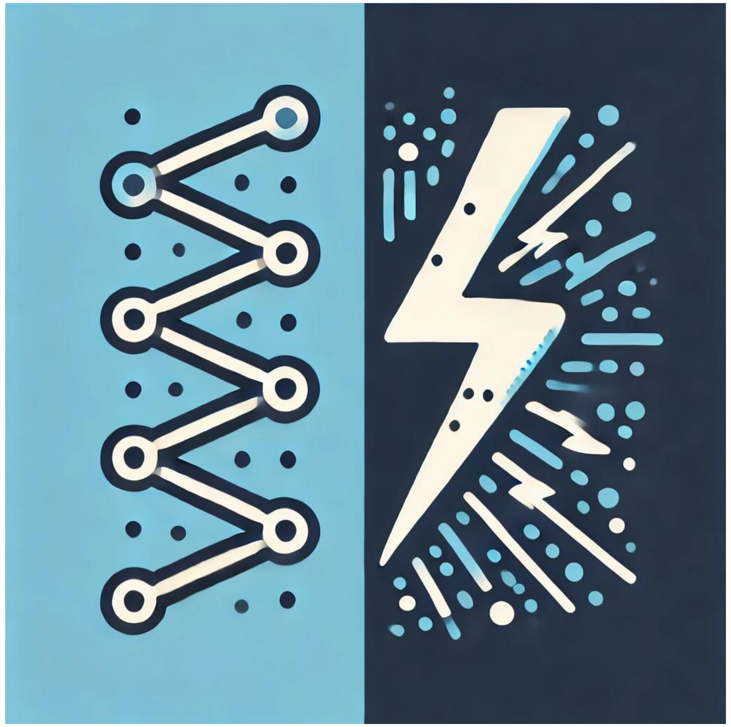
Empirical Software Engineering Research in the Age of LLMs

Christoph Treude

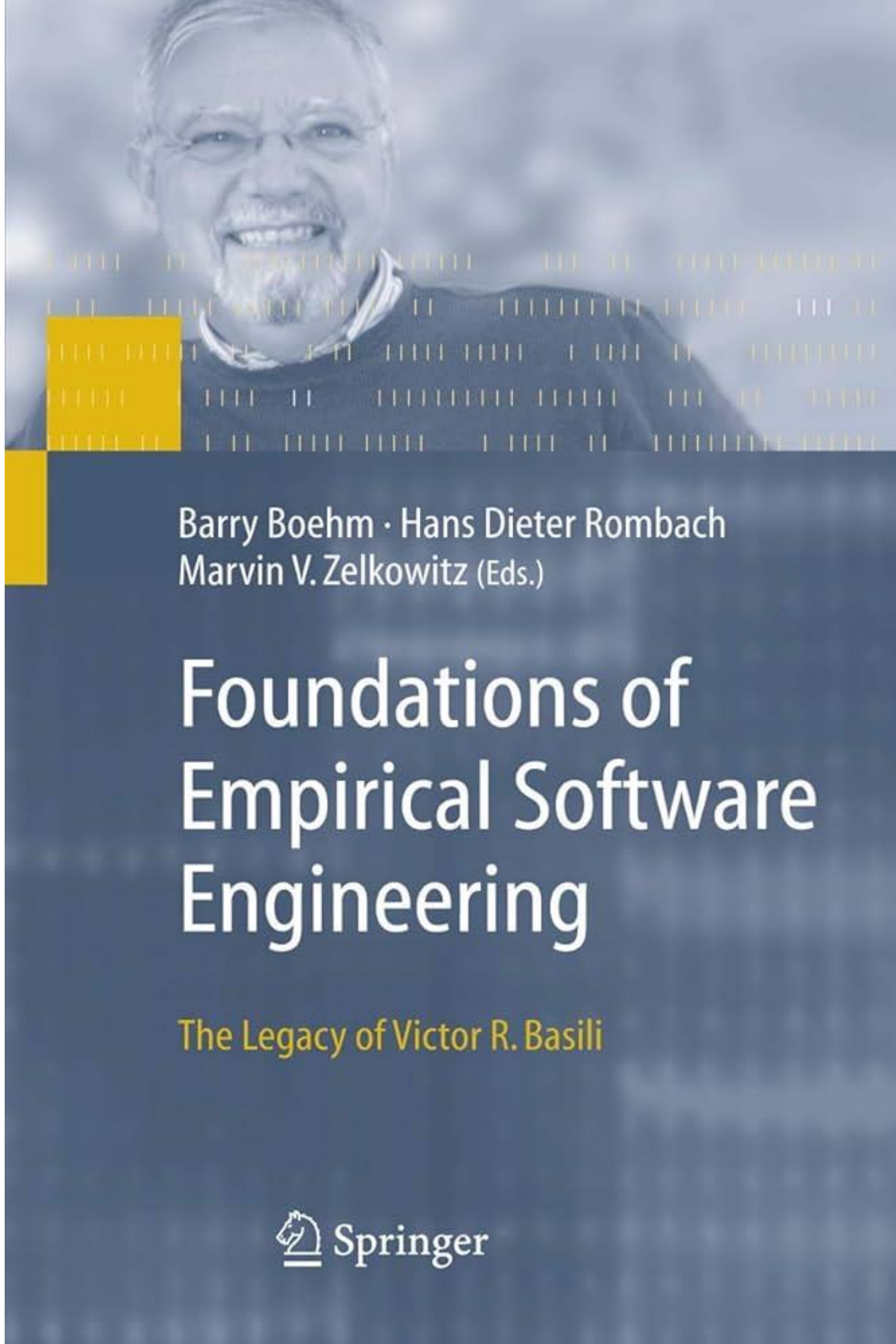


SINGAPORE MANAGEMENT UNIVERSITY

Empirical SE

“Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis.”

[Harrison and Basili]



Barry Boehm · Hans Dieter Rombach
Marvin V. Zelkowitz (Eds.)

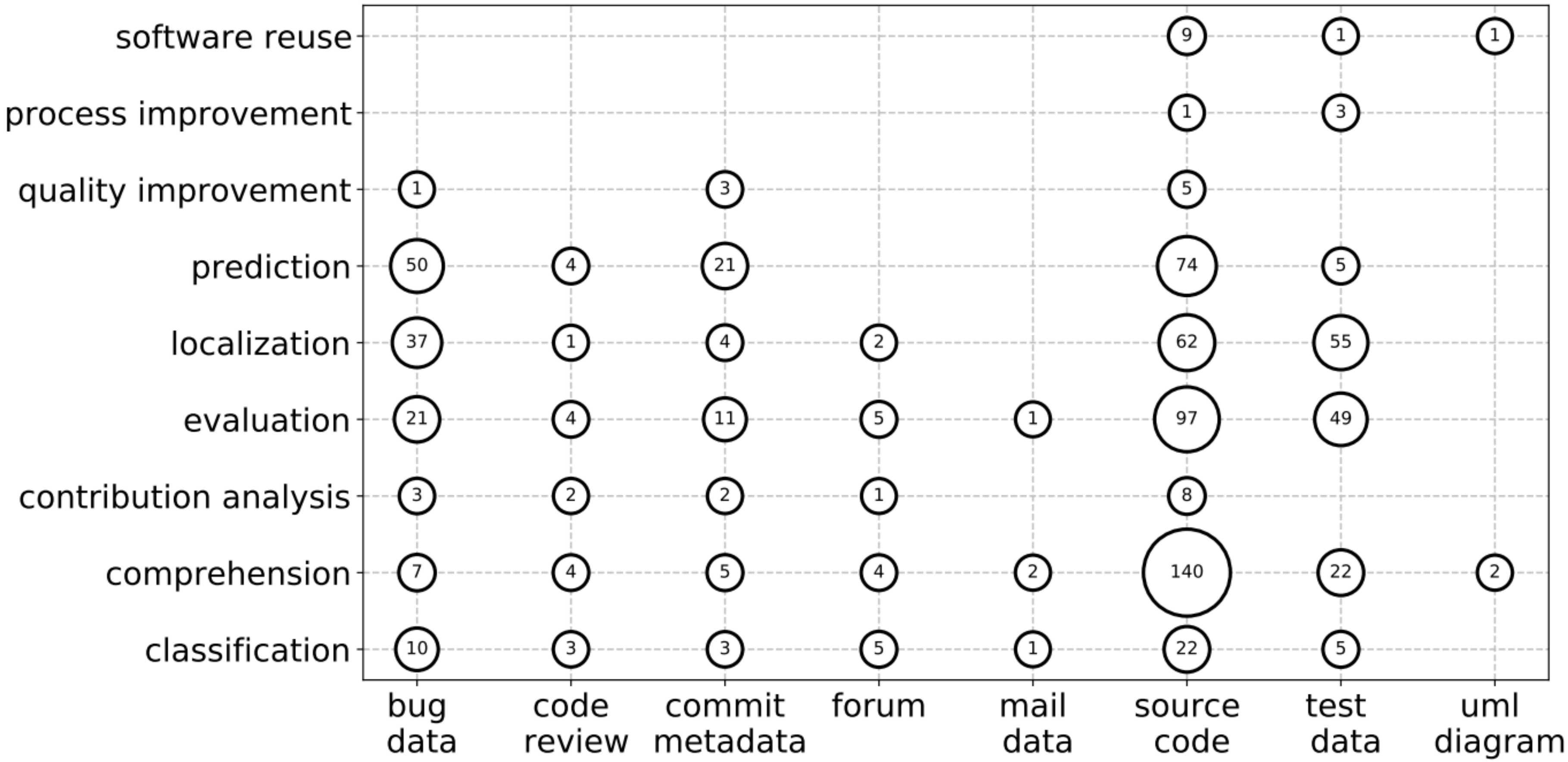
Foundations of Empirical Software Engineering

The Legacy of Victor R. Basili

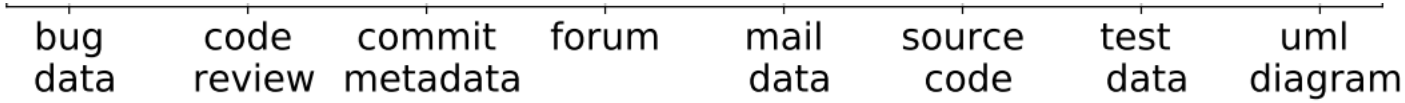
Springer

Software Artifact Mining

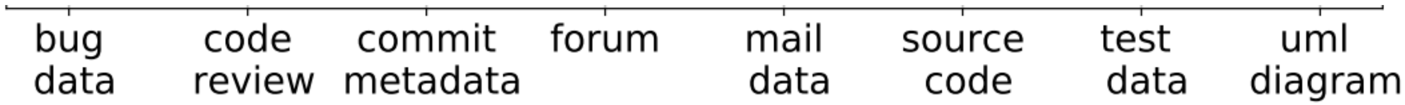
[Abou Khalil and Zacchiroli, ESEM 2022]



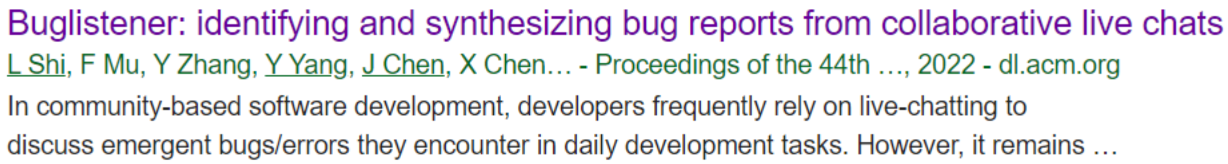
Software Artifact Mining



Software Artifact ~~Mining~~ Generation



Software Artifact ~~Mining~~ Generation

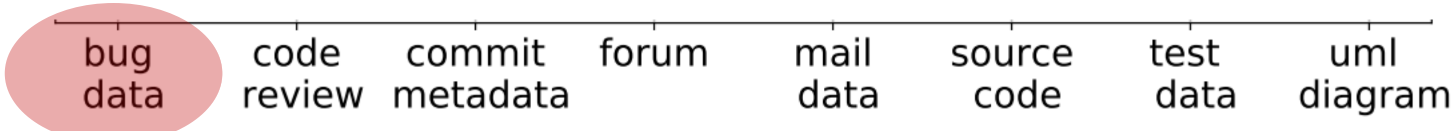


Buglistener: identifying and synthesizing bug reports from collaborative live chats

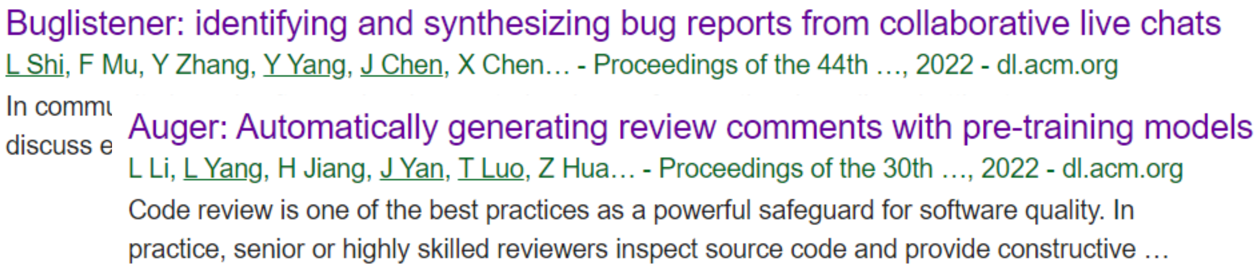
L Shi, F Mu, Y Zhang, Y Yang, J Chen, X Chen… - Proceedings of the 44th …, 2022 - dl.acm.org

In community-based software development, developers frequently rely on live-chatting to discuss emergent bugs/errors they encounter in daily development tasks. However, it remains …

[ICSE 2022]



Software Artifact ~~Mining~~ Generation



Buglistener: identifying and synthesizing bug reports from collaborative live chats

L Shi, F Mu, Y Zhang, Y Yang, J Chen, X Chen… - Proceedings of the 44th …, 2022 - dl.acm.org

In commu

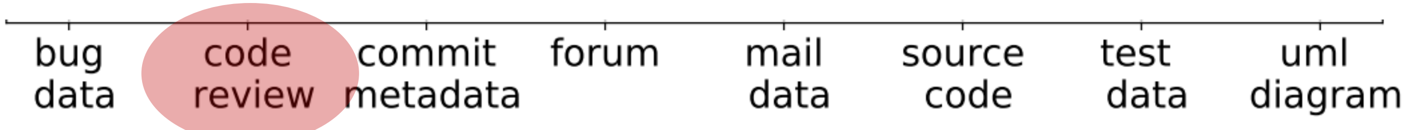
discuss e

Auger: Automatically generating review comments with pre-training models

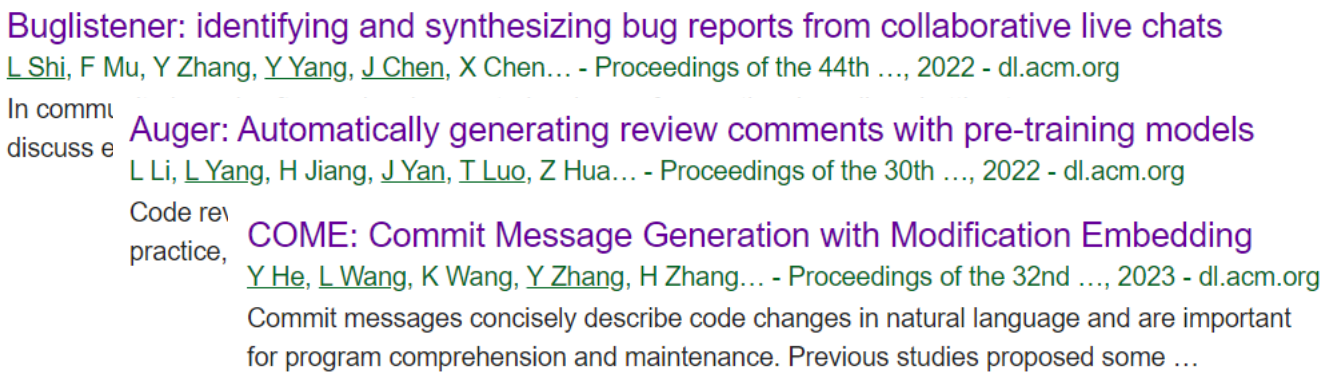
L Li, L Yang, H Jiang, J Yan, T Luo, Z Hua… - Proceedings of the 30th …, 2022 - dl.acm.org

Code review is one of the best practices as a powerful safeguard for software quality. In practice, senior or highly skilled reviewers inspect source code and provide constructive …

[ESEC/FSE 2022]



Software Artifact ~~Mining~~ Generation



Buglistener: identifying and synthesizing bug reports from collaborative live chats

L Shi, F Mu, Y Zhang, Y Yang, J Chen, X Chen… - Proceedings of the 44th …, 2022 - dl.acm.org

In commu

discuss e

Auger: Automatically generating review comments with pre-training models

L Li, L Yang, H Jiang, J Yan, T Luo, Z Hua… - Proceedings of the 30th …, 2022 - dl.acm.org

Code rev

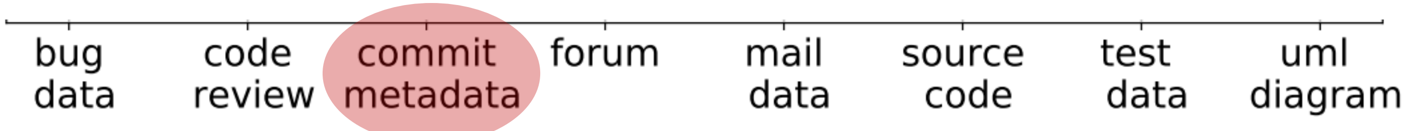
practice,

COME: Commit Message Generation with Modification Embedding

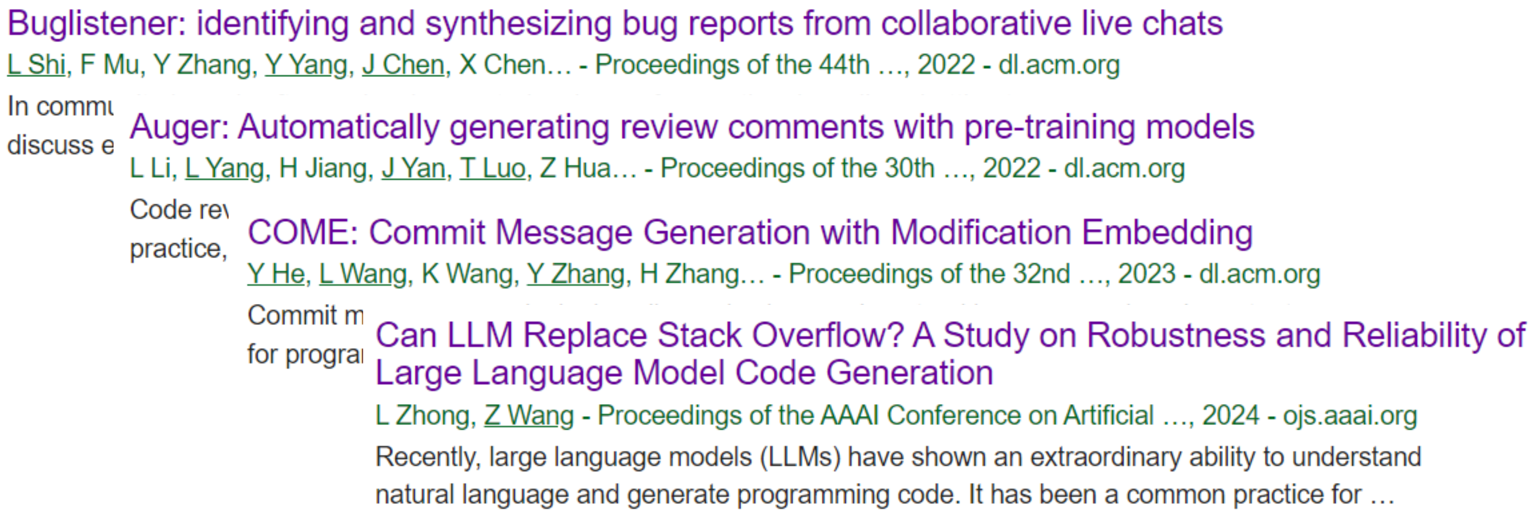
Y He, L Wang, K Wang, Y Zhang, H Zhang… - Proceedings of the 32nd …, 2023 - dl.acm.org

Commit messages concisely describe code changes in natural language and are important for program comprehension and maintenance. Previous studies proposed some …

[ISSTA 2023]



Software Artifact ~~Mining~~ Generation



Buglistener: identifying and synthesizing bug reports from collaborative live chats

L Shi, F Mu, Y Zhang, Y Yang, J Chen, X Chen… - Proceedings of the 44th …, 2022 - dl.acm.org

In commu

discuss e

Auger: Automatically generating review comments with pre-training models

L Li, L Yang, H Jiang, J Yan, T Luo, Z Hua… - Proceedings of the 30th …, 2022 - dl.acm.org

Code re

practice,

COME: Commit Message Generation with Modification Embedding

Y He, L Wang, K Wang, Y Zhang, H Zhang… - Proceedings of the 32nd …, 2023 - dl.acm.org

Commit m

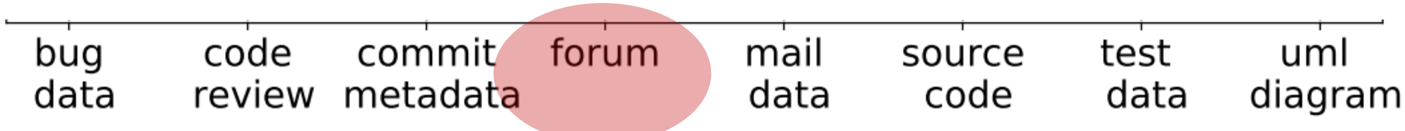
for progra

Can LLM Replace Stack Overflow? A Study on Robustness and Reliability of Large Language Model Code Generation

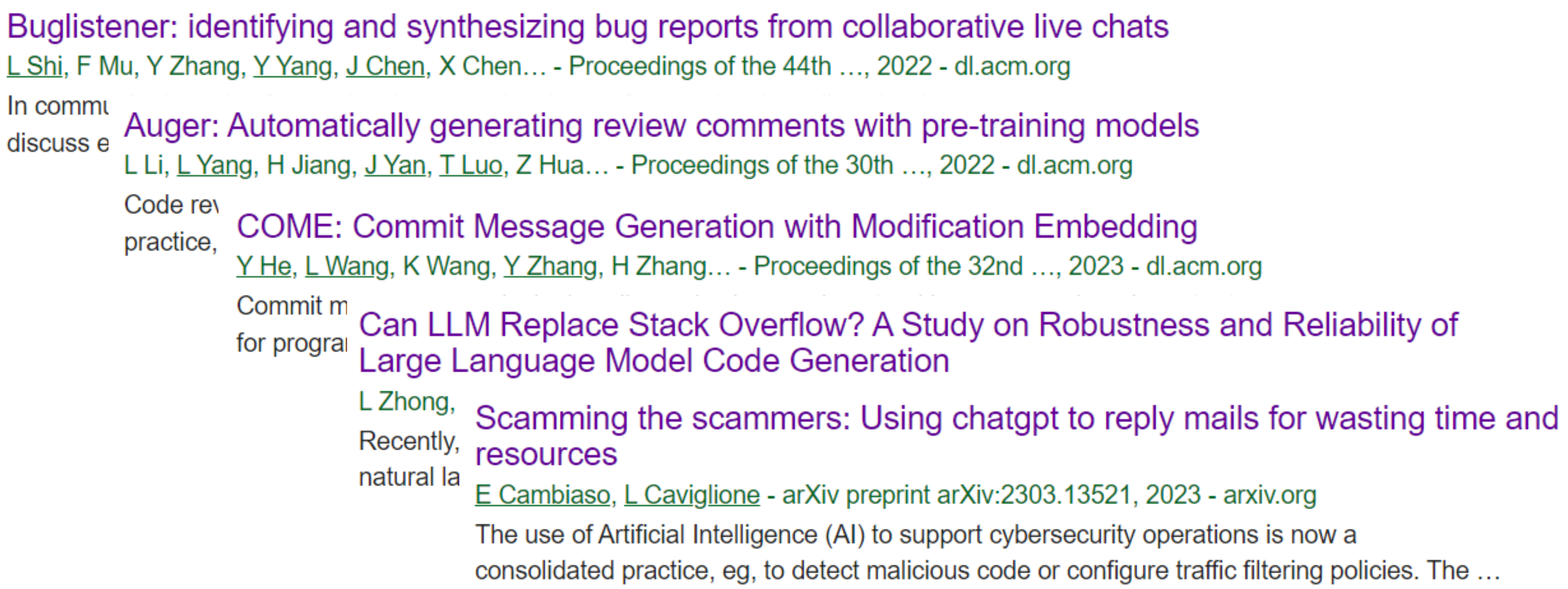
L Zhong, Z Wang - Proceedings of the AAAI Conference on Artificial …, 2024 - ojs.aaai.org

Recently, large language models (LLMs) have shown an extraordinary ability to understand natural language and generate programming code. It has been a common practice for …

[AAAI 2024]



Software Artifact ~~Mining~~ Generation



Buglistener: identifying and synthesizing bug reports from collaborative live chats

L Shi, F Mu, Y Zhang, Y Yang, J Chen, X Chen… - Proceedings of the 44th …, 2022 - dl.acm.org

In commu

discuss e

Auger: Automatically generating review comments with pre-training models

L Li, L Yang, H Jiang, J Yan, T Luo, Z Hua… - Proceedings of the 30th …, 2022 - dl.acm.org

Code rev

practice,

COME: Commit Message Generation with Modification Embedding

Y He, L Wang, K Wang, Y Zhang, H Zhang… - Proceedings of the 32nd …, 2023 - dl.acm.org

Commit m

for progra

Can LLM Replace Stack Overflow? A Study on Robustness and Reliability of Large Language Model Code Generation

L Zhong,

Recently,

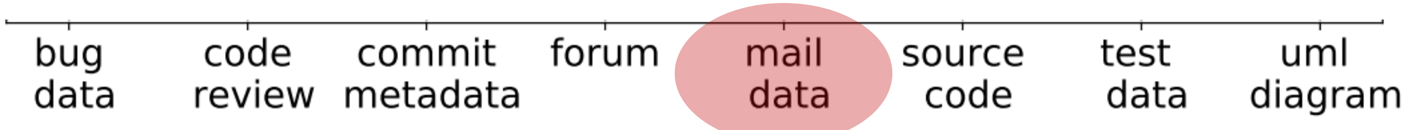
natural la

Scamming the scammers: Using chatgpt to reply mails for wasting time and resources

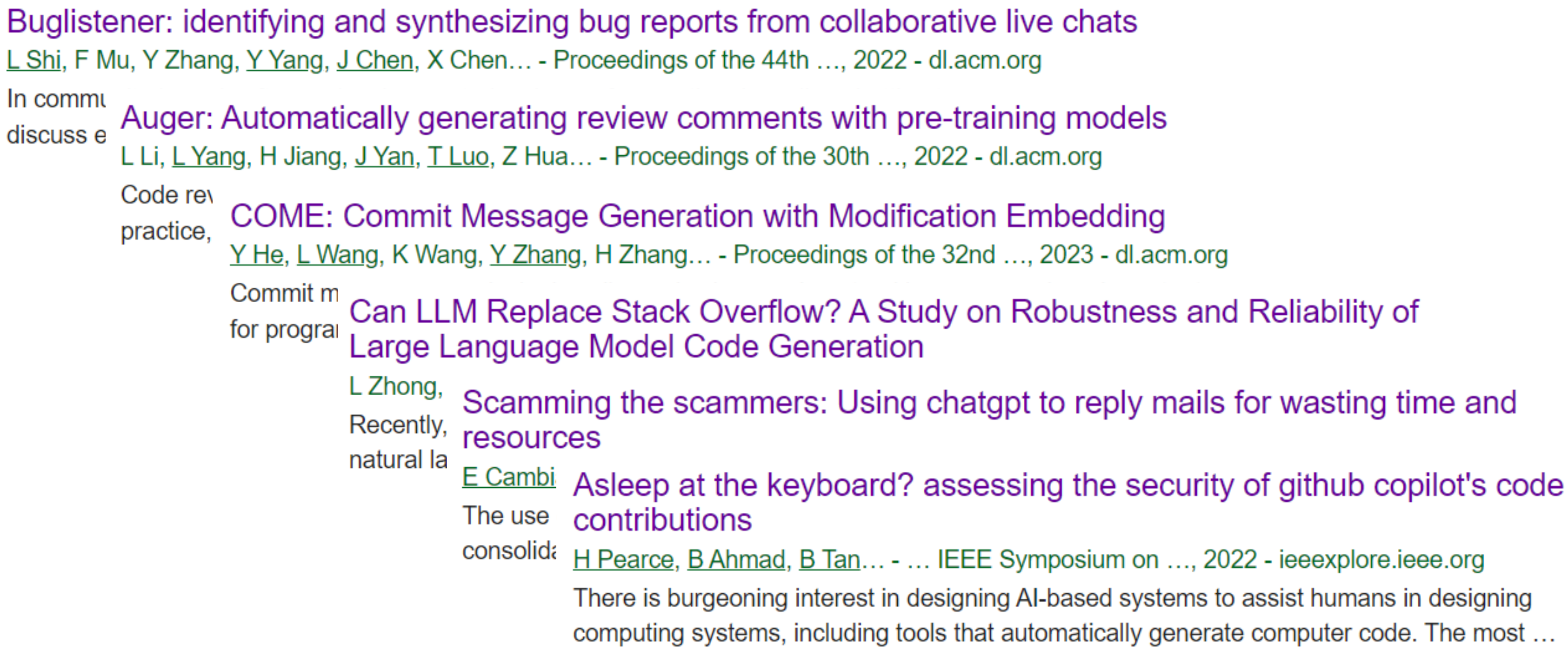
E Cambiaso, L Caviglione - arXiv preprint arXiv:2303.13521, 2023 - arxiv.org

The use of Artificial Intelligence (AI) to support cybersecurity operations is now a consolidated practice, eg, to detect malicious code or configure traffic filtering policies. The …

[arXiv 2023]



Software Artifact ~~Mining~~ Generation



Buglistener: identifying and synthesizing bug reports from collaborative live chats
L Shi, F Mu, Y Zhang, Y Yang, J Chen, X Chen… - Proceedings of the 44th …, 2022 - dl.acm.org
In commu
discuss e

Auger: Automatically generating review comments with pre-training models
L Li, L Yang, H Jiang, J Yan, T Luo, Z Hua… - Proceedings of the 30th …, 2022 - dl.acm.org
Code rev
practice,

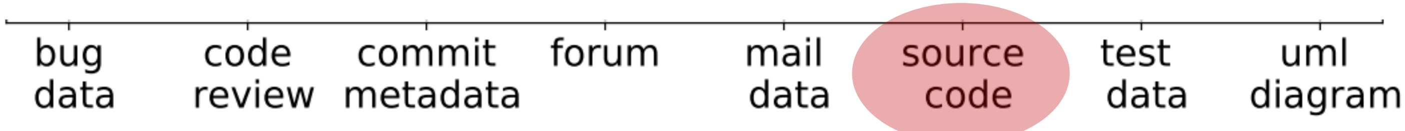
COME: Commit Message Generation with Modification Embedding
Y He, L Wang, K Wang, Y Zhang, H Zhang… - Proceedings of the 32nd …, 2023 - dl.acm.org
Commit m
for progra

Can LLM Replace Stack Overflow? A Study on Robustness and Reliability of Large Language Model Code Generation
L Zhong,
Recently,
natural la

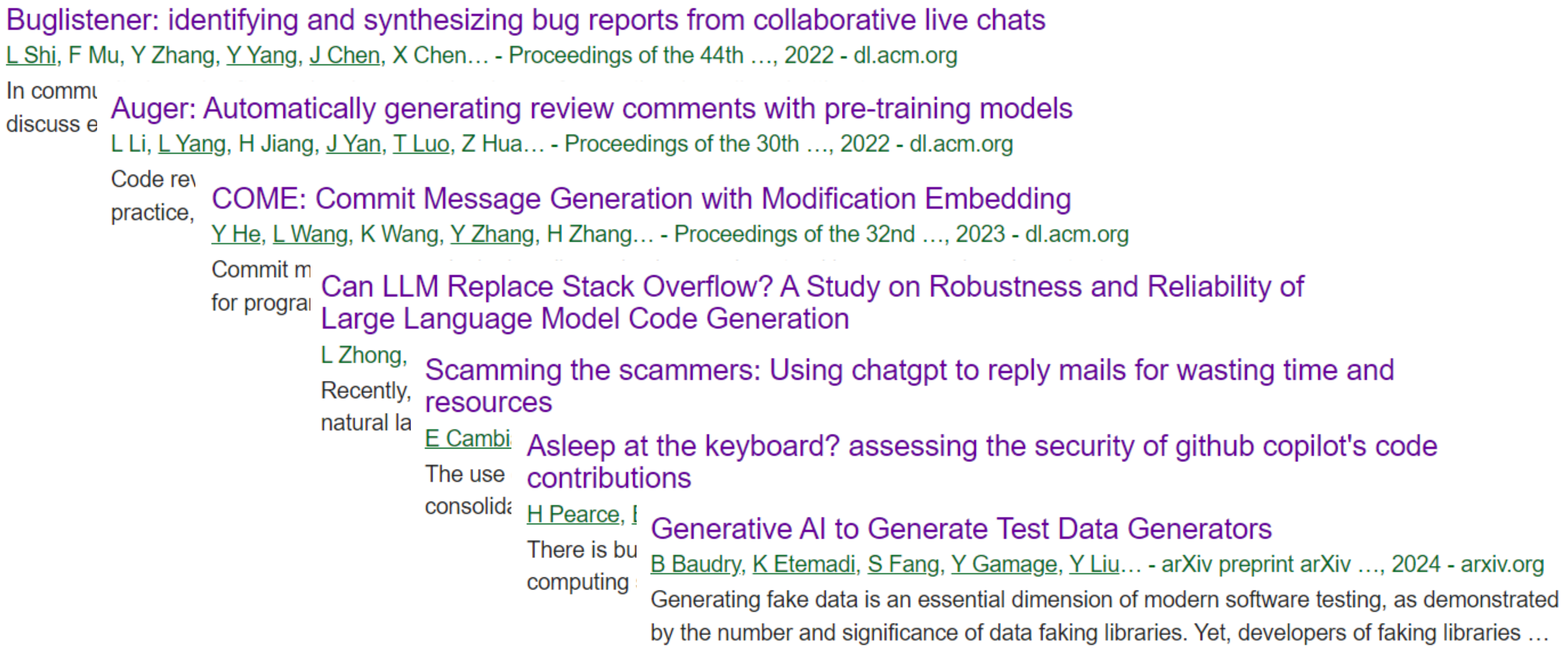
Scamming the scammers: Using chatgpt to reply mails for wasting time and resources
E Cambi
The use
consolid

Asleep at the keyboard? assessing the security of github copilot's code contributions
H Pearce, B Ahmad, B Tan… - … IEEE Symposium on …, 2022 - ieeexplore.ieee.org
There is burgeoning interest in designing AI-based systems to assist humans in designing computing systems, including tools that automatically generate computer code. The most …

[SP 2022]



Software Artifact ~~Mining~~ Generation



Buglistener: identifying and synthesizing bug reports from collaborative live chats
L Shi, F Mu, Y Zhang, Y Yang, J Chen, X Chen… - Proceedings of the 44th …, 2022 - dl.acm.org
In commu
discuss e

Auger: Automatically generating review comments with pre-training models
L Li, L Yang, H Jiang, J Yan, T Luo, Z Hua… - Proceedings of the 30th …, 2022 - dl.acm.org
Code rev
practice,

COME: Commit Message Generation with Modification Embedding
Y He, L Wang, K Wang, Y Zhang, H Zhang… - Proceedings of the 32nd …, 2023 - dl.acm.org
Commit m
for progra

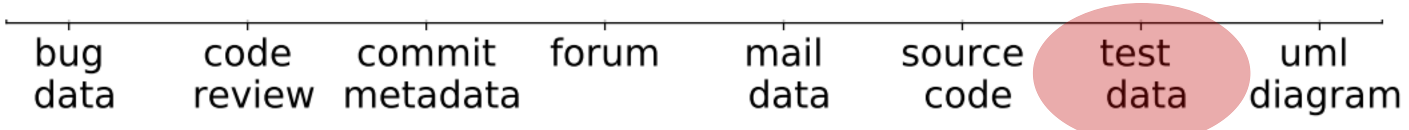
Can LLM Replace Stack Overflow? A Study on Robustness and Reliability of Large Language Model Code Generation
L Zhong,
Recently,
natural la

Scamming the scammers: Using chatgpt to reply mails for wasting time and resources
E Cambi
The use
consolid

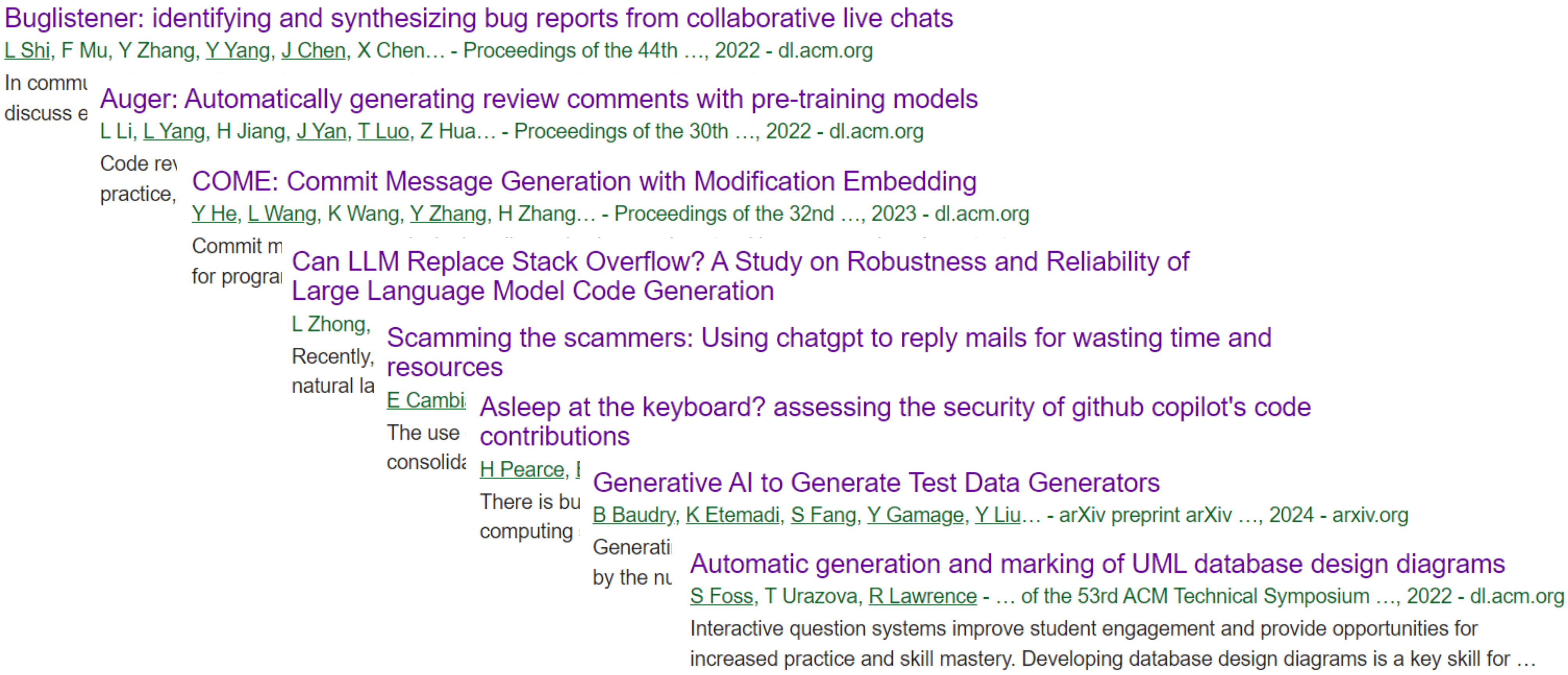
Asleep at the keyboard? assessing the security of github copilot's code contributions
H Pearce,
There is bu
computing

Generative AI to Generate Test Data Generators
B Baudry, K Etemadi, S Fang, Y Gamage, Y Liu… - arXiv preprint arXiv …, 2024 - arxiv.org
Generating fake data is an essential dimension of modern software testing, as demonstrated by the number and significance of data faking libraries. Yet, developers of faking libraries …

[arXiv 2024]



Software Artifact ~~Mining~~ Generation



Buglistener: identifying and synthesizing bug reports from collaborative live chats
L Shi, F Mu, Y Zhang, Y Yang, J Chen, X Chen… - Proceedings of the 44th …, 2022 - dl.acm.org
In commu
discuss e

Auger: Automatically generating review comments with pre-training models
L Li, L Yang, H Jiang, J Yan, T Luo, Z Hua… - Proceedings of the 30th …, 2022 - dl.acm.org
Code rev
practice,

COME: Commit Message Generation with Modification Embedding
Y He, L Wang, K Wang, Y Zhang, H Zhang… - Proceedings of the 32nd …, 2023 - dl.acm.org
Commit m
for progra

Can LLM Replace Stack Overflow? A Study on Robustness and Reliability of Large Language Model Code Generation
L Zhong,
Recently,
natural la

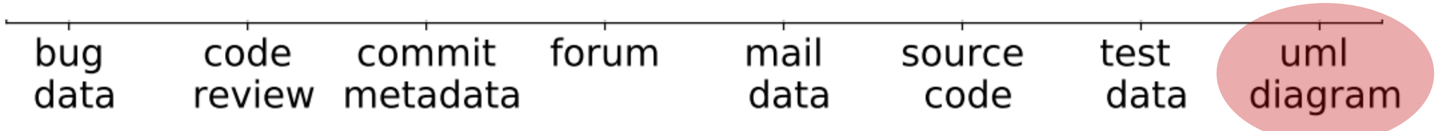
Scamming the scammers: Using chatgpt to reply mails for wasting time and resources
E Cambi
The use
consolida

Asleep at the keyboard? assessing the security of github copilot's code contributions
H Pearce,
There is bu
computing

Generative AI to Generate Test Data Generators
B Baudry, K Etemadi, S Fang, Y Gamage, Y Liu… - arXiv preprint arXiv …, 2024 - arxiv.org
Generati
by the nu

Automatic generation and marking of UML database design diagrams
S Foss, T Urazova, R Lawrence - … of the 53rd ACM Technical Symposium …, 2022 - dl.acm.org
Interactive question systems improve student engagement and provide opportunities for increased practice and skill mastery. Developing database design diagrams is a key skill for …

[SIGCSE 2022]



The Large Language Models Era

> “Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis.”
>
> [Harrison and Basili]

The Large Language Models Era

“Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis.”
[Harrison and Basili]

Are we just going to analyze LLM-generated output?

The Large Language Models Era

“Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis.”

[Harrison and Basili]

Are we just going to analyze LLM-generated output?

What is the role of Empirical Software Engineering Research in the LLM Era?

The Large Language Models Era

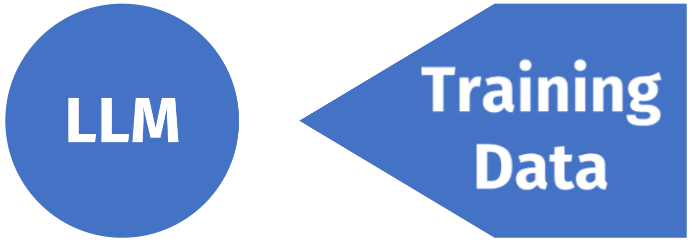
LLM

> “Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis.”
>
> [Harrison and Basili]

Are we just going to analyze LLM-generated output?

What is the role of Empirical Software Engineering Research in the LLM Era?

The Large Language Models Era



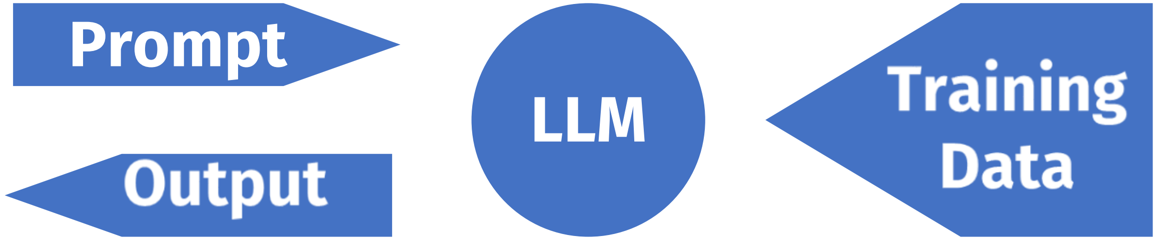
“Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis.”

[Harrison and Basili]

Are we just going to analyze LLM-generated output?

What is the role of Empirical Software Engineering Research in the LLM Era?

The Large Language Models Era



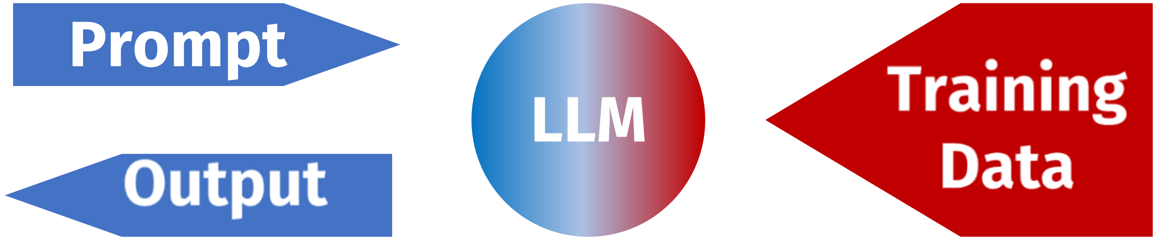
"Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis."

[Harrison and Basili]

Are we just going to analyze LLM-generated output?

What is the role of Empirical Software Engineering Research in the LLM Era?

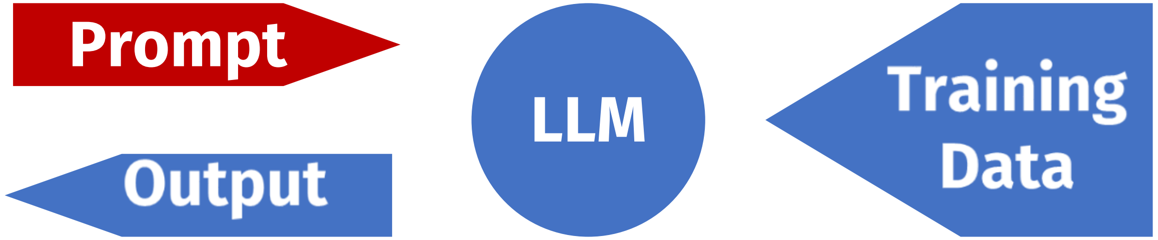
Analyzing the rich data available



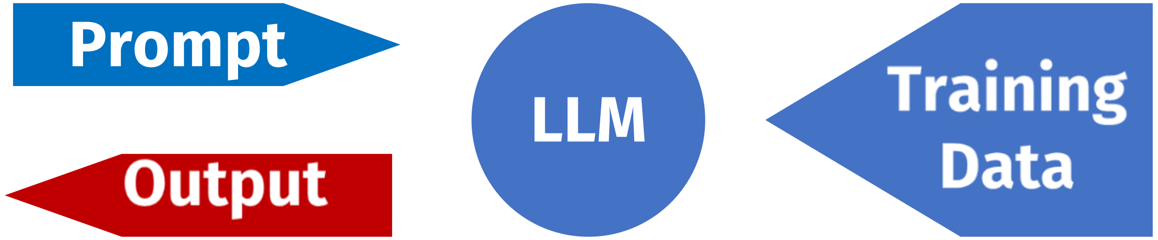
1) Analyze the training data and its impact on the LLM

Analyzing the rich data available

2) Analyze interaction with LLMs

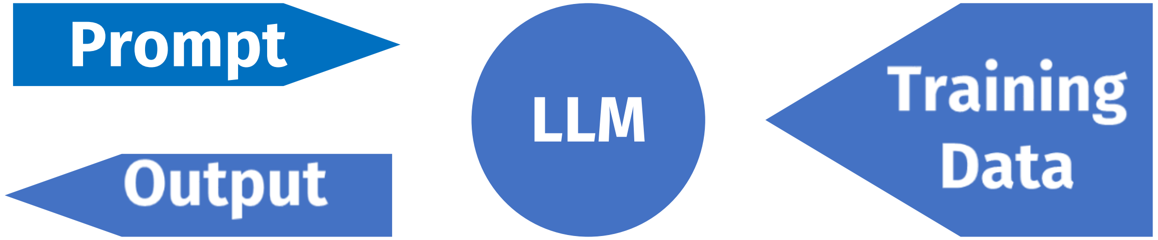


Analyzing the rich data available



3) Analyze LLM output

Analyzing the rich data available


Prompt

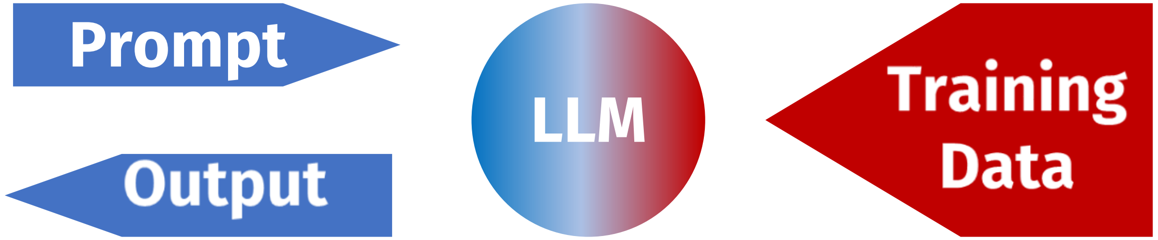
Output

LLM

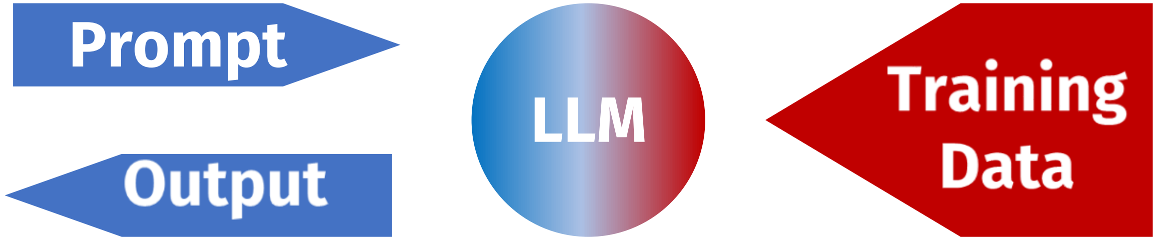
Training Data

4) The bigger picture?

Analyzing the training data & its impact



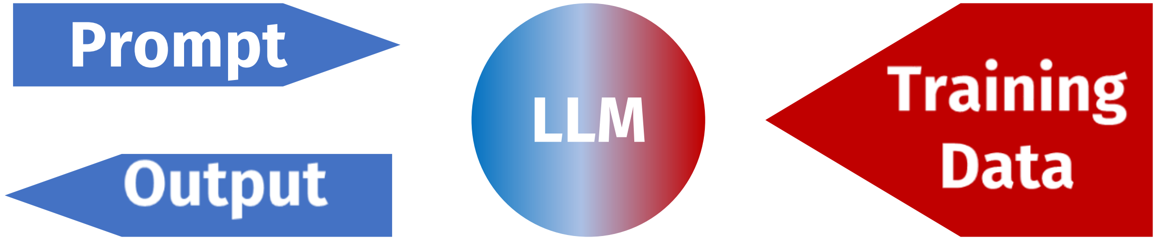
Analyzing the training data & its impact



[Treude and Hata, MSR 2023]

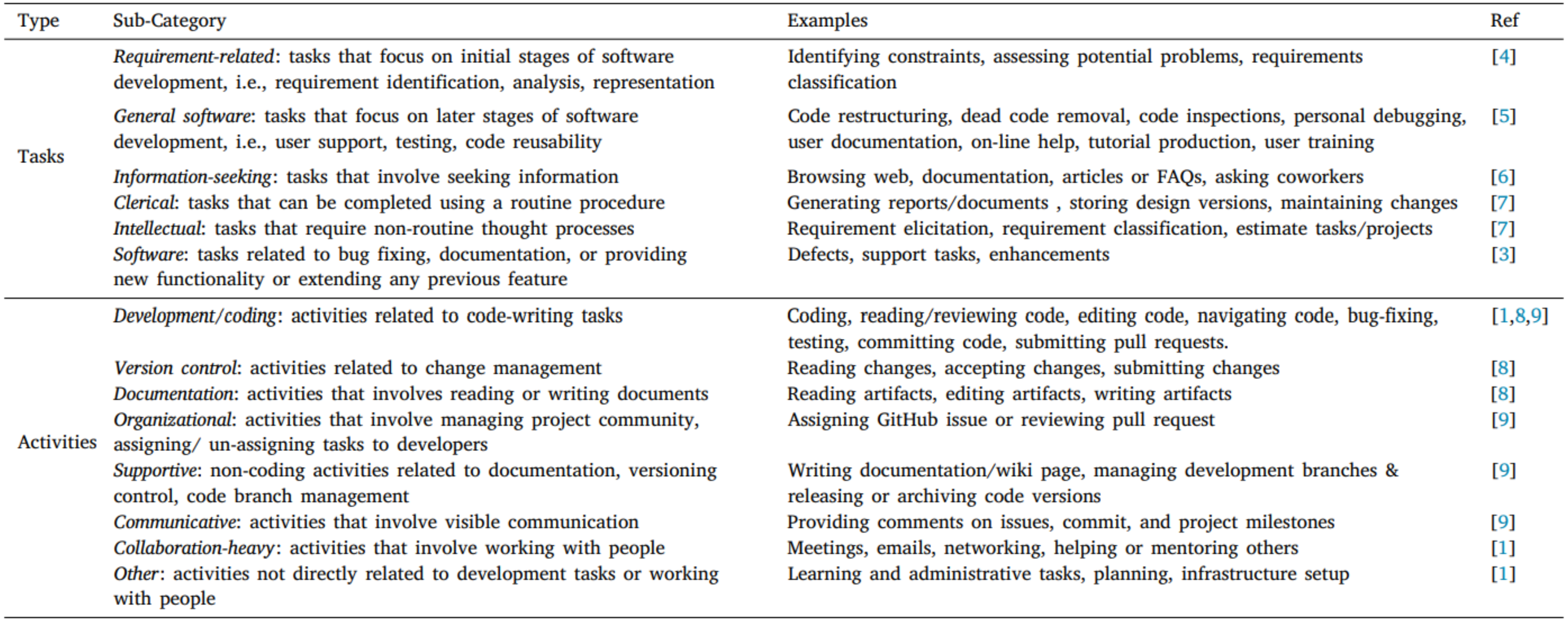
Analyzing the training data & its impact

require high standards and timeliness but offer
little substantive development or visibility

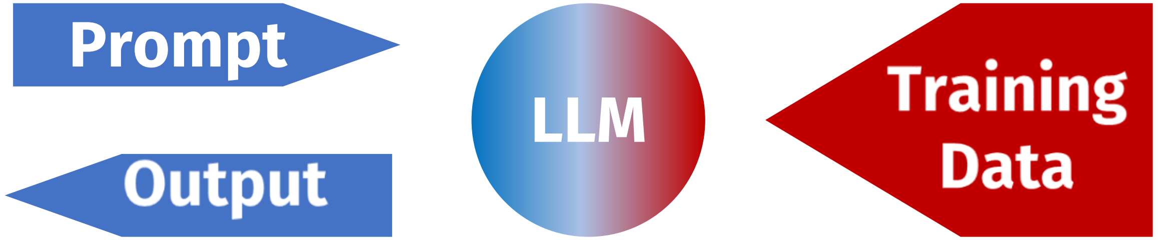

Prompt
Output
LLM
Training Data

[Treude and Hata, MSR 2023]

Analyzing the training data & its impact



Type	Sub-Category	Examples	Ref
Tasks	*Requirement-related*: tasks that focus on initial stages of software development, i.e., requirement identification, analysis, representation	Identifying constraints, assessing potential problems, requirements classification	[4]
	General software: tasks that focus on later stages of software development, i.e., user support, testing, code reusability	Code restructuring, dead code removal, code inspections, personal debugging, user documentation, on-line help, tutorial production, user training	[5]
	Information-seeking: tasks that involve seeking information	Browsing web, documentation, articles or FAQs, asking coworkers	[6]
	Clerical: tasks that can be completed using a routine procedure	Generating reports/documents , storing design versions, maintaining changes	[7]
	Intellectual: tasks that require non-routine thought processes	Requirement elicitation, requirement classification, estimate tasks/projects	[7]
	Software: tasks related to bug fixing, documentation, or providing new functionality or extending any previous feature	Defects, support tasks, enhancements	[3]
Activities	*Development/coding*: activities related to code-writing tasks	Coding, reading/reviewing code, editing code, navigating code, bug-fixing, testing, committing code, submitting pull requests.	[1,8,9]
	Version control: activities related to change management	Reading changes, accepting changes, submitting changes	[8]
	Documentation: activities that involves reading or writing documents	Reading artifacts, editing artifacts, writing artifacts	[8]
	Organizational: activities that involve managing project community, assigning/ un-assigning tasks to developers	Assigning GitHub issue or reviewing pull request	[9]
	Supportive: non-coding activities related to documentation, versioning control, code branch management	Writing documentation/wiki page, managing development branches & releasing or archiving code versions	[9]
	Communicative: activities that involve visible communication	Providing comments on issues, commit, and project milestones	[9]
	Collaboration-heavy: activities that involve working with people	Meetings, emails, networking, helping or mentoring others	[1]
	Other: activities not directly related to development tasks or working with people	Learning and administrative tasks, planning, infrastructure setup	[1]



Prompt

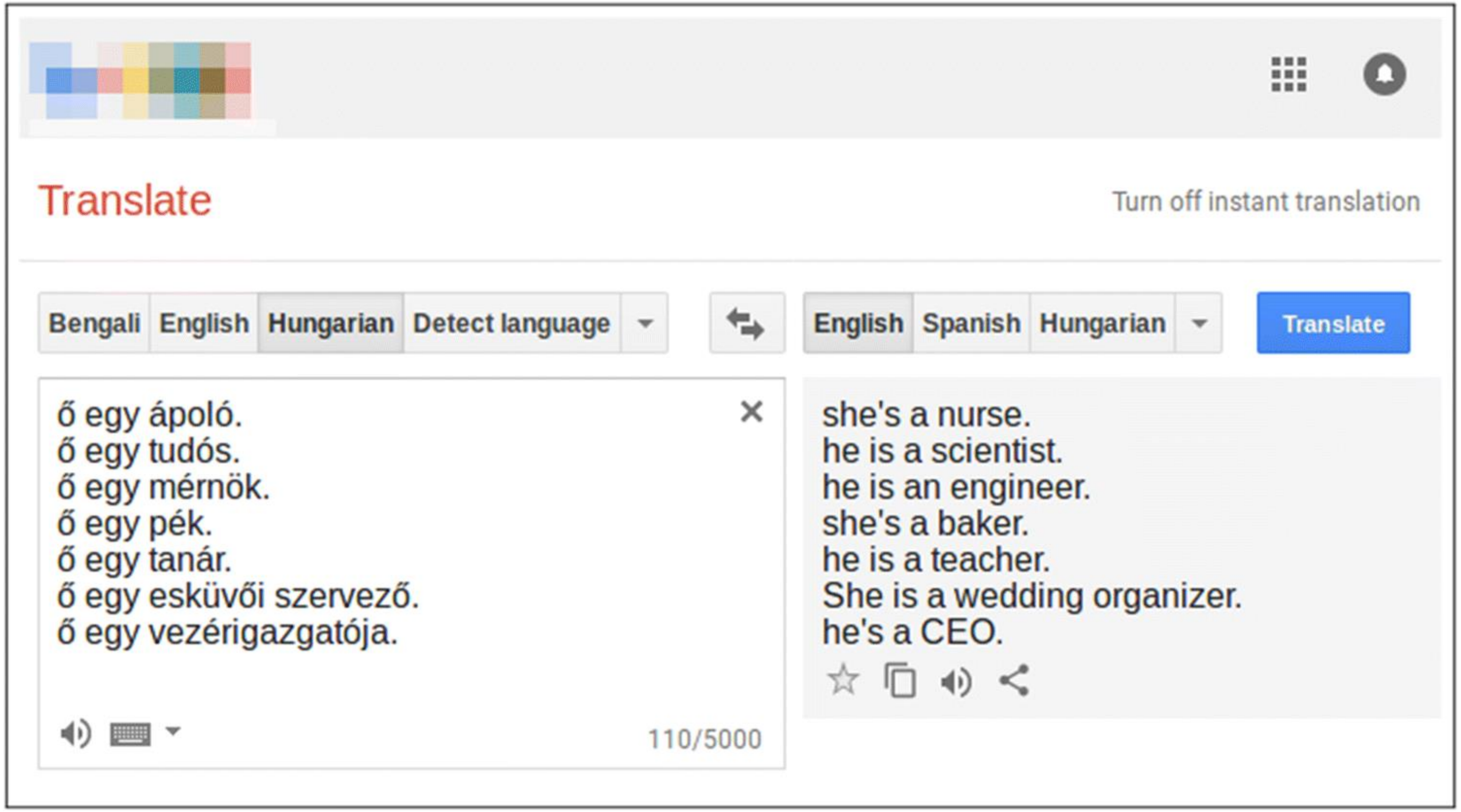
Output

LLM

Training Data

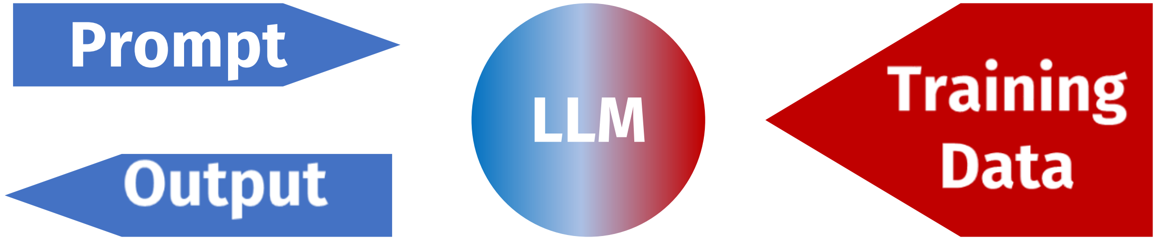
[Masood et al., IST 2022]

Analyzing the training data & its impact



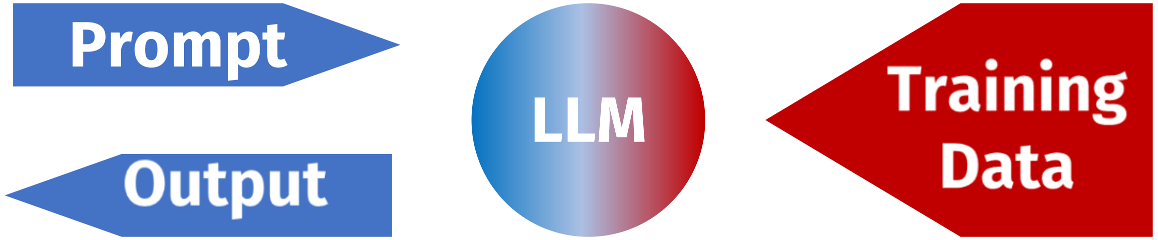
ő egy ápoló.
ő egy tudós.
ő egy mérnök.
ő egy pék.
ő egy tanár.
ő egy esküvői szervező.
ő egy vezérigazgatója.

she's a nurse.
he is a scientist.
he is an engineer.
she's a baker.
he is a teacher.
She is a wedding organizer.
he's a CEO.



[Treude and Hata, MSR 2023]

As a software engineer, she identifies constraints.
As a software engineer, she assesses potential problems.
As a software engineer, she classifies requirements.
As a software engineer, she restructures code.
As a software engineer, she removes dead code.
As a software engineer, she inspects code.
As a software engineer, she performs personal debugging.
As a software engineer, she produces user documentation.
As a software engineer, she produces on-line help.
As a software engineer, she produces tutorials.
As a software engineer, she performs user training.
As a software engineer, she browses the web.
As a software engineer, she browses documentation.
As a software engineer, she browses articles.
As a software engineer, she browses FAQs.
As a software engineer, she asks coworkers.
As a software engineer, she generates reports/documents.
As a software engineer, she stores design versions.
As a software engineer, she maintains changes.
As a software engineer, she elicits requirements.
As a software engineer, she estimates tasks/projects.
As a software engineer, she fixes defects.
As a software engineer, she performs support tasks.
As a software engineer, she provides enhancements.
As a software engineer, she codes.
As a software engineer, she reads/reviews code.
As a software engineer, she edits code.
As a software engineer, she navigates code.
As a software engineer, she fixes bugs.
As a software engineer, she tests.
As a software engineer, she commits code.
As a software engineer, she submits pull requests.
As a software engineer, she reads changes.
As a software engineer, she accepts changes.
As a software engineer, she submits changes.
As a software engineer, she reads artifacts.
As a software engineer, she edits artifacts.
As a software engineer, she writes artifacts.
As a software engineer, she assigns GitHub issues.
As a software engineer, she reviews pull requests.
As a software engineer, she writes documentation/wiki pages.
As a software engineer, she manages development branches.
As a software engineer, she releases code versions.
As a software engineer, she archives code versions.
As a software engineer, she provides comments on issues.
As a software engineer, she provides comments on commits.
As a software engineer, she provides comments on project milestones.
As a software engineer, she has meetings.
As a software engineer, she writes emails.
As a software engineer, she networks.
As a software engineer, she helps others.
As a software engineer, she mentors others.
As a software engineer, she learns.
As a software engineer, she performs administrative tasks.
As a software engineer, she plans.
As a software engineer, she performs infrastructure setup.



Prompt

LLM

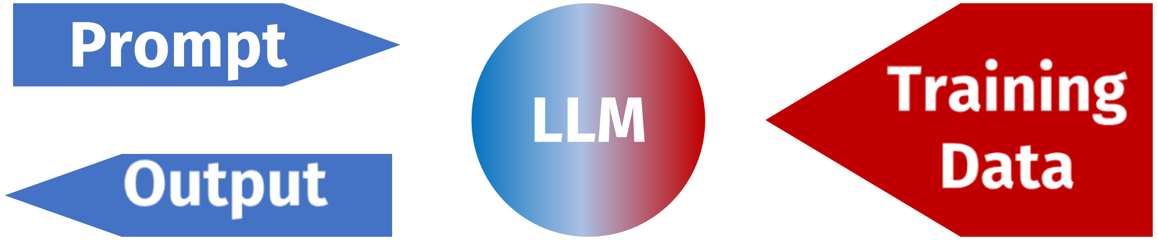
Training Data

Output

As a software engineer, she identifies constraints.
As a software engineer, she assesses potential problems.
As a software engineer, she classifies requirements.
As a software engineer, she restructures code.
As a software engineer, she removes dead code.
As a software engineer, she inspects code.
As a software engineer, she performs personal debugging.
As a software engineer, she produces user documentation.
As a software engineer, she produces on-line help.
As a software engineer, she produces tutorials.
As a software engineer, she performs user training.
As a software engineer, she browses the web.
As a software engineer, she browses documentation.
As a software engineer, she browses articles.
As a software engineer, she browses FAQs.
As a software engineer, she asks coworkers.
As a software engineer, she generates reports/documents.
As a software engineer, she stores design versions.
As a software engineer, she maintains changes.
As a software engineer, she elicits requirements.
As a software engineer, she estimates tasks/projects.
As a software engineer, she fixes defects.
As a software engineer, she performs support tasks.
As a software engineer, she provides enhancements.
As a software engineer, she codes.
As a software engineer, she reads/reviews code.
As a software engineer, she edits code.
As a software engineer, she navigates code.
As a software engineer, she fixes bugs.
As a software engineer, she tests.
As a software engineer, she commits code.
As a software engineer, she submits pull requests.
As a software engineer, she reads changes.
As a software engineer, she accepts changes.
As a software engineer, she submits changes.
As a software engineer, she reads artifacts.
As a software engineer, she edits artifacts.
As a software engineer, she writes artifacts.
As a software engineer, she assigns GitHub issues.
As a software engineer, she reviews pull requests.
As a software engineer, she writes documentation/wiki pages.
As a software engineer, she manages development branches.
As a software engineer, she releases code versions.
As a software engineer, she archives code versions.
As a software engineer, she provides comments on issues.
As a software engineer, she provides comments on commits.
As a software engineer, she provides comments on project milestones.
As a software engineer, she has meetings.
As a software engineer, she writes emails.
As a software engineer, she networks.
As a software engineer, she helps others.
As a software engineer, she mentors others.
As a software engineer, she learns.
As a software engineer, she performs administrative tasks.
As a software engineer, she plans.
As a software engineer, she performs infrastructure setup.

As a software engineer, she elicits requirements.

As a software engineer, she tests.

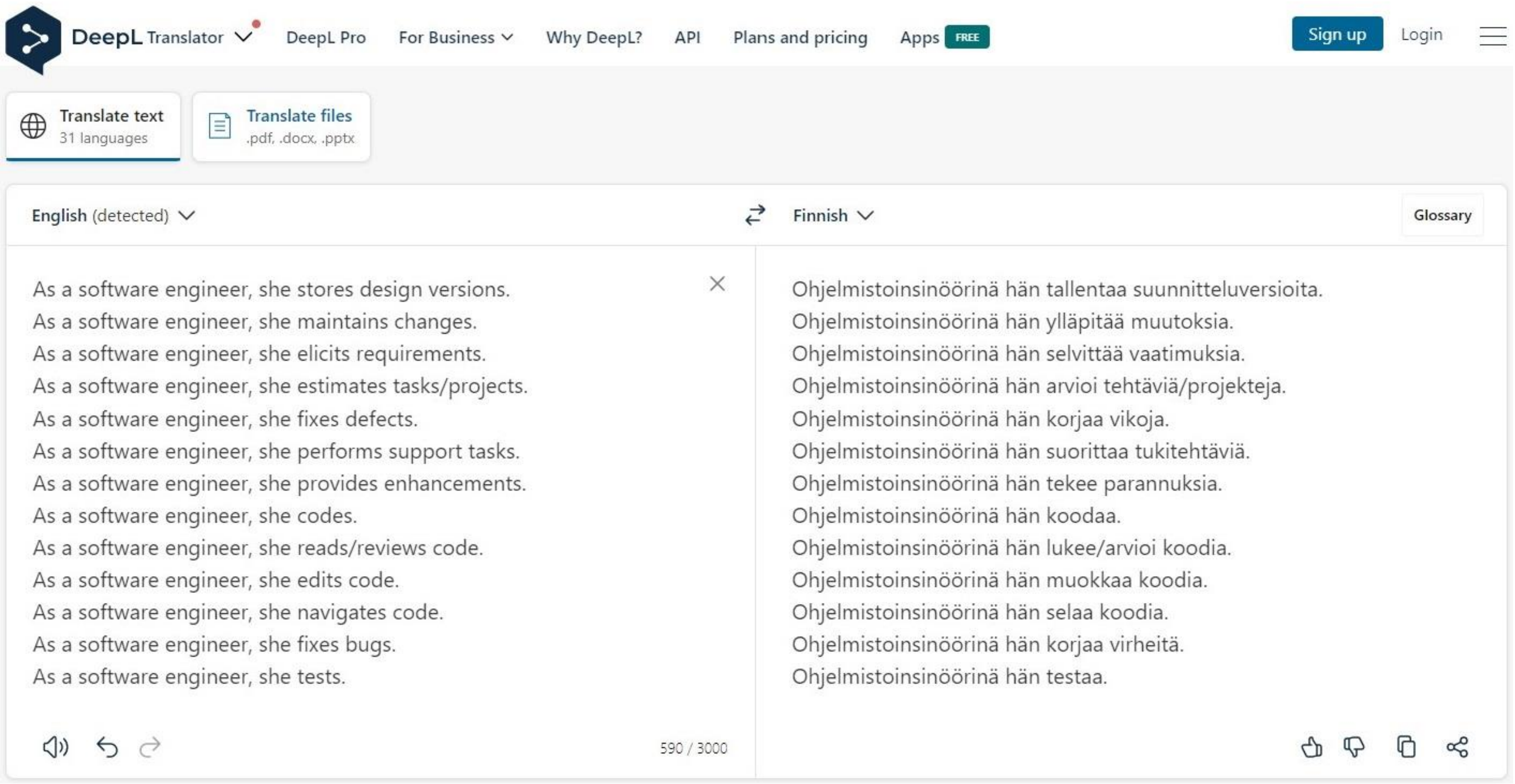

Prompt

Output

LLM

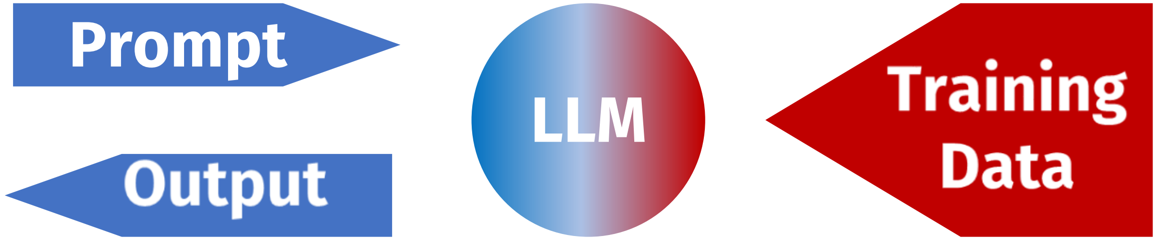
Training Data

Analyzing the training data & its impact



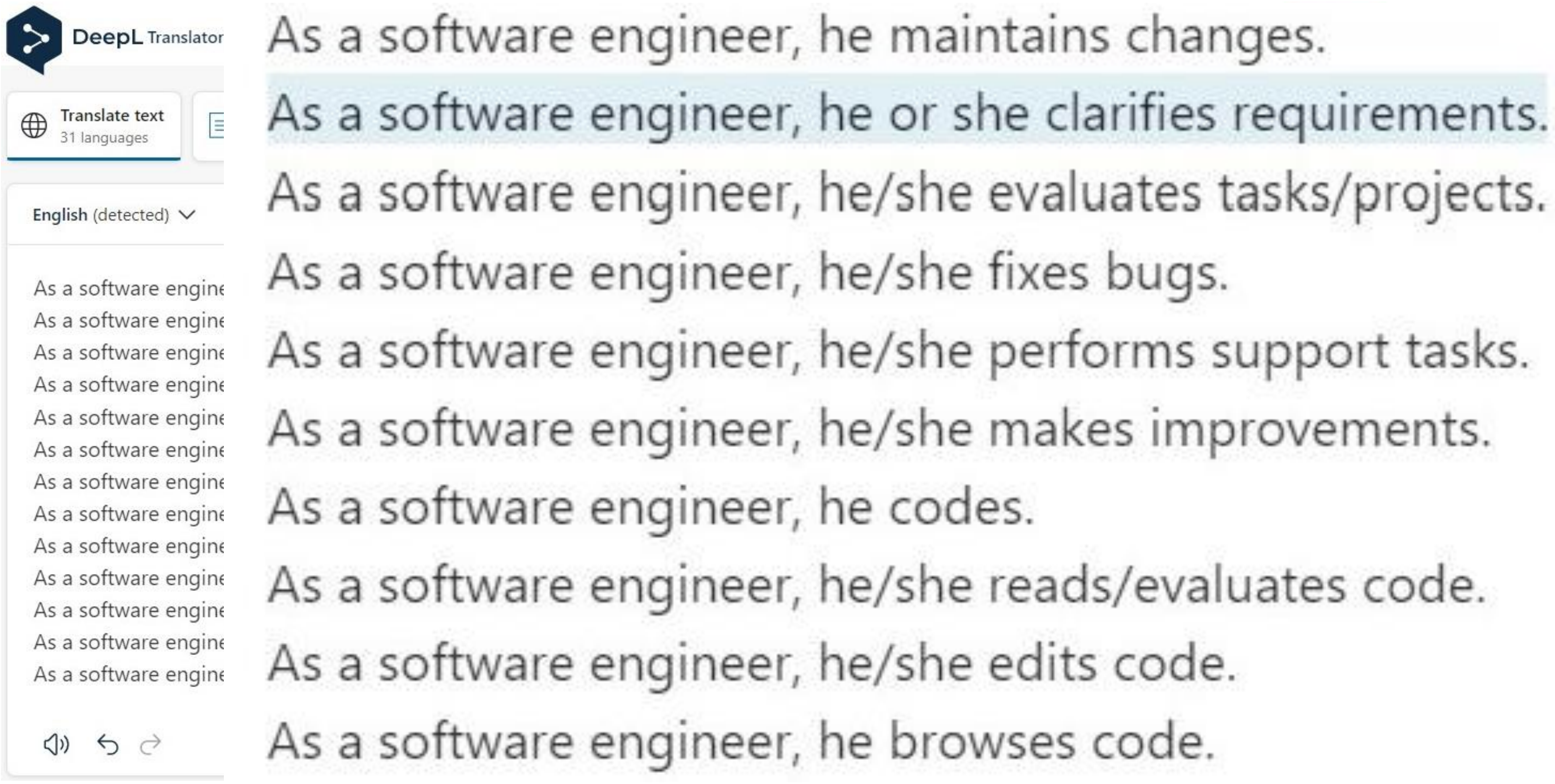
As a software engineer, she stores design versions.
As a software engineer, she maintains changes.
As a software engineer, she elicits requirements.
As a software engineer, she estimates tasks/projects.
As a software engineer, she fixes defects.
As a software engineer, she performs support tasks.
As a software engineer, she provides enhancements.
As a software engineer, she codes.
As a software engineer, she reads/reviews code.
As a software engineer, she edits code.
As a software engineer, she navigates code.
As a software engineer, she fixes bugs.
As a software engineer, she tests.

Ohjelmistoinsinöörinä hän tallentaa suunnitteluversioita.
Ohjelmistoinsinöörinä hän ylläpitää muutoksia.
Ohjelmistoinsinöörinä hän selvittää vaatimuksia.
Ohjelmistoinsinöörinä hän arvioi tehtäviä/projekteja.
Ohjelmistoinsinöörinä hän korjaa vikoja.
Ohjelmistoinsinöörinä hän suorittaa tukitehtäviä.
Ohjelmistoinsinöörinä hän tekee parannuksia.
Ohjelmistoinsinöörinä hän koodaa.
Ohjelmistoinsinöörinä hän lukee/arvioi koodia.
Ohjelmistoinsinöörinä hän muokkaa koodia.
Ohjelmistoinsinöörinä hän selaa koodia.
Ohjelmistoinsinöörinä hän korjaa virheitä.
Ohjelmistoinsinöörinä hän testaa.

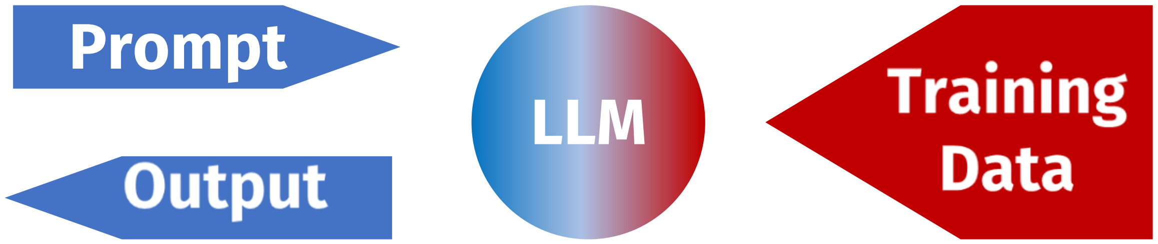


[Treude and Hata, MSR 2023]

Analyzing the training data & its impact



As a software engineer, he maintains changes.
As a software engineer, he or she clarifies requirements.
As a software engineer, he/she evaluates tasks/projects.
As a software engineer, he/she fixes bugs.
As a software engineer, he/she performs support tasks.
As a software engineer, he/she makes improvements.
As a software engineer, he codes.
As a software engineer, he/she reads/evaluates code.
As a software engineer, he/she edits code.
As a software engineer, he browses code.



Prompt

Output

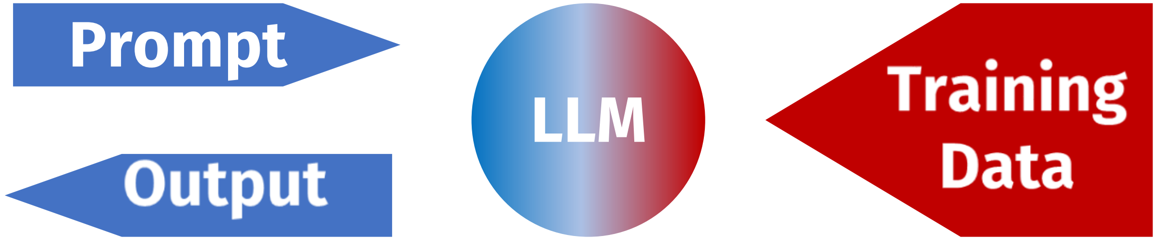
LLM

Training Data

[Treude and Hata, MSR 2023]

Analyzing the training data & its impact

Original Sentence	"she"	"he/she"	"he or she"	"he"
She elicits requirements.	0	51	43	6
She estimates tasks/projects.	0	61	0	39
She performs infrastructure setup.	0	39	14	47
She performs support tasks.	0	44	6	49
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…
She learns.	0	0	0	100
She provides comments on issues.	0	0	0	100
She tests.	0	0	0	100

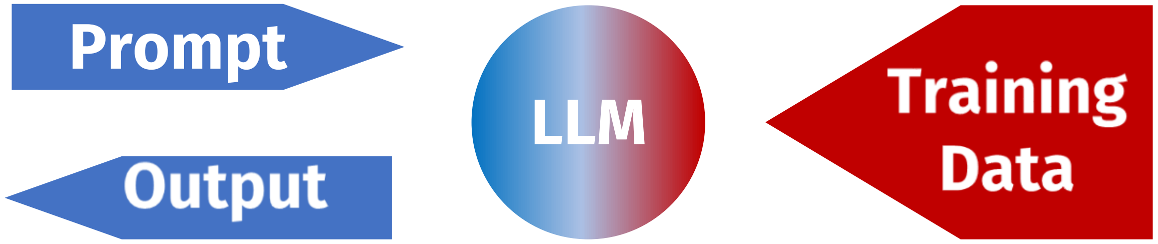


[Treude and Hata, MSR 2023]

Analyzing the training data & its impact

Heuristics don’t address the underlying problem

Lots of other potential biases, e.g., feature prioritization



[Treude and Hata, MSR 2023]

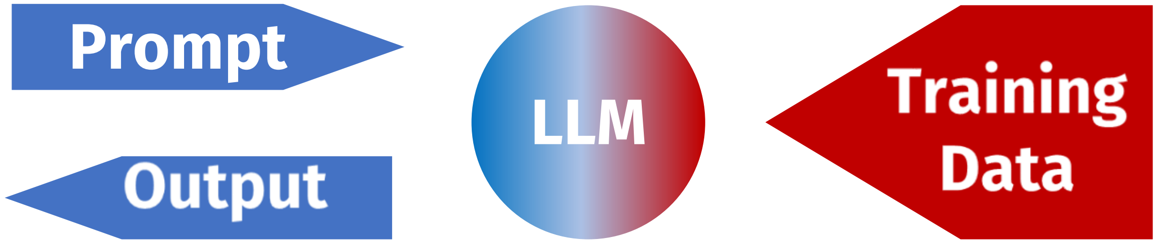
Analyzing the training data & its impact

Heuristics don't address the underlying problem

Consider ethical dilemmas

Lots of other potential biases, e.g., feature prioritization

Causal relationships between training data and LLMs



[Treude and Hata, MSR 2023]

Analyzing the training data & its impact

Heuristics don’t address the underlying problem

Consider ethical dilemmas

Lots of oth
potential bia
e.g., featu
prioritizati

Nigerian Software Engineer or American Data Scientist? GitHub Profile Recruitment Bias in Large Language Models

Takashi Nakano
Nara Institute of Science and Technology
Japan
nakano.takashi.nr1@is.naist.jp

Kazumasa Shimari
Nara Institute of Science and Technology
Japan
k.shimari@is.naist.jp

Raula Gaikovina Kula
Osaka University
Japan
raula-k@is.naist.jp

Christoph Treude
Singapore Management University
Singapore
ctreude@smu.edu.sg

Marc Cheong
The University of Melbourne
Australia
marc.cheong@unimelb.edu.au

Kenichi Matsumoto
Nara Institute of Science and Technology
Japan
matumoto@is.naist.jp

[Nakano et al., ICSME 2024]

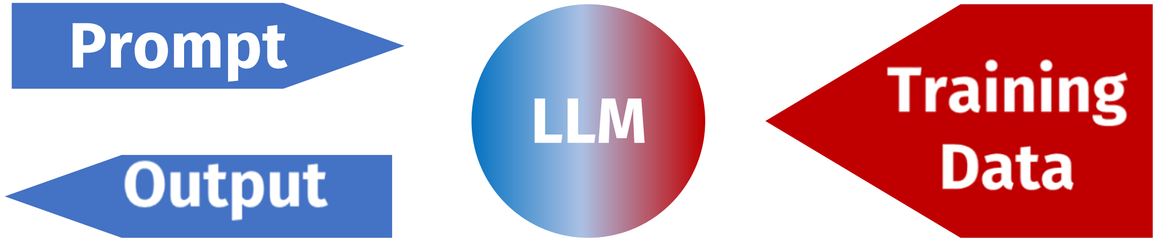
Analyzing the training data & its impact

Heuristics don't address the underlying problem

Consider ethical dilemmas

Lots of other potential biases, e.g., feature prioritization

Causal relationships between training data and LLMs



[Treude and Hata, MSR 2023]

Analyzing the training data & its impact

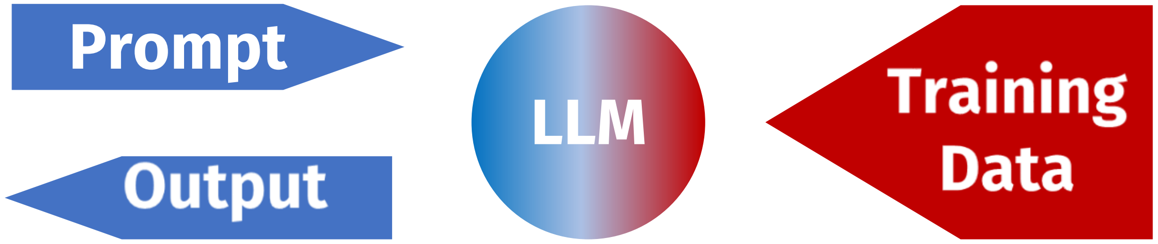
Heuristics don’t
address the
underlying
problem

C r ethical
lemmas

We know the data!

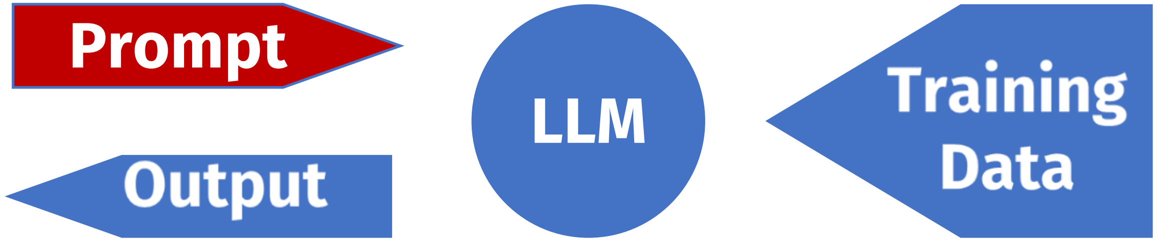
Lots of ot
potential s,
e.g., feat re
prioritization

Causal
relationships
between training
data and LLMs

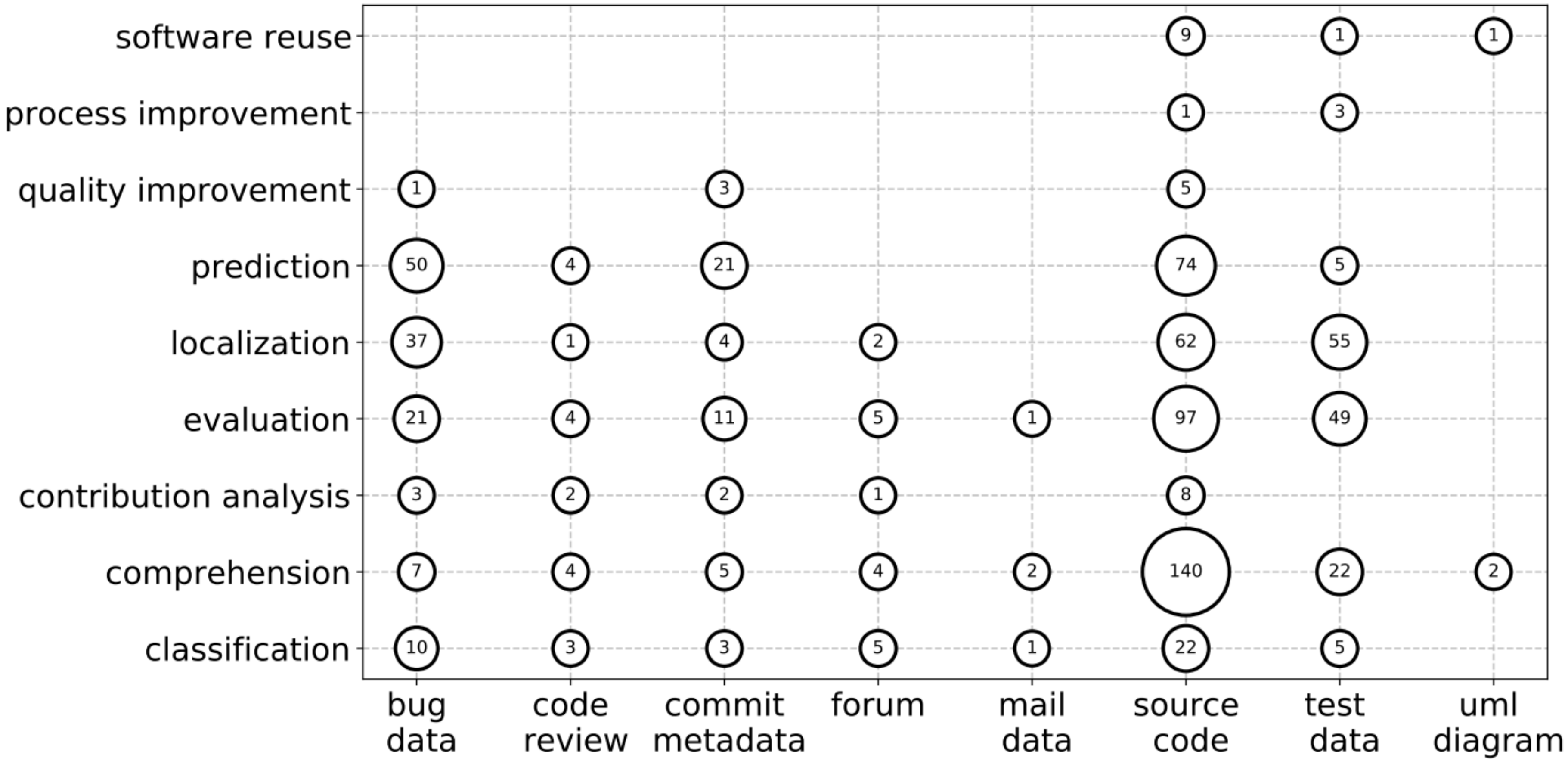


[Treude and Hata, MSR 2023]

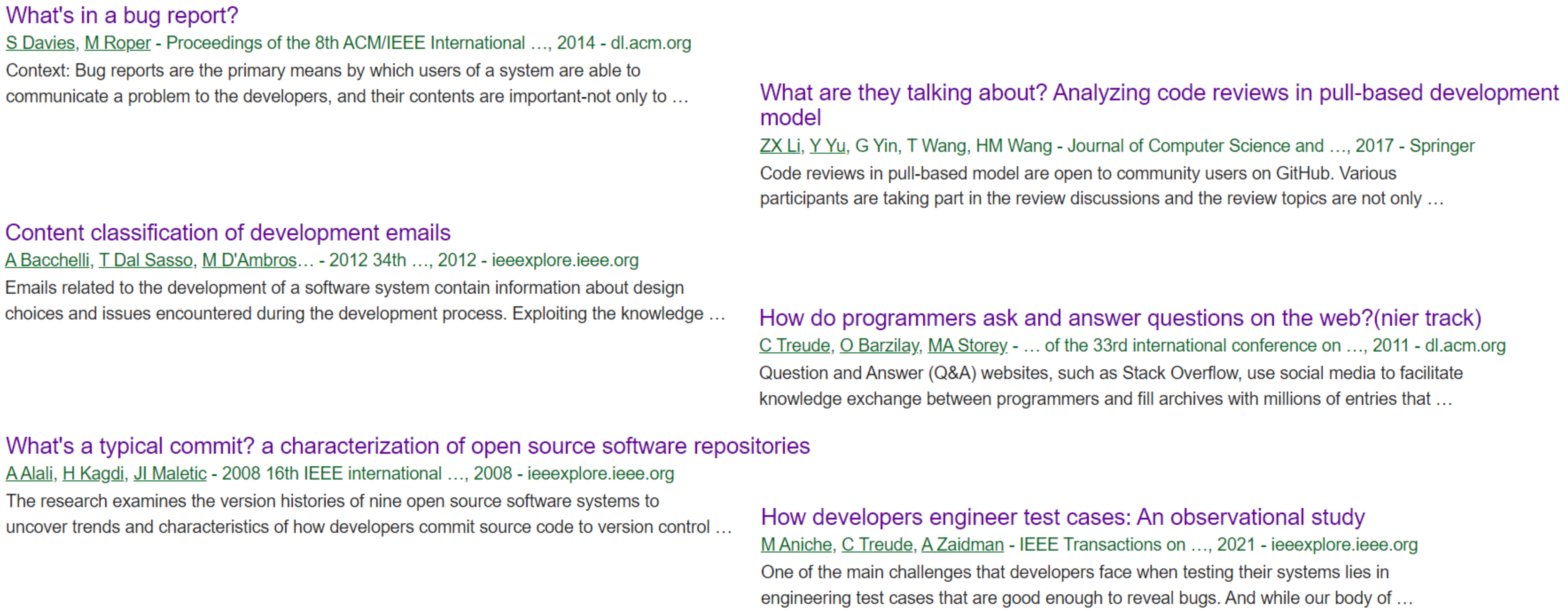
Analyzing interactions with LLMs



Analyzing interactions with ~~LLMs~~ artifacts



Analyzing interactions with ~~LLMs~~ artifacts



What's in a bug report?

S Davies, M Roper - Proceedings of the 8th ACM/IEEE International …, 2014 - dl.acm.org

Context: Bug reports are the primary means by which users of a system are able to communicate a problem to the developers, and their contents are important-not only to …

What are they talking about? Analyzing code reviews in pull-based development model

ZX Li, Y Yu, G Yin, T Wang, HM Wang - Journal of Computer Science and …, 2017 - Springer

Code reviews in pull-based model are open to community users on GitHub. Various participants are taking part in the review discussions and the review topics are not only …

Content classification of development emails

A Bacchelli, T Dal Sasso, M D'Ambros… - 2012 34th …, 2012 - ieeexplore.ieee.org

Emails related to the development of a software system contain information about design choices and issues encountered during the development process. Exploiting the knowledge …

How do programmers ask and answer questions on the web?(nier track)

C Treude, O Barzilay, MA Storey - … of the 33rd international conference on …, 2011 - dl.acm.org

Question and Answer (Q&A) websites, such as Stack Overflow, use social media to facilitate knowledge exchange between programmers and fill archives with millions of entries that …

What's a typical commit? a characterization of open source software repositories

A Alali, H Kagdi, JI Maletic - 2008 16th IEEE international …, 2008 - ieeexplore.ieee.org

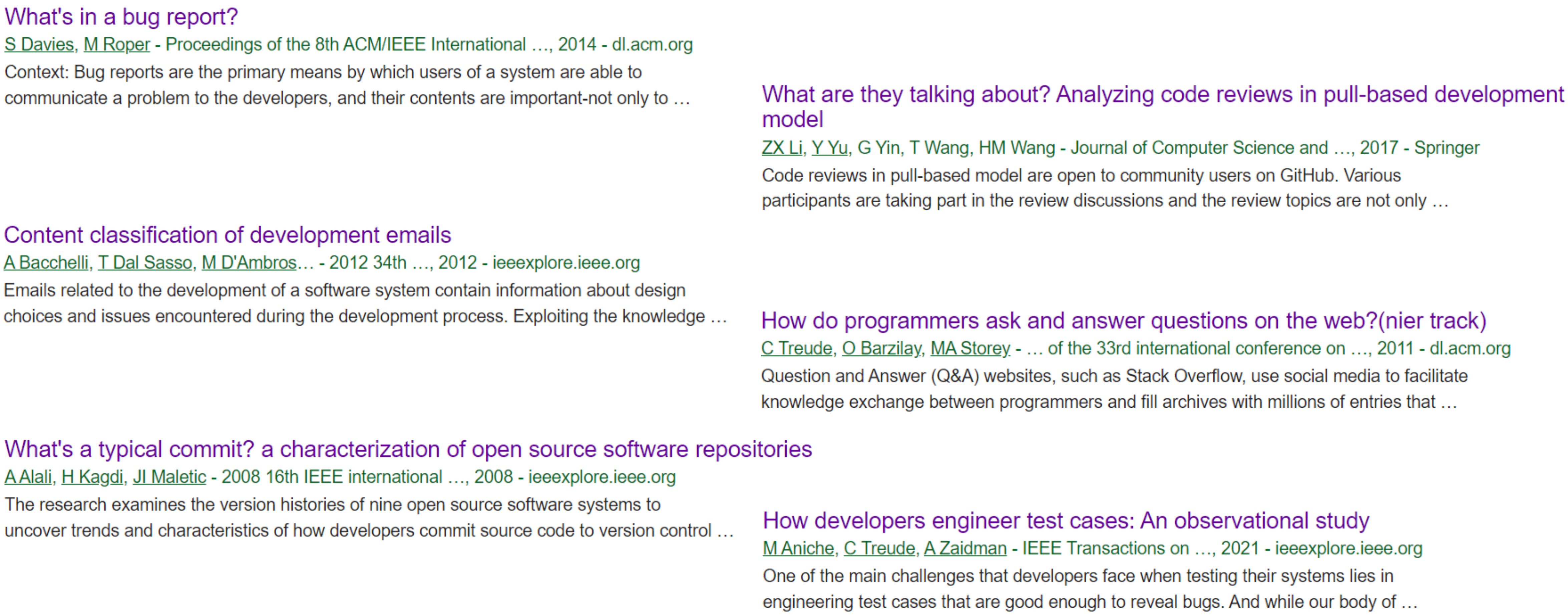
The research examines the version histories of nine open source software systems to uncover trends and characteristics of how developers commit source code to version control …

How developers engineer test cases: An observational study

M Aniche, C Treude, A Zaidman - IEEE Transactions on …, 2021 - ieeexplore.ieee.org

One of the main challenges that developers face when testing their systems lies in engineering test cases that are good enough to reveal bugs. And while our body of …

How do developers interact with LLMs?



What's in a bug report?
S Davies, M Roper - Proceedings of the 8th ACM/IEEE International …, 2014 - dl.acm.org
Context: Bug reports are the primary means by which users of a system are able to communicate a problem to the developers, and their contents are important-not only to …

What are they talking about? Analyzing code reviews in pull-based development model
ZX Li, Y Yu, G Yin, T Wang, HM Wang - Journal of Computer Science and …, 2017 - Springer
Code reviews in pull-based model are open to community users on GitHub. Various participants are taking part in the review discussions and the review topics are not only …

Content classification of development emails
A Bacchelli, T Dal Sasso, M D'Ambros… - 2012 34th …, 2012 - ieeexplore.ieee.org
Emails related to the development of a software system contain information about design choices and issues encountered during the development process. Exploiting the knowledge …

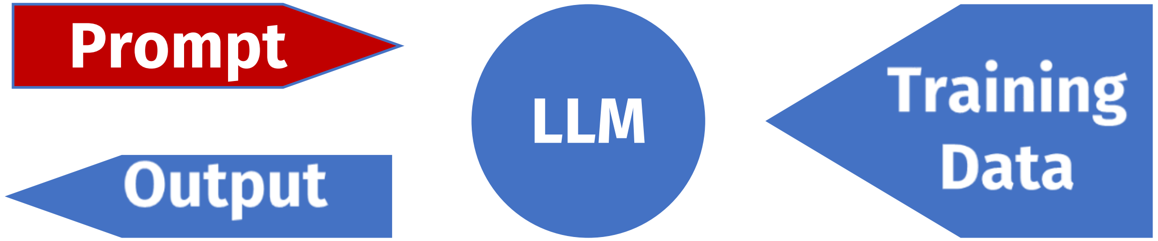
How do programmers ask and answer questions on the web?(nier track)
C Treude, O Barzilay, MA Storey - … of the 33rd international conference on …, 2011 - dl.acm.org
Question and Answer (Q&A) websites, such as Stack Overflow, use social media to facilitate knowledge exchange between programmers and fill archives with millions of entries that …

What's a typical commit? a characterization of open source software repositories
A Alali, H Kagdi, JI Maletic - 2008 16th IEEE international …, 2008 - ieeexplore.ieee.org
The research examines the version histories of nine open source software systems to uncover trends and characteristics of how developers commit source code to version control …

How developers engineer test cases: An observational study
M Aniche, C Treude, A Zaidman - IEEE Transactions on …, 2021 - ieeexplore.ieee.org
One of the main challenges that developers face when testing their systems lies in engineering test cases that are good enough to reveal bugs. And while our body of …

How do developers interact with LLMs?

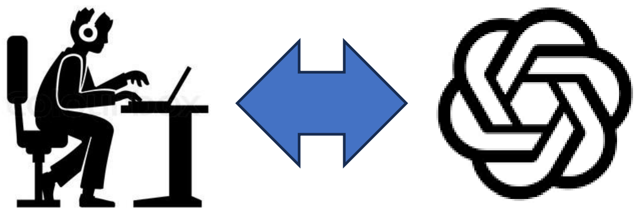
Mining Challenge: **DevGPT**

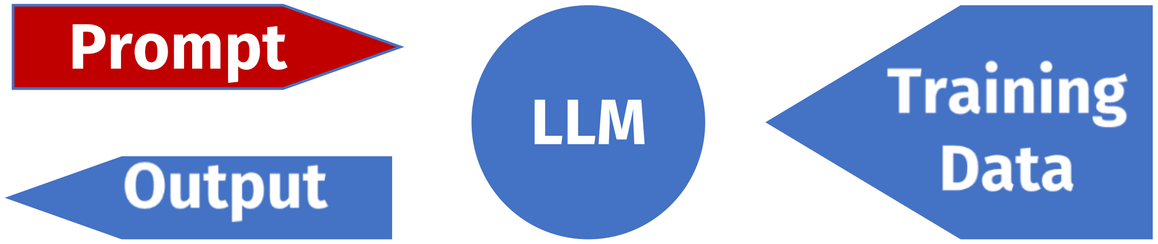


[Xiao et al., MSR 2024]

How do developers interact with LLMs?

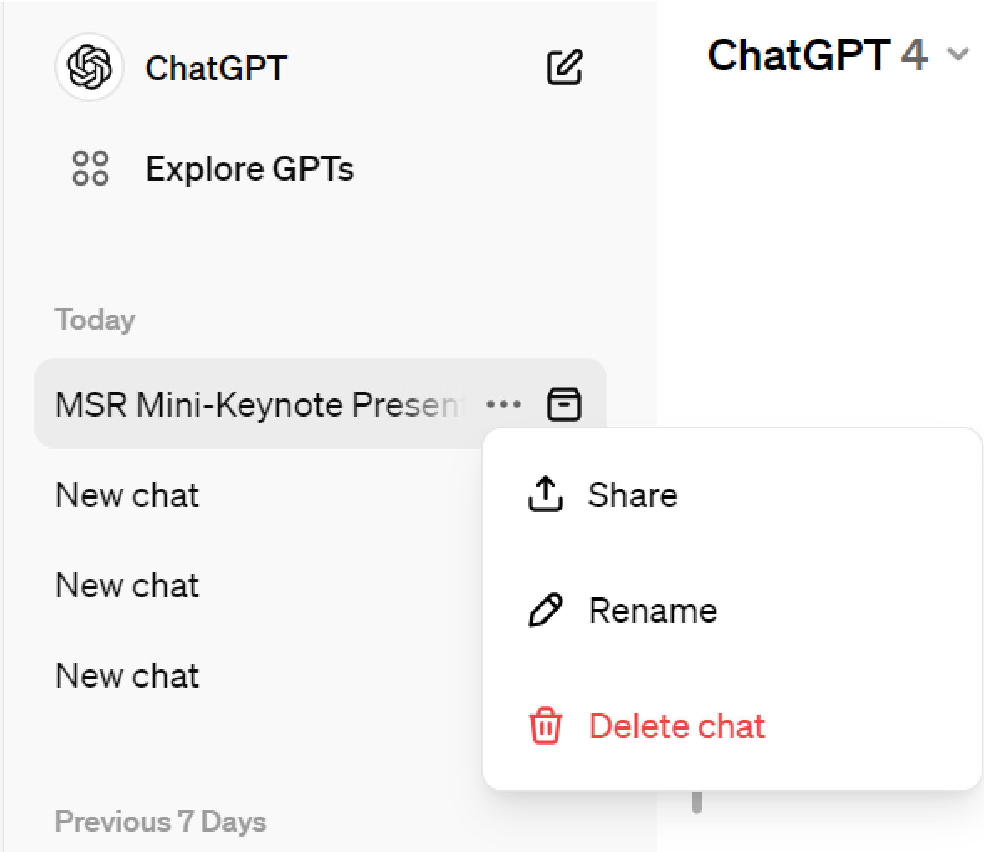
Mining Challenge: **DevGPT**

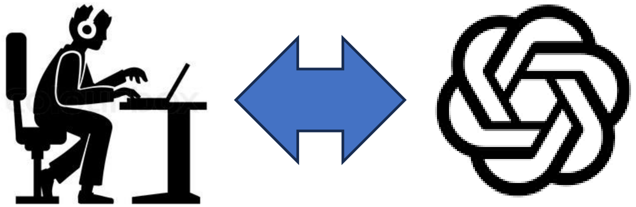




[Xiao et al., MSR 2024]

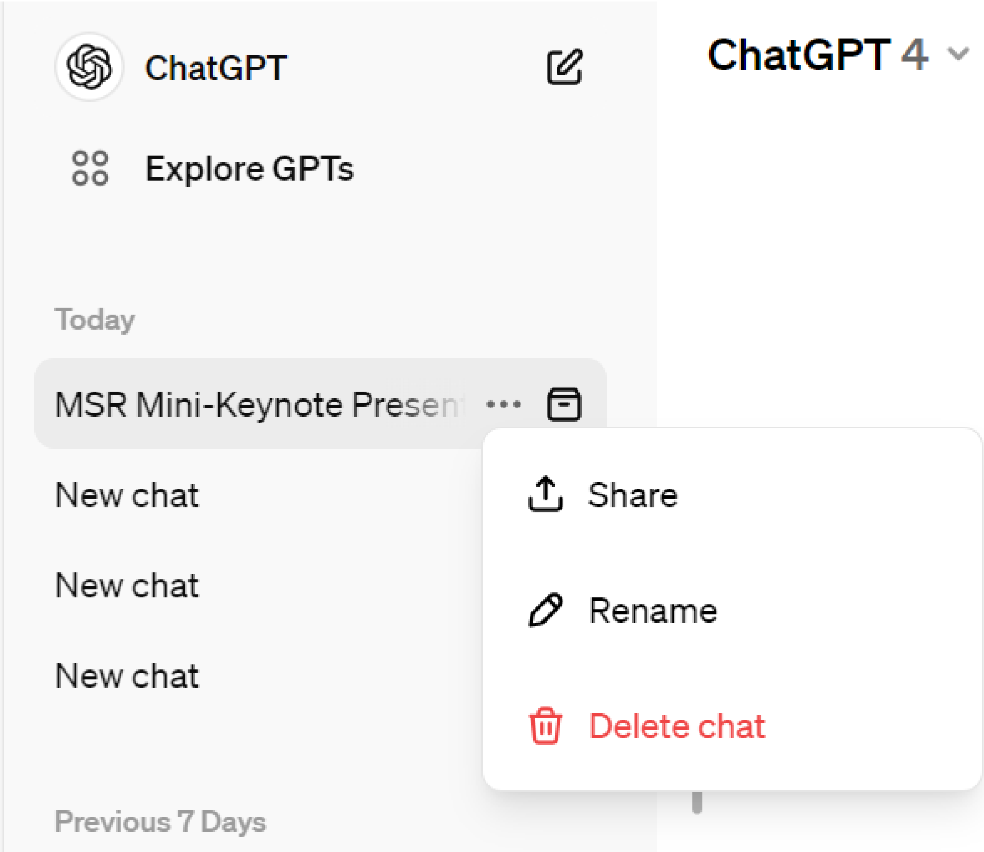
How do developers interact with LLMs?

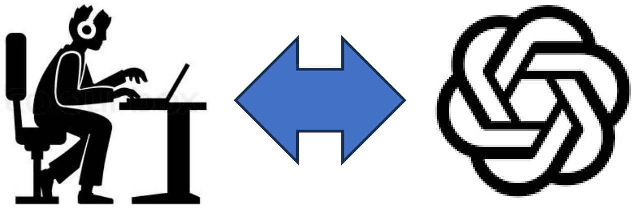




[Xiao et al., MSR 2024]

How do developers interact with LLMs?

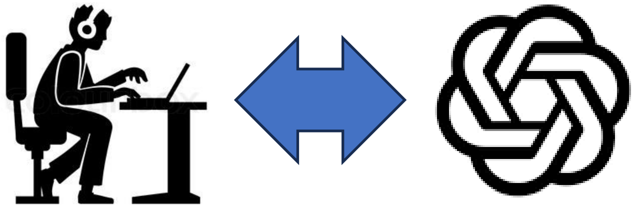




[Xiao et al., MSR 2024]

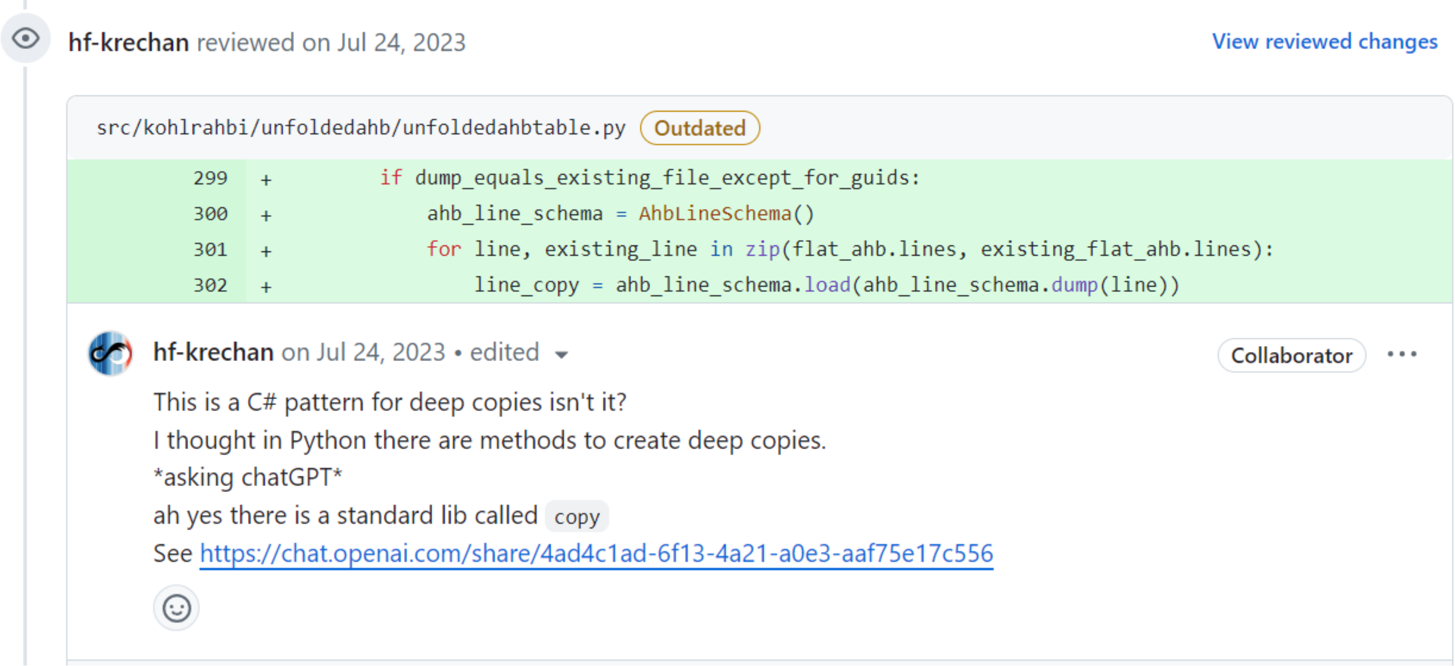
How do developers interact with LLMs?

	Links	Prompts	Code Snippets
Source Code	2,708	22,799	14,132
Commits	694	1,922	1,828
Issues	636	2,365	1,739
Pull Requests	301	1,160	975
Discussions	70	259	188
Hacker News	324	1,273	244
	4,733	**29,778**	**19,106**



[Xiao et al., MSR 2024]

How do developers interact with LLMs?



hf-krechan reviewed on Jul 24, 2023
View reviewed changes
Outdated

src/kohlrahbi/unfoldedahb/unfoldedahbtable.py

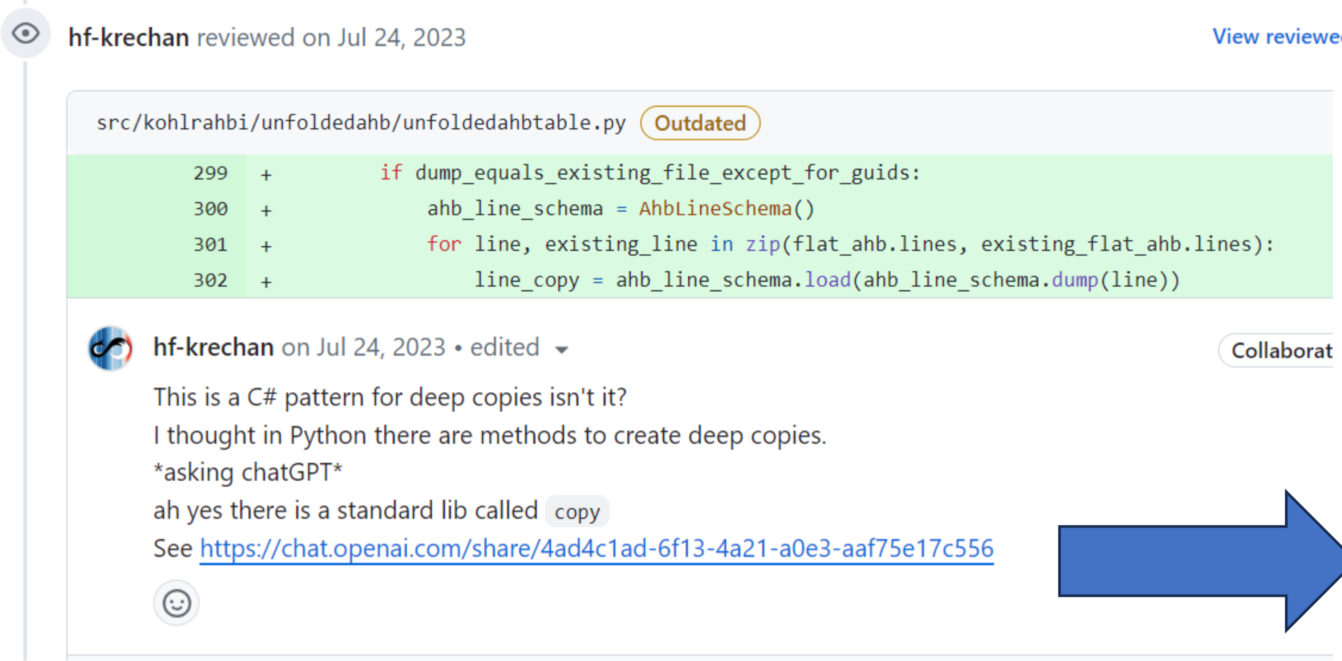
```
+        if dump_equals_existing_file_except_for_guids:
+            ahb_line_schema = AhbLineSchema()
+            for line, existing_line in zip(flat_ahb.lines, existing_flat_ahb.lines):
+                line_copy = ahb_line_schema.load(ahb_line_schema.dump(line))
```

hf-krechan on Jul 24, 2023 • edited
Collaborator

This is a C# pattern for deep copies isn't it?
I thought in Python there are methods to create deep copies.
asking chatGPT
ah yes there is a standard lib called `copy`
See https://chat.openai.com/share/4ad4c1ad-6f13-4a21-a0e3-aaf75e17c556

[Xiao et al., MSR 2024]

How do developers interact with LLMs?

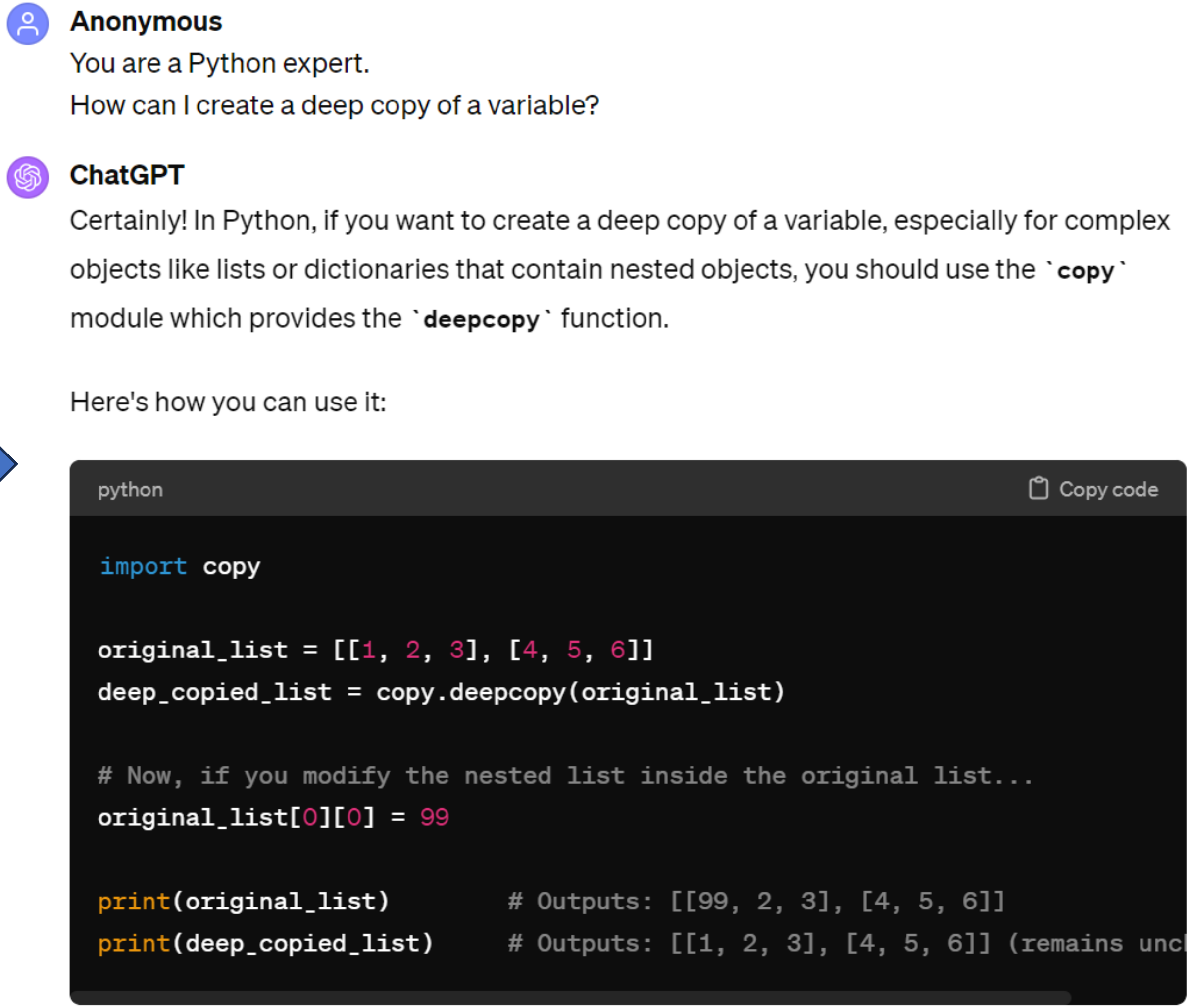


src/kohlrahbi/unfoldedahb/unfoldedahbtable.py

```
+        if dump_equals_existing_file_except_for_guids:
+            ahb_line_schema = AhbLineSchema()
+            for line, existing_line in zip(flat_ahb.lines, existing_flat_ahb.lines):
+                line_copy = ahb_line_schema.load(ahb_line_schema.dump(line))
```

hf-krechan on Jul 24, 2023 • edited

This is a C# pattern for deep copies isn't it?
I thought in Python there are methods to create deep copies.
asking chatGPT
ah yes there is a standard lib called `copy`
See https://chat.openai.com/share/4ad4c1ad-6f13-4a21-a0e3-aaf75e17c556



Anonymous

You are a Python expert.
How can I create a deep copy of a variable?

ChatGPT

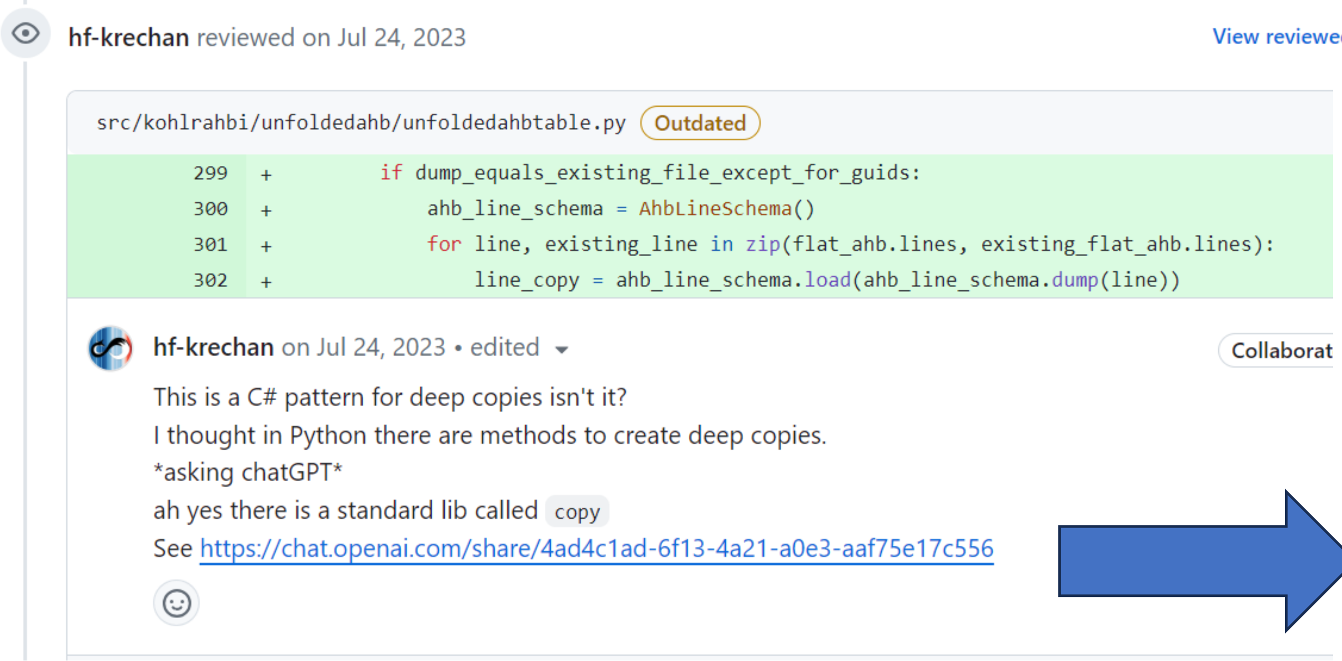
Certainly! In Python, if you want to create a deep copy of a variable, especially for complex objects like lists or dictionaries that contain nested objects, you should use the `copy` module which provides the `deepcopy` function.

Here's how you can use it:

```python
import copy

original_list = [[1, 2, 3], [4, 5, 6]]
deep_copied_list = copy.deepcopy(original_list)

# Now, if you modify the nested list inside the original list...
original_list[0][0] = 99

print(original_list)       # Outputs: [[99, 2, 3], [4, 5, 6]]
print(deep_copied_list)    # Outputs: [[1, 2, 3], [4, 5, 6]] (remains unc
```

[Xiao et al., MSR 2024]

How do developers interact with LLMs?



hf-krechan reviewed on Jul 24, 2023

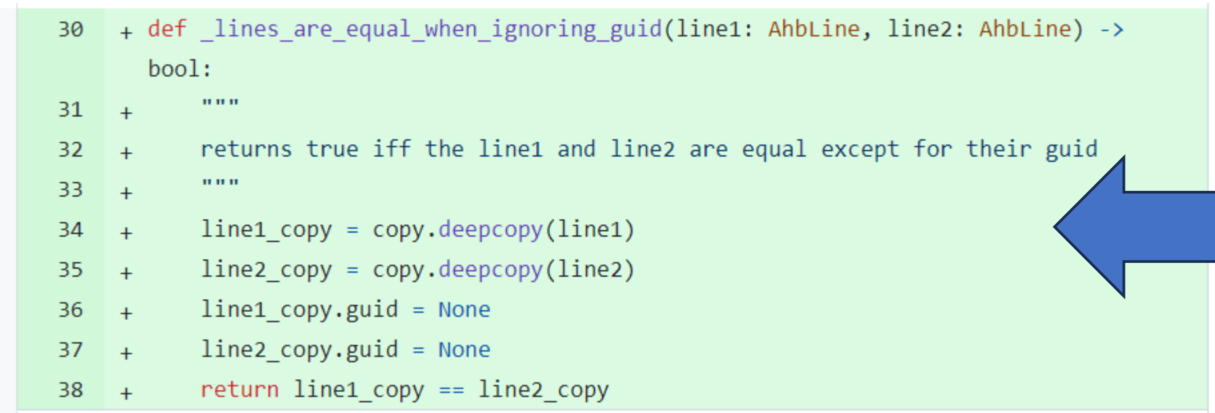
View reviewed changes

src/kohlrahbi/unfoldedahb/unfoldedahbtable.py

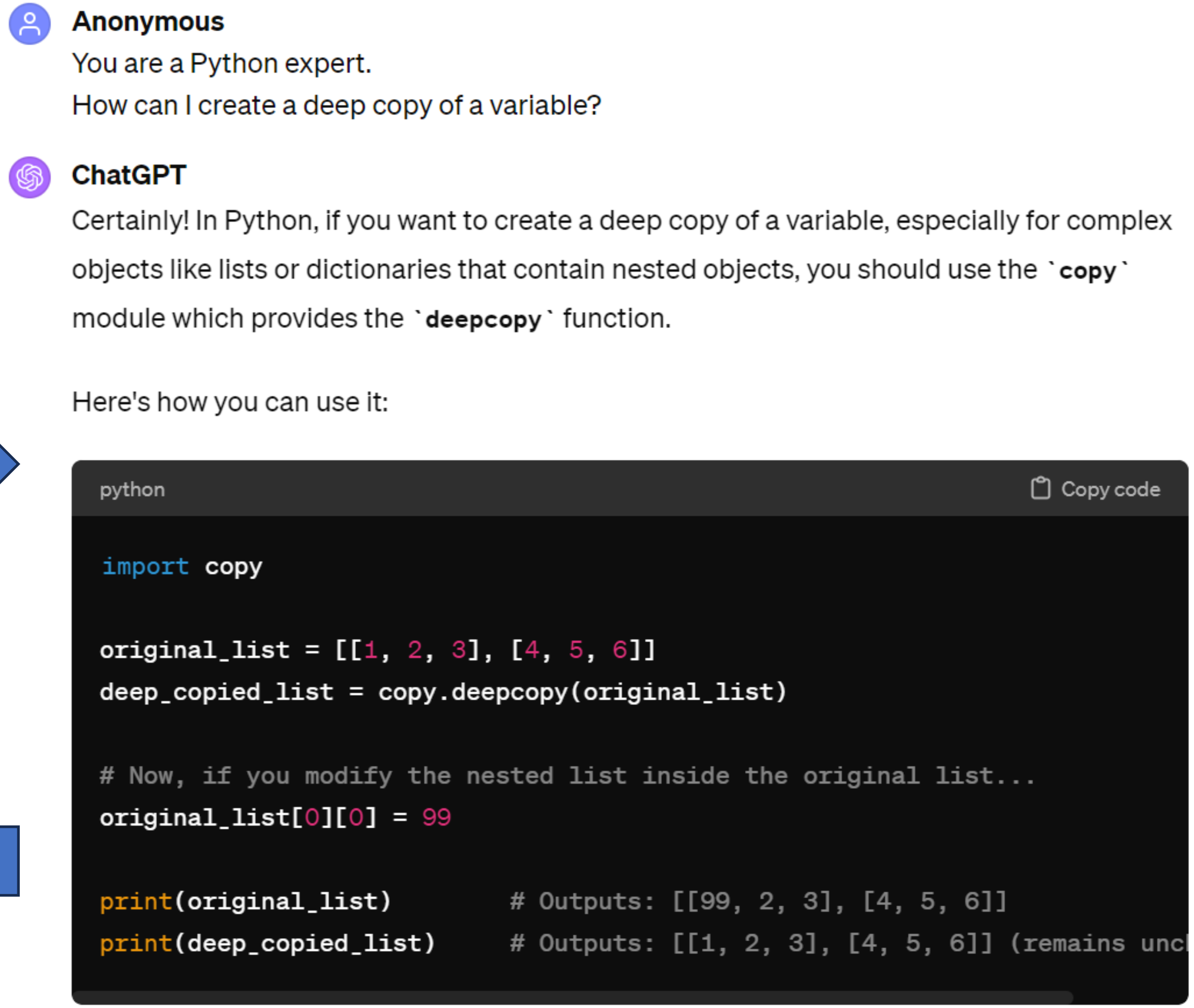
```
+        if dump_equals_existing_file_except_for_guids:
+            ahb_line_schema = AhbLineSchema()
+            for line, existing_line in zip(flat_ahb.lines, existing_flat_ahb.lines):
+                line_copy = ahb_line_schema.load(ahb_line_schema.dump(line))
```

hf-krechan on Jul 24, 2023 • edited

This is a C# pattern for deep copies isn't it?
I thought in Python there are methods to create deep copies.
asking chatGPT
ah yes there is a standard lib called `copy`
See https://chat.openai.com/share/4ad4c1ad-6f13-4a21-a0e3-aaf75e17c556



```
+ def _lines_are_equal_when_ignoring_guid(line1: AhbLine, line2: AhbLine) ->
  bool:
+     """
+     returns true iff the line1 and line2 are equal except for their guid
+     """
+     line1_copy = copy.deepcopy(line1)
+     line2_copy = copy.deepcopy(line2)
+     line1_copy.guid = None
+     line2_copy.guid = None
+     return line1_copy == line2_copy
```



Anonymous

You are a Python expert.
How can I create a deep copy of a variable?

ChatGPT

Certainly! In Python, if you want to create a deep copy of a variable, especially for complex objects like lists or dictionaries that contain nested objects, you should use the `copy` module which provides the `deepcopy` function.

Here's how you can use it:

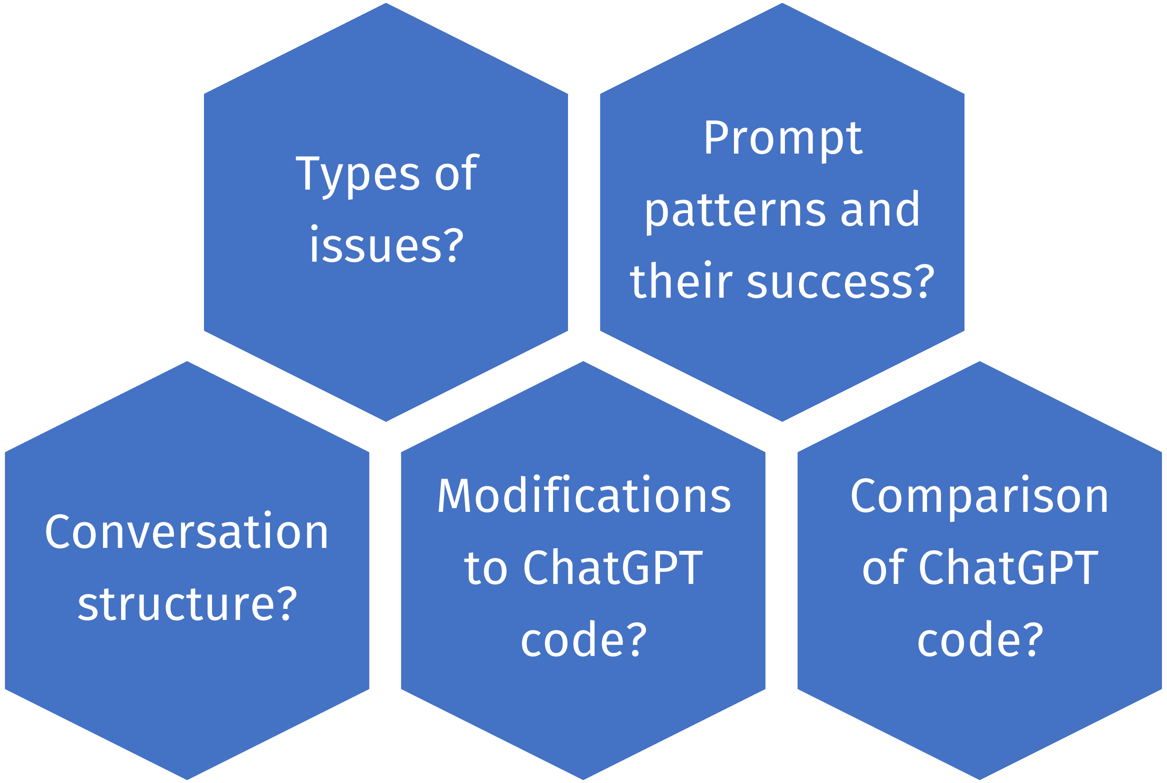
```python
import copy

original_list = [[1, 2, 3], [4, 5, 6]]
deep_copied_list = copy.deepcopy(original_list)

# Now, if you modify the nested list inside the original list...
original_list[0][0] = 99

print(original_list)        # Outputs: [[99, 2, 3], [4, 5, 6]]
print(deep_copied_list)     # Outputs: [[1, 2, 3], [4, 5, 6]] (remains unc
```

How do developers interact with LLMs?

Types of issues?

Prompt patterns and their success?

How do developers interact with LLMs?



Types of issues?

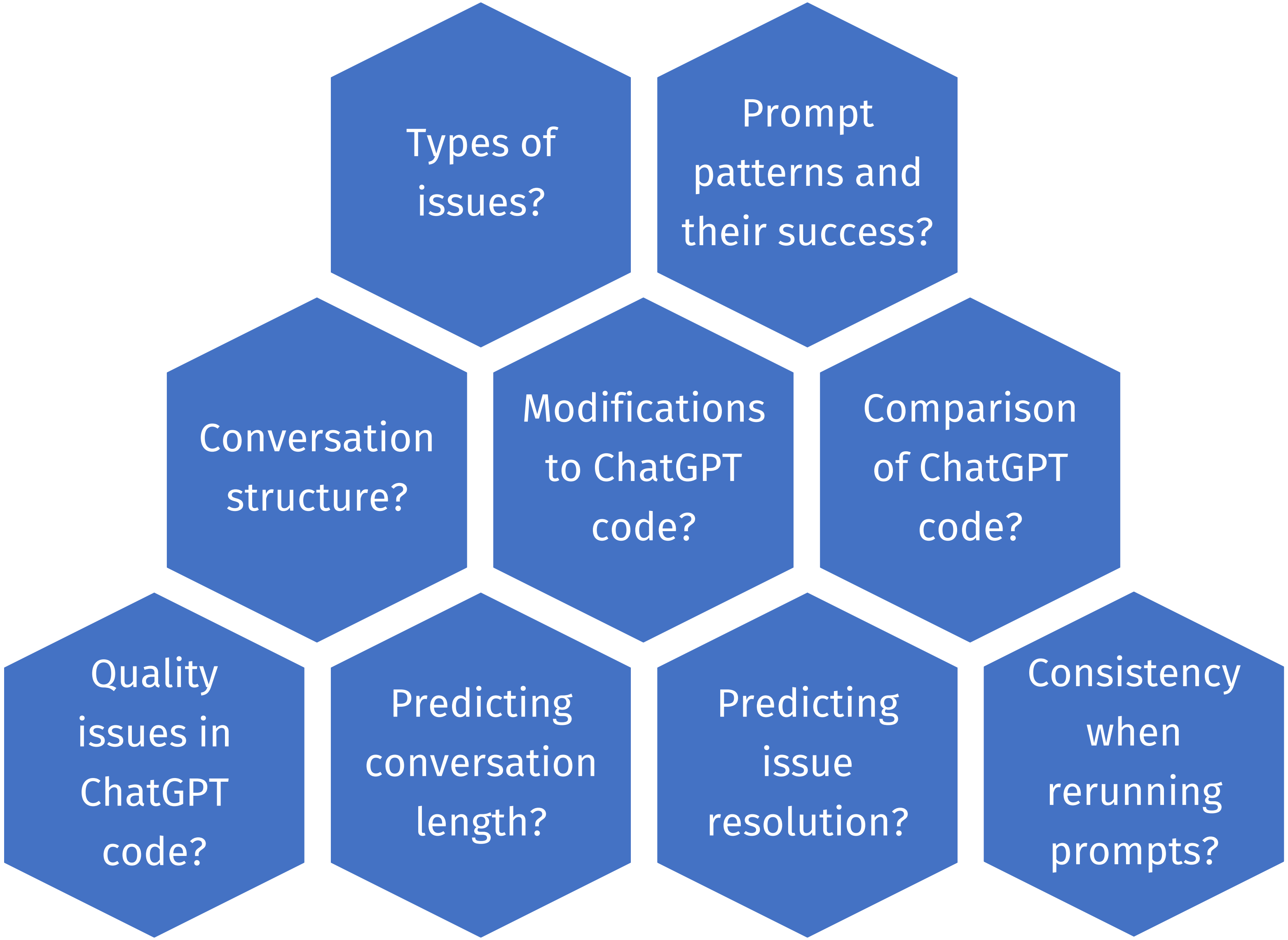
Prompt patterns and their success?

Conversation structure?

Modifications to ChatGPT code?

Comparison of ChatGPT code?

How do developers interact with LLMs?



Types of issues?

Prompt patterns and their success?

Conversation structure?

Modifications to ChatGPT code?

Comparison of ChatGPT code?

Quality issues in ChatGPT code?

Predicting conversation length?

Predicting issue resolution?

Consistency when rerunning prompts?

How do developers interact with LLMs?

Types of issues?

Prompt patterns an their su

Conversation structure?

Mod PT de?

Comparison of ChatGPT code?

Quality issue ChatGP code?

Predicting conversation length?

Predicting issue resolution?

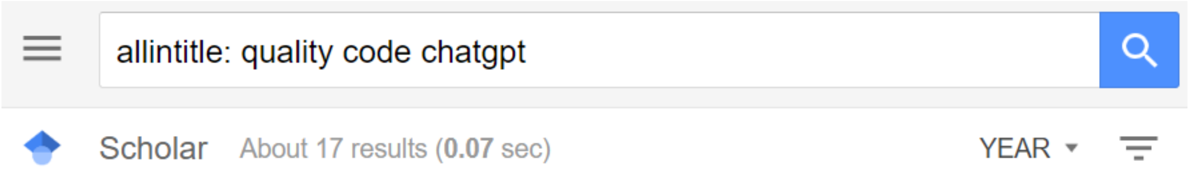
Consistency when rerunning prompts?

LLM interactions are data, too

Analyzing LLM output

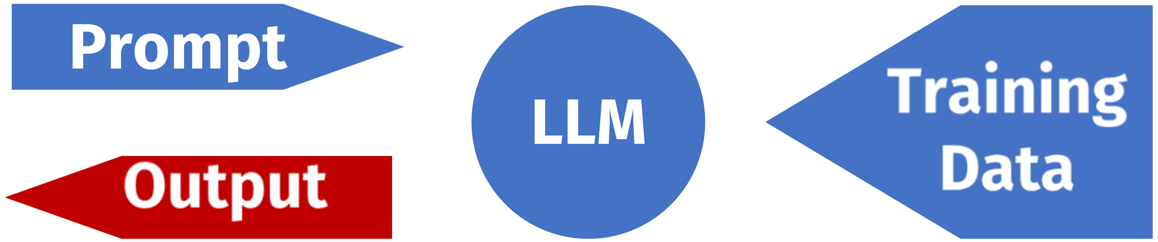
Prompt
Output
LLM
Training Data

Analyzing LLM output



allintitle: quality code chatgpt

Scholar About 17 results (**0.07** sec)



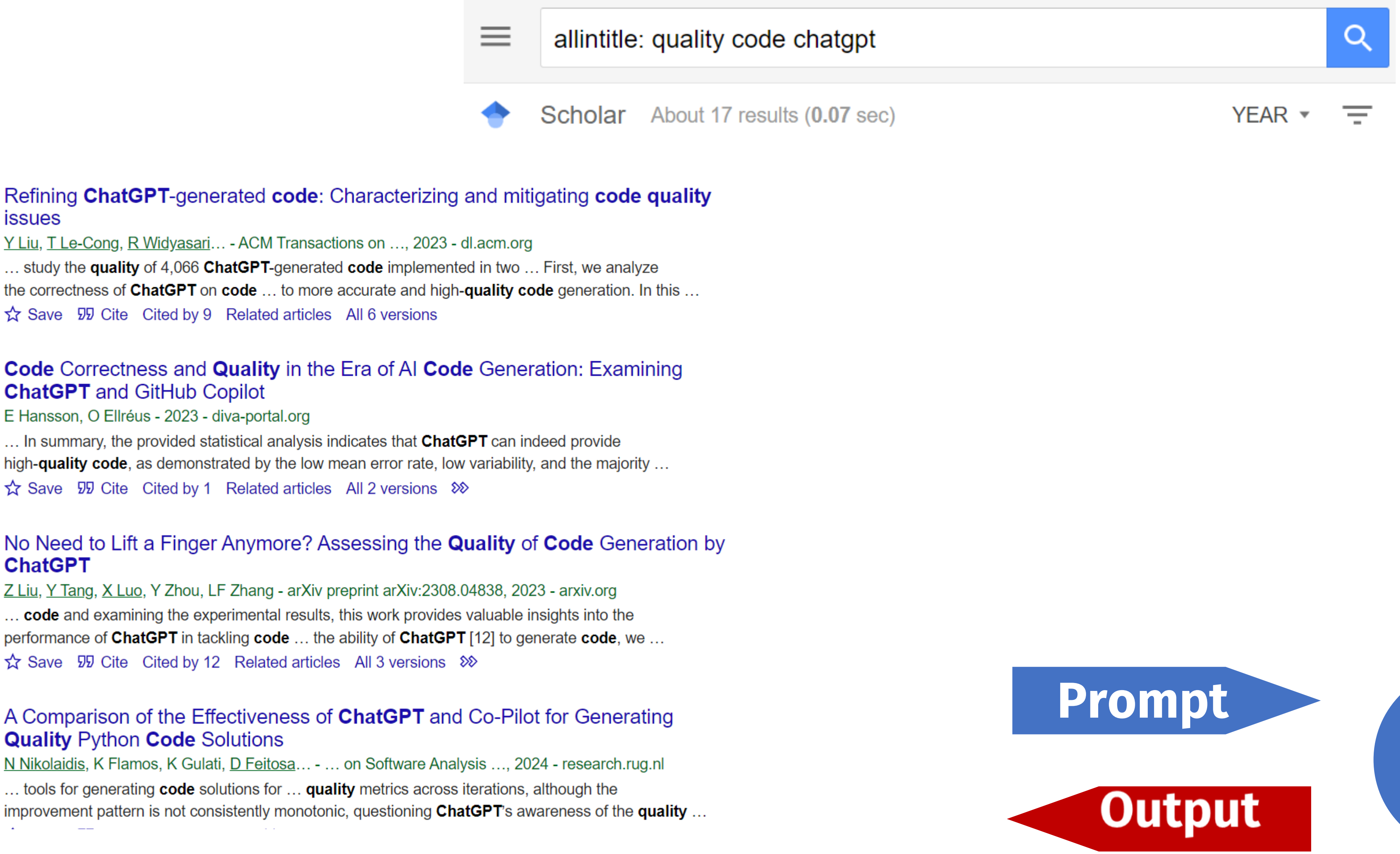
Prompt

Output

LLM

Training Data

Analyzing LLM output



allintitle: quality code chatgpt

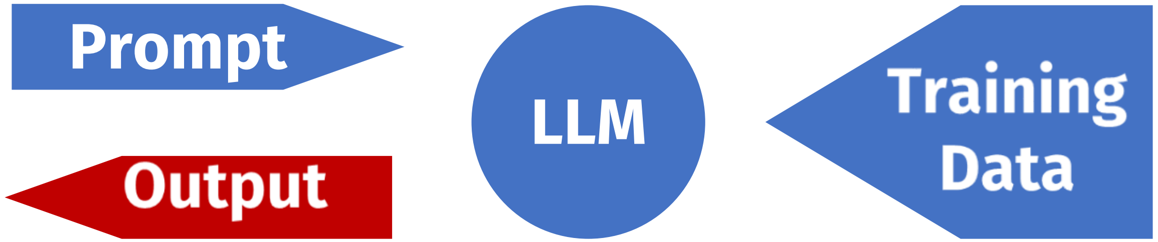
Scholar About 17 results (**0.07** sec) YEAR

Refining **ChatGPT**-generated **code**: Characterizing and mitigating **code quality** issues
Y Liu, T Le-Cong, R Widyasari… - ACM Transactions on …, 2023 - dl.acm.org
… study the **quality** of 4,066 **ChatGPT**-generated **code** implemented in two … First, we analyze the correctness of **ChatGPT** on **code** … to more accurate and high-**quality code** generation. In this …
☆ Save 99 Cite Cited by 9 Related articles All 6 versions

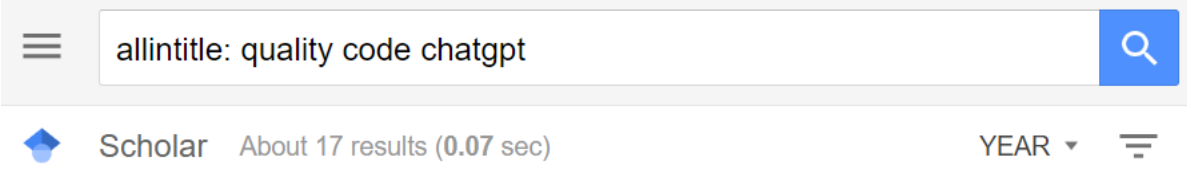
Code Correctness and **Quality** in the Era of AI **Code** Generation: Examining **ChatGPT** and GitHub Copilot
E Hansson, O Ellréus - 2023 - diva-portal.org
… In summary, the provided statistical analysis indicates that **ChatGPT** can indeed provide high-**quality code**, as demonstrated by the low mean error rate, low variability, and the majority …
☆ Save 99 Cite Cited by 1 Related articles All 2 versions »

No Need to Lift a Finger Anymore? Assessing the **Quality** of **Code** Generation by **ChatGPT**
Z Liu, Y Tang, X Luo, Y Zhou, LF Zhang - arXiv preprint arXiv:2308.04838, 2023 - arxiv.org
… **code** and examining the experimental results, this work provides valuable insights into the performance of **ChatGPT** in tackling **code** … the ability of **ChatGPT** [12] to generate **code**, we …
☆ Save 99 Cite Cited by 12 Related articles All 3 versions »

A Comparison of the Effectiveness of **ChatGPT** and Co-Pilot for Generating **Quality** Python **Code** Solutions
N Nikolaidis, K Flamos, K Gulati, D Feitosa… - … on Software Analysis …, 2024 - research.rug.nl
… tools for generating **code** solutions for … **quality** metrics across iterations, although the improvement pattern is not consistently monotonic, questioning **ChatGPT**'s awareness of the **quality** …

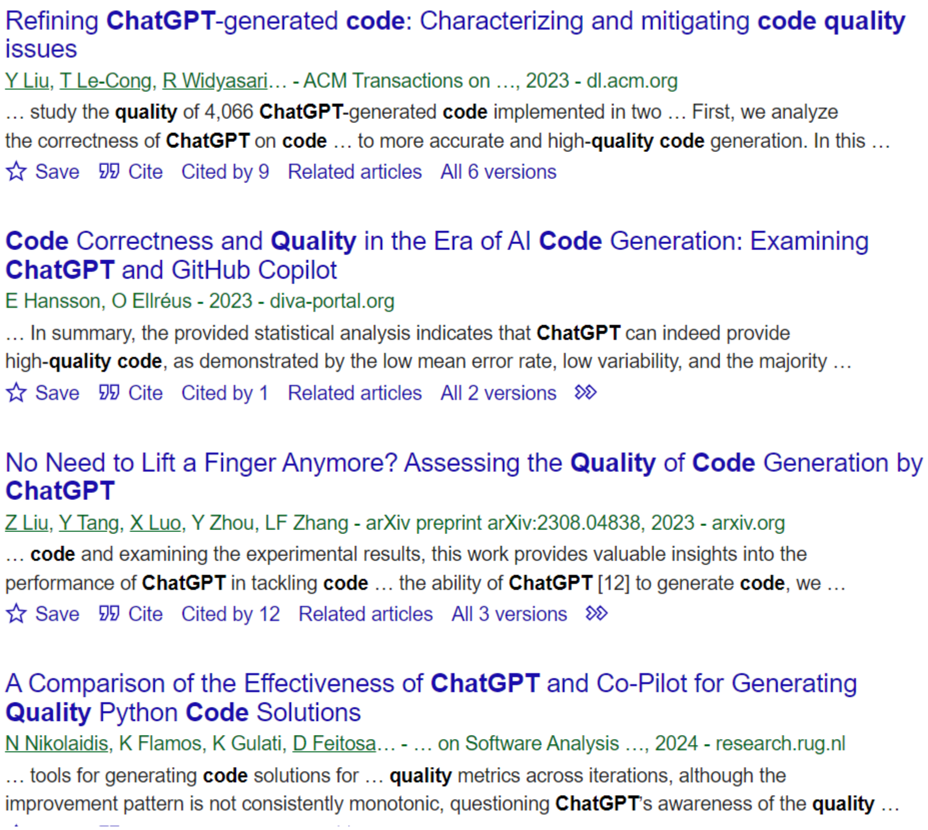


Analyzing LLM output



allintitle: quality code chatgpt

Scholar About 17 results (**0.07** sec)

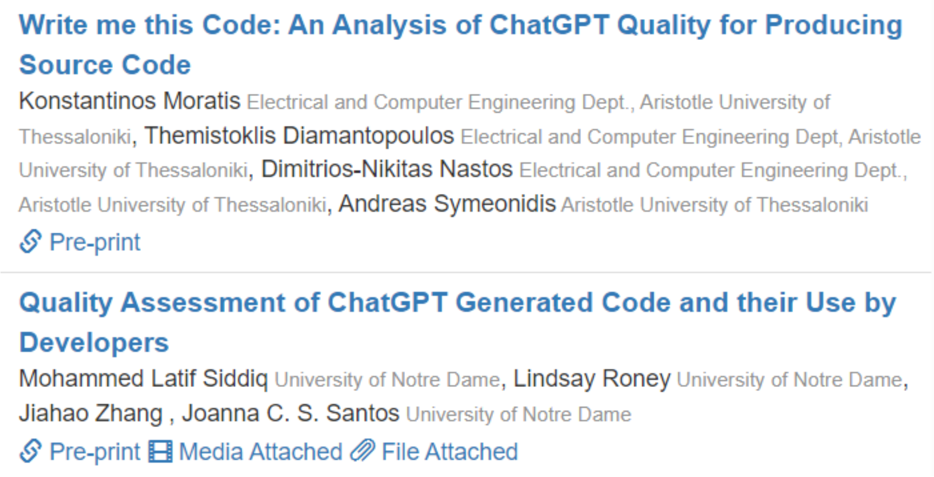


Refining **ChatGPT**-generated **code**: Characterizing and mitigating **code quality** issues
Y Liu, T Le-Cong, R Widyasari… - ACM Transactions on …, 2023 - dl.acm.org
… study the **quality** of 4,066 **ChatGPT**-generated **code** implemented in two … First, we analyze the correctness of **ChatGPT** on **code** … to more accurate and high-**quality code** generation. In this …

Code Correctness and **Quality** in the Era of AI **Code** Generation: Examining **ChatGPT** and GitHub Copilot
E Hansson, O Ellréus - 2023 - diva-portal.org
… In summary, the provided statistical analysis indicates that **ChatGPT** can indeed provide high-**quality code**, as demonstrated by the low mean error rate, low variability, and the majority …

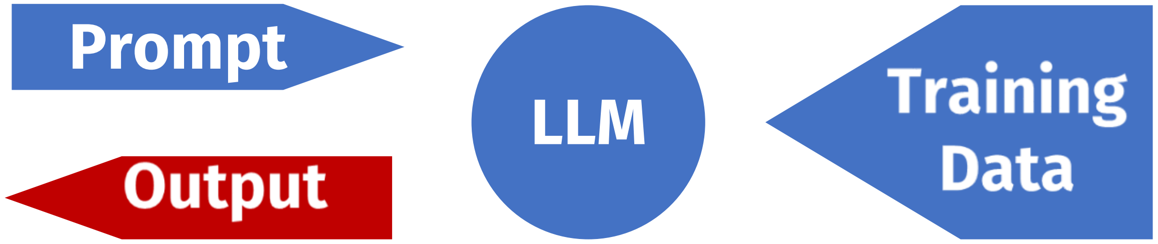
No Need to Lift a Finger Anymore? Assessing the **Quality** of **Code** Generation by **ChatGPT**
Z Liu, Y Tang, X Luo, Y Zhou, LF Zhang - arXiv preprint arXiv:2308.04838, 2023 - arxiv.org
… **code** and examining the experimental results, this work provides valuable insights into the performance of **ChatGPT** in tackling **code** … the ability of **ChatGPT** [12] to generate **code**, we …

A Comparison of the Effectiveness of **ChatGPT** and Co-Pilot for Generating **Quality** Python **Code** Solutions
N Nikolaidis, K Flamos, K Gulati, D Feitosa… - … on Software Analysis …, 2024 - research.rug.nl
… tools for generating **code** solutions for … **quality** metrics across iterations, although the improvement pattern is not consistently monotonic, questioning **ChatGPT**'s awareness of the **quality** …



Write me this Code: An Analysis of ChatGPT Quality for Producing Source Code
Konstantinos Moratis Electrical and Computer Engineering Dept., Aristotle University of Thessaloniki, Themistoklis Diamantopoulos Electrical and Computer Engineering Dept, Aristotle University of Thessaloniki, Dimitrios-Nikitas Nastos Electrical and Computer Engineering Dept., Aristotle University of Thessaloniki, Andreas Symeonidis Aristotle University of Thessaloniki

Quality Assessment of ChatGPT Generated Code and their Use by Developers
Mohammed Latif Siddiq University of Notre Dame, Lindsay Roney University of Notre Dame, Jiahao Zhang , Joanna C. S. Santos University of Notre Dame



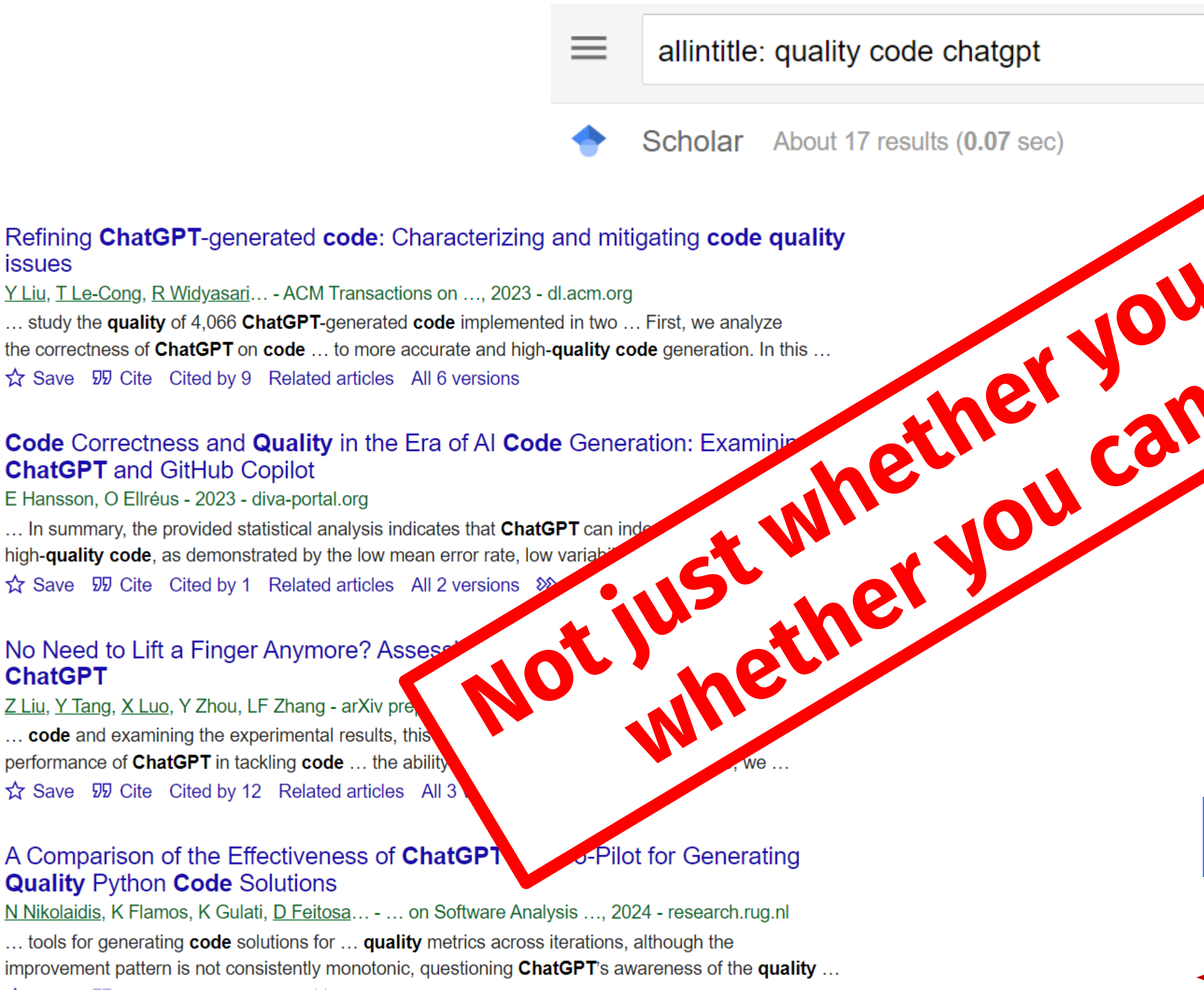
Prompt

Output

LLM

Training Data

Analyzing LLM output



allintitle: quality code chatgpt

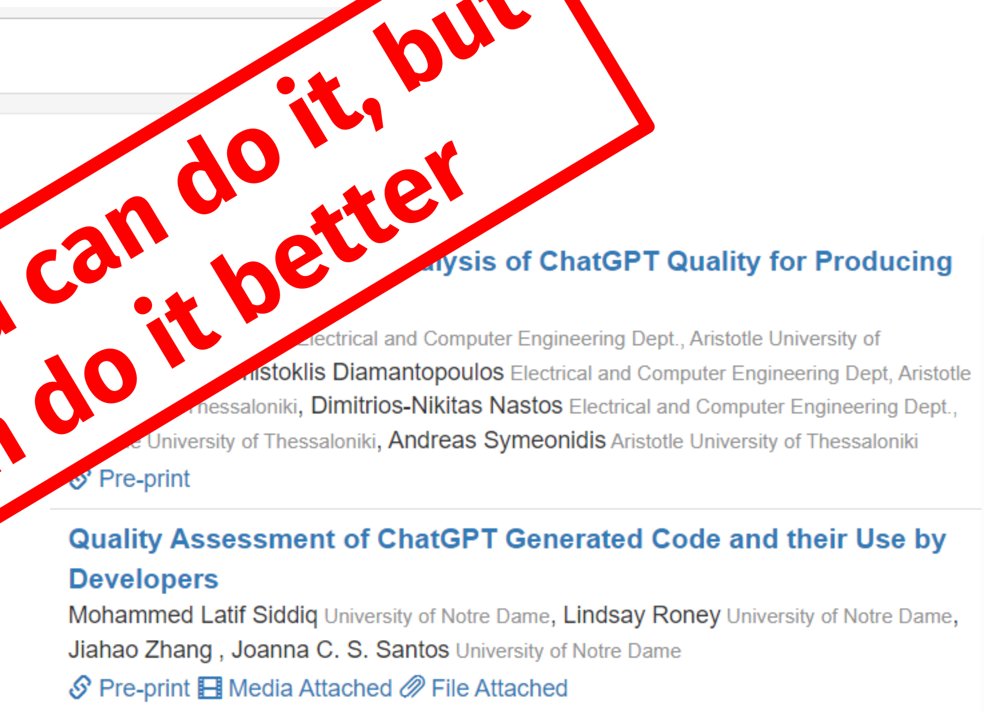
Scholar About 17 results (**0.07** sec)

Refining **ChatGPT**-generated **code**: Characterizing and mitigating **code quality** issues
Y Liu, T Le-Cong, R Widyasari… - ACM Transactions on …, 2023 - dl.acm.org
… study the **quality** of 4,066 **ChatGPT**-generated **code** implemented in two … First, we analyze the correctness of **ChatGPT** on **code** … to more accurate and high-**quality code** generation. In this …
☆ Save 99 Cite Cited by 9 Related articles All 6 versions

Code Correctness and **Quality** in the Era of AI **Code** Generation: Examinin
ChatGPT and GitHub Copilot
E Hansson, O Ellréus - 2023 - diva-portal.org
… In summary, the provided statistical analysis indicates that **ChatGPT** can ind
high-**quality code**, as demonstrated by the low mean error rate, low variab
☆ Save 99 Cite Cited by 1 Related articles All 2 versions

No Need to Lift a Finger Anymore? Assess
ChatGPT
Z Liu, Y Tang, X Luo, Y Zhou, LF Zhang - arXiv pre
… **code** and examining the experimental results, this
performance of **ChatGPT** in tackling **code** … the ability , we …
☆ Save 99 Cite Cited by 12 Related articles All 3

A Comparison of the Effectiveness of **ChatGP** -Pilot for Generating
Quality Python **Code** Solutions
N Nikolaidis, K Flamos, K Gulati, D Feitosa… - … on Software Analysis …, 2024 - research.rug.nl
… tools for generating **code** solutions for … **quality** metrics across iterations, although the improvement pattern is not consistently monotonic, questioning **ChatGPT**'s awareness of the **quality** …

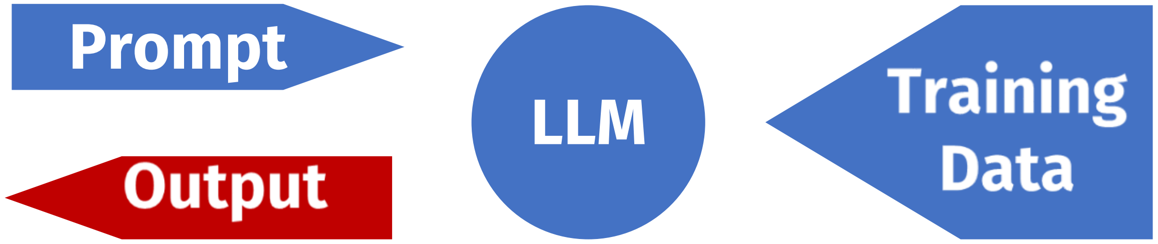


lysis of ChatGPT Quality for Producing

lectrical and Computer Engineering Dept., Aristotle University of
istoklis Diamantopoulos Electrical and Computer Engineering Dept, Aristotle
hessaloniki, Dimitrios-Nikitas Nastos Electrical and Computer Engineering Dept.,
University of Thessaloniki, Andreas Symeonidis Aristotle University of Thessaloniki
Pre-print

Quality Assessment of ChatGPT Generated Code and their Use by Developers
Mohammed Latif Siddiq University of Notre Dame, Lindsay Roney University of Notre Dame, Jiahao Zhang , Joanna C. S. Santos University of Notre Dame
Pre-print Media Attached File Attached

Not just whether you can do it, but whether you can do it better



Analyzing LLM output

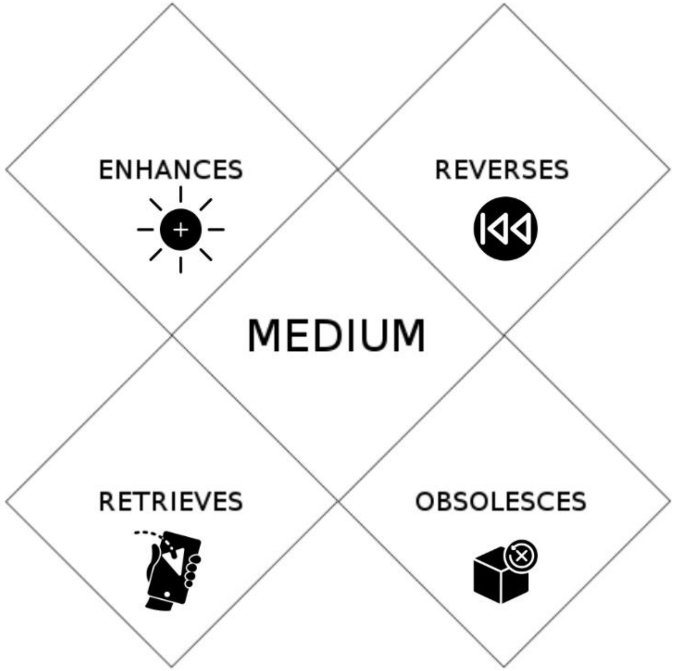
Prompt

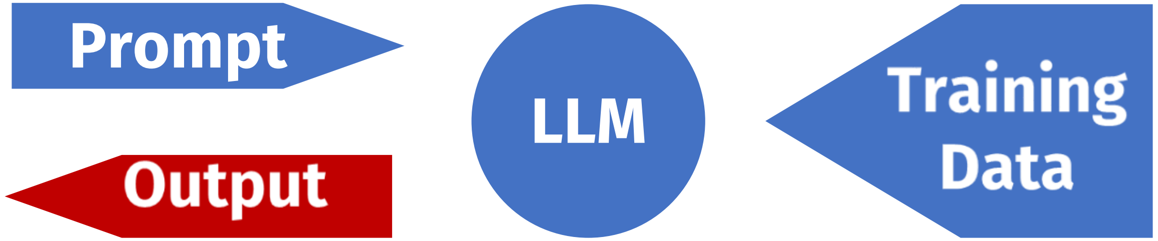
Output

LLM

Training Data

Analyzing LLM output




Prompt

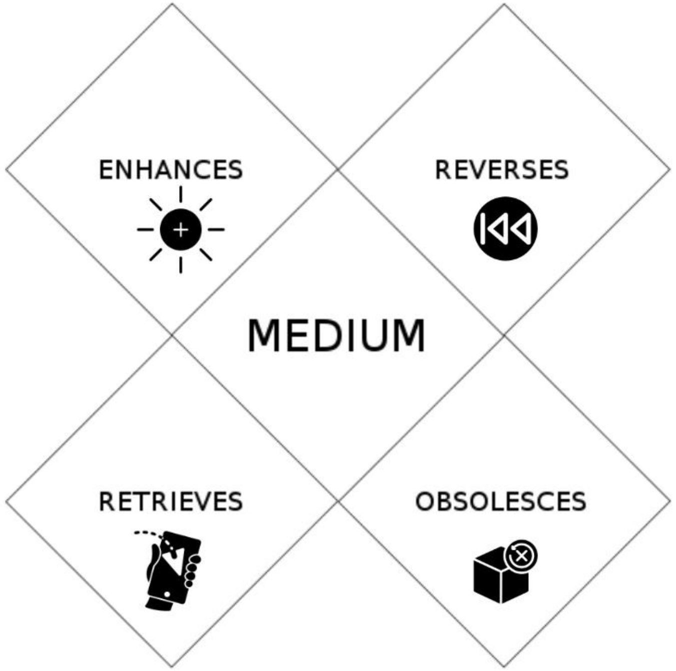
Output

LLM

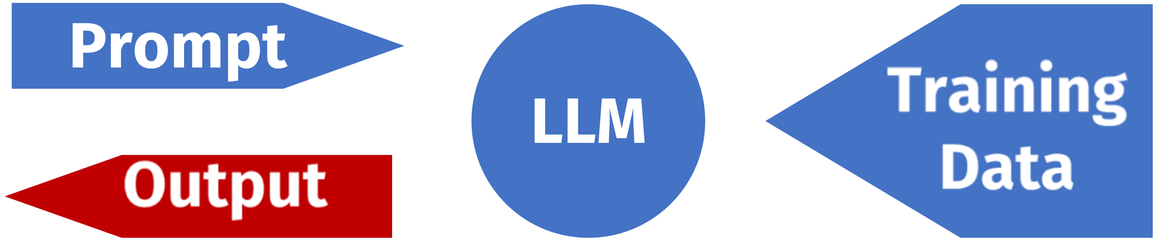
Training Data

[Storey et al., 2024]

Analyzing LLM output

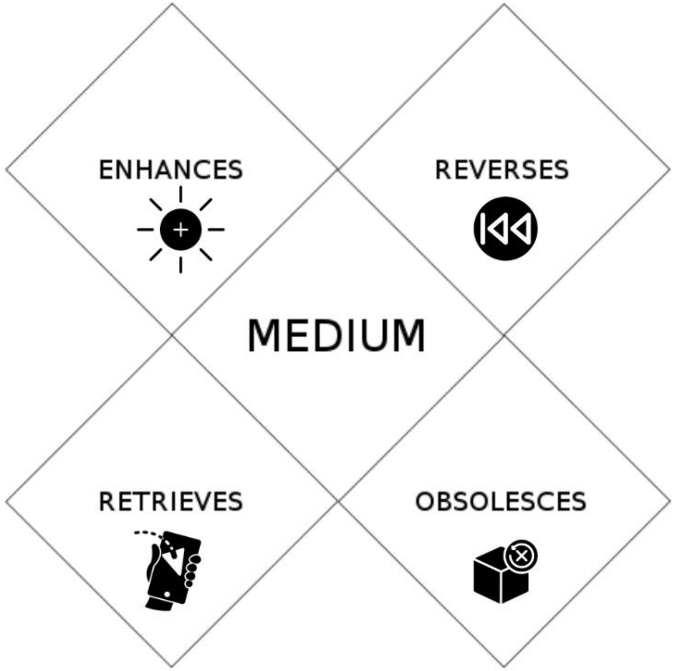


What does the technology
... enhance or amplify?
... make obsolete?
... retrieve that had been obsolesced earlier?
... reverse or flip into when pushed to extremes?



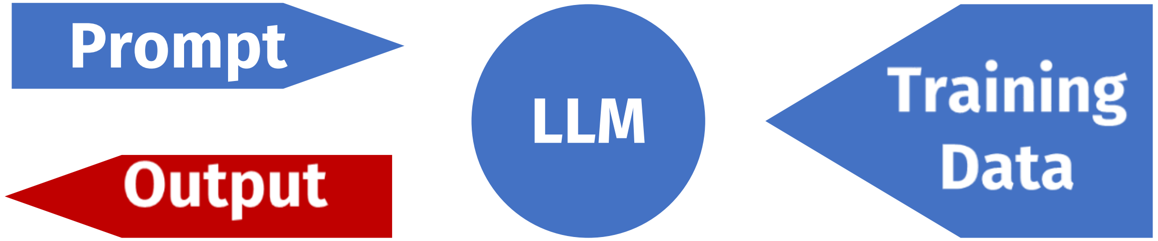
[Storey et al., 2024]

Analyzing LLM output



What do we lose from past solutions now rendered obsolete by this technology?

What does the technology
- … enhance or amplify?
- … **make obsolete?**
- … retrieve that had been obsolesced earlier?
- … reverse or flip into when pushed to extremes?


Prompt

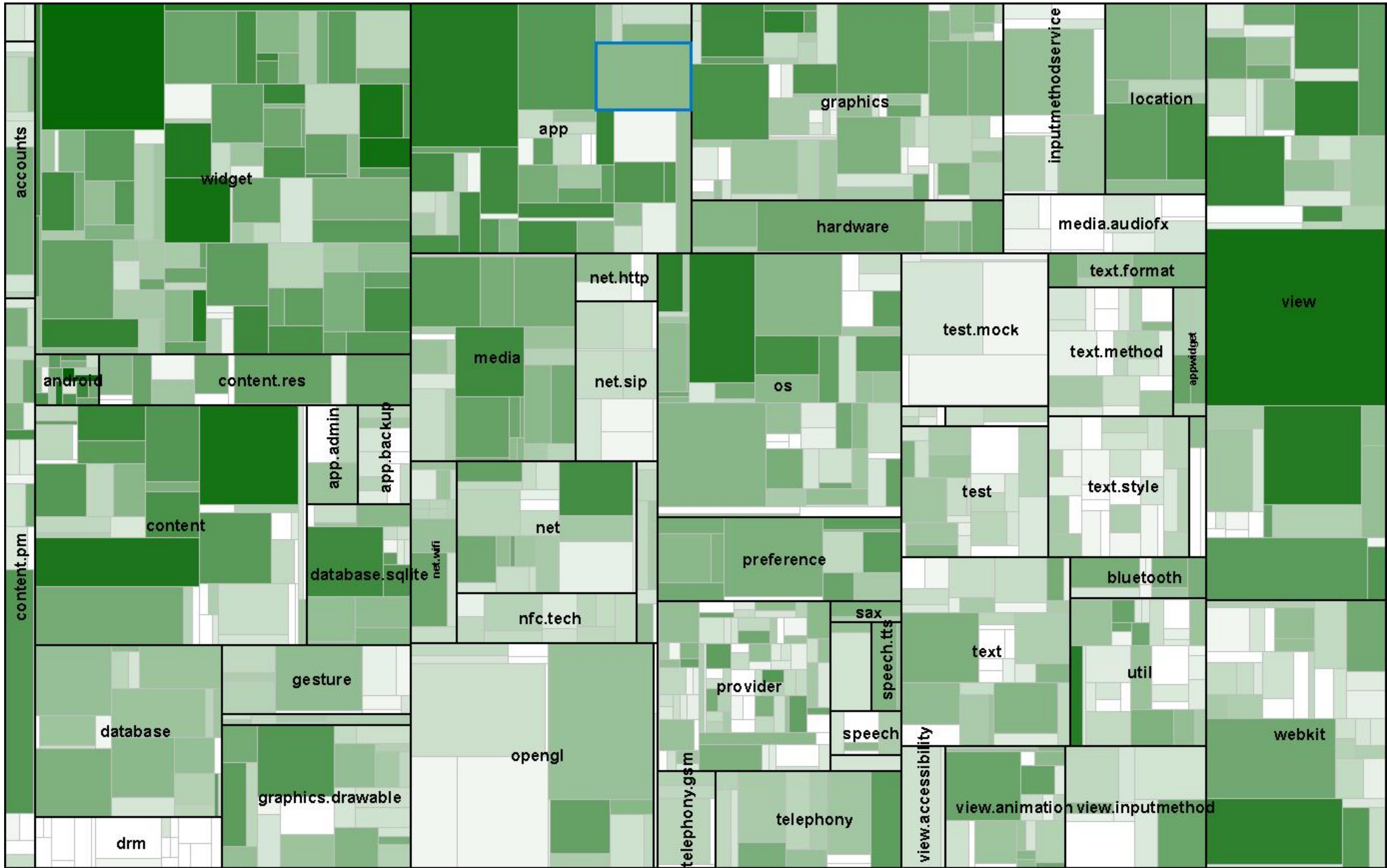
Output

LLM

Training Data

[Storey et al., 2024]

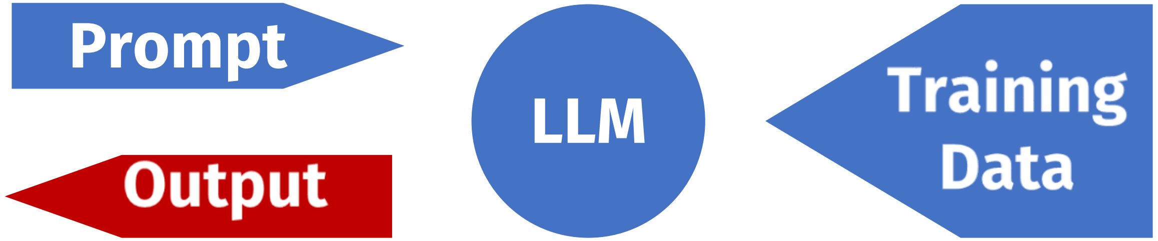
Stack Overflow replaces API documentation



[Parnin et al., 2012]

Analyzing LLM output

What do we lose from past solutions now rendered obsolete by this technology?



Analyzing LLM output

What do we lose from past solutions now rendered obsolete by this technology?

What human nuances are lost in **code** generated by LLMs?

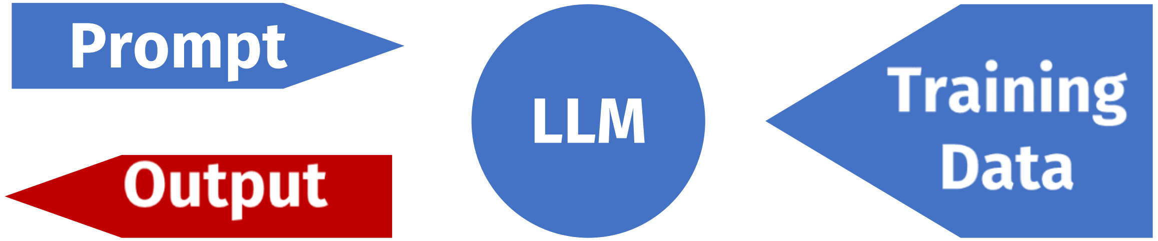
What do we miss from traditional **bug reports** with LLM error identification?

What collaborative and mentorship elements are lost with LLM **code reviews**?

What human insights are lost in **commit** documentation when handled by LLMs?

What human intuition is overlooked in LLM-generated **tests**?

What community aspects are lost when LLMs answer on **Stack Overflow**?



Analyzing LLM output

What do we lose from past solutions now rendered bsolete by this technology?

What human nuances are lost in **code** generated by

What do we miss from traditional **bug reports** error identification?

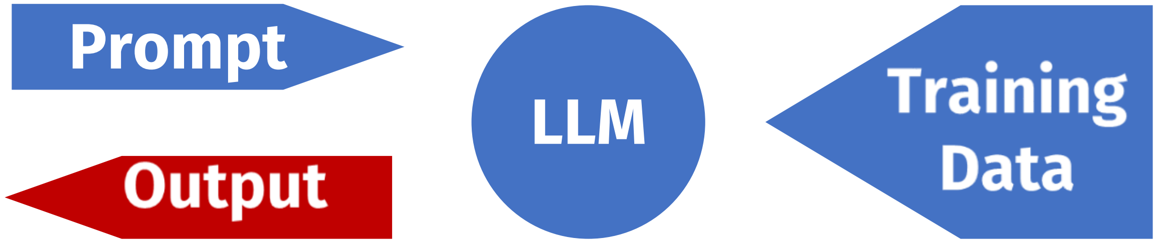
What collaborative and mentorship elem ost with LLM **code reviews**?

What human insights are lost in **co** mentation when handled by LLMs?

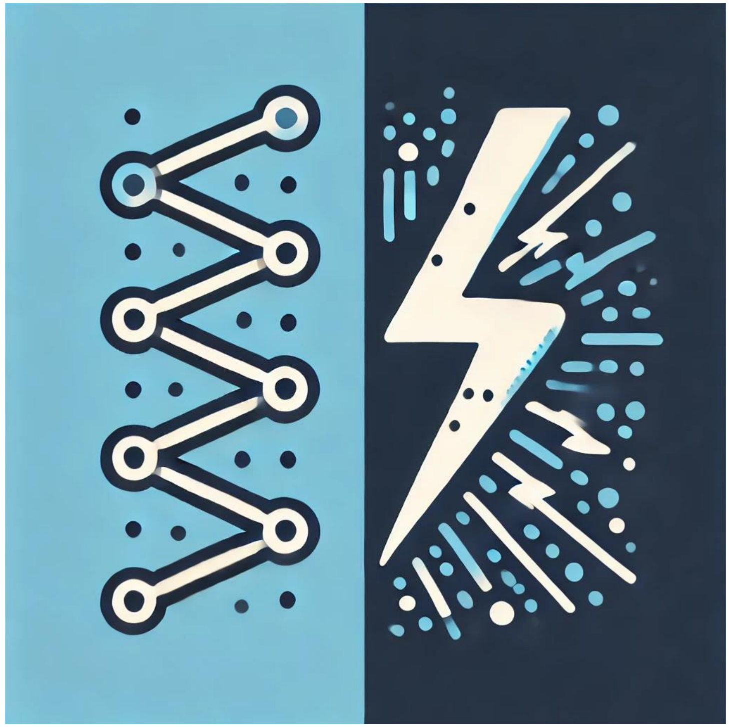
What human intuition is overl LM-generated **tests**?

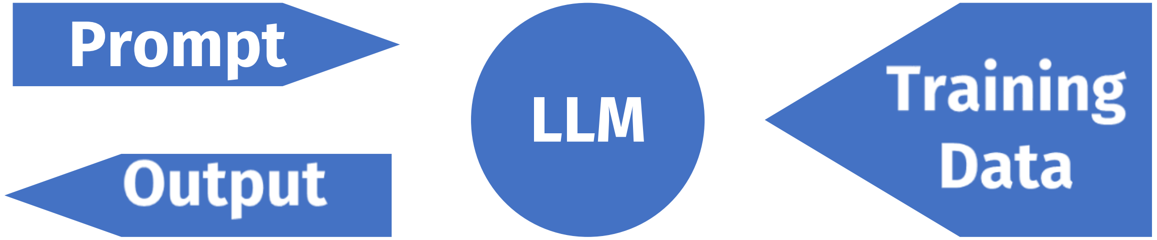
What community aspects hen LLMs answer on **Stack Overflow**?

LLM output beyond quality



EMSE Research in the Age of LLMs





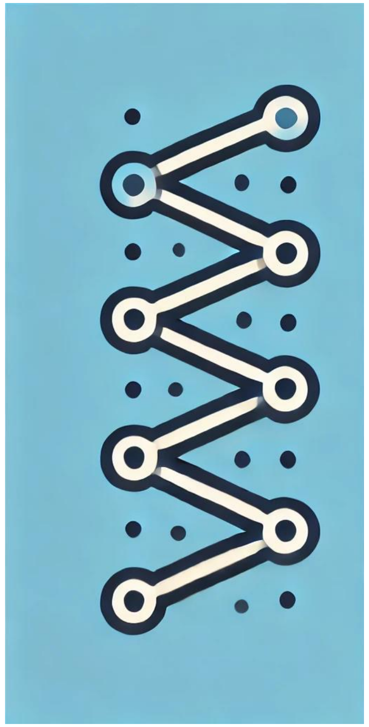
Prompt

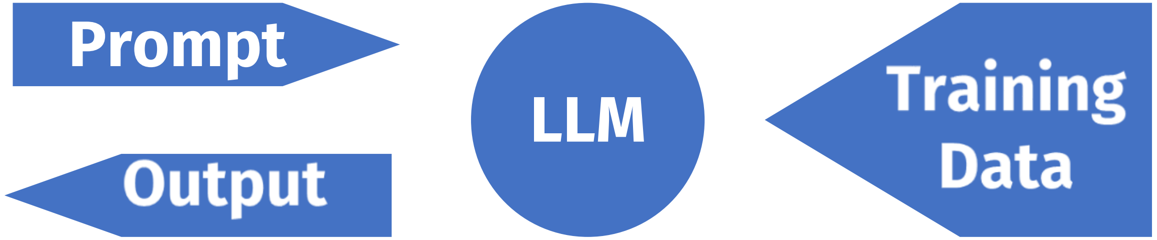
Output

LLM

Training Data

EMSE Research in the Age of LLMs





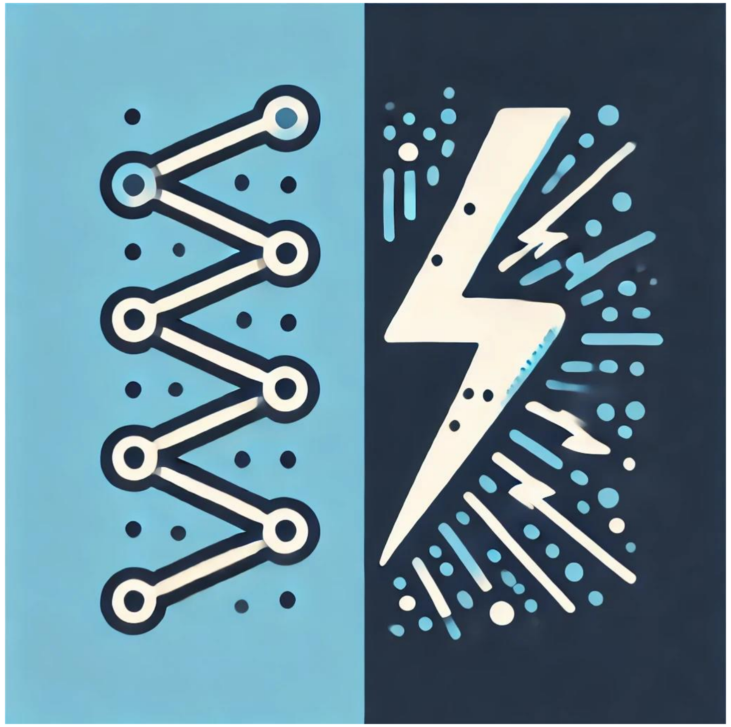
Prompt

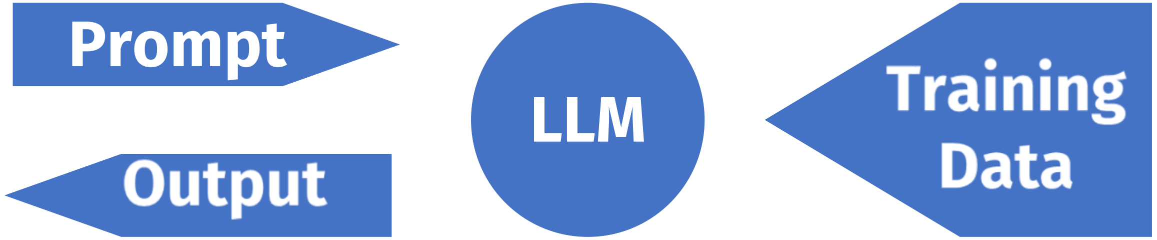
Output

LLM

Training Data

EMSE Research in the Age of LLMs





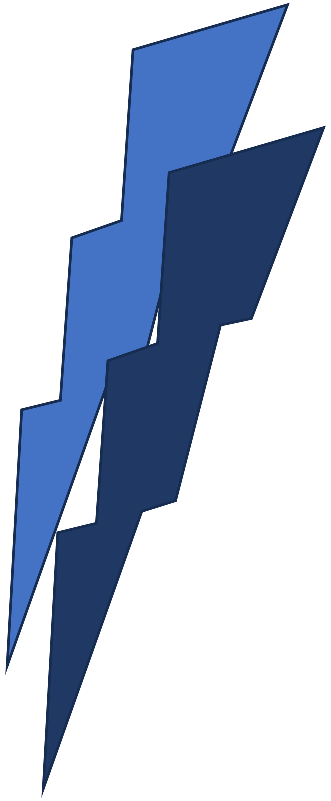
Prompt

Output

LLM

Training Data

Disruption



Disruption

Phenomena

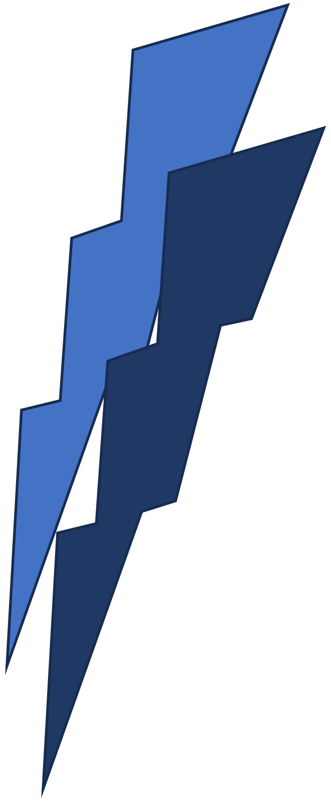
Methods

Theories

Threats

Ethics

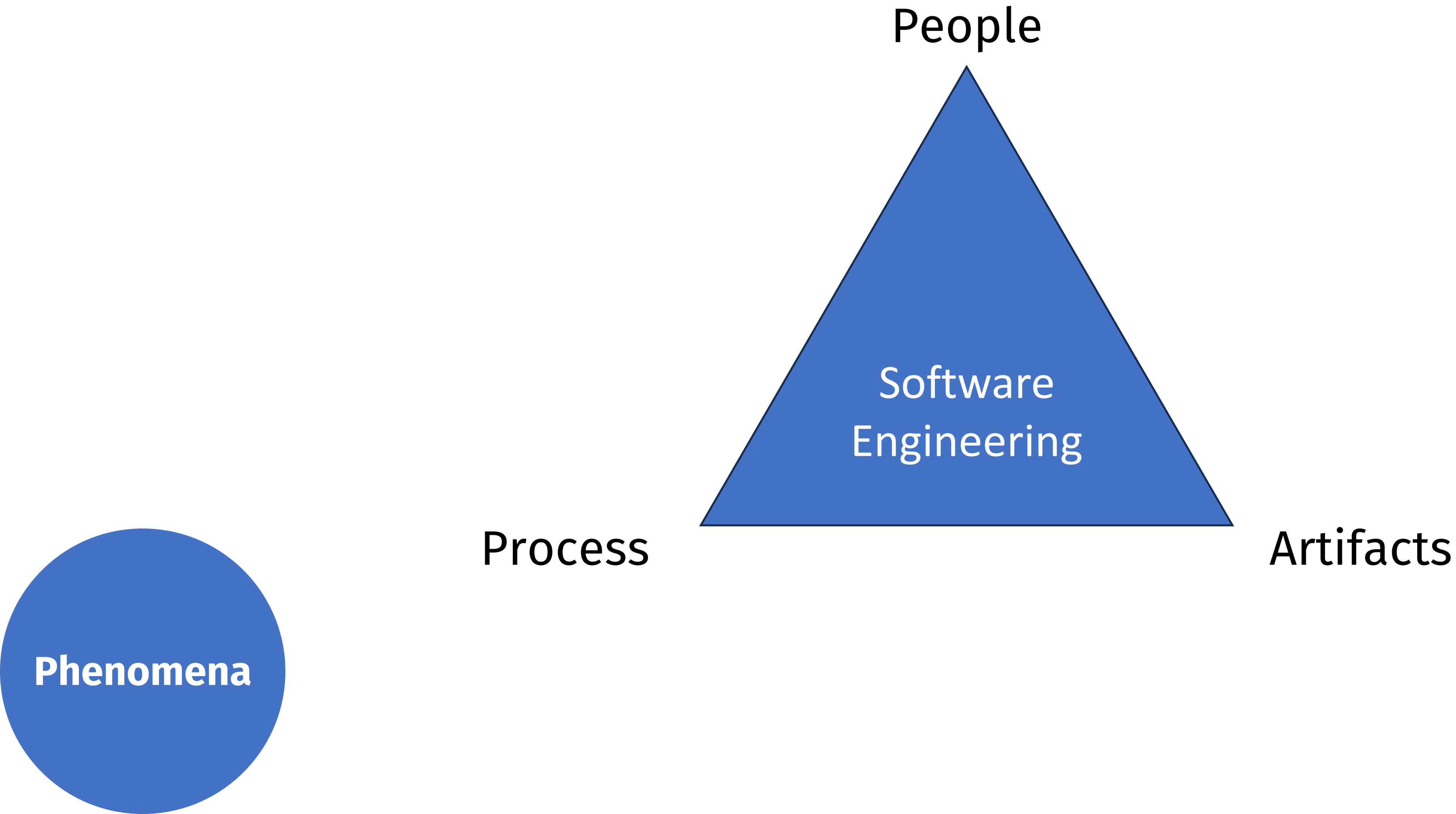
Tool



Disruption

Phenomena

Challenges existing definitions



People

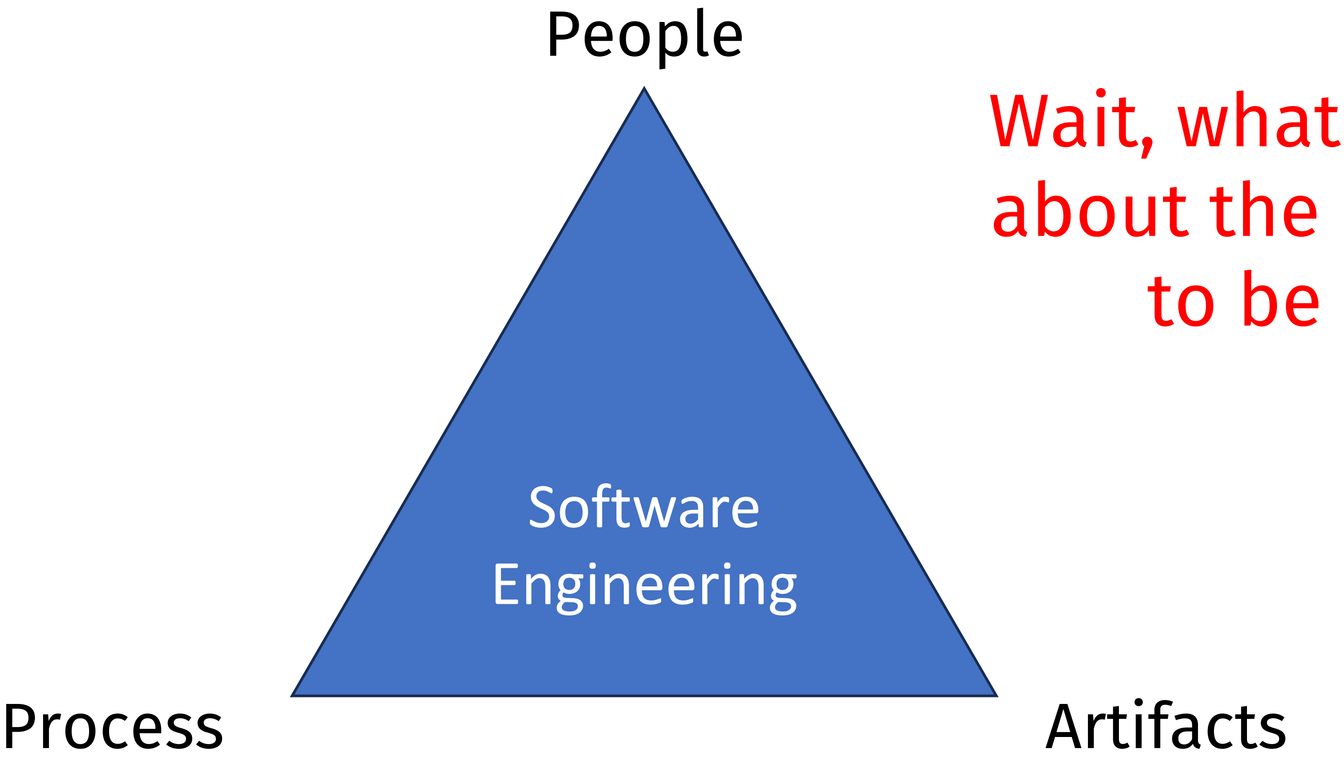
Software Engineering

Process

Artifacts

Phenomena

Challenges existing definitions



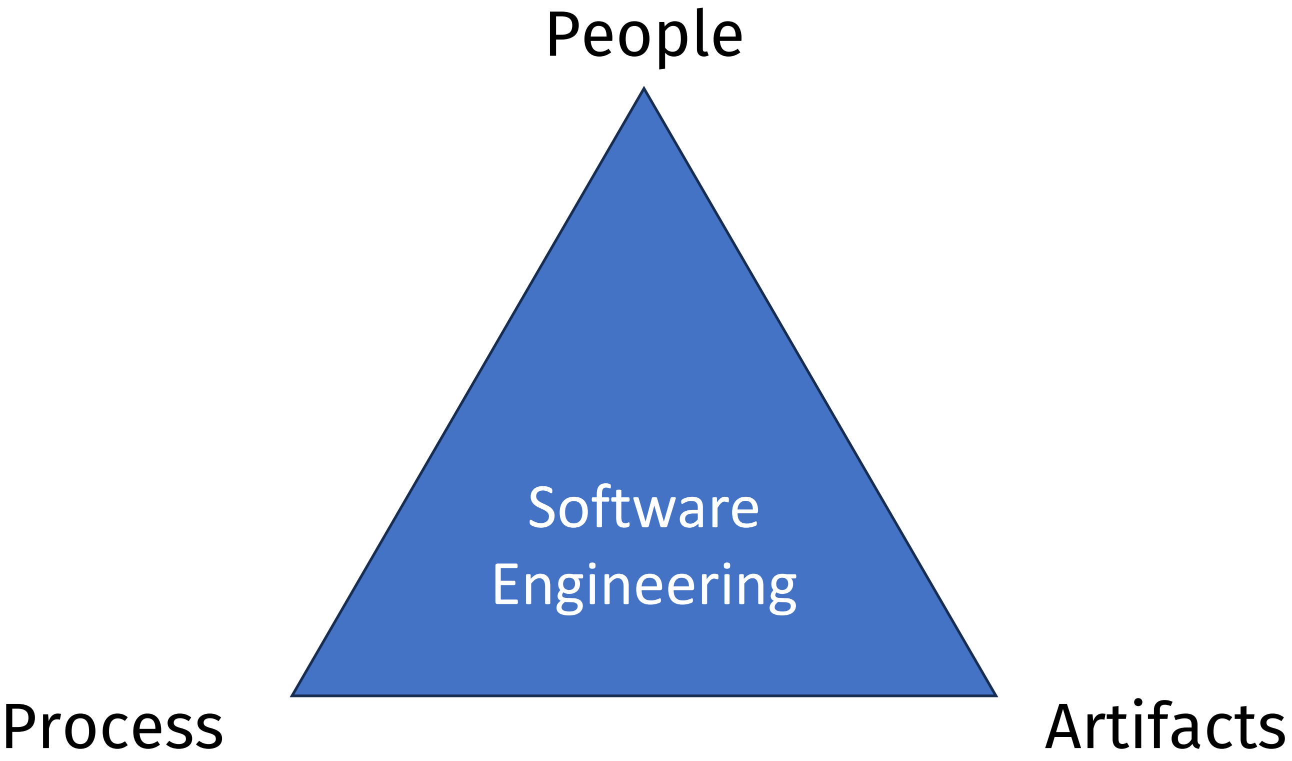
Wait, what do you know about the **phenomenon** to be studied?

Phenomena

Disruption

Methods

Mixed methods & interdisciplinary work



People

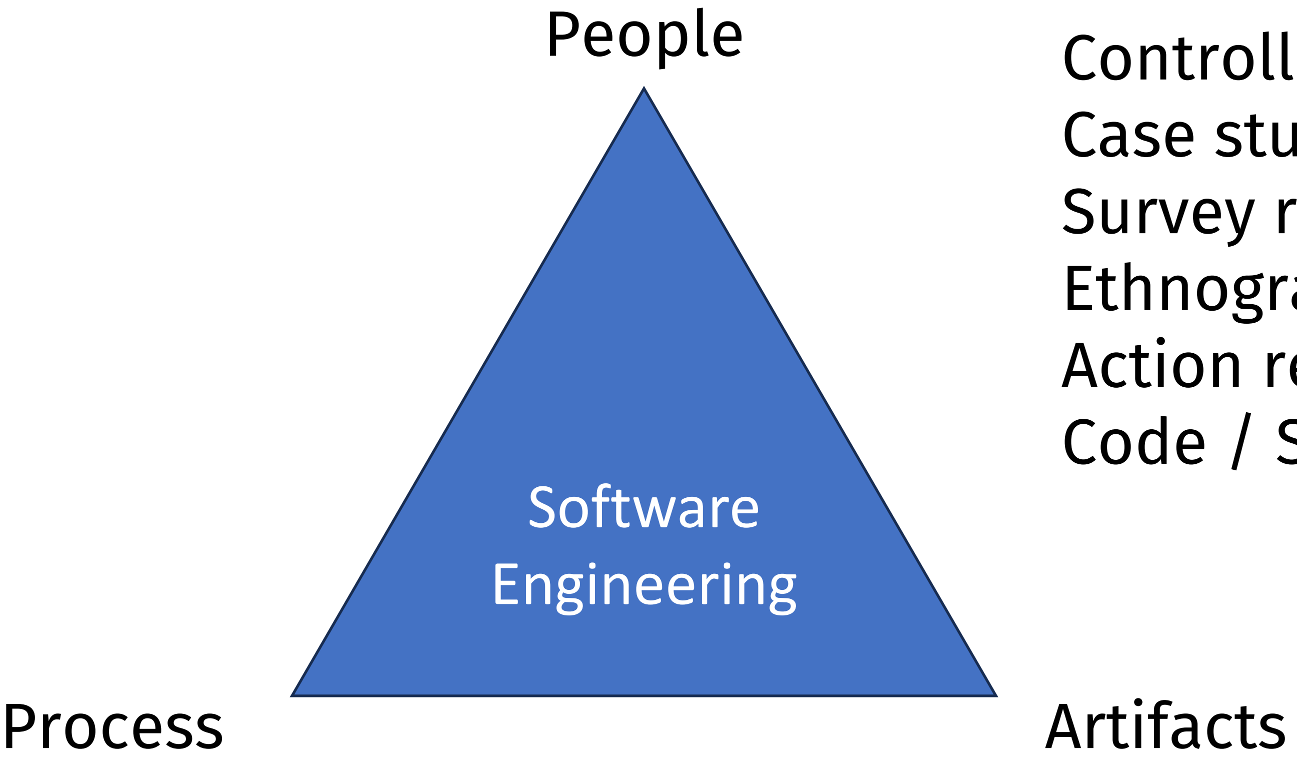
Software Engineering

Process

Artifacts

Methods

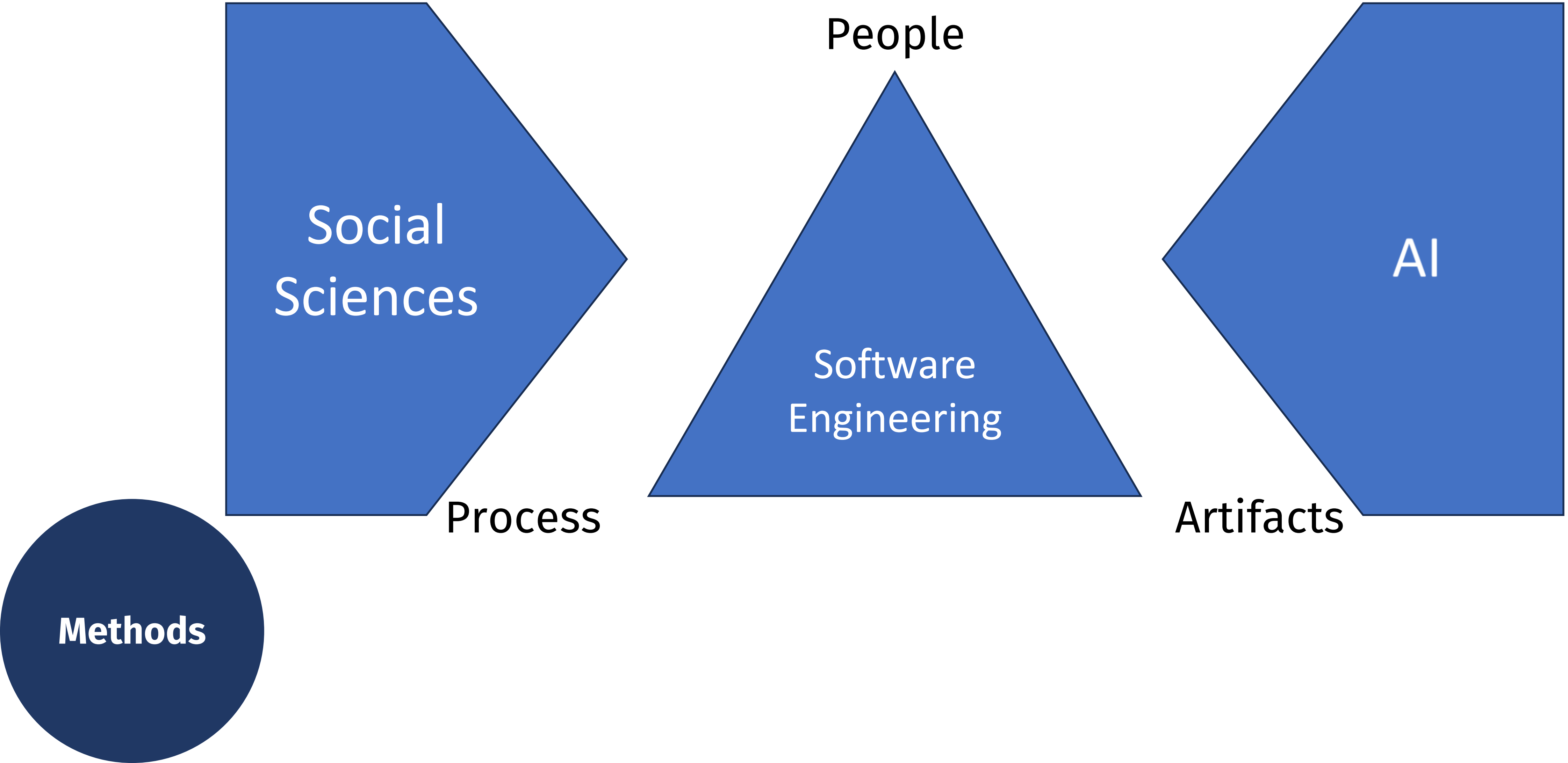
Mixed methods & interdisciplinary work



Controlled experiments
Case studies
Survey research
Ethnographies
Action research
Code / System analyses

Methods

Mixed methods & interdisciplinary work


Social Sciences

People

Software Engineering

Process

Artifacts

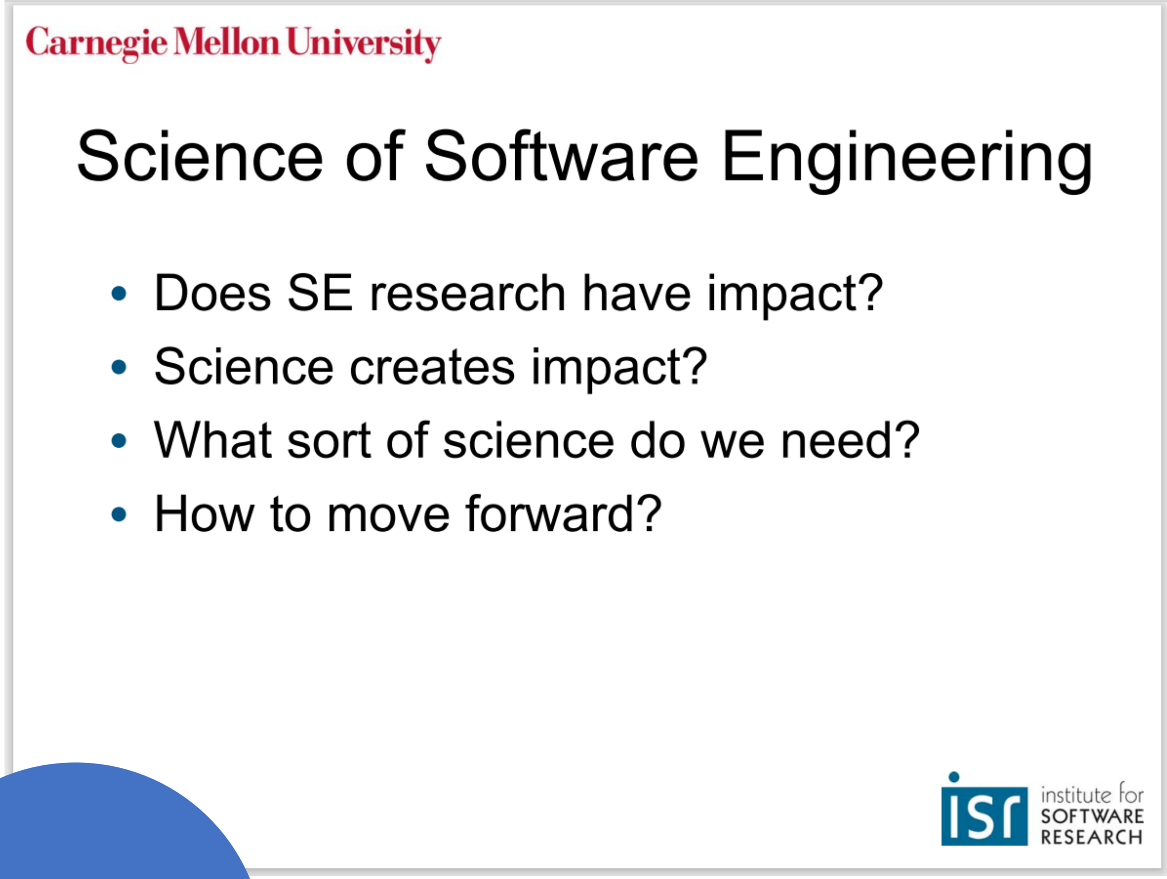
AI

Methods

Disruption

Theories

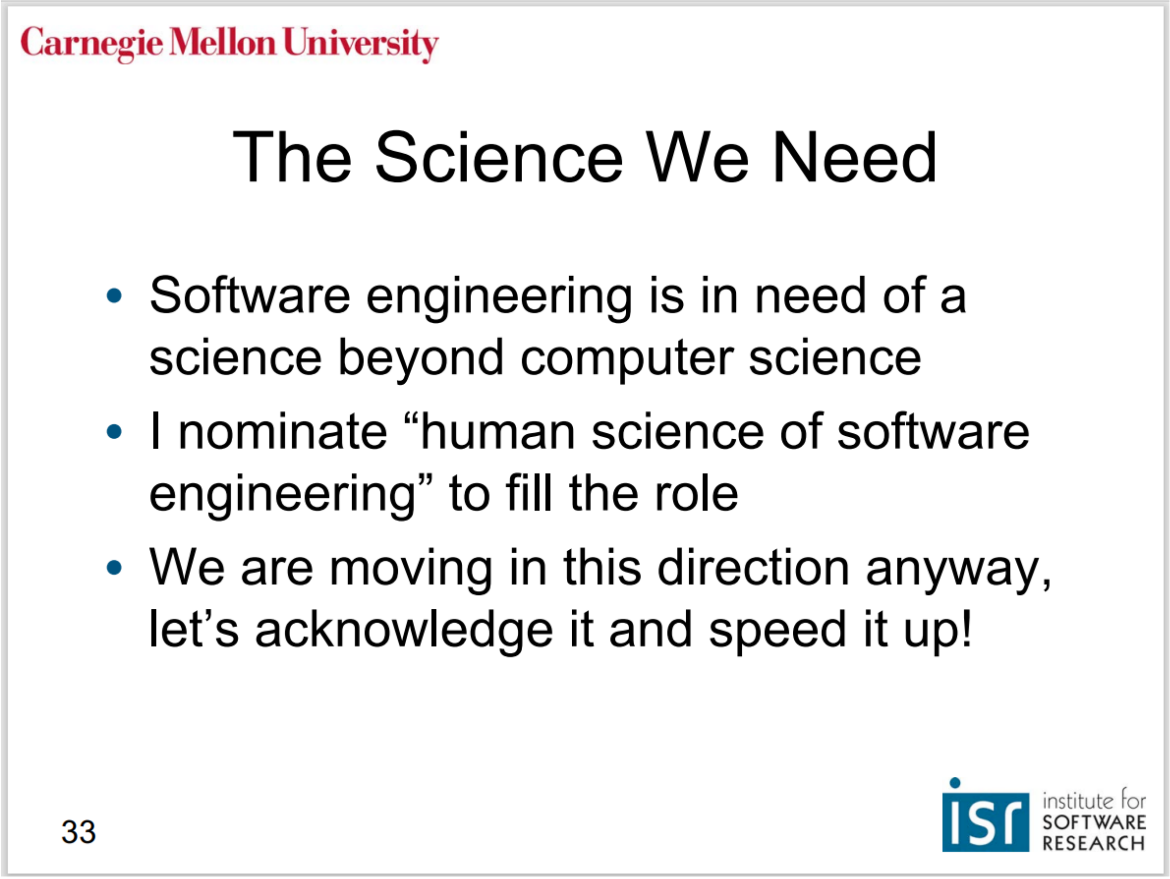
Understand and frame LLMs’ impact



Carnegie Mellon University

Science of Software Engineering

- Does SE research have impact?
- Science creates impact?
- What sort of science do we need?
- How to move forward?



Carnegie Mellon University

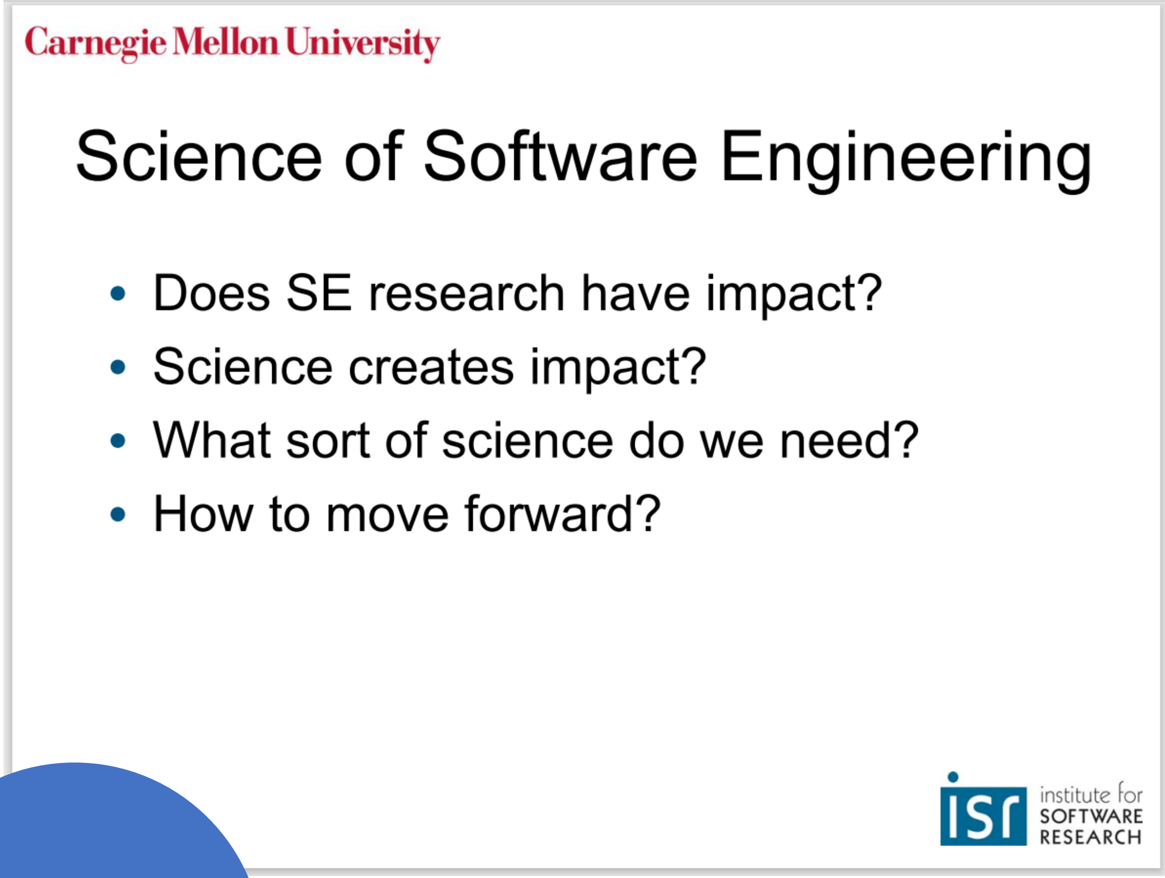
The Science We Need

- Software engineering is in need of a science beyond computer science
- I nominate “human science of software engineering” to fill the role
- We are moving in this direction anyway, let’s acknowledge it and speed it up!

33

Theories

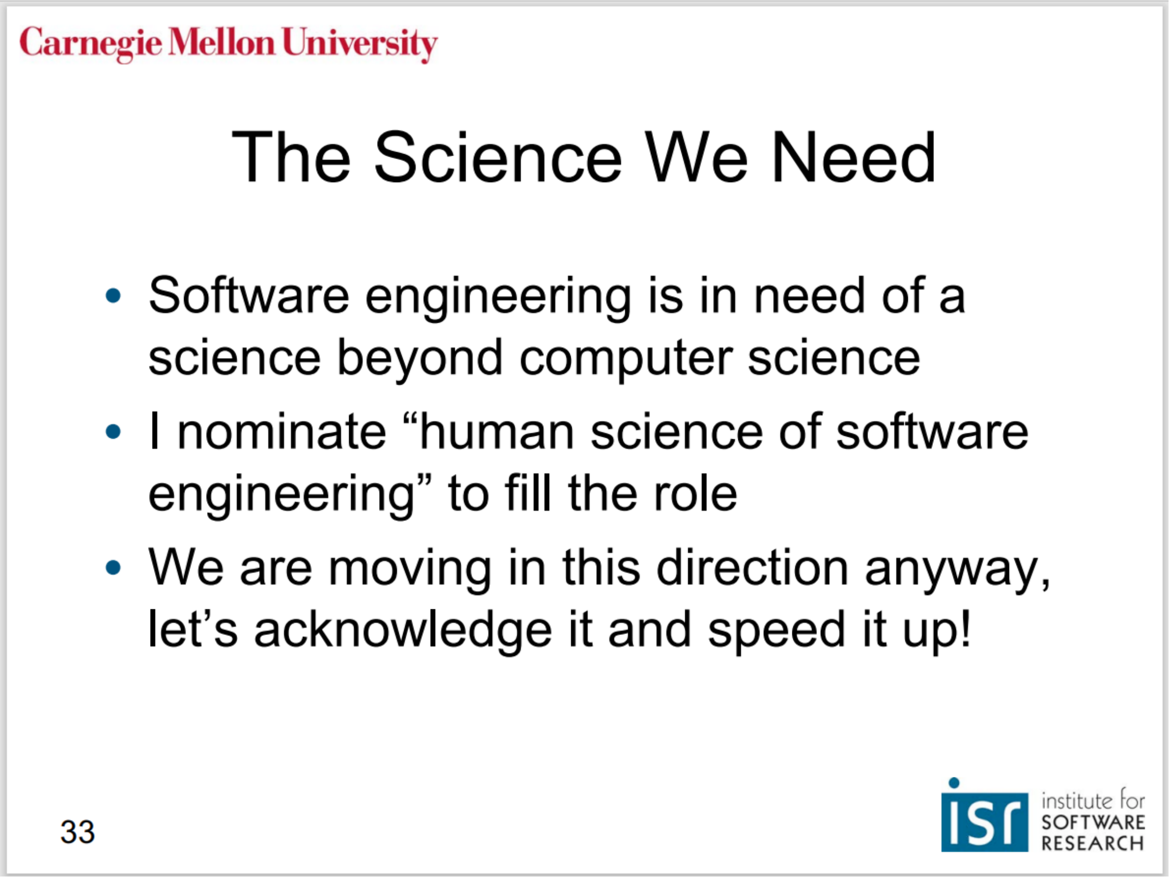
Understand and frame LLMs' impact



Carnegie Mellon University

Science of Software Engineering

- Does SE research have impact?
- Science creates impact?
- What sort of science do we need?
- How to move forward?



Carnegie Mellon University

The Science We Need

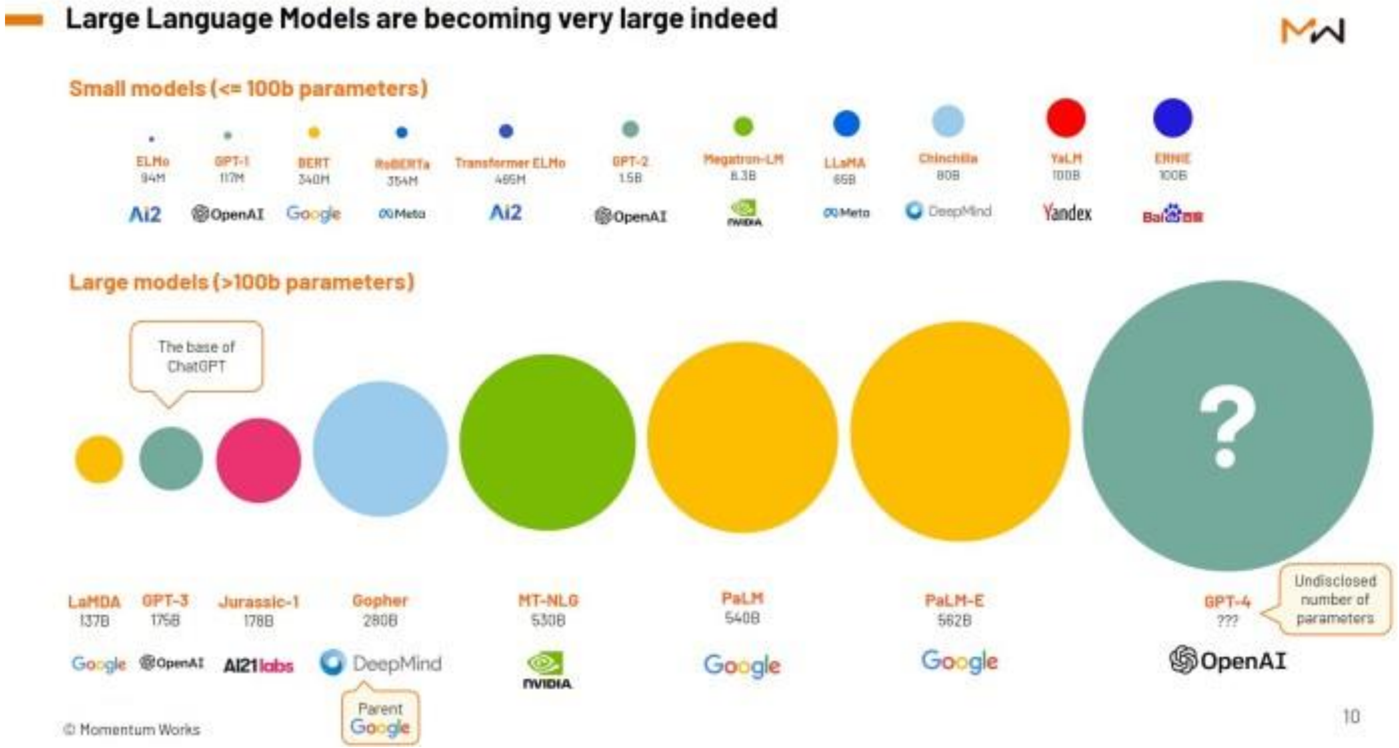
- Software engineering is in need of a science beyond computer science
- I nominate "human science of software engineering" to fill the role
- We are moving in this direction anyway, let's acknowledge it and speed it up!

Theories

Disruption

Threats

Non-deterministic and rapidly evolving



https://thelowdown.momentum.asia/the-emergence-of-large-language-models-llms/

Threats

Disruption

Ethics

Evolving legal and ethical frameworks

AI systems should respect human rights, diversity, and the autonomy of individuals.

Ethics

Evolving legal and ethical frameworks

AI systems should respect human rights, diversity, and the autonomy of individuals.

Documenting Ethical Considerations in Open Source AI Models

Haoyu Gao
The University of Melbourne
Victoria, Australia
haoyug1@student.unimelb.edu.au

Mansooreh Zahedi
The University of Melbourne
Victoria, Australia
mansooreh.zahedi@unimelb.edu.au

Christoph Treude
Singapore Management University
Singapore
ctreude@smu.edu.sg

Sarita Rosenstock
The University of Melbourne
Victoria, Australia
sarita.rosenstock@unimelb.edu.au

Marc Cheong
The University of Melbourne
Victoria, Australia
marc.cheong@unimelb.edu.au

2024

ABSTRACT

Background: The development of AI-enabled software heavily depends on AI model documentation, such as model cards, due to

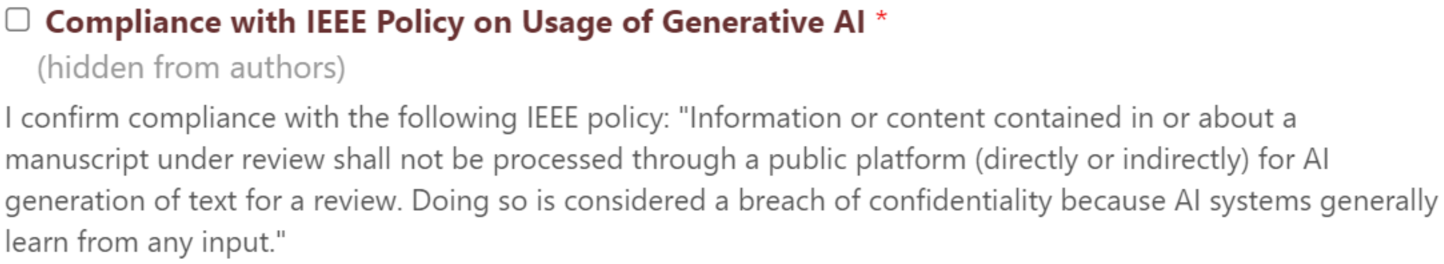
Models. In *Proceedings of 18th International Symposium on Empirical Software Engineering and Measurement (ESEM '24)*. ACM, New York, NY, USA, 12 pages. https://doi.org/XXXXXXX.XXXXXXX

Ethics

Disruption

Tool

AI-assisted SE research



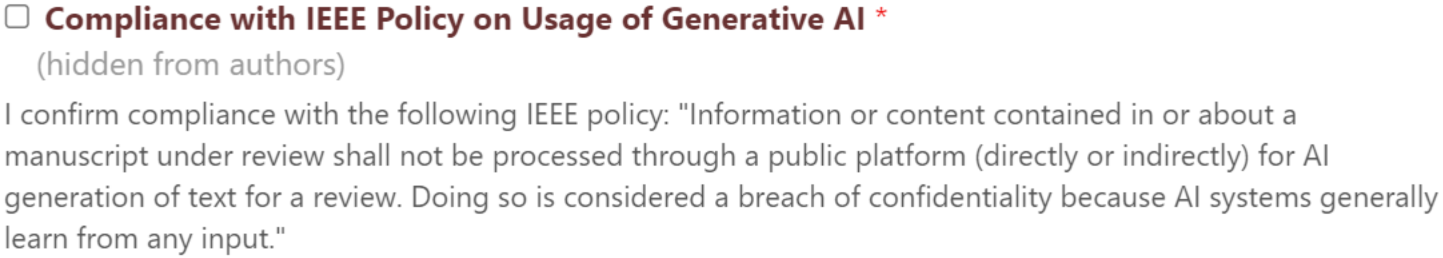
☐ **Compliance with IEEE Policy on Usage of Generative AI** *

(hidden from authors)

I confirm compliance with the following IEEE policy: "Information or content contained in or about a manuscript under review shall not be processed through a public platform (directly or indirectly) for AI generation of text for a review. Doing so is considered a breach of confidentiality because AI systems generally learn from any input."

Tool

AI-assisted SE research

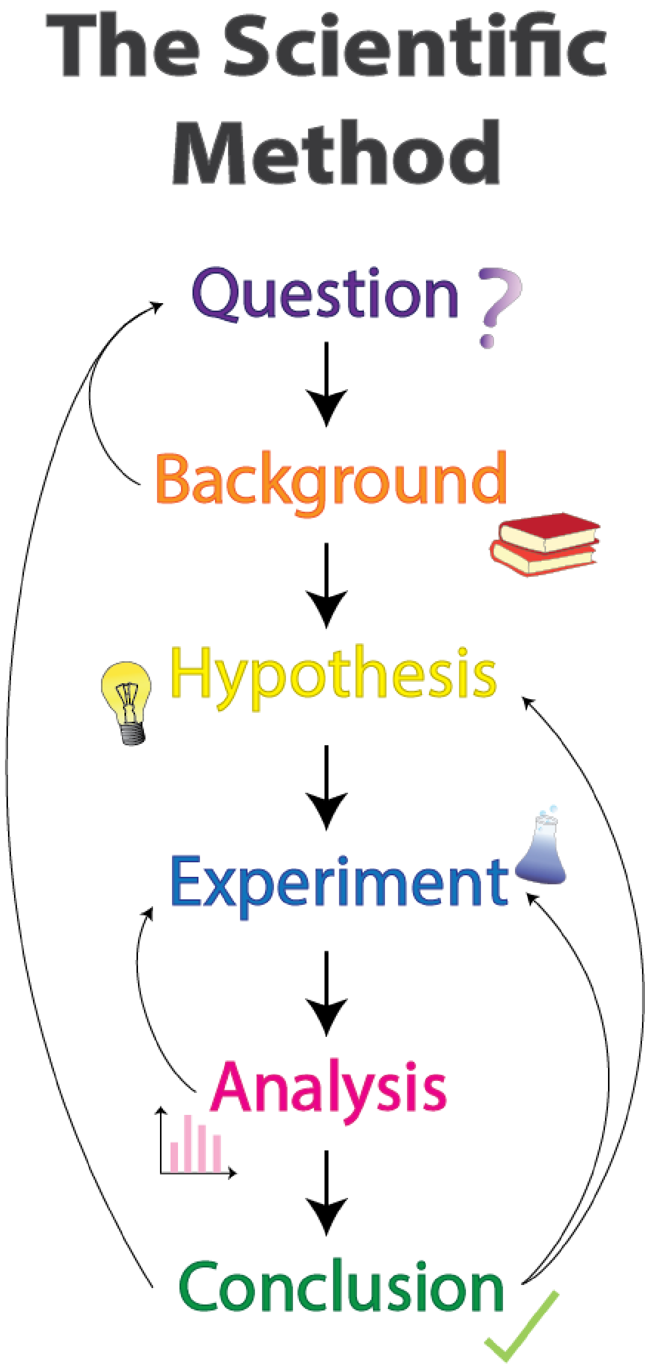


☐ **Compliance with IEEE Policy on Usage of Generative AI** *

(hidden from authors)

I confirm compliance with the following IEEE policy: "Information or content contained in or about a manuscript under review shall not be processed through a public platform (directly or indirectly) for AI generation of text for a review. Doing so is considered a breach of confidentiality because AI systems generally learn from any input."

Tool



The Scientific Method

Question

Background

Hypothesis

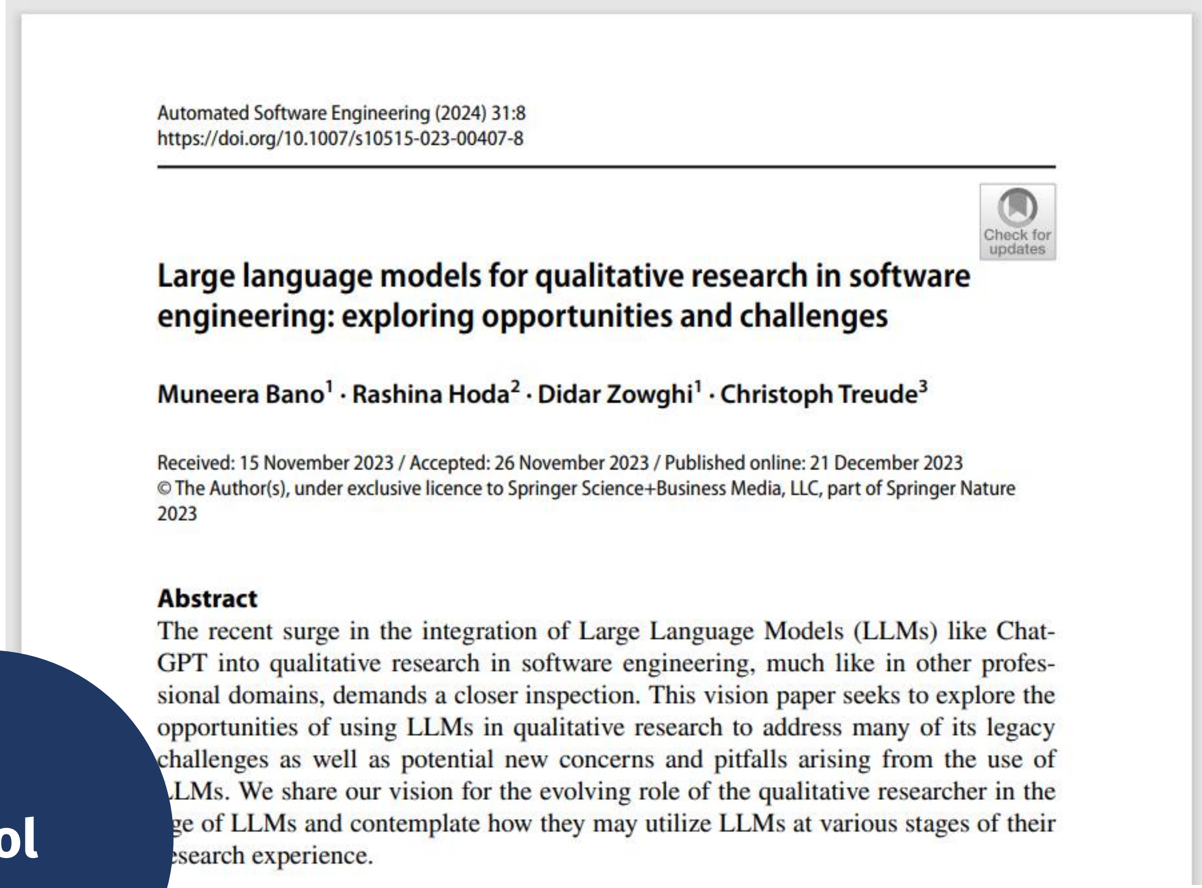
Experiment

Analysis

Conclusion

https://study.com/learn/lesson/scientific-method-example-steps.html

AI-assisted SE research



Automated Software Engineering (2024) 31:8
https://doi.org/10.1007/s10515-023-00407-8

Check for updates

Large language models for qualitative research in software engineering: exploring opportunities and challenges

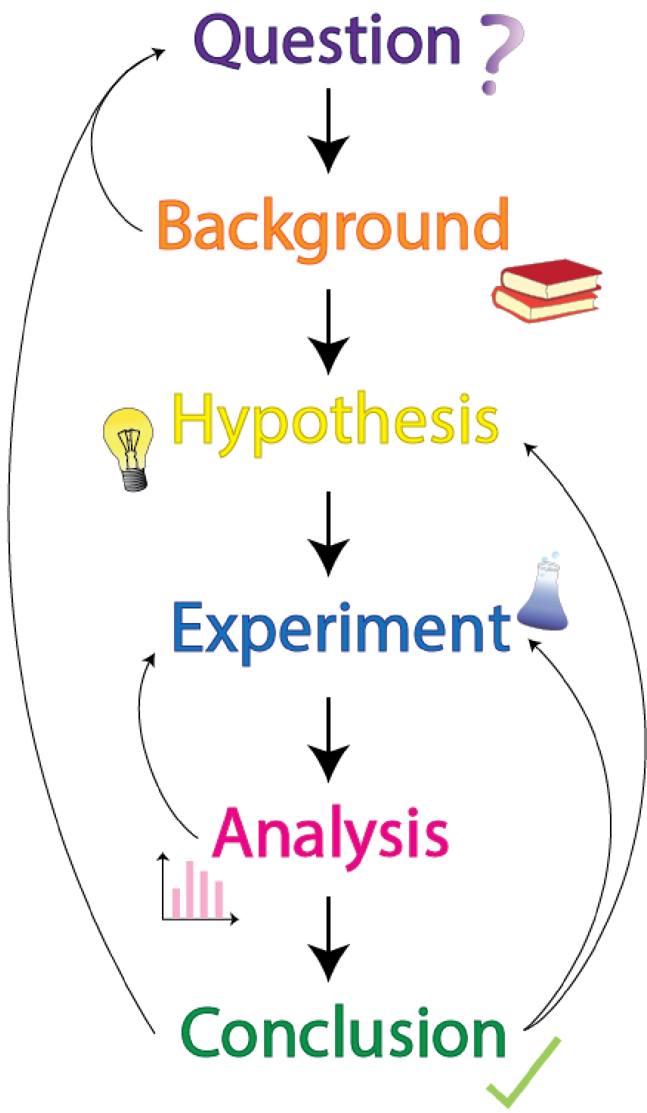
Muneera Bano[1] · Rashina Hoda[2] · Didar Zowghi[1] · Christoph Treude[3]

Received: 15 November 2023 / Accepted: 26 November 2023 / Published online: 21 December 2023
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2023

Abstract
The recent surge in the integration of Large Language Models (LLMs) like ChatGPT into qualitative research in software engineering, much like in other professional domains, demands a closer inspection. This vision paper seeks to explore the opportunities of using LLMs in qualitative research to address many of its legacy challenges as well as potential new concerns and pitfalls arising from the use of LMs. We share our vision for the evolving role of the qualitative researcher in the ge of LLMs and contemplate how they may utilize LLMs at various stages of their esearch experience.

Tool

The Scientific Method



Question

Background

Hypothesis

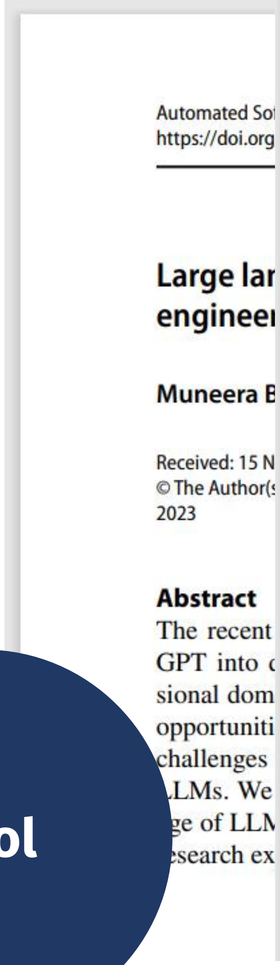
Experiment

Analysis

Conclusion

https://study.com/learn/lesson/scientific-method-example-steps.html

AI-assisted SE research



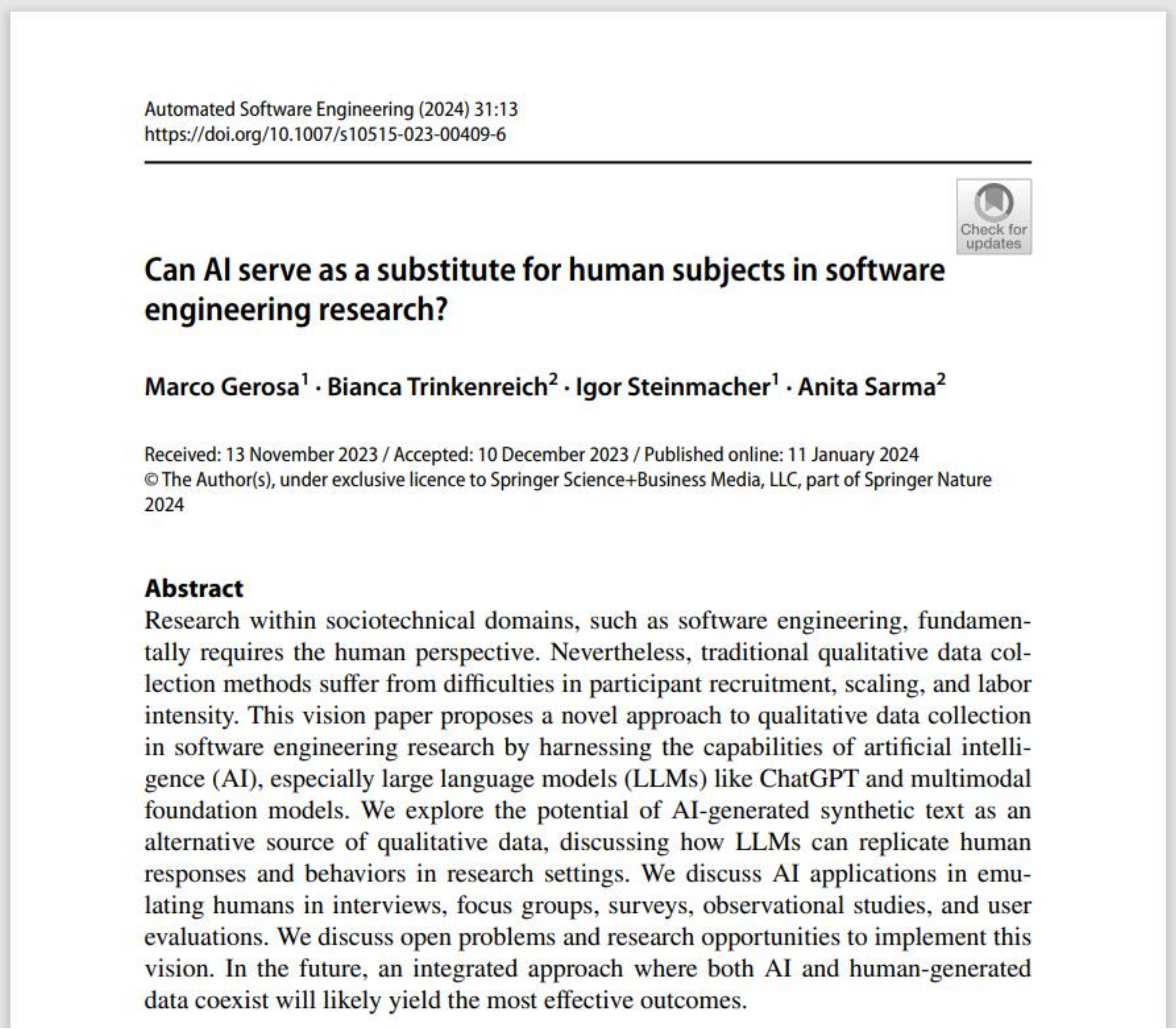
Automated So
https://doi.org

Large lar
engineer

Muneera B

Received: 15 N
© The Author(
2023

Abstract
The recent
GPT into
sional dom
opportuniti
challenges
LLMs. We
ge of LLM
esearch ex



Automated Software Engineering (2024) 31:13
https://doi.org/10.1007/s10515-023-00409-6

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

Marco Gerosa[1] · Bianca Trinkenreich[2] · Igor Steinmacher[1] · Anita Sarma[2]

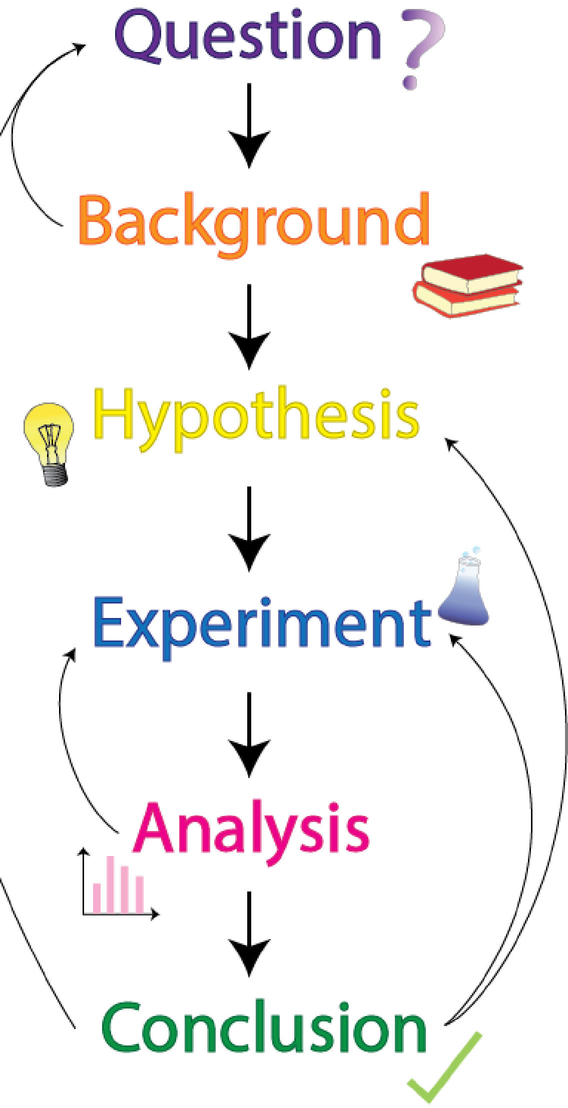
Received: 13 November 2023 / Accepted: 10 December 2023 / Published online: 11 January 2024
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

Abstract

Research within sociotechnical domains, such as software engineering, fundamentally requires the human perspective. Nevertheless, traditional qualitative data collection methods suffer from difficulties in participant recruitment, scaling, and labor intensity. This vision paper proposes a novel approach to qualitative data collection in software engineering research by harnessing the capabilities of artificial intelligence (AI), especially large language models (LLMs) like ChatGPT and multimodal foundation models. We explore the potential of AI-generated synthetic text as an alternative source of qualitative data, discussing how LLMs can replicate human responses and behaviors in research settings. We discuss AI applications in emulating humans in interviews, focus groups, surveys, observational studies, and user evaluations. We discuss open problems and research opportunities to implement this vision. In the future, an integrated approach where both AI and human-generated data coexist will likely yield the most effective outcomes.

Tool

The Scientific Method



Question

Background

Hypothesis

Experiment

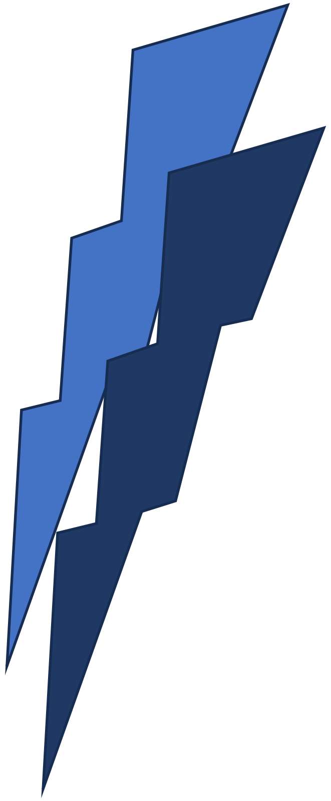
Analysis

Conclusion

https://study.com/learn/lesson/scientific-method-example-steps.html

Disruption

- Phenomena
- Methods
- Theories
- Threats
- Ethics
- Tool

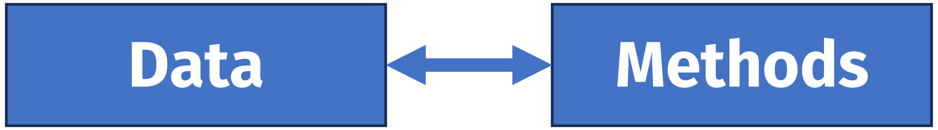


The Big Picture

Data

> “Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis.”
>
> [Harrison and Basili]

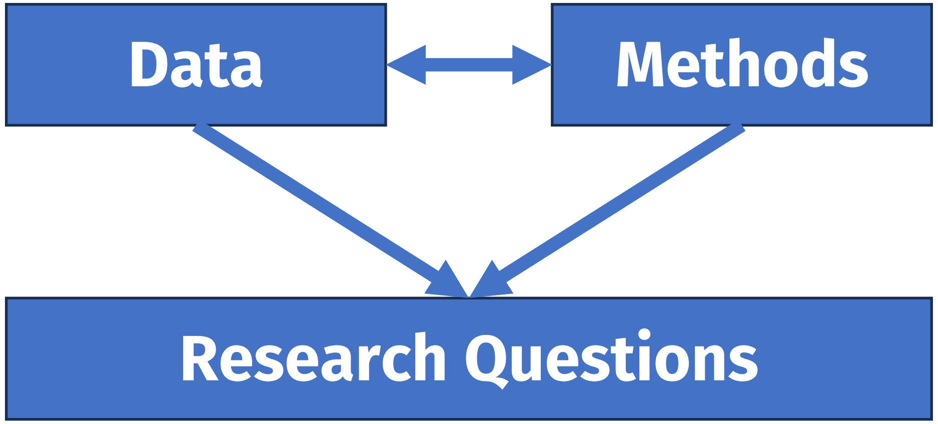
The Big Picture



“Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis.”

[Harrison and Basili]

The Big Picture



Data

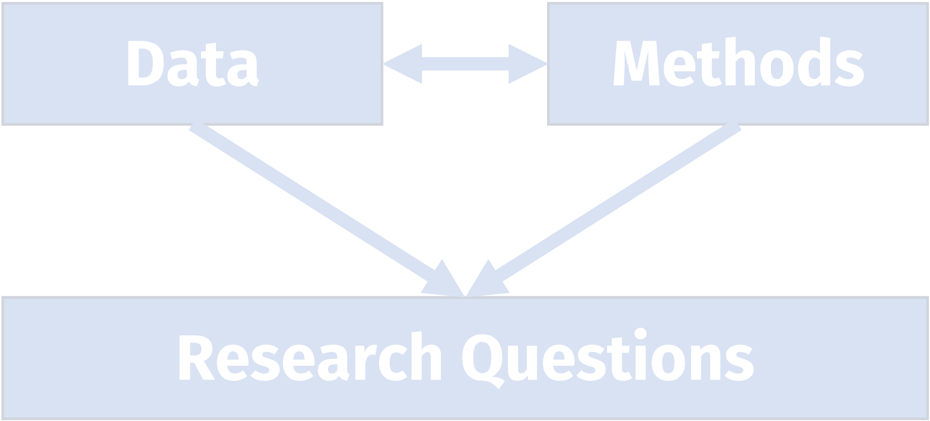
Methods

Research Questions

“Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis.”

[Harrison and Basili]

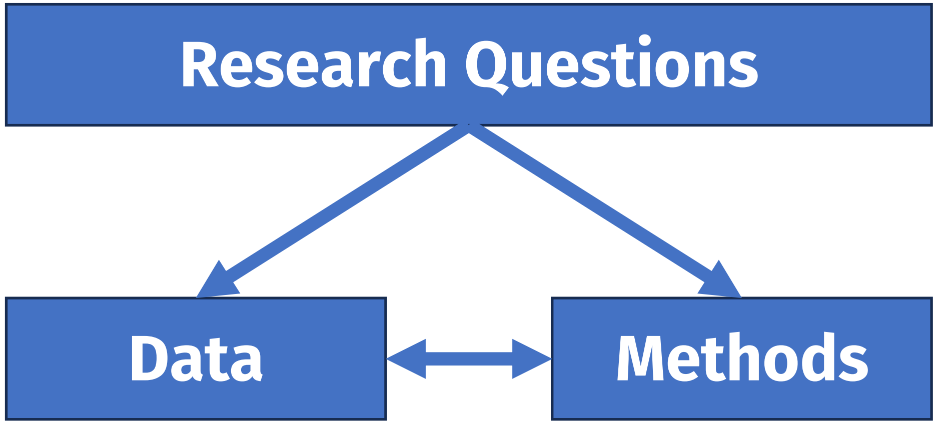
The Big Picture



Data

Methods

Research Questions



Research Questions

Data

Methods

"Empirical software engineering is the study of **software-related artifacts** for the characterization, understanding, evaluation, prediction, control, management, or improvement through qualitative or quantitative analysis."

[Harrison and Basili]

"Quantifying the evidence or making sense of it in qualitative form, a researcher can answer **empirical questions**, which should be clearly defined and answerable with the evidence collected (usually called data)."

[Wikipedia]

The Role of EMSE in the LLM Era

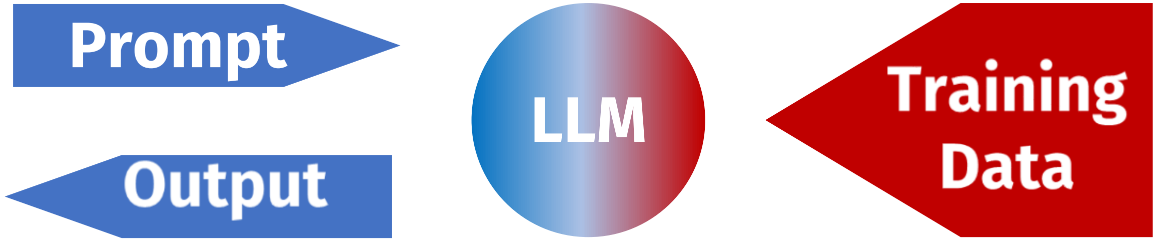
Prompt

Output

LLM

Training Data

The Role of EMSE in the LLM Era



Prompt

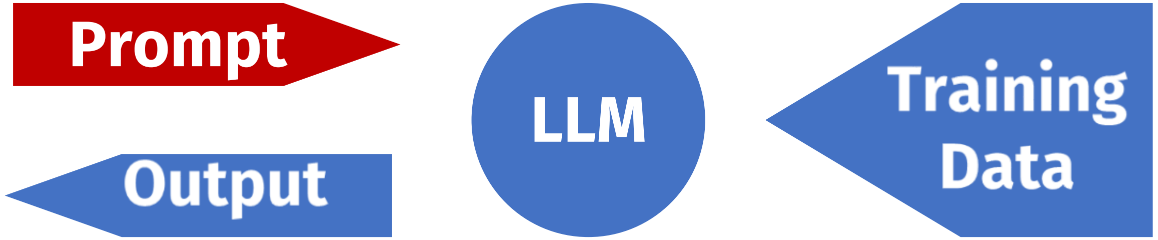
Output

LLM

Training Data

We know the data!

The Role of EMSE in the LLM Era



LLM interactions are data, too

Prompt

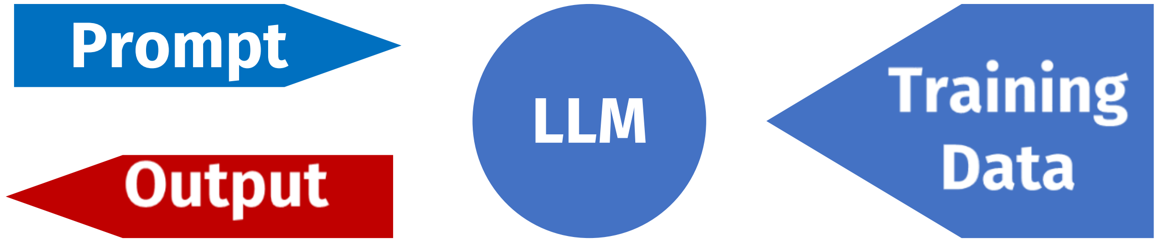
Output

LLM

Training Data

We know the data!

The Role of EMSE in the LLM Era

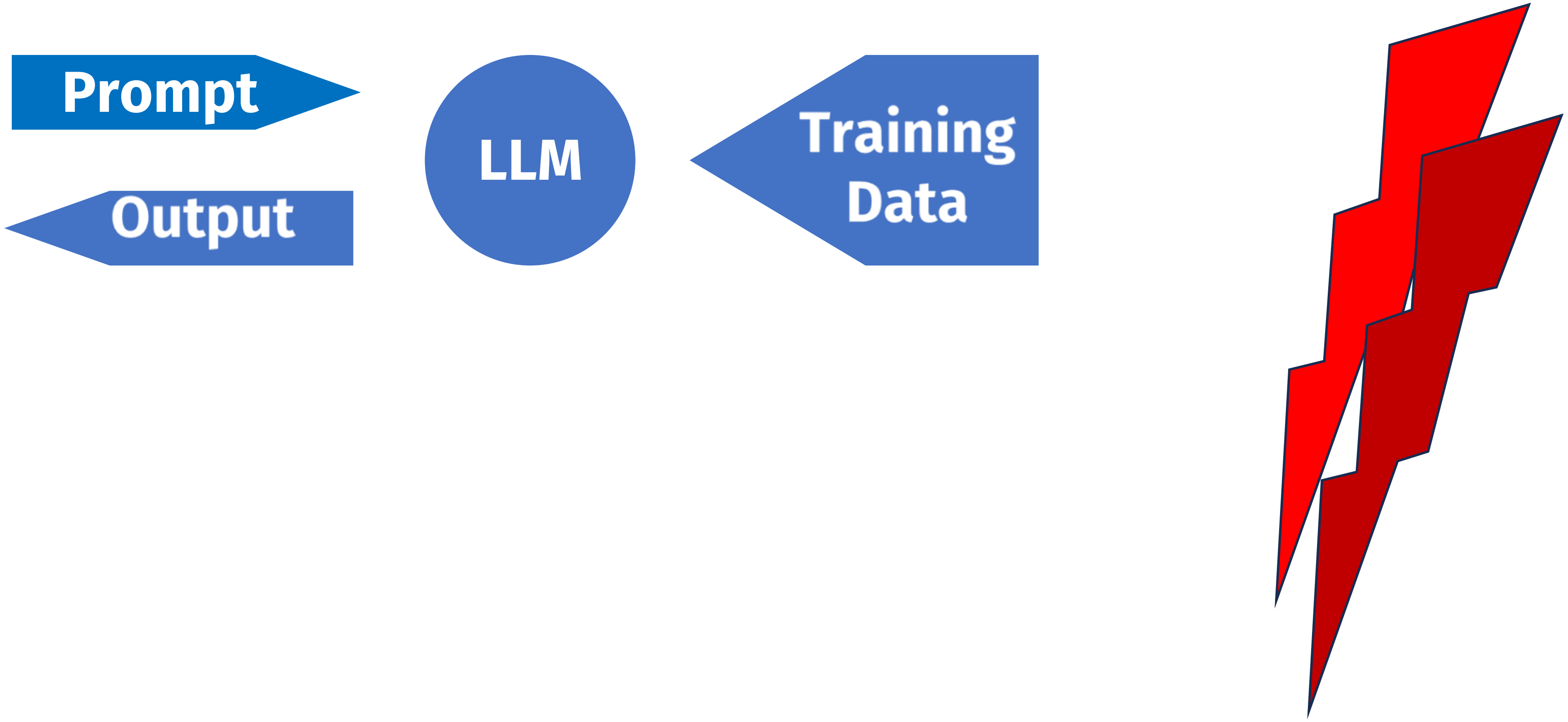


LLM interactions are data, too

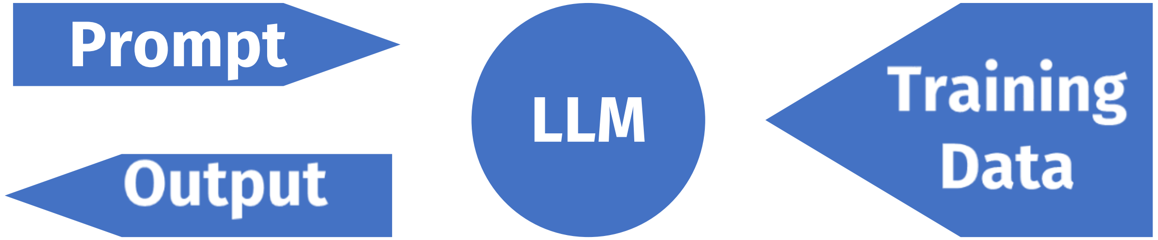
LLM output beyond quality

We know the data!

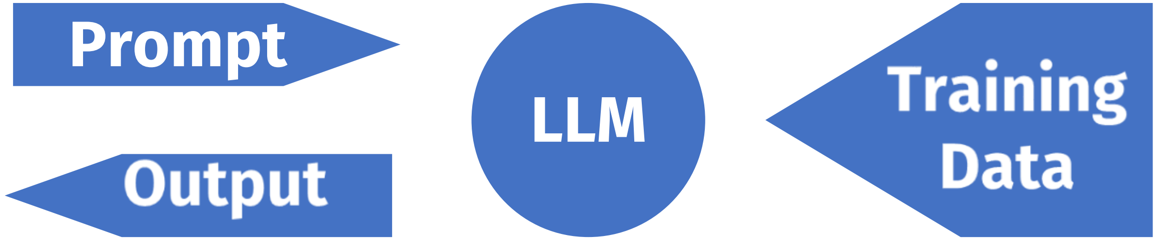
The Role of EMSE in the LLM Era



Let’s start with the research questions!



Let’s start with the research questions!



ctreude@smu.edu.sg