

大規模言語モデルとソフトウェア工学～ ChatGPTはソフトウェア工学に変革をもたら らるか [menti.com 62447074](https://menti.com/62447074)

座長

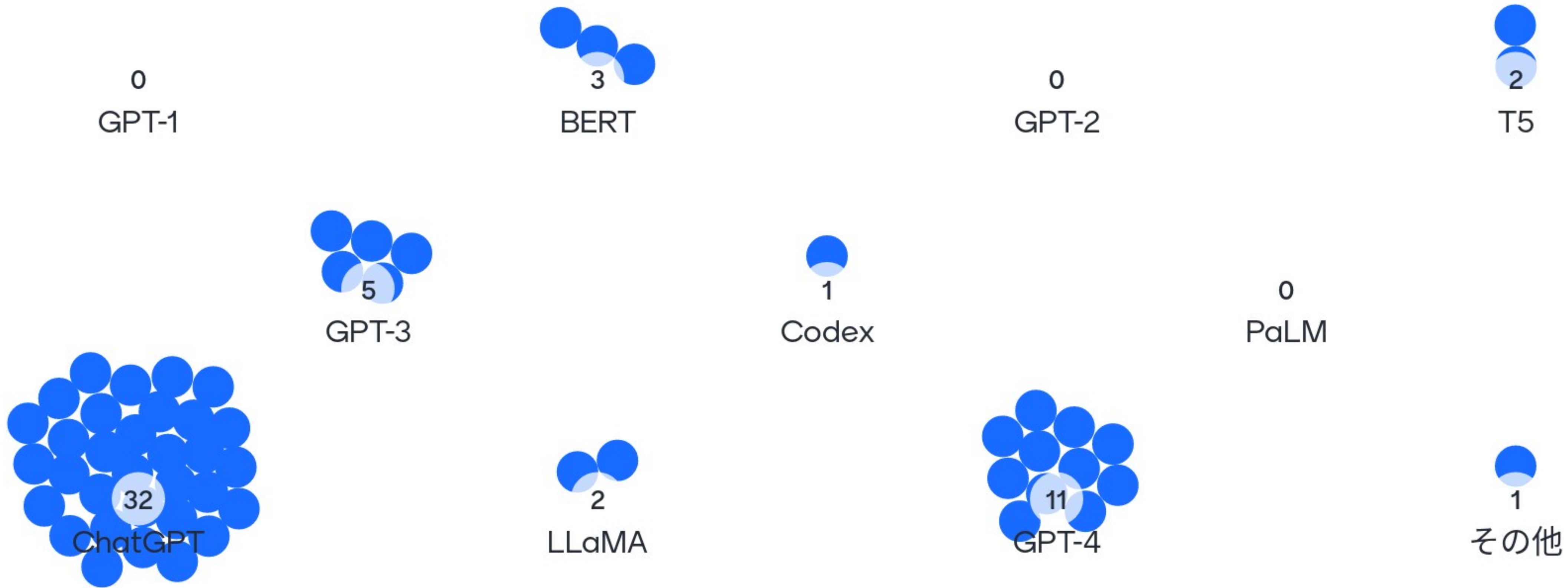
鷺崎 弘宜 (早稲田大学)

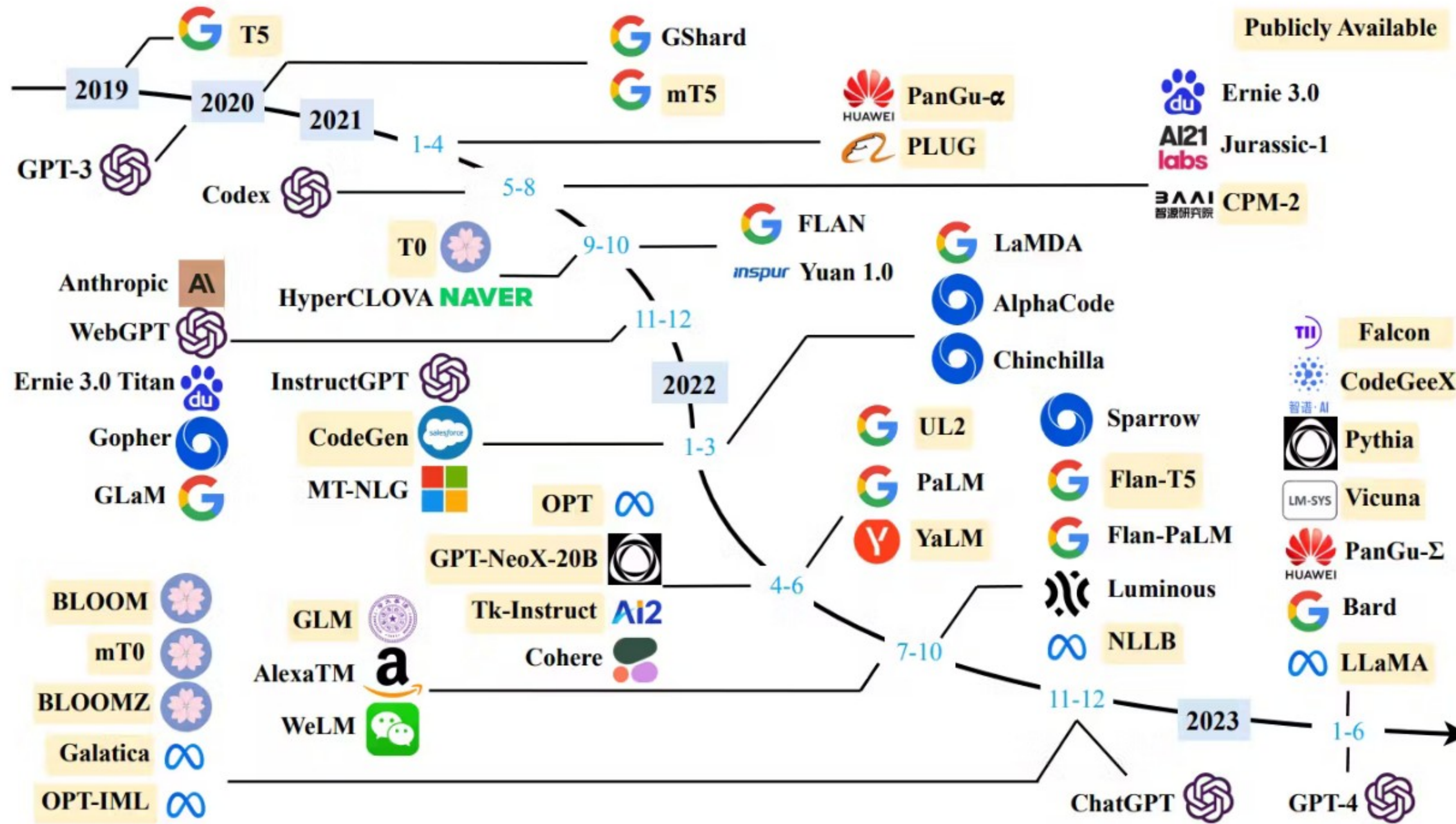
パネリスト

石川 冬樹 (国立情報学研究所)、倉光 君郎 (日本女子大学)、丸山 宏 (花王)、三浦 真樹 (富士通)

ChatGPTを始めとする大規模言語モデル(LLM)の進化は、ソフトウェアエンジニアリングの実践と研究に対して新たな展望を提示しています。本パネルでは、LLMがソフトウェア開発のパラダイムやエンジニアのスキルセットにどのような影響を及ぼし、研究者と実務者がどのように変化に適応していくべきかについて議論します。

LLM、何を使ったことがありますか？





A Survey of Large Language Models (arXiv:2303.18223)

直近1か月以内でLLMを主にソフトウェアの開発や教育、研究に用いましたか？



Agenda



- ・ 生成AIの利用を前提にしたソフトウェア工学教育
 - 新しい教育の「コツ」共有
- ・ ソフトウェア工学が対象とするようなソフトウェアの開発における生成AI
 - 「普通の」生成AIだと・・・
 - ・ プロンプトエンジニアリングの重要性
 - いわゆる「コード生成AI」(GitHub Copilot, OpenAI Codex)だと・・・
 - ・ コード生成AIの拡がり
- ・ プログラミング以外への適用
 - モデルベースソフトウェア開発
- ・ 生成AIによって解決された点と解決されていない点
- ・ おわりに

2

生成AI時代のソフトウェア工学 (深澤良彰, SES2023 基調講演)



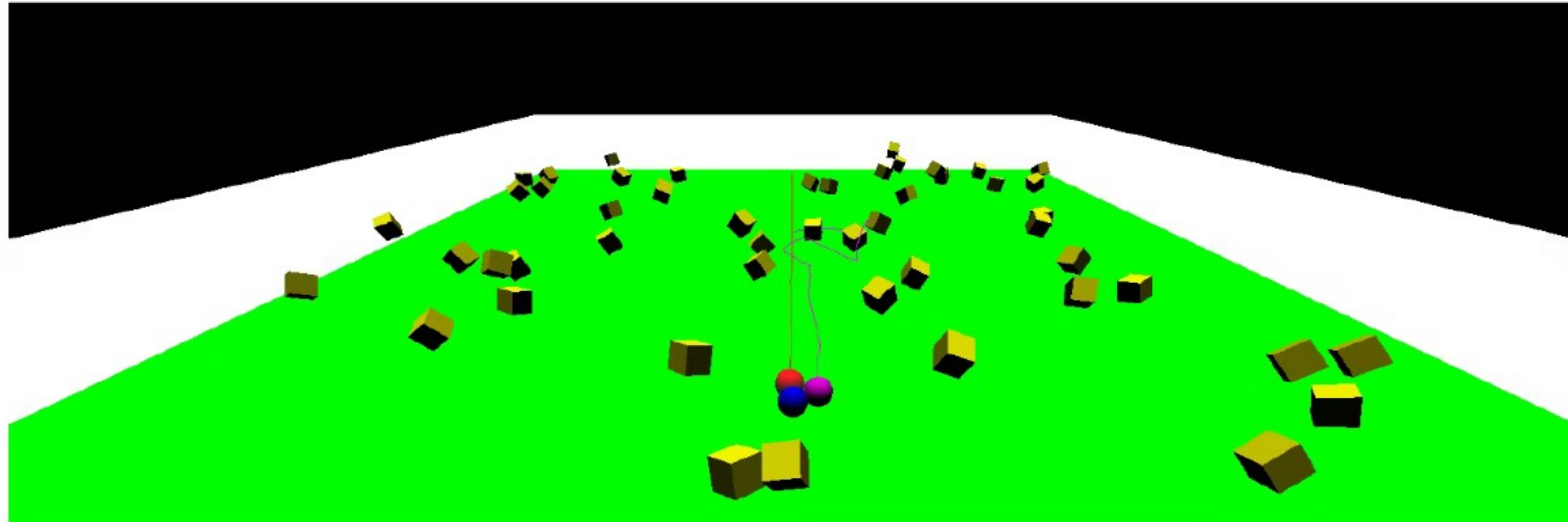
Basic Skill	α (std err)	β (std err)	ζ (std err)
<i>All skill importance scores are normalized to be between 0 and 1.</i>			
Constant	0.082*** (0.011)	-0.112*** (0.011)	0.300*** (0.057)
Active Listening	0.128** (0.047)	0.214*** (0.043)	0.449*** (0.027)
Mathematics	-0.127*** (0.026)	0.161*** (0.021)	0.787*** (0.049)
Reading Comprehension	0.153*** (0.041)	0.470*** (0.037)	-0.346*** (0.017)
Science	-0.114*** (0.014)	-0.230*** (0.012)	-0.346*** (0.017)
Speaking	-0.028 (0.039)	0.133*** (0.033)	0.294*** (0.042)
Writing	0.368*** (0.042)	0.467*** (0.037)	0.566*** (0.047)
Active Learning	-0.157*** (0.027)	-0.065** (0.024)	0.028 (0.032)
Critical Thinking	-0.264*** (0.036)	-0.196*** (0.033)	-0.129** (0.042)
Learning Strategies	-0.072* (0.028)	-0.209*** (0.025)	-0.346*** (0.034)
Monitoring	-0.067** (0.023)	-0.149*** (0.020)	-0.232*** (0.026)
Programming	0.637*** (0.030)	0.623*** (0.022)	0.609*** (0.024)

GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models (arXiv:2303.10130)

Prompt:

Can you write a 3D game in HTML with Javascript, I want:

- There are three avatars, each is a sphere.
- The player controls its avatar using arrow keys to move.
- The enemy avatar is trying to catch the player.
- The defender avatar is trying to block the enemy.
- There are also random obstacles as cubes spawned randomly at the beginning and moving randomly. The avatars cannot cross those cubes.
- The player moves on a 2D plane surrounded by walls that he cannot cross. The wall should cover the boundary of the entire plane.
- Add physics to the environment using cannon.
- If the enemy catches the player, the game is over.
- Plot the trajectories of all the three avatars.



Episode 1: the player avatar (blue) stands still. The enemy (red) rushes straight towards the player, while the defender (magenta) attempts to 'body block' the enemy (see its curved trajectory in magenta).

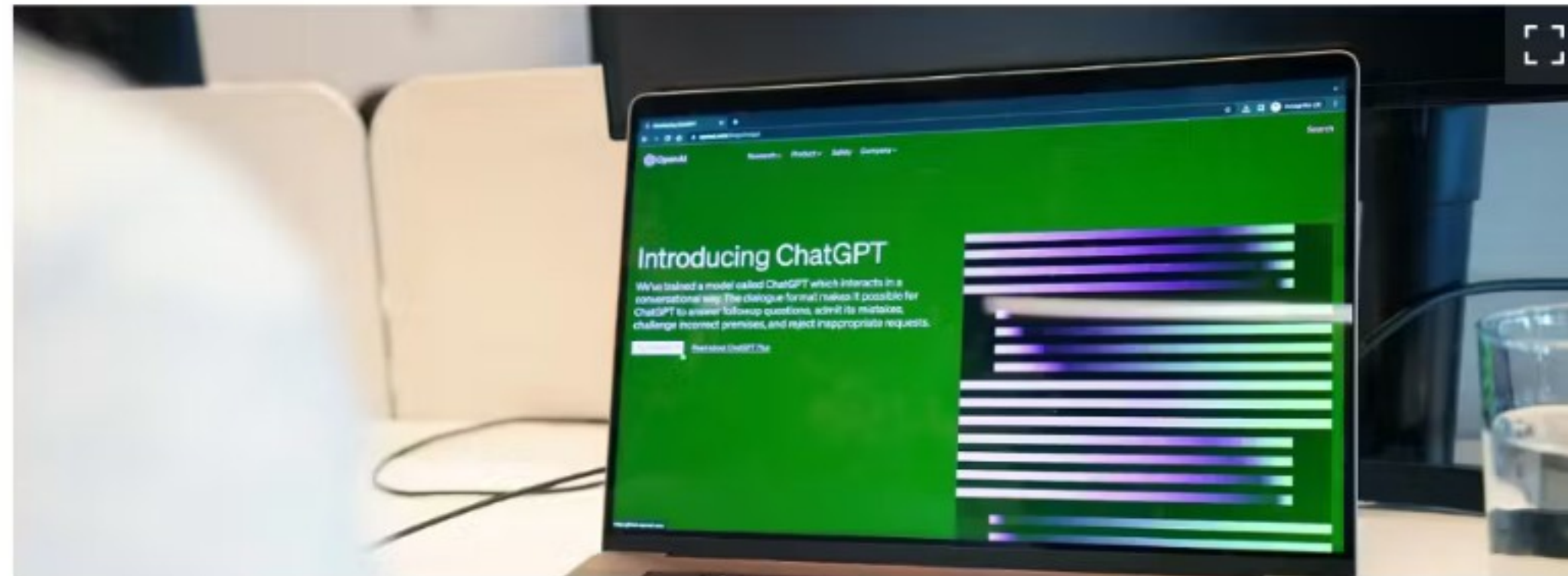
Sparks of Artificial General Intelligence: Early experiments with GPT-4 (arXiv:2303.12712v5)

ChatGPT answers more than half of software engineering questions incorrectly

You may want to stick to Stack Overflow for your software engineering assistance.



Written by Sabrina Ortiz, Editor on Aug. 9, 2023



Who Answers It Better? An In-Depth Analysis of ChatGPT and Stack Overflow Answers to Software Engineering Questions (arXiv:2308.02312, ZDNet 8月9日)



ポジション表明への質問や感想をお寄せください（石川氏、倉光氏、丸山氏、三浦氏）

44 Responses

ソフトウェアを作らないソリューション工学

リッチなモデリングのハードルを下げる

スピードが本当にキツイので、技術を追いかけるのにLLMを使うのが一番良いのではないかと思いますね

速すぎてしんどいというのはわかります。もうついていきません。

ちなみに今回の資料は公開いただけますか？

技術者や研究者としてどう向き合っていくか。活動の在り方。

成長や変動が早い生成AIの文脈で3年以上のプロジェクトを組む難しさや動き出しの難しさを感じた（石川さん）

KOGIが可愛いです

プロジェクト申請もそうですし、卒論修論のネタとしてもこわいです（春にネタふって夏に先を越されて...？

ポジション表明への質問や感想をお寄せください（石川氏、倉光氏、丸山氏、三浦氏）

44 Responses

研究していても、あっという間にChatGPTに飲まれてしまう

KOGI素晴らしいと思います。この授業では成績評価はどのようにしておられるのでしょうか？

プログラミングスキルは不要になるか？

コードを書けなくて良くて、より深く読める必要はあると思います。

AIで人の能力を伸ばす教育

機械学習工学研究会の設立

再利用を劇的に変えるという点に共感します。いままでは再利用のための事前準備を頑張ってきたのだと思いますが、それが不要になる、つまり、その場でできるようになると、大きく世界が変わると思います。

倉光さんKOGI(プログラミング学習支援システム)はGPTに質問を投げて返ってきた答え(プログラムに対する指摘など)をどのように整形されているのでしょうか？デモ動画を見る限り、かなり簡潔にまとめられているように見受けられたのでお聞きしたいです。また、自身の研究室で似たようなシステムを製作中なのでお話ししたいです。

再利用が劇的に効率化する中で、再利用でない部分はなにか

ポジション表明への質問や感想をお寄せください（石川氏、倉光氏、丸山氏、三浦氏）

44 Responses

不確定性のあるAIで高信頼なソフトウェアを作る難しさ

人月の神話が成立する（納期に間に合わなくなりそうになったらAIをお金を払って追加すると、追加した分だけ(人と違って)速度が上がる）未来が来るのではという話があがっていたように思います。 > AI for SE

生成AIへの指示によるプログラミングが主流になるのであれば、高級プログラミング言語が持つ「人間にとっての理解しやすさ、扱いやすさ」の価値が低減するのでしょうか。生成AIの存在を前提にした場合、選択されるプログラミング言語や開発環境はどのように変化していくと思われますか？

現状のLLMの評価は答えがある問題に対して正誤を問うているように思います。しかしLLMは正解がある問題は得意分野ではなく、正解がない問題のほうが得意だと思います。要求工学に効くか、コンサルができるかは、答えのない問題への評価をどう考えるか次第で、私個人はむしろLLMの得意技とってしまいます。どうせ人間も間違えますし。

なぜ人間なら信じるのかが謎ですね

教育: CSが見につかないジレンマ

AI for SE: 現状プロセスでフィットするツールとして用いる

SE for AI: 成果物の検証、ハルシネーション、信用・責任の難しさ

ワクチン接種予約サイトを考えるとコロナが無い頃なら接種証明は不要で、コロナに対しては必須の項目になりました。つまり、要求は常に新しい形態に変化する可能性があるため、そこを解決しなければいけません。よって現在取れる方法はプロンプトで頑張るのか、モデル改善を頑張るのかの2パターンに大別できます。ベースを研究するか、使用法を研究するかの2パターンがしばらく続くと思います。

ポジション表明への質問や感想をお寄せください（石川氏、倉光氏、丸山氏、三浦氏）

44 Responses

本当にスピードの速い分野だと思います。

KOGIをうちの大学でも使ってみたいですね

UMLのようなモデリングは今後不要になる？

SW開発＝プログラミング＋保守進化、SWエンジニアリング＝開発＋大規模化、という見方もあるかと思います。どのあたりまで生成AIに支援してもらえそうでしょうか。

スピード感もそうですが、LLMを利用した技術開発をチャレンジした後に、強烈な性能のモデル(自分の取り組みを越えてしまう)が公開されないかと思うと、恐ろしい

ありがとうございます。昔ながらの課題だと簡単に解かれてしまうので、課題の難度をあげていくのは良いと思うのですが、そうすると、生成AIをうまく使えるかどうか（プログラミングスキルではなく）評価対象になってしまうのではと思いました。

再利用という視点で見たことがなかったので新鮮でした

プログラミング言語、ライブラリ、フレームワークやソフトウェアが動く環境も変化していく中で、ホントにプログラミングが不要になる？

大規模で高信頼性が要求されるソフトは、生成AIが、いくら「これだけテスト、検証しました。証拠はこれです」と言っても信頼できるのか。

ポジション表明への質問や感想をお寄せください（石川氏、倉光氏、丸山氏、三浦氏）

44 Responses

学生対応で手一杯なので、KOGIのようなプログラミング初学者向けのサポートはとても欲しいです。プログラミング学習の評価については、フローチャート/UML作図などの構造を示させたり、ソースコードの意味を書き下させたり、というのはあるかと思います。（採点が大変ですが）

コンパイラの最適化が失敗したとして今の人間が責任持ちますかね？

AIが人間を納得させるためにUMLなどの現状ある手法が必要になってくるのかも知れませんね

LLMデータの汚染に関する対応はどうするのでしょうか。StackOverflowでも脆弱性のあるコードがばらまかれています。自動生成に利用されるコードが汚染されると問題になりませんか。悪意のある人による汚染もあると思います。

最終的に人が確認・納得しやすい形のプログラミング言語が残る、という視点では、今後、検証しやすい新しい言語を作っていく方向はあり得るでしょうか？

生成AI時代の学生の教育として、どういう側面をもっと伸ばしていくべきでしょうか？

Code LLM用の圧縮言語というのを聞いて、かなり前の中間言語を連想しました

抽象度の高い言語なら、入力トークン数の問題はある程度解決できそうですね

LLMによりSE技術・活動の何が変革・進展しつつありますか？

22 Responses

日本語と英語の壁がさらに薄くなっていくと様々な恩恵がありそうですね

自然言語で表現できる仕事（例：企画，要件定義，設計）は，今後，どんどんLLM活用が進んでゆくように思います。

人月の神話が成立する（AIを増やせば増やすだけ生産性があがる）未来が来るのではと期待しています

技術的な変化は小さくとも、開発できる人が変わることは、業界構造を大きく変えるため、企業の一部は戦々恐々です。が、温度感が人によって違いますね

ある程度の離散構造は Mermaid.js 形式のテキストで出せるようになってきているので、モデリング全般が変わってきそうです。

コーディングの高速化・短時間化が進むので、パソコンの外側の活動により重点を置くようになりそうです

ボカロは歌手を無くしたわけではないですが、音楽業界を変えてしまったような変化があるのだと思います。

LLMを前提とした領域（ローコード開発など）では人の寄与よりLLMの寄与が大きくなるためガバナンスが重要。LLMを個人の使いこなしや作業支援とする領域では、既存のガバナンスモデル・プロセスと同様になりつつも、ガバナンスする側をLLM使って何かできないかという話ができる

LLMを前提とした領域（ローコード開発など）では人の寄与よりLLMの寄与が大きくなるためガバナンスが重要。LLMを個人の使いこなしや作業支援とする領域では、既存のガバナンスモデル・プロセスと同様になりつつも、ガバナンスする側をLLM使って何かできないかという話ができる

LLMによりSE技術・活動の何が変革・進展しつつありますか？

22 Responses

Formal Specification/Verificationの工程を人の代わりに代替・自動化してくれるかもしれませんが、AIにお任せしてしまう副作用として脆弱性のような致命的なバグが紛れ込んで（埋め込まれて）しまう懸念もあるような気がします。

まだプログラミングやテストの部分が中心ですが、ソフトウェア・コンピューターに詳しくないユーザー/施主にも使ってもらえるならば開発側にちょっと近づいてもらえるのでは、という期待があります。

新しい開発プロセス

人月商売(私は嫌い)に対して、LLMで変わるという願いがあります。

新しい開発プロセス

ソフトウェア開発者のコスト（人月単価）とLLMを比較すると圧倒的にLLMが低コストのため、SE活動（アプリ開発など）が低価格化すると期待しています

LLMの文字列を扱うという性質上、そちらに適用可能な範囲で技術が発展し、開発形態もそっちに動くと思います（>slack上のごちゃごちゃ議論をAIがまとめる）

開発者の質が平均化する（または下限が上がる）ので、将来のSE人材は今より少なくなるかもしれません

SE従事者の評価の仕組みを変えていかないと自己利益のためにハックされるだけになるのでは、という懸念もあります

LLMによりSE技術・活動の何が変革・進展しつつありますか？

22 Responses

LLMにより比較的容易な作業が効率化されたため、低スキルのエンジニアが担当する業務が薄くなり、教育が難しくなりつつある

LLMを使いこなすソフトウェア技術者と、そうでないソフトウェア技術者の間で、パフォーマンスの差が大きくなるかもしれません。

AIが人間を納得させるためにUMLなど現状使われている手法活用するということはあるかも知れませんね

LLMをどの程度のレベルの開発者とみなすかによって違うと思いますが、新人が作成したものを正しく評価できないといけないという意味では今も一緒ではないかなと思いました。

LLMのSE技術・活動の進展における留意点や研究の展望は？

10 Responses

LLMが生成したコードをLLMが学習することで、いきづまりは起こらないのでしょうか？

ChatGPTなど誰でも無料で使える形のLLMがある日突然公開停止したり使用料が高額になった場合はどうすれば...

NLPとSEの2分野を見るので、調査コストが大きくなりますね

現状はLLMを用いていない研究のうち、LLMの組み込みが可能な研究は今後淘汰されますか？アイデア次第でしょうか

AIが生成したコードの著作権の問題ですが、どんどんコードは生成されるので膨大なチェックが難しく通常の著作権と同じ親告罪になりそうで、今後プログラムの著作権の価値も薄れる気がします

ICSEなどでの研究・論文は機械学習技術が半分近い状況ですが、日本でのSE研究

ここまでソフトウェア開発の実践や教育の議論がありましたが、SE研究への影響をどう捉えられているかお聞きしたいです。いままで地道に積み上げてきた研究がLLMで簡単にできてしまうため、研究としても危機感があるのではないかと思います。

LLM関連の研究のスピード感の違いから産学連携の形が変わってしまうことは考えられないでしょうか？

LLMのSE技術・活動の進展における留意点や研究の展望は？

10 Responses

半年ごと、毎年というレベルでLLMの性能が上がっていくので、SE技術・活動での評価の考え方（どの時点の何のタスクで比較するのか）が必要だと思います。

LLMベースのシステム・サービスに求められるこれからのソフトウェア工学とは？

6 Responses

LLMを活用することで価値のあるソフトウェアやITサービスを創り出すことが重要ではないかと思います。ソフトウェア工学としては、そうした取り組みを支援できれば望ましいと思います。

細かな技術の積み上げだった學問が、細かな作業はLLMで置き換えられることによって、今度こそ実学になるのかもしれない

LLMという「部下」をうまく使う「上司」を育成するための、学問としてのマネジメントスキルがより求められていくのかなと思います

教育の観点で、初級プログラミングの範囲について自動生成である程度できるようになると、より要求・設計の知識や技術が求められると思うので、単純な初学者向けプログラミングの勉強時間をソフトウェア工学の時間に当てられるように、学習教材の提供も重要になってくると思います。

今後、LLMベースのシステムを前提とした業務システム、社会に変わっていく可能性があります。ソフトウェア工学には、LLMを活用した新しい産業の創生や基盤技術へと進化することを期待します。

LLMによってEUCの担い手が今後増えるので、そのような人たち向けのソフトウェア工学を提示してほしい

LLMとソフトウェア工学について
質問やコメントをお寄せください