CHALMERS UNIVERSITY OF TECHNOLOGY | UNIVERSITY OF GOTHENBURG

Requirements Engineering for Machine Learning: Non-functional Requirements as Core Functions

Jennifer Horkoff

jenho@chalmers.se
Jennifer.Horkoff@gu.se

AIRE 2022

RISE OF ML (MACHINE LEARNING)

- What I really want to report: what % of software in use today uses some form of ML
  - (spoiler, I can't find this number)
- What I can find...
  - https://learn.g2.com/machine-learning-statistics#machine-learning-adoption-statistics

14%

increase in global GDP by 2030 is forecasted with the advancements of ML and AI.

Source: WSJ

- 20% of C-level executives (across 10 countries and 14 different industries) report using machine learning as a core part of their business.
- Budgets for ML programs are growing most often by 25%, and the banking, manufacturing, and IT industries have seen the most significant budget growth this year.
- 33% of IT leaders will adopt ML for improving business analytics.

91.5%

of leading businesses have ongoing investments in AI.

Source: Businesswire

RE for ML (Selected) Challenges

- Software with ML has “new” characteristics:
  - Uncertainty, non-determinism
  - Incomplete specifications
- Can our set of concepts and methods be applied?
  - As is? With adjustments? Extensions?
  - Or not? Do we need all new approaches
- Processes and methods are different
  - We have been dealing with Agile RE, now Agile RE for ML
- NFRs are (even more) important
  - But are they the same? New NFRs? New definitions? New measurements?

Agenda

- RE for ML SOA (in brief)
- NFRs for ML
- Case Study – Perception Systems in Autonomous Driving
- Summary
- Future Work

RE for AI State of the Art (Partial View)

Problem Exploration

- **[Vogelsang & Borg, 2019]** Interviewed data scientists, quantitative targets are functional requirements, need data requirements
- **[Belani et al. 2019]** Provides an RE4AI taxonomy with mapped challenges to AI-related data, models and system
- **[Horkoff 2019]** NFRs are important and must be reconsidered (more on this later)

Reviews

- **[Villamizar et al. 2021]** Mapping Study, covers 35 studies, topics, NFRs covered, paper types, evaluation types

RE for AI State of the Art (Partial View)

Solutions

- **Ontologies**
  - **[Rahimi et al. 2019]** Hard to decompose NF targets, introduced domain, dataset, model, ML component development specifications. Create domain ontology, map to dataset and ML model, finding underspecified domain concepts.
- **Robustness**
  - **[Hu et al. 2020]** Focus on safety, changes not visible to humans cause different classifications, methods to specify and test robustness requirements for ML
- **Quality**
  - **[Hamada et al. 2020]** Focuses on quality assurance for AI, evaluation techniques, domain specificity, examples
  - **[Anisetti et al. 2020]** Proposes a taxonomy of NFRs for ML, multi-armed bandit method for selecting ML model based on an NFR
  - **[Nakamichi]** Extended the quality characteristics defined by ISO25010 to those unique to ML, defining a method to derive the quality characteristics and measurement methods
- **Safety**
  - **[AMLAS]** Guidance on the Assurance of Machine Learning in Autonomous Systems
- **Modeling**
  - **[Nalchigar et al. 2019]** Matches common problems to ML solutions using a type of goal model.
  - **[Barrera et al., 2021]** Extending i* to deal with ML concepts
  - **[Ishikawa & Matsuno, 2020]** Evidence/hypotheses-based RE, importance of experimentation. Using goal models.

Sample View from outside RE

[Bosch et al., 2018] “In addition to having development teams executing on requirements specified by product management, the development of software systems is progressing towards a data driven practice where teams receive an outcome to realize and where design decisions are taken based on continuous collection and analysis of data”

Development approach	Definition
Requirement driven development	Software is built to specification. This development approach is predominantly used when new features or functionality are well understood and defined.
Outcome/data driven development	Development teams receive a quantitative target to realize and are asked to experiment with different solutions to improve the metric. Examples of this development approach are new features (used frequently by customers) and innovation efforts.
AI driven development	A company has a large data set available and use artificial intelligence techniques such as machine learning and deep learning to create components that act based on input data and that learn from previous actions. Examples of this development approach include e.g. object recognition in autonomous cars and speech recognition in modern user interfaces.

We are here

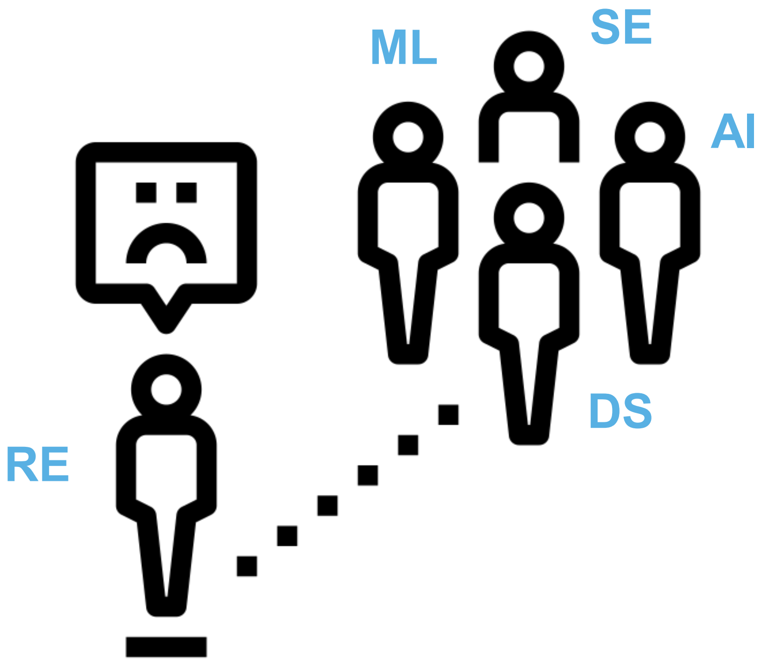
Quantitative targets are also requirements (NFRs)

There are requirements here too!

But we should also be/be seen to be here

Why?

The Future of RE?



Bully by Kamin Ginkaew from NounProject.com

Back to Basics - Why do we conduct RE?

What I tell my students	For ML-based Systems (outcome/AI-driven)
To avoid wasting time coding something that won't be successful	Some experimentation is inevitable, can we reduce this via clearer, more realistic targets?
To avoid having to make many changes	Drift is inevitable, can we use requirements to monitor and manage it?
To make the final product better	Yes! Clear quality and functional requirements
To anticipate the effects of your product/software	Yes! Same techniques?
To avoid being sued	Yes! Complex role of safety and standardizations
+ Internal/external communication	Yes! But are our current representations working?
+ Organizational memory	Yes! But what do we need to remember?

CHALMERS UNIVERSITY OF TECHNOLOGY | UNIVERSITY OF GOTHENBURG

NFRs for ML

NFRs for ML

Non-Functional Requirements for Machine Learning: Facilitating Continuous Quality Awareness (iNFoRM)

- VR Project (Swedish Research Council – Vetenskapsrådet)
- 2020-2025
- Many (interesting) questions
- Not yet so many answers

With PhD Student Khan Mohammad Habibullah (Habib)

khanmo@chalmers.se

Co-supervisor Gregory Gay

Qualities of ML Solutions

- Accuracy & Performance (Correctness)
- Fairness
- Transparency
- Security & Privacy
- Testability
- Reliability
- What else?
  - Trainability?
  - Maintainability?
  - Sustainability?

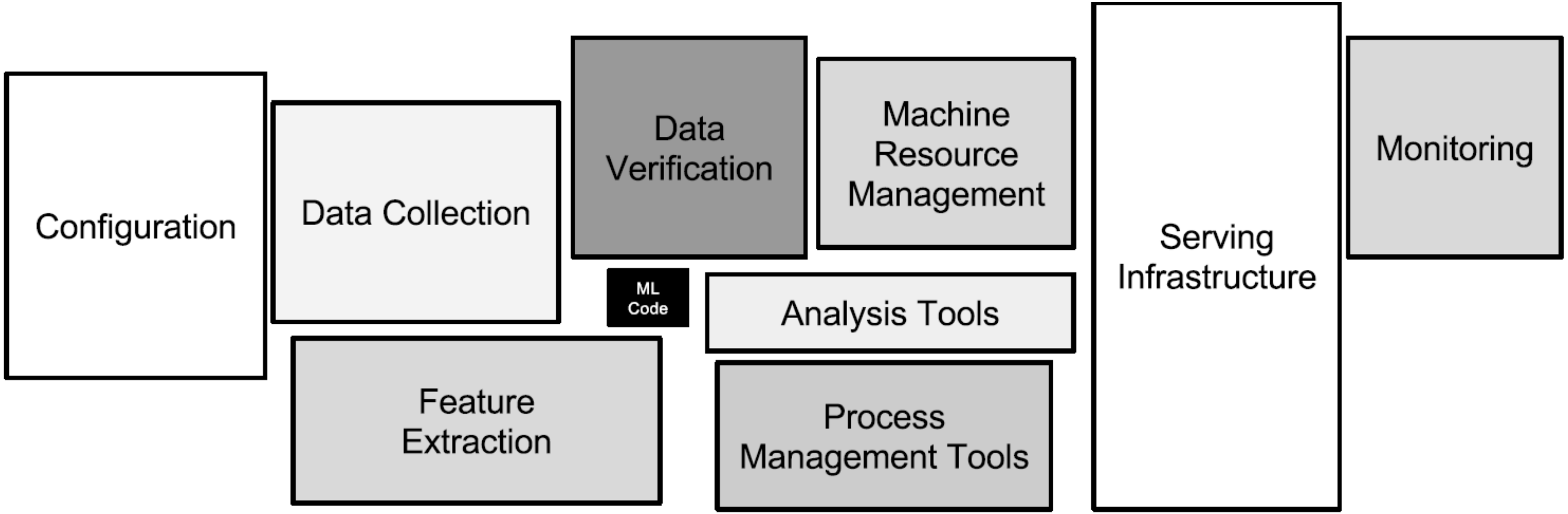
[Wan et al., 2020] “Instead of functional requirements in non-ML software systems, quantitative measures comprise the majority of requirements for ML systems.”

Project overview (thus Far)

- Problem Exploration Stage
  - Interview Study [Habibullah & Horkoff, 2021]
  - Follow-up Survey [Habibullah et al., submitted]
  - SLR [Habibullah et al., 2022] [TBD]
- Solution Stage
  - Work in progress

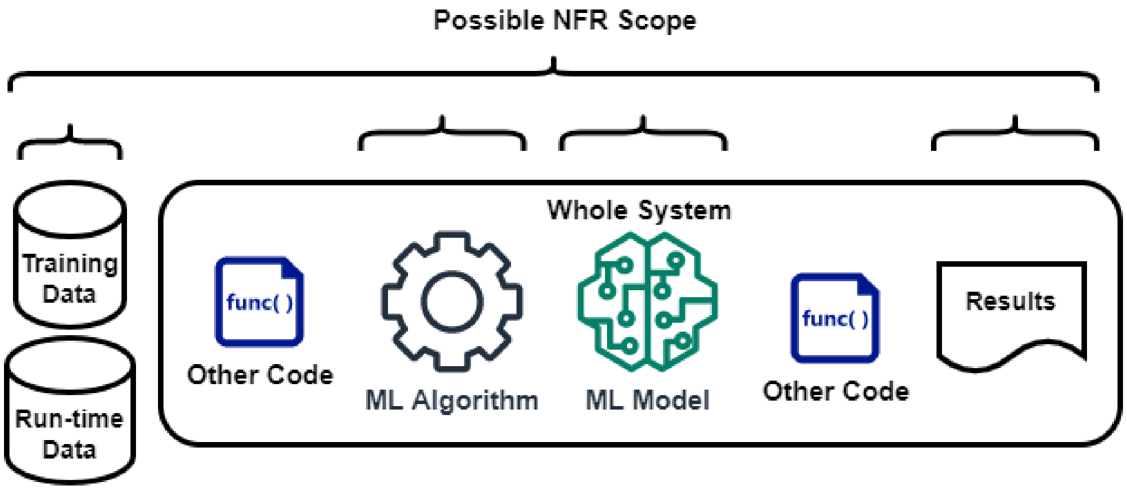
ML as Part of a Larger System

Sculley et al., 2015, NIPS



NFR Scope

Over which elements of an ML system can individual NFRs be defined?

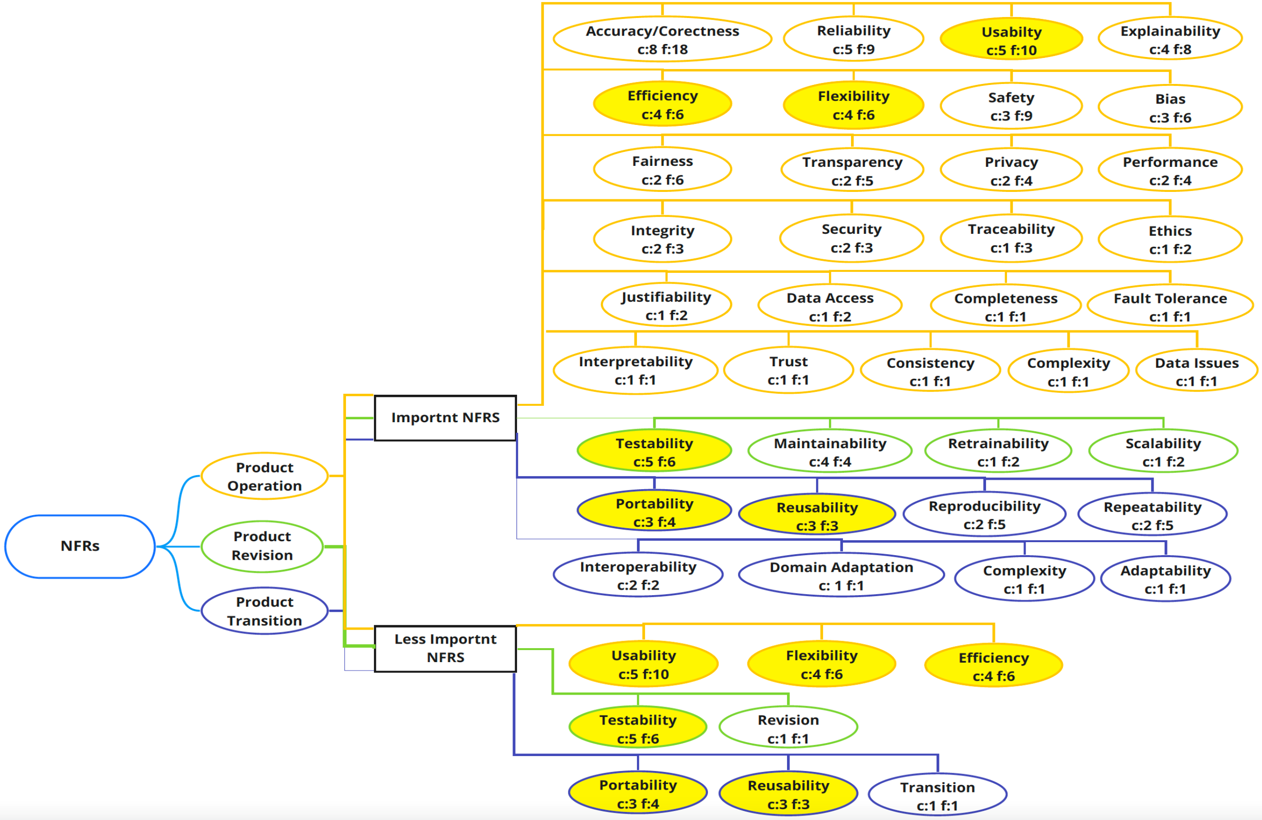


Possible scope for NFRs over system elements.

Interview Study [RE'20]

- What is the perception and current treatment of NFRs in ML in industry?
- Conduct an interview with 10 participants with industrial ML experience
- (Selected) Questions
  - Which ML-related NFRs are more or less important in industry?
  - Over what aspects of the system are NFRs defined and measured?
  - What NFR and ML-related challenges are perceived?
  - What measurement-related challenges for NFRs in an ML-context exist?

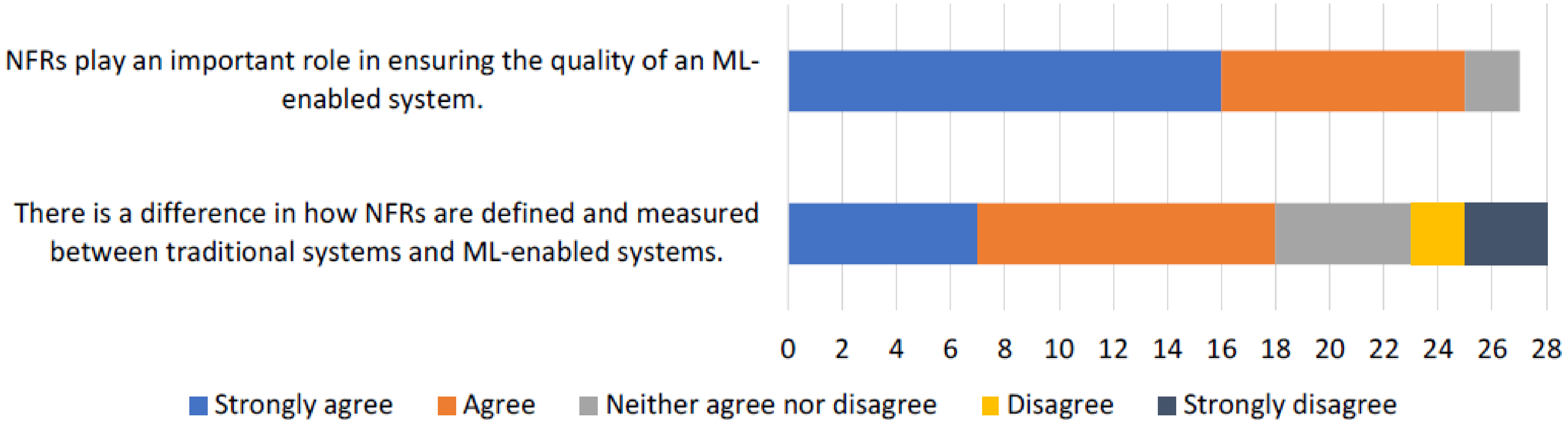
Which ML-related NFRs are more or less important in industry?



Survey Extension

- Created a survey based on our interview findings
- 42 (at least partial) responses
- Mix of industry and academic respondents
  - Able to compare findings between groups

Survey Results – General Question



Survey Results – Challenge Ranking

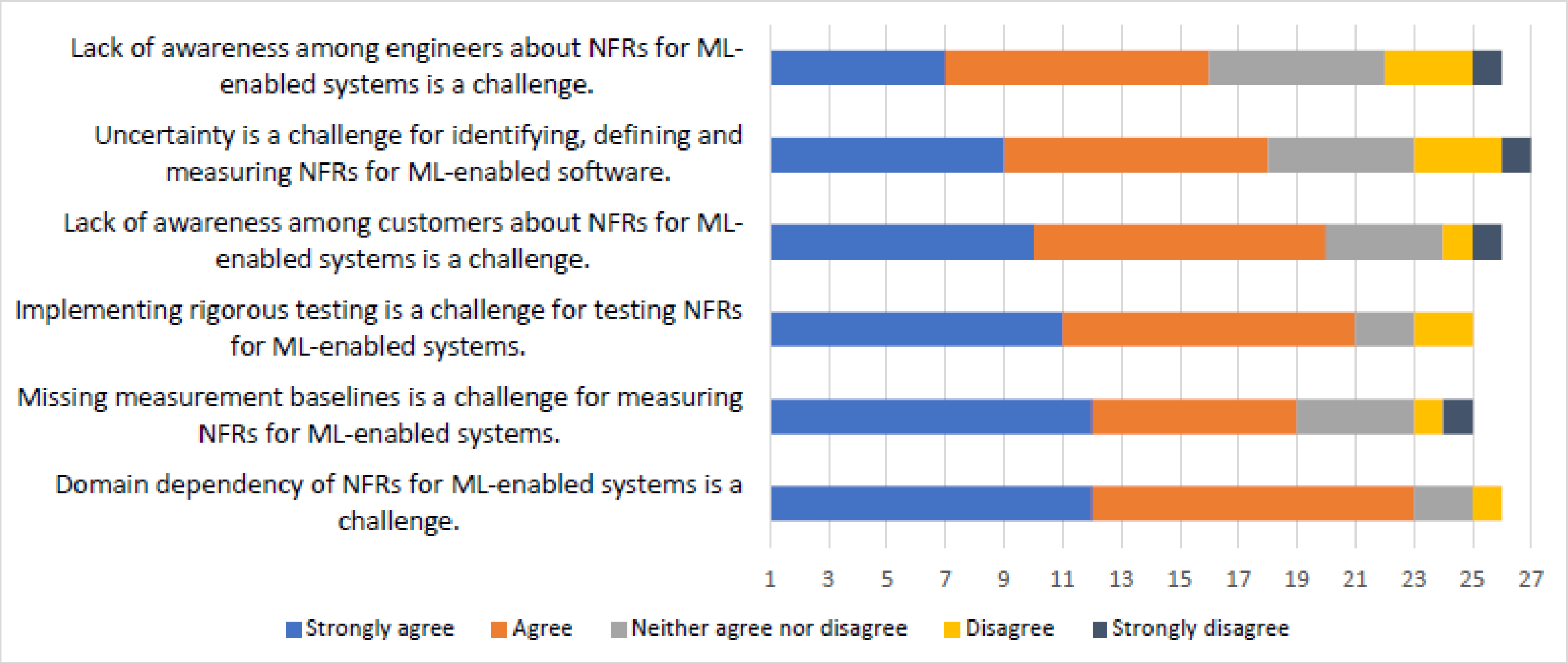


Fig. 8 NFR and NFR-measurements related challenges.

Survey Results – Challenges Defining NFRs

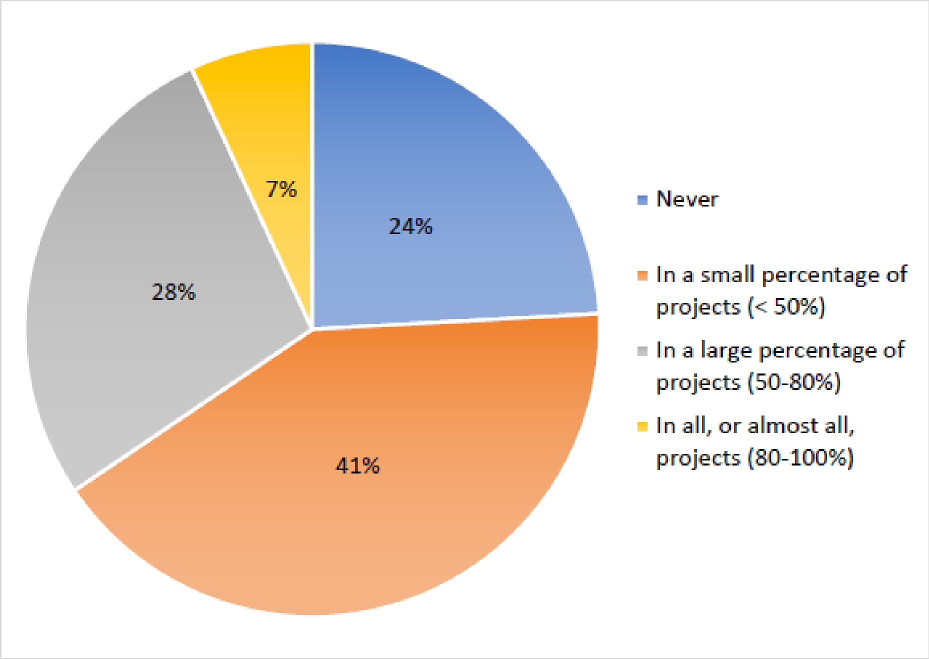
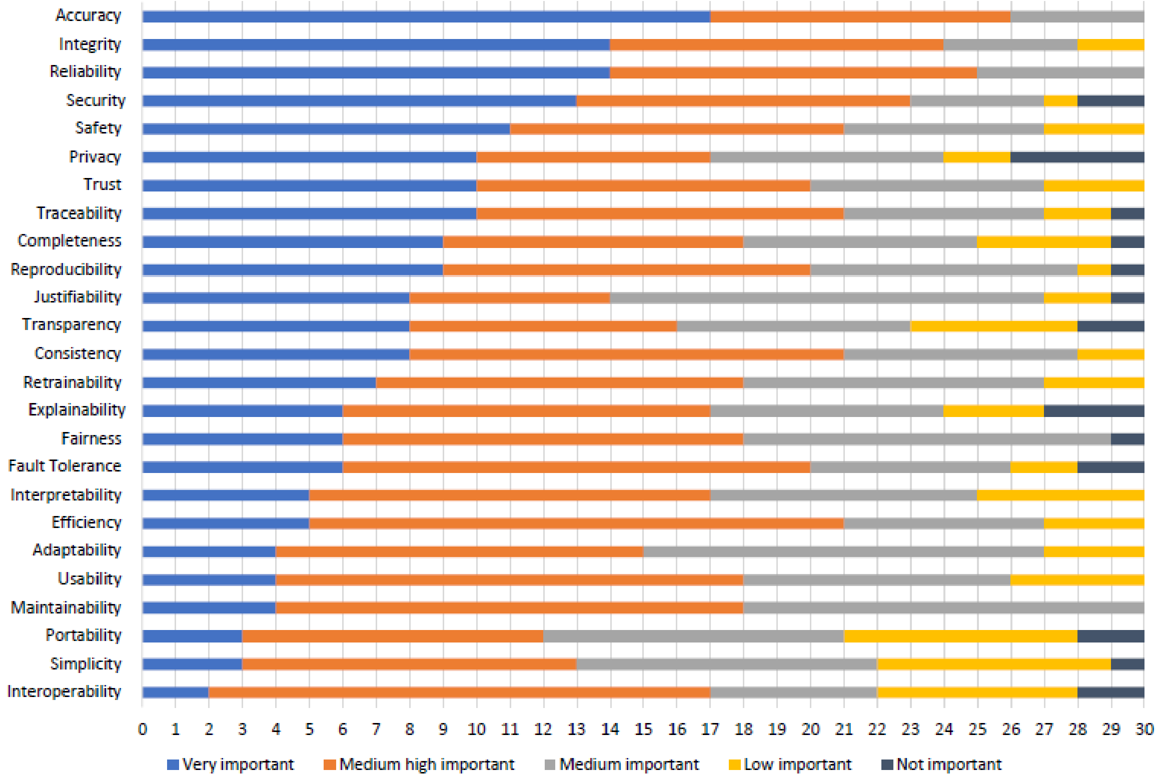


Fig. 7 How often survey participants face challenges defining NFRs for ML systems.

Survey Results – NFR for ML Ranking



Compare Different Backgrounds

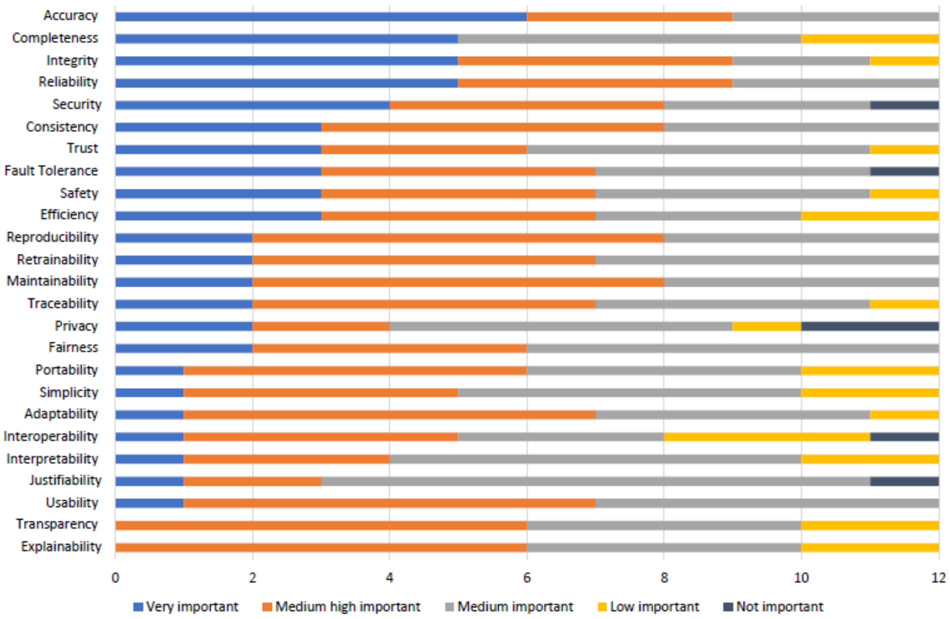


Fig. 12 The importance of NFRs, as identified by participants in **academic positions**.

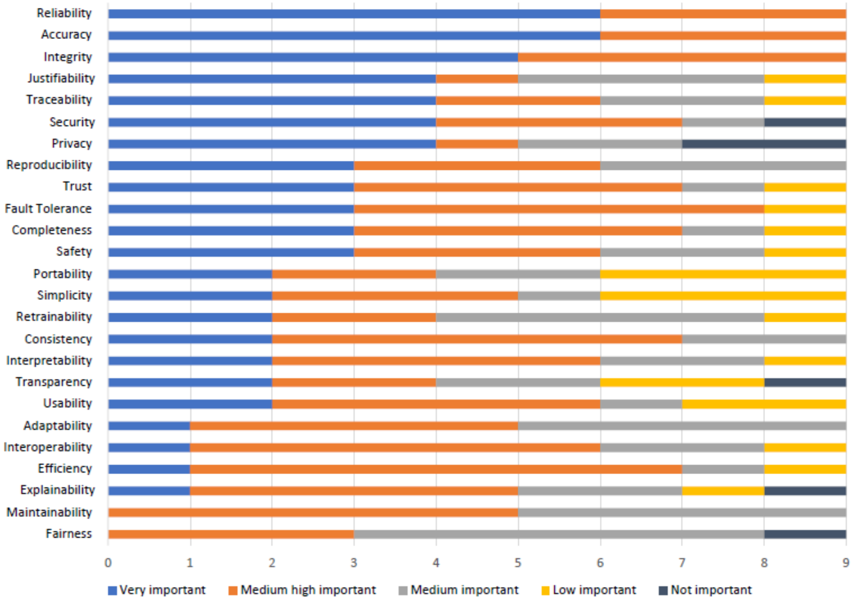


Fig. 13 The importance of NFRs, as identified by participants in **industrial positions**.

Interview and Survey Summary

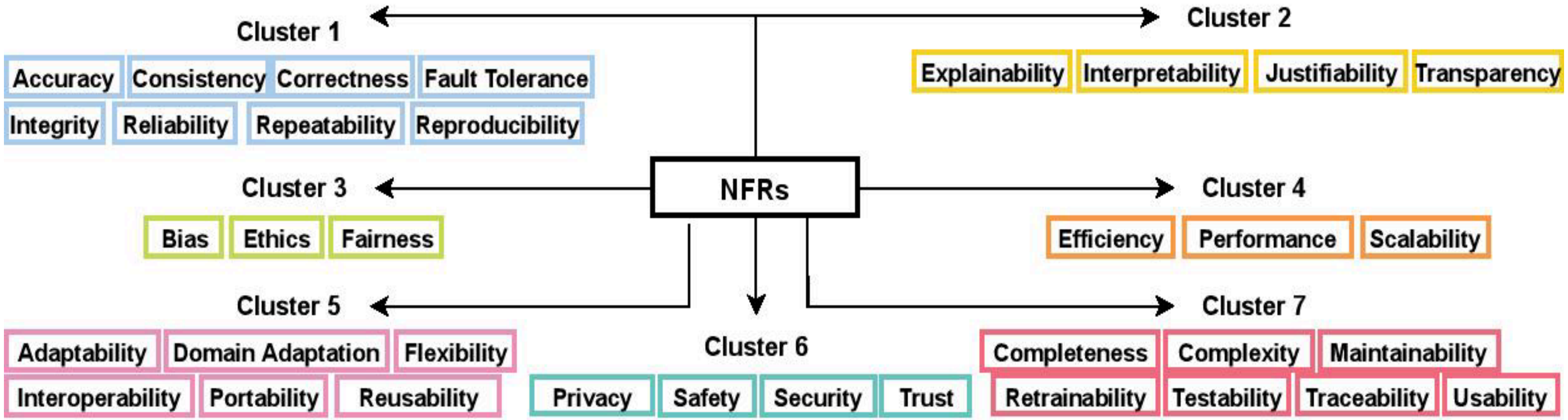
- Most participants agreed that NFRs are important in ensuring ML system quality, and that there are differences in how NFRs are defined and measured from traditional systems (e.g., adaptability, maintainability).
- Accuracy, reliability, integrity, and security are particularly important for ML systems.
- Most NFRs for traditional software are still relevant, while a few become less prominent (e.g., revision, transition).
- Perception on the importance of efficiency, fairness, flexibility, portability, reusability, testability, and usability are split among participants.
- Most practitioners focused on defining NFRs over the whole system. Several also define NFRs on models. Few have considered NFRs for data.
- NFR challenges relate to uncertainty, domain dependence, awareness, regulations, dependency among requirements, and specific NFRs (e.g., safety, transparency, and completeness).

NFRs for ML Pre-Systematic Mapping

- What is the perception and treatment of NFRs for ML in academia?
- Topic too big for one SLR
  - Which NFRs?
- Performed an initial mapping study to estimate the number of relevant papers on ML for each NFR
- Took the top 50 or 100 papers for most NFRs and classified them in/out of scope
  - Independently with 3 researchers + discussion
- Clustered NFRs

NFR CLUSTERING

Can the ML system NFRs be grouped into a small number of clusters based on shared characteristics?



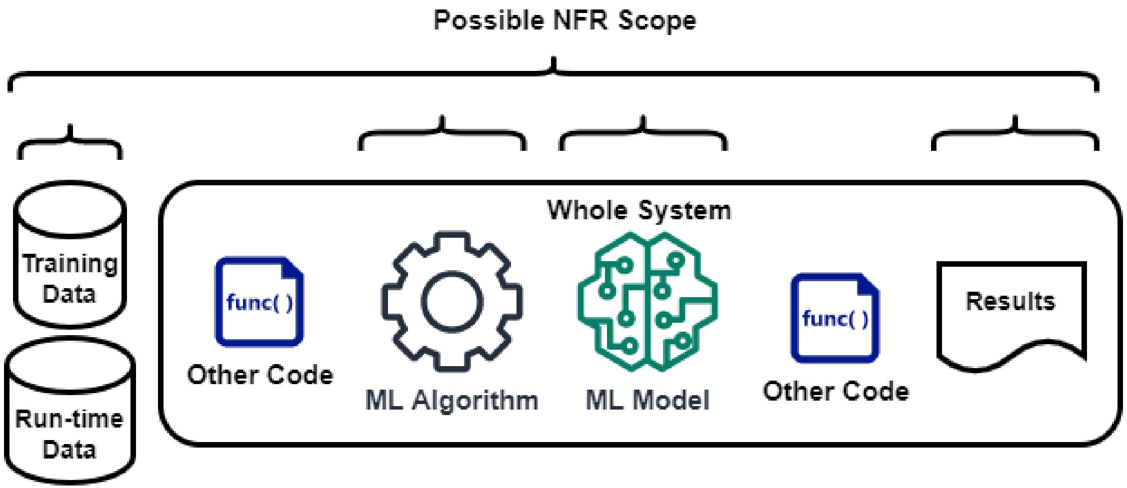
NFRS WITH NUMBER OF SEARCH RESULTS

RQ2: Which NFRs have received the most—or least—attention in existing research literature?

NFR	Cluster	Search Results	Relevant Papers (Sample 1)	Fleiss' kappa (Sample 1)	Relevant Papers (Sample 2)	Fleiss' kappa (Sample 2)	Estimated Num. Relevant Pubs.
Performance	4	114853					
Accuracy	1	92669					
Efficiency	4	22247					
Security	6	19142					
Complexity	7	16997					
Privacy	6	6388					
Safety	6	5848					
Reliability	1	5620					
Bias	3	4118					
Scalability	4	3595					
Consistency	1	2936					
Flexibility	5	2764	23 (46%)	0.54			1271
Interpretability	2	2418					
Trust	6	1965					
Reproducibility	1	1796					
Domain Adaptation	5	1732	47 (94%)	0.63			1628
Usability	7	1270	21 (42%)	0.50	29 (58%)	0.44	635
Adaptability	5	1177	34 (68%)	0.50			800
Fairness	3	1089	45 (90%)	0.41			980
Correctness	1	1045	16 (32%)	0.53			334
Integrity	1	1015					
Transparency	2	851	44 (88%)	0.70			749
Explainability	2	706	44 (88%)	0.22			621
Fault Tolerance	1	553	26 (52%)	0.68			288
Interoperability	5	532	9 (18%)	0.45			96
Completeness	7	372	23 (46%)	0.40	25 (50%)	0.58	179
Portability	5	346	21 (42%)	0.45			145
Ethics	3	331	31 (62%)	-0.03			205
Reusability	5	321	24 (48%)	0.55			154
Maintainability	7	277	6 (12%)	0.30	9 (18%)	0.72	42
Traceability	7	214	4 (8%)	0.61	6 (12%)	0.61	21
Repeatability	1	171	17 (34%)	0.44			58
Testability	7	77	4 (8%)	0.54	2 (4%)	1.00	5
Justifiability	2	3	0 (0%)	1.00			0
Retrainability	7	0					0

Reminder NFR Scope

Over which elements of an ML system can individual NFRs be defined?



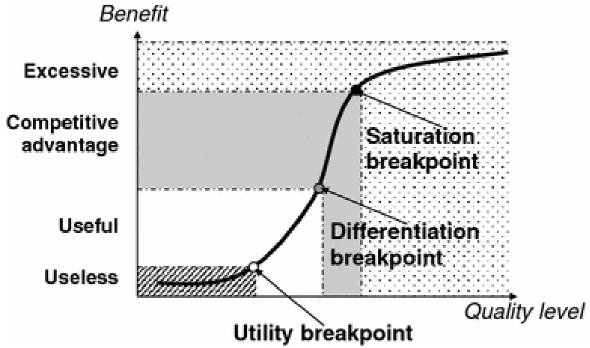
Possible scope for NFRs over system elements.

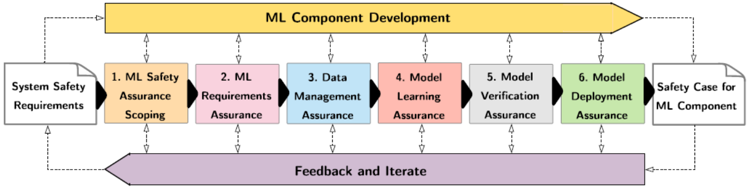
NFR SCOPE (VERY SUBJECTIVE, WORK IN PROGRESS)

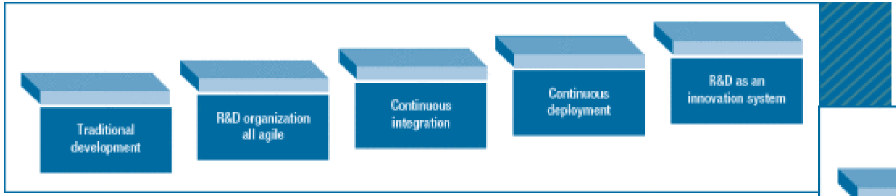
NFR	Cluster	System Element the NFR Can be Defined Over: Train. Data	Algo.	Model	Results	Whole System
Completeness	1	✔	✖	✔	✖	✔
Correctness	1	✔	✔	✔	✔	✔
Fault Tolerance	1	✖	✔	✔	✖	✔
Integrity	1	✔	✔	✔	✔	✔
Repeatability	1	✖	✖	✖	✔	✔
Explainability	2	✖	✔	✔	✔	✔
Transparency	2	✖	✔	✔	✔	✔
Ethics	3	✔	✔	✔	✔	✔
Fairness	3	✔	✔	✔	✔	✔
Adaptability	5	✔	✔	✔	✔	✔
Domain Adaptation	5	✔	✔	✔	✔	✔
Flexibility	5	✖	✔	✔	✖	✔
Interoperability	5	✖	✔	✔	✖	✔
Portability	5	✔	✔	✔	✖	✔
Reusability	5	✔	✔	✔	✖	✔
Maintainability	7	✔	✔	✔	✖	✔
Testability	7	✖	✔	✔	✔	✔
Traceability	7	✔	✔	✔	✔	✔
Usability	7	✖	✔	✔	✔	✔

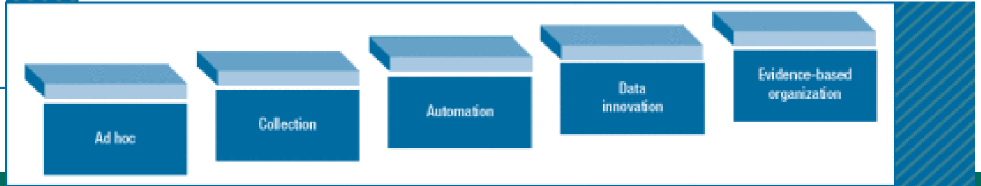
NFRs for ML – Towards solutions

- NFRs for NFRs for…
  - Lightweight, business-friendly, intuitive
    - (conflict with publishable?) ☺
  - Templates?
  - Models? Inspiration?
    - Quper? [Berntsson Svensson et al., 2012]
    - AMLAS?
    - Bosch et al. Stairways? [Bosch & Olsson, 2017]









CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Case Study – Perception Systems in Autonomous Driving

PRECOG

Case Study – Perception Systems in Autonomous Driving

Vinnova pre-study

Authors: Markus Borg, Hans-Martin Heyn, Jennifer Horkoff, Khan Mohammad Habibullah, Alessia Knauss, Eric Knauss, Polly Jing Li

RISE (Research Institute of Sweden)

Annotell AB

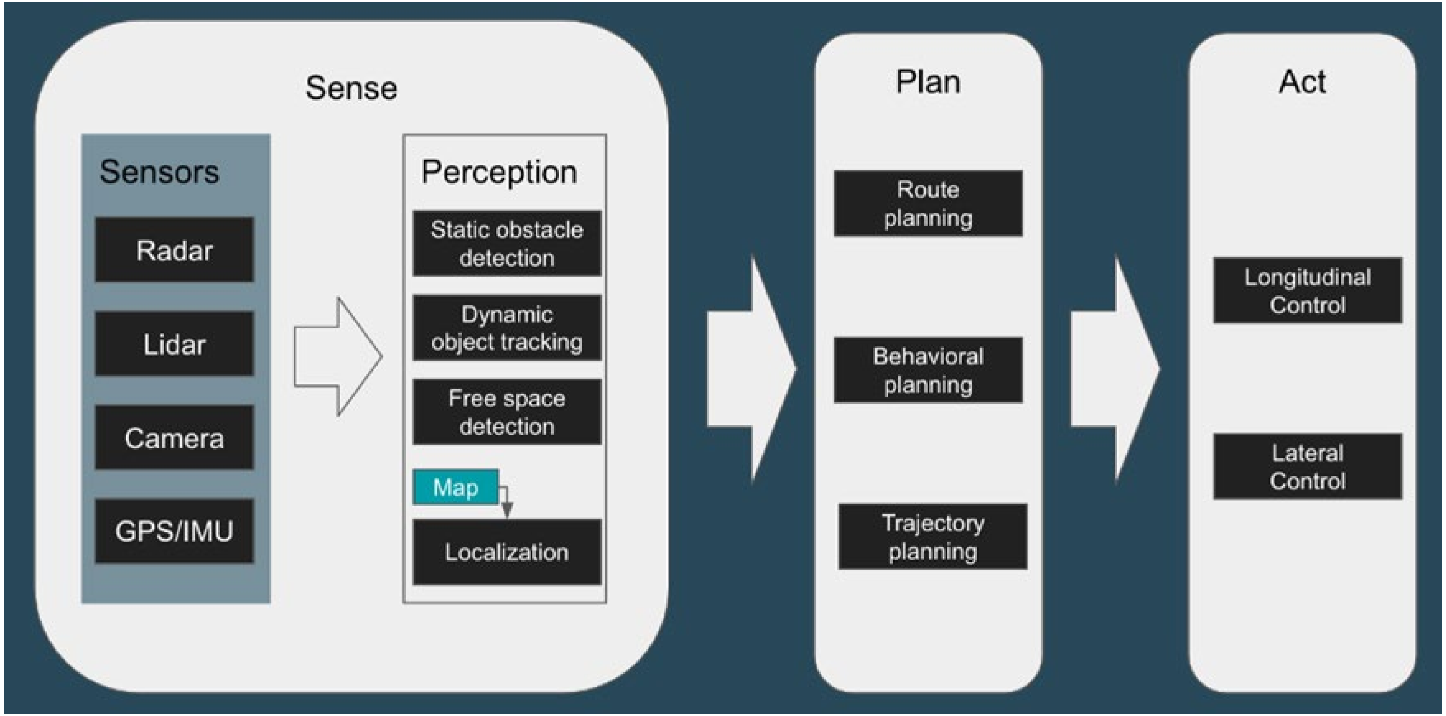
Zenseact AB

University of Gothenburg

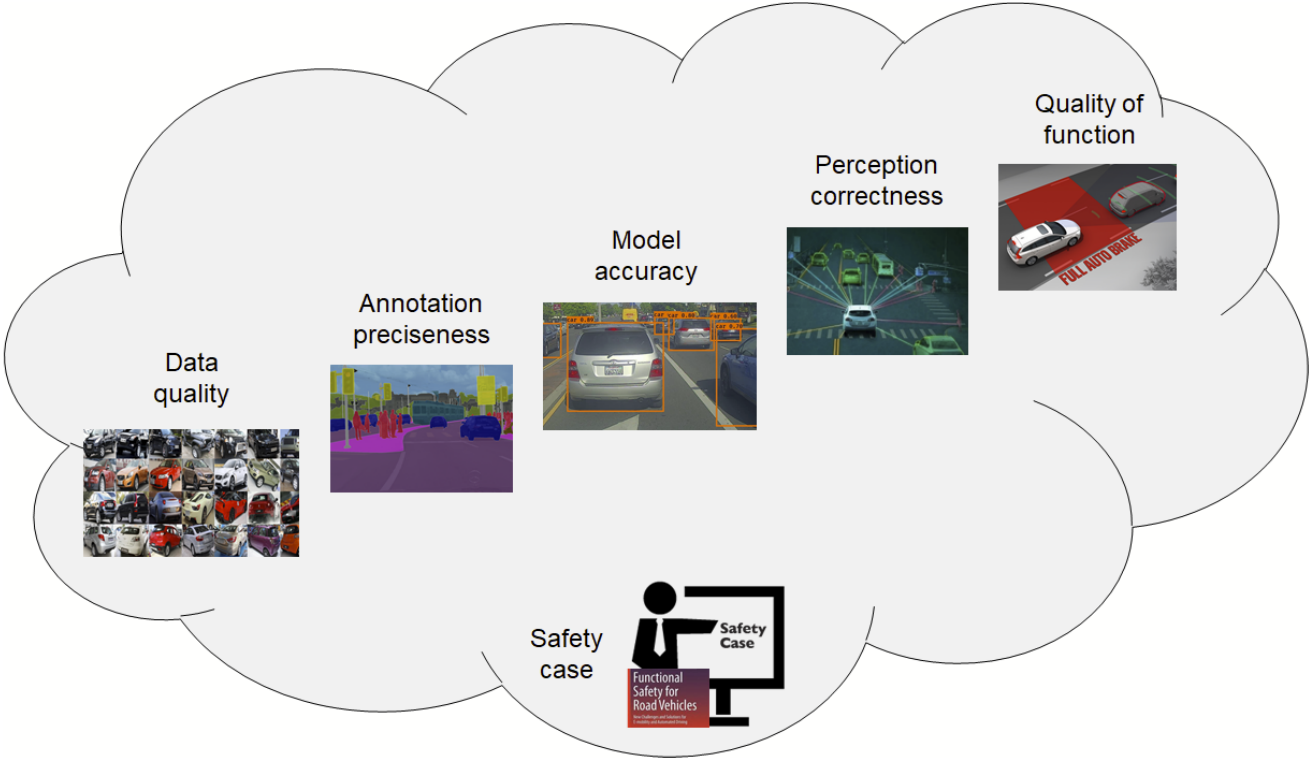
Method

- Group and individual interviews with relevant company personnel in all tiers of autonomous driving
  - purposive, convenience, and snowball sampling
  - 19 participants from 5 companies
- Group thematic coding
- Follow-up workshop with invited partners
- Writing up results

Background: perception systems



System Overview – Links between areas?



Results - Themes

In total eight major themes were identified as results of this study:

1. Data
2. Perception
3. Artificial intelligence and machine learning concerns
4. System and Software Engineering
5. Quality
6. Ecosystem and Business
7. Requirements Engineering
8. Annotation

Requirements Engineering (1/4)

- Breakdown
  - Requirements breakdown
  - Requirements allocation
- Documentation/Ways of working
  - Specification
  - Test-based specification
  - Documentation of requirements
- Scenarios
  - Scenario database
  - Edge cases
- Operational Context & Scope
  - ODD
  - Context
  - Context challenges
- Tracing & Change
  - Dependencies
  - Requirements changes
  - Traceability

Requirements Engineering (2/4)

- Ways of Working
  - They don’t follow the traditional form of requirements engineering
- Specifications
  - They don’t necessarily have a traditional requirements specification
  - No large specifications, may have a rough specification to start
  - Requirements that do exist may have unclear origins, not sure if the setter understands the impact on ML results, different spaces of knowledge
    - Sometimes requirements given are not feasible with data
  - They have a specification for data annotation
  - Have data specification and classes, data distribution, quality
  - Specifications on external hardware, sensors, still
  - Data is the requirements to some extent – defines behavior

Requirements Engineering (3/4)

- Breakdown
  - They don’t really do requirements breakdown, have rough requirements, then deal with scenarios and experimentation
  - Requirements breakdown described as “trickle-down”
- Requirements allocation
  - Dividing between components e.g., sensors, algorithms
  - Related to redundancy, hardware vs. software vs. ml
  - Requirements on data, on sensors, on function, etc.
  - Still allocation on other parties

Requirements Engineering (4/4)

- Scenarios
  - Scenarios came up often as an important way to ensure the models cover common and uncommon sequence of events
  - Edge cases were important to make sure data was present to support important cases that don't occur very often
  - Have thousands of scenarios, look at distribution based on real world
  - Scenarios drive development
- ODD is important
  - Relationship to scenario? complicated

Quality (NFRs)

- NFRs/Quality Raised in Interviews
  - Model-level: performance, correctness, accuracy, efficiency, robustness, explainability, tradeoffs
  - System-level: performance, robustness, comfort, integrity, trust, reliability
  - Function-level: performance, accuracy, suitability
  - Safety: standards, goals, case, risk, integration, redundancy

”most of our development … it’s (an) iterative process is not more about literally how to achieve a certain goal is more about how to avoid certain error”

- Tradeoffs
  - Mostly Safety vs. X
  - Doesn’t seem to be an explicit topic

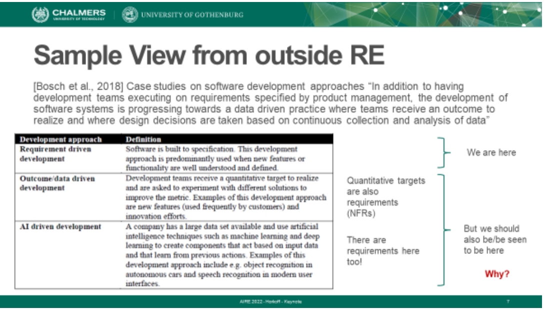
Case Study – Key Findings thus Far

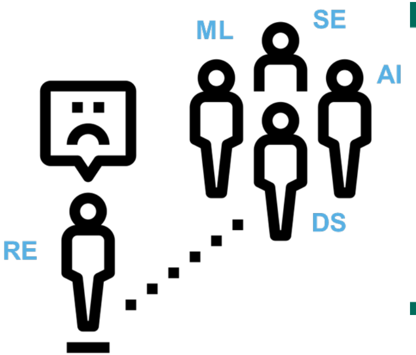
- No clear links between quality of features, models, data, annotation.
  - They are linked, but no formulas or way to quantitatively predict quality from one side to another
- Traditional RE methods are only followed to a certain point
  - Boundary of features to ML involves more negotiation, experimentation, trickle
- Safety and standard are key challenging issues
- Requirements as data
- Requirements over data, annotations
- Requirements as scenarios
- Requirements as requirements – high-level, to sensors, to components, to external parties
- Abandon notion of complete and correct requirements specification for ML sub-parts
  - But requirements can and should still play a role

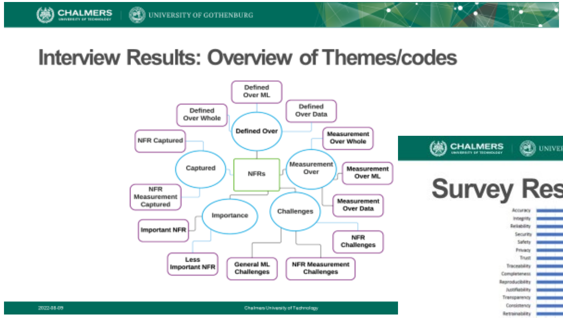
CHALMERS UNIVERSITY OF TECHNOLOGY | UNIVERSITY OF GOTHENBURG

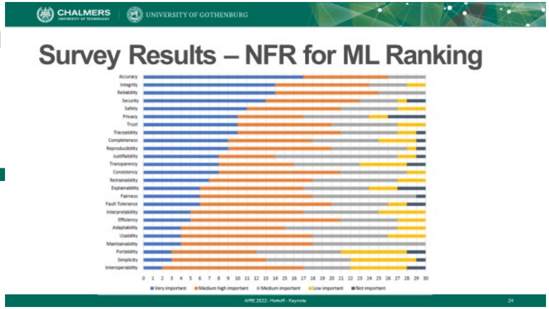
Summary

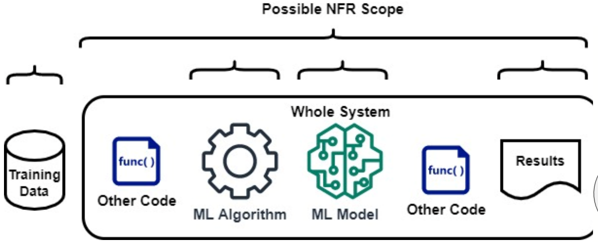
Relieved Face by Anniken & Andreas from NounProject.com

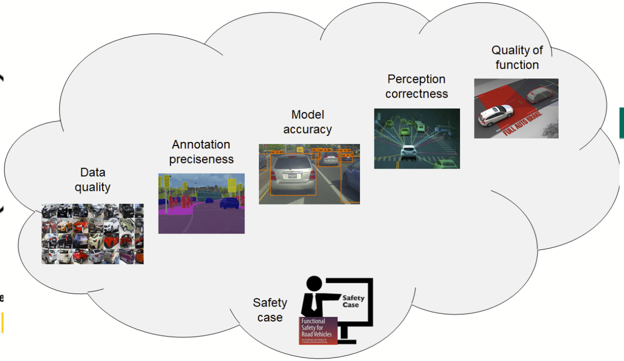

Sample View from outside RE
[Bosch et al., 2018] Case studies on software development approaches "In addition to having development teams executing on requirements specified by product management, the development of software systems is progressing towards a data driven practice where teams receive an outcome to realize and where design decisions are taken based on continuous collection and analysis of data"
Development approach
Definition
Requirement driven development
Software is built to specification. This development approach is predominantly used when new features or functionality are well understood and defined.
Outcome/data driven development
Development teams receive a quantitative target to realize and are asked to experiment with different solutions to improve the metric. Examples of this development approach are new features (used frequently by customers) and innovation efforts.
AI driven development
A company has a large data set available and use artificial intelligence techniques such as machine learning and deep learning to create components that act based on input data and that learn from previous actions. Examples of this development approach include e.g. object recognition in autonomous cars and speech recognition in modern user interfaces.
We are here
Quantitative targets are also requirements (NFRs)
There are requirements here too!
But we should also be/be seen to be here
Why?

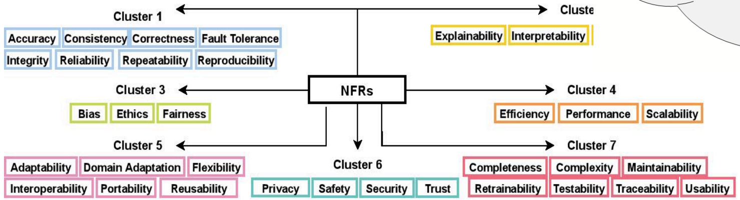

ML
SE
AI
DS
RE

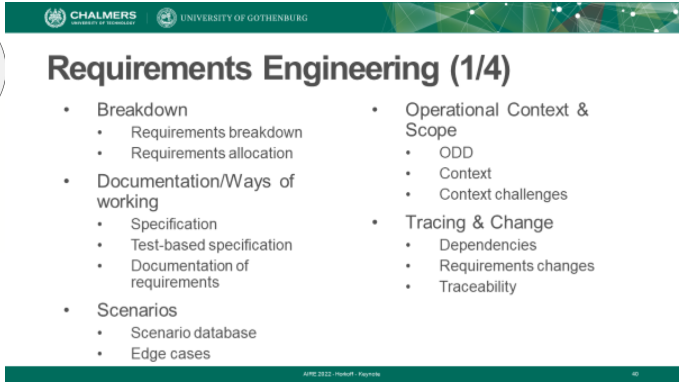

Interview Results: Overview of Themes/codes
Defined Over ML
Defined Over Whole
Defined Over Data
Defined Over
NFR Captured
Measurement Over Whole
Captured
NFRs
Measurement Over
Measurement Over ML
NFR Measurement Captured
Importance
Challenges
Measurement Over Data
Important NFR
NFR Challenges
Less Important NFR
General ML Challenges
NFR Measurement Challenges


Survey Results – NFR for ML Ranking


Possible NFR Scope
Whole System
Training Data
func( )
Other Code
ML Algorithm
ML Model
func( )
Other Code
Results


Quality of function
Perception correctness
Model accuracy
Annotation preciseness
Data quality
Safety case
Safety Case

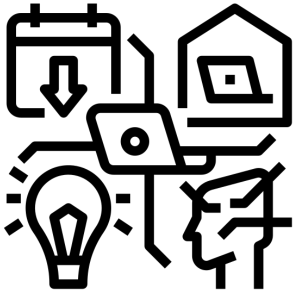

Cluster 1
Accuracy
Consistency
Correctness
Fault Tolerance
Integrity
Reliability
Repeatability
Reproducibility
Cluste
Explainability
Interpretability
NFRs
Cluster 3
Bias
Ethics
Fairness
Cluster 4
Efficiency
Performance
Scalability
Cluster 5
Adaptability
Domain Adaptation
Flexibility
Interoperability
Portability
Reusability
Cluster 6
Privacy
Safety
Security
Trust
Cluster 7
Completeness
Complexity
Maintainability
Retrainability
Testability
Traceability
Usability

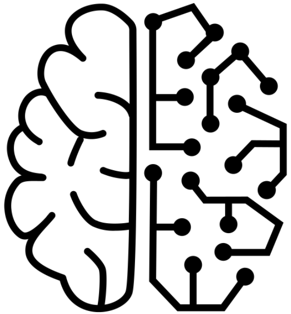

Requirements Engineering (1/4)
Breakdown
Requirements breakdown
Requirements allocation
Documentation/Ways of working
Specification
Test-based specification
Documentation of requirements
Scenarios
Scenario database
Edge cases
Operational Context & Scope
ODD
Context
Context challenges
Tracing & Change
Dependencies
Requirements changes
Traceability

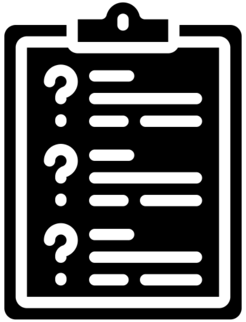
(Our) Future Work

- Process and publish results from case study
- Finish focused sub-set of NFRs for ML SLR
- Design and Evaluation NFRs for ML solutions
- Process and methods for RE for ML

Your Future Work?







Future Of Work by WiStudio from NounProject.com

Machine Learning by Angela from NounProject.com

Requirements by Juicy Fish from NounProject.com

Future Work by AmruID from NounProject.com

CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Questions? ☺

References

[AMLAS] Hawkins, Richard, et al. "Guidance on the assurance of machine learning in autonomous systems (AMLAS)." arXiv preprint arXiv:2102.01564 (2021).

[Anisetti et al., 2020] Anisetti, Marco, et al. "A methodology for non-functional property evaluation of machine learning models." Proceedings of the 12th International Conference on Management of Digital EcoSystems. 2020.[Barrera et al., 2021] J. M. Barrera, A. Reina, A. Maté, J. C. Trujillo, Applying i∗ in conceptual modelling in machine learning, volume 2983, 2021, pp. 56–62. URL: http://ceur-ws.org.

[Belani et al., 2019] H. Belani, M. Vukovic, and ´ Zeljka Car, "Requirements engineering challenges in building AI-based complex systems," in the 6th International Workshop on Artificial Intelligence for Requirements Engineering (AIRE 2019), September 2019.

[Bosch & Olsson, 2017] J. Bosch and H. H. Olsson, "Toward Evidence-Based Organizations: Lessons from Embedded Systems, Online Games, and the Internet of Things," in IEEE Software, vol. 34, no. 5, pp. 60-66, 2017, doi: 10.1109/MS.2017.3571569.]

[Bosch et al., 2018] Bosch, Jan, Helena Holmström Olsson, and Ivica Crnkovic. "It takes three to tango: Requirement, outcome/data, and AI driven development." SiBW. 2018.

[Berntsson Svensson et al., 2012] Berntsson Svensson, R., Sprockel, Y., Regnell, B. et al. Setting quality targets for coming releases with QUPER: an industrial case study. Requirements Eng 17, 283–298 (2012). https://doi.org/10.1007/s00766-011-0125-0]

[Habibullah & Horkoff, 2021] Habibullah, Khan Mohammad, and Jennifer Horkoff. "Non-functional requirements for machine learning: understanding current use and challenges in industry." 2021 IEEE 29th International Requirements Engineering Conference (RE). IEEE, 2021.

[Habibullah et al., 2022] Habibullah, Khan Mohammad, Gregory Gay, and Jennifer Horkoff. "Non-Functional Requirements for Machine Learning: An Exploration of System Scope and Interest." ICSE Workshops, arXiv preprint arXiv:2203.11063 (2022).

[Hamada et al. 2020] Hamada, Koichi, et al. "Guidelines for Quality Assurance of Machine Learning-based Artificial Intelligence." SEKE. 2020.

[Horkoff 2019] J. Horkoff, Non-functional requirements for machine learning: Challenges and new directions, IEEE, 2019, pp. 386–391. doi:10.1109/RE.2019.00050.

[Hu et al. 2020] Hu, Boyue Caroline, et al. "Towards requirements specification for machine-learned perception based on human performance." 2020 IEEE Seventh International Workshop on Artificial Intelligence for Requirements Engineering (AIRE). IEEE, 2020.

[Ishikawa & Matsuno, 2020] F. Ishikawa, Y. Matsuno, Evidence-driven requirements engineering for uncertainty of machine learning-based systems, IEEE, 2020, pp. 346–351. doi:10.1109/RE48521.2020.00046.

[Nalchigar et al. 2019] S. Nalchigar, E. Yu, Y. Obeidi, S. Carbajales, J. Green, A. Chan, Solution patterns for machine learning, volume 11483 LNCS, Springer Verlag, 2019, pp. 627–642. URL: https://doi.org/10.1007/978-3-030-21290-2_39. doi:10.1007/978- 3- 030- 21290- 2_39.

[Rahimi et al. 2019] Rahimi, Mona, et al. "Toward requirements specification for machine-learned components." *2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)*. IEEE, 2019.

[Villamizar et al. 2021] Villamizar, Hugo, Tatiana Escovedo, and Marcos Kalinowski. "Requirements engineering for machine learning: A systematic mapping study." 2021 47th Euromicro Conference on Software Engineering and Advanced Applications (SEAA). IEEE, 2021.

[Vogelsang & Borg, 2019] A. Vogelsang and M. Borg, "Requirements engineering for machine learning: Perspectives from data scientists," in the 6th International Workshop on Artificial Intelligence for Requirements Engineering (AIRE 2019), September 2019.

[Wan et al., 2020] Z. Wan, X. Xia, D. Lo, G. C. Murphy, How does machine learning change software development practices?, IEEE Transactions on Software Engineering (2020) 1–1. doi:10.1109/TSE.2019.2937083.

UNIVERSITY OF GOTHENBURG

CHALMERS

UNIVERSITY OF TECHNOLOGY