How Practitioners Perceive the Relevance of Software Engineering Research

Test-of-Time Award Talk

David Lo, Nachiappan Nagappan, Thomas Zimmermann

Microsoft Research

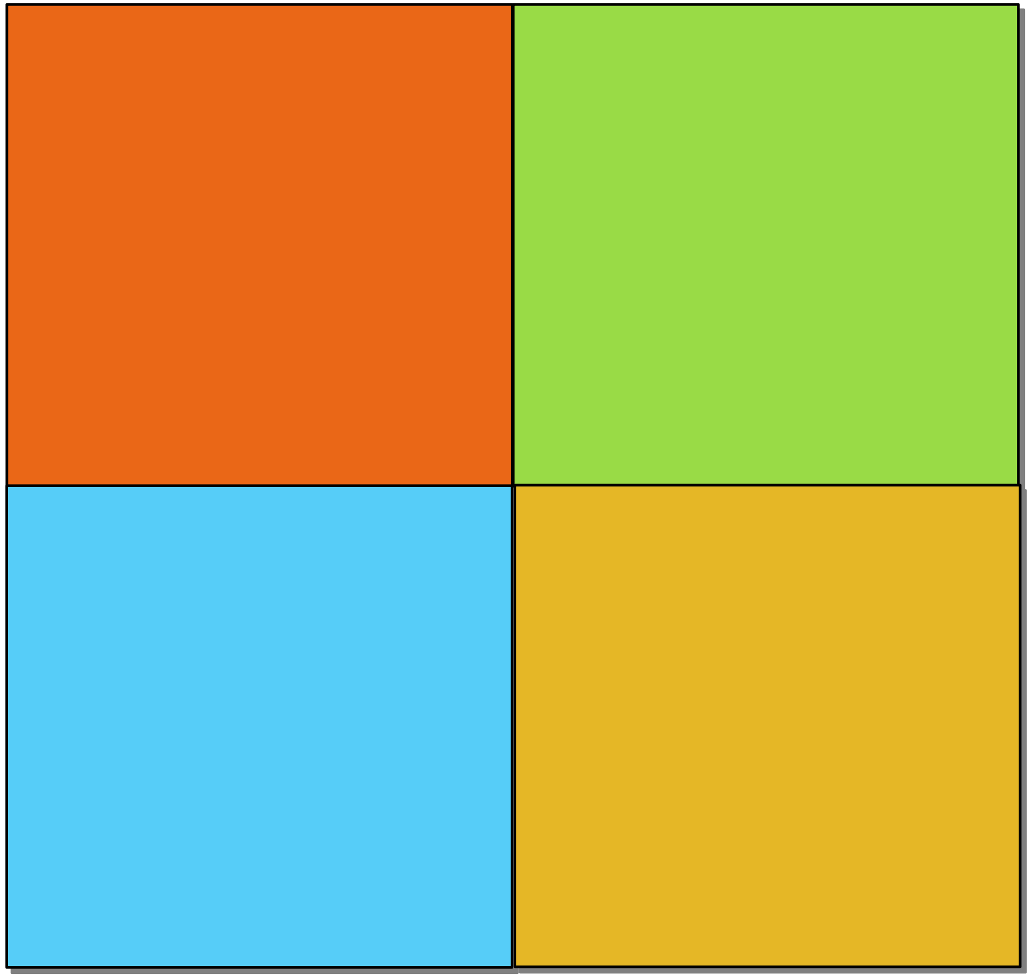
SMU Singapore Management University

Meta

The University of California Irvine

FSE 2025, Trondheim, Norway, June 2025

Going Back a Decade



What Is the Paper About?

How Has It Influenced Subsequent Studies?

What Is the Road Ahead?

Going Back a Decade

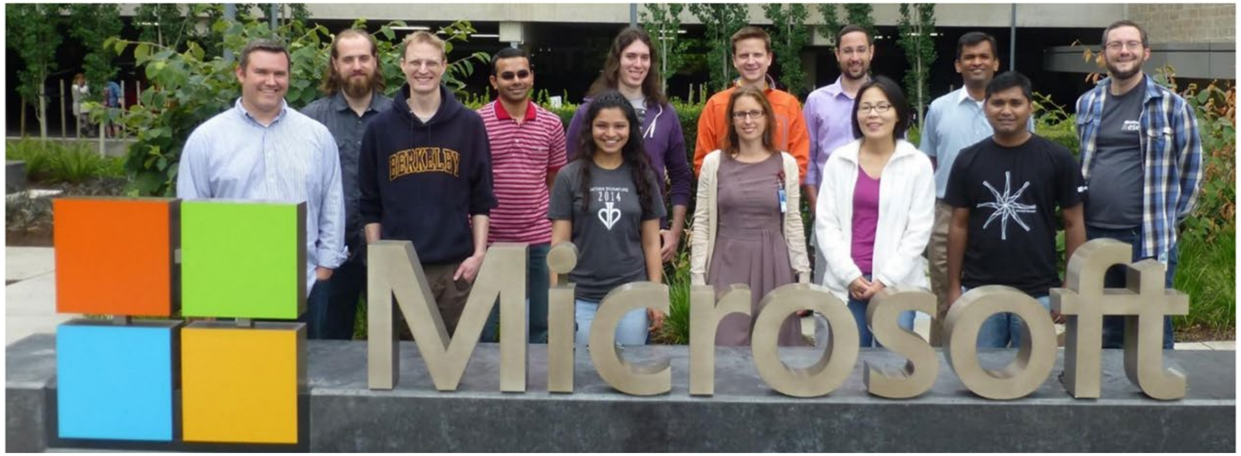
What Is the Paper About?

How Has It Influenced Subsequent Studies?

What Is the Road Ahead?

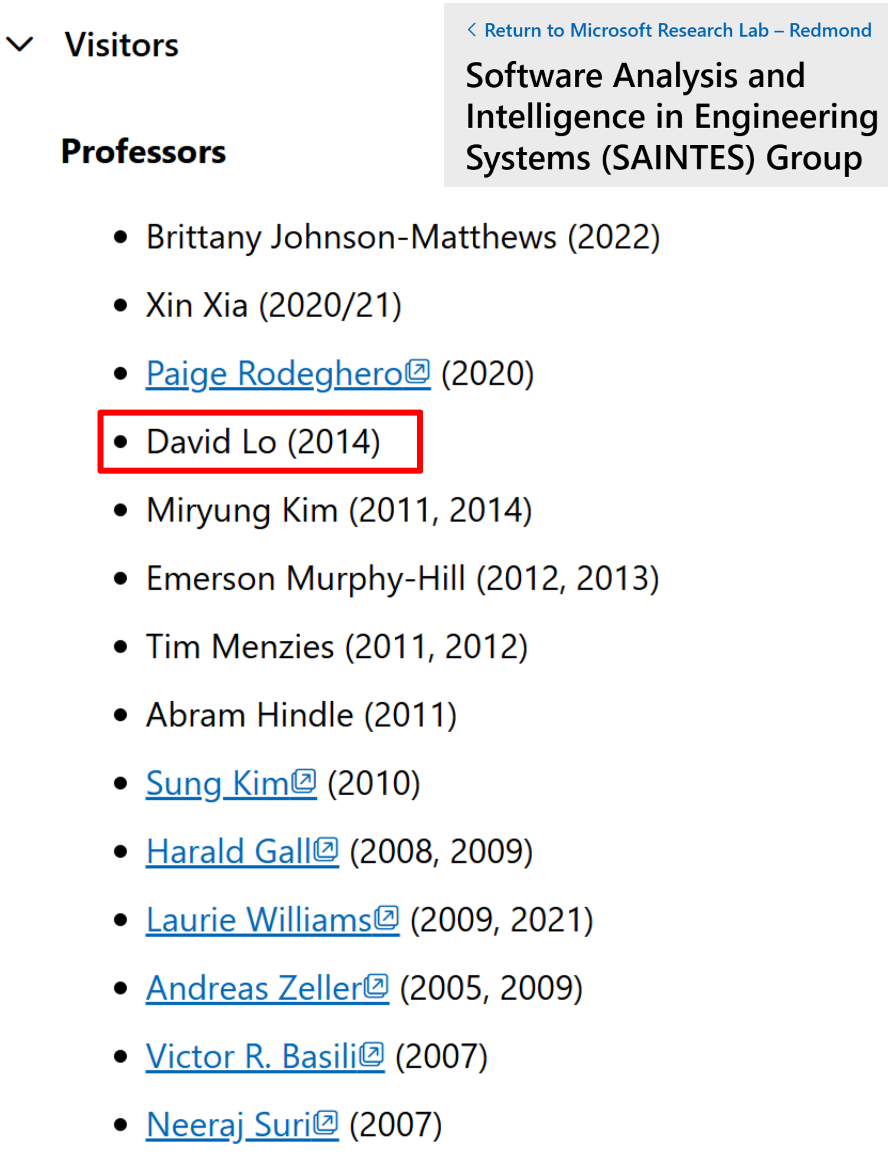
Going Back a Decade

Empirical Software Engineering Group (ESE)



ESE Group in Summer 2014

David (2014) – started the visit 1 week after the group photo



‹ Return to Microsoft Research Lab – Redmond

Software Analysis and Intelligence in Engineering Systems (SAINTES) Group

Visitors

Professors

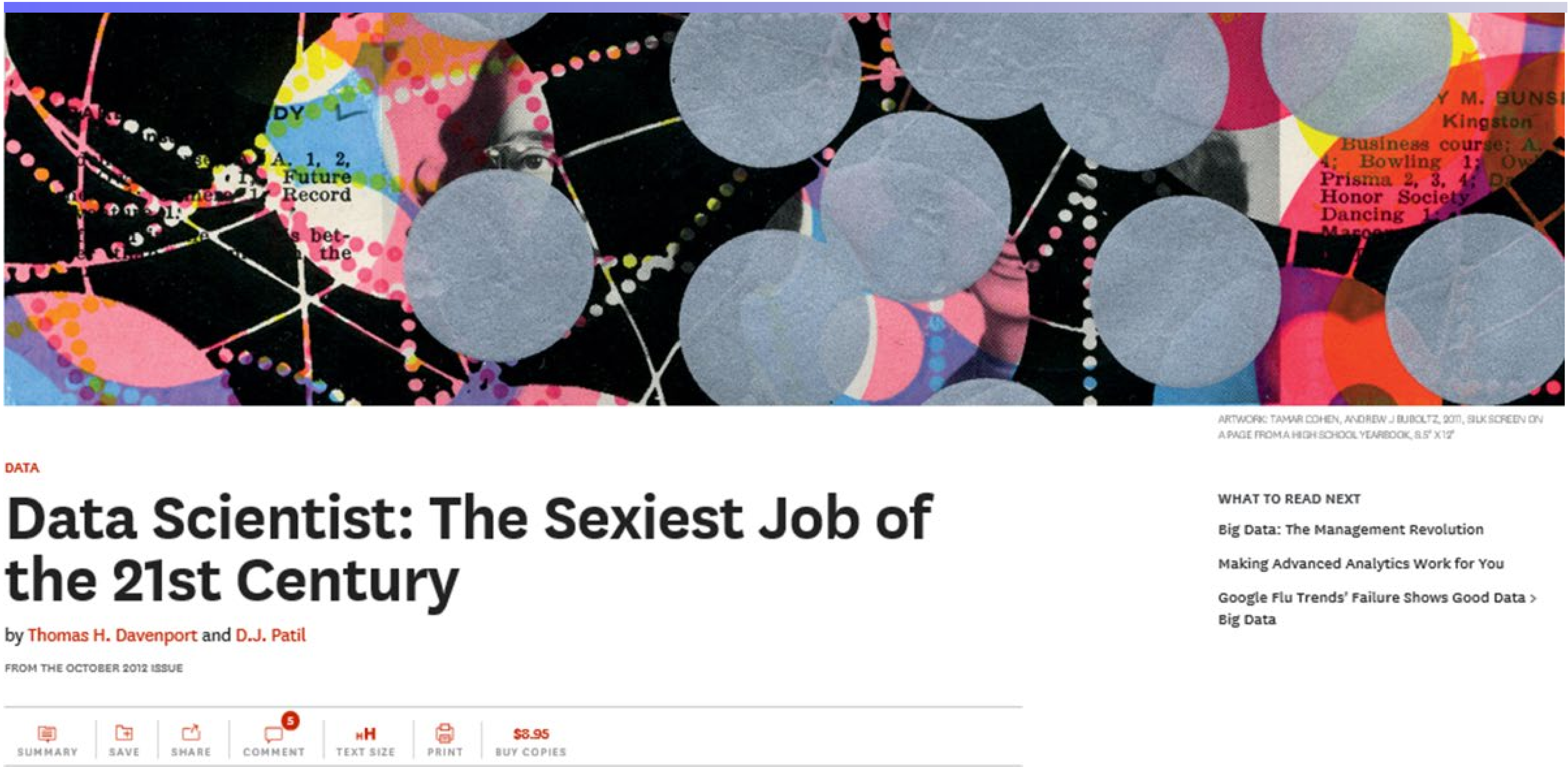
- Brittany Johnson-Matthews (2022)
- Xin Xia (2020/21)
- Paige Rodeghero (2020)
- David Lo (2014)
- Miryung Kim (2011, 2014)
- Emerson Murphy-Hill (2012, 2013)
- Tim Menzies (2011, 2012)
- Abram Hindle (2011)
- Sung Kim (2010)
- Harald Gall (2008, 2009)
- Laurie Williams (2009, 2021)
- Andreas Zeller (2005, 2009)
- Victor R. Basili (2007)
- Neeraj Suri (2007)

Microsoft Research

SMU SINGAPORE MANAGEMENT UNIVERSITY

Meta

Going Back a Decade



ARTWORK: TAMAR COHEN, ANDREW J BUBOLTZ, 2011, SILK SCREEN ON A PAGE FROM A HIGH SCHOOL YEARBOOK, 8.5" X 12"

DATA

Data Scientist: The Sexiest Job of the 21st Century

by Thomas H. Davenport and D.J. Patil

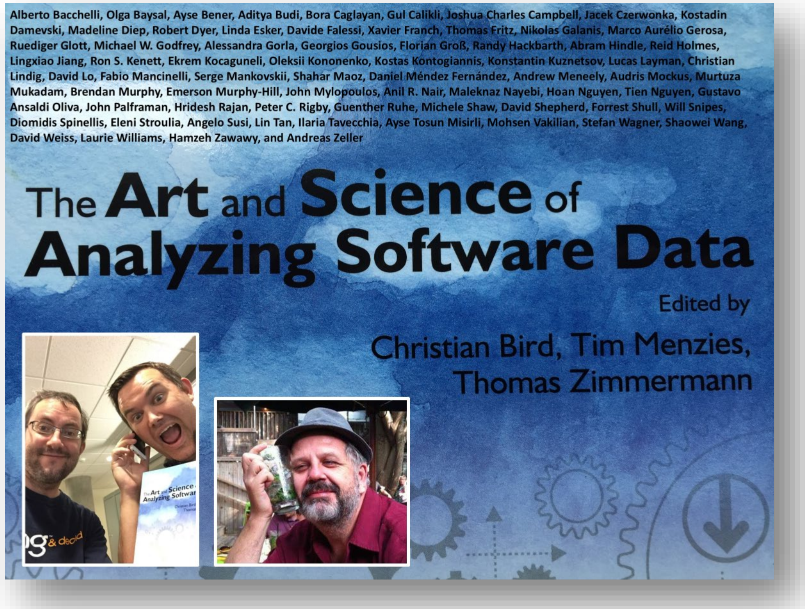
FROM THE OCTOBER 2012 ISSUE

WHAT TO READ NEXT

Big Data: The Management Revolution

Making Advanced Analytics Work for You

Google Flu Trends' Failure Shows Good Data > Big Data

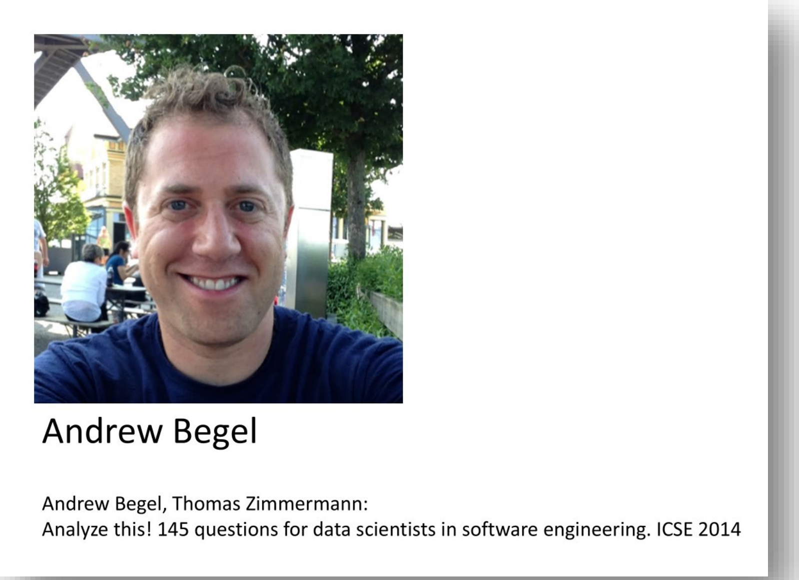


Alberto Bacchelli, Olga Baysal, Ayse Bener, Aditya Budi, Bora Caglayan, Gul Calikli, Joshua Charles Campbell, Jacek Czerwonka, Kostadin Damevski, Madeline Diep, Robert Dyer, Linda Esker, Davide Falessi, Xavier Franch, Thomas Fritz, Nikolas Galanis, Marco Aurélio Gerosa, Ruediger Glott, Michael W. Godfrey, Alessandra Gorla, Georgios Gousios, Florian Groß, Randy Hackbarth, Abram Hindle, Reid Holmes, Lingxiao Jiang, Ron S. Kenett, Ekrem Kocaguneli, Oleksii Kononenko, Kostas Kontogiannis, Konstantin Kuznetsov, Lucas Layman, Christian Lindig, David Lo, Fabio Mancinelli, Serge Mankovskii, Shahar Maoz, Daniel Méndez Fernández, Andrew Meneely, Audris Mockus, Murtuza Mukadam, Brendan Murphy, Emerson Murphy-Hill, John Mylopoulos, Anil R. Nair, Maleknaz Nayebi, Hoan Nguyen, Tien Nguyen, Gustavo Ansaldi Oliva, John Palframan, Hridesh Rajan, Peter C. Rigby, Guenther Ruhe, Michele Shaw, David Shepherd, Forrest Shull, Will Snipes, Diomidis Spinellis, Eleni Stroulia, Angelo Susi, Lin Tan, Ilaria Tavecchia, Ayse Tosun Misirli, Mohsen Vakilian, Stefan Wagner, Shaowei Wang, David Weiss, Laurie Williams, Hamzeh Zawawy, and Andreas Zeller

The Art and Science of Analyzing Software Data

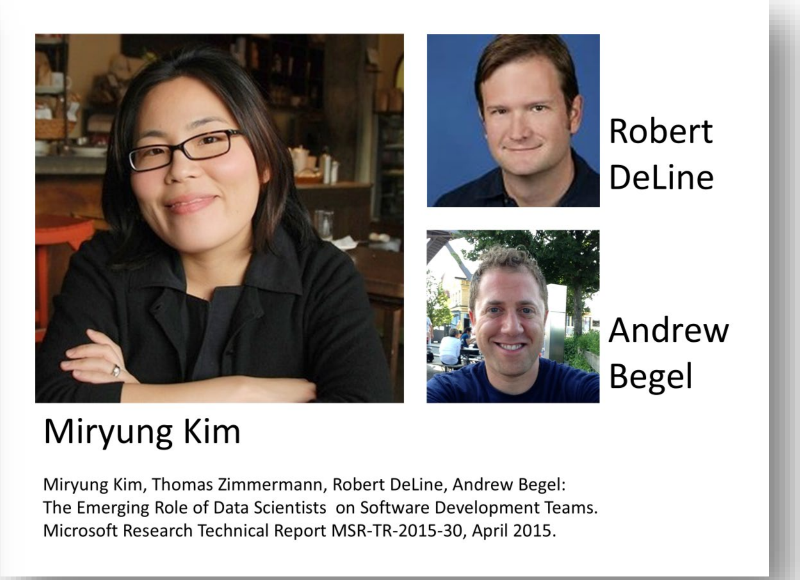
Edited by

Christian Bird, Tim Menzies, Thomas Zimmermann



Andrew Begel

Andrew Begel, Thomas Zimmermann:
Analyze this! 145 questions for data scientists in software engineering. ICSE 2014



Robert DeLine

Andrew Begel

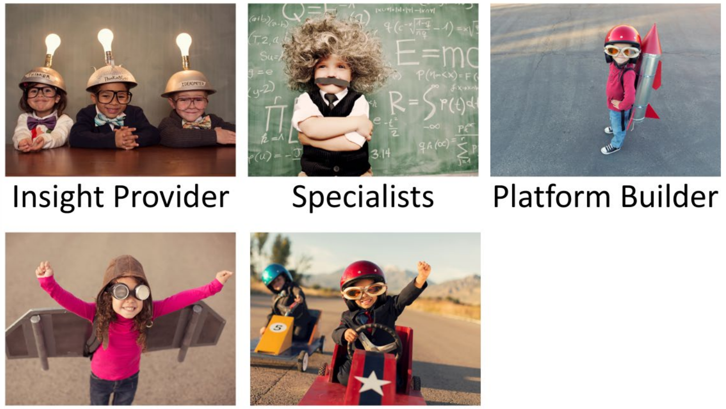
Miryung Kim

Miryung Kim, Thomas Zimmermann, Robert DeLine, Andrew Begel:
The Emerging Role of Data Scientists on Software Development Teams.
Microsoft Research Technical Report MSR-TR-2015-30, April 2015.

JULY/AUGUST 2013
WWW.COMPUTER.ORG/SOFTWARE
IEEE Software
SOFTWARE ANALYTICS: SO WHAT?
IEEE
IEEE computer society
SEPTEMBER/OCTOBER 2013
WWW.COMPUTER.ORG/SOFTWARE
IEEE Software
THE MANY FACES OF SOFTWARE ANALYTICS
IEEE
IEEE computer society

http://aka.ms/145Questions

Working Styles of Data Scientists



Insight Provider

Specialists

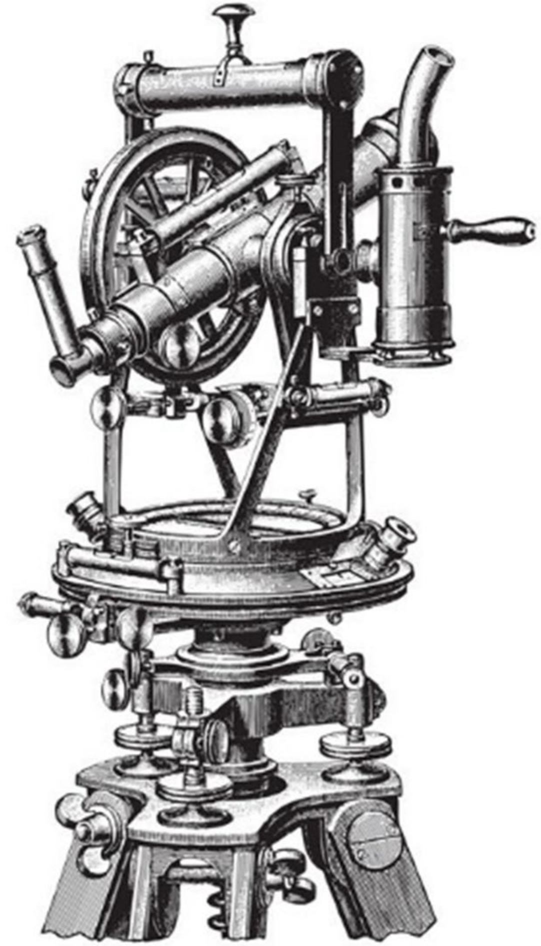
Platform Builder

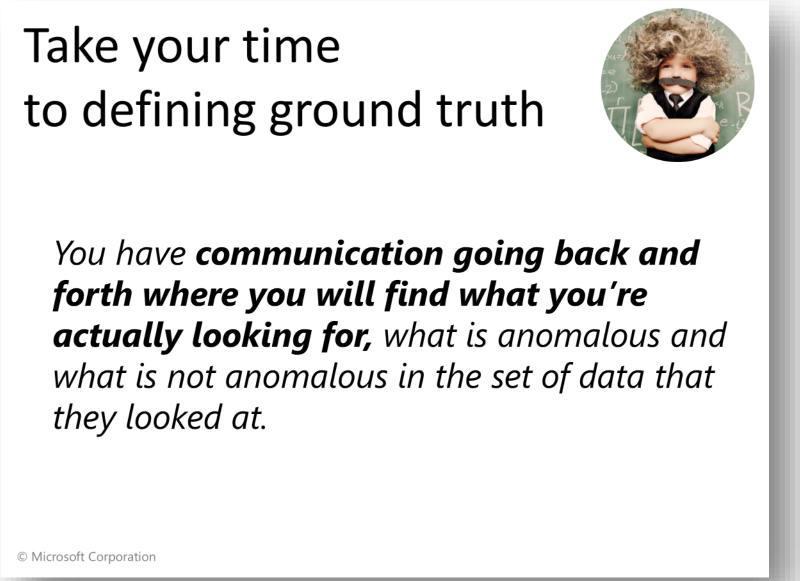
Polymath

Team Leader

The Lens of

RELEVANCE

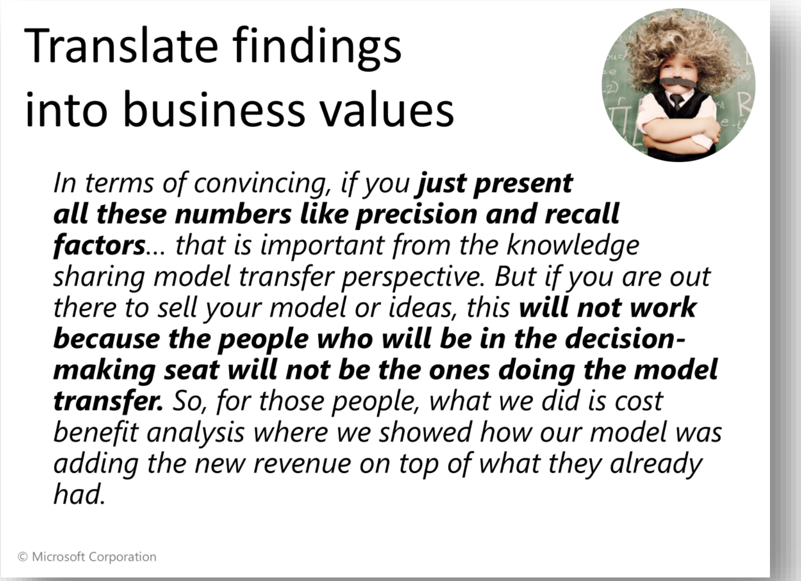




Take your time to defining ground truth

*You have **communication going back and forth where you will find what you're actually looking for,** what is anomalous and what is not anomalous in the set of data that they looked at.*

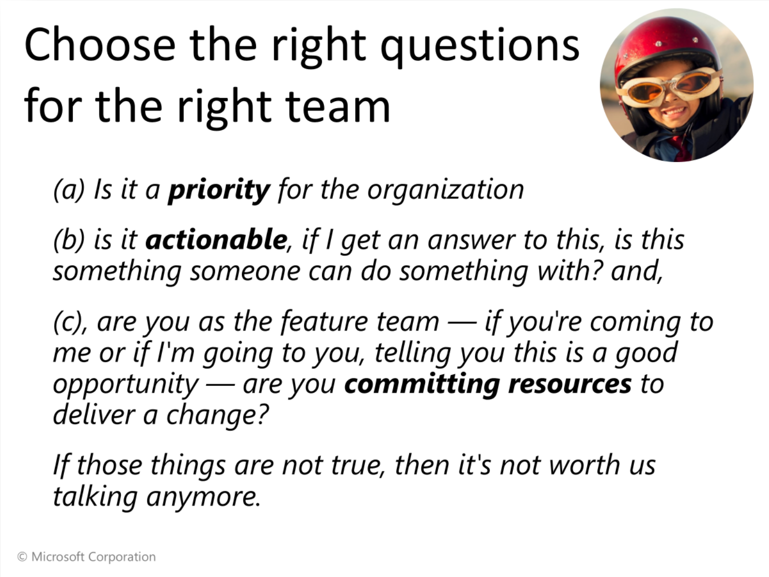
© Microsoft Corporation



Translate findings into business values

*In terms of convincing, if you **just present all these numbers like precision and recall factors**... that is important from the knowledge sharing model transfer perspective. But if you are out there to sell your model or ideas, this **will not work because the people who will be in the decision-making seat will not be the ones doing the model transfer.** So, for those people, what we did is cost benefit analysis where we showed how our model was adding the new revenue on top of what they already had.*

© Microsoft Corporation



Choose the right questions for the right team

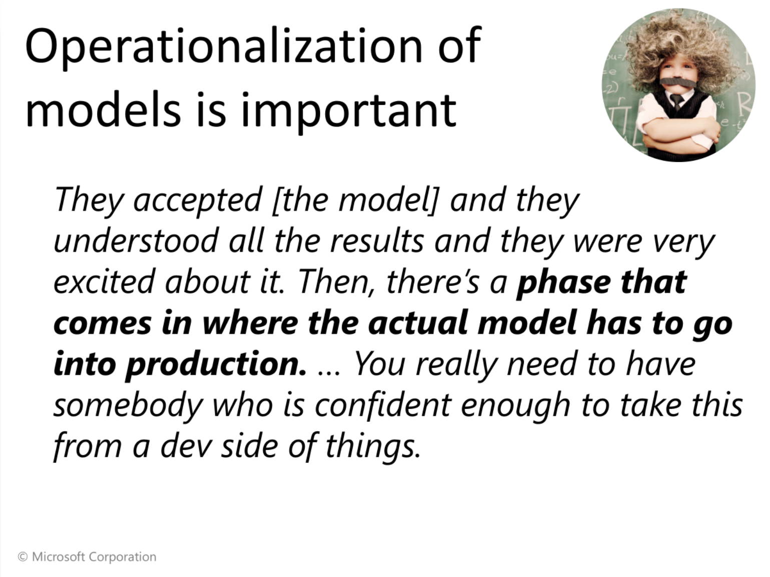
*(a) Is it a **priority** for the organization*

*(b) is it **actionable**, if I get an answer to this, is this something someone can do something with? and,*

*(c), are you as the feature team — if you're coming to me or if I'm going to you, telling you this is a good opportunity — are you **committing resources** to deliver a change?*

If those things are not true, then it's not worth us talking anymore.

© Microsoft Corporation

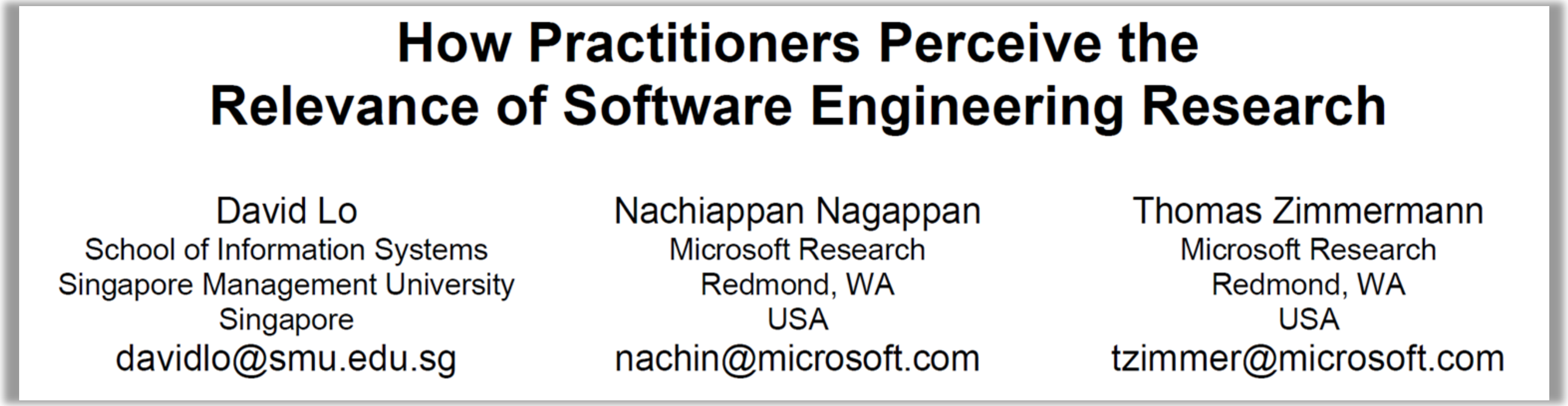


Operationalization of models is important

*They accepted [the model] and they understood all the results and they were very excited about it. Then, there's a **phase that comes in where the actual model has to go into production.** ... You really need to have somebody who is confident enough to take this from a dev side of things.*

© Microsoft Corporation

Going Back a Decade



How Practitioners Perceive the Relevance of Software Engineering Research

David Lo
School of Information Systems
Singapore Management University
Singapore
davidlo@smu.edu.sg

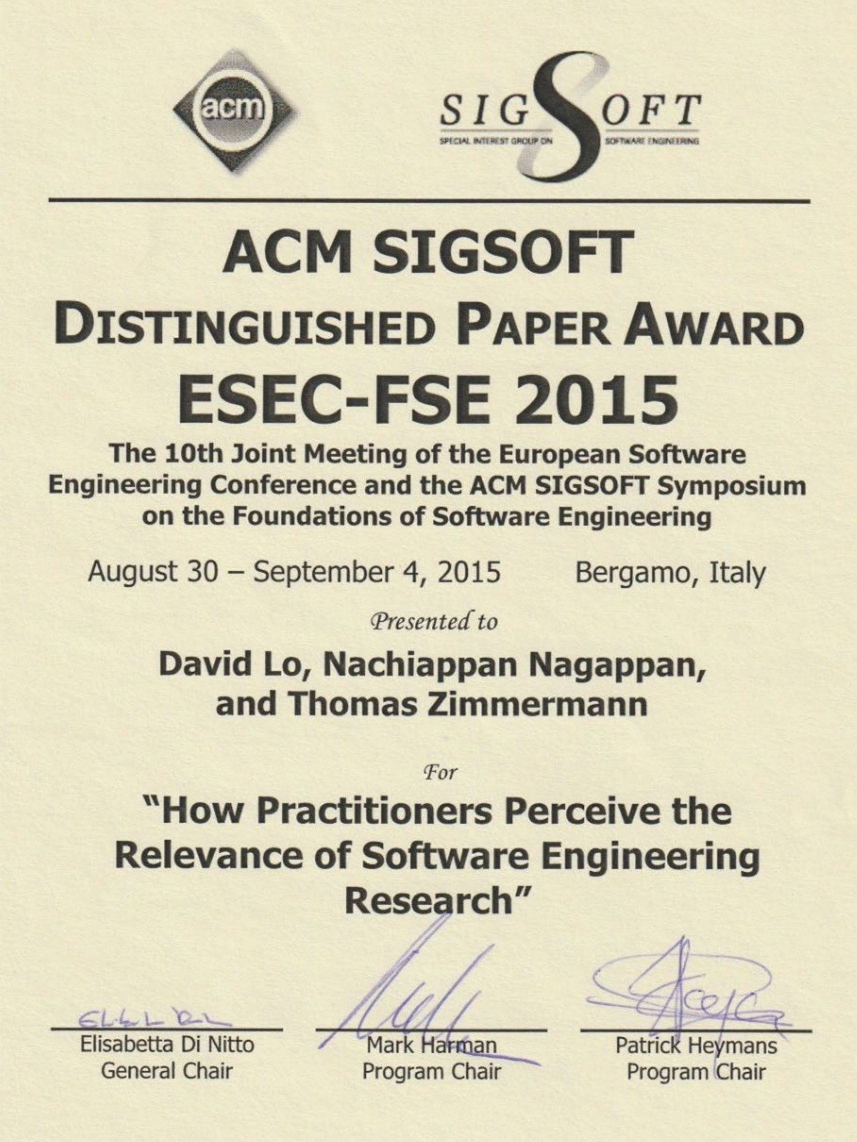
Nachiappan Nagappan
Microsoft Research
Redmond, WA
USA
nachin@microsoft.com

Thomas Zimmermann
Microsoft Research
Redmond, WA
USA
tzimmer@microsoft.com

10th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering

BERGAMO, ITALY, August 30 – September 4

Going Back a Decade



acm

SIGSOFT
SPECIAL INTEREST GROUP ON SOFTWARE ENGINEERING

ACM SIGSOFT

DISTINGUISHED PAPER AWARD

ESEC-FSE 2015

The 10th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering

August 30 – September 4, 2015 Bergamo, Italy

Presented to

David Lo, Nachiappan Nagappan, and Thomas Zimmermann

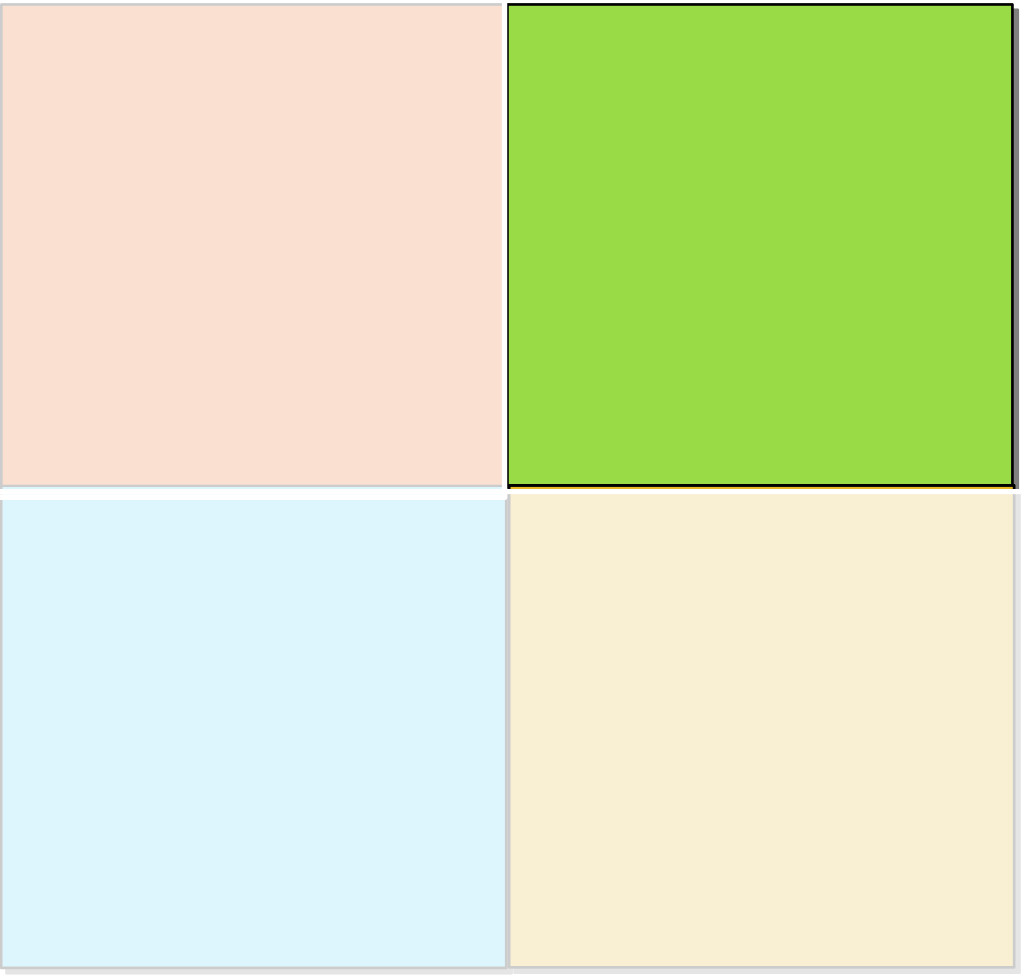
For

"How Practitioners Perceive the Relevance of Software Engineering Research"

Elisabetta Di Nitto
General Chair

Mark Harman
Program Chair

Patrick Heymans
Program Chair



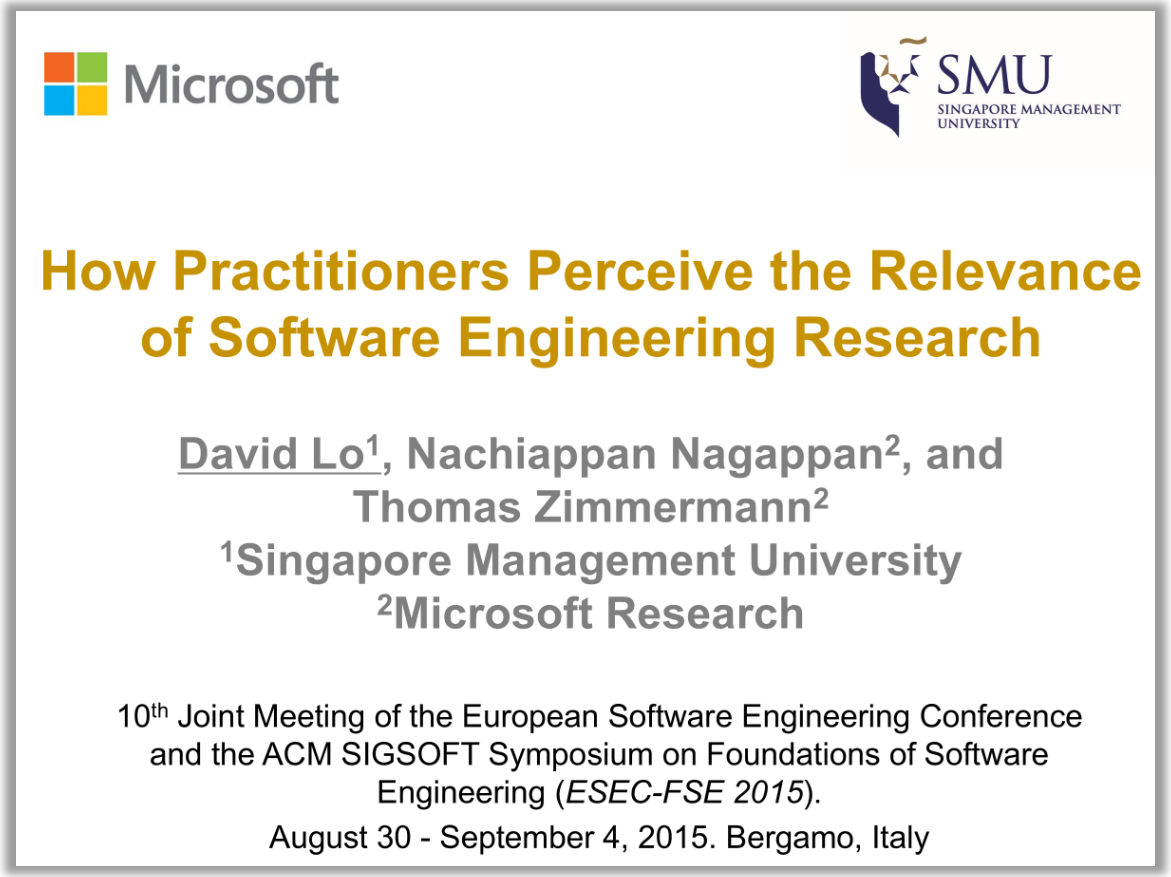
Going Back a Decade

What Is the Paper About?

How Has It Influenced Subsequent Studies?

What Is the Road Ahead?

What’s The Paper All About?



Microsoft

SMU SINGAPORE MANAGEMENT UNIVERSITY

How Practitioners Perceive the Relevance of Software Engineering Research

David Lo[1], Nachiappan Nagappan[2], and Thomas Zimmermann[2]
[1]Singapore Management University
[2]Microsoft Research

10th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on Foundations of Software Engineering (*ESEC-FSE 2015*).
August 30 - September 4, 2015. Bergamo, Italy

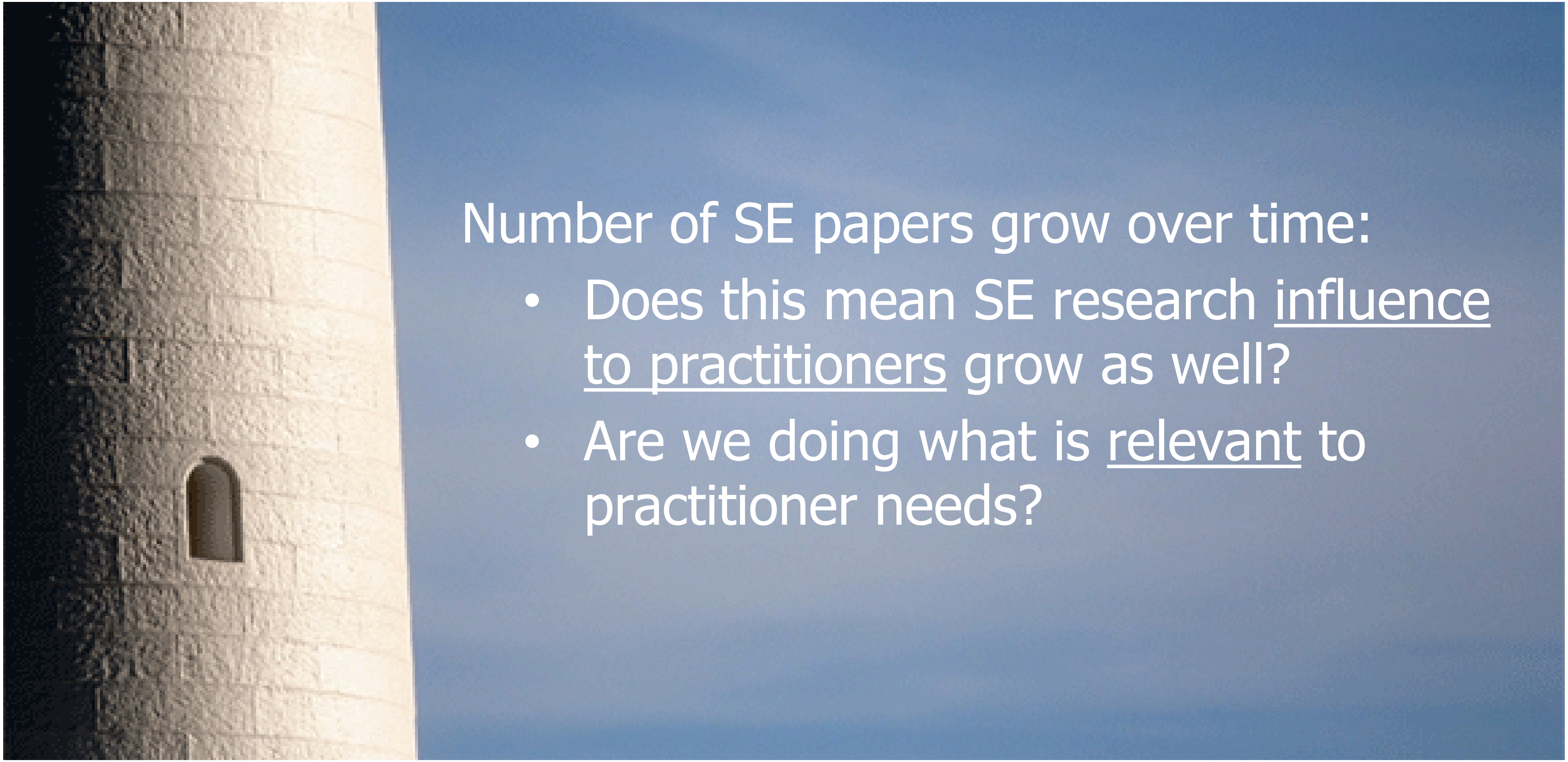
Microsoft Research

SMU SINGAPORE MANAGEMENT UNIVERSITY

Meta

THE UNIVERSITY OF CALIFORNIA • IRVINE • LET THERE BE LIGHT

Motivation: Relevance of SE Research



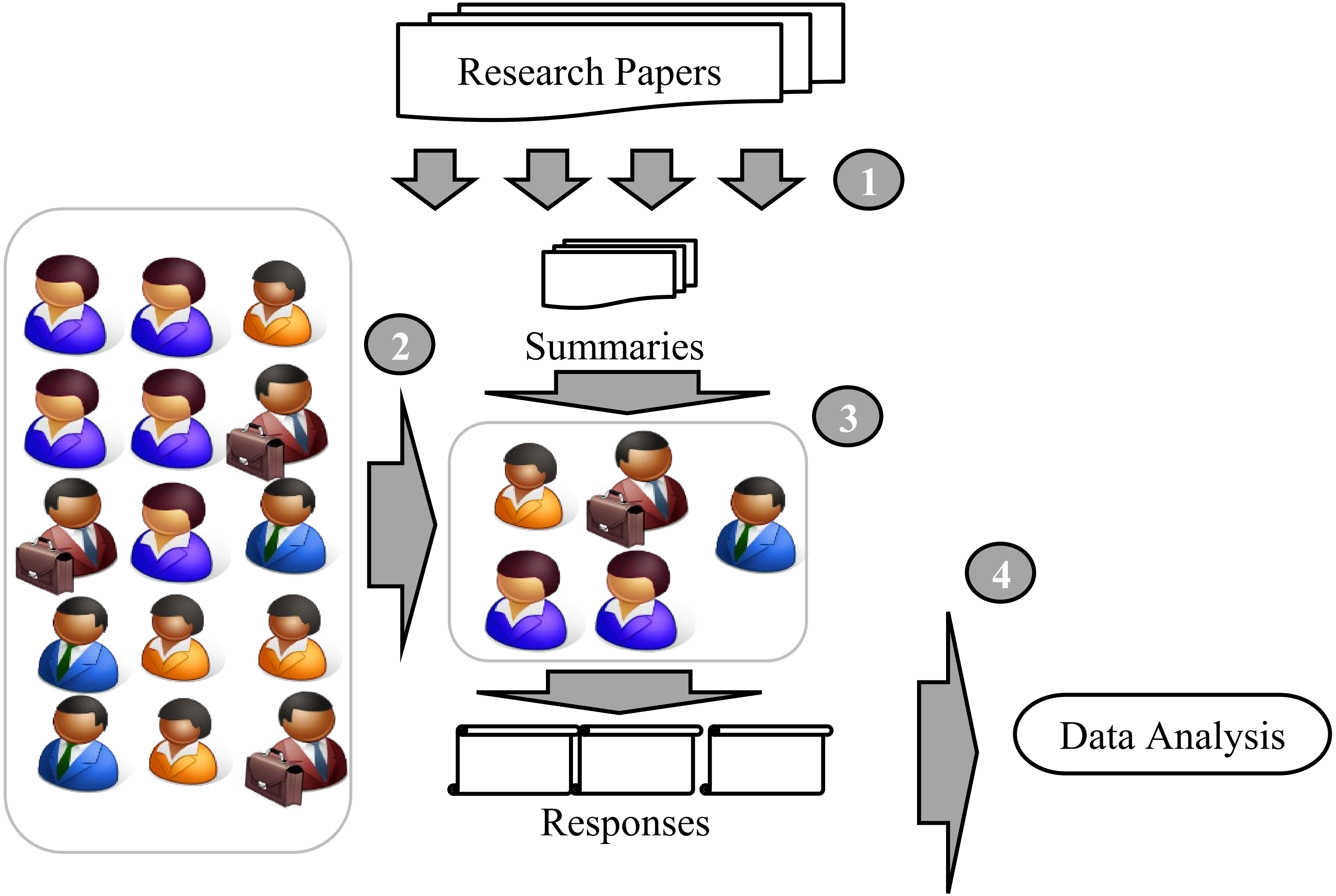
Number of SE papers grow over time:

- Does this mean SE research influence to practitioners grow as well?
- Are we doing what is relevant to practitioner needs?

Our Study

- Use practitioners as a sounding board of high-level research ideas
- Get practitioners feedback on the relevancy of software engineering studies from their perspectives
- Assess the degree-of-disconnect between researchers and practitioners
  - Health of software engineering research

Experimental Design

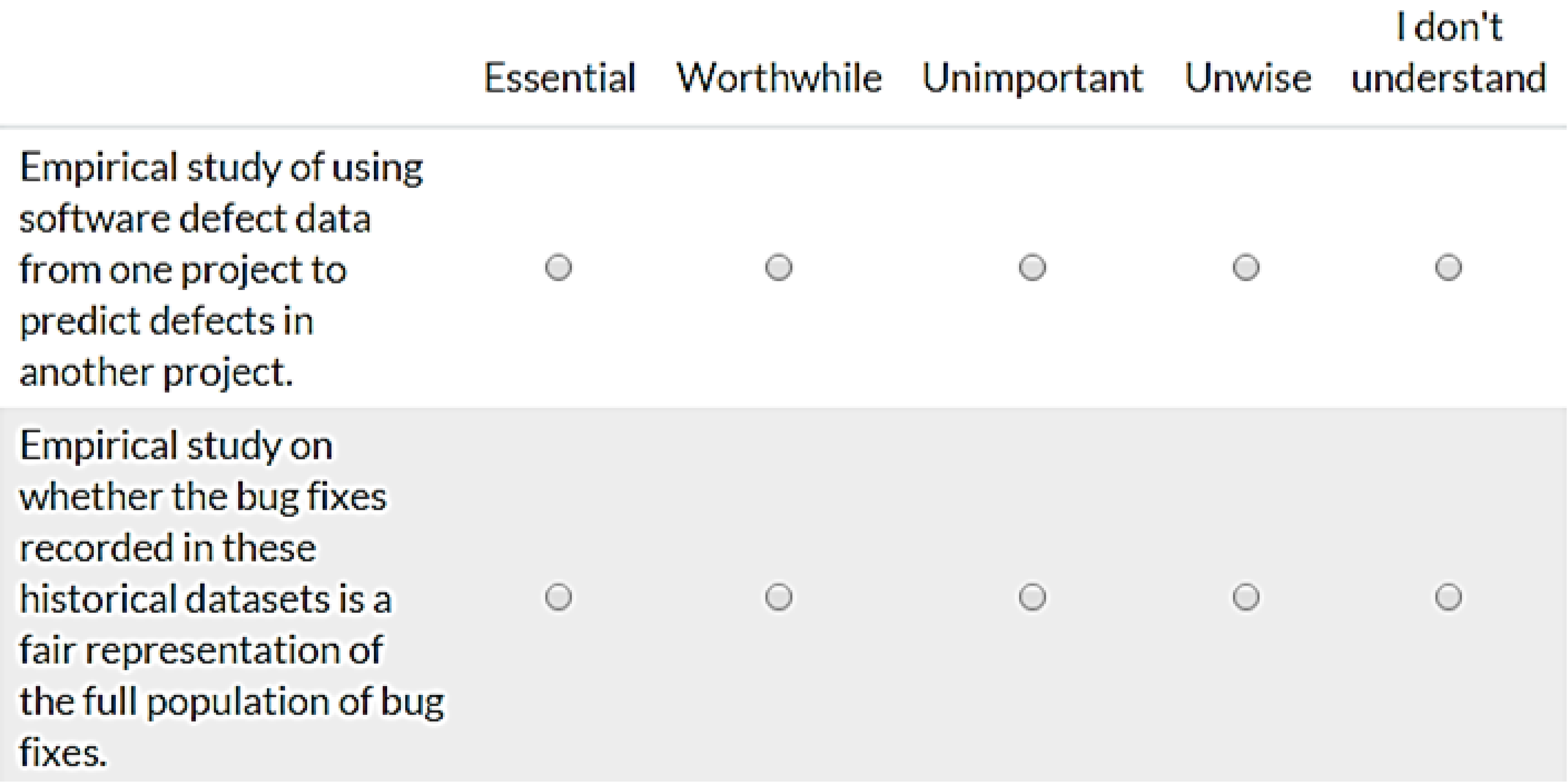


Survey

- Part I: Demographics
  - Primary work area: development, testing, PM
  - Role: individual contributor, lead, architect, manager, executive, other
  - Experience (in years)
  - CS or related degree/Not
  - Advanced degree/Not
- Part II: Relevance of SE research

Survey

In your opinion, how important are the following pieces of research? Please respond to as many as possible. (at least 1 response is required)*



	Essential	Worthwhile	Unimportant	Unwise	I don't understand
Empirical study of using software defect data from one project to predict defects in another project.	○	○	○	○	○
Empirical study on whether the bug fixes recorded in these historical datasets is a fair representation of the full population of bug fixes.	○	○	○	○	○

Survey

On the previous page, you selected the following research idea as “Unwise”:

“*Technique to identify files that contain a bug from a bug report.*”

To help us better understand your response, could you please explain why.

Response Statistics

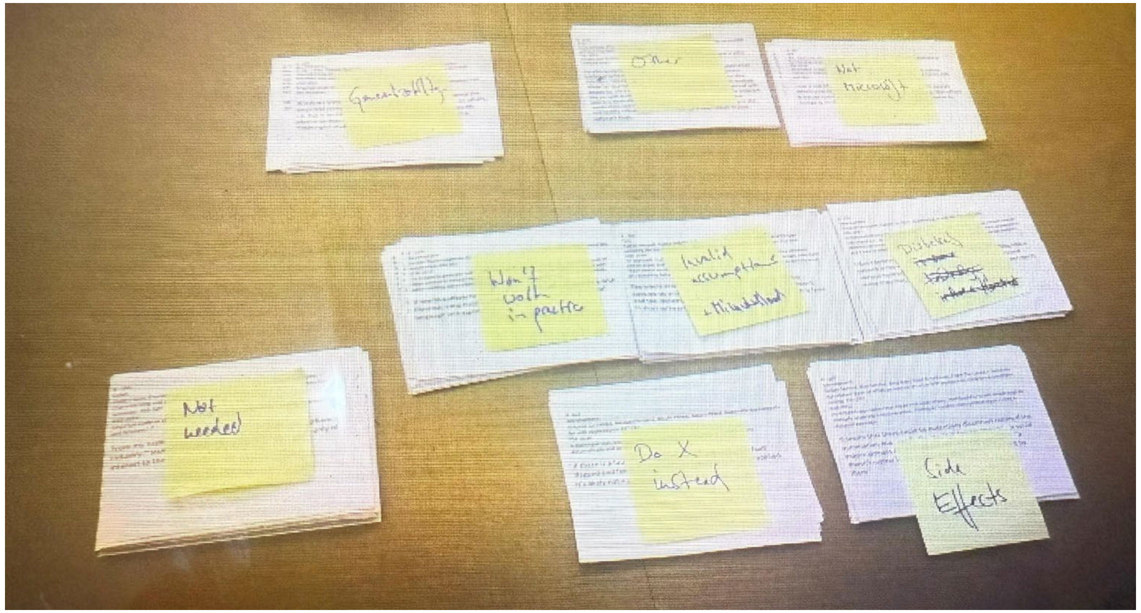
- Invite 3,000 randomly selected Microsoft practitioners working in technical roles
- 512 responded (17% response rate)
    - 291, 87, 102 are devs, testers, and PMs
- # of ratings: 17,913
    - 16-47 ratings per paper
- 173 provide reasons why they rate papers as unwise

Data Analysis: Scores

- E-Score: Proportion of ratings that are “Essential”
- EW-Score: Proportion of ratings that are “Essential” or “Worthwhile”
- U-Score: Proportion of ratings that are “Unwise”

Data Analysis: Open Card Sort

- Purpose: Create taxonomy from data
- Preparation:
  - A card for each “why unwise?” response
- Execution:
  - All authors discuss and sort the cards into meaningful groups with descriptive titles
  - Open
    - No predefined groups
    - Groups emerge and evolve during sorting



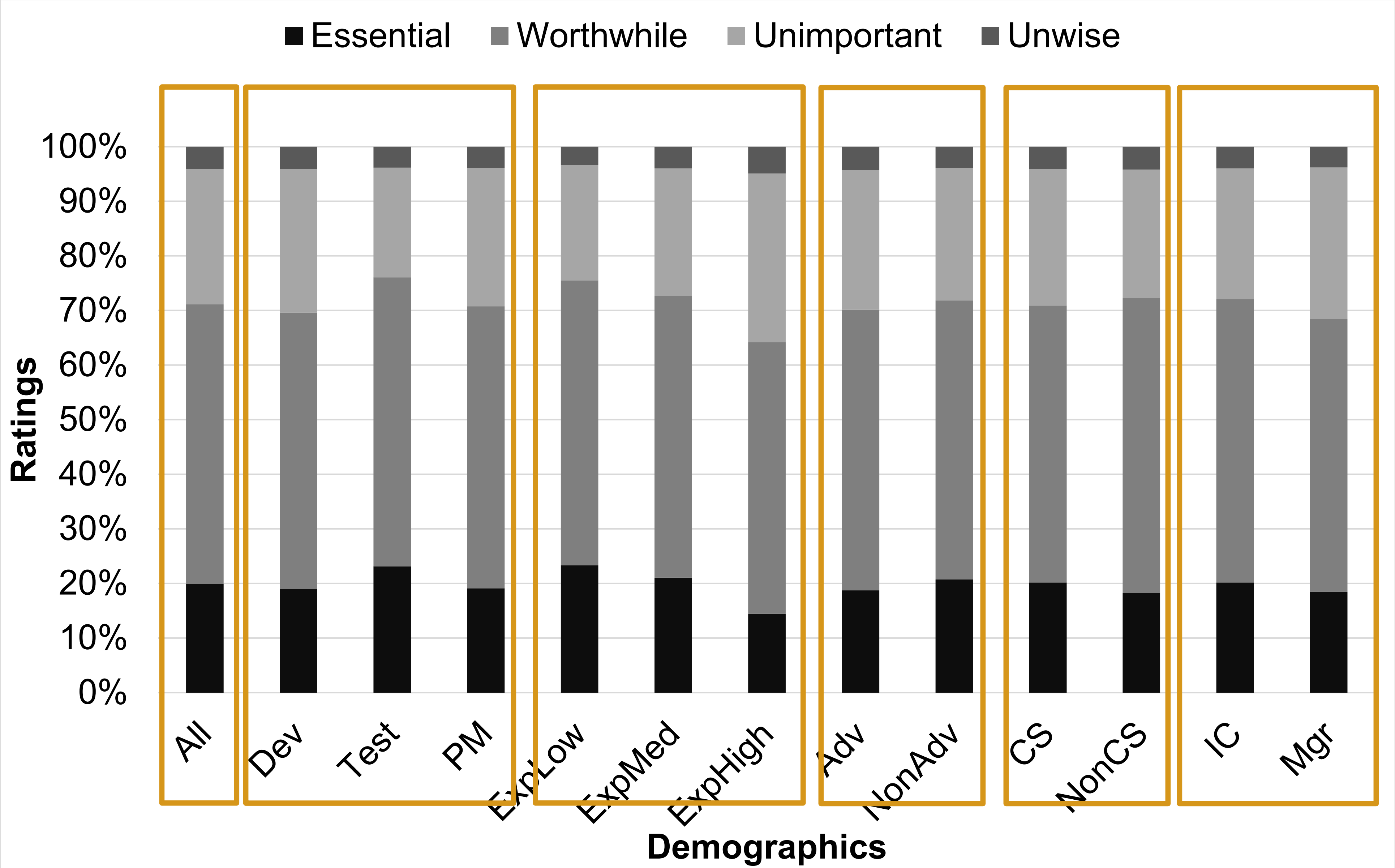
Research Questions

- RQ1: How do practitioners view software engineering research as a whole?
- RQ2: What research ideas do practitioners consider to be most important?
- RQ3: Why practitioners view some research ideas as unwise?

Findings

- RQ1: How do practitioners view software engineering research as a whole?
- RQ2: What research ideas do practitioners consider to be most important?
- RQ3: Why practitioners view some research ideas as unwise?

RQ1: Practitioner Perception



Findings

- RQ1: How do practitioners view software engineering research as a whole?
- RQ2: What research ideas do practitioners consider to be most important?
- RQ3: Why practitioners view some research ideas as unwise?

RQ2: Highly Rated Research

	Paper Summary	T	E	EW	U
P1	An approach to help developers identify and resolve conflicts early during collaborative software development, before those conflicts become severe and before relevant changes fade away in the developers' memories.	39	0.62	0.85	0.00
P2	Technique that clusters call stack traces to help performance analysts effectively discover highly impactful performance bugs (e.g., bugs impacting many users with long response delay).	30	0.60	1.00	0.00
P3	Symbolic analysis algorithm for buffer overflow detection that scale to millions of lines of code (MLOC) and can effectively handle loops and complex program structures.	29	0.55	0.97	0.03
P4	Automatic generation of efficient multithreaded random tests that effectively trigger concurrency bugs.	29	0.55	0.90	0.03
P5	Debugging tool that uses objects as key abstractions to support debugging operations. Instead of setting breakpoints that refer to source code, one sets breakpoints with reference to a particular object.	29	0.55	0.90	0.03

RQ2: Highly Rated Research

Devs:

- Performance
- Collaboration conflicts
- Debugging techniques
- Concurrency bugs

Testers:

- Monitoring
- Adaptive systems
- Traceability
- Lightweight verification

PMs:

- Agile teams
- Team awareness
- Product line
- Bug finding

Findings

- RQ1: How do practitioners view software engineering research as a whole?
- RQ2: What research ideas do practitioners consider to be most important?
- RQ3: Why practitioners view some research ideas as unwise?

Microsoft Research

SMU SINGAPORE MANAGEMENT UNIVERSITY

Meta

THE UNIVERSITY OF CALIFORNIA • IRVINE • LET THERE BE LIGHT

RQ3: Why Unwise?

- Reason 1: A tool is not needed

💬*“The tool that would result would not be something I would use or can imagine anyone else using”*

💬*“I don’t know how it could be used for daily work”*

💬*“I don’t believe that a framework will make the design and maintenance of such systems any easier”,*

💬*“The proposed tool is already available in the form of TFS or SharePoint list”*

RQ3: Why Unwise?

- Reason 2: An empirical study is not actionable

💬*"I wouldn't expect anything actionable or relevant to come out of this study"*

💬*"I don't see what's the value to study the difference between these two development (methodologies)"*

💬*"Don't see any need for this study since enough is known about common fallacies of this type",*

💬*"Don't know why there would be any benefit of knowing the answer to the proposed question", etc.*

RQ3: Why Unwise?

- Reason 3: Generalizability issue

  - *“Empirical study on this platforms may not be reusable on others”*
  - *“Case study for a project is always less useful than researching around a topic. Lessons learned from one project can be very specific to this project”*
  - *“Might want to consider bugs in same applications over different platforms”*
  - *“Developers are not alike”*

RQ3: Why Unwise?

Reason 3: Generalizability Issue - Scalability

- *“I don’t see this being used for large-scale systems”*

- *“For a complex program, there will be too much info, and the developer will not be able to understand”*

- *“The set of software update that needs testing is not a small number and new software updates happen almost every week. And it is not the same set of software installed by different users”*

- *“Energy consumption characteristics will vary from device to device and over time”*

- *“As the complexity of the bug goes up, the solution may or may not go up”*

RQ3: Why Unwise?

- Reason 4: Cost outweighs benefit

🗨*“Huge time investment for little return”*

🗨*“I believe the cost of implementing and maintain such a solution would be greater than the cost of developers fixing bugs manually”*

🗨*“Development cost of this approach will overkill the gain it gives”*

🗨*“I have experience with similar systems and I’ve never seen one where I thought they were of net-value”*

RQ3: Why Unwise?

- Reason 5: Questionable assumptions about inputs or conditions

🗨*"The whole research assumes that there are requirement documents and design documents in software development… which is false in most software projects nowadays"*

🗨*"Such a tool makes it easier for people to focus on test coverage & state coverage. Which doesn't really help detect bugs"*

🗨*"Description is often not filled correctly. hence it is unwise to rely on it"*

🗨*"Analyzing documentation written by humans seems inherently risky. Engineers are not known for writing good documentation, and I suspect that will only get worse as we accelerate our deliverables"*

RQ3: Why Unwise?

- Reason 6: Disbelief in a particular technology or methodology

- *“I don’t believe in design patterns, force fitting something into a pattern is not wise”*
- *“UML is half dead!”*
- *“I don’t think UML is a good tool to use in the development process”*

RQ3: Why Unwise?

- Reason 7: Another solution seems better

🗨*“Making yet another language isn’t really solving anything. Instead, give me more functionality within my language and/or give me tools to do these types of things”*

🗨*“Better organization of how Linux is packaged and distributed would solve this issue without the need of deep analysis and investigations”*

🗨*“Not sure if this is the best or the easiest way to find new uses. Usually I look at forums/books/tools for that”*

🗨*“I don’t think natural language is that important. Instead helping users find the keywords or tags is should be the focus”*

Microsoft Research

SMU SINGAPORE MANAGEMENT UNIVERSITY

Meta

THE UNIVERSITY OF CALIFORNIA IRVINE LET THERE BE LIGHT

RQ3: Why Unwise?

- Reason 8: Proposed solution has side effects

- *“Design Patterns … derive their flexibility at the expense of comprehensibility of the interacting parts of a system”*
- *“Specific techniques to rank devs can lead to devs not working together and lower productivity/morale”*
- *“Drag and drop solutions have always seemed to me as a quick and easy way to write inefficient code”*

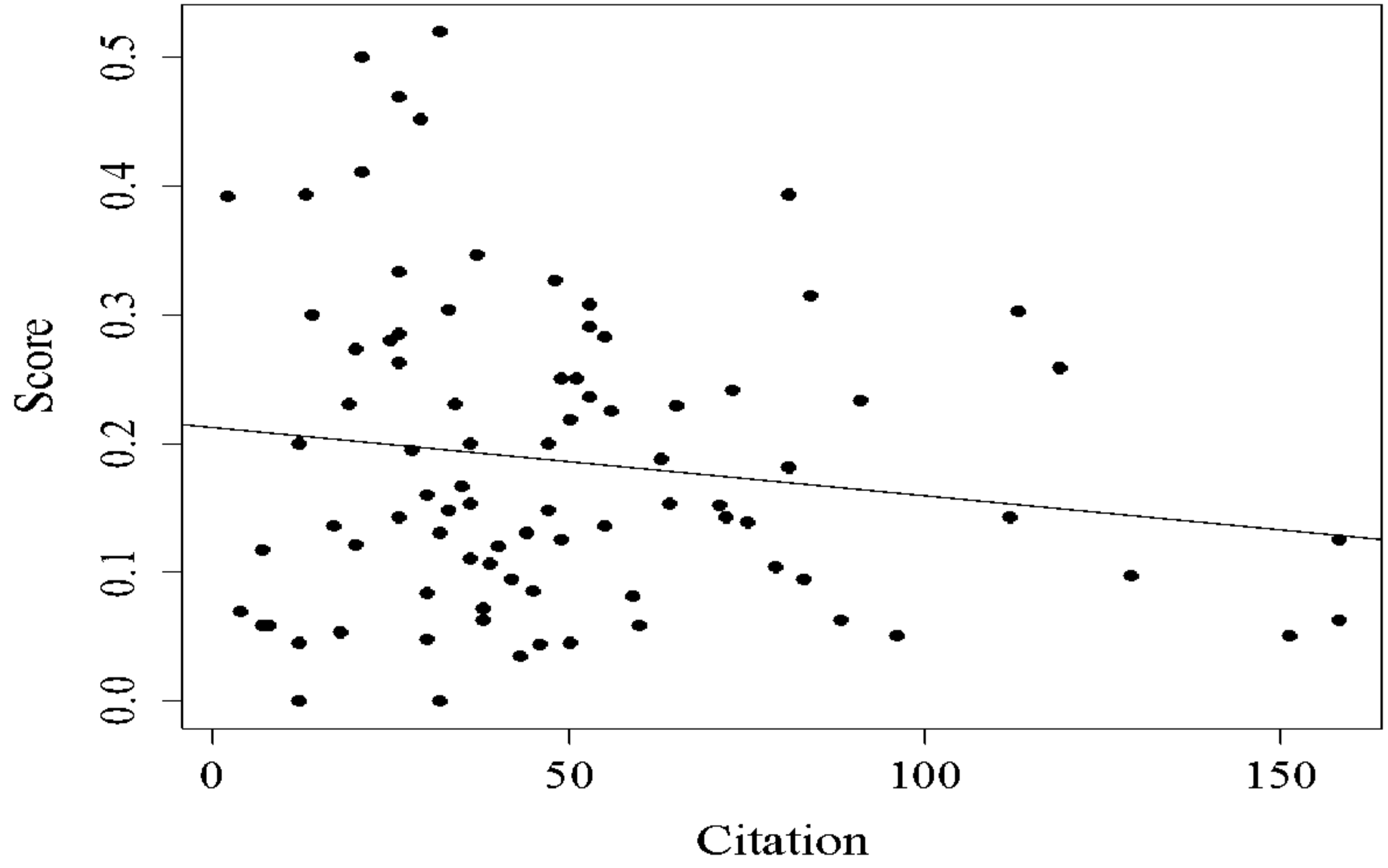
Limitations

- Summaries might not be the best ones possible
- Only Microsoft practitioners participate in this study

Caveats

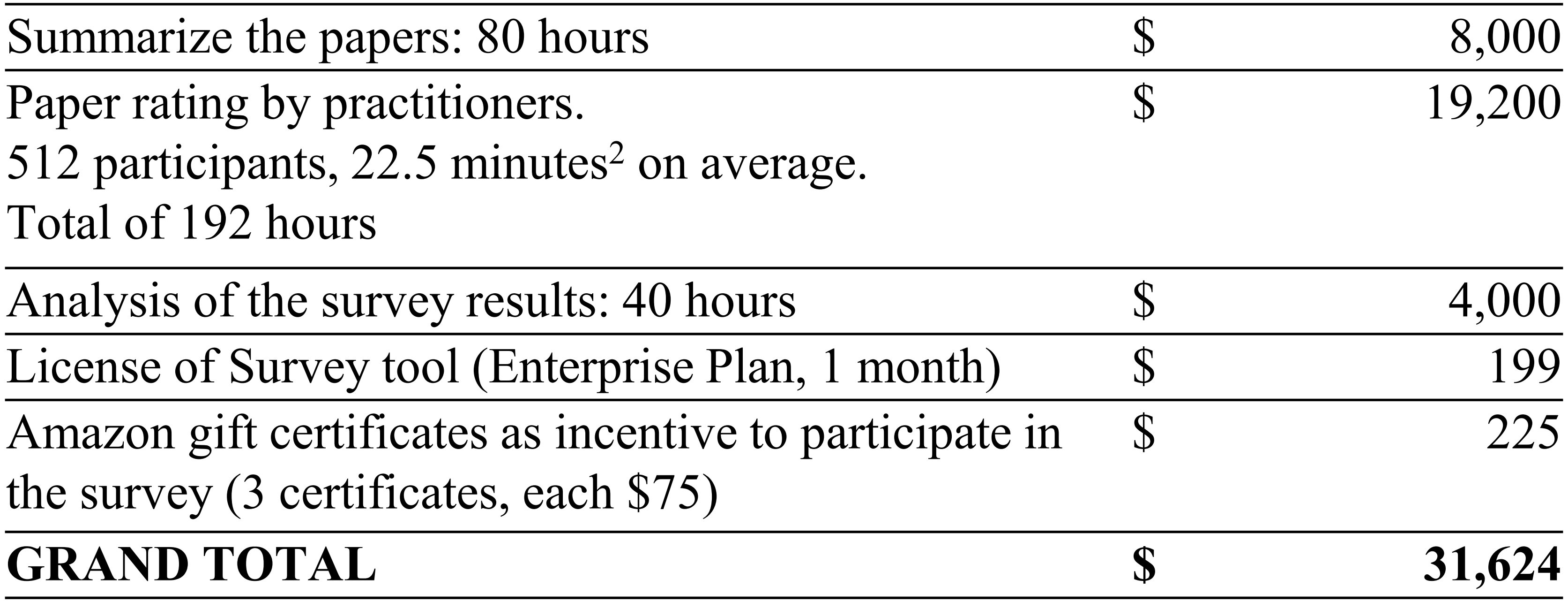
- Practitioners can be wrong
- We are measuring *relevance* rather than *adoptability* or *adoption*

Discussion: Citation vs. E-Score



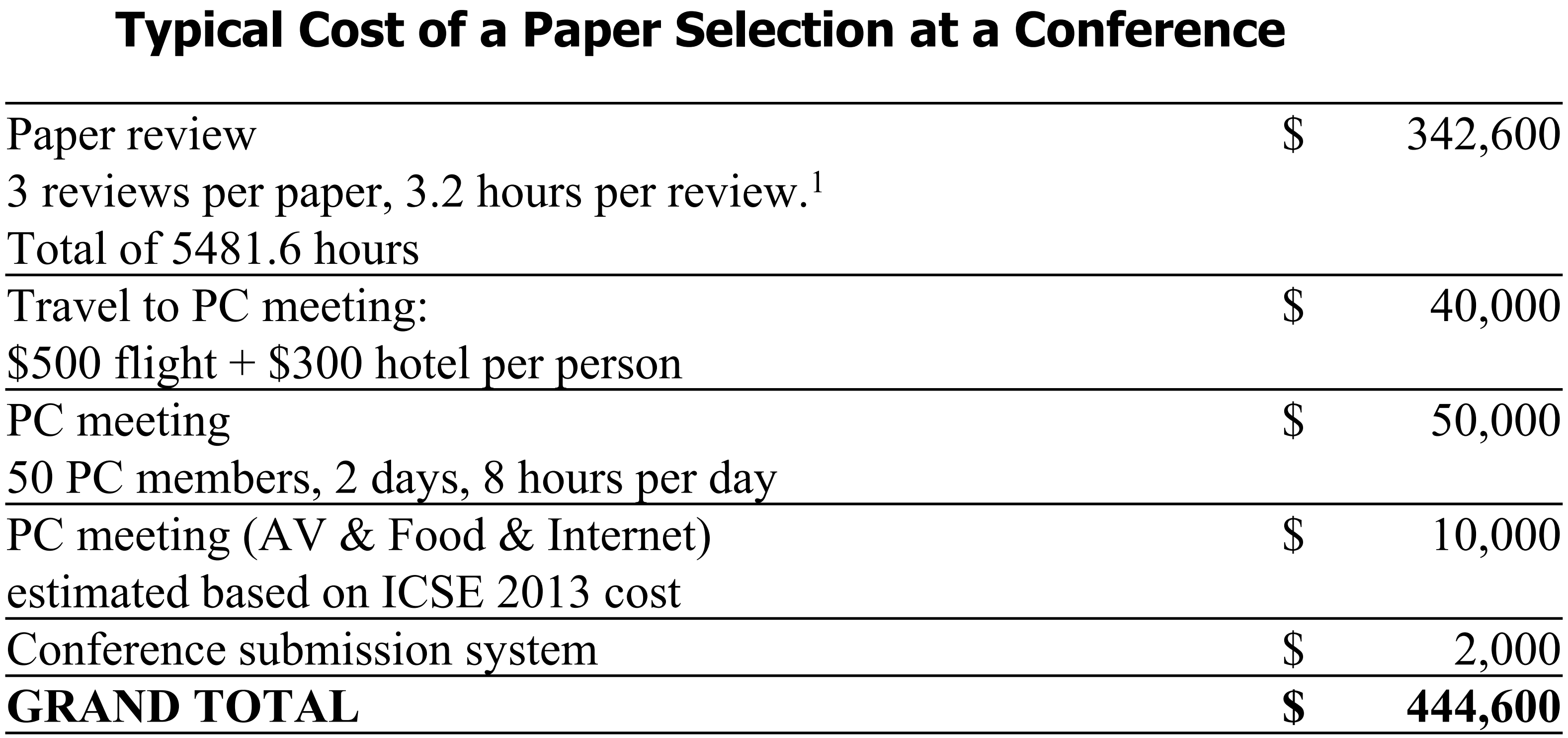
Discussion: Lightweight Assessment

Cost of This Study



Summarize the papers: 80 hours	$	8,000
Paper rating by practitioners. 512 participants, 22.5 minutes[2] on average. Total of 192 hours	$	19,200
Analysis of the survey results: 40 hours	$	4,000
License of Survey tool (Enterprise Plan, 1 month)	$	199
Amazon gift certificates as incentive to participate in the survey (3 certificates, each $75)	$	225
GRAND TOTAL	**$**	**31,624**

Discussion: Lightweight Assessment



Typical Cost of a Paper Selection at a Conference

Paper review 3 reviews per paper, 3.2 hours per review.[1] Total of 5481.6 hours	$	342,600
Travel to PC meeting: $500 flight + $300 hotel per person	$	40,000
PC meeting 50 PC members, 2 days, 8 hours per day	$	50,000
PC meeting (AV & Food & Internet) estimated based on ICSE 2013 cost	$	10,000
Conference submission system	$	2,000
GRAND TOTAL	**$**	**444,600**

Summary of Findings - I

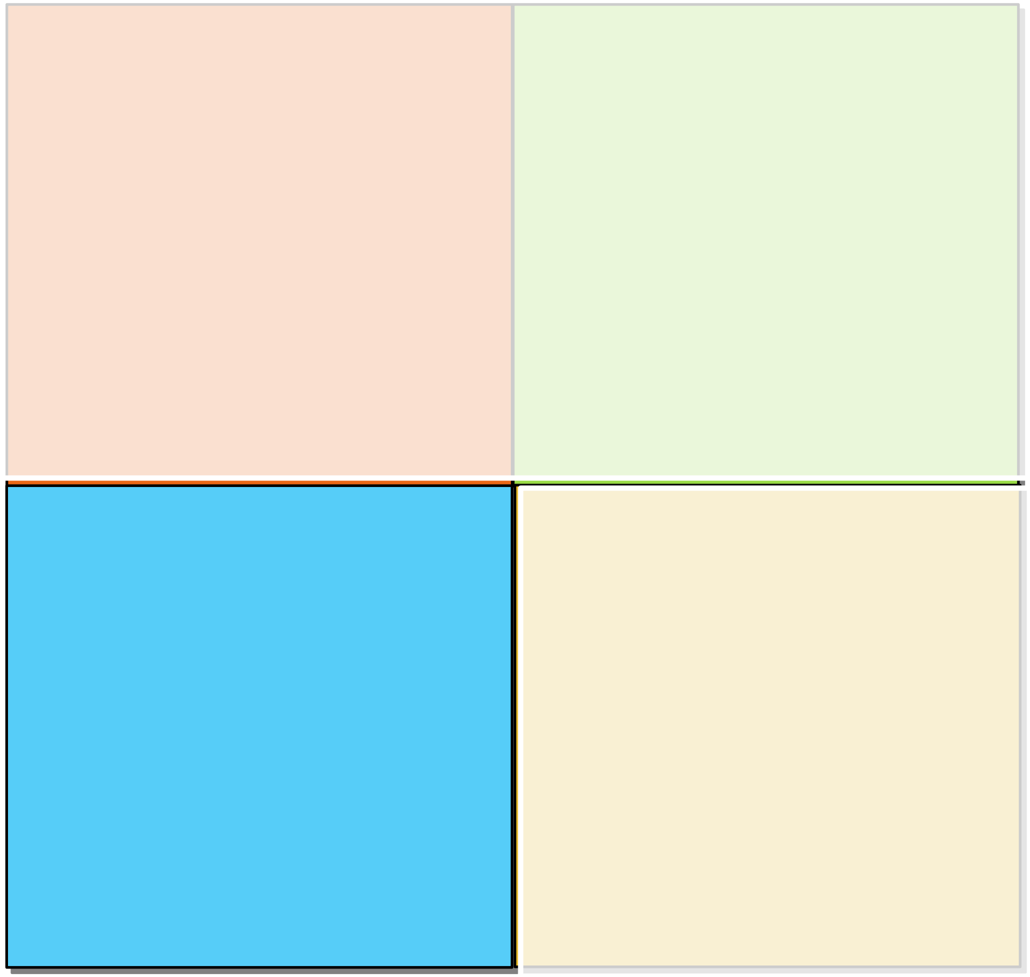
- Practitioners are generally positive
- Topics that interest them include:
  - Collaboration conflict detection
  - Improving system performance
  - Debugging tools
  - Adaptive systems
  - Testing multi-threaded programs
  - Etc.

Summary of Findings - II

- Threats to relevance of SE research:
  - A tool is not needed
  - An empirical study is not actionable
  - Generalizability issue
  - Cost outweighs benefit
  - Questionable assumptions
  - Disbelief in a particular technology or methodology
  - Another solution/problem seems better/more important
  - Proposed solution has side effects

Future Work

- Replicate our study on other companies that are based in various countries.
- Replicate our study on open-source developers.
- Collaborate with conferences to continuously replicate these studies in the future.



Going Back a Decade

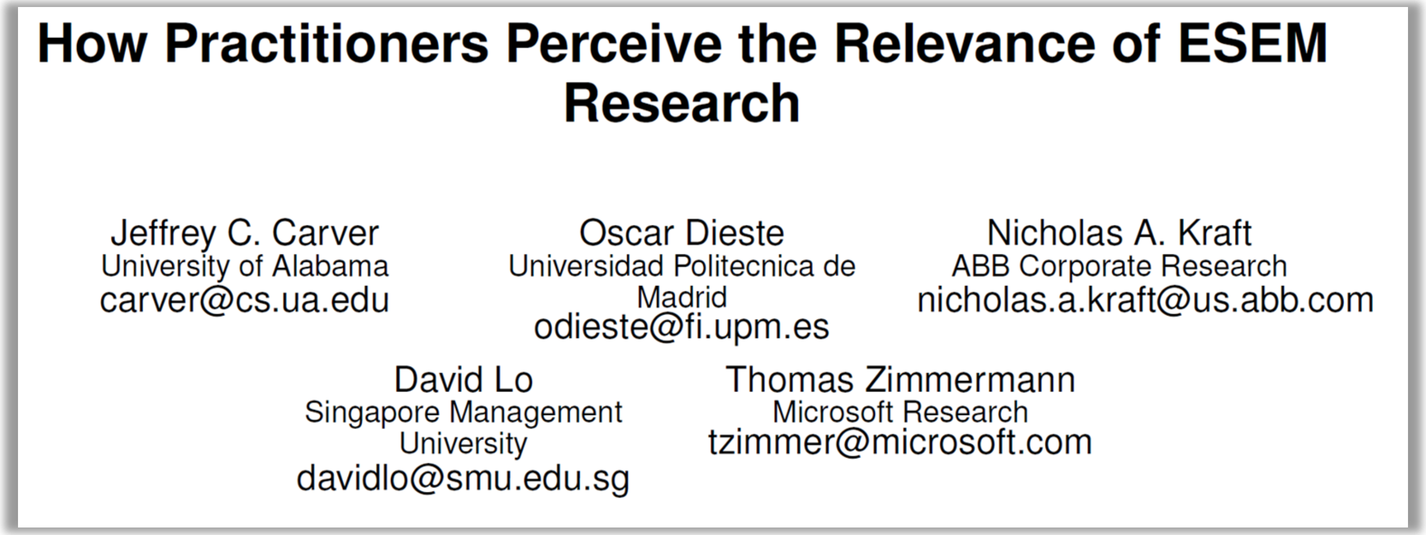
What Is the Paper About?

How Has It Influenced Subsequent Studies?

What Is the Road Ahead?

Similar Effort in SE Sub Communities

ESEM 2016



How Practitioners Perceive the Relevance of ESEM Research

Jeffrey C. Carver
University of Alabama
carver@cs.ua.edu

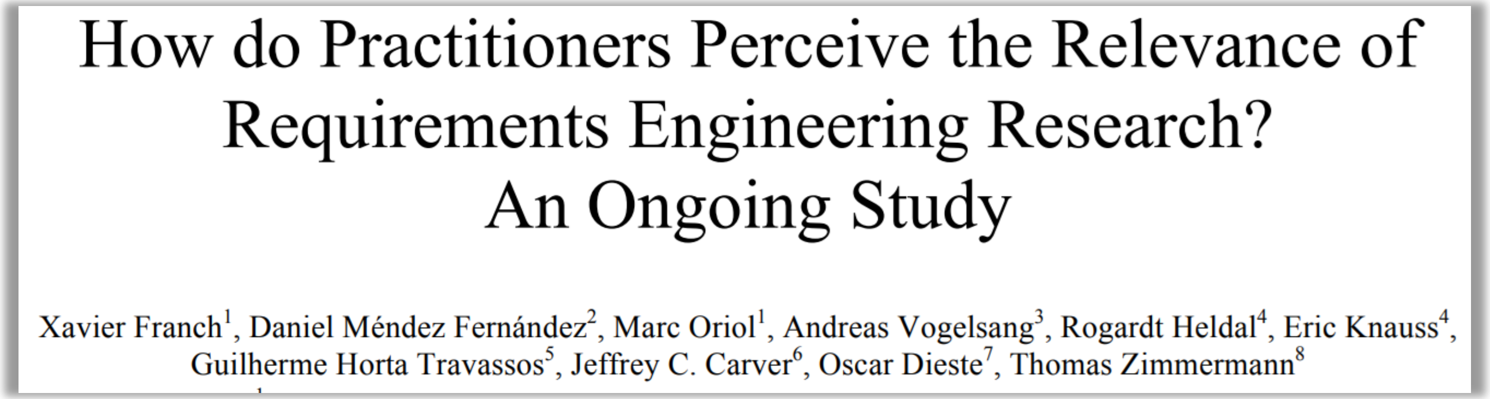
Oscar Dieste
Universidad Politecnica de Madrid
odieste@fi.upm.es

Nicholas A. Kraft
ABB Corporate Research
nicholas.a.kraft@us.abb.com

David Lo
Singapore Management University
davidlo@smu.edu.sg

Thomas Zimmermann
Microsoft Research
tzimmer@microsoft.com

RE 2017

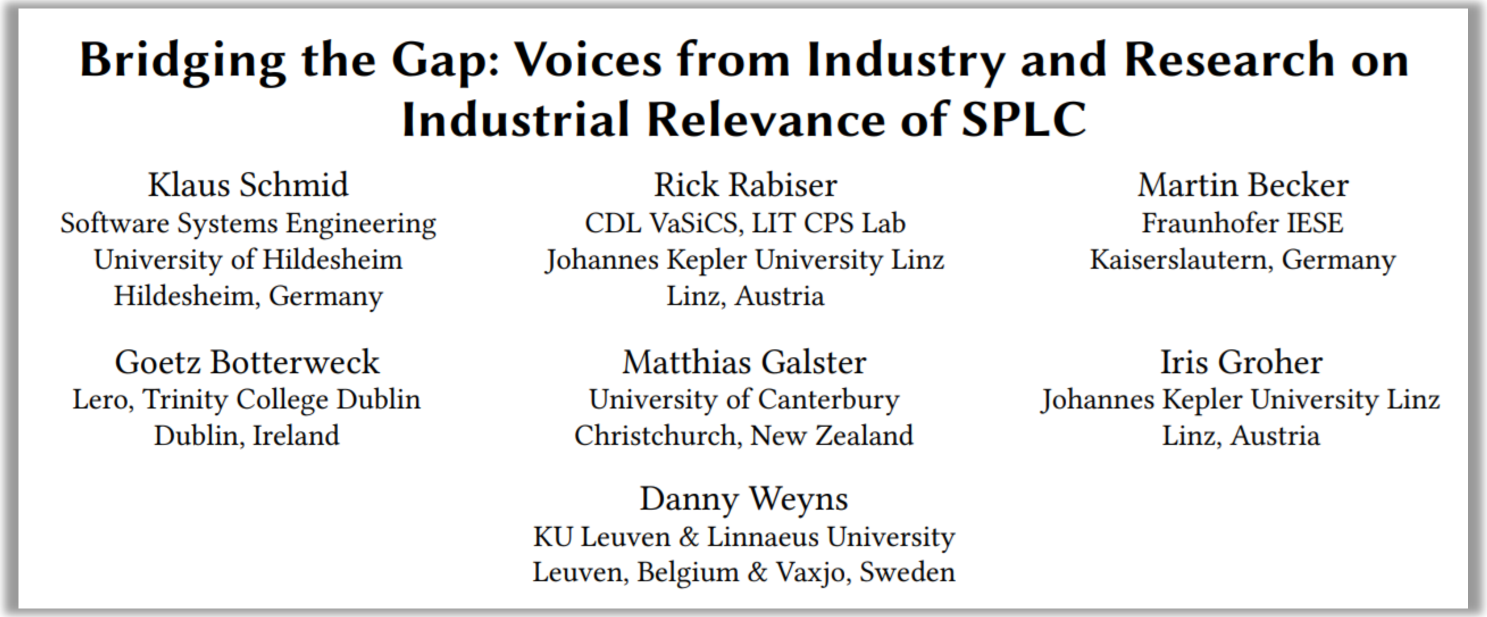


How do Practitioners Perceive the Relevance of Requirements Engineering Research? An Ongoing Study

Xavier Franch[1], Daniel Méndez Fernández[2], Marc Oriol[1], Andreas Vogelsang[3], Rogardt Heldal[4], Eric Knauss[4], Guilherme Horta Travassos[5], Jeffrey C. Carver[6], Oscar Dieste[7], Thomas Zimmermann[8]

Similar Effort in SE Sub Communities

SPLC 2021



Bridging the Gap: Voices from Industry and Research on Industrial Relevance of SPLC

Klaus Schmid
Software Systems Engineering
University of Hildesheim
Hildesheim, Germany

Rick Rabiser
CDL VaSiCS, LIT CPS Lab
Johannes Kepler University Linz
Linz, Austria

Martin Becker
Fraunhofer IESE
Kaiserslautern, Germany

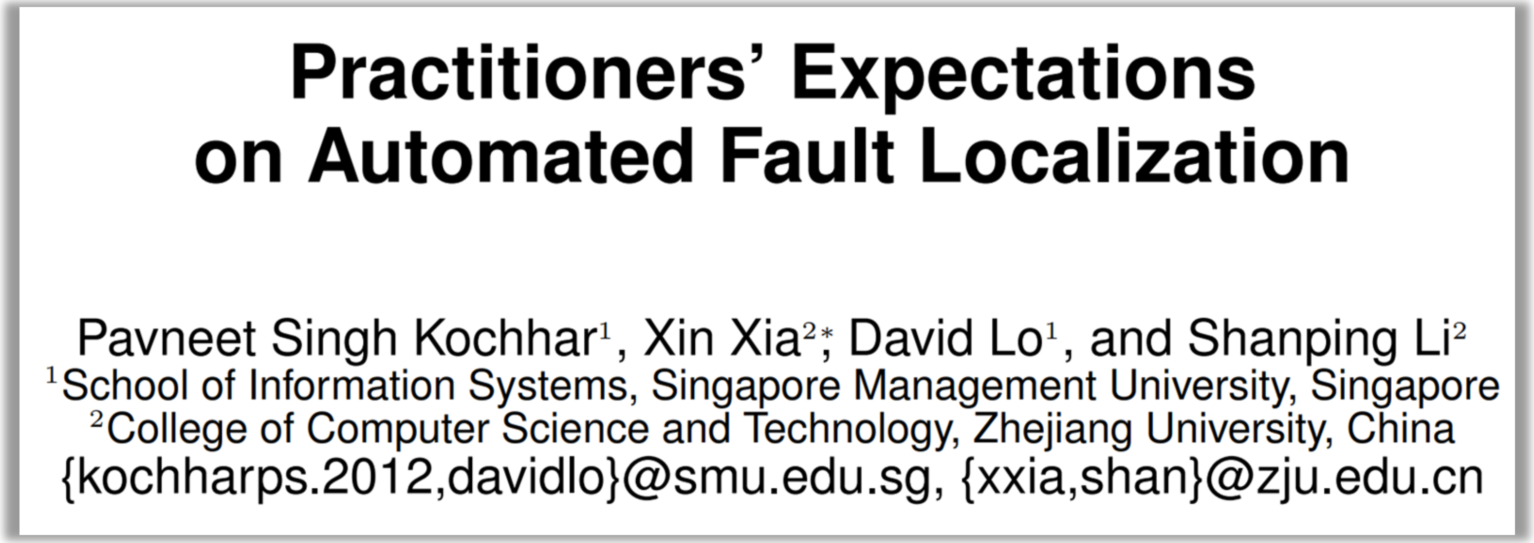
Goetz Botterweck
Lero, Trinity College Dublin
Dublin, Ireland

Matthias Galster
University of Canterbury
Christchurch, New Zealand

Iris Groher
Johannes Kepler University Linz
Linz, Austria

Danny Weyns
KU Leuven & Linnaeus University
Leuven, Belgium & Vaxjo, Sweden

Continuation Effort within Our Research Groups



Practitioners' Expectations on Automated Fault Localization

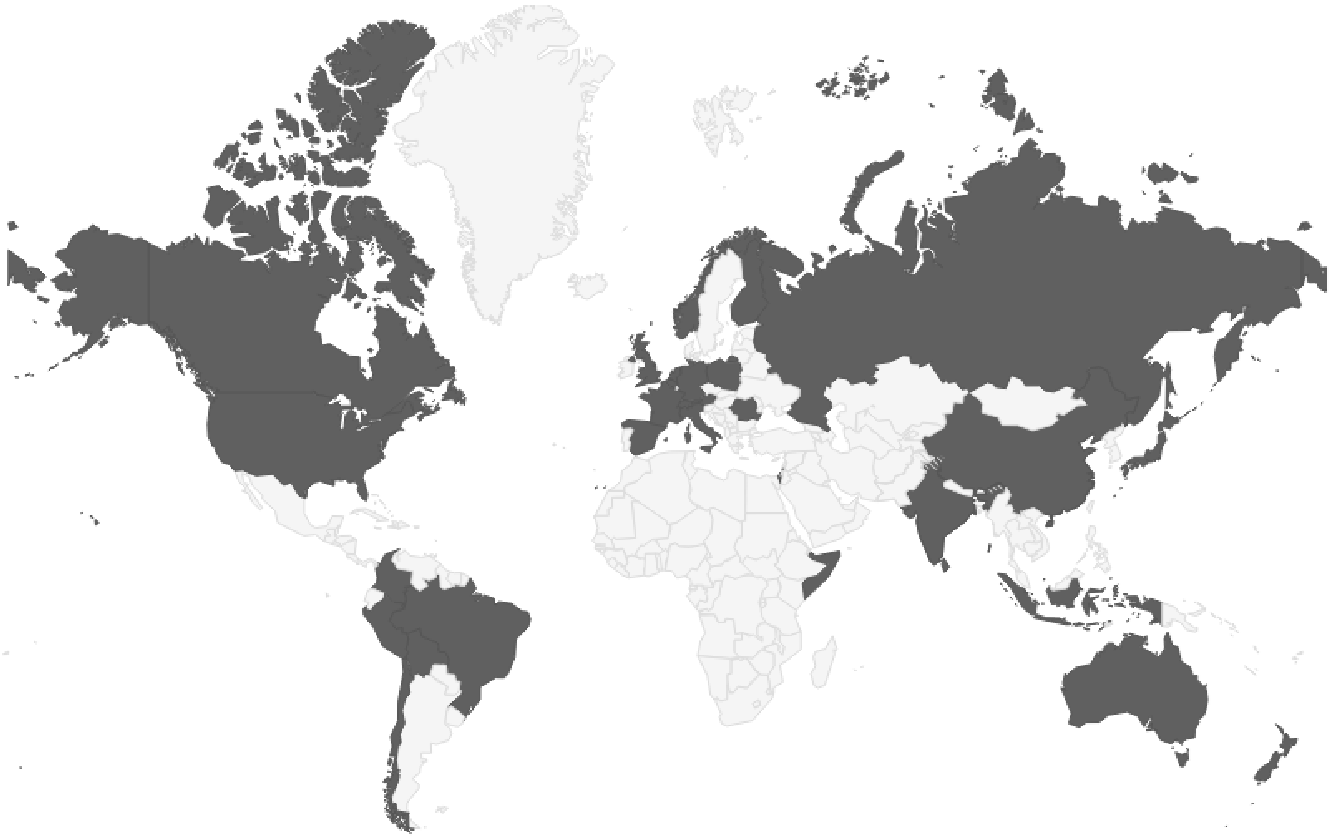
Pavneet Singh Kochhar[1], Xin Xia[2]*, David Lo[1], and Shanping Li[2]
[1]School of Information Systems, Singapore Management University, Singapore
[2]College of Computer Science and Technology, Zhejiang University, China
{kochharps.2012,davidlo}@smu.edu.sg, {xxia,shan}@zju.edu.cn

ISSTA 2016
400+ citations

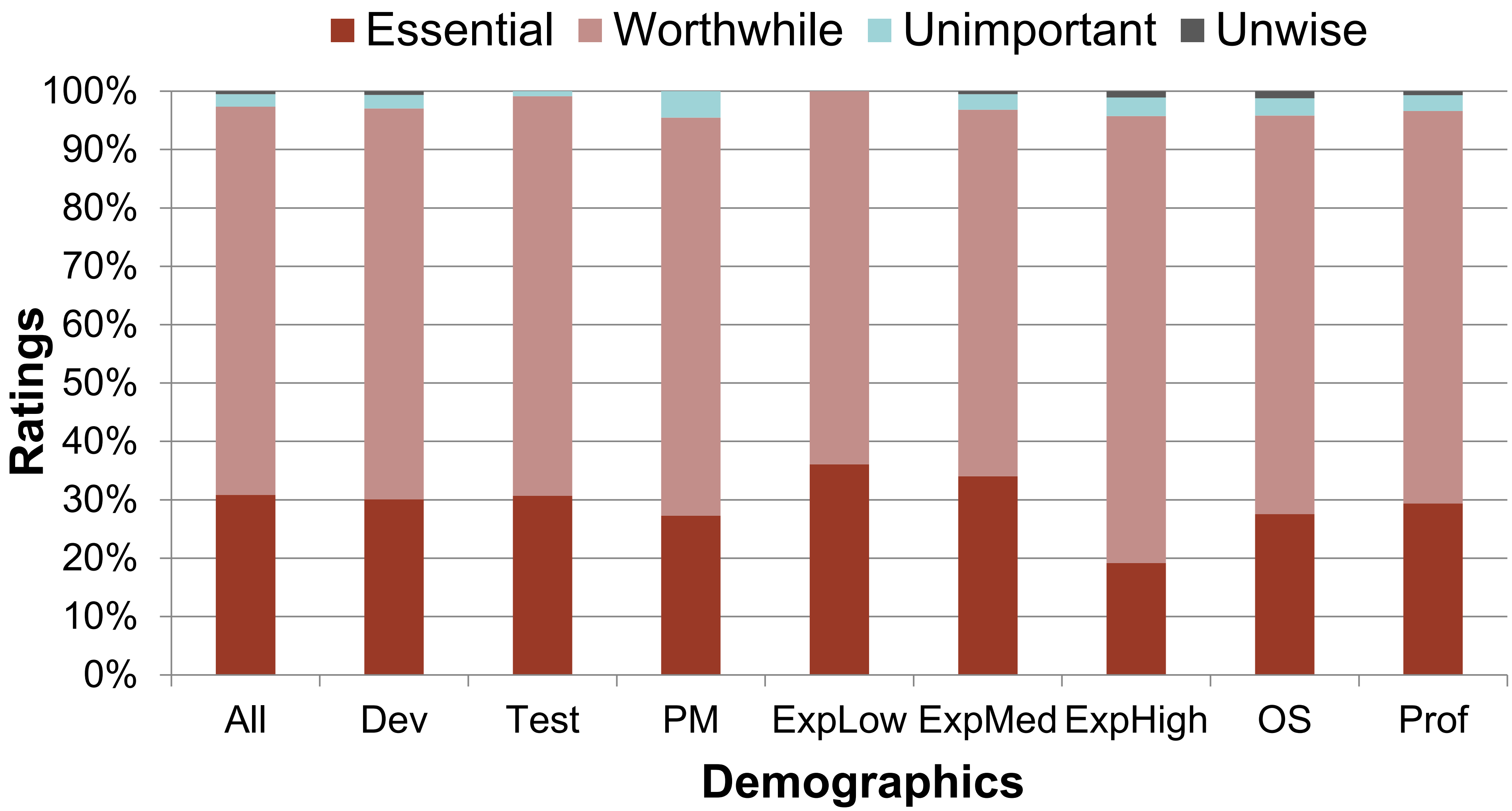
- General vs. specific
- Perceptions *to* Expectations
  - FSE'15: Essential vs. …vs. unwise
  - ISSTA'16: Adoption thresholds & factors to consider
- Beyond Microsoft

Survey Demographics

- 386 responses
- 33 countries



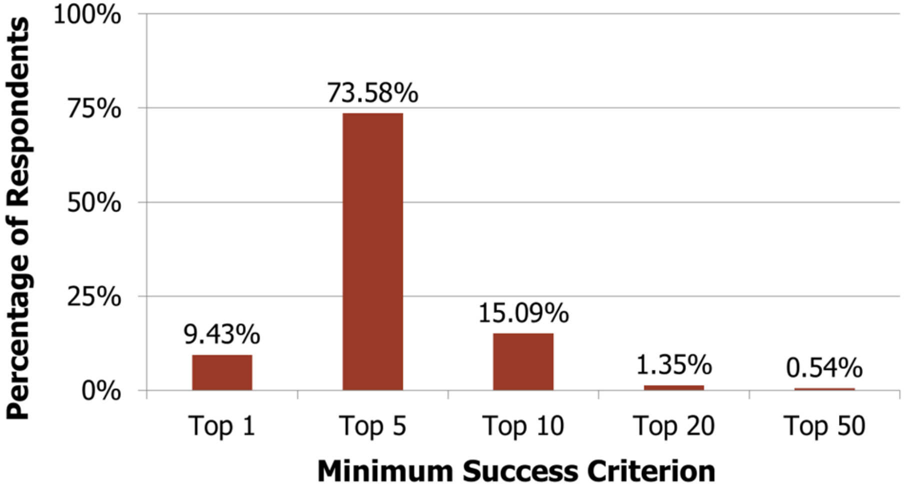
RQ1: Importance of Fault Localization



Fisher's Exact Test = p-values < 0.05

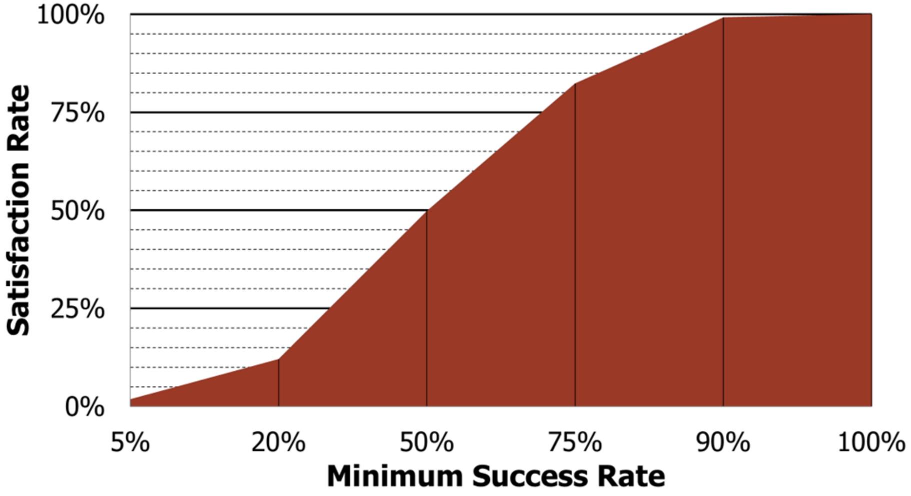
RQ4: Minimum Success Criterion

Position of the buggy element in returned list



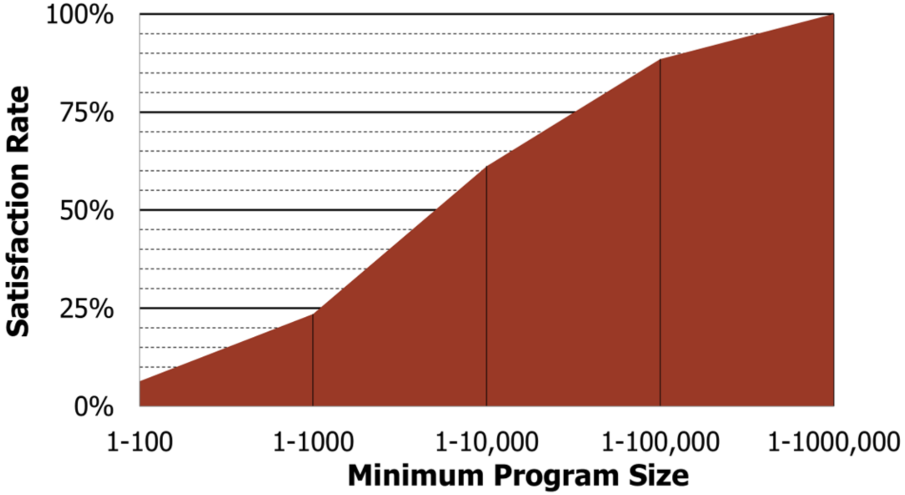
RQ5: Trustworthiness

Proportion of times a technique works.



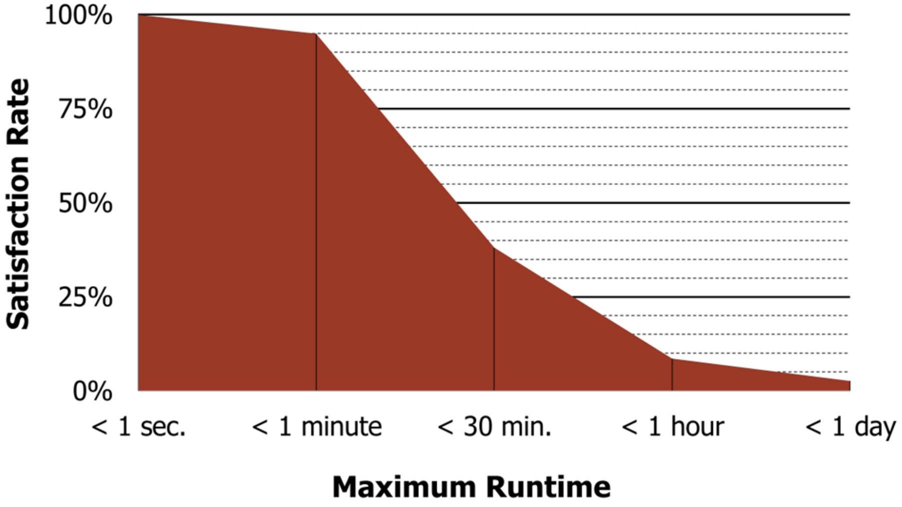
RQ6: Scalability

Program sizes a technique can work on.

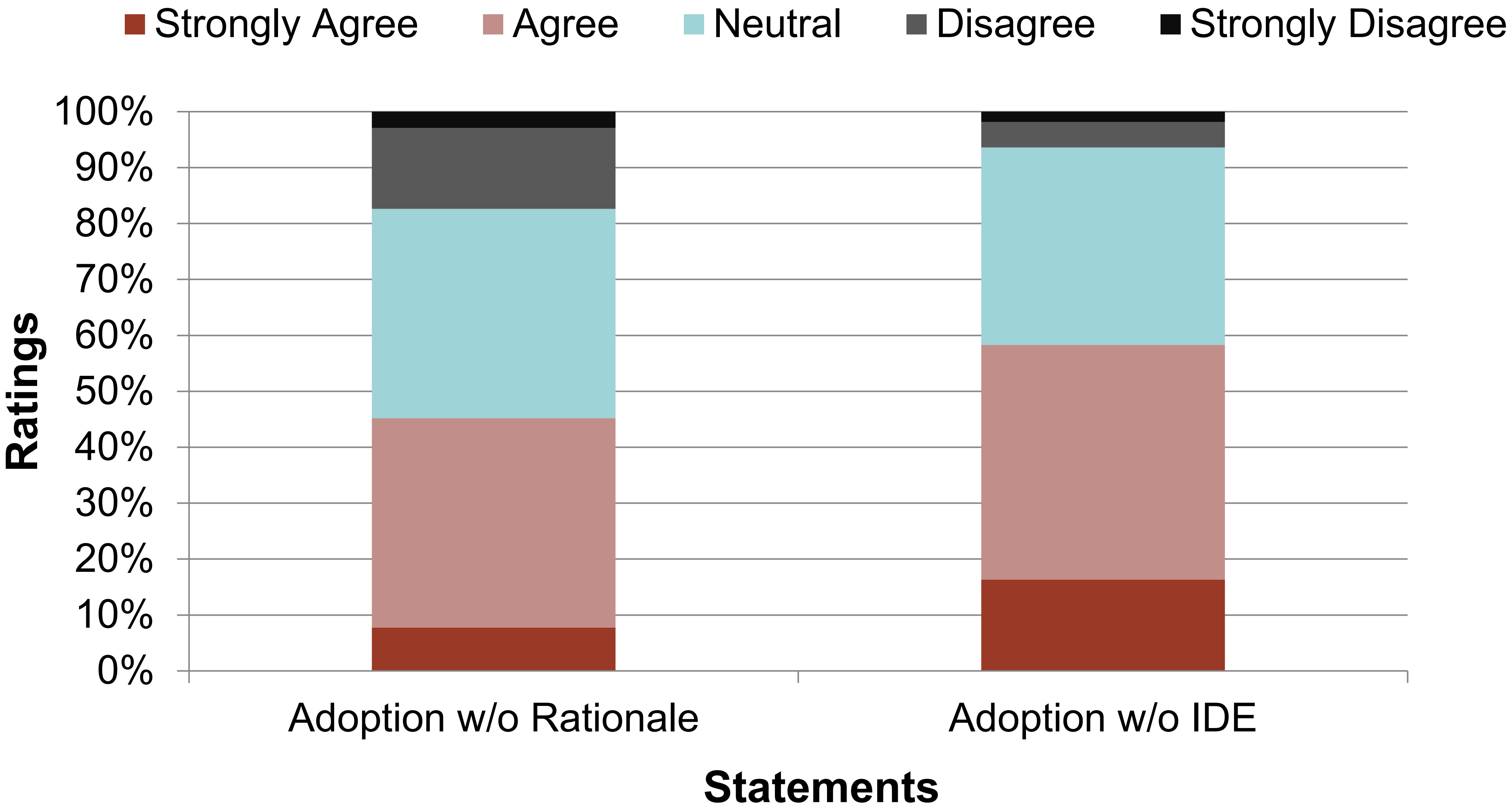


RQ7: Efficiency

Time taken to produce the results.

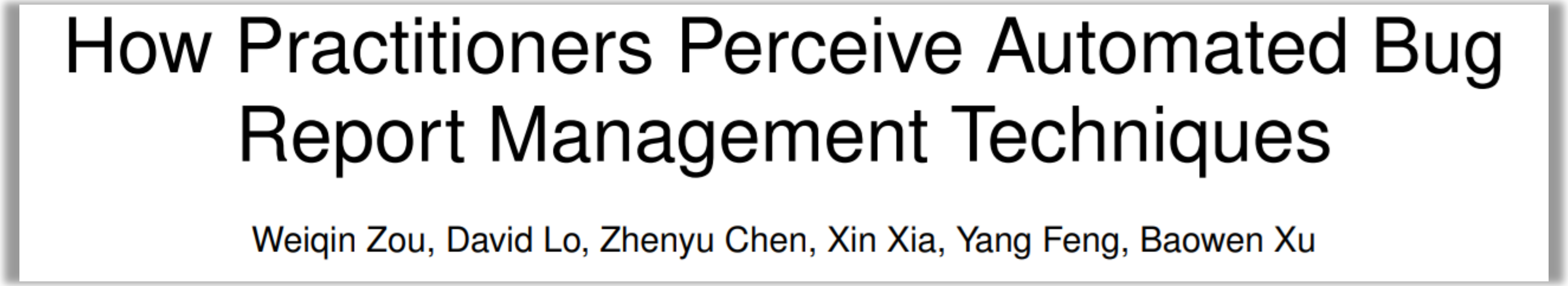


RQ9: Other Factors



Continuation Effort within Our Research Groups

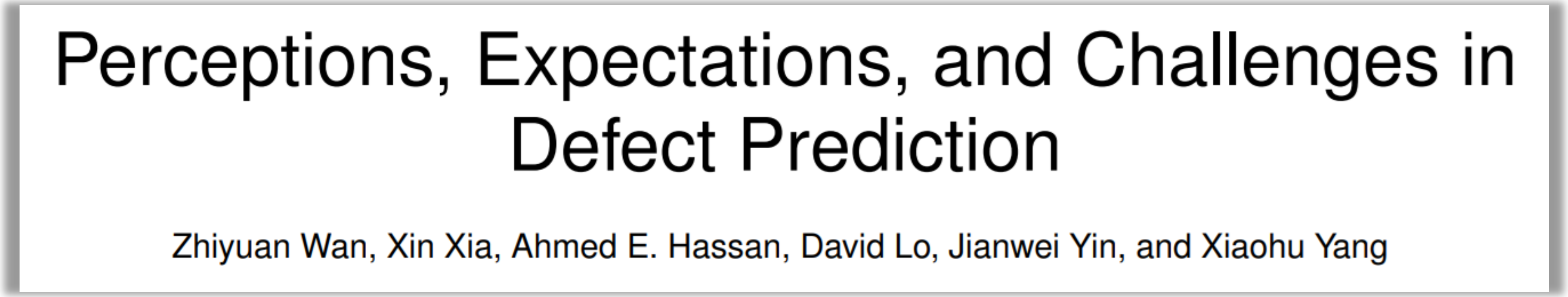
TSE 2020 100+ citations



How Practitioners Perceive Automated Bug Report Management Techniques

Weiqin Zou, David Lo, Zhenyu Chen, Xin Xia, Yang Feng, Baowen Xu

TSE 2020 100+ citations



Perceptions, Expectations, and Challenges in Defect Prediction

Zhiyuan Wan, Xin Xia, Ahmed E. Hassan, David Lo, Jianwei Yin, and Xiaohu Yang

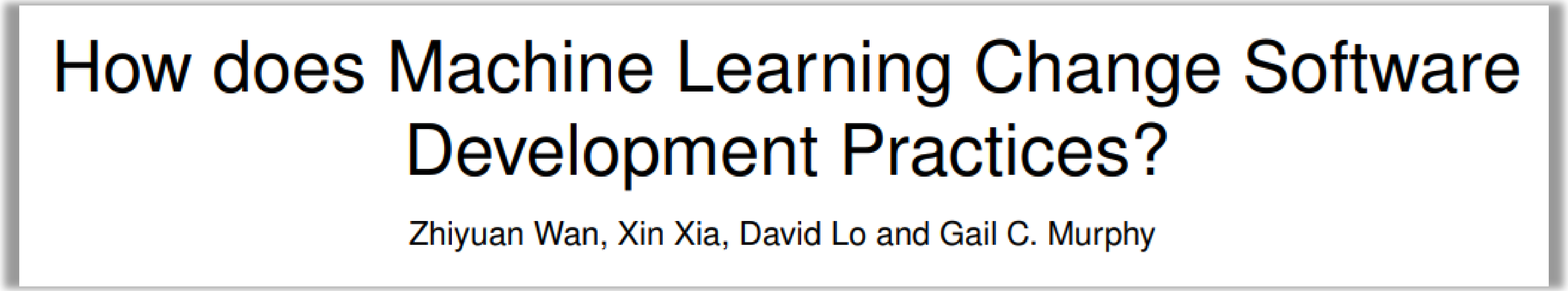
Continuation Effort within Our Research Groups

TSE 2021 800+ citations

Smart Contract Development: Challenges and Opportunities

Weiqin Zou, David Lo, Pavneet Singh Kochhar, Xuan-Bach Dinh Le, Xin Xia, Yang Feng, Zhenyu Chen, Baowen Xu

TSE 2021 250+ citations



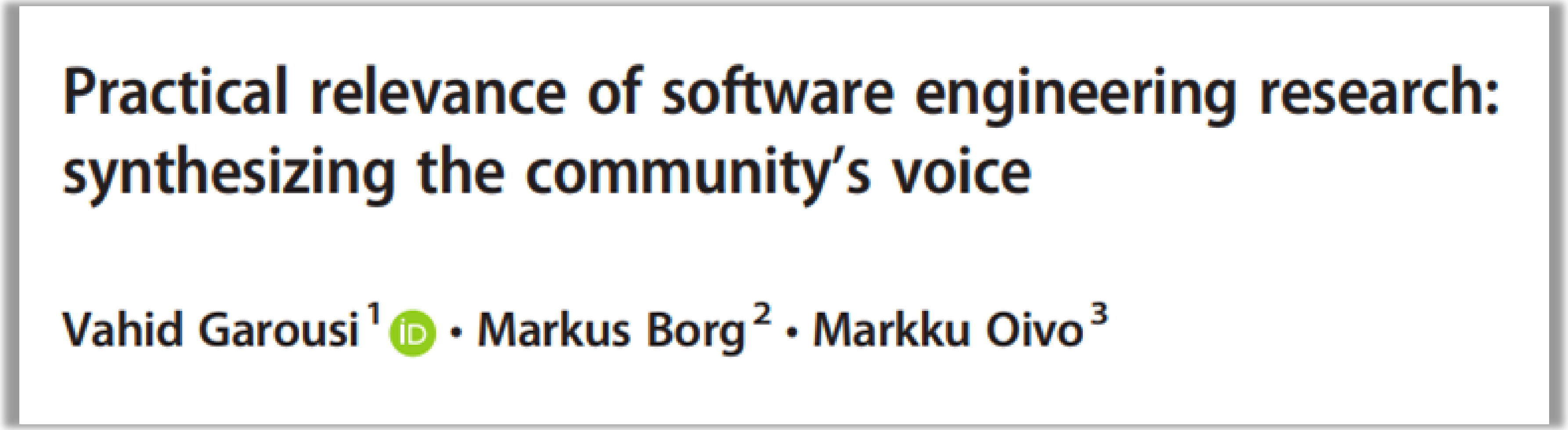
How does Machine Learning Change Software Development Practices?

Zhiyuan Wan, Xin Xia, David Lo and Gail C. Murphy

Best Paper Runner Up

Follow-Up Studies by Others (General SE)

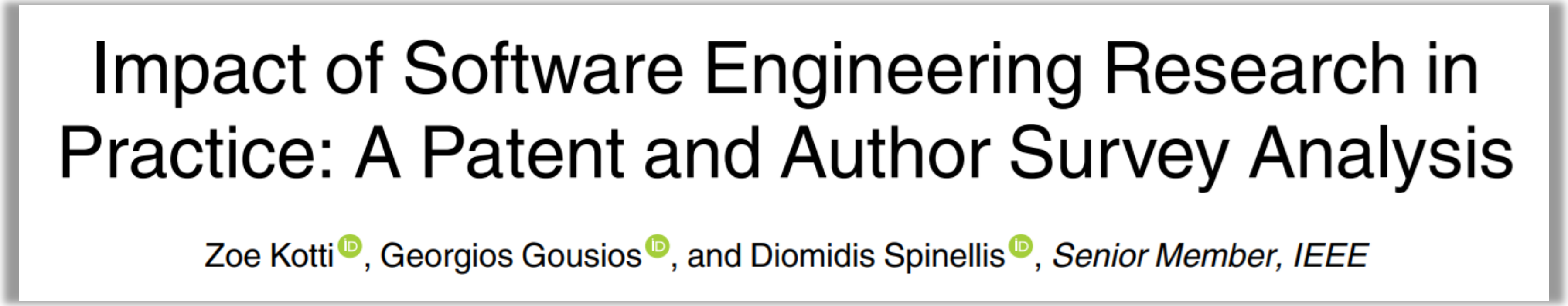
EMSE 2020



Practical relevance of software engineering research: synthesizing the community's voice

Vahid Garousi[1] • Markus Borg[2] • Markku Oivo[3]

TSE 2023



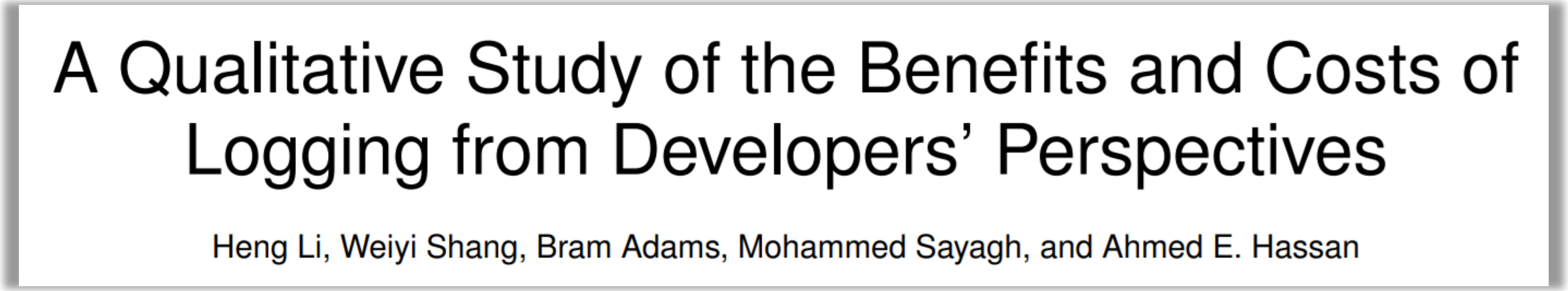
Impact of Software Engineering Research in Practice: A Patent and Author Survey Analysis

Zoe Kotti, Georgios Gousios, and Diomidis Spinellis, *Senior Member, IEEE*

Follow-Up Studies by Others (Specific Topics)

TSE 2021

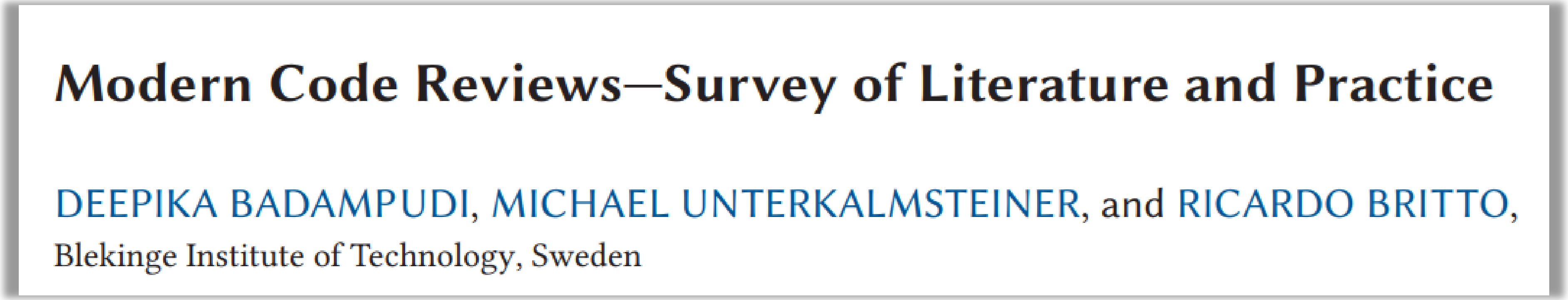
100+ citations



A Qualitative Study of the Benefits and Costs of Logging from Developers' Perspectives

Heng Li, Weiyi Shang, Bram Adams, Mohammed Sayagh, and Ahmed E. Hassan

TOSEM 2023

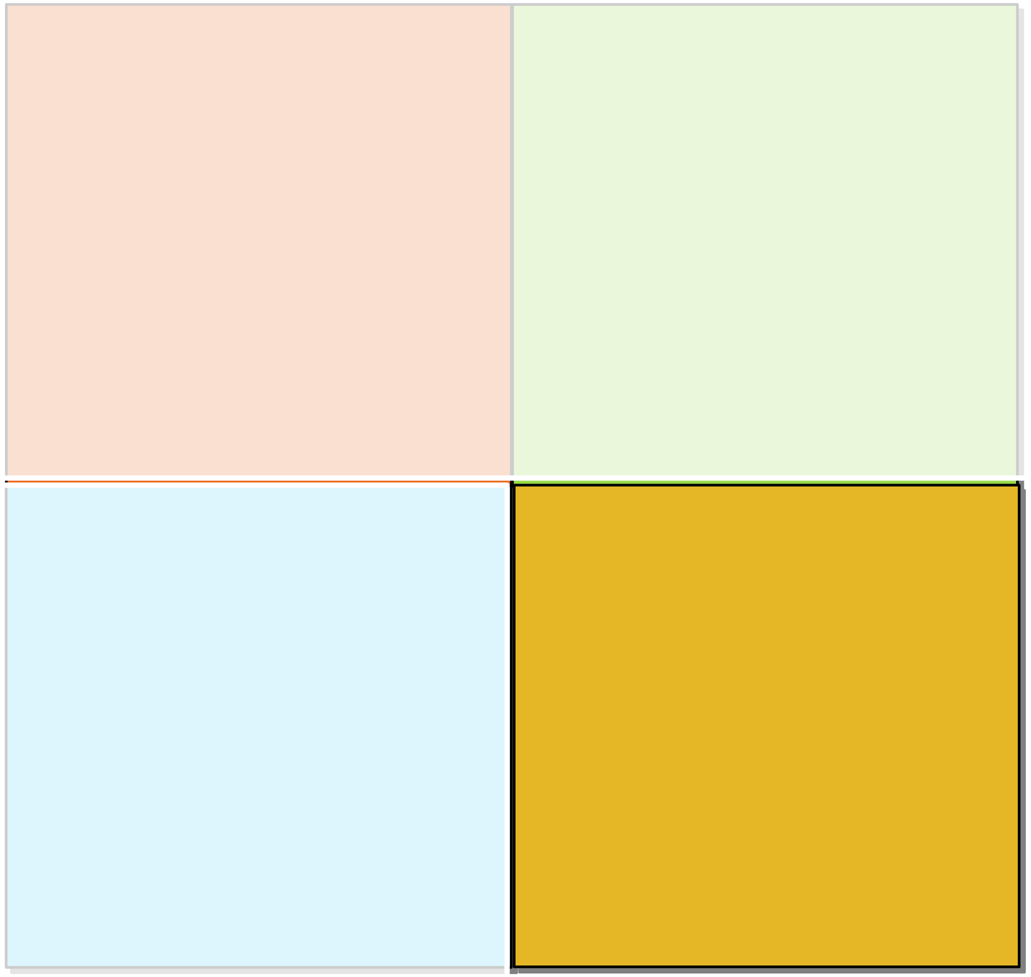


Modern Code Reviews—Survey of Literature and Practice

DEEPIKA BADAMPUDI, MICHAEL UNTERKALMSTEINER, and RICARDO BRITTO,
Blekinge Institute of Technology, Sweden

Impact

- **Hold up a mirror** to SE research
  - Revealed how practitioners perceive our work
- Pioneered a **feedback loop at scale**
  - Hundreds of practitioners, hundreds of papers
  - Transforming data into insights to inform future directions
- Helped **bridge research-practice gap** and inspired a wave of follow-up studies
  - Did *requirements engineering* for SE research
  - Elicited, modeled, and validated practitioner needs



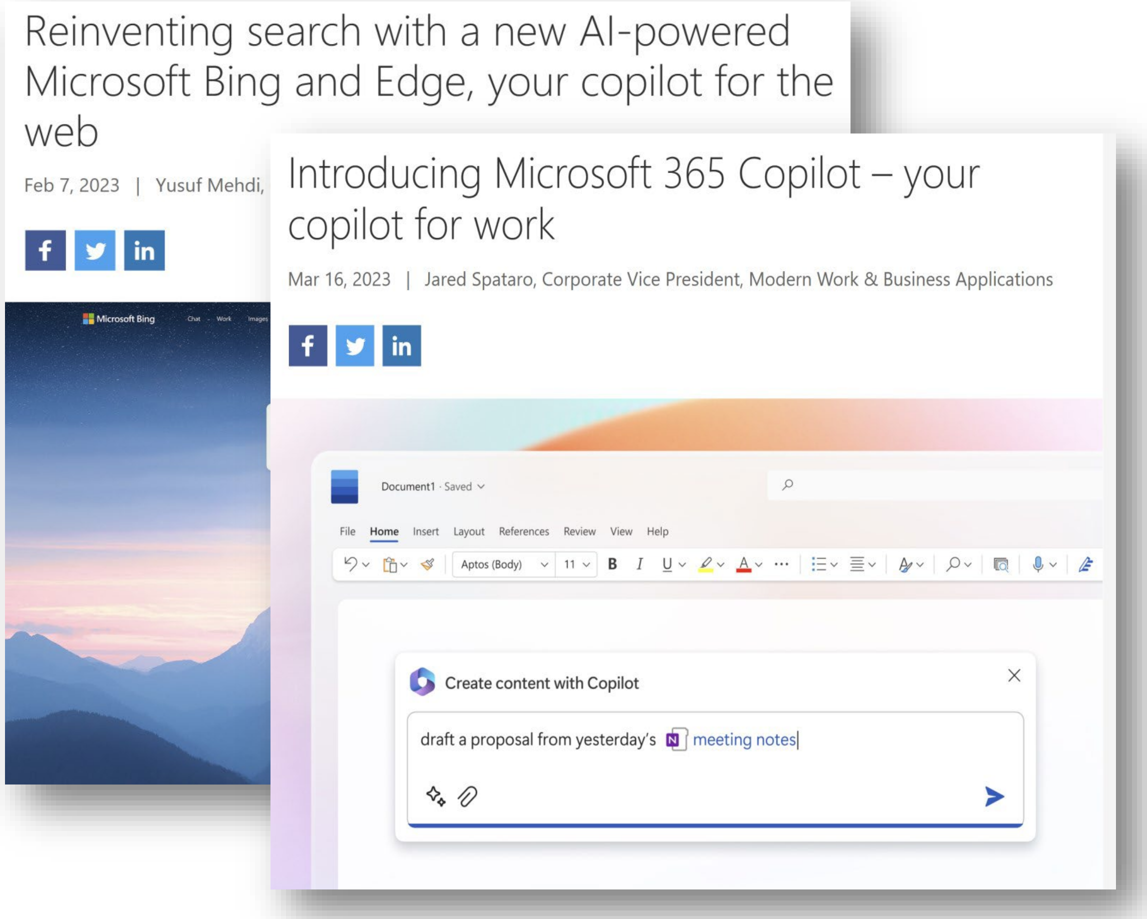
Going Back a Decade

What Is the Paper About?

How Has It Influenced Subsequent Studies?

What Is the Road Ahead?

The AIware Revolution



Reinventing search with a new AI-powered Microsoft Bing and Edge, your copilot for the web

Feb 7, 2023 | Yusuf Mehdi,

Introducing Microsoft 365 Copilot – your copilot for work

Mar 16, 2023 | Jared Spataro, Corporate Vice President, Modern Work & Business Applications

Create content with Copilot

draft a proposal from yesterday's meeting notes

LEADERSHIP > CAREERS

AI Writes Over 25% Of Code At Google—What Does The Future Look Like For Software Engineers?

By Jack Kelly, Senior Contributor. ⓘ Jack Kelly covers career growth, job mar...

Follow Author

Published Nov 01, 2024, 06:30am EDT

Forbes

Nvidia's CEO Says It No Longer Matters If You Never Learned to Code: 'There's a New Programming Language'

At London Tech Week, Nvidia CEO Jensen Huang said even non-programmers can write code thanks to AI.

BY SHERIN SHIBU EDITED BY MELISSA MALAMUT JUN 9, 2025

Share

Entrepreneur

Microsoft Research

SMU SINGAPORE MANAGEMENT UNIVERSITY

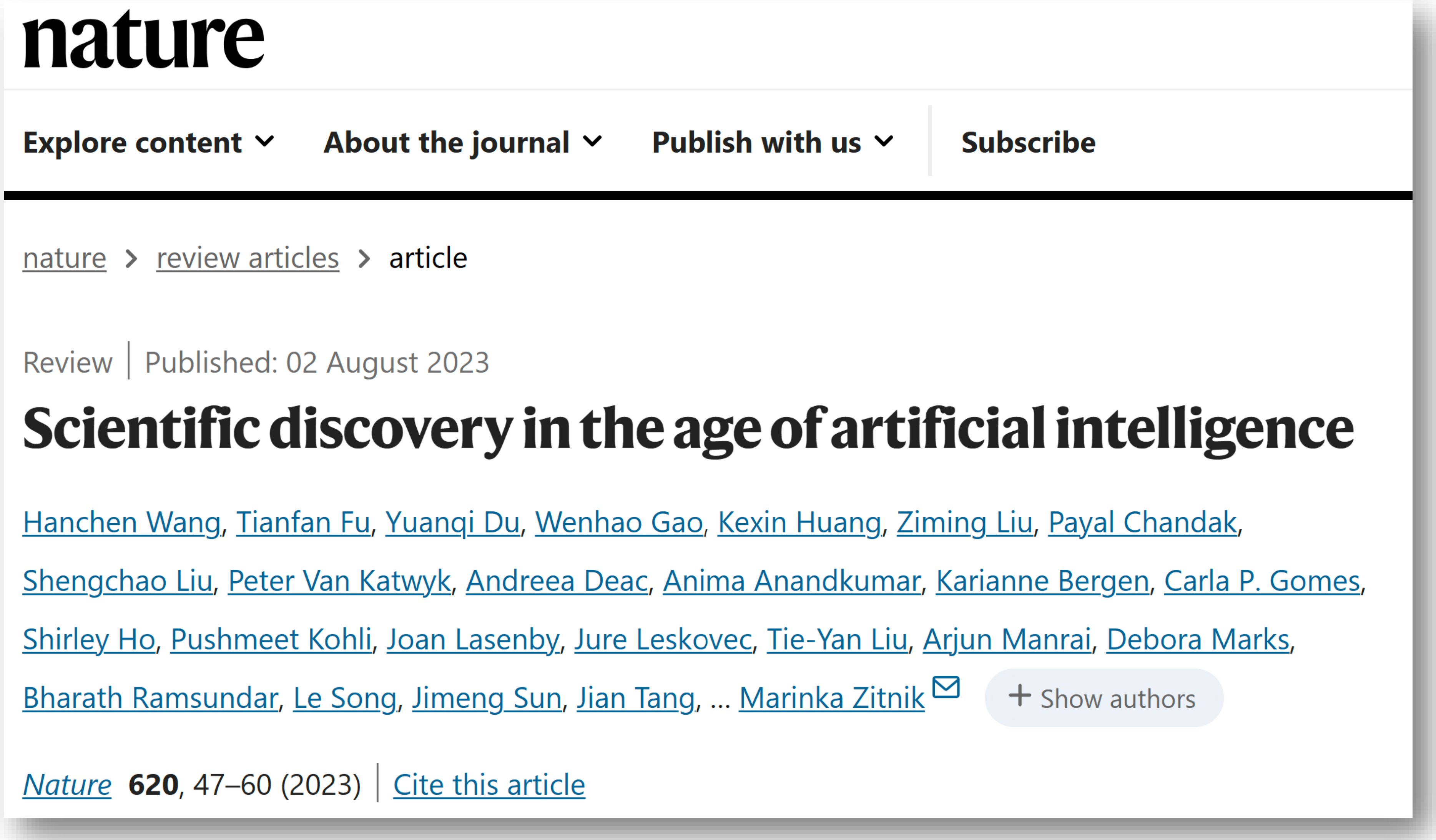
Meta

THE UNIVERSITY OF CALIFORNIA · IRVINE · LET THERE BE LIGHT

The Road Ahead

AI is changing how we do science and build software

AI is Changing How We Do Science



Review | Published: 02 August 2023

Scientific discovery in the age of artificial intelligence

Hanchen Wang, Tianfan Fu, Yuanqi Du, Wenhao Gao, Kexin Huang, Ziming Liu, Payal Chandak, Shengchao Liu, Peter Van Katwyk, Andreea Deac, Anima Anandkumar, Karianne Bergen, Carla P. Gomes, Shirley Ho, Pushmeet Kohli, Joan Lasenby, Jure Leskovec, Tie-Yan Liu, Arjun Manrai, Debora Marks, Bharath Ramsundar, Le Song, Jimeng Sun, Jian Tang, ... Marinka Zitnik

Nature **620**, 47–60 (2023) | Cite this article

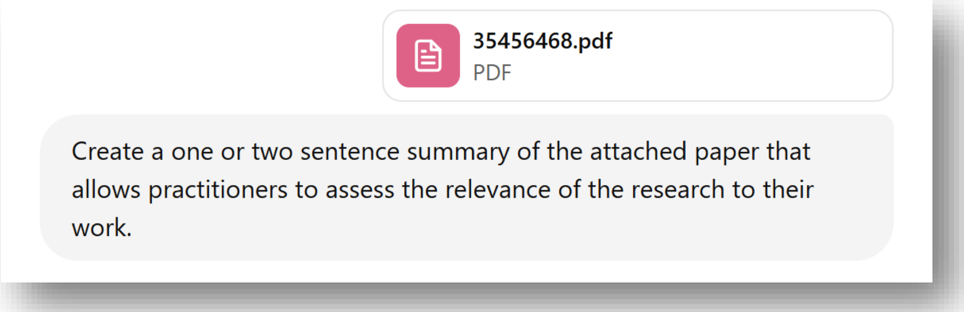
AI is Changing How We Do Science

2015

2025

David summarized 571 papers manually in ~80 hours

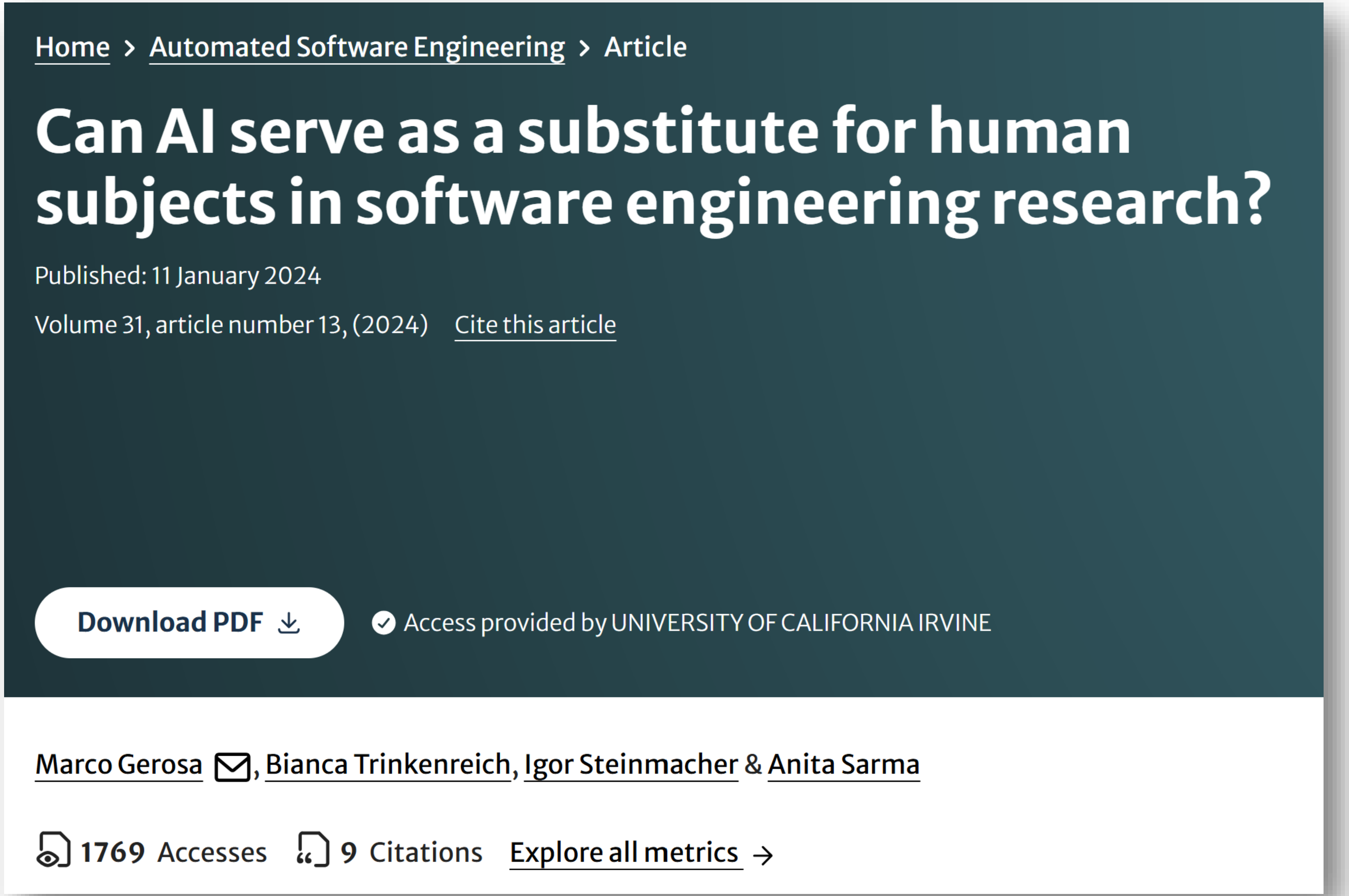
GPT can summarize 571 papers within minutes



35456468.pdf
PDF

Create a one or two sentence summary of the attached paper that allows practitioners to assess the relevance of the research to their work.

AI is Changing How We Do Science



Home > Automated Software Engineering > Article

Can AI serve as a substitute for human subjects in software engineering research?

Published: 11 January 2024

Volume 31, article number 13, (2024) Cite this article

✓ Access provided by UNIVERSITY OF CALIFORNIA IRVINE

Marco Gerosa ✉, Bianca Trinkenreich, Igor Steinmacher & Anita Sarma

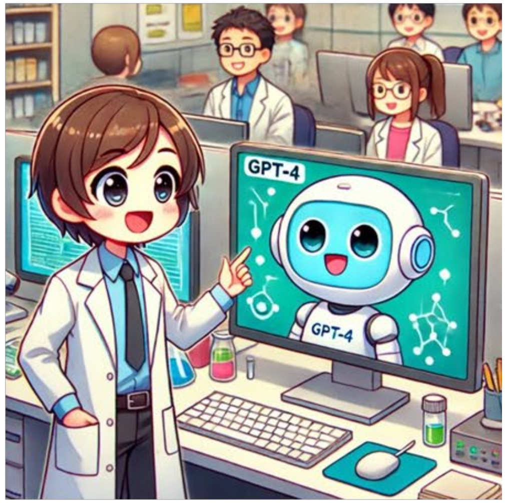
1769 Accesses 9 Citations Explore all metrics →

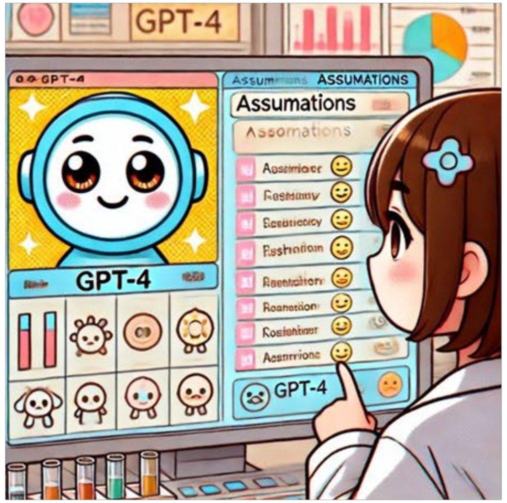
AI is Changing How We Do Science

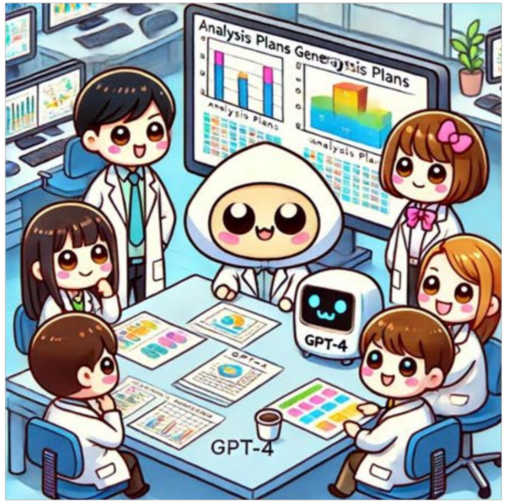
Can GPT-4 Summarize Papers as Cartoons? Yes! :-)

Can GPT-4 Replicate Empirical Software Engineering Research?

Jenny T. Liang, Carmen Badea, Christian Bird, Robert DeLine, Denae Ford, Nicole Forsgren, Thomas Zimmermann
PACMSE (FSE) 2024.

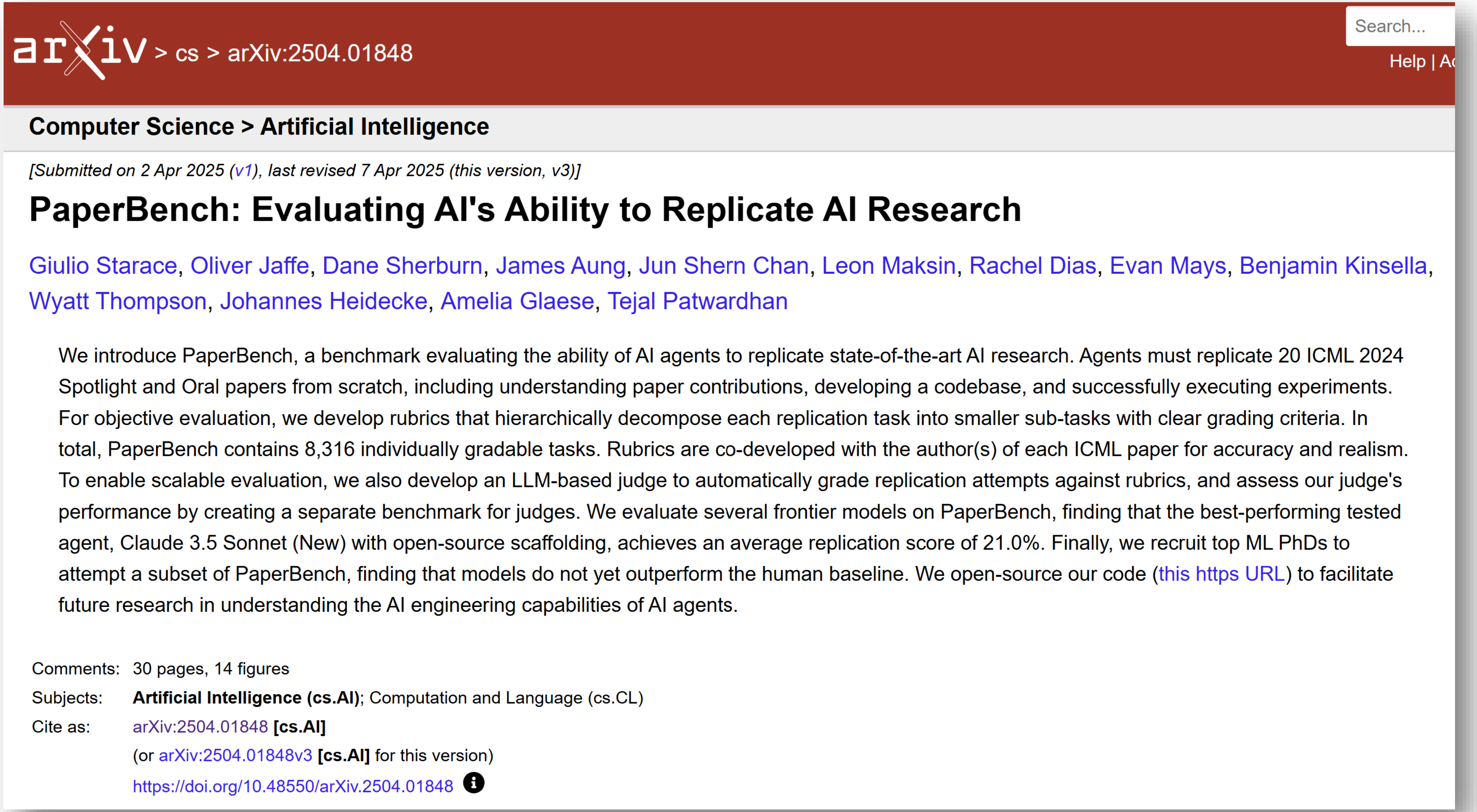






AI-generated images may be incorrect. None of the authors wore a lab coat during this research. :-)

AI is Changing How We Do Science



Computer Science > Artificial Intelligence

[Submitted on 2 Apr 2025 (v1), last revised 7 Apr 2025 (this version, v3)]

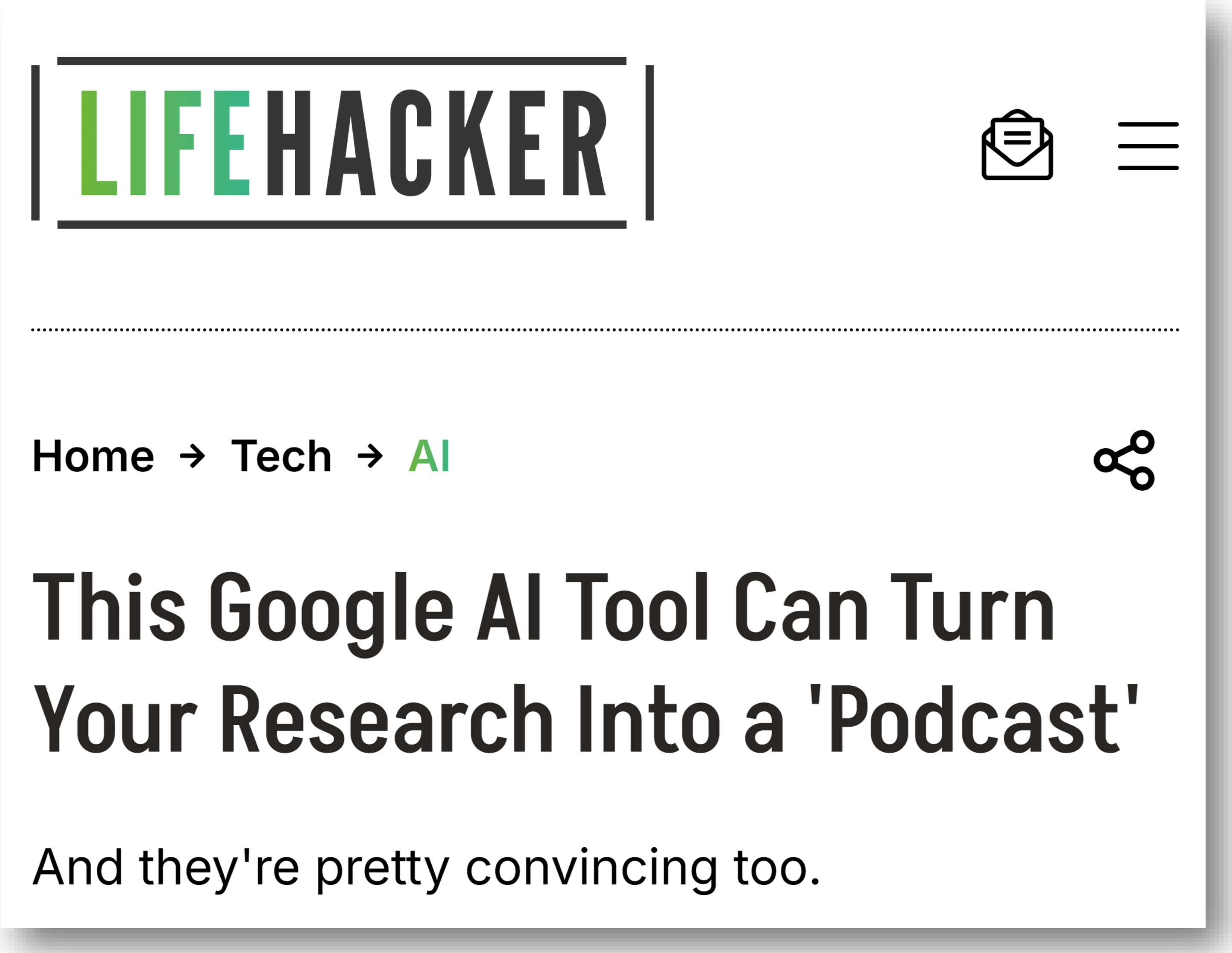
PaperBench: Evaluating AI's Ability to Replicate AI Research

Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, Johannes Heidecke, Amelia Glaese, Tejal Patwardhan

We introduce PaperBench, a benchmark evaluating the ability of AI agents to replicate state-of-the-art AI research. Agents must replicate 20 ICML 2024 Spotlight and Oral papers from scratch, including understanding paper contributions, developing a codebase, and successfully executing experiments. For objective evaluation, we develop rubrics that hierarchically decompose each replication task into smaller sub-tasks with clear grading criteria. In total, PaperBench contains 8,316 individually gradable tasks. Rubrics are co-developed with the author(s) of each ICML paper for accuracy and realism. To enable scalable evaluation, we also develop an LLM-based judge to automatically grade replication attempts against rubrics, and assess our judge's performance by creating a separate benchmark for judges. We evaluate several frontier models on PaperBench, finding that the best-performing tested agent, Claude 3.5 Sonnet (New) with open-source scaffolding, achieves an average replication score of 21.0%. Finally, we recruit top ML PhDs to attempt a subset of PaperBench, finding that models do not yet outperform the human baseline. We open-source our code (this https URL) to facilitate future research in understanding the AI engineering capabilities of AI agents.

Comments: 30 pages, 14 figures
Subjects: **Artificial Intelligence (cs.AI)**; Computation and Language (cs.CL)
Cite as: arXiv:2504.01848 **[cs.AI]**
(or arXiv:2504.01848v3 **[cs.AI]** for this version)
https://doi.org/10.48550/arXiv.2504.01848

AI is Changing How We Do Science

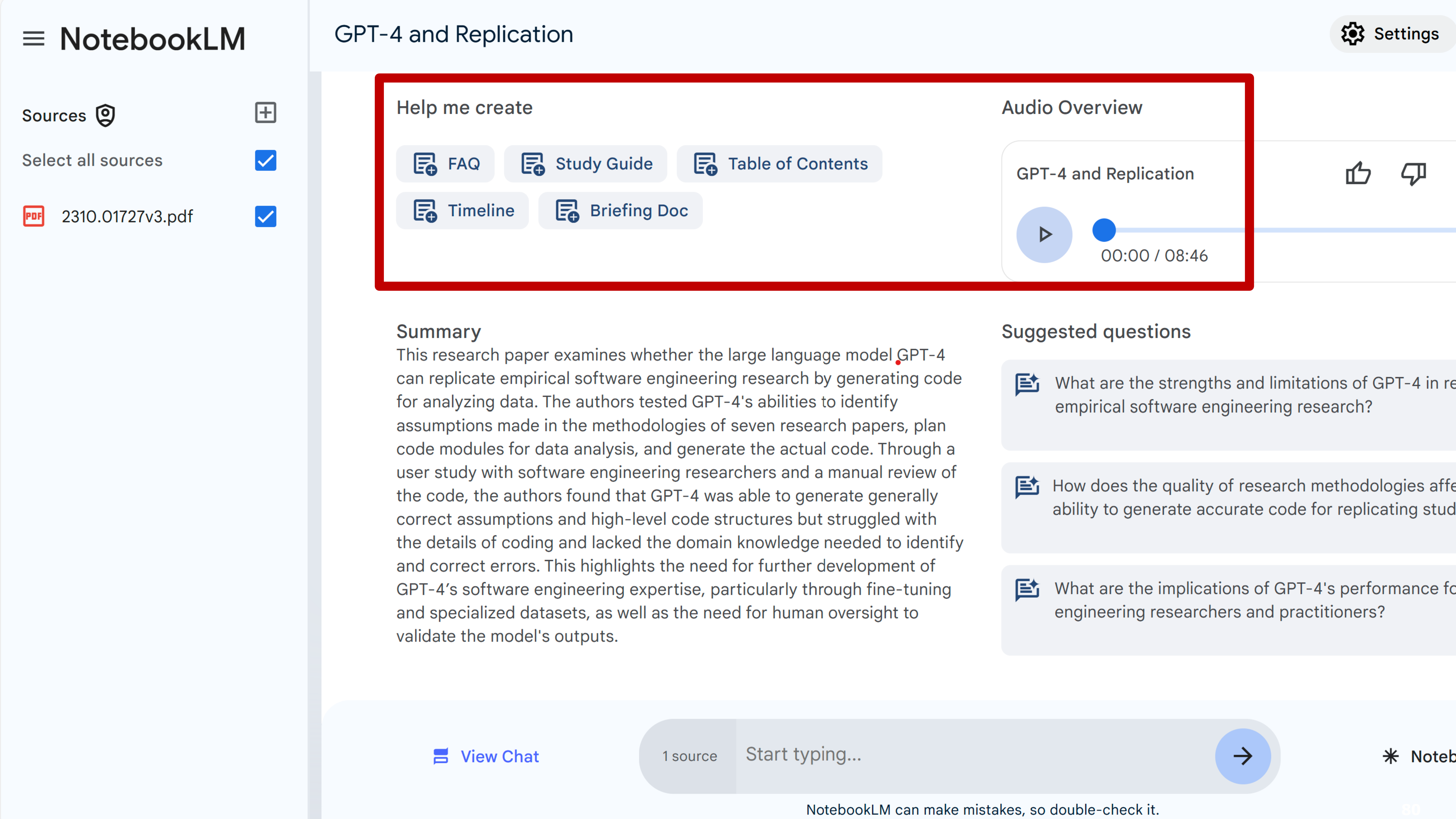


LIFEHACKER

Home → Tech → AI

This Google AI Tool Can Turn Your Research Into a 'Podcast'

And they're pretty convincing too.



Summary

This research paper examines whether the large language model GPT-4 can replicate empirical software engineering research by generating code for analyzing data. The authors tested GPT-4's abilities to identify assumptions made in the methodologies of seven research papers, plan code modules for data analysis, and generate the actual code. Through a user study with software engineering researchers and a manual review of the code, the authors found that GPT-4 was able to generate generally correct assumptions and high-level code structures but struggled with the details of coding and lacked the domain knowledge needed to identify and correct errors. This highlights the need for further development of GPT-4's software engineering expertise, particularly through fine-tuning and specialized datasets, as well as the need for human oversight to validate the model's outputs.

Suggested questions

What are the strengths and limitations of GPT-4 in re empirical software engineering research?

How does the quality of research methodologies affe ability to generate accurate code for replicating stud

What are the implications of GPT-4's performance fo engineering researchers and practitioners?

NotebookLM can make mistakes, so double-check it.

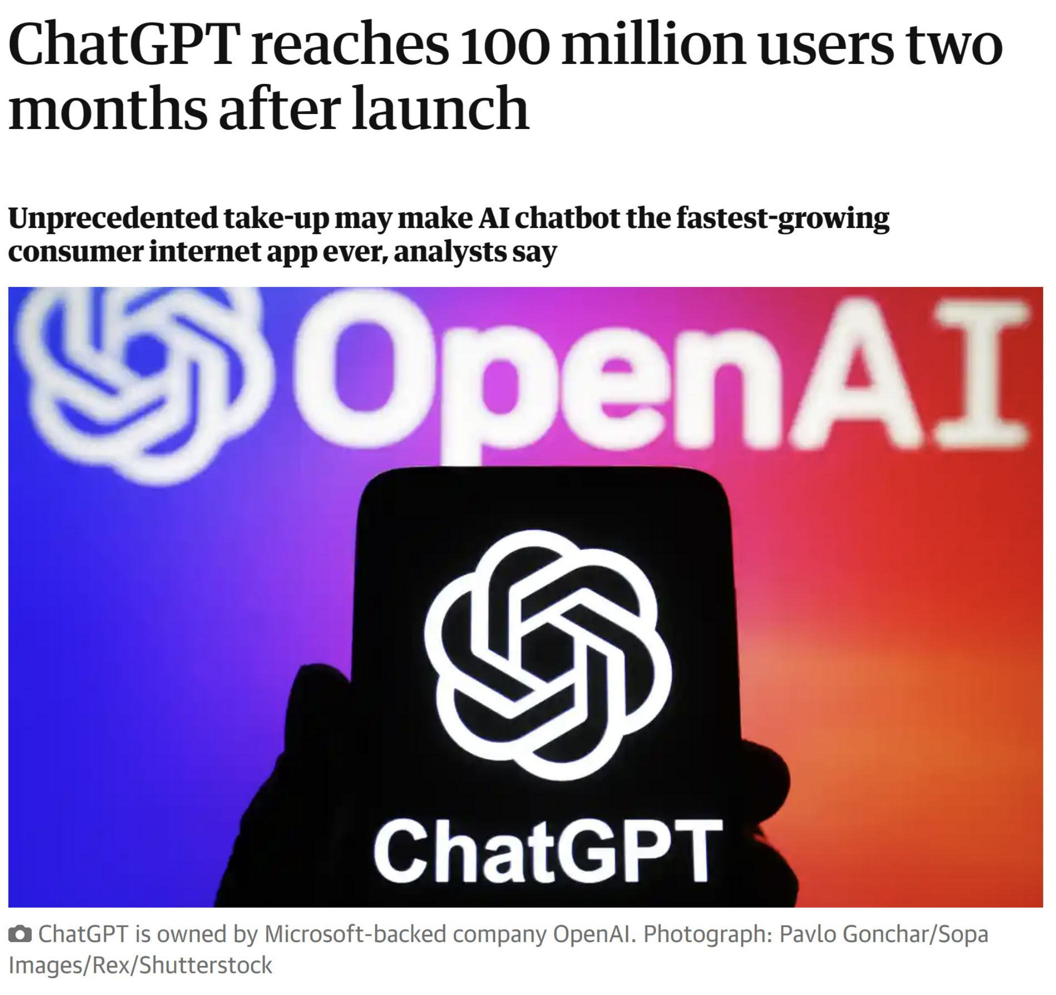
The Road Ahead

AI is changing how we do science and build software

Focusing on relevant research is even more important now

Focusing on Relevant Research is More Important Now

The speed of innovation has increased dramatically…
(at least in industry)



ChatGPT reaches 100 million users two months after launch

Unprecedented take-up may make AI chatbot the fastest-growing consumer internet app ever, analysts say

ChatGPT is owned by Microsoft-backed company OpenAI. Photograph: Pavlo Gonchar/Sopa Images/Rex/Shutterstock

Focusing on Relevant Research is More Important Now

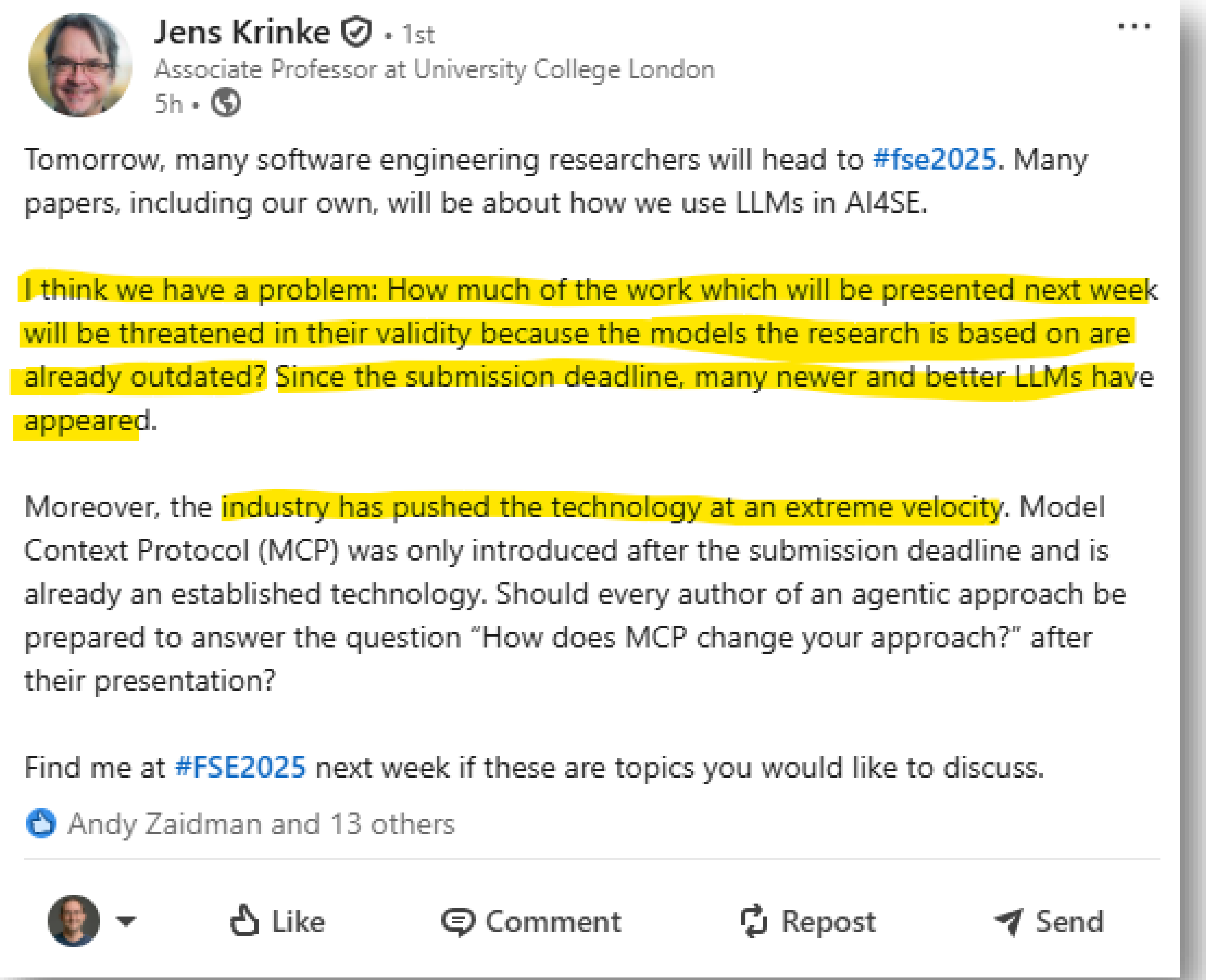
We publish many more research papers, but are they all relevant?

# Submissions (after desk rejection)

	2015	2025
ASE	317	1137
FSE ESEC/FSE	291	603
ICSE	452	1150

Focusing on Relevant Research is More Important Now

And research papers are published too slow.



Jens Krinke • 1st
Associate Professor at University College London
5h •

Tomorrow, many software engineering researchers will head to #fse2025. Many papers, including our own, will be about how we use LLMs in AI4SE.

I think we have a problem: How much of the work which will be presented next week will be threatened in their validity because the models the research is based on are already outdated? Since the submission deadline, many newer and better LLMs have appeared.

Moreover, the industry has pushed the technology at an extreme velocity. Model Context Protocol (MCP) was only introduced after the submission deadline and is already an established technology. Should every author of an agentic approach be prepared to answer the question "How does MCP change your approach?" after their presentation?

Find me at #FSE2025 next week if these are topics you would like to discuss.

Andy Zaidman and 13 others

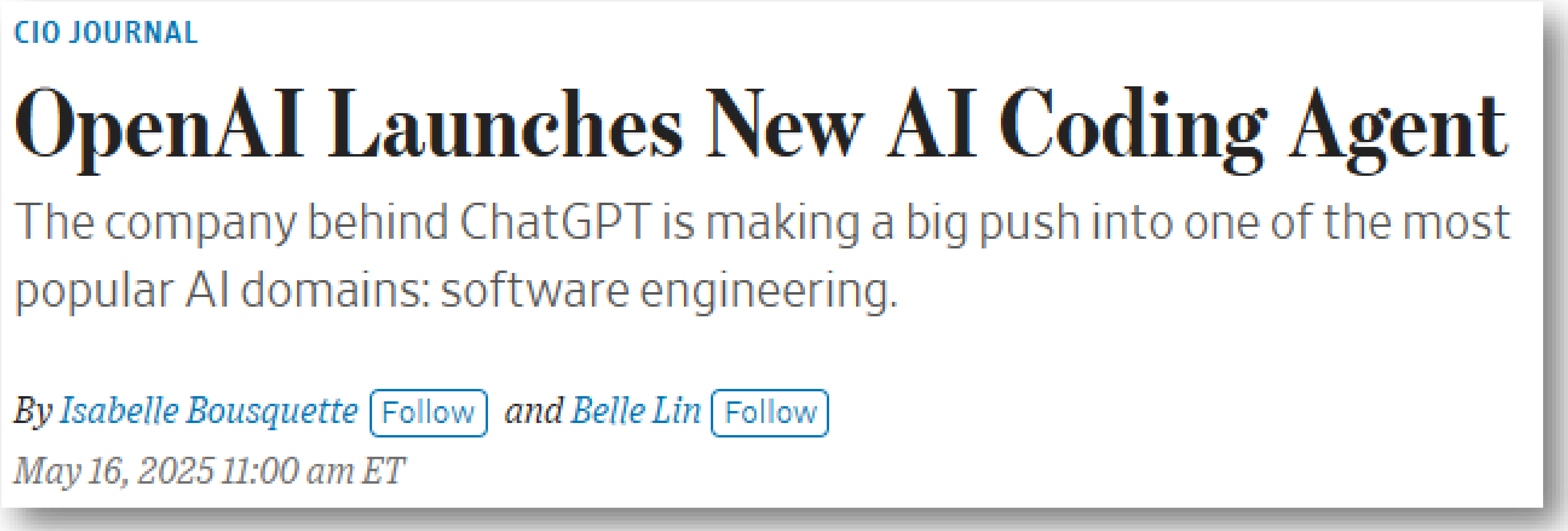
The Road Ahead

AI is changing how we do science and build software

Focusing on relevant research is even more important now

Ensure that software engineering stays relevant for the future

Ensure that Software Engineering Stays Relevant



CIO JOURNAL

OpenAI Launches New AI Coding Agent

The company behind ChatGPT is making a big push into one of the most popular AI domains: software engineering.

By *Isabelle Bousquette* Follow and *Belle Lin* Follow

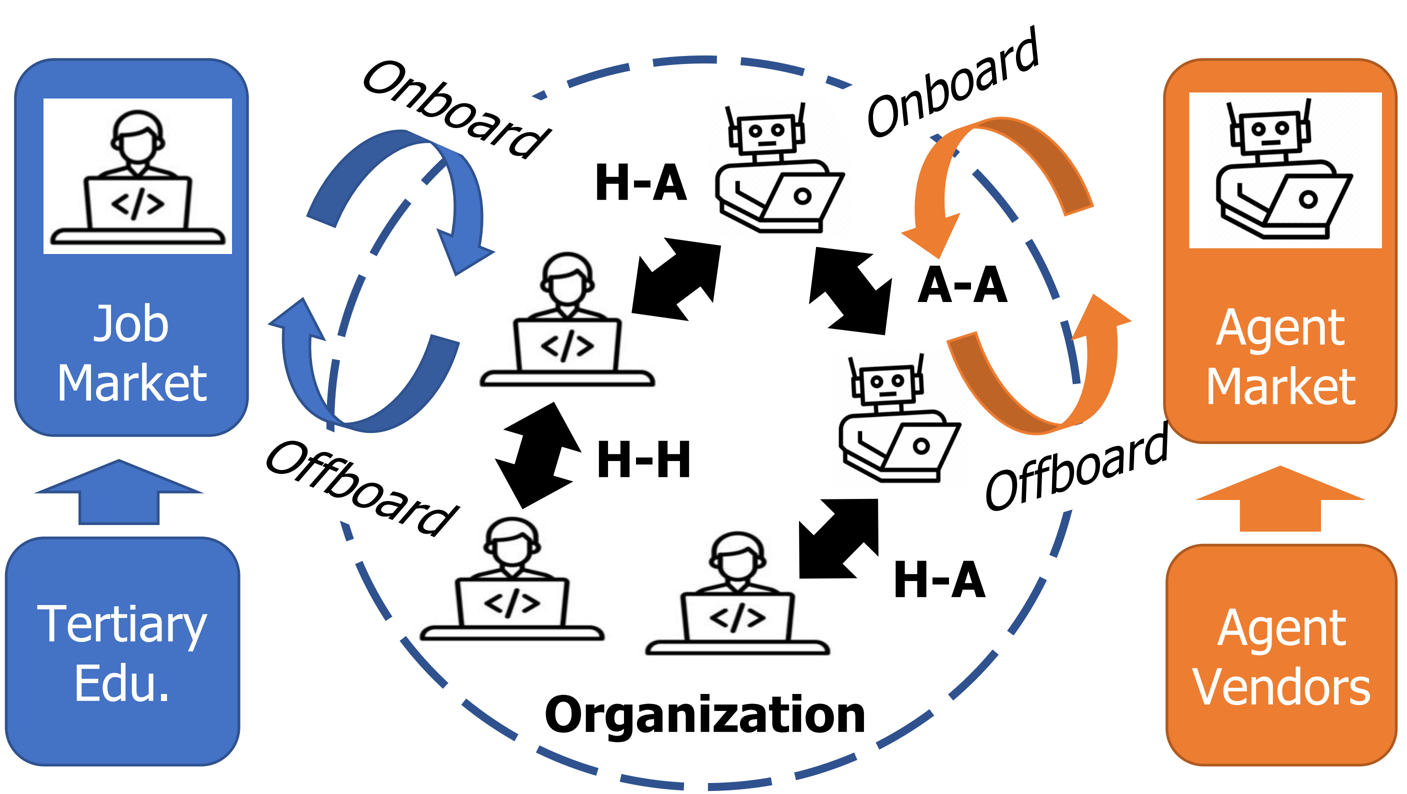
May 16, 2025 11:00 am ET

THE WALL STREET JOURNAL.

The Future of Software Engineering

ICSE 2023
Future of SE

Symbiotic workforce of autonomous, responsible, intelligent agents & engineers



Job Market

Tertiary Edu.

Onboard

Offboard

H-A

H-H

A-A

H-A

Onboard

Offboard

Organization

Agent Market

Agent Vendors

Smart Tool

Smart Workmate

Economics

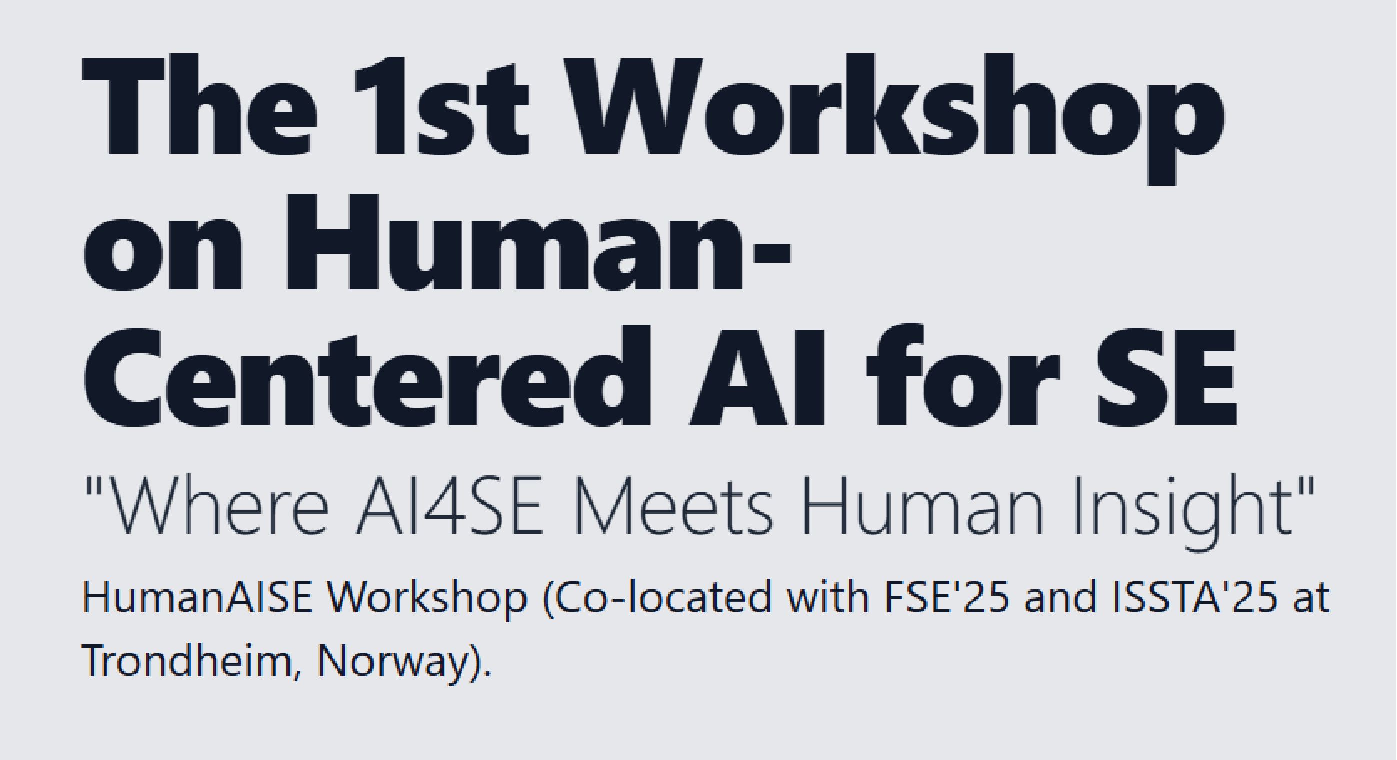
Law

Ethics

How Practitioners Perceive the Relevance of Software Engineering Research

...
...
Law Researchers
Regulators
AI Researchers
AI Practitioners
SE Researchers
SE Practitioners
SE Researchers
AI Researchers

Relevance in the Era of AI powered Software Engineering



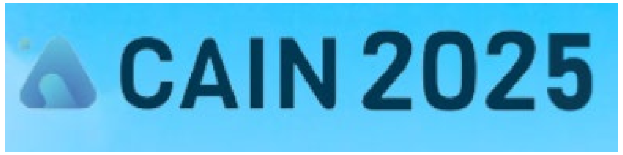
The 1st Workshop on Human-Centered AI for SE

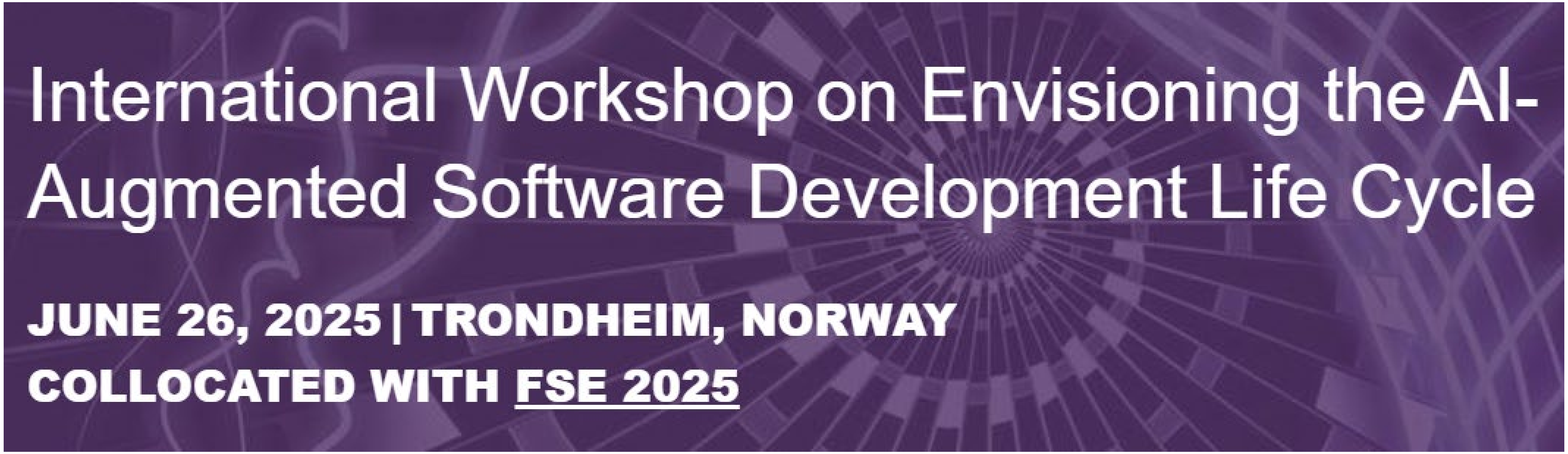
"Where AI4SE Meets Human Insight"

HumanAISE Workshop (Co-located with FSE'25 and ISSTA'25 at Trondheim, Norway).

AIware 2025

58 posts



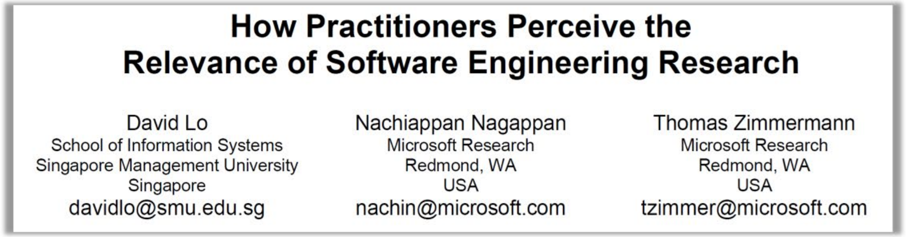


International Workshop on Envisioning the AI-Augmented Software Development Life Cycle

JUNE 26, 2025 | TRONDHEIM, NORWAY

COLLOCATED WITH FSE 2025

Going Back a Decade



How Practitioners Perceive the Relevance of Software Engineering Research

David Lo
School of Information Systems
Singapore Management University
Singapore
davidlo@smu.edu.sg

Nachiappan Nagappan
Microsoft Research
Redmond, WA
USA
nachin@microsoft.com

Thomas Zimmermann
Microsoft Research
Redmond, WA
USA
tzimmer@microsoft.com

10th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering

BERGAMO, ITALY, August 30 – September 4

Summary of Findings - II

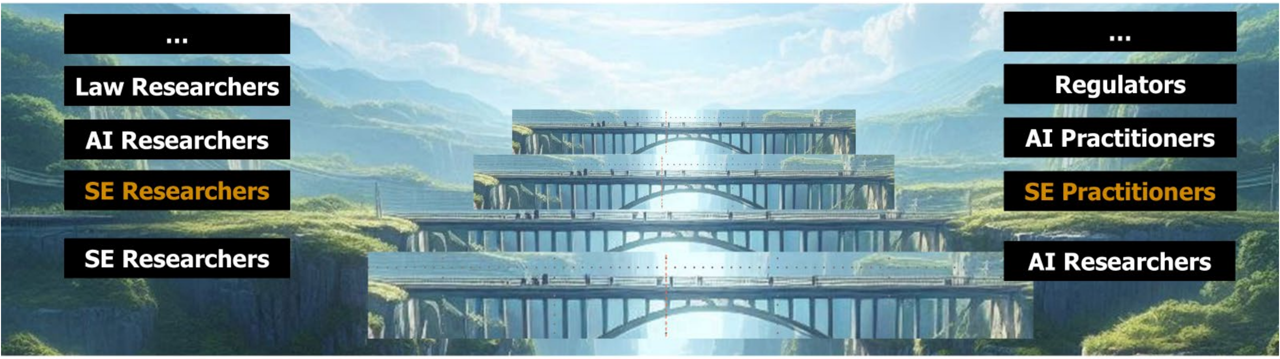
- Threats to relevance of SE research:
  - A tool is not needed
  - An empirical study is not actionable
  - Generalizability issue
  - Cost outweighs benefit
  - Questionable assumptions
  - Disbelief in a particular technology or methodology
  - Another solution/problem seems better/more important
  - Proposed solution has side effects

Microsoft Research SMU Singapore Management University Meta

Impact

- **Hold up a mirror** to SE research
  - Revealed how practitioners perceive our work
- Pioneered a **feedback loop at scale**
  - Hundreds of practitioners, hundreds of papers
  - Transforming data into insights to inform future directions
- Helped **bridge research-practice gap** and inspired a wave of follow-up studies
  - Did *requirements engineering* for SE research
  - Elicited, modeled, and validated practitioner needs

How Practitioners Perceive the Relevance of Software Engineering Research



Relevance in the Era of AI powered Software Engineering

Thank You!

- FSE’25 Test-of-Time Award Committee
- Tom Ball, Christian Bird, Prem Devanbu, Miryung Kim, Emerson Murphy-Hill, Andreas Zeller, and anonymous ESEC-FSE’15 reviewers
- Everyone who responded to our survey a decade ago
- Everyone who have inspired us, collaborated with us, and extended our work

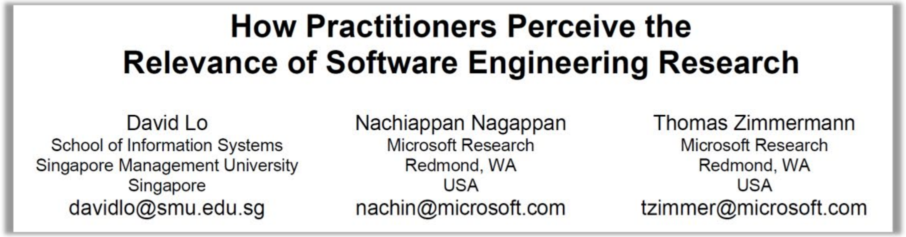
OUB Chair
Professorship Fund

Thank You!

Questions? Comments? Advice?

davidlo@smu.edu.sg, nnachi@meta.com, and tzimmer@uci.edu

Going Back a Decade



How Practitioners Perceive the Relevance of Software Engineering Research

David Lo
School of Information Systems
Singapore Management University
Singapore
davidlo@smu.edu.sg

Nachiappan Nagappan
Microsoft Research
Redmond, WA
USA
nachin@microsoft.com

Thomas Zimmermann
Microsoft Research
Redmond, WA
USA
tzimmer@microsoft.com

10th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering

BERGAMO, ITALY, August 30 – September 4

Summary of Findings - II

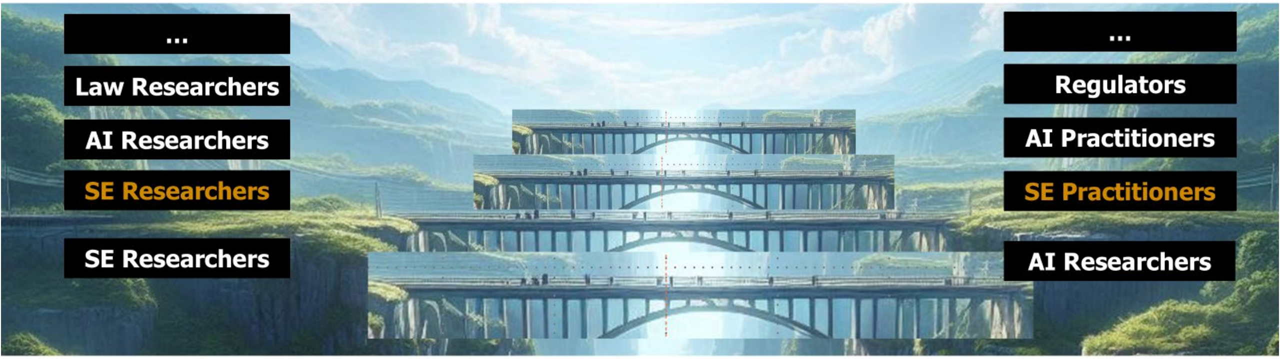
- Threats to relevance of SE research:
  - A tool is not needed
  - An empirical study is not actionable
  - Generalizability issue
  - Cost outweighs benefit
  - Questionable assumptions
  - Disbelief in a particular technology or methodology
  - Another solution/problem seems better/more important
  - Proposed solution has side effects

Microsoft Research   SMU SINGAPORE MANAGEMENT UNIVERSITY   Meta   THE UNIVERSITY OF CALIFORNIA · IRVINE ·

Impact

- **Hold up a mirror** to SE research
  - Revealed how practitioners perceive our work
- Pioneered a **feedback loop at scale**
  - Hundreds of practitioners, hundreds of papers
  - Transforming data into insights to inform future directions
- Helped **bridge research-practice gap** and inspired a wave of follow-up studies
  - Did *requirements engineering* for SE research
  - Elicited, modeled, and validated practitioner needs

How Practitioners Perceive the Relevance of Software Engineering Research



Relevance in the Era of AI powered Software Engineering