Approach and Guiding Principles for Developing AI/ML Components and their Standards

Dr. Natasha Neogi
NASA Langley Research Center
FAA Technical Exchange Meeting on ML
March 5th, 2024

Outline

- Motivation and Scope: Why are we here?
- Definition(s): When I say AI/ML, I mean…
- Question(s): What constitutes sufficient evidence that an AI/ML component meets its requirements?
- A way forward

Motivation and Scope: Why are we here?

Problem and Goal

Problem

- The inability to establish appropriate assurance for AI/ML components leaves us unable to effectively manage their risks and benefits.
  - Drives cost of development uneconomically high
  - Delays adoption of AI/ML at scale in safety critical systems
  - Results in unknown and unmanageable risks

Goal

- Discover and define what constitutes sufficient scientific-based evidence to substantiate a safety claim related to an AI/ML component performing a safety-critical function.

Courtesy of Baron Maddock, CC BY 4.0, https://commons.wikimedia.org/w/index.php?curid=114048157

Elevators carry two billion passengers a day over hundreds of millions of vertical miles in over 200 nations.

Courtesy of Unknown author - Copie de gravure ancienne, Public Domain, https://commons.wikimedia.org/w/index.php?curid=30135037

AI/ML will not see widespread adoption in safety-critical aviation systems until it is properly assured.

Towards standards for AI/ML…

- Standards require a stakeholder consensus on a driving need, commitment to support development, and subsequent application.
  - Broad sector of stakeholders should be involved, or uptake will suffer
  - SME contributions from all relevant or impacted stakeholders (e.g. aircraft OEMs, avionics OEMs, pilots, controllers, researchers, regulators, etc.)
- Standards development activities require a keen and deep understanding of the problem being solved and the technologies deployed in its reference implementation.
  - Understand mechanisms and limits of fundamental, underlying science of implementation and verification technologies
- NASA's Role is to:
  - Provide SME(s) to help create the standard;
  - Provide relevant findings from NASA R&D activities;
  - Learn of gaps or challenges that need to be addressed, then start up (or change) R&D activities to help fill these gaps or address the challenges.

But wait! AI/ML is already here…

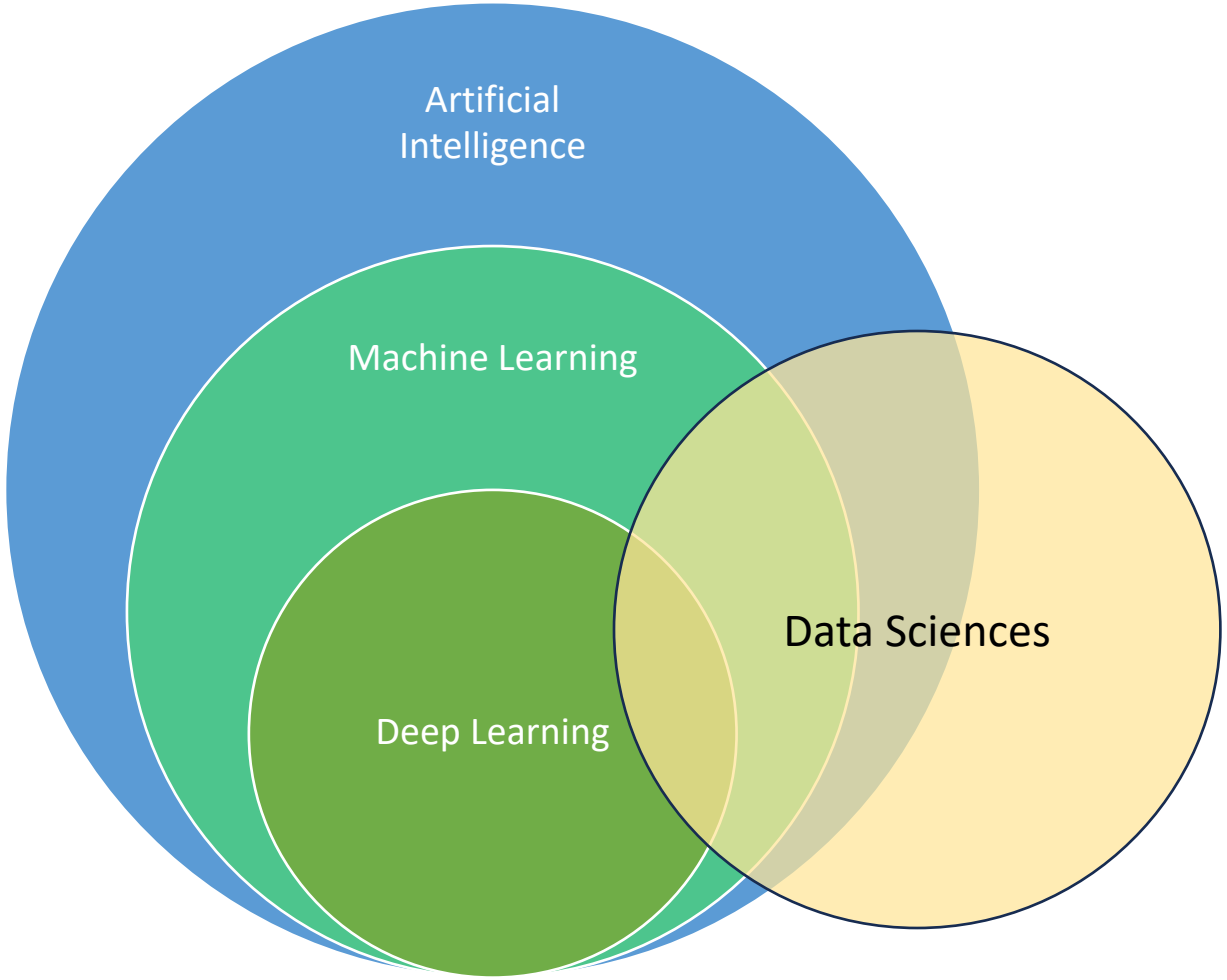
- Use of optimization tools during design of Boeing 787
  - Wing-Body Design
  - Composite Material Design
- Initial efforts at NASA to deploy AI/ML techniques in aviation contexts
  - Using Large Language Models to look for positive contributions to safety in ASRS reports
  - Anomaly Detection/Vulnerability Discovery
- and others…

Courtesy of Timo Breidenstein – http://www.airliners.net/photo/United-Airlines/ Boeing-787-822-Dreamliner/2142634/L/, GFDL 1.2, https://commons.wikimedia.org/w/index.php?curid=20544763

Definitions:
When I say AI/ML, I mean…

Definitions: AI and ML

- AI: Intelligence displayed or simulated by technological means, [where intelligence is defined] by the standard of human intelligence, the sort of intelligence that humans display.
  - "AI Ethics" by Mark Coeckelbergh, MIT Press
- ML: Use of statistical techniques to analyze data and create algorithms that can generalize to unseen data without explicit programming
  - https://gradml.mit.edu/intro/



When I say AI, I mean…

Machine Learning Systems	• Reinforcement Learning, Supervised Learning, Unsupervised Learning, Generative Systems…
Rule Based Learning Systems	• Production Systems, Expert Systems, Fuzzy Logic,…
Search and Optimization Techniques	• Uninformed search, Informed Search, Parallel Search,…
Decision Making under Uncertainty	• Bayesian Inference, Parameter Learning, Structured Learning,…
Evolutionary Strategy	• Evolutionary Algorithms, Swarm Based Algorithms,…
⋮	⋮

Question(s):
What constitutes sufficient evidence that an ML component meets its requirements?

The three “E’s”

- Explicit Claims
  - Required emergent properties must follow from the combination of the properties of the system component (that is, ML component implementation) and the domain assumptions (context); environmental assumptions (including interfaces); and constraints.
  - They should indicate explicitly the level of assurance claimed.
- Evidence
  - Concrete evidence is usually a combination of testing, analysis (including modelling and simulation), and appeals to process.
    - e.g., software deployed in the field is the same as software under test/analysis
- Expertise
  - Developers should be familiar with best practices and deviate from them only when needed.
  - Experts can wisely tailor their approach to assuring novel elements with respect to methods, languages, tools, and processes.

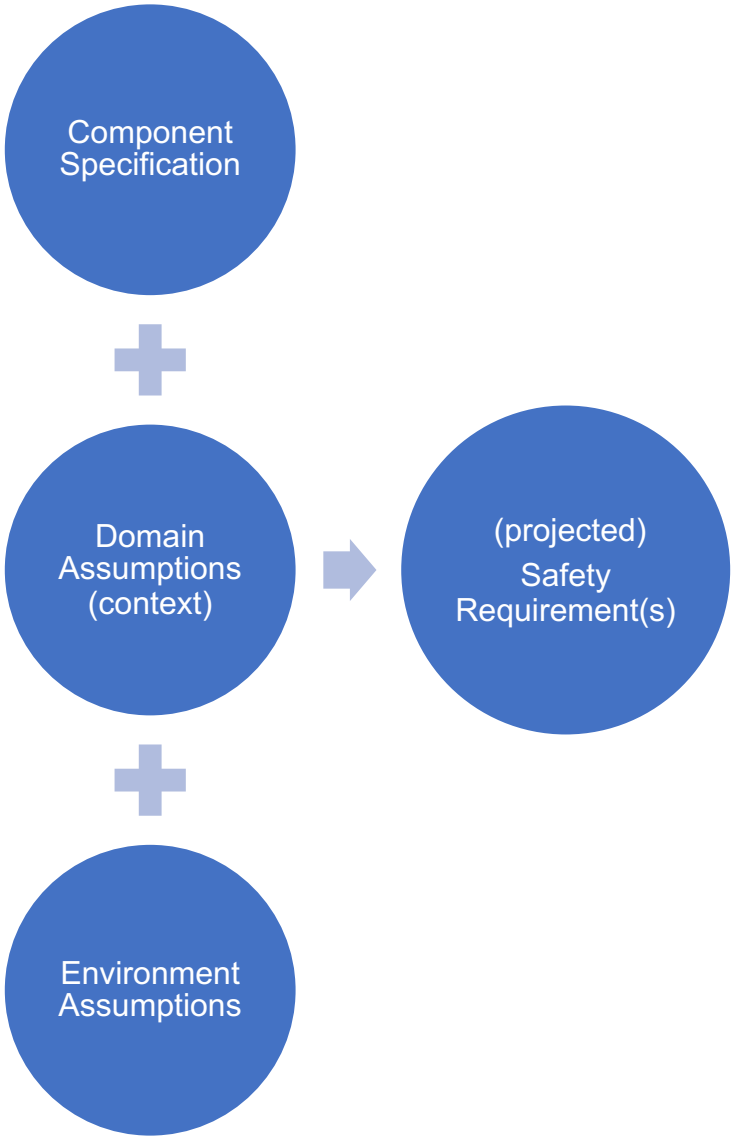
Adapted from: National Research Council. 2007. *Software for Dependable Systems: Sufficient Evidence?*. Washington, DC: The National Academies Press. https://doi.org/10.17226/11923.

Explicit Claims: ML as a component of a system

- Safety is not an intrinsic property of an ML component.
  - An ML component may be safe in the context of one system but not in the context of another.
- The specification of an ML component characterizes the behavior of the ML software at its interface with other system components and the environment.
  - It is important to distinguish this specification from the desired emergent (safety) properties of the system in the physical world.
- If a ML component only meets its assurance criteria if humans interacting with the system behave in a certain way, then this becomes an assumption on the environment of the component (or a constraint of system components) that must be evaluated for validity.

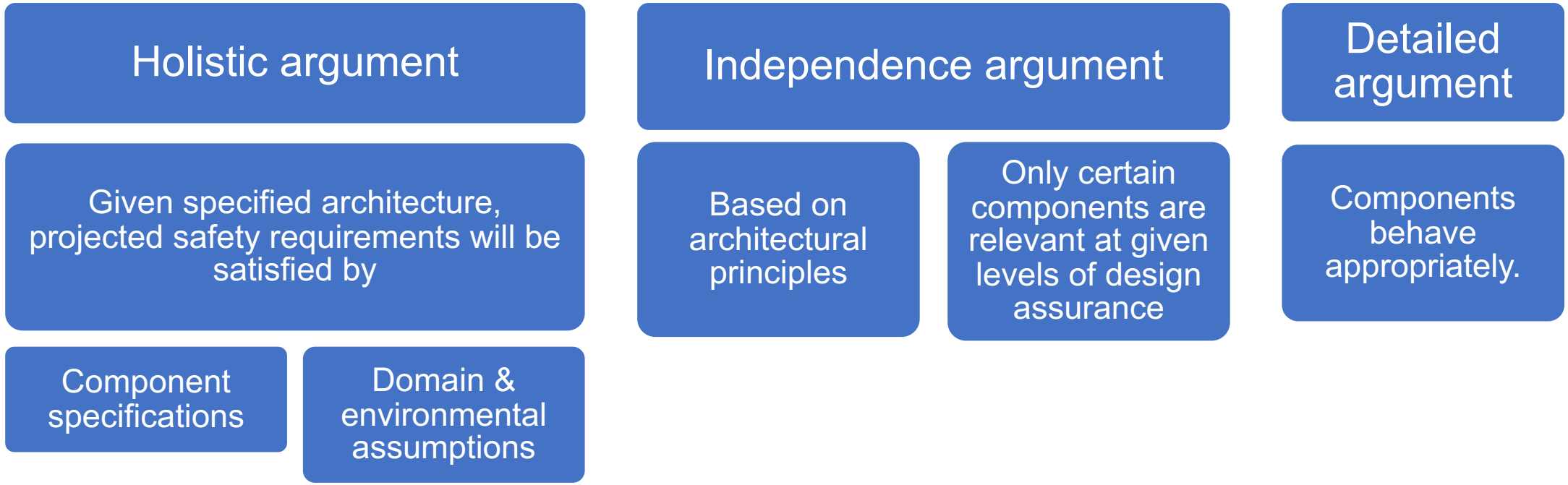
Explicit Claims: The role of assumptions

- Domain Assumptions (context): An explicit statement of domain assumptions (context) of the ML component is required to evaluate any safety argument associated with that component.
  - This requires an argument that the specification of the ML component and the domain assumptions together imply the projection of the desired emergent property on the component.
  - Must perform all three activities: (1) check the ML component, (2) check the domain assumptions, and (3) check that they have the correct combined effect.
- Environmental Assumptions: An explicit statement of environmental assumptions (often physical parameters outside the scope of design authority) is needed to evaluate any safety argument associated with a component.
  - These assumptions must be validated at design time and during operations.



Explicit Claims: The role of architecture

- The case that a system with an ML component with architectural mitigations satisfies a safety claim (or requirement) may proceed as follows.



- Note that restrictions related to architectures and relief from required design assurance levels specified in current standards must be considered in such an argument.

Evidence: Why traditional techniques do not translate for ML

	Traditional (Physical) Components	ML (Software) Components
Criteria	Criteria are simple (e.g., failure/breakage rate etc.) for the component as a whole.	Complexity of ML and its interdependence on its domain and environment make it difficult to have explicit and precise articulation of meaningful criteria that can be measured.
Feasibility of Testing	For physical artifacts, limited testing provides compelling evidence of quality, with the continuity of physical phenomena allowing widespread inferences to be drawn from only a few sample points.	Limited testing of ML cannot provide compelling evidence of behavior under all conditions.
Process & Product Correlation	Underlying principle of statistical quality control is that sampling the product coming out of a process gives a measure of the quality of the process itself, which in turn will determine the quality of items that are not sampled.	More rigorous ML design processes will likely lead to better quality ML components. However, this correlation is not sufficient as the sole provider of evidence, as correlation does not imply causation.

Adapted from: National Research Council. 2007. *Software for Dependable Systems: Sufficient Evidence?*. Washington, DC: The National Academies Press. https://doi.org/10.17226/11923.

Evidence: Testing, simulation and analysis, and formal methods

- Testing for ML components is indispensable.
  - However, testing alone is insufficient, as it is unclear what coverage means in terms of ML components.
- Simulation and analysis can provide needed checks for ML components.

Validation of environmental assumptions, interface assumptions, and constraints	Feasibility or satisfiability analysis of temporal behaviors	Verification of code implementation against component specifications	Checking that components in aggregate achieve appropriate system-level effects

  - However, simulation and analysis is insufficient due to model inaccuracy, incorrect assumptions (e.g., environmental, operator response, execution platform), etc.
- Formal methods can provide guarantees for ML components.
  - Formal methods can provide formal proofs of correctness.
  - Formal methods techniques often lack scalability.

Expertise: Transparency and credibility of claims

- To establish that a system containing an ML component is safe will involve inspection and analysis of the safety claim and the evidence offered in its support.
  - Assurance of ML components requires explicit safety claims (or assurance requirements), evidence for those claims, and a rigorous argument that demonstrates that the evidence is sufficient to establish the validity of the claims (or satisfaction of the requirements).
- Evaluator should be able to calibrate not only the technical claims and evidence but also the organization that produced them, because the integrity of the evidence chain is vital and cannot easily be assessed without supporting data.

Qualifications of the personnel involved in the development of the ML component	Track record of the organization in providing ML components	Process by which the ML component was developed	Process by which data used to train/test the ML component is collected/curated/maintained etc.

Key Questions to Answer (I)

How do we know when an ML component's behavior meets its requirements?	• Sufficient representation and size of training dataset, accuracy vs. generalizability, what constitutes an actionable specification, etc.
What are the limits of current processes and metrics currently used in developing and evaluating both traditional and ML systems?	• How do you use testing (i.e., creating logical based oracles, etc.), simulation (i.e., model validity), (formal) analysis (i.e., scalability), runtime verification frameworks, etc. in assurance?
What are the set of characteristics and parameters of an ML system that allows you to bound its behavior (e.g., capabilities, limitations, etc.)?	• Data and information quality, architecture, associated metrics, etc.
What is the minimum set of information required to reconstruct and audit ML application performance in the case of an accident?	• State, Environmental, and Input Information, Decision Making Logic, Configuration Management, Version Control, etc. • How do you create an encoding scheme that would reduce the volume of state information into a tractable, compact form?

Key Questions to Answer (II)

How should information assurance be handled for ML components in order to yield (composable) safe systems?	• Data fusion; information synthesis; data collection, curation, and assurance; etc.
How can change be managed in ML systems in order to preserve assurance?	• Configuration management, version control, database management, etc. • Full recertification, continuous authorization to operate, etc.
When is it appropriate to use an ML implementation for a function?	• Clear (and testable) set of requirements, outputs easily checked for correctness, corrective action can easily be taken, etc.
When is it appropriate to use ML in the development and/or accident/incident analysis process?	• Tool qualification (DO-330), ASAP/ASRS database querying for research, prognostics/diagnostics, scheduling, maintenance, etc.

Key Questions to Answer (III)

What are key domain specific considerations that may dominate the safety of ML implementations and how will we address them?	• Lack of safe default mode/state, inability of pilot to intervene, etc.
What is the current human contribution to safety in the function being replaced by an ML/AI implementation (i.e., full extent of the capabilities and limitations of the human role)?	• Consider critical information dependencies across tasks executed collaboratively by diverse agents, etc.
Can the open world problem be solved (and standardized) without humans to handle edge cases while maintaining the current level of NAS safety?	• Handling epistemic uncertainty, applicability of real-world data across different environmental assumptions, etc.
Can an actionable specification for a function be extracted from a dataset?	• Functional requirements, Safety requirements, Environmental assumptions, Domain specific constraints, etc.

A way forward…

Going Forward: Deploying ML in aviation systems

- Start with ML in design/analysis/maintenance (offline system, offline learning)
  - Start with simple, well-defined, non-safety critical applications
    - Recognizing normal and anomalous patterns in large datasets for research purposes (collaborative),
    - Querying large databases for research purposes (ASRS/ASAP), etc.
- Progress to ML in embedded flight/operational systems (online system, offline learning)
  - Start with functions which have
    - Clearly defined requirements,
    - Means of checking the answer/output, and
    - Means of intervention and mitigation of incorrect answers/outputs.

Going Forward: Standards development for ML in aviation systems

- Identify current and ongoing standards efforts that may be applicable (e.g., DO-330, etc.) to ML components.
  - Leverage other standards bodies when appropriate to avoid duplicative efforts.
  - Standards efforts should be targeted at areas in which gaps are found.
- A measured approach to standards development for ML applications should target those functions for which there are actionable specifications and traditional implementation and assurance techniques.
  - Standardize criteria for what constitutes sufficient evidence for ML safety.

Takeaways

- Deployment and standardization efforts should proceed methodically and with a justifiable basis, thereby enabling safe adoption of ML applications in aviation.

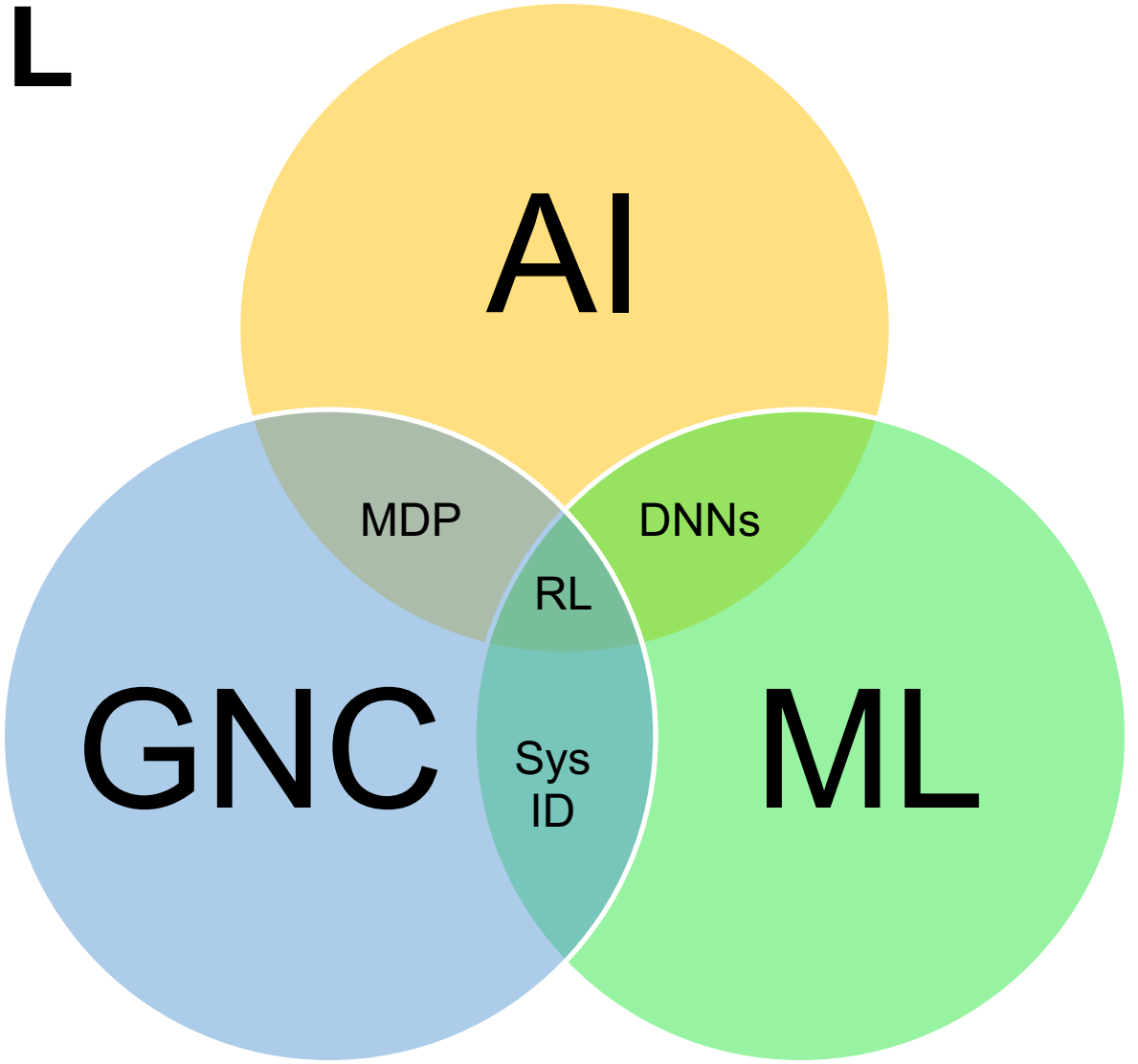
- Premature efforts to (deploy and) standardize may damage paths to transition for ML technologies, engender technical debt, or set back the entire aviation industry.

Questions?

natasha.a.neogi@nasa.gov

Definitions: AI and ML

- AI: Provide scientific understanding of the mechanisms underlying thought and intelligent behavior and their embodiment in machines (AAAI)
- ML: Use of statistical techniques to analyze data and create algorithms that can generalize to unseen data without explicit programming (MIT)



Adapted from Atkins, SciTech 2024

Explicit Claims: The role of assumptions

- Domain Assumptions (context): An explicit statement of domain assumptions (context) of the AI/ML component is required to evaluate any safety argument associated with that component.
  - This requires an argument that the specification of the AI/ML component and the domain assumptions together imply the projection of the desired emergent property on the component.
  - Must perform all three activities: (1) check the AI/ML component, (2) check the domain assumptions, and (3) check that they have the correct combined effect.
- Environmental Assumptions: An explicit statement of environmental assumptions (often physical parameters outside the scope of design authority) is needed to evaluate any safety argument associated with a component.
  - These assumptions must be validated at design time and during operations.

(Application) Domain

- circumstances, background, or setting in which something occurs or exists
- framework for understanding and interpreting information, events, or situations

Environment

- surroundings or conditions in which component exists
- physical, social, cultural, and natural factors

Expertise: Reuse, change management, and criticality creep

- As AI/ML components evolve, they will require re-evaluation to determine whether they still satisfy their safety arguments.
  - Explicit articulation of assumptions (domain, environmental, interface, etc.) is critical.
- Current scientific understanding of AI/ML components and state-of-the-art verification and validation techniques does not provide the ability to reason about system-level properties based solely on the properties of the system's components.
  - Pace of change is what distinguishes AI/ML from other software/hardware.
- As systems evolve, scope and criticality creep may require re-evaluation of the sufficiency of the target level of assurance for an AI/ML component.

Expertise: Managing assurance costs

- Only implement AI/ML when necessary.
  - The key to achieving requisite assurance at reasonable cost is simplicity, including simplicity of critical functions and simplicity in system interactions.
- Use architectural means to mitigate complexity caused by AI/ML when possible.
    - Establish independence so system level properties are guaranteed by individual components which preserve the emergent property despite failures in the rest of the system.
- Use rigorous processes to develop AI/ML.
  - Each step in developing the AI/ML software needs to preserve the chain of evidence on which will be based the argument that the resulting AI/ML component meets its requirements (and the overall system is safe).

Guiding Principles

Create a culture where the design and assurance processes for AI/ML systems embrace the application of rigor, reproducibility, and reusability.

Design

- Rigor
  - Principles, methodologies and theoretical frameworks for data collection and curation, specifying requirements, prototyping, etc.

- Reproducibility
  - Repeatability, Reproducibility, and Replicability of design practices
- Reusability
  - Cannot engineer point solutions. Need diversity, extensibility, portability, etc.

Assurance

- Rigor
  - Understanding of the science underpinning the assurance of intelligent systems
- Reproducibility
  - Diverse assurance methods should not yield contradictory results
- Reusability
  - Assurance arguments must be extensible over multiple contexts and update throughout lifecycle

Focus on science underpinning AI/ML design & assurance to establish sufficiency guarantees.

Technical challenges and leadership

Challenges

- Verification and Validation of AI/ML Systems
  - Properties of Concern: Safety, Liveness, Security, Fairness…
- Human Machine Teaming Interactions
  - Role Allocation: Authority and Responsibility
- Bounding Behavior of AI/ML Functions in Uncertain Environments
  - Contingency Management
  - Fault Containment
  - Heterogeneous Vehicles
  - Mixed ConOps
- Trusted Decision Making
  - Adaptive/Non-Deterministic
  - Shifting control paradigm

Technical Leadership

- Scalable methods addressing formal verification of safety and liveness properties of AI/ML systems
- Methods for designing, assessing, and assuring safety over diverse role allocation and decision-making paradigms
  - Mathematical models for describing adaptive/nondeterministic processes as applied to humans and machines.
- Provably Correct Synthesis of Assurance Monitors
  - Formally Verified Runtime Monitors, Steering Functions
- Simulation and Testing approaches to increase confidence in safety critical decision making for AI/ML systems
- Certification Standards

Technical leadership will advance state-of-the-art and state-of-practice in AI/ML systems

Assurance barriers to fielding AI/ML components in civil aviation

- Lack of scalability of current approaches
  - DO-178C and software complexity
  - Pace of change and update rates for AI/ML components
- Lack of approaches, tools and techniques for evaluating safety properties in AI/ML components
  - Current approaches geared towards obtaining quantitatively predictable outcomes
  - Need models, methods and tools to develop high confidence in systems with
    - Shifting locus of control between humans and automation
    - Non-deterministic and/or adaptive decision making
  - Require a confluence of analytic, simulation, test and evaluation techniques
- Lack of certification standards
  - Need rigorously defined processes and procedures to establish system-level performance requirements and functionality derived from specified levels of safety
  - Cost Effectiveness, Barrier to Entry, Change Management

Research Issues for AI/ML

- Behavior of Adaptive/Nondeterministic Systems. Develop methodologies to characterize and bound the behavior of AI/ML components over their complete life cycle.
- Modeling and Simulation. Develop the theoretical basis and methodologies for using modeling and simulation to accelerate the development and maturation of advanced AI/ML components.
- Verification, Validation, and Certification. Develop standards and processes for the verification, validation, and certification of AI/ML components, and determine their implications for design.
- Nontraditional Methodologies and Technologies. Develop methodologies for accepting processes and technologies not traditionally used in civil aviation (e.g., open-source software, DevSecOps) in AI/ML systems.
- Operation Without Human Intervention. Develop the system architectures and technologies that would enable increasingly sophisticated AI/ML components to operate for extended periods of time without real-time human cognizance and control.
- Roles of Personnel and Systems. Determine how the roles of key personnel and systems, as well as related human–machine interfaces, should evolve to enable the operation of AI/ML components.