Proceedings of the 35th European Safety and Reliability & the 33rd Society for Risk Analysis Europe Conference
Edited by Eirik Bjorheim Abrahamsen, Terje Aven, Frederic Bouder, Roger Flage, Marja Ylönen
 Published by Research Publishing, Singapore.
doi: 10.3850/978-981-94-3281-3_ESREL-SRA-E2025-P5318-cd

Assuring Safety of AI-based Systems: Lessons Learned for a Driverless Regional Train Case Study

Marc Zeller
Siemens AG, Germany. E-mail: marc.zeller@siemens.com

Ronald Schnitzer
Siemens AG, Germany. E-mail: ronald.schnitzer@siemens.com

Artificial Intelligence (AI) offers great potential to enable the fully automated operation of trains. Mandatory novel functions to replace the tasks of a human train driver, such as obstacle detection on the tracks, can be realized using state-of-the-art Machine Learning (ML) approaches. However, the use of AI/ML to implement perception tasks in the railway context poses a new challenge: How to link AI/ML techniques with the requirements and approval processes that are applied in the railway domain in practical way? Within the safe.trAIn project we laid the foundation for the safe use of AI/ML to achieve the driverless operation of a regional train. Based on the requirements for the certification process in the railway domain, safe.trAIn investigated methods to develop trustworthiness AI-based functions, taking data quality, robustness, uncertainty, and explainability aspects of the ML model into account. In addition, the project developed a safety argumentation strategy for an AI-based obstacle detection function of a driverless regional train. In this paper, we describe the challenges to assess an AI-based obstacle detection function according to the given regulation in the railway domain. Moreover, we describe our safety assurance strategy applied to our case study in the safe.trAIn project.

Keywords: Driverless regional train, Safety assurance, AI/ML safety, Safety approval, Railway, Autonomous driving.

1. Introduction

With the introduction of driverless train operation (*Grade of Automation (GoA)* 4 operation corresponding to SAE level 5 in automotive) a significant performance increase of railway systems can be achieved. This includes the enhancement of the transport capacity in existing tracks, energy savings by means of an optimized driving strategy, reduced mechanical wear and tear as well as increased passenger comfort by means of homogeneous driving, and increased flexibility for demand-oriented train services. To enable fully automated operation of trains, novel functions must be implemented to replace the tasks of a human train driver, such as detecting obstacles on the tracks. Traditional automation technologies alone are not sufficient to perform an obstacle detection function for a driverless train. However, *Artificial Intelligence (AI)* and *Machine Learning (ML)* offers great potential to solve this task. The problem, which still remains unresolved, is to find a practical way to link AI/ML techniques with the requirements and approval processes that are applied in the railway domain.

We foresee that different tasks need to be addressed so that an Automated Driving System (ADS) in rail can be approved for operation, including a) Linking requirements originating from functional safety on system-level (e.g., originating from EN 50126-1:2018-10 (2018)) to the ML-based obstacle detection function and formalizing a sound safety argumentation, b) Providing insight into the ML behavior and how it relates to data and further to the safety requirements to provide evidences for the safety case, and c) addressing the challenge that ADS also in rail will operate in an open world, which is difficult to specify a priori and prone to changes during its lifecycle, hence it requires agile MLOps cycles including testing & validation.

As described in Zeller et al. (2023)), the safe.trAIn project (https://safetrain-projekt.de/en/) aims to lay the foundation for the safe use of AI/ML for the driverless operation

of rail vehicles and thus addresses these key technological challenges that hinder the adoption of unmanned rail transport. Therefore, safe.trAIn creates a safety argumentation for an AI-based obstacle detection function of a driverless regional train based on the requirements for the certification process in the railway domain. Furthermore, the project investigates methods to assess AI-based functions, taking into account the robustness, performance, and transparency aspects of the AI/ML models. These methods are integrated into a comprehensive and agile, development, testing, and *Verification & Validation (V&V)* cycle for AI-based functions in trains. The feasibility of the assurance methods developed in safe.trAIn was evaluated with a case study – limited in scope to certain functions and AI specific aspects, in which an example safety case for a regional driverless train was created and assessed.

In this paper, we present the main results of the safe.trAIn project to cope with the challenges mentioned above. Thereby, we focus on (a) the so-called *Landscape of AI Safety Concerns (LAISC)*, which guides the creation of a sound safety argumentation and (b) a process for the continuous development and safety assurance of ML-based systems in the railway domain in which evidences for the safety case are created.

The rest of the paper is organized as follows: In Section 2, we summarize relevant related work. Section 3, introduces the concept of Landscape of AI Safety Concerns, a methodology to guide the creation of a safety argument for AI-based systems. In Section 4, we introduce the safe MLOps process for ML-based systems in the railway domain as defined in the safe.trAIn project. Afterwards, we introduce the safety argumentation in safe.trAIn which is based on the combination of the Landscape of AI Safety Concerns and the safe MLOps process. At the end, we summarize the main results of the paper and provide an outlook on future research work.

2. Related Work

The use of AI technologies introduces new sources of systematic failures and, thus, unique challenges in system assurance. Existing safety standards such as IEC 61508-1:2010-04 (2010) or EN 50126-1:2018-10 (2018) in the railway domain do not address the development and assurance of ML models yet. Only the EN 50716:2023-11 (2023) describes the challenges of AI-based functions on an abstract level in Annex C.

There are first standards related to safety and AI in other domains. The ISO/PAS 8800:2024-12 (2024) specification, an automotive safety standard for AI, describes the development of an assurance argument. To develop the assurance argument, an AI safety lifecycle consisting of 4 phases is defined: Selection of AI approach and AI system design, Data specification and collection for training and test, AI safety analysis, and AI system verification and validation. In these phases, evidences for the assurance argument are created. However, the standard does not specify AI safety requirements, which must be fulfilled.

The assurance of AI-based systems is still an active field of research, with various aspects being explored, e.g., in the ”ExamAI” project presented in Adler and Klaes (2022), the ”Assuring Autonomy International Programme” introduced in Hawkins et al. (2021), the project ”KI Absicherung” (safe AI for automated driving) described in Burton et al. (2022) or the International Workshop for Autonomous System Safety (IWASS)” series as depicted in the whitepaper by Correa-Jullian et al. (2023). A thorough survey on assuring ML-based systems is provided by Ashmore et al. (2021). In their work, the authors segment the ML life cycle into four phases, suggesting desiderata for each and discussing available assurance methods and associated challenges. Also Schwalbe and Schels (2020) presents a survey on specific considerations for safety argumentation targeting DNNs, organized into four development phases. The *Assurance of Machine Learning in Autonomous Systems (AMLAS)* process, described Hawkins et al. (2021), also employs the AI life cycle as a framework, defining assurance patterns to derive from top-level safety goals the evidence that needs to be generated in the AI life cycle. Thereby, the AMLAS approach remains generic, as it does not presume specific AI capabilities and shortcomings. An application

of the AMLAS process for a pedestrian detection system in the automotive domain is presented in Borg et al. (2023). Even this case study with a very restricted *Operational Design Domain (ODD)* shows that the resulting safety argumentation is huge and it is not clear if it is complete and detailed enough.

Roßbach et al. (2024) presents an approach for evaluating AI components in autonomous railway applications in terms of safety. This approach focuses on the following 4 pillars: Ontology & ODD specification, test case generation, evaluation of the AI component, and monitoring at runtime. Again, the four activities provide evidences for the safety case of the autonomous railway application. However, when using this approach, it is not clear if the resulting safety case is complete.

Other approaches in this area, such as Houben et al. (2022), Schnitzer et al. (2024a), and Willers et al. (2020) focus on the specific AI properties. However, the focus of these works is on the identification of AI-related safety concerns and required actions to manage them, but not on the derivation of a convincing assurance case. For instance, Schwalbe et al. (2020) systematically establishes and refines safety requirements to argue the sufficient absence of risk arising from SOTIF functional insufficiencies for autonomous vehicle. Parts of this work can also be reused for the safety assurance of a driverless train, but certification requirements in the railway are different from the automotive domain.

3. Landscape of AI Safety Concerns

In this section, we briefly outline the concept of the so-called Landscape of AI Safety Concerns (LAISC), a methodology to systematically support safety assurance of AI-based autonomous systems. For a more detailed description of LAISC, we refer to Schnitzer et al. (2024b). The methodology focuses on AI/ML-specific properties that cause traditional methods for assuring safety to lose effectiveness when applied to AI-based systems. In alignment with Willers et al. (2020) and Schnitzer et al. (2024b), we refer to these properties as *AI Safety Concerns (AI-SCs)*, defined as ”AI-specific, underlying issues that may negatively impact the safety of a system.”

The core concept of LAISC relies on addressing the gap in safety assurance for AI systems by demonstrating that all AI-SCs are sufficiently mitigated, building upon the contributions of Houben et al. (2022), Schnitzer et al. (2024a), Schwalbe et al. (2020), and Willers et al. (2020), which provided a comprehensive overview of known AI-SCs identified in the literature. From this, a list of AI-SCs, relevant to the use case of a driverless train, was derived in the safe.trAIn project, which is depicted in Fig. 1. Note, that building upon a list such as given in Schnitzer et al. (2024a) is a good starting point, since it represents the state-of-the-art and is comparable with established hazard identification methods. However, such lists do not claim to be complete. Therefore, we recommend expanding the list if use case or domain-specific conditions reveal additional AI safety concerns.

Having a list of relevant AI-SCs, a central part of the LAISC approach is generating evidence using state-of-the-art metrics and mitigation methods to demonstrate the absence of all AI-SCs. To employ a systematic approach, it is crucial to orchestrate the metrics and mitigation methods throughout the AI lifecycle, as AI-SCs manifest and require actions at different stages. For instance, data quality related AI-SCs should be addressed during the data collection and preprocessing steps, whereas other issues, such as concept drift, may arise only after deploying the system.

Another significant challenge is making verifiable claims about the absence of AI-SCs, as they are typically described at a relatively abstract level, while the evidence provided by the metrics and mitigation methods is very specific. For example, the AI-SC (17) ”Lack of robustness” is a typical issue for the majority of AI applications, but specific requirements for robustness vary significantly depending an use case’s conditions, such as the operational environment. For example, weather conditions such as fog or heavy rain significantly impact the operational safety of a regional driverless train, while these conditions are generally not a concern for autonomous vehicles operating within factory buildings.

To address the issue of varying levels of ab-

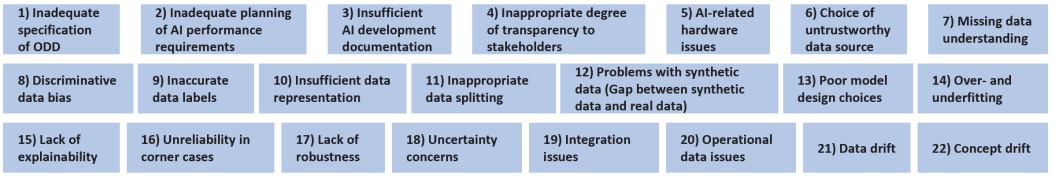

Fig. 1. List of AI-SCs (adopted from Schnitzer et al. (2024a) and slightly modified), laying the foundation for the Landscape of AI Safety Concerns in the safe.trAIn project.

straction, the LAISC methodology employs a two-layered approach: first, AI-SCs must be decomposed and tailored to use-case-specific conditions to create more concrete sub-goals; second, *Verifiable Requirements (VRs)* need to be specified to enable a clear evaluation of whether the evidence generated by the metrics and mitigation methods is adequate to fulfill the respective sub-goals. While this process establishes a sound and logical argumentation pattern, defining Verifiable Requirements remains challenging. This step typically involves defining thresholds for metrics and mitigation methods, yet the impact of certain AI-SCs (or even sub-goals) on safety at the system level may not be clearly determinable. Therefore, this step requires the application of expert judgment. Ultimately, we recommend that this step be performed by a team consisting of domain experts, safety engineers, and AI specialists to ensure a multi-expertise perspective. Notably, not all AI-SCs are equally important for demonstrating the safety of AI-based systems. Nevertheless, by defining Verifiable Requirements, it is possible to effectively manage and prioritize which AI-SCs are particularly critical for a specific use case.

4. Safe MLOps Process

In this section, we briefly outline the safe MLOps process specified in the safe.trAIn project to develop and assess safe AI-based functions for driverless trains. For more details, see Zeller et al. (2024b). As depicted in Fig. 2, the process integrates 3 parts: (a) the system engineering lifecycle (based on the development process of EN 50126-1:2018-10 (2018), (b) the data & ML lifecycle (based on ISO/IEC 23053:2021-06 (2022) and the AMLAS process), and (c) the safety assurance lifecycle (also based on the EN 50126-1).

In the *Dev* phase, the development is extended with a process to develop and verify ML models that implement functions of the driverless train, such as obstacle detection (Data & ML Lifecycle). Since we need to assess ADS in terms of safety, additional process steps are incorporated into the data & ML lifecycle to assess data quality and ensure the performance of the ML model in terms of safety, reliability, transparency, and robustness. Thereby, pieces of evidence (quantitative & qualitative) for the ML model are created and incorporated into the system safety case.

Moreover, the Operational Design Domain (ODD) is specified in the *Dev* phase. The ODD is a representative model of the real world in which an ADS is intended to operate. The definition of ODD is a crucial part of the development process for an AI-enabled system. This is due to the fact that the ODD is the basis for several critical development activities, such as defining system-level requirements, test & verification, and building a well-founded safety case, see Weiss et al. (2024).

As the pieces of evidence generated during the execution of the data & ML lifecycle are generated continuously, the development of the system safety case needs to be iterative. To create the safety case iteratively as part of the safe MLOps process, model-based techniques such as the *Goal Structuring Notation (GSN)* presented in Kelly (2004) or *Claim-Argument-Evidence (CAE)* described in Bloomfield and Bishop (2010) are used to specify the safety argumentation and the evidence created during the development and assurance process (see Sec. 5).

After a successful independent assessment of the safety case, the ML-based system (train) is put into operation (the *Ops* phase), in which safety-relevant parameters must be monitored.

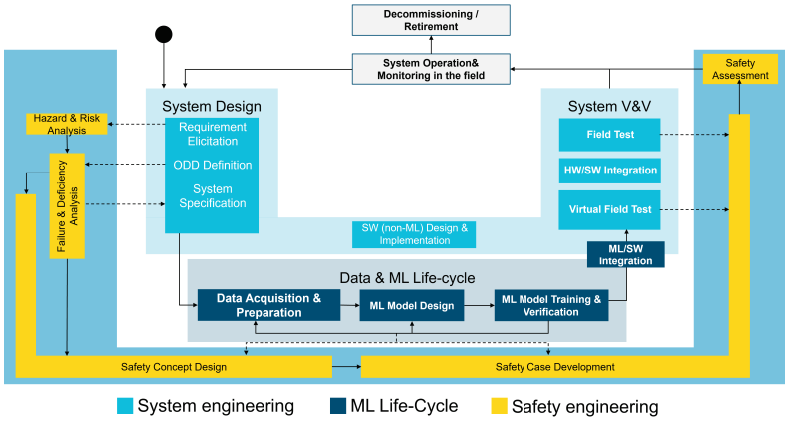

Fig. 2. Safe MLOps Process for ML-based ADS Systems

The MLOps lifecycle is crucial as it determines when AI-SCs manifest and also when they can be mitigated. For systematic mitigation of AI-SCs and generation of evidence for the safety case, AI-SCs and metrics/mitigation methods are assigned to the phases of the safe MLOps process. To efficiently perform these mitigation methods, the MLOps process is implemented in safe.trAIn using a Git-based workflow and the appropriate tool support to achieve the required degree of automation. For more details see Zeller et al. (2024a).

5. Safety Argumentation in safe.trAIn

In order to demonstrate the safety of an ML-based obstacle detection function in the context of ADS in the railway domain, we need to show that all AI Safety Concerns are sufficiently mitigated. Therefore, all Verifiable Requirements derived from the LAISC for the specific use case are mapped to one or multiple phases of the safe MLOps process, in which a specific metric/mitigation method is applied. Moreover, if possible, a threshold is defined which allows experts to judge whether the requirements are met or not.

For many of the AI-SCs in the LAISC, existing methods can be used to demonstrate that the concerns are fulfilled for the use case of an ML-based obstacle detection system in the railway domain. There are many metrics and methods available to demonstrate during the ”ML Model Training & Verification” phase that the AI-SCs (14) ”Over- and underfitting”, (15) ”Lack of explainability”, and (17) ”Lack of robustness” are sufficiently mitigated by generating training curves, conducting performance analyses as described in Schlosser et al. (2024), performing robustness tests as presented in Tocchetti et al. (2025), and doing explainability evaluations as shown in Linardatos et al. (2021). Moreover, the data-related AI-SCs (8) ”Discriminative data bias”, (9) ”Inaccurate data labels”, (10) ”Insufficient data representation”, and (11) ”Inappropriate data splitting” can be mitigated by applying state-of-the-art approaches (partly developed in the safe.trAIn project), such as the ones described in Shahinfar et al. (2020), Ben Saad et al. (2024), Cheng et al. (2018), Geerkens et al. (2024), Sieberichs et al. (2024), and Gannamaneni et al. (2024) during the ”Data Acquisition & Preparation” phase of the MLOps process. Many of these methods can be automated in the MLOps pipeline and hence the evidences for the safety case are generated automatically as described in Zeller et al. (2024a). However, some of the metrics require a human-in-the-loop, since experts need to judge whether the data calculated by the metrics reach a defined goal/threshold. For example, QI^2 presented in Geerkens et al. (2024) provides three-dimensional histograms that quantify the local (non-)linearity of the data, which must be analyzed by human experts.

In contrast, other AI-SCs of the LAISC, such

as (2) ”Inadequate planning of AI performance requirements”, (3) ”Insufficient AI development documentation”, (6) ”Choice of untrustworthy data source”, (7) ”Missing data understanding”, and (13) ”Poor model design choices”, can be mitigated with process-based evidences (such as reviews) and sufficient documentation during the ”Data & ML Life-cycle”. In addition, the AI-SC (19) ”Integration issues” can be mitigated by applying existing and well-established development practices of safety-critical software during the ”ML/SW Integration” phase. Please note that process-based evidences cannot be created automatically in an MLOps pipeline, but within the pipeline the required documentation can partly be generated automatically.

During ”System V&V” phase, the AI-SC (5) ”AI-related hardware issues” can be mitigated by testing an AI-based function in a hardware-in-the-loop environment and showing an acceptable inference time when operating the ML component integrated in the system in the target hardware. Moreover, the AI-SC (20) ”Operational data issues” is mitigated by testing the AI component in the field under operational conditions, e.g., using dedicated test tracks for autonomous cars/trains or run the AI-based function in a so-called shadow mode during normal operation.

In order to demonstrate that the AI-SC (4) ”Inappropriate degree of transparency to stakeholders” is mitigated, documenting the MLOps and data management along the entire lifecycle of the system and providing an end-user manual, describing the basics of decision-making process of the ML model is necessary.

For the AI-SC (1) ”Inadequate specification of ODD”, Weiss et al. (2024) introduced a new approach for the definition and maintenance of an ODD during the development of safety-critical AI-based ADS in safe.trAIn. With this process-based approach, we provide a set of heterogeneous evidences to argue the sufficient completeness and consistency of the ODD in the ”System Design” phase. Since an inadequately defined ODD poses a major safety concern for the entire development, it is important to mitigate this AI-SC.

However, there are AI-SCs in the LAISC for which no sufficient mitigation method or metric is available to show the fulfillment of the derived Verifiable Requirement. Thereby, the mitigation of the following AI-SCs could not be demonstrated in the safe.trAIn project:

Regarding AI-SC (12) ”Problems with synthetic data (Gap between synthetic data and real data)” there are approaches available in the literature to measure the reality gap between real and synthetic data and a theoretical argument has been created on how to handle the concern, see Schnitzer et al. (2024b). However, to the best of our knowledge there are no publications that demonstrate the absence of this AI-SC sufficiently. Also in safe.trAIn it was not possible demonstrate this.

To demonstrate the mitigation of the AI-SC (16) ”Unreliability in corner cases”, methods are needed to identify the corner cases of a defined ODD to show then sufficient performance of the ML model on these corner cases during the ”Model Evaluation” and ”System V&V” phases. Moreover, during the phase ”System Operation & Monitoring in the field” an reliable Out-of-Distribution (OoD) detection method is needed, which is reliably detecting scenarios that have not been part of the ML model training.

The AI-SC (18) ”Uncertainty concerns” requires to demonstrate that the different kinds of uncertainty (domain, aleatoric, and epistemic) defined in Brando et al. (2023) are mitigated sufficiently during the development and a reliable runtime monitoring approach is available. This topic was not addressed during the safe.trAIn project.

In order to mitigate the AI-SC (21) ”Data drift”, we require a runtime monitor which is capable of detecting significant model performance decrease and identifying significant changes in operational distributions automatically during operation in the field in real-time. For both tasks, methods are available in the literature, but the methods have not yet been applied to the use case of safe.trAIn.

The AI-SC (22) ”Concept drift” is not relevant for the use case in the safe.trAIn project, since we assume that the obstacles to be detected by the driverless train are not changing significantly over time (e.g., a human will not evolve during the

lifetime of a regional train).

6. Conclusions and Outlook

In order to assess an AI-based obstacle detection function of a driverless regional train in terms of safety, we developed a safety argumentation strategy in the safe.trAIn project. This strategy is guided by the so-called Landscape of AI Safety Concerns. The LAISC defines a set of AI-specific issues that may negatively impact the safety of a system - the so-called AI Safety Concerns. To demonstrate the safety of the obstacle detection function, we need to show that all AI Safety Concerns are sufficiently mitigated. Therefore, metrics and mitigation methods are specified. The methods are assigned to the phases of the safe MLOps process specified in the safe.trAIn project to continuously develop and assess AI-based functions for driverless trains. For many of the AI Safety Concerns, we identified suitable methods within the safe.trAIn project to demonstrate the mitigation of the concerns. Many of the methods can be implemented in a MLOps pipeline and automatically create evidences for the safety case.

However, for some of the concerns in the LAISC for which no sufficient mitigation method or metric was identified during the project. Future work is to identify suitable methods and metrics and to evaluate their capabilities to sufficiently demonstrate the mitigation of the concerns for an AI-based obstacle detection function of a driverless regional train.

Acknowledgement

Research leading to this paper receives funding from the Federal Ministry for Economic Affairs and Climate Action (BMWK; grant agreement 19I21039A).

References

Adler, R. and M. Klaes (2022). Assurance cases as foundation stone for auditing ai-enabled and autonomous systems: Workshop results and political recommendations for action from the examai project. In *HCI International 2022 – Late Breaking Papers: HCI for Today's Community and Economy*, pp. 283–300.

Ashmore, R., R. Calinescu, and C. Paterson (2021). Assuring the machine learning lifecycle: Desiderata, methods, and challenges. *ACM Comput. Surv. 54*(5).

Ben Saad, A., G. Facciolo, and A. Davy (2024). On the importance of large objects in cnn based object detection algorithms. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 533–542.

Bloomfield, R. and P. Bishop (2010). Safety and assurance cases: Past, present and possible future–an adelard perspective. In *Making Systems Safer: Proceedings of the 18th Safety-Critical Systems Symposium*, pp. 51–67.

Borg, M. et al. (2023). Ergo, SMIRK is safe: a safety case for a machine learning component in a pedestrian automatic emergency brake system. *Software Quality Journal 31*, 335–403.

Brando, A. et al. (2023). Standardizing the probabilistic sources of uncertainty for the sake of safety deep learning. In *Proceedings of the Workshop on Artificial Intelligence Safety 2023 (SafeAI 2023)*.

Burton, S. et al. (2022). Safety assurance of machine learning for perception functions. In *Deep Neural Networks and Data for Automated Driving: Robustness, Uncertainty Quantification, and Insights Towards Safety*, pp. 335–358.

Cheng, C.-H., C.-H. Huang, and H. Yasuoka (2018). Quantitative projection coverage for testing ml-enabled autonomous systems. In *Automated Technology for Verification and Analysis*, pp. 126–142.

Correa-Jullian, C. et al. (2023). The safety case for autonomous systems: an overview.

EN 50126-1:2018-10 (2018). Railway Applications – The Specification and Demonstration of Reliability, Availability, Maintainability and Safety (RAMS) - Part 1: Generic RAMS Process.

EN 50716:2023-11 (2023). Railway Applications - Requirements for software development.

Gannamaneni, S. S., M. Mock, and M. Akila (2024). Assessing systematic weaknesses of dnns using counterfactuals. *AI and Ethics 4*(1), 27–35.

Geerkens, S., C. Sieberichs, A. Braun, and T. Waschulzik (2024). QI^2: an interactive tool for data quality assurance. *AI and Ethics 4*(1),

141–149.

Hawkins, R. et al. (2021). Guidance on the assurance of machine learning in autonomous systems (AMLAS). *arXiv preprint arXiv:2102.01564*.

Houben, S. et al. (2022). *Inspect, Understand, Overcome: A Survey of Practical Methods for AI Safety*, pp. 3–78.

IEC 61508-1:2010-04 (2010). Functional safety of electrical/electronic/programmable electronic safety-related systems – Part 1: General requirements.

ISO/IEC 23053:2021-06 (2022). Framework for Artificial Intelligence (AI) Systems Using Machine Learning (ML).

ISO/PAS 8800:2024-12 (2024). Road Vehicles – Safety and artificial intelligence.

Kelly, T. (2004). The goal structuring notation a safety argument notation. In *Proceedings of the Dependable Systems and Networks 2004 Workshop on Assurance Cases*.

Linardatos, P., V. Papastefanopoulos, and S. Kotsiantis (2021). Explainable ai: A review of machine learning interpretability methods. *Entropy 23*(1).

Roßbach, J. et al. (2024). Evaluating ai-based components in autonomous railway systems: A methodology. In *KI 2024: Advances in Artificial Intelligence: 47th German Conference on AI, Proceedings*, pp. 190–203.

Schlosser, T. et al. (2024). A consolidated overview of evaluation and performance metrics for machine learning and computer vision.

Schnitzer, R. et al. (2024a). "AI Hazard Management: A Framework for the Systematic Management of Root Causes for AI Risks". In *Frontiers of Artificial Intelligence, Ethics, and Multidisciplinary Applications*, pp. 359–375.

Schnitzer, R. et al. (2024b). Landscape of AI safety concerns - A methodology to support safety assurance for AI-based autonomous systems. In *8th International Conference of Safety and System Reliability (ICSRS)*.

Schwalbe, G. et al. (2020). Structuring the safety argumentation for deep neural network based perception in automotive applications. In *Computer Safety, Reliability, and Security. SAFECOMP 2020 Workshops*, pp. 383–394.

Schwalbe, G. and M. Schels (2020). A Survey on Methods for the Safety Assurance of Machine Learning Based Systems. In *10th European Congress on Embedded Real Time Software and Systems (ERTS)*.

Shahinfar, S., P. Meek, and G. Falzon (2020). "how many images do i need?" understanding how sample size per class affects deep learning model performance metrics for balanced designs in autonomous wildlife monitoring. *Ecological Informatics 57*, 101085.

Siebericchs, C., S. Geerkens, A. Braun, and T. Waschulzik (2024). ECS: an interactive tool for data quality assurance. *AI and Ethics 4*(1).

Tocchetti, A. et al. (2025). A.I. Robustness: a human-centered perspective on technological challenges and opportunities. *ACM Comput. Surv. 57*(6).

Weiss, G., M. Zeller, H. Schoenhaar, C. Drabek, and A. Kreutz (2024). Approach for argumenting safety on basis of an operational design domain. In *3rd International Conference on AI Engineering - Software Engineering for AI (CAIN)*, pp. 184–193.

Willers, O., S. Sudholt, S. Raafatnia, and S. Abrecht (2020). Safety concerns and mitigation approaches regarding the use of deep learning in safety-critical perception tasks. In *Computer Safety, Reliability, and Security. SAFECOMP 2020 Workshops*, pp. 336–350.

Zeller, M. et al. (2024a). Continuous development and safety assurance pipeline for ml-based systems in the railway domain. In *Computer Safety, Reliability, and Security. SAFECOMP 2024 Workshops*, pp. 446–459.

Zeller, M. et al. (2024b). Towards a safe mlops process for the continuous development and safety assurance of ml-based systems in the railway domain. *AI Ethics 4*(1), 123–130.

Zeller, M., M. Rothfelder, and C. Klein (2023). safe.trAIn – Engineering and Assurance of a Driverless Regional Train. In *2023 IEEE/ACM 2nd International Conference on AI Engineering – Software Engineering for AI (CAIN)*, pp. 197–197.