

MOTIVATION

AI/ML-enabled systems are increasingly embedded in software products across industry and day-to-day sectors, becoming central to decision-making, automation, and user-facing functionalities as well as development and technical processes. Despite this growth, the engineering practices used to build these systems remain highly ad hoc and often poorly structured. Nascimento et al. (2020) [8] show that AI/ML development frequently occurs outside formalized processes and in disconnected cycles, where data collection, experimentation, model training, and deployment are performed with ad hoc practices and minimal alignment with software engineering principles.

According to some reports on literature these gaps produce well-known problems in industry: difficulties in scoping AI/ML projects, inconsistent interpretation of requirements, rework due to misaligned expectations, poor data readiness, and integration failures between trained models and production systems. These industry problems are often closely related to the overall software quality overview and metrics that are often overlooked due to the ad hoc and research first process adopted by these types of projects.

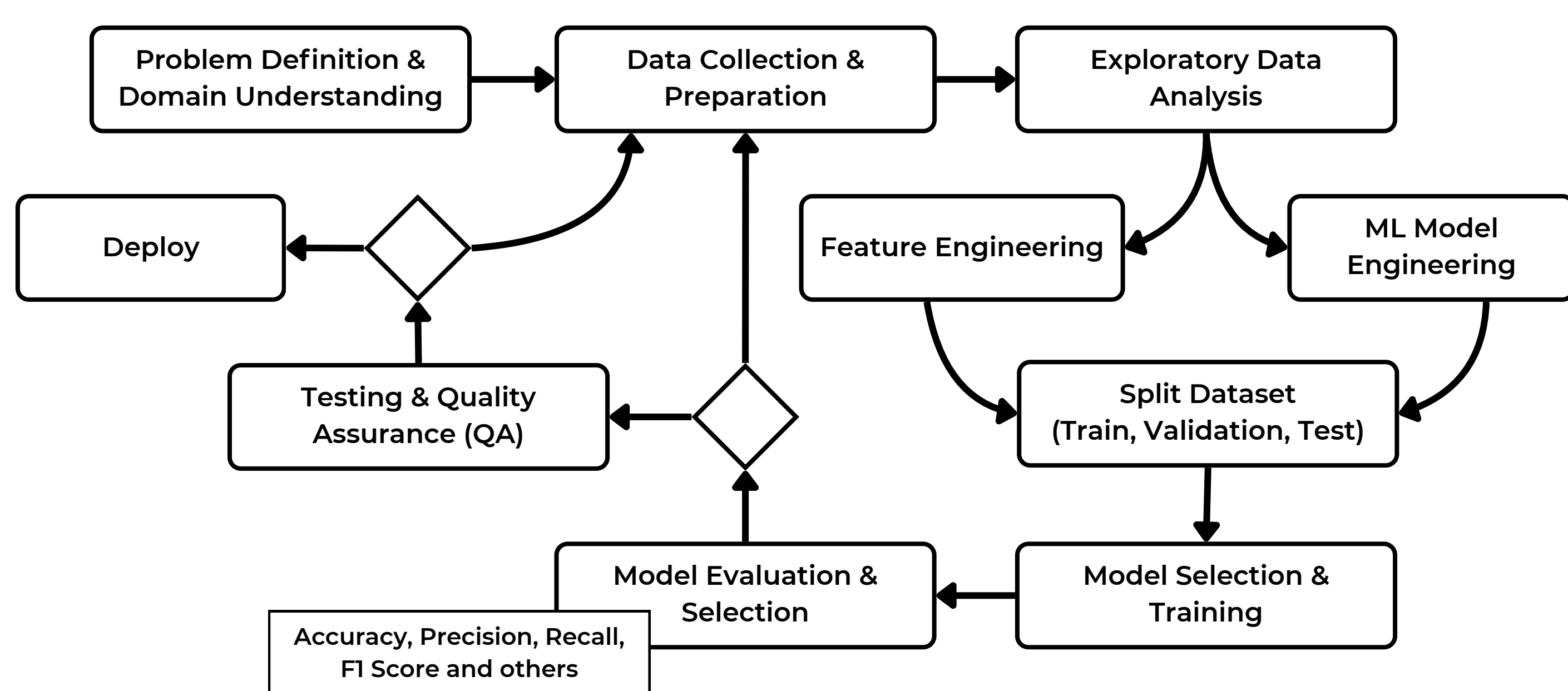


Image 1. Conceptual representation of commonly adopted engineering activities in AI/ML embedded system development

PROBLEM STATEMENT

The current state of practice in AI/ML software development suffers from a lack of integrated, well-defined software engineering processes that explicitly account for AI/ML-specific characteristics such as data life cycle management, experimental iteration loops, model validation, and technical constraints. Several authors have previously proposed unitary and specific improvements for processes under the software development, however maintaining a fragmented process steps on each solution.

This fragmentation on process step solutions leads to misalignment between model development and software engineering practices, resulting in brittle integrations, insufficient quality assurance, and limited traceability of decisions [7]. More importantly, there is no actionable framework that project managers and developers can use to scope, size, and execute AI/ML projects with predictable quality and delivery confidence.

RESEARCH HYPOTHESIS

The main research hypothesis for this work is as follows:

"A structured and empirically grounded software development process life cycle that can measurably improve the quality, predictability, and alignment of AI/ML-enabled software systems."

RESEARCH OBJECTIVES

The objectives for this research can be represented by the following graph flow, where objectives 3, 4, 5 & 6 are a cyclic execution until evaluation provides satisfactory results.

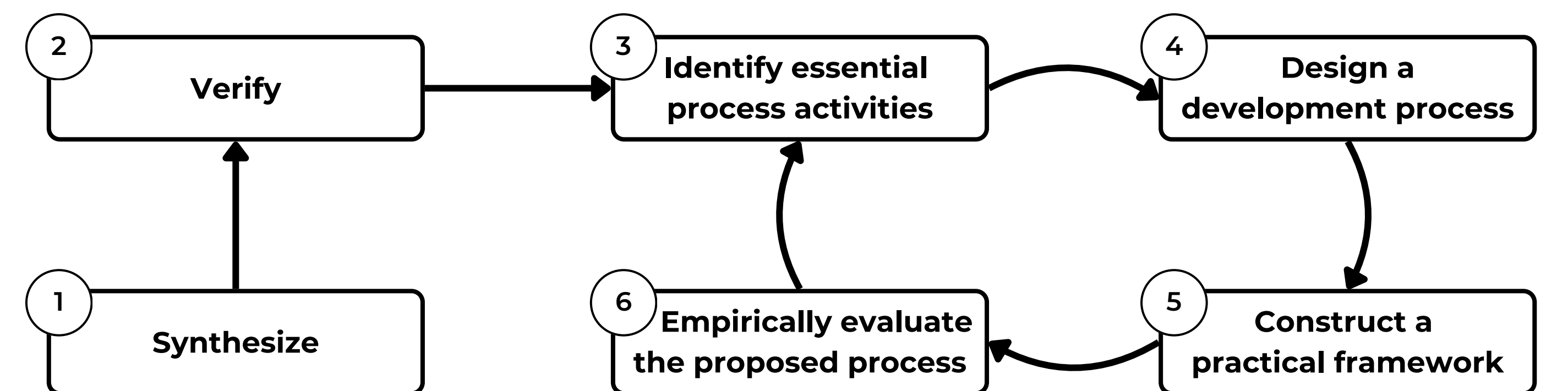


Image 2. Research objectives & incremental evaluation cycle

CONTRIBUTIONS AND IMPACT

Scientific Contributions

The overall expected contributions of this research regarding the scientific view of the Software Engineering for AI/ML field are as follows:

1. An empirically grounded, end-to-end development process specifically designed for AI/ML-enabled software systems, bridging the gap between scattered research insights;
2. Clarifying the essential activities and dependencies in AI/ML development, offering a structured vocabulary and process definition for researchers and practitioners;
3. Integrate quality assurance principles into all phases of AI/ML development, addressing quality concerns raised by past researches [4, 7-9];
4. Delivering a reusable framework that can support future empirical studies on AI/ML project management, process refinement, and SE/ML integration practices.

Industrial Impact

This research have an expected impact on the industry directly to practitioners and AI/ML teams as follows:

1. Improved project scoping and estimation, enabling realistic expectations about effort, complexity, and resource needs;
2. Higher alignment between models and business rules, reducing rework and integration issues;
3. Earlier and more systematic quality assurance, decreasing defects related to data, model behavior, and software integration;
4. A shared process framework that aligns cross-functional teams (software engineers, ML engineers, QA analysts, product owners and others);
5. Transferability across projects, providing organizations with a repeatable, scalable way to develop AI/ML-enabled systems.

Industrial feedback from early observations already indicates that structured quality activities during dataset creation, model evaluation, and integration help prevent common delivery issues and tangibly increase confidence in the final system or at least on model development and evaluation phases.

TL;DR & REFERENCES



References can be found on the page through the QR.

Also, if *TL;DR* just go to the QR!