Safety Hazards Analysis and Mitigation Strategies for Machine Learning-Based Safety-Critical Systems

Ana Pereira
University of Applied Sciences Berlin (HTW)
Germany
Ana.Pereira@htw-berlin.de

Carsten Thomas
University of Applied Sciences Berlin (HTW)
Germany
Carsten.Thomas@htw-berlin.de

Introduction

- Machine Learning (ML) is increasingly applied for the control of safety-critical Cyber-Physical Systems (CPS). However, the probabilistic characteristics and black-box nature of ML algorithms conflict with the safety culture traditionally adopted a safety-critical system.
- To properly address the safety of ML-based systems, one must implement the rigorous processes prescribed in functional safety, and utilize both: safety strategies known from conventional systems development, and safety strategies developed specifically for ML-based systems.

ML Lifecycle and Hazards Identification

The ML lifecycle is depicted in Figure 1. Our previous work [1] focused on the identification of safety hazards that could be introduced along the ML lifecycle. Table 1 presents a summary of the main results. This extensive and detailed list of hazards provides the foundation for identifying applicable safety strategies addressing each one of these hazards. We believe that the certification of ML-based systems needs to based on both **product-** and **process**-oriented measures.

For example, the "incomplete definition of data" hazard, encountered during the Requirements phase, has a direct impact on the distribution of the final dataset. An inadequate distribution may produce incorrect outputs, and ultimately result in an unsafe system reaction.

Safety strategies based on **product-oriented measures**, such as the fail-safe principle, could successfully mitigate such hazards. In the work of [2], the principle aims to ensure that a system remains safe when it fails in its intended operation, usually by assuming a safe state with reduced functionality. When a failure is identified (reject option is produced) internal approaches, such as explainability components, and as well external approaches to the ML component, such as safety bag architectures or parallel execution of diversified ML components, could be applied to identify or manage wrong model output.

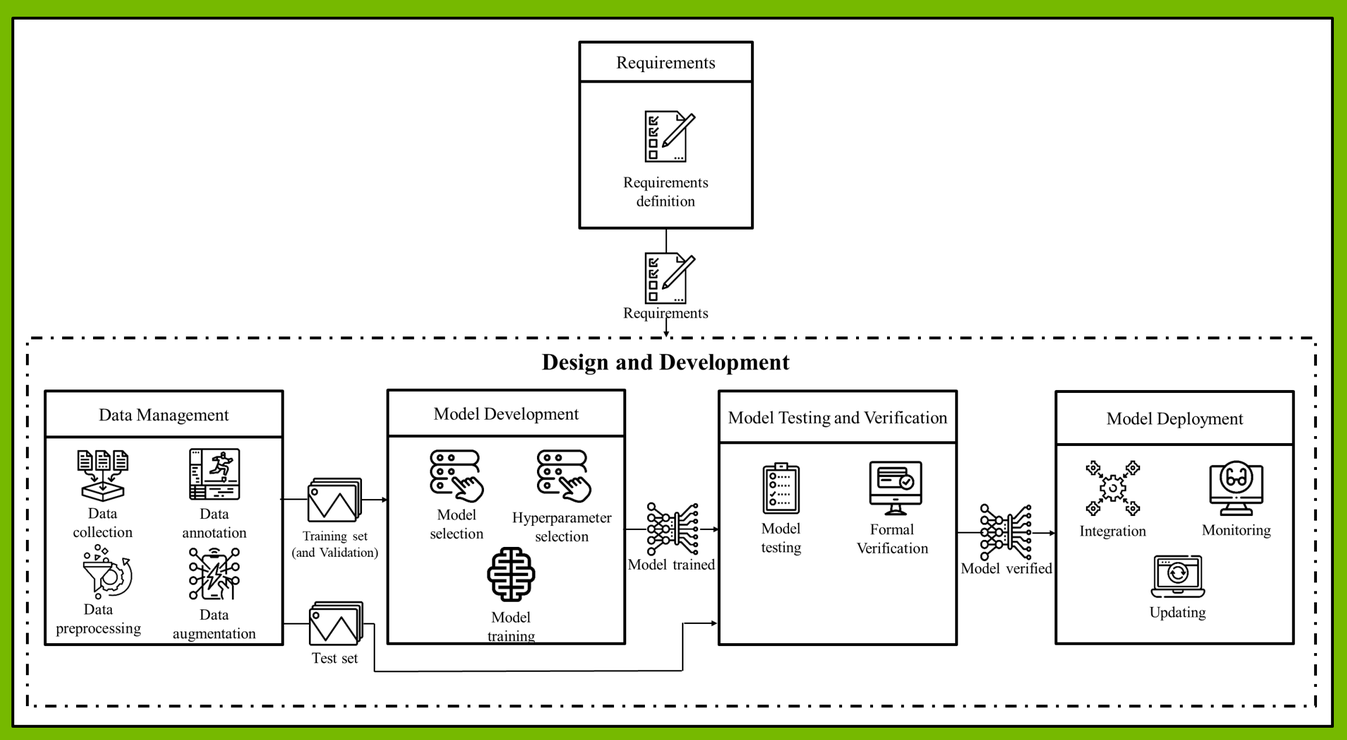

Figure 1: Machine Learning lifecycle.

On the other hand, **process-oriented approaches** based on quality metrics to assess the completeness of data should also be explored. The work of [3] proposes metrics such as scenario coverage for ensuring that the data used in training has possibly covered all important scenarios. Furthermore, the work of [4] also proposed a Feature Space Partitioning Tree (FSPT) technique which splits the feature space into multiple partitions with different training data densities, in order to identify those in which training samples are insufficient. Both techniques mentioned could support the fail-safe strategy. Their implementation aims to trigger a fail-safe behavior for two different cases: when a sample is close to the decision, and as well, when a sample is in an area represented by too few training examples.

Table 1: Overview of the hazards identified for the different phases of the ML lifecycle (here the hazards are briefly described, a detailed identification can be found in [1]).

Phase	Hazard	Description
Requirements	Incomplete definition of data	- Incomplete definition of all operation conditions, including corner and edge cases.
	Incorrect objective function	- Objective functions that overlook or do not correctly prioritize important safety aspects.
	Inadequate performance measure	- Inadequate selection of a performance measure leading to wrong output for portions of the dataset.
	Incompleteness on testing/verification	- Incomplete coverage of operation scenarios and/or inadequate definition of threshold values for performance metrics.
	Inadequate safe operating values	- Incorrect definition of values for "measure of confidence" or a comparison to "reasonable values" (threshold under which the risk level is acceptable).
Data Management	Inadequate distribution	- Inadequate coverage of dataset identified in Requirements phase. - Different distribution of training data and real operation data due to distribution shift. - Different distribution of training data and real operation data due to lost or corrupt data. - Absence or under-representation of rare examples (i.e., corner and edges cases).
	Bias	- Sample bias when samples are not representative. - Measurement bias when data is collected from different sources. - Confirmation bias which focus on information that confirms already held perceptions. - Exclusion bias when important samples and/or features are removed.
	Irrelevance	- Data acquired contains extraneous and irrelevant information.
	Quality deficiencies	- Corrupt data due to measurement issues (sensors accuracy related). - Incorrectly annotated data. - Inadequate delta between cleaned data and real data. - Introduction of non-realistic examples trough data augmentation techniques.
Model Development	Error rate	- Measured error rate differing from real error rate due to finite nature of samples set. - Failing to identify wrong predictions as incorrect (model "silently" fails).
	Lack of interpretability	- Lack of interpretability hides potential misbehaviour.
Model Testing and Verification	Incompleteness	- Insufficient number of test samples which may result in an operational risk much larger than the identifiable actual risk for the test set.
	Non-representative distribution	- Test set is not representative, therefore the testing performance may not be accurate.
Model Deployment	Differences in operational environment	- Failure in the subsystems that provide inputs.
	Adversarial attacks	- Incorrect output produced due to adversarial attacks.

Conclusion

- We believe that our work on machine learning related hazards is a step forward into future holistic approaches for safety engineering and certification of ML-based systems.
- Future work will focus on addressing safety strategies based on product- and process-oriented approaches.

References

[1] Ana Pereira and Carsten Thomas. Challenges of machine learning applied to safety-critical cyber-physical systems. Machine Learning and Knowledge Extraction, 2(4):579–602, 2020
[2] Kush R Varshney and Homa Alemzadeh. On the safety of machine learning: Cyber-physical systems, decision sciences, and data products. Big data, 5(3):246–255, 2017.
[3] Chih-Hong Cheng, et al. Towards dependability metrics for neural networks. In 2018 16th ACM/IEEE International Conference on Formal Methods and Models for System Design, pages 1–4. IEEE, 2018.
[4] Xiaozhe Gu and Arvind Easwaran. Towards safe machine learning for CPS: infer uncertainty from training data. In Proceedings of the 10th ACM/IEEE International Conference on CPS, pages 249–258, 2019.

Acknowledgments

Funded by the German Federal Ministry of Education and Research (Grant Number 01IS18061D), within the scope of ITEA project 17032 CyberFactory#1.