mark Fišel.** Optimizing Statistical Machine Translation via Input Modification. Tartu 2011, 104 p.
67. **Margus Niitsoo**. Black-box Oracle Separation Techniques with Applications in Time-stamping. Tartu 2011, 174 p.
71. **Siim Karus.** Maintainability of XML Transformations. Tartu 2011, 142 p.
72. **Margus Treumuth.** A Framework for Asynchronous Dialogue Systems: Concepts, Issues and Design Aspects. Tartu 2011, 95 p.
73. **Dmitri Lepp.** Solving simplification problems in the domain of exponents, monomials and polynomials in interactive learning environment T-algebra. Tartu 2011, 202 p.

74. **Meelis Kull.** Statistical enrichment analysis in algorithms for studying gene regulation. Tartu 2011, 151 p.
77. **Bingsheng Zhang.** Efficient cryptographic protocols for secure and private remote databases. Tartu 2011, 206 p.
78. **Reina Uba.** Merging business process models. Tartu 2011, 166 p.
79. **Uuno Puus.** Structural performance as a success factor in software development projects – Estonian experience. Tartu 2012, 106 p.
81. **Georg Singer.** Web search engines and complex information needs. Tartu 2012, 218 p.
83. **Dan Bogdanov.** Sharemind: programmable secure computations with practical applications. Tartu 2013, 191 p.
84. **Jevgeni Kabanov.** Towards a more productive Java EE ecosystem. Tartu 2013, 151 p.
87. **Margus Freudenthal.** Simpl: A toolkit for Domain-Specific Language development in enterprise information systems. Tartu, 2013, 151 p.
90. **Raivo Kolde.** Methods for re-using public gene expression data. Tartu, 2014, 121 p.
91. **Vladimir Šor.** Statistical Approach for Memory Leak Detection in Java Applications. Tartu, 2014, 155 p.
92. **Naved Ahmed.** Deriving Security Requirements from Business Process Models. Tartu, 2014, 171 p.
94. **Liina Kamm.** Privacy-preserving statistical analysis using secure multi-party computation. Tartu, 2015, 201 p.
100. **Abel Armas Cervantes.** Diagnosing Behavioral Differences between Business Process Models. Tartu, 2015, 193 p.
101. **Fredrik Milani.** On Sub-Processes, Process Variation and their Interplay: An Integrated Divide-and-Conquer Method for Modeling Business Processes with Variation. Tartu, 2015, 164 p.
102. **Huber Raul Flores Macario.** Service-Oriented and Evidence-aware Mobile Cloud Computing. Tartu, 2015, 163 p.
103. **Tauno Metsalu.** Statistical analysis of multivariate data in bioinformatics. Tartu, 2016, 197 p.
104. **Riivo Talviste.** Applying Secure Multi-party Computation in Practice. Tartu, 2016, 144 p.
108. **Siim Orasmaa.** Explorations of the Problem of Broad-coverage and General Domain Event Analysis: The Estonian Experience. Tartu, 2016, 186 p.
109. **Prastudy Mungkas Fauzi.** Efficient Non-interactive Zero-knowledge Protocols in the CRS Model. Tartu, 2017, 193 p.
110. **Pelle Jakovits.** Adapting Scientific Computing Algorithms to Distributed Computing Frameworks. Tartu, 2017, 168 p.
111. **Anna Leontjeva.** Using Generative Models to Combine Static and Sequential Features for Classification. Tartu, 2017, 167 p.
112. **Mozhgan Pourmoradnasseri.** Some Problems Related to Extensions of Polytopes. Tartu, 2017, 168 p.

113. **Jaak Randmets.** Programming Languages for Secure Multi-party Computation Application Development. Tartu, 2017, 172 p.
114. **Alisa Pankova.** Efficient Multiparty Computation Secure against Covert and Active Adversaries. Tartu, 2017, 316 p.
116. **Toomas Saarsen.** On the Structure and Use of Process Models and Their Interplay. Tartu, 2017, 123 p.
121. **Kristjan Korjus.** Analyzing EEG Data and Improving Data Partitioning for Machine Learning Algorithms. Tartu, 2017, 106 p.
122. **Eno Tõnisson.** Differences between Expected Answers and the Answers Offered by Computer Algebra Systems to School Mathematics Equations. Tartu, 2017, 195 p.

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

1. **Abdullah Makkeh**. Applications of Optimization in Some Complex Systems. Tartu 2018, 179 p.
2. **Riivo Kikas**. Analysis of Issue and Dependency Management in Open-Source Software Projects. Tartu 2018, 115 p.
3. **Ehsan Ebrahimi**. Post-Quantum Security in the Presence of Superposition Queries. Tartu 2018, 200 p.
4. **Ilya Verenich**. Explainable Predictive Monitoring of Temporal Measures of Business Processes. Tartu 2019, 151 p.
5. **Yauhen Yakimenka**. Failure Structures of Message-Passing Algorithms in Erasure Decoding and Compressed Sensing. Tartu 2019, 134 p.
6. **Irene Teinemaa**. Predictive and Prescriptive Monitoring of Business Process Outcomes. Tartu 2019, 196 p.
7. **Mohan Liyanage.** A Framework for Mobile Web of Things. Tartu 2019, 131 p.
8. **Toomas Krips.** Improving performance of secure real-number operations. Tartu 2019, 146 p.
9. **Vijayachitra Modhukur.** Profiling of DNA methylation patterns as biomarkers of human disease. Tartu 2019, 134 p.
10. **Elena Sügis.** Integration Methods for Heterogeneous Biological Data. Tartu 2019, 250 p.
11. **Tõnis Tasa.** Bioinformatics Approaches in Personalised Pharmacotherapy. Tartu 2019, 150 p.
12. **Sulev Reisberg.** Developing Computational Solutions for Personalized Medicine. Tartu 2019, 126 p.
13. **Huishi Yin.** Using a Kano-like Model to Facilitate Open Innovation in Requirements Engineering. Tartu 2019, 129 p.
14. **Faiz Ali Shah.** Extracting Information from App Reviews to Facilitate Software Development Activities. Tartu 2020, 149 p.
15. **Adriano Augusto**. Accurate and Efficient Discovery of Process Models from Event Logs. Tartu 2020, 194 p.
16. **Karim Baghery.** Reducing Trust and Improving Security in zk-SNARKs and Commitments. Tartu 2020, 245 p.
17. **Behzad Abdolmaleki.** On Succinct Non-Interactive Zero-Knowledge Protocols Under Weaker Trust Assumptions. Tartu 2020, 209 p.
18. **Janno Siim.** Non-Interactive Shuffle Arguments. Tartu 2020, 154 p.
19. **Ilya Kuzovkin.** Understanding Information Processing in Human Brain by Interpreting Machine Learning Models. Tartu 2020, 149 p.
20. **Orlenys López Pintado.** Collaborative Business Process Execution on the Blockchain: The Caterpillar System. Tartu 2020, 170 p.
21. **Ardi Tampuu.** Neural Networks for Analyzing Biological Data. Tartu 2020, 152 p.

22. **Madis Vasser.** Testing a Computational Theory of Brain Functioning with Virtual Reality. Tartu 2020, 106 p.
23. **Ljubov Jaanuska.** Haar Wavelet Method for Vibration Analysis of Beams and Parameter Quantification. Tartu 2021, 192 p.
24. **Arnis Parsovs.** Estonian Electronic Identity Card and its Security Challenges. Tartu 2021, 214 p.
25. **Kaido Lepik.** Inferring causality between transcriptome and complex traits. Tartu 2021, 224 p.
26. **Tauno Palts.** A Model for Assessing Computational Thinking Skills. Tartu 2021, 134 p.
27. **Liis Kolberg.** Developing and applying bioinformatics tools for gene expression data interpretation. Tartu 2021, 195 p.
28. **Dmytro Fishman.** Developing a data analysis pipeline for automated protein profiling in immunology. Tartu 2021, 155 p.
29. **Ivo Kubjas.** Algebraic Approaches to Problems Arising in Decentralized Systems. Tartu 2021, 120 p.
30. **Hina Anwar.** Towards Greener Software Engineering Using Software Analytics. Tartu 2021, 186 p.
31. **Veronika Plotnikova.** FIN-DM: A Data Mining Process for the Financial Services. Tartu 2021, 197 p.
32. **Manuel Camargo.** Automated Discovery of Business Process Simulation Models From Event Logs: A Hybrid Process Mining and Deep Learning Approach. Tartu 2021, 130 p.
33. **Volodymyr Leno.** Robotic Process Mining: Accelerating the Adoption of Robotic Process Automation. Tartu 2021, 119 p.
34. **Kristjan Krips.** Privacy and Coercion-Resistance in Voting. Tartu 2022, 173 p.
35. **Elizaveta Yankovskaya.** Quality Estimation through Attention. Tartu 2022, 115 p.
36. **Mubashar Iqbal.** Reference Framework for Managing Security Risks Using Blockchain. Tartu 2022, 203 p.
37. **Jakob Mass.** Process Management for Internet of Mobile Things. Tartu 2022, 151 p.
38. **Gamal Elkoumy.** Privacy-Enhancing Technologies for Business Process Mining. Tartu 2022, 135 p.
39. **Lidia Feklistova.** Learners of an Introductory Programming MOOC: Background Variables, Engagement Patterns and Performance. Tartu 2022, 151 p.
40. **Mohamed Ragab.** Bench-Ranking: A Prescriptive Analysis Approach for Large Knowledge Graphs Query Workloads. Tartu 2022, 158 p.
41. **Mohammad Anagreh.** Privacy-Preserving Parallel Computations for Graph Problems. Tartu 2023, 181 p.
42. **Rahul Goel.** Mining Social Well-being Using Mobile Data. Tartu 2023, 104 p.

43. **Anti Ingel.** Algorithms using information theory: classification in brain-computer interfaces and characterising reinforcement-learning agents. Tartu 2023, 142 p.
44. **Shakshi Sharma.** Fighting Misinformation in the Digital Age: A Comprehensive Strategy for Characterizing, Identifying, and Mitigating Misinformation on Online Social Media Platforms. Tartu 2023, 158 p.
45. **Kristiina Rahkema.** Quality Analysis of iOS Applications with Focus on Maintainability and Security Aspects. Tartu 2023, 182 p.
46. **Ivan Slobozhan.** Studying Online Social Media Engagement in CIS Countries during Protests, Mass Demonstrations and War. Tartu 2023, 81 p.
47. **Nurlan Kerimov.** Building a catalogue of molecular quantitative trait loci to interpret complex trait associations. Tartu 2023, 248 p.
48. **Pavlo Tertychnyi.** Machine Learning Methods for Anti-Money Laundering Monitoring. Tartu 2023, 117 p.
49. **Abasi-amefon Obot Affia.** A Framework and Teaching Approach for IoT Security Risk Management. Tartu 2023, 180 p.
50. **Raimond-Hendrik Tunnel.** Video Game Design and Development Bachelor's Curriculum for Estonia. Tartu 2024, 137 p.
51. **Ahto Salumets.** Bioinformatics analysis of various aspects in immunology. Tartu 2024, 198 p.
52. **Mohammed Abdulhameed Shaif Ali.** Deep Learning Methods for Cell Microscopy Image Analysis. Tartu 2024, 143 p.
53. **Pille Pullonen-Raudvere**. Foundations of Efficient and Secure Algorithm Development for Secure Multiparty Computation. Tartu 2024, 265 p.
54. **Marili Rõõm.** Multiple approaches to learners' success and factors affecting it in computer programming MOOCs. Tartu 2024, 170 p.
55. **Shivananda Rangappa Poojara.** Design and Orchestration of Scalable, Event-Driven Serverless Data Pipelines for Internet of Things (IoT) Applications. Tartu 2024, 172 p.
56. **Hassan Abdulgaleel Hassan Salim Eldeeb.** Empowering Machine Learning Pipelines with Automated Feature Engineering. Tartu 2024, 121 p.
57. **Muhammad Uzair.** Soft decision making for agri-food 4.0. Tartu 2024, 158 p.
58. **Kirill Milintsevich.** Estimation of Depression Level from Text: Symptom-Based Approach, External Knowledge, Dataset Validity. Tartu 2024, 130 p.
59. **Maksym Del.** Multilingual and Multi-Domain Representational Patterns Across Trpansformer-Based Models. Tartu 2024, 131 p.
60. **Kristo Raun.** Adaptive Out-of-order Handling in Streaming Conformance Checking. Tartu 2024, 118 p.
61. **Toivo Vajakas.** Towards integration of mobile network data into analyzing human mobility. Tartu 2024, 103 p.
62. **Katsiaryna Lashkevich.** Data-Driven Analysis and Optimization of Waiting Times in Business Processes. Tartu 2024, 169 p.
63. **Alejandra Duque-Torres.** Classifying, Constraining and Ranking Metamorphic Relations. Tartu 2025, 159 p.

64. **Mariia Bakhtina.** A Method for Information Security and Privacy Management in Smart Solutions. Tartu 2025, 199 p.
65. **Andre Tättar.** Multilingual Machine Translation for Under-Resourced Languages. Tartu 2025, 170 p.
66. **Mahmoud Shoush.** Prescriptive Process Monitoring Under Uncertainty and Resource Constraints. Tartu 2025, 178 p.
67. **Alireza Akhavi Zadegan.** A Multimodal approach for refining Mapping and Localization by Integrating Generative AI and Pedestrian-Centric Data. Tartu 2025, 147 p.
68. **Eerik Muuli.** Automating the assessment and feedback processes in IT teaching – improving creation and maintenance from the teaching staff perspective. Tartu 2025, 196 p.
69. **Kateryna Kubrak.** Towards User-Centered Prescriptive Process Monitoring Systems. Tartu 2025, 151 p.
70. **Zhigang Yin.** Computing and Sensing in a Smart Ring. Tartu 2025, 251 p.