Multilayered Review of Safety Approaches for Machine Learning-based Systems in the Days of AI

Sangeeta Dey* and Seok-Won Lee*a,b,1

*Department of Artificial Intelligence, bDepartment of Software and Computer Engineering
Ajou University, Suwon City, Republic of Korea 443-749

Abstract

Abstract: The unprecedented advancement of artificial intelligence (AI) in recent years has altered our perspectives on software engineering and systems engineering as a whole. Nowadays, software-intensive intelligent systems rely more on a learning model than thousands of lines of codes. Such alteration has led to new research challenges in the engineering process that can ensure the safe and beneficial behavior of AI systems. This paper presents a literature survey of the significant efforts made in the last fifteen years to foster safety in complex intelligent systems. This survey covers relevant aspects of AI safety research including safety requirements engineering, safety-driven design at both system and machine learning (ML) component level, validation and verification from the perspective of software and system engineers. We categorize these research efforts based on a three-layered conceptual framework for developing and maintaining AI systems. We also perform a gap analysis to emphasize the open research challenges in ensuring safe AI. Finally, we conclude the paper by providing future research directions and a road map for AI safety.

Keywords: Autonomous systems; Intelligent software systems; Machine learning; Safety analysis; Software engineering.

1Corresponding Author
1. Introduction

Recently, artificial intelligence (AI) has received increased attention from many sectors and fields. These diverse areas include autonomous driving, computer vision, medical diagnosis systems, gaming systems, etc. Various industries are overwhelmed by the positive potential of AI. This unprecedented advancement of AI has altered our perspectives on various business functions, including software-intensive intelligent systems development process. Machine learning (ML) is one of the greatest contributors to this revolution. The task of a software engineer has now transformed from writing thousands of lines of code to training, retraining, testing, and maintaining a learning model. However, it is essential to realize that every advancement of technology comes with surprises and concerns, and AI is no exception. AI revolution is not at its inception any more. However, the means to ensuring safety, transparency, and level of fidelity of AI systems are still unclear. Many researchers have already expressed apprehension related to the advent of AI especially in high-risk environments [1, 2]. Moreover, AI systems that equal or even exceed humans in cognitive tasks are both appealing and alarming [3, 4, 5]. Introducing a sense of ethics and morality has long been discussed as the top priority for the days of AI [6, 7]. Therefore, amid the excitement about improved efficiency due to AI, it is worth engaging in discussions on its potential risks, new challenges, and questionable impact on safety issues [8].

The process of reduction of human interventions in many sectors of industries is already underway. Automotive, aerospace etc. are heavily using ML algorithms to paving their way towards increasing autonomy [9]. A fully automated driver-less car is not a far-fetched dream in the days of AI [10]. However, a few recent accidents involving semi-autonomous cars [11] have negatively impacted our trust in full autonomy. It has been argued that the increasing autonomy of AI systems can have enormous impact on humanity [8, 9]. Use of systems that rely heavily on the decision making of an ML model should be a tremendous concern in high-risk environment. Although introduction of ML in controlling
complex engineering systems has been proved to be more efficient than unpredictable erroneous human controls in many cases, the robustness of such ML models should be assessed.

However, as stakeholders from various backgrounds and expertise participate in the process of ensuring safety of a complex systems, a consensus on the fundamental properties of autonomous systems; e.g. controllability, explainability, robustness to uncertain environment, etc. is necessary [12, 13]. Functional safety experts worry about the gap between the vision of the current automotive industry and the scope of the current safety standards landscape [7]. Traditional risk model and safety analysis [14] are also inadequate to handle the immense use of AI in safety-critical systems. Moreover, how the activities performed by experts of diverse expertise integrate to foster an emergent property like safety is not clear. For instance, how the artifacts flow across the various layers of methodical systems engineering at different levels of abstractions is still an open question. It is still unclear how the traceability can be maintained in a collaborative environment where the stakeholders not necessarily speak the same terms. Singla et al. have analyzed the differences between ML-based agile software projects and traditional agile software projects in terms of their execution processes, issues faced and the terms used to describe same concepts in these two types of workload [15].

Therefore, we believe that, we need to analyze the state-of-the-art safety approaches through the lens of a methodological engineering process. To the best of our knowledge, no systematic literature review of AI safety from a software engineering perspective has been done in recent times. The landscape of AI safety and beneficence research [16] discusses a vast range of research topics directed to AI safety. This article includes research areas such as validation, verification, control, security, etc. However, in order to perform a gap analysis on the state-of-the-art AI safety approaches, it is imperative to have a bigger picture of the recent research efforts from an engineering process perspective.

Research questions that we try to answer through our research are:

RQ1: How can we easily comprehend the complexity and challenges involved
in fostering safety of a complex intelligent systems?

Rationale: AI systems today are very complex at many layers. We argue that, not only the technical advances force us to give a deeper thought to safety related issues, but also the diversity of stakeholders participating in the systems engineering process amplifies the complexity. For example, a system level safety engineer may not be fully aware of the impact of the minute loopholes in the software module which vastly relies on ML algorithms. As different stakeholders with diverse expertise work at different levels of abstractions in the process of fostering safety, it is important to visualize the state-of-the art safety approaches from a layered perspective.

RQ2: How have safety concerns been addressed by the researchers along the phases of SE process?

Rationale: We acknowledge the paradigm shift that has taken place in the software development process. However, we believe that, the foundation of software engineering (SE) process entailing phases such as requirements engineering (RE), design, development, validation, verification and maintenance, still provides a strong methodological foundation to the whole process of intelligent software engineering even in this new paradigm. Understanding data requirements, designing the parameters and features of a learning model, training, tuning and testing of the ML model, and finally, maintaining or updating the model over time are new additions to the activities of phases such as RE, design, development, V & V, maintenance respectively. In our study we would like to map the research efforts in the field of AI safety along the phases of SE.

RQ3: What are the gaps in the current research efforts?

Rationale: After we find the answers to RQ1 and RQ2, we can have an accurate visualization of the areas that have not been explored yet. Finding the gap can eventually help us provide possible directions for future research.

RQ4: What are the future directions that may help reduce the gaps?

Rationale: We further want to analyze the existing gaps and provide our preliminary vision on the ways to reduce those gaps. Our paper finds the scope of improved collaborations among diverse stakeholders while analyzing current
state-of-the-art safety approaches from a multidisciplinary engineering process perspective.

In summary, the major contributions of this paper are:

- We provide a conceptual three-layered framework that helps us visualize the inherent complexity in developing and maintaining complex intelligent systems involving various stakeholders.

- We identify the challenges and risks that are of major concern along each layer of the framework.

- We review significant work conducted to address safety-related issues in intelligent systems in the last fifteen years.

- We perform a gap analysis to identify what aspects we are missing in literature and practice.

- Finally, we provide a research road map to expand the ongoing research on safe AI.

The remainder of the paper is organized as follows. In Section 2 we discuss the background of the research on safe AI systems. We summarize the related surveys in this area and explain the position of our survey in Section 3. A detailed research method of conducting a literature review is explained in Section 4. Section 5 introduces the proposed three-layered conceptual framework to better visualize the engineering process of complex intelligent systems. It also elaborates on the challenges and significant research efforts address the AI safety-related problems along each layer of the framework. We discuss our findings of literature review based on the research questions in Section 6. The limitation and threats to validity of this study are discussed in Section 7. Finally, we conclude the paper in Section 8.

2. Background

2.1. What is a “Safe AI system”?  

Before we investigate the formal explanation of a safe AI system, we discuss the definition of ‘safety’. The concept of safety has long been defined and
analyzed from multiple perspectives. Various domains have explained safety in different ways. Leveson explained that safety means absolutely no harm to people and no accidents with or without harming people \cite{17}. According to IEC 61508, “Safety is freedom from unacceptable risk”. This definition leads us to the concept of ‘Risk’ \cite{18}. To analyze safety, the level of risk should be assessed. This definition also seems to be more practical than the previous one as it is almost impossible to engineer a fault-free system. Rather, this definition leaves an existing scope of risk up to a certain level of tolerance. Another definition of safety given by ISO 26262 \cite{19} is “safety means the absence of unreasonable risk”. This definition involves not only the probabilistic analysis of risk, but also the mechanisms to avoid harm and the possibilities of occurrences of such situations.

Recently, with the unprecedented emergence of AI based systems, the concept of ‘safety’ has started to entail a broader concept than the discussed definitions. AI techniques come with various inherent risks and behavior uncertainties due to a vast range of reasons: data-driven behavior (instead of code or rule-driven), self-learning or exploration, black-box nature, etc. As these systems are mostly driven by a few system-level objectives specified by the designers, it is important to ensure that the systems will not try to achieve their goals in an undesirable manner. In other words, the current safety concerns are not limited to failure, non-availability, and wrong outcomes; rather the concerns include reward hacking, negative side-effects, unsafe exploration, and insufficient robustness to distributional shift, etc. \cite{8}. As argued by Varshney \cite{20}, it is not enough to minimize risk by defining a risk-minimal loss function. Instead, it is also important to address epistemic uncertainty in the underlying distribution of training and test instances. Moreover, as these AI systems are deployed in a socio-technical environment, the perception of safety or ‘safe system’ depends considerably on the confidence of society in such systems. From this perspective, a safe AI system should not only be engineered to behave safely but also be capable of explaining its behavior to a wide range of society.
2.2. When is a system “safe enough”?

While nowadays we have a common understanding that no system can be absolutely fail-free, we still do not have a consensus on the safety measurement criteria. We have various safety standards to certify a system as safe enough to deploy. However, the overall quantification (formalism) of how much safe is safe enough vary widely across various domains. Littlewood made a significant contribution in quantifying software safety with the help of evidence-based arguments and the confidence in those arguments [21]. With the help of multi-legged arguments, he showed how diversity in arguments can be utilized to prove dependability of a software system with a certain confidence. For example: if the overall argument is to support a claim of $10^{-4}$ pfd (probability of failure on demand), a statistical leg can support this claim with direct evidence whereas a logical leg can also fine-tune its confidence in the claim based on the confidence in the statistical leg.

Automotive industries that produce AI-based autonomous vehicles have also faced with the same challenge to acquire sufficient confidence in the safe behavior of a newly designed vehicle. As discussed earlier, with the advent of data-driven learning techniques, there is little scope of finding bugs in the code or models. Apart from gaining confidence in the usage of right set of training data, one of the promising ways to ensure safety is to test the system thoroughly in simulation-based or real-world situations. However, the challenge is to make sure that the designers have explored all possible scenarios and surprises that the system can encounter during testing. As Koopman mentioned in [13], miles on road is regarded as evidence of the safe behavior of the vehicle without any clear explanation of how many miles are enough to gain a certain level of confidence in the safety claim. For instance, if we assume there are 100 surprises in total, each arriving in every 100 million miles, then to prove that the systems have encountered all the surprises and has been corrected to handle each of those surprises, it has to be tested on at least one or two billion miles. This is nearly unattainable. Therefore, many researchers are now focusing on designing frameworks to plan to test wisely while limiting expensive on-road testing. We
discuss more on the testing and statistical evidences of AI systems safety in Section 3.3.

3. Related Surveys and our position

Since safety concerns of AI systems have gained attention in the last few years, many literature reviews, technical reports, and surveys have consolidated the relevant state-of-the-art approaches. We collected the related surveys while searching for the primary studies.

1. Based on the search strings (discussed in detail in the next section), we first collected the secondary studies (literature surveys, questionnaires, reports, summary, case studies, etc.). We found more than 40 such studies that matched with the relevant search keywords.

2. We selected only 14 studies based on their relevance to safety, perspective, and coverage. The selection criteria were as follow:
   
   (i) The study is published between 2015-2020.
   
   (ii) The study is peer-reviewed and written in English.
   
   (iii) The study is focused on the safety aspects of AI, ML, intelligent software engineering, safety-critical systems (that use ML heavily).

The studies that review the application of various ML techniques to automate software engineering or other domain (without covering the safety aspect) have been excluded. We have deliberately included three recent studies on the integration of the software engineering process and ML life-cycle as part of our related work (despite violating the third criteria) [22, 23, 24, 25]. These studies do not solely focus on the safety aspects of intelligent software. However, we include them as these studies solidify our view of performing a literature survey on safety approaches from a software engineering perspective.

We can generally categorize the selected secondary studies as— domain-based (concerning a particular domain such as automobile, robotics, etc.), AI algorithm/ ML techniques-based (focusing on a particular type of algorithm or ML technique like deep learning, reinforcement learning), general AI systems-
based(surveying a vast range of safety-related research efforts applicable to all AI systems in general), and engineering process-based (surveys the safety approaches from a systematic engineering process point-of-view that considers collective contributions from diverse stakeholders at each phase).

There are a few domain-specific surveys [26, 27, 28, 29, 30] in the field of safety-critical systems such as driverless cars and robotics, that cover a vast range of engineering aspects such as design, control, architecture, etc. However, it is either difficult to understand the safety concerns of AI algorithms or ML techniques used in these systems [26, 27, 28] or it is difficult to generalize the safety concerns for all safety-critical domains [29, 30]. Numerous reviews have been published on the robustness of ML techniques [31, 32, 33, 34] and assurance of ML-based systems [35, 36, 37, 38]. The target readers of those surveys are expected to have deep technical knowledge in ML/AI algorithms to comprehend such studies as they entail many minute technical details. There are also a few technical reports, research agendas published by Stanford AI research groups [39, 40]. These reports cover a broad range of topics related to AI safety, security, privacy, fairness, trust, etc. While these studies provide an overview of the scope of research contributions in the field of safe AI, analyzing the current research efforts from an engineering process perspective is not in the scope of those studies.

Our paper differs from the other three types of surveys mentioned above in various ways. The objective of this paper is not to discuss the detailed technicality of state-of-the-art safety approaches. Rather our goal is to explore how research efforts fit from a layered systems engineering perspective. Recently, few researchers and practitioners have conducted literature reviews, questionnaires, surveys to understand the challenges of engineering ML-based software [22, 23, 24, 25]. Wan et al. and Ishikawa et al. have conducted questionnaire surveys including experienced ML-based software engineers to summarize the software engineering challenges and corresponding sources of difficulties faced during the development and validation of such software [22, 25]. Serban et al. have conducted a survey including ML practitioners to analyze how software en-
<table>
<thead>
<tr>
<th>Year</th>
<th>Authors</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>2015</td>
<td>Dewry et al.</td>
<td>[39]</td>
</tr>
<tr>
<td>2016</td>
<td>Chakraborty et al.</td>
<td>[32]</td>
</tr>
<tr>
<td>2017</td>
<td>Xiang et al.</td>
<td>[35]</td>
</tr>
<tr>
<td>2018</td>
<td>Mesudt Ozdag</td>
<td>[31]</td>
</tr>
<tr>
<td>2019</td>
<td>Guiochet et al.</td>
<td>[28]</td>
</tr>
<tr>
<td>2019</td>
<td>Gracia et al.</td>
<td>[34]</td>
</tr>
<tr>
<td>2020</td>
<td>Xu et al.</td>
<td>[33]</td>
</tr>
<tr>
<td>2020</td>
<td>Luckkuck et al.</td>
<td>[26]</td>
</tr>
<tr>
<td>2020</td>
<td>Yurtserver et al.</td>
<td>[27]</td>
</tr>
</tbody>
</table>

**Figure 1: Related surveys and out position**

Engineering best practices are being used by the ML teams and how effective those practices are in the case of ML-based systems [23]. Silverio et al. have discussed challenges of trustworthy AI-based autonomous systems in industrial settings and also have provided future directions on reducing the gap between the development and operation of such systems [11]. Lwakatare et al. have summarized 23 software challenges and 8 solutions related to the large-scale ML systems after conducting a literature survey [24]. In quite a similar direction, Anh and Pekka have discussed the challenges and have provided a research agenda on continuous experimentation of AI software in large-scale systems [42]. Few of the challenges discussed in our survey overlap with their findings. However, their studies do not focus on safety. Instead that study focus on the quality attributes of software in general.

To the best of our knowledge, no survey has been done yet to analyze the relevant research areas through the lens for diverse stakeholders and their con-
tribution to the engineering process from a safety perspective. Our paper aims to find the scope of collaboration to fill the gaps in the continuity of the recent research efforts when analyzed from a multidisciplinary engineering process perspective. Fig. 1 summarizes the relevant literature surveys and our position.

4. Research Method

4.1. Research questions

We have already explained the research questions in Section 1. RQ1 is slightly different than the rest of the questions as we try to answer RQ1 based on the overall view of how a complex AI system is conceived, designed, developed, verified, and maintained. We envision a three-layered conceptual framework that facilitates further analysis of the recent research efforts along the layers. We describe our vision in detail in Section 5 and explore the state-of-the-art safety approaches based on our conceptualization of the problem space and solution space regarding safe AI.

4.2. Search strings used

We conducted the search on 29-Oct-2020 with the search strings to retrieve the recent studies as mentioned in Table 1. We initially attempted to search based on the title and abstract. However, we were overwhelmed by more than 10,000 spurious search results for most of the queries. Therefore, we decided to limit our initial search on the titles and follow snowballing method afterward to retrieve other relevant studies that were missed out by the search operation. The search results were filtered by the range of publication years from 2005 to 2020. In case of Springer, the search results were further filtered by discipline and sub-discipline.

4.3. Search Strategies and data sources

We conducted the search in two phases. In Phase-1, we directly searched databases like IEEE xplore, ScienceDirect, Springer-link, ACM Library, Scopus
Figure 2: Study selection process
and arXiv. We collected research papers published within the 15-year time period between 2005 and 2020. After getting a moderate number of search results from these data sources, we followed the forward and backward snowballing approach [43] to find new relevant papers (from the references and the citation lists) in Phase-2.

4.4. Study Selection Process

In Phase-1, we initially found more than 3000 papers directly by performing database search. However, not all of those studies were relevant to our literature review. Our main aim was to collect studies focusing on the intersection of safety engineering, artificial intelligence, and software engineering. Therefore, for example, studies focusing only on safety engineering without considering autonomy, AI techniques, or SE process were excluded from the collection. The detailed inclusion and exclusion criteria are described in Table 2. In Phase-1, we removed duplicates and excluded irrelevant papers after reading the title, abstract, and in some cases introduction. In Phase-2, we read the full-text to thoroughly understand the motivation and contribution of each work. Finally, we selected only 112 papers that we agreed were relevant and significant for our study (Fig. 2).

4.5. Study quality assessment

We assessed the quality of the selected studies based on a set of six questions. A complete list of the questions is provided in Table 3. Most of the questions can have three answers: Yes (Y)=1, No (N)=0, Partial (p)=0.5 except for Q3, Q4 and Q6. Q3 can only have two answers Y=1 for general and N=0 for a domain-specific solution. Similarly, Q4 has two answers Y=1 for contribution to research and N=0 for contribution to practice. In general, the papers that scored above 4 out of 7, were selected. Each of the selected papers was thoroughly assessed by each of the authors individually and any disagreement was discussed till a consensus was reached.
Table 1: Search strings used to retrieve relevant primary studies

<table>
<thead>
<tr>
<th>Objective</th>
<th>Search String</th>
</tr>
</thead>
<tbody>
<tr>
<td>To retrieve studies on Artificial Intelligence and safety or trust or reliability or assurance-related concerns and corresponding solutions</td>
<td>“artificial intelligence” AND (safe OR trust OR reliab OR assurance)</td>
</tr>
<tr>
<td>To retrieve studies on safety, reliability, risks or robustness or assurance-related concerns of machine learning or deep learning and corresponding solutions</td>
<td>(“machine learning” OR “deep learning”) AND (safe OR robust OR risk OR reliab OR assurance)</td>
</tr>
<tr>
<td>To retrieve studies on automotive industries or any autonomous/unmanned safety-critical systems and safety-related issues.</td>
<td>(auto OR unmanned OR safety-critical) AND (safe OR verification)</td>
</tr>
<tr>
<td>To retrieve studies on machine learning (especially neural networks)-based systems verification or testing.</td>
<td>(“machine learning” OR “neural networks” OR “deep learning”) AND (verif OR testing)</td>
</tr>
<tr>
<td>To retrieve studies that focus on the correlation between software engineering, requirements engineering, and machine learning.</td>
<td>(“software engineering” OR “requirements”)AND “machine learning”</td>
</tr>
</tbody>
</table>

Table 2: Overview of the inclusion and exclusion criteria

<table>
<thead>
<tr>
<th>Inclusion Criteria</th>
<th>Exclusion Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Primary studies or safety standards.</td>
<td>1. Secondary studies.</td>
</tr>
<tr>
<td>2. Studies published in the English language.</td>
<td>2. Duplicate studies. (only longer and more complete versions were accepted.)</td>
</tr>
<tr>
<td>3. Peer-reviewed studies except for the studies published on Arxiv.</td>
<td>3. Studies that only focus on safety-related research (safety lifecycle, safety artifacts, etc.) without consideration of the recent advancement of AI.</td>
</tr>
<tr>
<td>4. If the study is published on Arxiv then it should have at least one citation by a peer-reviewed paper.</td>
<td>4. Short studies (less than 4 pages).</td>
</tr>
<tr>
<td>5. Studies that focus on AI, ML intelligent Software Engineering and corresponding safety-related issues.</td>
<td></td>
</tr>
</tbody>
</table>
Table 3: Quality assessment criteria

<table>
<thead>
<tr>
<th>QA Scenario Component</th>
<th>Security Requirements Components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Are the motivation and goal of the study well explained in the paper?</td>
<td>Y=1, N=0, p=0.5</td>
</tr>
<tr>
<td>Is the proposed approach adequately explained with necessary details?</td>
<td>Y=1, N=0, p=0.5</td>
</tr>
<tr>
<td>Is the study applicable to all domains or to a specific domain?</td>
<td>Y=1, N=0</td>
</tr>
<tr>
<td>Does the study provide value for research or practice?</td>
<td>Y=1, N=0</td>
</tr>
<tr>
<td>Is there any discussion about the results or threats to validity?</td>
<td>Y=1, N=0, p=0.5</td>
</tr>
<tr>
<td>Does the study has a section on related work?</td>
<td>Y=1, N=0</td>
</tr>
<tr>
<td>Does the study provide future direction or open scope for further research or investigation?</td>
<td>Y=1, N=0, p=0.5</td>
</tr>
</tbody>
</table>

Except for one paper that was published in IEEE software magazine, all other papers qualified to be part of our study. The reason we excluded that 4 page long paper on “Software engineering for machine learning applications” was it lacked detailed explanation of the solution and discussion on the limitation as it was only a theme issue. We made another exception regarding the quality assessment in case of safety standards. Most of the articles discussing standards are from web-based search or part of technical reports, etc. These articles did not go through the rigorous quality assessment process as per the criteria mentioned in Table 3. Instead, they have been selected based on their timeliness and relevance to our study.
5. Three-layered conceptual framework for safety-driven AI system engineering

As appropriately argued by Koopman, many different areas require coordination to ensure safety [44]. Acknowledging the fact that there is a cognition gap among the various disciplines involved in the process of risk analysis, we need to provide a set of guidelines to the diverse stakeholders. To formalize the participation and responsibility of each stakeholder, we propose a three-layered framework to conceptualize complex intelligent systems (Fig. 3). The framework shows how a safe output space is achieved for such systems starting from an initial incomplete problem space. We can visualize how diverse the stakeholders and the primary artifacts are, in each layer at a system level and ML-based component level. In the days of AI, at the ML-based component level, the stakeholders come from AI or ML specific background. This was not very common in the case of traditional systems without any machine intelligence. These paradigm shifts bring in a lot of challenges in exploring the problem space and solution space. The layers of the framework are listed below:

- **Problem definition layer:**
  
  In this layer, at a system level, domain experts and requirements engineers work closely with users to gather requirements. The aim of these participants is to understand the expectations of the users and the society, goals of the system, possible situations/scenarios, etc. With the introduction of ML-based components, conceiving a rich problem space has become more complex than in traditional systems. ML experts, data scientists, HCI experts need to work closely with system-level stakeholders to understand the ML model requirements, data requirements, domain definitions, quantitative targets, etc.

- **Safety-driven modeling and development layer:**
  
  In this layer, system engineers and safety experts work in collaboration to perform system-level risk analysis, risk handling of emergent behavior of
the system. In other words, in this layer risk-minimizing safety goals, requirements, etc. are derived from system-level risks. Safety standards are followed throughout the process of designing and developing the system. At the component level, inherent risks of ML models, effects of adversarial attacks, corner cases are identified. Unlike traditional software-intensive systems, ML-based components are designed and trained based on objective functions following AI/ML specific safety standards. Safe learning decisions on the safety-aware rewards, penalty, etc. are made by the ML experts at this layer.

- **Verified safety compliance layer:**

A system is assumed to be acceptable safe only when the involved stakeholders gain enough confidence in the emergent safe behavior. Traditionally, a system is usually verified by formal models and various types of testing. QA experts and users play a key role to come to a consensus regarding the level of safety of the system. However, safety assurance and verification have become extremely challenging with the inherent uncertainty and data dependency of ML models. Therefore, at this layer, ML experts, HCI experts, data scientists play a crucial role not only to validate training and testing data but also to evaluate overall robustness, scenario coverage, performance, etc. of the ML-based component.

All the above-mentioned layers are based on a layer of basic factors, like assumptions, level of autonomy, regulatory constraints, financial constraints, ethical constraints, data constraints, etc. This framework provides better visualization of how a safe output space of AI systems is designed across the layers starting from the initial naive problem space.

### 5.1. Layer-1: Challenges and research efforts

In the last few years, researchers started discussing the challenges faced by software engineers while working on ML techniques for complex AI systems [22, 45, 46, 47]. As most of the current intelligent systems are software-intensive,
Figure 3: An integrated three-layered conceptual framework for AI systems engineering
we choose to analyze the challenges from a software engineering perspective in
detail. Software research community has always asserted the importance of ini-
tiating the engineering process by having comprehensive understanding of the
problem space and expectations of customers (users). We reflect the same under-
standing in our layered framework (Fig. 3). In order to successfully explore the
safe solution space in Layer-3, the first step should be to start with exploration
of the rich and relevant problem space. Therefore, in this subsection, we ex-
plain the challenges faced by requirements engineers, domain experts, and data
scientists in the early phase of engineering.

(1) Shortcomings of traditional requirements engineering-driven

approach:

With the advent of deep learning, the traditional end-to-end RE process
became futile. It is hard to make users’ expectations explicit when the operating
environment itself is extremely uncertain. Stating all possible ‘shall’ statements
in the early phase of engineering is impractical with the inherent uncertainty and
lack of clarity on the relevant context. Bosch et al. discuss that requirements-
driven approaches may need to be complemented by other approaches [48].
The authors identify three distinct approaches that may coexist during the
development of intelligent software:

(i) RE-driven: Development according to well-understood specifications.

This approach may be used only when the new feature is well understood.

(ii) Output/data-driven: Development according to a given quantitative tar-
get. The goal here is to improve the metric by experimenting with different
solutions.

(iii) AI-driven: Development of components using ML/DL based on an al-
ready available large dataset.

The authors envisioned a holistic integrated development approach (DevOp)
incorporating all the above-mentioned approaches in which the system consists
of traditional software and AI components and also there is scope for continuous
deployment. However, this integrated framework requires further validation
and comprehensive guidelines for its usage. We need to further understand the
real challenges at Layer-1 by thoroughly investigating the core issues from RE perspective.

(2) Understanding the problem domain:

Problem domain understanding is equally important in the case of traditional and intelligent system engineering practices. It is always one of the most complex activities during the early phase of engineering. The complexity is even higher in systems with ML components where the system learns about the domain-specific concepts through training data [22, 49]. Rahimi et al. [50] explain this challenge using an example from an automated pedestrian collision avoidance system. The concept of ‘pedestrian’ is not clear in terms of what it means in a particular context. A person walking a bike (instead of riding it) or a person in a wheelchair should also be included in the semantics of pedestrians to ensure safety. The author suggested that a thorough understanding of the domain is possible by:

(i) Benchmarking the domain

(ii) Interpreting the domain dataset

(iii) Interpreting the domain learned by ML model for further validation

The authors proposed to create a domain ontology based on the web search result of the domain concept (‘pedestrian’ in the example). This ontology can be further used to assess whether the training data represents the problem domain as depicted in the ontology by extracting features from the dataset. Incorrect or missing correlation between ontology element and the dataset can be helpful in identification of the gap or incompleteness of the dataset.

(3) Setting quantitative targets (functional requirements):

The main objective of RE activities for any system is to state the users’ expectations explicitly. However, setting explicit quantitative targets (which are often referred to as functional requirements of ML component) is a great challenge, not only because it is hard to declare desired output, but also it is difficult for users to understand ML related performance metrics such as recall, precision, etc. [51]. Data scientists need to take part in RE actively to help the client to set a metric that is not too technical, yet scientific enough to
measure the performance. For example: ‘Lift’ (improvements of performance) can be explained by introducing a comparison of ML predictions to random predictions. More about these topics are discussed in the upcoming sections on setting targets for validations and testing.

(4) Setting qualitative targets (non-functional requirements):

While establishing quantitative targets is already gaining the attention of the research community, a few researchers are also discussing qualities or non-functional requirements associated with the functional requirements [52, 53]. It is argued by Horkoff, in today’s time we are unknowingly relying more on quality than quantity to gain trust [53]. For example: rather than relying on 99.99% of the accuracy of the target to ensure safety, more efforts need to be invested in justifying transparency, testability, explainability, reliability, fairness, etc. She also mentioned that further research needs to be conducted regarding a thorough understanding of the new definition, catalog, target, and trade-offs for NFRs for different kinds of ML algorithms used in intelligent systems. Nakamichi et al. [54] made a recent contribution by proposing a quality model, quality characteristics, and a measurement method for ML-based software systems. One of the most important qualities from the safety perspective is the robustness of the ML model, especially for perception tasks where a slight change in the input can cause misclassification. Hu et al. [55] proposed how to formally specify robustness requirements for such ML-based component by identifying the invariant and equivariant. Each robustness requirements mainly consists of three components: a formal definition of transformation over the inputs, a range of values for transformation parameters, and a corresponding indication of invariant and equivariant. However, deriving an exhaustive list of invariant and equivariant is an open area of research for ML-based components deployed in an uncertain environment.

(5) Lack of requirements analysis and modeling techniques to address uncertainty:

To analyze the intrinsic uncertainty and unpredictability of ML models, concrete analysis and modeling technique is required to be performed during RE
phase. However, not enough attention has been paid to end-to-end modeling and analysis methods. Recently, F. Ishikawa proposed a goal-based evidence-driven RE modeling to analyze the goals and unresolved uncertainty of ML models [56]. The authors explained how the model can depict the evidences captured by experiments and operation to validate or invalidate the feasibility or performance level. More efforts need to be invested in developing automated tools to link between decision making by the ML models and the collected evidences during testing and operation.

(6) Explaining ‘black-box’ to users:

‘Safety’ and ‘trust’ are correlated. Lack of transparency/explainability leads to weak assurance of safety and as a result users or the society as a whole lack trust in such systems. Therefore, some of the literature emphasizes the importance of the explainability of AI systems. Kohl et al. [52] report that a lack of understanding of some phenomenon motivates certain groups of stakeholders to seek explanation in a certain context. Therefore, we can consider including explainability requirements as a new addition to the non-functional requirements family. While eliciting explainability requirements, other requirements, such as security and cost, may conflict with them. This will eventually lead to exploring the rich problem space even further by performing trade-off analysis. Vogelsang and Borg mentioned that in case of ML, explainability is twofold: explain what has been learned and how each prediction has been made by the model (in the context of the predictive model) [51]. Although explainable AI systems have been garnering attention recently, the ML research community is yet to formalize situations that should demand an explanation. Moreover, further research is necessary on the appropriate level of abstraction to attain the required explainability of a system.

(7) Declaring hidden consumers:

For any system, it is of the highest priority to know the target users and consumers of the output. Complex AI systems with one or more ML components are often developed without a proper understanding of who is accessing the output (e.g., the prediction output directly or log files indirectly later) [45].
Inadequate access control strategy can have a significant impact on the overall emergent safe behavior of the systems as undeclared consumers may use the output in an unintended way [47]. Belani et al. express their concern over undeclared consumers as this hidden risk factor has not been sufficiently scrutinized [49]. We believe that identifying and documenting the right set of consumers are essential activities at Layer-1 to safeguard the system from unintended usage in the future.

Discussion:
We believe that research on RE for AI systems is at its beginning. Only a limited part of the literature has discussed the challenges faced at an early stage of AI systems engineering. Instead, research has focused on the technical excellence of AI algorithms and their applications. However, as we discuss the challenges at Layer-1 in this subsection, we summarize the current understanding of RE and SE research community for AI systems in Table 4. This table shows the latest guidelines for each of the phases of RE and specifies recent advances on new types of requirements that should be part of the requirements specification in the days of AI.

5.2. Layer-2: Challenges and research efforts

In this section, we focus on the analysis and design issues that are closely related to safety concerns. As shown in the framework, in Layer-2, safety engineers, designers, developers, ML experts are required to collaborate to explore the safety constrained design space. Therefore, we will discuss risk analysis, safety requirements, safety standards of autonomous systems.

(1) Safety-driven design of complex systems:

The preferred way to build a safe system is to consider safety from the very beginning of requirement engineering. Firesmith described a taxonomy for four kinds of safety requirements named as- pure safety requirements, safety-significant requirements, safety constraints, and requirements for safety systems [57]. However, elicitation of safety requirements was out of the scope of this work. To the best of our knowledge, the necessity of a change in the tradi-
<table>
<thead>
<tr>
<th>RE Activity</th>
<th>Guidelines</th>
</tr>
</thead>
</table>
| Requirements Elicitation [22, 50, 49, 54] | - Include data scientists and legal experts  
                        - Benchmark the domain  
                        - Elicit new data sources  
                        - Identify sensitive/protected features of the data  
                        - Situations that demand an explanation to help users |
| Requirements Analysis [54, 56] | - Discuss performance measures that are easily understandable for users  
                                - Conditions for data pre-processing, cleaning, etc.  
                                - Required level of automation needed in the process to meet the constraint  
                                - Evidence-driven goal-based requirements modeling and analysis  
                                (Insufficient attention is paid to requirements analysis activities) |
| Requirements Specification [51, 52, 53, 51, 54, 55] | - Data requirements  
                                - ML model requirements  
                                - ML process requirements  
                                - Quantitative targets  
                                - Measurable qualitative targets  
                                - Explainability requirements  
                                - Ethical and legal requirements  
                                - Robustness requirements |
| Requirements V & V [50, 55] | - Confidence in data  
                                - Data dependencies  
                                - ML process requirements  
                                - Quantitative targets  
                                - Robustness targets  
                                (Not enough work is done on requirements validation) |
| Requirements Evolution [22, 46] | - Documenting dataset and model versions  
                                - Data dependencies  
                                - Requirement-data-feature-output traceability  
                                (Insufficient attention is paid to requirements evolution) |
tional accident model due to the emergence of software was first identified by Leveson [58]. The authors explained how the traditional accident model could be enhanced by accommodating social, cultural, and organizational aspects. A complete hazard analysis methodology named STAMP (System Theoretic Accident Model and Process) was proposed by researchers in [59] and [60], in which accidents were viewed as control flaws (failure in the interactions among the components) instead of a component failure. This methodology not only helped to analyze the source of hazards and deriving safety constraints but also helped to make appropriate design decisions to enforce the derived constraints. This safety-driven design approach STPA (System Theoretic Process Analysis) is explained in [61] with the help of a case study on spacecraft. Although this work encompassed the socio-technical factors influencing the design of a safe system, it precludes the hazards caused by human error. Moreover, this work does not exclusively address the inherent uncertainty and opacity of ML-based software-intensive systems. Therefore, we will not probe this research further.

Kuper et al. introduced the concept of verification-friendly design of neural networks [62]. Varshney explained how four common safety strategies (inherently safe design, safety reserves, safe fail, and procedural safeguards) can be mapped to an ML context [20]. The authors emphasized that ML models are very different in terms of the inherent uncertainty in train and test data, and their probability distribution. Rejecting a less confident decision made by the mode can be an option to fail safely. However, it is important to define the decision boundary carefully as the distance from the boundary is not always inversely proportional to confidence. A part of the input space with low density can contain much epistemic uncertainty as the boundary may be based on inductive bias. In the same direction, Gu and Easwaran proposed Feature Space Partitioning Tree (FSPT) to partition feature space and to reject input instances from the low-density feature space [63]. In the case of reinforcement learning, safe outcome has been confirmed by introducing risk-aware policies and rewards [64], safety-aware planner [65], safety supervisor [66], etc. Although these contributions were very significant in terms of ensuring safety at the ML component
level, they lack insight into how the safety standard can be defined and mapped against system or sub-system level safety goals.

(2) Lack of method to model uncertain environment including human behavior:

For safety-critical AI systems, combinations of numerous variables lead to an unlimited number of situations to model, which results in an impractical verification process. The complexity amplifies when human is in the loop. Human's behavior is mostly variable and uncertain. Therefore, for semi-autonomous systems, the problem of environment modeling is two-fold. Research on verification of AI systems can get momentum only if there is a well-understood technique to model the uncertain environment. As complete formal modeling of the uncertain environment seems to be challenging by many researchers, attention is paid towards introspective modeling of the environment. For example: identifying and analyzing the assumptions that a system makes about its environment can be helpful to verify whether the system is capable of monitoring the right variables or not. Seshia et al. have addressed such issues with the help of control theory. Extraction of monitorable assumptions is proved to be feasible for simple controllers \cite{67, 68, 69}. To address the issues of modeling unpredictable human behavior, one way could be gathering data about real and simulated environments to learn about the environment model. Some researchers have shown it to be effective for verification and control of an autonomous vehicle to generate human behavior models from driving simulators and human subject experiments \cite{70, 71, 72}.

(3) Scenario-based safe design and development of autonomous systems:

For safety-critical systems like driverless cars, deployed in an uncertain environment, collecting sufficient safety requirements is an exhaustive task. The risks and potential hazards are mainly hidden in the operational world and its numerous variations. Therefore, very recently, researchers started focusing on purposefully varying the operation scenarios of systems to elicit safety requirements in the early phase of the system life-cycle. Bach et al. \cite{73} presented a
methodology for a model-based design of scenarios from real-world test data. Abstraction of temporal and spatial information act as the key enables of not only coherent modeling of scenario but also support specification of requirements and derivation of test cases. In a similar direction, Till et al. [74], proposed three levels of abstraction for scenarios along the V-model of development process following ISO-26262 (this standard will be discussed later). These researchers elaborated the concept of ‘Scenario’ and defined three layers of abstractions of scenarios:

(i) Abstract functional scenario in concept phase: Involves identifying semi-formal hazardous scenarios including operational scenarios and malfunctioning behavior.

(ii) Detailed logical (technical) safety scenarios in development phase: Involves describing scenarios including parameter ranges of the state values that are used to represent functional scenarios.

(iii) Concrete scenarios for validation and verification: These scenarios represent the operational scenarios with concrete values of each parameter. Proposed process steps of the usage of such safety concerned scenarios can be very helpful to generate the artifacts at each step and to maintain traceability among them.

(4) Safety standard to guide for analysis and design process:

Standards are believed to be one of the best ways to guide and assess (in later phases) the development of a particular system through specification. Especially for AI systems, standards can provide explicit specification/requirements for explainability, robustness, fail-safe design [75]. ISO and IEEE are the two leading bodies that have been developing various standards regarding safety requirements of autonomous machinery. Table 5 summarizes some of the relevant standards in this area. However, none of them exclusively considers AI. The use of ML commences an overall paradigm shift in the design and development process. Unfortunately, these standards fail to ML-specific concerns. Nevertheless, some of the safety considerations to ensure safety are rightly identified by Google [76].

1. Is the objective function appropriate?
2. Has the exploration space been sufficiently constrained?
3. Does the model’s training reflect the current real world?
4. Can the risk of data poisoning be mitigated?
5. Has the AI system been adversarially tested?

Best practices to address these concerns should be collected in formal standards. The formulation of safety standards entails risk analysis, risk control, and risk monitoring. These standards are mostly under development. Formalizing AI safety standards is an ongoing endeavor. Many researchers are working on the same from different perspectives. For example, Ozlati and Yampolsky mention the diversity of AI systems in and considered that different categories of AI systems may have diverse risks and mitigation strategies. Therefore the authors suggested using a modified Delphi methodology study as a starting point of a standing body that can develop and evaluate AI safety standards under AI SDO. Keeping the diversity of AI systems in mind, the authors recommended that the modified Delphi study should cover separate risk assessments for different system categories. Luo et al. focused on environment-centric safety requirements of automated unmanned systems. Environment safety requirements are elicited from the entities of the environment- other systems, human, constraints. The authors classify such requirements along MAPE-K process. After conducting a literature survey, the authors conclude that few gray areas need further research, such as methodology to solve safety concerns (collision avoidance), optimizing safety constrained learning technique (MDP), etc. Most discussed domain-specific standards in the days of AI: Recently, the automotive industry has made significant progress in developing and testing driverless cars. However, as of today, neither the industry nor the government can fully assess the safety of self-driving cars. Therefore, there is a sudden rush to set standards for such AI-based autonomous vehicles among the public organization like IEEE and also public sectors like Safety First for Automated Driving (SaFAD) led by Audi, BMW, etc. In this subsection, we will explain some of the standards related to an autonomous vehicle in detail.
ISO 26262. ISO 26262 is the derivative of IEC 61508 [79] [19]. It mainly covers automotive development, production, and maintenance of safety-critical systems. The key component of ISO 26262 is the automotive safety lifecycle (ASL) which describes the fundamental concepts of safety plan, safety manager, safety review and audit. ASL consists of six phases: management, development, production, operation, service and decommission. This standard also defines the automotive safety integrity level (ASIL). Based on safety analysis of the critical functions of the system, a risk analysis is performed. The risk analysis combines the probability of exposure, controllability of a driver, the severity of outcome resulting in ASIL from A to D. As per this standard, development and verification practices should correspond to the corresponding ASIL. However, as argued by Borg et al. [36], with the advent of ML algorithms (especially less transparent DNN), traditional standards fall short. Salay et al. analyzed ISO 26262 from ML perspective to identify the key factors of conflict [80]. The researchers found gaps in the software development requirements of ISO 26262, and proposed requirements to fill those gaps. One of the major contributions of their work is the elaboration on connecting data to safety concerns while working with ML algorithms. Prior knowledge of the function to be performed by ML component plays a significant role in fostering safety. The ways input can change without affecting the output are termed as invariants. Invariance to lighting level, positions, etc. is directly associated with the safe outcome of the learning model. Similarly, equivariants describe types of change in input that should result in a particular type of change in the output. Along with these two specifications, different kinds of constraints like probabilistic, pattern-based, and context-based constraints on the input and output and control the safe behavior of ML component. Moreover, analyzing data distribution, its coverage of edge-cases also enable developers and safety engineers to assess the expected safe behavior of the AI-based system. The researchers also mention that model selection, feature selection, training, and testing specification are the key artifacts in arguing the safety case of an ML algorithm.
UL-4600. Koopman et al. working with UL (Underwriters’ Laboratories) proposed an initial draft of UL-4600 standard for fully autonomous vehicles [81]. To the best of our knowledge, UL-4600 is the latest and the most advanced standard that aims to address the challenges of the full autonomy of HAV.

The traditional standards lack flexibility as they are usually census-based and updated every 5 to 10 years. However, in the days of AI, developers are exploring new technologies rapidly to provide a better solution to the concerned problem. Therefore, flexible standards like UL-4600 gained attention in recent times. UL-4600 does not prescribe direct guidance to the proper development process. Rather, it guides on building the safety case (which will be discussed later in the next section) for HAV [82]. A safety case is a very important artifact for safe systems design. Identified topics that are planned to be addressed in this standard are:

(i) Definition of operational design domain (e.g., weather, scenarios, etc.)
(ii) ML faults (e.g., training data gaps, etc.)
(iii) External operation faults (e.g., fault of other vehicles, etc.)
(iv) Faulty behavior of the non-driver humans (e.g., pedestrians, etc.)
(v) Non-deterministic system behavior (e.g., test planning, etc.)
(vi) High residual unknowns (e.g., requirements gaps, etc.)
(vii) Lack of human oversight (e.g., passenger handling, etc.)
(viii) System-level safety metrics.

This standard emphasizes that it is more effective to continually evaluate and improve the residual risk present in the system than to conform to a standard during deployment. Therefore, developers need to be actively involved and take responsibility for safety risk identification and self-assessment over the iterations of the development life-cycle.

Discussion: In 2020, several new standards from various sources for AI-based systems are likely to be rolled out [90, 91, 92, 93, 94, 95, 96]. We could not discuss these standards in detail as most of the standards are still under development and not many open-access documents are available regarding these standards. However, as Riccardo Mariani [97] mentioned that, with the arrival of the new
Table 5: Overview of the safety standards and the relevant autonomous systems

<table>
<thead>
<tr>
<th>Safety Standards</th>
<th>Autonomous Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISO/DIS 3691-4</td>
<td>Driverless trucks</td>
</tr>
<tr>
<td>ISO 13482</td>
<td>Personal Care Robots</td>
</tr>
<tr>
<td>ISO 19014</td>
<td>Earth moving machinery</td>
</tr>
<tr>
<td>ISO17757</td>
<td>Earth moving machinery and mining</td>
</tr>
<tr>
<td>ISO 18497</td>
<td>Agricultural machines</td>
</tr>
<tr>
<td>IEC 62267</td>
<td>Automated urban guided transport</td>
</tr>
<tr>
<td>ISO 26262</td>
<td>Road Vehicles</td>
</tr>
<tr>
<td>ISO/PAS 21448</td>
<td>Road Vehicles</td>
</tr>
</tbody>
</table>

safety standards from various sources, it is expected that those standards will be written from diverse perspectives of AI-based systems. Therefore, the main goal should be to analyze those standards and minimize the overlap.

5.3. Layer-3: Challenges and research efforts

As rightly argued by J. Morton et al., Trust is the prime concern that needs to be established before a complete release [98]. Many researchers focused on validation and verification of the system life cycle to prove that the system is acceptably safe enough to be deployed. In this subsection, we will explain the challenges to perform end-to-end validation and verification of AI systems. Seshia et al. explained the shortcomings of the current practices and proposed to move towards the paradigm of Verified Artificial Intelligence [99]. Further, we discuss the directions that researchers have been following recently to address those challenges.

(1) Lack of formal specification to verify the system:

Traditional formal verification is mostly founded upon strong mathematical statements of the way system should behave. However, for ML-based sys-
tems, it is extremely difficult to describe the expected behavior precisely in a mathematical way. For instance: there is no concrete method for a module of an autonomous car that uses computer vision to perform object recognition, human-object classification. As it seems to be nearly impossible to formally specify the exact behavior of any ML component, few researchers have tried to solve the problem from a different perspective. Instead of formalizing the component level behavior, end-to-end system-level behavior specification can be used for verification purposes. Seshia et al. suggested that specification mining techniques can be used in such cases [100]. Jan Leike et al presented a suite of reinforcement learning environment to assess the conformity with the intended safe behavior [101]. The authors also classified various safety problems in the case of RL. For example, specification problems, safe interruptability, reward gaming, safe exploration, etc. However, the cost of verification and validation of such components is not trivial. Especially for domains like automotive systems, aircraft systems, verification approach is not scalable for the real world unless they are designed that way. However, more research is required to ensure that the intention of the designer is rightly articulated into the ML component through well-designed cost and reward functions.

(2) Lack of system modeling approaches for data-driven ML components:

Formally modeling complex deep neural networks with millions of data, several layers, stochastic behavior, and hundreds of features to learn poses a challenge. To model such ML-based components, an explanation based on generalization and abstraction is needed and both input and probabilistic (uncertain) output need to be formally modeled and explained. Furthermore, such uncertain output and its corresponding effect on the system-level specification needs to be formally modeled. More significant research on formalism of Markov Decision Process [102], probabilistic logics [103, 71], and counterfactual reasoning can lead to the mitigation of the problem of systematic modeling of data-driven ML component.
(3) Insufficient method for quantitative verification:

Apart from formal specification and modeling of ML components, training and testing data play a very important role in verification of the ML algorithm or component. The behavior of a frozen learning model (with no continuous learning strategy) may vary with a small perturbation of the test data. These adversarial perturbations pose a new challenge to verifying ML-based component and using them as a part of safety-critical systems such as autonomous cars [104, 105, 106, 107]. Moreover, the traditional boolean outcome of the verification and validation process is inadequate for ML. Quantitative requirements (data requirements, equivariant, and invariant specification, etc.) can contribute to design a quantitative verification process. Semi-autonomous systems with both machine and human controllers can be considered as hybrid systems [108, 109, 110] and therefore can follow a probabilistic process of verification. As suggested by Seshia et al. in [99], randomized formal methods to systematically generating training and testing data can be one of the options to move towards formal verification of ML algorithms. Randomized formal methods need to be improved to address the constraints on the legal input and output space. Randomness requirements can define the output distribution. More rigorous research on constrained random sampling is expected to aid the process [111]. Additionally, SMT solving can also be extended by combining with optimization problems to handle similar issues [112, 113]. An in-depth survey verification of neural networks can be found in [35].

Recent surge in research on safe deep neural network (DNN)

Safety assurance of neural networks in particular has received much attention in the last few years. Various strategies of testing have been adopted by researchers to gain the confidence that the neural network can be safely used in safety-critical systems; for instance, white box testing [114, 115], gray box testing [116], feature-guided black box testing [117], mutation testing [118, 119], concolic testing [120], etc. Researchers have also paid attention to the testing criteria suitable to ensure safe output of a DNN-based system [121, 122]. Byun and Rayadurgam proposed a manifold-based ML testing framework [123]. The
authors argue that compared to neuron coverage manifold-based coverage is a more effective measure of assurance. In the same direction, another recent study by Harel-Canada et al. argued the unsuitability of neuron coverage as a V & V metric for neural networks [124]. The assessment showed that the increase of neuron coverage may rather hinder the way of generating an effective test suite for neural networks. Falsification approach (testing against corner cases) has also gained interest among researchers as this approach can successfully analyze the situation where a system may fail [125, 126]. In case of formal verification, numerous approaches have been adopted by the researchers. For example, input-output range specification-based verification [127, 128, 129, 130], solver-based verification [131], reachability-based verification [132, 133, 134, 135], etc.

(4) Difficult to perform rigorous run-time testing to address the uncertainty:

Morton et al. proposed a method for deriving close-loop testing strategies for safety-critical systems [98]. Koopman et al. made remarkable contributions in the field of safety validation of automated vehicles [10, 13]. The authors proposed a phase-wise testing approach that not only mitigates the risks but also identifies the assumption violations and the unexpected situation at runtime [10]. Researchers have also focused recently on safety verification of autonomous systems with neural networks-based controller at the system-level in the presence of hardware faults like lidar faults etc. [136, 137]. Although this approaches can be helpful to reduce the testing load, more thoughts need to be given towards the traceability of the testing artifacts from the system-level to the component-level.

(5) Difficult to evaluate robustness against adversarial attacks on ML-based systems (safety meets security):

In the last few years, many researchers have expressed their concerns about the potential threats to the stability of ML-based systems posed by adversarial attacks. It is difficult to predict such attacks, and it is also complicated to model the response of a component or system as a whole to such attacks. Therefore, to develop a reliable system it is imperative to not only analyze the inherent
risks of ML-techniques, but also to protect them from intentional adversarial attacks. To ensure functional integrity of modern ML-based systems safety and security related knowledge should be used in combination. Szegedy et al., Biggio et al. first paid attention to adversarial examples that can easily deceive neural networks [138, 139]. If the attacker adds a small perturbation to an image with the proper calculation, a well-trained neural network (NN) can misclassify the image with surprisingly high confidence. Such unstable behavior can lead to catastrophic consequences in case of safety-critical systems that depend heavily on computer vision to take decisions. Numerous recent works have analyzed different ways to fool deep neural networks (DNN) [105, 140, 141, 142]. Bastani et al. proposed metrics to measure robustness against such adversarial examples [143]. Very recently, Naseer et al. proposed formal methods to analyze noise tolerance, training bias, and input sensitivity of neural networks [144]. Similarly, RL agents can also be heavily manipulated by malicious attacks. Huang et al. [145] have recently shown that RL-algorithms such as DWN, TRPO, A3C can be vulnerable to malicious inputs. Even a small perturbation can lure an RL agent to move to an undesirable state and take unsafe action. White and black-box attacks have been well investigated in this work. In case of white-box attacks, the attacker is assumed to have access to the policy network. Whereas, black-box attacker have only partial or no such information. The researchers show that white-box attackers are more effective than black-box attackers. It is possible to confuse an RL agent with trained policy even in real-world black-box scenario. For instance, lane following policy of an autonomous car can be deliberately altered by placing a small mark on the road surface or road signs.

Defenses against adversarial attacks:

Various defense mechanisms have been proposed to ensure robustness of ML models. We briefly consider the two most discussed defense mechanisms.

(i) Adversarial training:

This is a brute force process of generating as many adversarial examples as possible and using them to train the model in advance. Since Szegedy et al. [138] showed the existence of adversarial examples to fool DNNs, many
researchers focused on ensuring robustness of learning models from such adversarial attacks \cite{140, 146, 147}. As discussed by Hazan et al., dealing with adversarial perturbation can eventually help to optimise a ML model in a more robust way \cite{148}. One of the major challenges faced in training DNNs with adversarial examples is synthesizing a sizable number of adversarial examples. Various ways have been proposed to synthesize such examples in whitebox settings \cite{106, 107, 138, 140, 149, 150}. Another challenge is to generate physical-world adversarial examples (2D photos, adversarial patches, 3D prints, etc.). These physical or real-world adversarial examples have been demonstrated for various domains such as face recognition, image classification, speech-to-text, etc. \cite{151, 152, 153}.

(ii) Defensive distillation:

In this defense mechanism, a model is trained to provide outputs in terms of probability instead of hard labels. The distillation process reduces the gradients used to create the adversarial example. As a result, this defensive mechanism of a DNN can reduce the effectiveness of adversarial sample from 95% to less than 0.5%. Detailed analysis of the state-of-the-art defensive measures can be found in recent surveys on robust deep learning \cite{31, 32, 33}.

(7) Inadequate practice of demonstrating assurance cases for ML-based system:

Despite ML showing promising improvements in performance due to its introduction in many complex systems, it will always lack confidence of the users and society unless a proper assurance case is provided. However, unfortunately, there is currently no formal practice of producing end-to-end safety assurance cases. Palin et al. \cite{151} proposed patterns for designing safety cases covering all aspects of ISO 26262 for the automotive domain. However, these proposed reusable safety arguments did not consider the complexities that ML techniques bring. A fault-free system can still behave in an unintended way due to the intrinsic uncertainty of ML techniques. Therefore Gauerhof et al. proposed a safety assurance case for a pedestrian detection function using Graphical Structuring Notation (GSN) \cite{155}. In this approach, the risks of under-specification
(unclear tasks and environments), deductive gap (incorrect learning of features from insufficient data), and semantic gap (unclear domain concepts) have been reduced by defining corresponding arguments and evidences. This type of well-structured validation targets helps gain confidence. Matsuno et al. [156] explained how the inherent uncertainty of ML models affects the strategy and activities for safety assurance. According to these researchers, continuous argument engineering is useful in such cases to determine the weakness of a model, which can never be pre-calculated. They also developed tool support to assess and track the latest state of assurance. A similar idea was previously proposed by Denney et al. [157] and it was applied to the aviation system domain. The authors showed how safety management system needs to manage the safety cases not only during the development and deployment period but also at run-time based on real-time operational data. However, this approach did not explicitly handle the risks and uncertainty of ML models. Recently, patterns for arguments safety assurance of ML in the medical diagnosis system have been proposed as well [158, 159]. In these approaches, the researchers provided a detailed structure of assurance cases using GSN while considering medical settings and ML activities. These patterns can be widely used in other safety-critical domains, given sufficient settings or context-specific information about the domain.

**Discussion.** Despite putting rigorous efforts on V & V of ML-based systems, as argued by Brundage et al., not having a formal consensus on metric to measure general property like safety of these systems, there will be always a tension between the verifiability claims and the generality of such claims [160].

### 6. Discussion of research questions

In this section, we revisit our research questions to finally map the primary studies to answer the research questions from an engineering process perspective. Although in the previous section, we discussed the challenges and research efforts along each layer, we are going to summarize our findings from the literature review in this section. For the ease of understanding and to have a better
traceability, we first make a list of the primary studies and assign a unique ID to each of them as shown in Table 6. In this table, we also summarize the perspective, domain, objective and specific ML techniques that each of the studies focuses on. In case of a paper written for (or tested on) a specific domain, we specify the name of the domain. Otherwise, we mention it as “General”. Similarly, we mention NS (non-specific), if the study is not exclusively designed for any particular ML technique. DNN, RL, NN stand for Deep Neural Networks, Reinforcement Learning, and Neural Networks respectively.

6.1. RQ1: How can we easily comprehend the complexity and challenges involved in fostering safety of complex intelligent systems?

To visualize the complexity and challenges faced by multiple stakeholders involved in the process of engineering safe AI systems, we analyzed the primary studies based on the proposed three-layered framework depicted in Fig. 3. In Table 7, we map the primary studies to each layer of the three-layered framework. Compared to the other two layers, the Problem Definition Layer has received less attention. As this layer involves activities like analyzing the problem to be solved, understanding the domain, setting the right targets, it is imperative to focus more on this layer to avoid late realization of setting incorrect targets and objectives. As the table shows, many researchers are focusing on the validation and verification-related challenges in the third layer. While we acknowledge the absolute necessity of formal verification of ML-based systems to gain trust, we also believe having a clear idea about “what to verify”, “against which metrics to verify”, “what are the qualitative and quantitative targets” is equally important to verify and validate a system in the right way. In layer-2, more work needs to be done on modeling uncertain environment from a software engineering perspective. Most of the standards included in the review need to be updated to accommodate the new challenges that are brought in by the advent of ML algorithms.

In a nutshell, with the help of the three-layered framework, we can easily observe that a lot of areas are still open for research to strengthen a strong
<table>
<thead>
<tr>
<th>ID</th>
<th>References</th>
<th>Perspective</th>
<th>Domain</th>
<th>Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>P001</td>
<td>Ishikawa Nobukaju</td>
<td>Software engineer</td>
<td>General</td>
<td>Understanding engineering process-related challenges</td>
</tr>
<tr>
<td>P002</td>
<td>Arpteg et al. [22]</td>
<td>Software engineer</td>
<td>General</td>
<td>Specifying concepts in safety-critical domain to reduce conceptual uncertainty in ML-based components</td>
</tr>
<tr>
<td>P003</td>
<td>Arendt et al. [23]</td>
<td>Software engineer</td>
<td>General</td>
<td>Understanding engineering process-related challenges</td>
</tr>
<tr>
<td>P004</td>
<td>Rahimi et al. [24]</td>
<td>Software Engineer, Domain expert, Requirements engineer</td>
<td>General</td>
<td>Unifying the notion of explainability in case of ML-based systems</td>
</tr>
<tr>
<td>P005</td>
<td>Koli et al. [25]</td>
<td>Software Engineer, Domain expert, Requirements engineer</td>
<td>General</td>
<td>Exploring ML-specific risk factors for maintainable ML-based systems</td>
</tr>
<tr>
<td>P006</td>
<td>Scalley et al. [26]</td>
<td>Software engineer, system engineer</td>
<td>General</td>
<td>Discussing holistic approach of software development in the days of AI</td>
</tr>
<tr>
<td>P007</td>
<td>Bosch et al. [27]</td>
<td>Software engineer, AI expert, Requirements engineer</td>
<td>General</td>
<td>Setting qualitative targets for ML-based systems</td>
</tr>
<tr>
<td>P008</td>
<td>Veghisian et al. [28]</td>
<td>Data scientist, Requirements Engineer</td>
<td>General</td>
<td>Understanding RE challenges for ML</td>
</tr>
<tr>
<td>P009</td>
<td>Belani et al. [29]</td>
<td>AI expert, requirements engineer, systems engineer</td>
<td>General</td>
<td>Discussing goal-based requirements engineering methods to address intrinsic uncertainty of ML-based systems</td>
</tr>
<tr>
<td>P010</td>
<td>Ishikawa et al. [30]</td>
<td>AI expert, requirements engineer, systems engineer</td>
<td>General</td>
<td>Defining quality model, quality characteristics and measurements for ML-based systems</td>
</tr>
<tr>
<td>P011</td>
<td>Cihon Peter [31]</td>
<td>Governance</td>
<td>General</td>
<td>Specifying and testing robustness requirements of ML-based components</td>
</tr>
<tr>
<td>P012</td>
<td>Hu et al. [32]</td>
<td>Governance</td>
<td>General</td>
<td>To refine international standards as tools of AI policy</td>
</tr>
<tr>
<td>P013</td>
<td>Google [33]</td>
<td>Governance</td>
<td>General</td>
<td>Discussing concrete issues on AI governance</td>
</tr>
<tr>
<td>ID</td>
<td>References</td>
<td>Perspective</td>
<td>Domain</td>
<td>ML-technique</td>
</tr>
<tr>
<td>----</td>
<td>------------</td>
<td>-------------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>P016</td>
<td>Ozlati Shabnam and Roman Yampolskiy. [77]</td>
<td>System engineer, AI expert</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P017</td>
<td>Luo et al. [78]</td>
<td>System engineer, safety engineer</td>
<td>Unmanned automated systems</td>
<td>NS</td>
</tr>
<tr>
<td>P018</td>
<td>ISO 3691-4:2020</td>
<td>NA</td>
<td>Driverless industrial truck</td>
<td>NS</td>
</tr>
<tr>
<td>P019</td>
<td>ISO 13482:2014</td>
<td>NA</td>
<td>Robots and robotic devices</td>
<td>NS</td>
</tr>
<tr>
<td>P020</td>
<td>ISO 19014-1:2018</td>
<td>NA</td>
<td>Earth-moving machinery</td>
<td>NS</td>
</tr>
<tr>
<td>P021</td>
<td>ISO 17757:2017</td>
<td>NA</td>
<td>Earth-moving machinery and mining</td>
<td>NS</td>
</tr>
<tr>
<td>P022</td>
<td>ISO 18497:2018</td>
<td>NA</td>
<td>Highly automated agricultural machines</td>
<td>NS</td>
</tr>
<tr>
<td>P023</td>
<td>IEC 62267:2009</td>
<td>NA</td>
<td>Railway applications</td>
<td>NS</td>
</tr>
<tr>
<td>P024</td>
<td>ISO/PAS 21448:2019</td>
<td>NA</td>
<td>Road vehicles</td>
<td>NS</td>
</tr>
<tr>
<td>P025</td>
<td>Koopman et al. [81]</td>
<td>System engineer, safety engineer</td>
<td>Fully autonomous vehicle</td>
<td>NS</td>
</tr>
<tr>
<td>P026</td>
<td>ISO/IEC PAS 21448:2019</td>
<td>AI experts</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P027</td>
<td>ISO/IEC CD 23053</td>
<td>AI experts</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P028</td>
<td>ISO/IEC AWI 23894</td>
<td>AI experts</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P029</td>
<td>ISO/IEC AWI TR 24027</td>
<td>AI experts</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P030</td>
<td>ISO/IEC AWI TR 24027</td>
<td>AI experts</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>ID</td>
<td>References</td>
<td>Perspective</td>
<td>ML-technique</td>
<td>Domain</td>
</tr>
<tr>
<td>------</td>
<td>--------------------------</td>
<td>----------------------</td>
<td>--------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>P031</td>
<td>ISO/IEC CD 24026-105</td>
<td>AI experts</td>
<td>General</td>
<td>Road vehicles</td>
</tr>
<tr>
<td>P032</td>
<td>ISO/IEC CD 24029-1</td>
<td>AI experts</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P033</td>
<td>Salay et al. [80]</td>
<td>AI experts</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P034</td>
<td>ISO AWI 38507</td>
<td>System engineer,</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P035</td>
<td>UL 4000 [54]</td>
<td>System engineer,</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P036</td>
<td>Bach et al. [72]</td>
<td>System engineer</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P037</td>
<td>Chen et al. [64]</td>
<td>System engineer</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P038</td>
<td>Gu, Xiaozhe, and Arvind Easwaran [63]</td>
<td>ML experts</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P039</td>
<td>Chow et al. [66]</td>
<td>ML experts</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P040</td>
<td>Rong, Jihun, and Nan Luan [65]</td>
<td>ML experts</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P041</td>
<td>K. R. Vasudevan [61]</td>
<td>ML experts</td>
<td>General</td>
<td>Autonomous car</td>
</tr>
<tr>
<td>P042</td>
<td>Lee et al. [52]</td>
<td>ML expert</td>
<td>General</td>
<td>General</td>
</tr>
<tr>
<td>P043</td>
<td>Kuper et al. [52]</td>
<td>ML expert</td>
<td>General</td>
<td>General</td>
</tr>
<tr>
<td>ID</td>
<td>References</td>
<td>Perspective</td>
<td>Domain</td>
<td>ML-technique</td>
</tr>
<tr>
<td>----</td>
<td>------------</td>
<td>-------------</td>
<td>--------</td>
<td>--------------</td>
</tr>
<tr>
<td>P046</td>
<td>Naseer et al. [144]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P047</td>
<td>Sun et al. [131]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P048</td>
<td>Harel et al. [124]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P049</td>
<td>Byun, T., and Rayadurgam, S. [123]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P050</td>
<td>Fazlyab et al. [132]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P051</td>
<td>Ivanov et al. [136]</td>
<td>ML expert, system engineer</td>
<td>Autonomous car</td>
<td>DNN</td>
</tr>
<tr>
<td>P052</td>
<td>Hoang et al. [137]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P053</td>
<td>Koopman et al. [10]</td>
<td>System engineer</td>
<td>Autonomous car</td>
<td>NS</td>
</tr>
<tr>
<td>P054</td>
<td>Seshia et al. [99]</td>
<td>ML expert</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P055</td>
<td>Ghosh et al. [108]</td>
<td>Control system engineer</td>
<td>Autonomous driving, aircraft controller</td>
<td>NS</td>
</tr>
<tr>
<td>P056</td>
<td>Li et al. [69]</td>
<td>Control system engineer</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P057</td>
<td>Nilim et al. [102]</td>
<td>Control system engineer</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P058</td>
<td>Fawzi et al. [104]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P059</td>
<td>Seshia et al. [106]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P061</td>
<td>Dutta et al. [127]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P062</td>
<td>Ehlers, R. [128]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P063</td>
<td>Huang et al. [129]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P064</td>
<td>Katz et al. [130]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P065</td>
<td>Nguyen et al. [105]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>ID</td>
<td>References</td>
<td>Perspective</td>
<td>Domain</td>
<td>ML-technique</td>
</tr>
<tr>
<td>------</td>
<td>----------------------</td>
<td>-------------------------------</td>
<td>-----------------</td>
<td>--------------</td>
</tr>
<tr>
<td>P066</td>
<td>Dezfooli et al. [106]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P067</td>
<td>Goodfellow et al. [107]</td>
<td>ML expert</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P068</td>
<td>Chakraborty et al. [111]</td>
<td>Control system engineer</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P069</td>
<td>Yasser et al. [112]</td>
<td>Control system engineer</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P070</td>
<td>Tuncali et al. [113]</td>
<td>Control system engineer</td>
<td>General</td>
<td>DNN</td>
</tr>
<tr>
<td>P071</td>
<td>Pei et al. [115]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P072</td>
<td>Tian et al. [116]</td>
<td>ML expert</td>
<td>Autonomous car</td>
<td>NN</td>
</tr>
<tr>
<td>P073</td>
<td>Wicker et al. [117]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P074</td>
<td>Lei et al. [121]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P075</td>
<td>Sun et al. [122]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P076</td>
<td>Wang et al. [118]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P077</td>
<td>Ma et al. [119]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P078</td>
<td>Sun et al. [120]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P079</td>
<td>Dreossi et al. [125]</td>
<td>ML expert, system engineer</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P080</td>
<td>Tuncali et al. [126]</td>
<td>Control system engineer</td>
<td>Autonomous vehicle</td>
<td>NN</td>
</tr>
<tr>
<td>P081</td>
<td>Xiang, W., Johnson, T. T. [133]</td>
<td>Control system engineer</td>
<td>Control systems</td>
<td>NN</td>
</tr>
<tr>
<td>P082</td>
<td>Ivanov et al. [134]</td>
<td>Control system engineer, ML expert</td>
<td>Control systems</td>
<td>NS</td>
</tr>
<tr>
<td>P083</td>
<td>Akintunde et al. [135]</td>
<td>ML expert, system engineer</td>
<td>Hybrid systems</td>
<td>NS</td>
</tr>
<tr>
<td>P084</td>
<td>Dennis et al. [109]</td>
<td>ML expert</td>
<td>Agent-based systems</td>
<td>NS</td>
</tr>
<tr>
<td>P085</td>
<td>Rajeev Alur [110]</td>
<td>System engineer, ML expert</td>
<td>Robotics systems</td>
<td>NS</td>
</tr>
<tr>
<td>P086</td>
<td>Szegedy et al. [138]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P087</td>
<td>Biggio et al. [139]</td>
<td>ML expert</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P088</td>
<td>Papernot et al. [140]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>ID</td>
<td>References</td>
<td>Perspective</td>
<td>Domain</td>
<td>ML-technique</td>
</tr>
<tr>
<td>-----</td>
<td>------------------</td>
<td>-------------------</td>
<td>-------------------</td>
<td>--------------</td>
</tr>
<tr>
<td>P089</td>
<td>Bastani et al. [143]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P090</td>
<td>Grosse et al. [141]</td>
<td>ML expert</td>
<td>Malware classification system</td>
<td>NN</td>
</tr>
<tr>
<td>P091</td>
<td>Gu, S., Rigazio, L. [146]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P092</td>
<td>Hazan et al. [148]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P093</td>
<td>Papernot et al. [142]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P094</td>
<td>Huang et al. [145]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P095</td>
<td>Shaham et al. [147]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P096</td>
<td>Carlini et al. [149]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P097</td>
<td>Chen et al. [150]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P098</td>
<td>kurakin et al. [151]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P099</td>
<td>Carlini et al. [152]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P100</td>
<td>Eykholt et al. [153]</td>
<td>ML expert</td>
<td>General</td>
<td>NN</td>
</tr>
<tr>
<td>P101</td>
<td>Palin et al. [154]</td>
<td>Safety engineer</td>
<td>Automotive</td>
<td>NS</td>
</tr>
<tr>
<td>P102</td>
<td>Gauerhof et al. [155]</td>
<td>ML expert, systems engineer</td>
<td>Autonomous car</td>
<td>NS</td>
</tr>
<tr>
<td>P103</td>
<td>Matsuno et al. [156]</td>
<td>ML expert</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P104</td>
<td>Denney et al. [157]</td>
<td>ML expert</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P105</td>
<td>Picardi et al. [158]</td>
<td>ML expert</td>
<td>Medical diagnosis system</td>
<td>DNN</td>
</tr>
<tr>
<td>P106</td>
<td>Picardi et al. [159]</td>
<td>ML expert</td>
<td>Medical diagnosis system</td>
<td>DNN</td>
</tr>
<tr>
<td>P107</td>
<td>Morton et al. [160]</td>
<td>Systems engineer</td>
<td>General</td>
<td>NA</td>
</tr>
<tr>
<td>P108</td>
<td>Leike et al. [161]</td>
<td>ML expert</td>
<td>General</td>
<td>RL</td>
</tr>
<tr>
<td>P109</td>
<td>Lwakatare et al. [162]</td>
<td>Software engineers</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P110</td>
<td>Varshney, K. R. [163]</td>
<td>ML expert, system engineer</td>
<td>General</td>
<td>NS</td>
</tr>
<tr>
<td>P111</td>
<td>Sadigh et al. [71]</td>
<td>Control system engineer</td>
<td>autonomous control systems</td>
<td>NS</td>
</tr>
<tr>
<td>P112</td>
<td>Sadigh et al. [72]</td>
<td>Control system engineer</td>
<td>autonomous control systems</td>
<td>NS</td>
</tr>
<tr>
<td>Layer</td>
<td>Challenges</td>
<td>Primary studies addressing the challenges</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------------</td>
<td>------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Layer-1: Problem Definition Layer</td>
<td>1. Understanding the problem domain.</td>
<td>P001, P004, P010</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Setting quantitative targets</td>
<td>P009</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Setting qualitative targets</td>
<td>P005, P008, P012, P013</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4. Requirements modeling techniques to address uncertainty</td>
<td>P011</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5. Explaining black box to users</td>
<td>P035, P039</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6. Declaring hidden customers</td>
<td>P002, P006, P010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Layer-2: Safety-driven modeling and analysis layer</td>
<td>1. Safety-driven system-level design</td>
<td>P036, P039, P040, P041, P042, P043</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Modeling uncertain environment</td>
<td>P056, P111, P112</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. Scenario-based safe design and development</td>
<td>P037, P038</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Layer-3: Verified safety compliance layer</td>
<td>1. Formal method to verify the system</td>
<td>P059, P135</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. Safety-modeling approaches for ML-based components</td>
<td>P059, P135</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4. Rigorous run-time testing</td>
<td>P053, P107, P051, P052</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5. Robustness against adversarial attacks</td>
<td>P086, P087, P088, P089, P090, P091, P092, P092, P093, P094, P095, P096, P097, P098, P099, P100, P046, P065, P066, P067</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6. Demonstrating assurance cases</td>
<td>P101, P102, P103, P104, P105, P106</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
foundation (layer-1 and 2) of the overall system engineering of complex ML-based systems.

6.2. RQ2: How have safety concerns been addressed by the researchers along the phases of SE process?

Summary of the state-of-the-art safety approaches at the system-level and ML-based component level along the phases of software engineering life-cycle is shown in Table 8 and Fig. 4. We did not map [22, 45, 46, 80, 161, 162] as these papers do not exclusively focus on any of the particular SE activities. Instead, these studies focus on an overall engineering process of developing ML-based systems. The reason why no primary study could be directly mapped to the development phase for ML-based components is that in case of ML, the line between the design and development is very thin. The key concept here is to take safe design decisions while training the ML model. For example, safe exploration, safe policies, safety-aware rewards, etc. All these approaches are mapped to the design phase of ML-based component, rather than its development phase. All the safety-related standards are mapped to system-level development as the standards usually provide guidelines to the overall development process of the safety-critical systems. As shown in Fig. 5, there has been a recent surge in the research on verification and validation of safety-critical systems with ML-based components since 2016. However, more effort needs to be paid to conceptualize and analyze the rich problem space of AI systems in the early phases such as requirements engineering and design. It is very important to have a systematic start of the engineering process as it often plays a key role in successful product development. Moreover, not enough attention is paid to the seamless maintenance and evolution of complex AI systems as it can be seen in both Fig. 4 and 5.

6.3. RQ3: Research Gap Analysis

The most significant concern is that two different communities or engineering disciplines are responsible for ensuring safety of intelligent systems. Safety as
<table>
<thead>
<tr>
<th>Level</th>
<th>SE Phases</th>
<th>Design</th>
<th>Development</th>
<th>V &amp; V</th>
<th>Maintenance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Requirements Engineering</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ML component level</td>
<td>P004, P005, P008, P009, P012, P013</td>
<td>P039, P040, P041, P042, P043, P044, P056, P067, P091</td>
<td>-</td>
<td>P044, P045, P046, P047, P048, P049, P050, P052, P057, P058, P059, P061, P062, P063, P064, P065, P066, P067, P069, P070, P073, P074, P075, P076, P077, P078, P086, P087, P088, P089, P090, P091, P092, P093, P095, P097, P098, P099, P100, P101, P102, P103, P105, P106</td>
<td>-</td>
</tr>
</tbody>
</table>
Figure 4: Number of primary studies along the phases of SE activities at system-level and ML component level

Figure 5: Trend of research on safety approaches from an engineering perspective in the last 15 years
argued by many researchers is an emergent property of a system. The source of hazard at a system level could be control flaws, inadequate control action, and inadequate control execution, etc. Artifacts and metrics for the safety analysis at the system level, such as safety goal model, safety argument, and safety integrity level, etc. have already been identified. However, when intelligence is introduced in the system through an ML component mainly designed by AI experts, the identified source of hazards are different in terms of complexity and level of abstraction. An ML component can perform incorrectly because of the wrong choice of training data (supervised learning) or wrong choice of policy, unsafe exploration, delayed rewards, etc. (unsupervised learning). There is no common baseline of the metrics and artifacts to ensure or explain the safety constraint from an ML point of view. In the same direction, Varshney discussed a much-needed future technical agenda on defining how trust, reliability, robustness, etc. are traced to safety [162]. There is a significant gap between the two levels of safety analysis at the system level and the component level whenever any intelligence is introduced by using ML component (Fig. 6). The open research questions are:

(1) How to bridge the safety analysis gap from a safety viewpoint?

(2) How to enhance traceability of these artifacts from the system level down to the component level?

(3) What is the formal taxonomy for safety analysis in the case of ML components?

(4) Is there any standard specially designed to assess the level of safety and the acceptable range of uncertainty of ML-based components in complex AI systems?

The gap between research and practice: Koopman explained that there is a significant contrast between the safety principles in research and practice in automotive industry [163]. He argued that although ISO 26262 looks promising as a standard for the level of safety of automotive systems, uncertainty in the operating environment still poses some threat to complete deployment. Salay made a notable contribution to update ISO 26262 to accommodate technologies such
Figure 6: Gap analysis of the state-of-the-art safety approaches of complex intelligent systems as ML [80]. Lawakate et al. conducted an empirical investigation to deduce a taxonomy of SE challenges for several domains that use ML-based components heavily [161]. The authors emphasized that the challenges of a seamless engineering process that includes the development and evolution of ML-based components are very significant.

6.4. RQ4: Future Scope:
6.4.1. Addressing multi-disciplinary challenge: Collective Intelligence

The recent accidents on the autonomous systems were mostly linked to insufficient training (inadequate dataset) or choosing the wrong level of automation. The root cause was insufficient coordination between ML experts and safety engineers at an early phase of systems engineering. Requirement specification should describe data requirements, values consensus (from multiple participants), the reaction of a system to a fault, etc. This is a multidisciplinary challenge where collective intelligence can be a great asset. To ensure safety, many different domain areas need to be coordinated as shown in the framework. Data scientists, HCI experts, safety engineers, software engineers, etc. all need to collaborate to ensure safety. Instead of relying on the knowledge of individuals, we should use the collection as an emergent intelligence to solve engineering
problems. We can rely on all to work together, collaborate, and share individual knowledge to achieve a goal. In a nutshell, we can have a paradigm shift from traditional safety engineering to a broader concept of collective knowledge engineering.

6.4.2. Enhanced traceability of the artifacts across the layers of the integrated framework

Due to the lack of an integrated framework, it is unclear how we can trace safety-driven design of AI systems. Most of the research work focusing on safety-driven ML do not specify clearly how their design/learning decisions relate to the system level safety-related concepts. Traditional understanding of forward and backward traceability has less impact in the days of AI. Therefore, we define horizontal and vertical traceability of safety-driven design of AI systems.

Horizontal Traceability: traces of safety engineering process and design decisions along the same layer at system-level and ML-based component-level involving multiple participants.

Vertical Traceability: traces of safety analysis across the layers.

Instead of relying on the tacit knowledge of the experts, we recommend keeping a record of all design decisions and the rationales behind them. Therefore, we need to specify ML related categories of risks, the way those risks are adding on to the system-level risks. Thereafter, the experts can mitigate each risk with safety constraints and record them in the artifacts.

7. Threats to validity

In this section we discuss the known threats to the validity of our literature survey. We also explain how we mitigated the threats. Moreover, we summarize the limitation of our study. As discussed in [164], identified threats to four types of validity (internal, external, construct and conclusion) can be mapped to the following phases of the literature survey.

1. Planning phase:
In this phase, we set the valid research questions which were evaluated later by the graduate researchers working on AI systems engineering. We documented the rationale behind each of the research questions. We followed a stepwise process for the search method with explicit actions and outcomes. The complete list of venues or databases and search strings are mentioned in Section 4.3 and 4.2, so that the SLR can be replicated in future. The finalized list of primary studies after applying the inclusion and exclusion criteria are rechecked at least twice before moving to the next phase.

2. Conducting phase:

In order to mitigate the threat of incomplete research information or inaccessible full version of papers, we have contacted the relevant authors whenever needed. We mitigated the publication bias by excluding the grey literature from the study. During this phase, each of the paper has been cross-checked for their completeness. Along with the title and abstract, the introduction of each paper was carefully read to confirm that the perspective the work is not misunderstood. The quality of the papers is evaluated quantitatively to mitigate the threat of subjective quality assessment. Threat of duplication was mitigated by carefully choosing the full paper version (if available) over a smaller and earlier version of the same work.

3. Reporting phase:

In this phase, we report our analysis based on the research questions after analyzing 112 primary studies. These many studies are good enough to mitigate the threat of low generalizability of primary studies. We have covered safety-related papers written from various perspectives (software engineer, ML experts, safety engineer, etc) and applied to a diverse domain, to increase the generalizability of our final report.

Limitations:

Our study is primarily based on the proposed three-layered conceptual framework to engineer AI systems. This framework is designed based on our knowledge on the relevant area of research. It helps us to analyze the state-of-the-art safety approaches from the perspective of multiple stakeholders along each phase.
of the system engineering. However, an end-to-end evaluation of the effectiveness of the proposed framework is out of the scope of this paper. The survey covers primary studies mainly from the software engineering process perspective. However, there are other relevant studies on AI safety can be found from the discipline of statistics, human-computer interaction, etc. Those studies are not discussed in this literature review.

8. Conclusion

The use of ML techniques to impart intelligence creates many challenges to ensuring safety. In this paper, we summarized the current state-of-the-art research contributions in this area. We explained a three-layered conceptual framework that can help visualize the stakeholders and their contributions to engineering a complex intelligent system from a safety perspective. We analyzed the gap in the current research that should be addressed. We also described how this three-layered framework can help enhance the traceability of safety-driven design across all the layers in the future. We believe that it is worth exploring how the artifacts move across the layers in different forms to facilitate the safety analysis. Instead of relying on tacit design knowledge (or tacit machine learning knowledge of AI experts), it is helpful to document all the rationales behind each decision to eventually gain the trust of the stakeholders. For future work, we plan to design a proper stepwise methodology to guide multiple disciplines to work together for safety analysis and verification of AI systems.

Acknowledgement. This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (NRF-2020R1F1A1075605).

References


56


on Software Engineering: Software Engineering in Practice (ICSE-SEIP),
IEEE, 2019, pp. 291–300.

V. Chaudhary, M. Young, J.-F. Crespo, D. Dennison, Hidden technical
debt in machine learning systems, in: Advances in neural information
processing systems, 2015, pp. 2503–2511.

[48] J. Bosch, H. H. Olsson, I. Crnkovic, It takes three to tango: Requirement,

[49] H. Belani, M. Vukovic, Ž. Car, Requirements engineering challenges in
building ai-based complex systems, in: 2019 IEEE 27th International Re-
quirements Engineering Conference Workshops (REW), IEEE, 2019, pp.
252–255.

[50] M. Rahimi, J. L. Guo, S. Kokaly, M. Chechik, Toward requirements spec-
ification for machine-learned components, in: 2019 IEEE 27th Interna-
tional Requirements Engineering Conference Workshops (REW), IEEE,
2019, pp. 241–244.

[51] A. Vogelsang, M. Borg, Requirements engineering for machine learning:
Perspectives from data scientists, in: 2019 IEEE 27th International Re-
quirements Engineering Conference Workshops (REW), IEEE, 2019, pp.
245–251.

Explainability as a non-functional requirement, in: 2019 IEEE 27th In-
ternational Requirements Engineering Conference (RE), IEEE, 2019, pp.
363–368.

[53] J. Horkoff, Non-functional requirements for machine learning: Challenges
and new directions, in: 2019 IEEE 27th International Requirements En-
gineering Conference (RE), IEEE, 2019, pp. 386–391.


[61] B. D. Owens, M. S. Herring, N. Dulac, N. G. Leveson, M. D. Ingham, K. A. Weiss, Application of a safety-driven design methodology to an


