機械学習工学研究会

MACHINE LEARNING SYSTEMS ENGINEERING

Introduction of MLSE Activities in Japan

Fuyuki Ishikawa

National Institute of Informatics

Chair of MLSE

AI Dependability?

AI and ML

- Everyone is talking about AI and ML, recently more about risks and concerns in terms of dependability

Accidents of autonomous cars

[http://www.dailymail.co.uk/news/article-3677101/Tesla-told-regulators-fatal-Autopilot-crash-nine-days-happened.html]

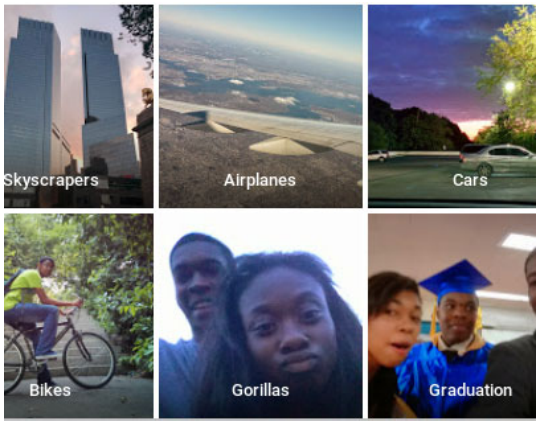
TECHNOLOGY

Microsoft Created a Twitter Bot to Learn From Users. It Quickly Became a Racist Jerk.

By DANIEL VICTOR MARCH 24, 2016

Improper online learning

Technically unsolved problems at Google Photo



[https://www.nytimes.com/2016/03/25/technology/microsoft-created-a-twitter-bot-to-learn-from-users-it-quickly-became-a-racist-jerk.html]

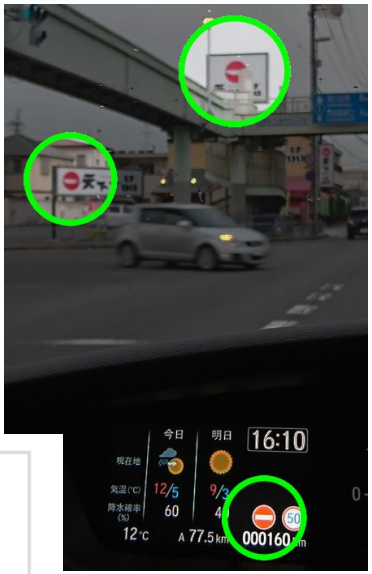
[https://www.theguardian.com/technology/2015/jul/01/google-sorry-racist-auto-tag-photo-app]
[https://www.theguardian.com/technology/2018/jan/12/google-racism-ban-gorilla-black-people]

What will you do if you are responsible? (1)

- When Honda sees ramen shop sign
  - First buzz in Dec 2017

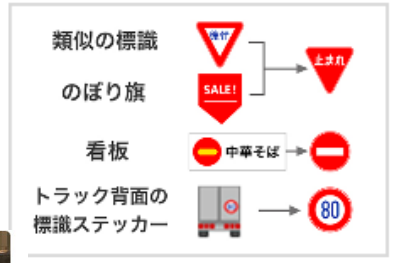
[http://www.tenkaippin.co.jp/company.html]

[https://twitter.com/_gyochan_/status/938240168078622720]
[https://twitter.com/Bleu_kakeru727/status/937680760491753473]



  - Now a caution on the web site

http://www.honda.co.jp/hondasensing/feature/srf/



色や形の判別が
つきにくく、
対象の標識が無いのに、
標識を表示する

  - Second buzz in Sep 2018

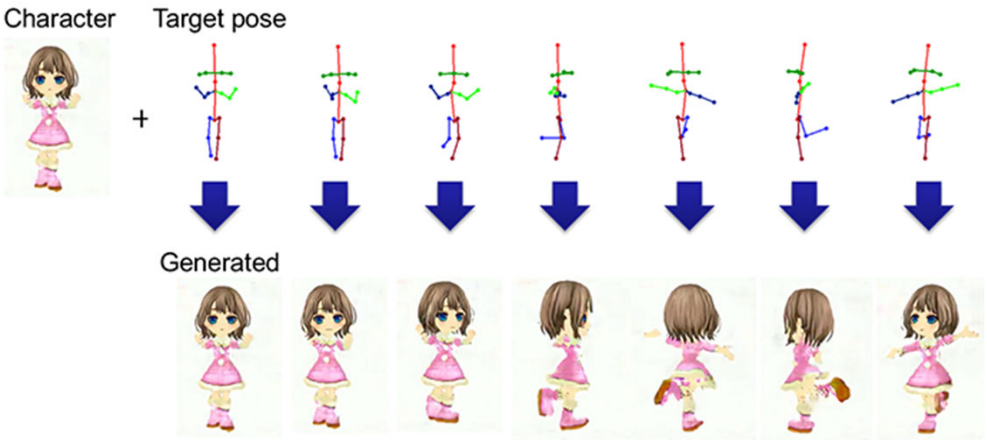
8:24 PM - 12 Sep 2018

15,099 Retweets 20,908 Likes

Can you find beforehand or prevent adverse (?) news??

What will you do if you are responsible? (2)

- From DeNA (May 2018)
  - Generate an image of a certain pose
  - Generate a movie given a pose sequence while smoothly changing the character



[https://dena.com/intl/anime-generation/]

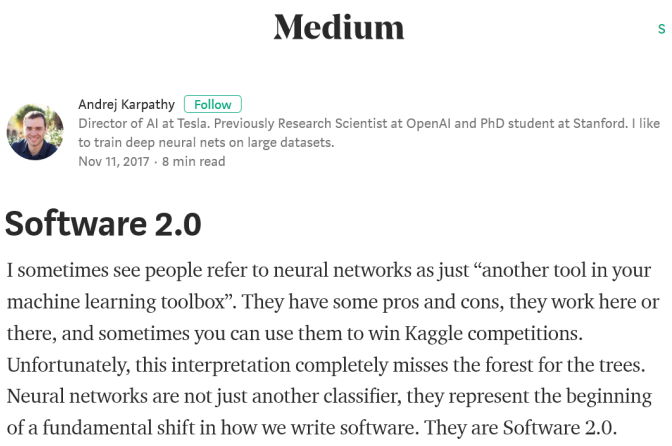
What do you ensure to sell this to anime companies?

Cause of New Difficulties

Software 2.0 or Inductive Software Dev.

- Let us focus on ML
  - Present movement on AI was driven by ML, specifically advance in deep learning techniques
  - With ML, we construct a software component in a different way: derive the rule that governs the behavior from training data (not directly from engineers)
  - In the Japanese industry, the terms "inductive software development" and "inductive programming" is also used

[https://medium.com/@karpathy/software-2-0-a64152b37c35]


Medium

Andrej Karpathy Follow
Director of AI at Tesla. Previously Research Scientist at OpenAI and PhD student at Stanford. I like to train deep neural nets on large datasets.
Nov 11, 2017 · 8 min read

Software 2.0

I sometimes see people refer to neural networks as just "another tool in your machine learning toolbox". They have some pros and cons, they work here or there, and sometimes you can use them to win Kaggle competitions. Unfortunately, this interpretation completely misses the forest for the trees. Neural networks are not just another classifier, they represent the beginning of a fundamental shift in how we write software. They are Software 2.0.

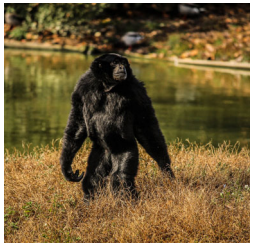
Example

Boundary created from training data

Gibbon

35 24 210	20 121 24	122 81 20
211 54 42	12 222 90	88 79 116
24 36 98	98 181 31	66 31 198

13 83 33	13 45 94	75 74 111
111 8 73	192 1 221	237 31 1
74 35 122	93 76 244	73 211 45

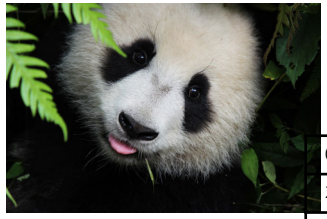


0 245 210	20 12 114	84 99 100
11 86 99	121 88 91	18 0 77
46 87 121	70 76 122	122 14 94

Panda

254 32 67	222 88 1	108 76 14
12 86 222	98 75 122	111 74 74
198 87 33	188 173 4	68 176 83

77 81 123	122 158 6	76 63 42
3 3 78	19 183 84	76 63 123
98 83 111	123 7 99	253 48 91



0 24 31	20 21 124	12 101 50
21 54 242	112 22 90	8 79 214
124 56 85	98 99 141	166 1 198

[http://free-photos.gatag.net/]

Resulting Characteristics (1)

Imperfect and has limitation on performance, impossible to estimate the performance before construction or changes

- Contract not based on the specification but only on “the best effort together”
- Half-a-year effort to find “we should give up”
- Requiring much courage of users/customers and creative ideas to find acceptable usages with the inevitable imperfectness

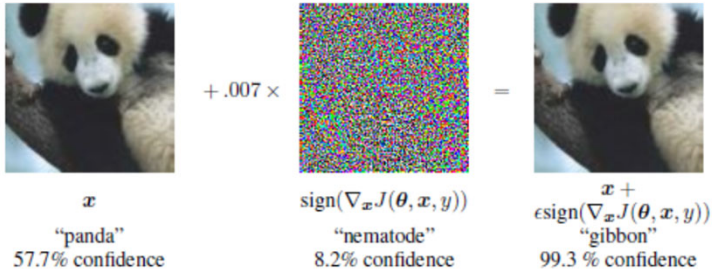
Resulting Characteristics (2)

Often difficult/costly or impossible to define the right output for each arbitrary input
(no deductive/logical specification, non-testable)

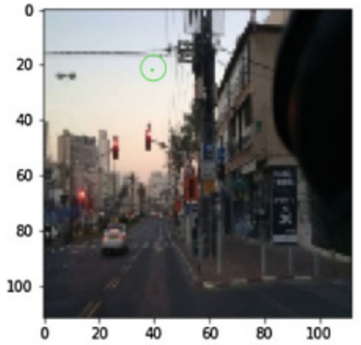
- The “unit testing” principle invalidated
- Obvious faults (in training data, configuration, learning-algorithm code) not detected, possibly
- Fault localization (debugging) very difficult
- Not to straightforward to have enormous number of test cases to increase the confidence (e.g., random testing)

Resulting Characteristics (3)

Impossible to describe the boundary of what can be done and what cannot be done (in a human-interpretable way)



Have adversarial examples (slight input changes cause large output change)



- Very difficult to have confidence on the quality
- The "equivalence class" principle invalidated

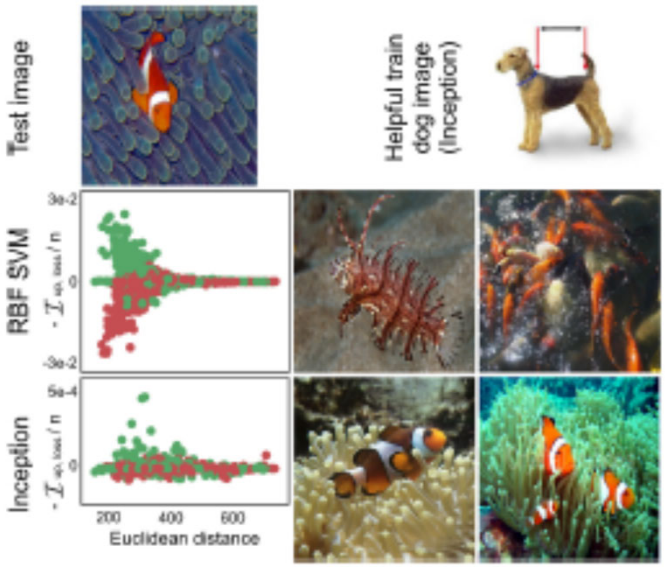
[Goodfellow et al., Explaining and Harnessing Adversarial Examples, 2015]

[Wicker et al., Feature-Guided Black-Box Safety Testing of Deep Neural Networks, 2018]

Resulting Characteristics (4)

Blackbox and unexplainable for
why individual results are obtained
or how the component behaves
(in a human-interpretable way)

Naïve decision for the correct answer "fish"



a) Husky classified as wolf (b) Explanation

Judging with snow

- Possible semantic gaps from human expectation
- Obstacles for user/customer acceptance

[Ribeiro et. al., " Why Should I Trust You?": Explaining the Predictions of Any Classifier, 2016]

[Koh et. al., Understanding Black-box Predictions via Influence Functions, 2017]

e.g., Active Research on Testing/Verification

- SMT-based verification [CAV'17]
- Search-based testing with "neuron coverage" criteria [SOSP'17] [ICSE'18]
- Testing based on system-level requirements [NFM'17]
- Verification by stochastic game [TACAS'18]
- Metamorphic testing [ISSTA'18]
- Verification by abstract interpretation [ICML'18]
- Empirical study on bug statistics [ISSTA'18]
- Mutation analysis [ISSRE'18]
- Updated neuron coverage criteria [ASE'18]
- Concolic testing [ASE'18]
- Fairness testing [ASE'18]
- Uncertainty classification [WAISE'18]
- Structuring of assurance aspects [WAISE'18]
- …

MLSE Activities

MLSE

- *Focus on ML, not general AI*
- *Discuss systems not only software*
- *Focus on SE4ML not ML4SE*

■Special Interest Group on

Machine Learning Systems Engineering

■Formally since Apr. 2018, with JSSST (an academic society in Japan)

■Trigger: panel discussion at a SE event in Sep. 2017, and following interactions in Facebook

■10 events so far (plus 3 planned):

symposium, discussion camps, survey presentation, …

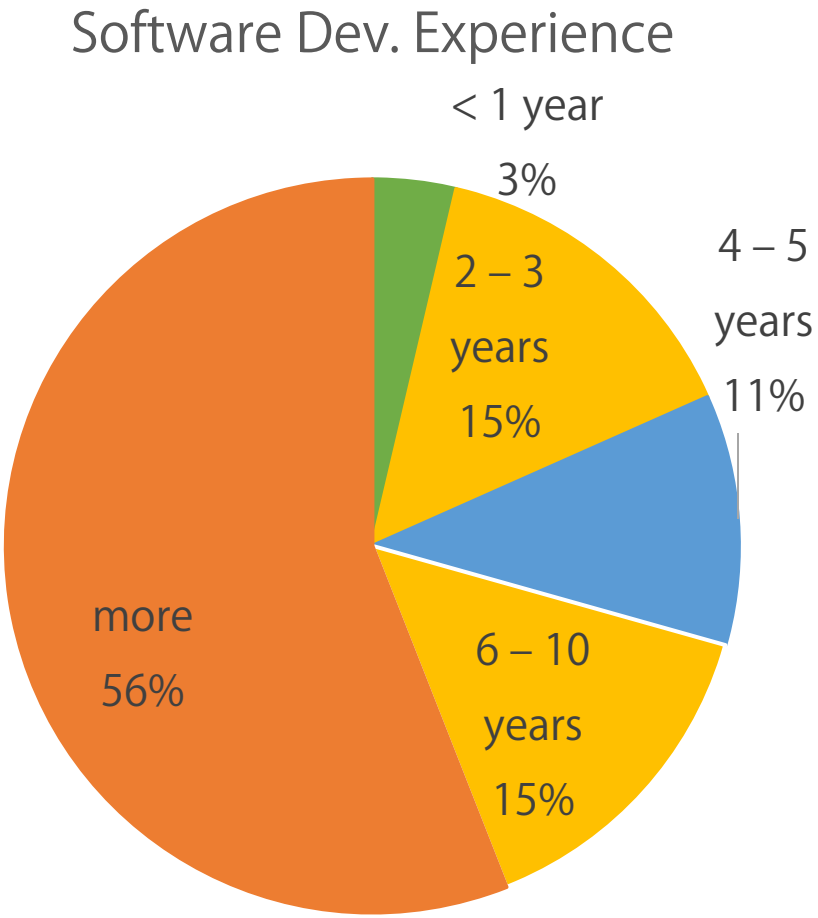
Got 26 sponsors and almost 500 attendees

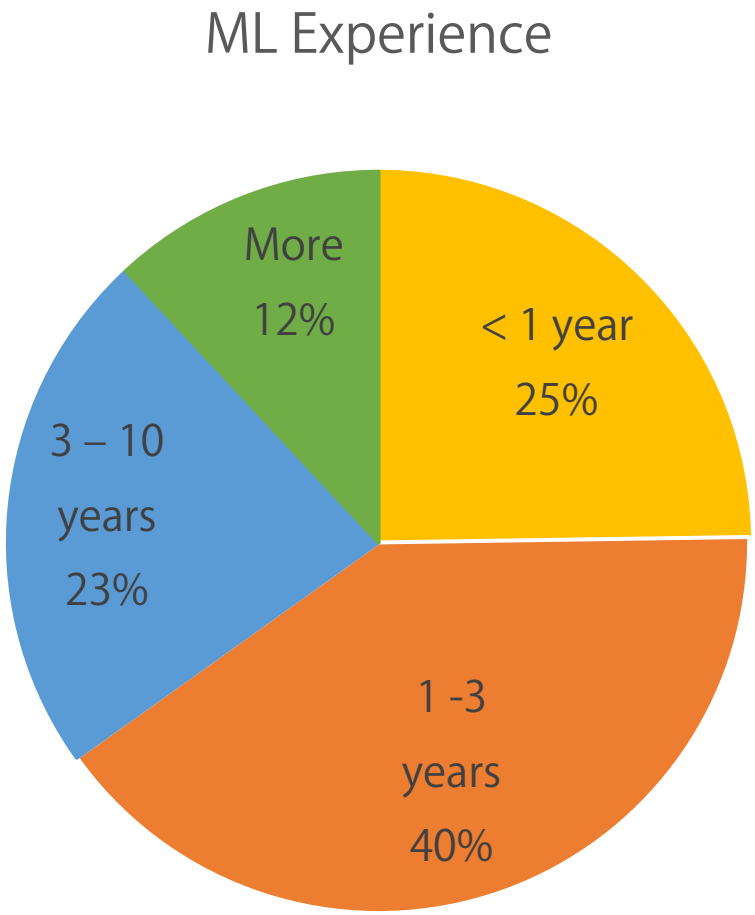
Most of the attendees have been from the industry

Questionnaire Survey

- Method
  - Dissemination by mailing lists and social networks (software engineering, ML, and AI)
  - “those who have used ML at work”
  - 280 answers
- Question aspects
  - Experience on SE activities and on ML techniques
  - Past projects that used ML
  - Quality attributes considered significant
  - Perception of difficulties
  - Characteristics of ML that lead to the difficulties

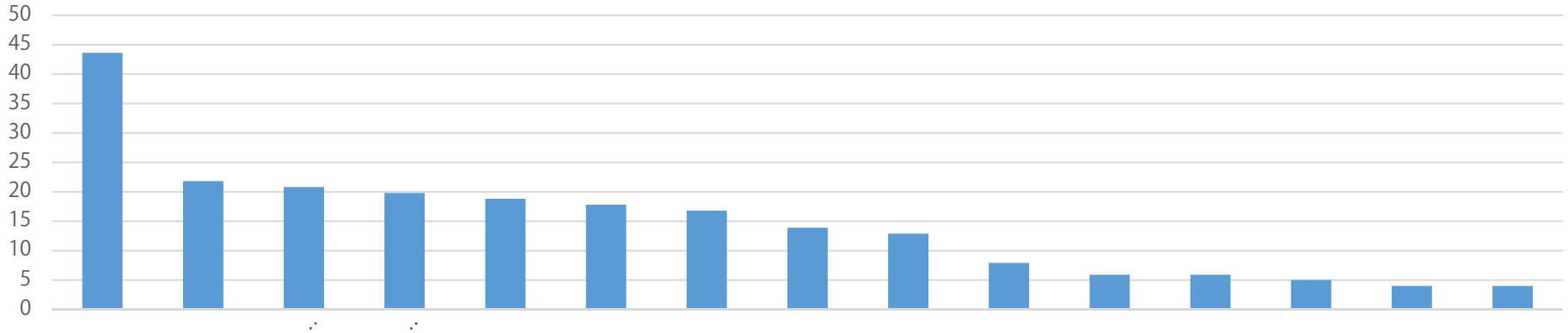
Profiling





Experienced engineers were recently pushed to learn and use ML

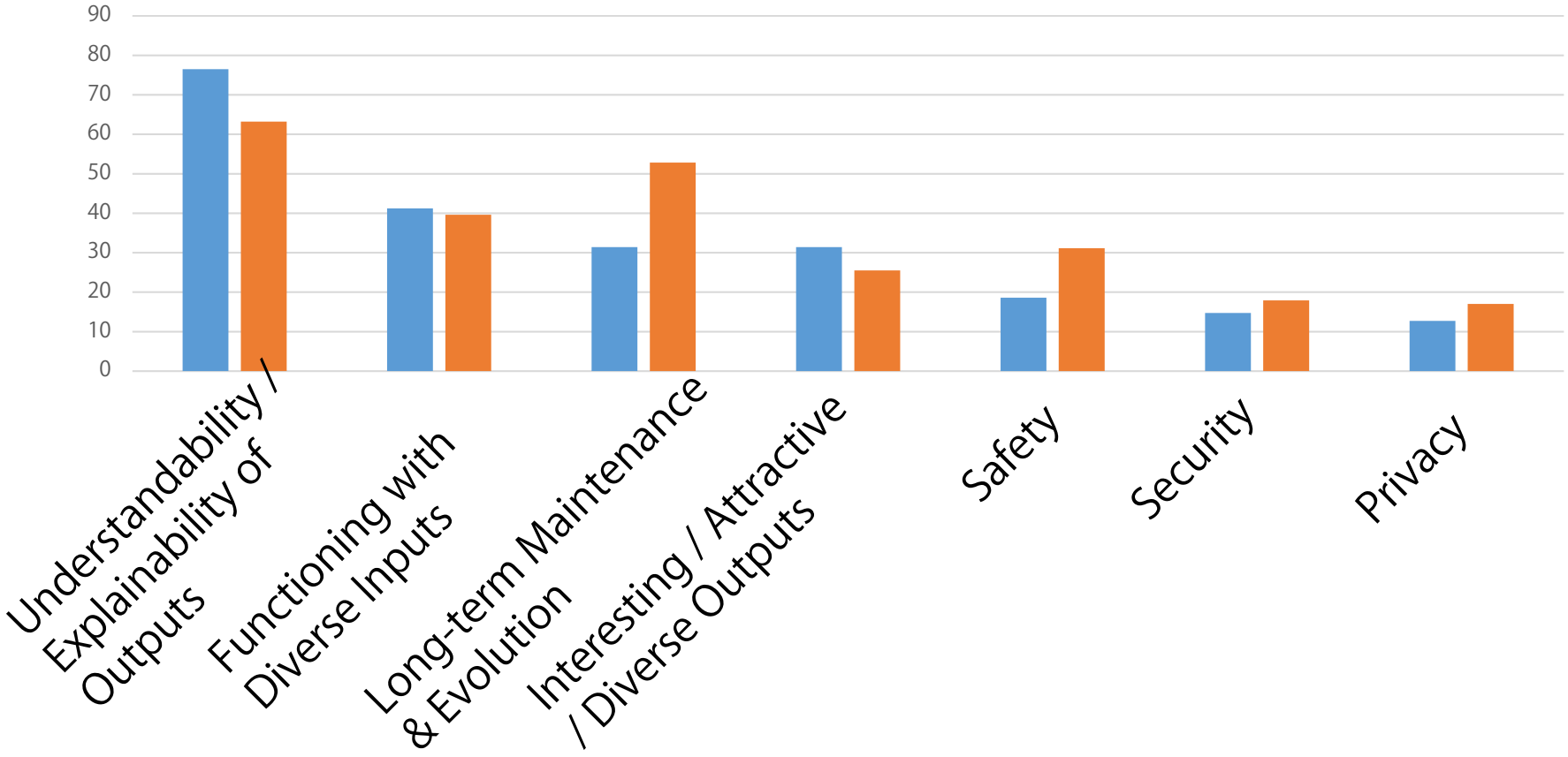
ML Usage Domain



Manufacturing
Information & Communication
Company-Specific Services
Foundational Development (e.g., middleware)
Academic Research
Automotive / Railway
Finance / Insurance
Home Appliance
Wholesale and Retail
Education
Life Service / Amusement
Medical / Welfare
Construction
Real Estate
Utility

Application in manufacturing is large (in Japan)

Significant Quality Attributes



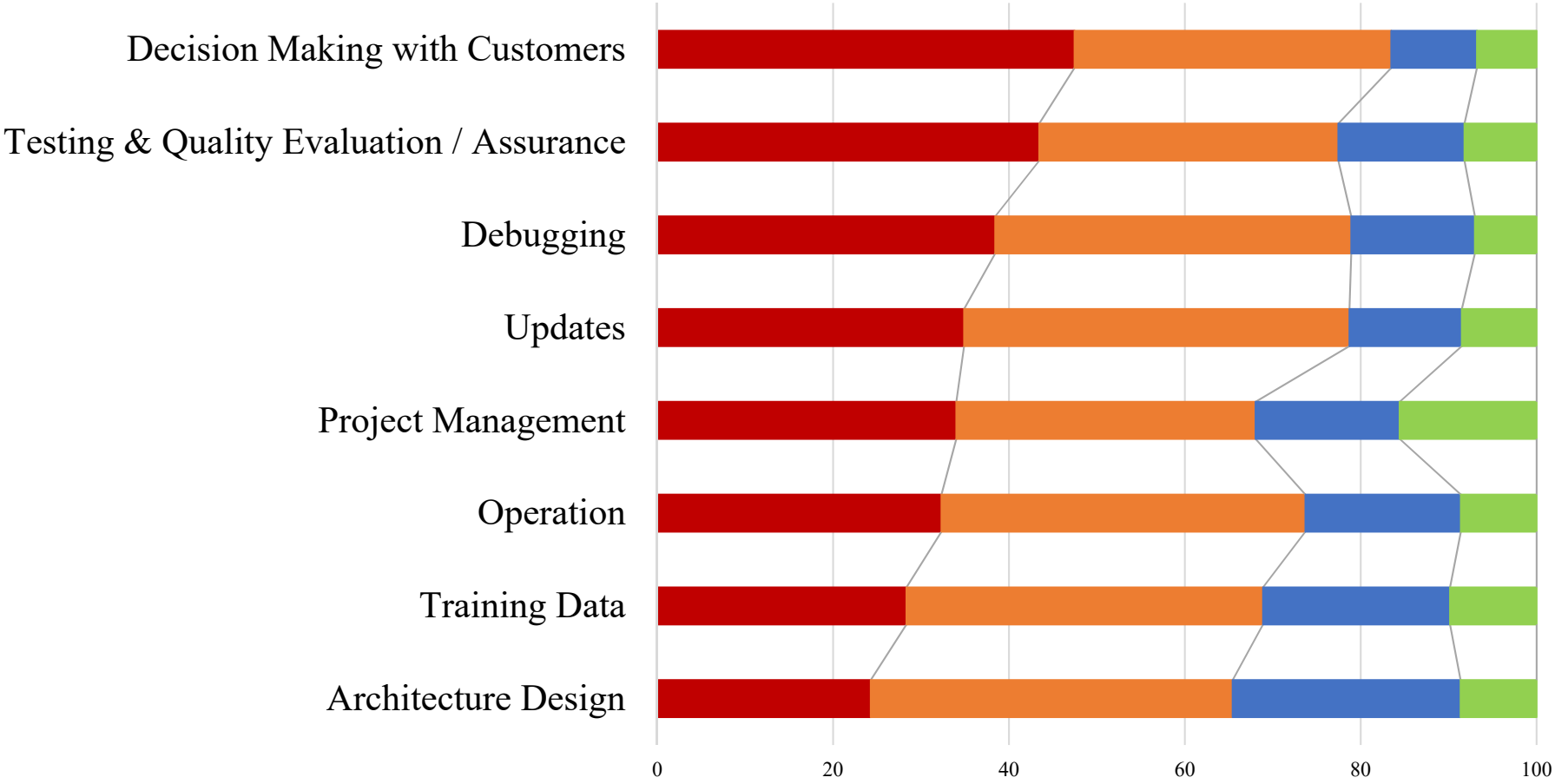
Significant attributes in the past projects
Attributes expected to be significant in future

XAI (eXplainable AI) is thought as significant also in practice
Maintenance, security, and privacy are somewhat left behind

Difficulty Level

We need to use new approaches as the existing ones do not work anymore

We can apply the same approaches but methods, etc., are still immature



We already have dedicated methods, etc.

We can use existing methods, frameworks, or tools

Some Insights

(at least in Japan)

- Engineers are facing with difficulties in the engineering aspect, rather than (or not only) algorithmic aspects that they were unfamiliar with but could learn

- High uncertainty in the implementation, not only requirements/environments, is one of the core causes

- Research papers are emerging but often limited to "easy to evaluate" studies, such as testing and verification for finding adversarial examples, rather than practical disciplines

Summary

- MLSE is a new paradigm
  - Essentially different with unique difficulties (not a buzzword)
  - Strong demands from the industry

We need your research contributions!

機械学習工学研究会
MACHINE LEARNING SYSTEMS ENGINEERING