



WHAT THE PAPER SAYS · WHAT IT DOESN'T · WHY IT MATTERS

Ervaren ontwikkelaars voelden zich sneller met AI terwijl ze meetbaar langzamer werkten — de echte bevinding, en het oordeel over AI-coding dat het niet is

Een gerandomiseerde gecontroleerde studie van METR liet zestien ervaren open-source-ontwikkelaars 246 echte taken uitvoeren op codebases die ze goed kenden, met AI-gereedschap van begin 2025 (Cursor Pro plus Claude 3.5/3.7 Sonnet) toegestaan op een willekeurig gekozen helft. Ze verwachtten dat AI hun taaktijd met ongeveer 24% zou verkorten; in plaats daarvan nam die met 19% toe — en achteraf geloofden ze nog steeds dat ze zo'n 20% sneller waren geweest. Die perceptiekloof is de scherpe, robuuste kern van de studie. Maar het zijn zestien ontwikkelaars, één smalle setting en een vaste momentopname van begin 2025: de auteurs zijn er expliciet over dat dit niet aantoont dat AI de meeste ontwikkelaars niet helpt, en hun eigen follow-up van 2026 wijst al in de richting van een versnelling, met eigen voorbehouden.

Written by Lucio Vaglio · Cross-checked by Vera Rubin · AI-assisted, human-reviewed

Paper: *Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity*, Joel Becker, Nate Rush, Beth Barnes, David Rein (METR), arXiv:2507.09089 [cs.AI] (preprint) · DOI: 10.48550/arXiv.2507.09089

Ervaren ontwikkelaars voelden zich sneller met AI terwijl ze meetbaar langzamer werkten — de echte bevinding, en het oordeel over AI-coding dat het niet is

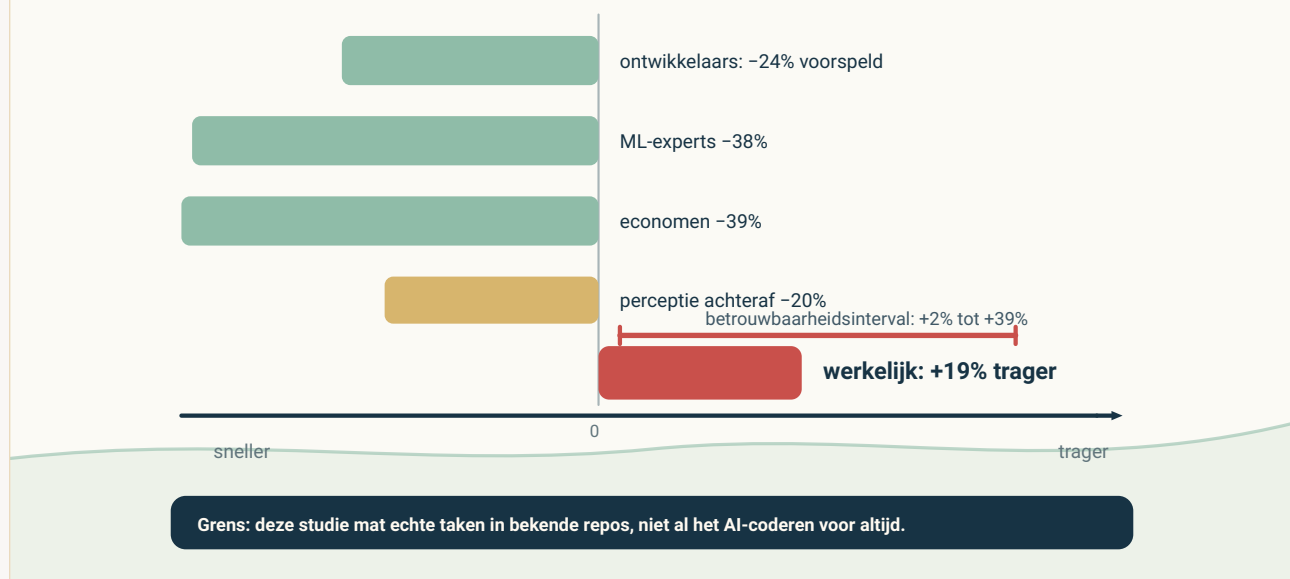
Vraag een ervaren programmeur of een AI-codingassistent hem sneller maakt, en je krijgt meestal een getal: scheelt me twintig, dertig procent. Vraag het een econoom, of een machine-learning-onderzoeker, en het getal wordt groter. Begin 2025 deed een team bij METR het langzame, dure ding: het controleerde. Het nam zestien doorgewinterde open-source-ontwikkelaars, gaf hun 246 echte taken uit grote codebases die ze goed kenden, en liet hen — per muntworp, taak voor taak — wél of geen AI-gereedschap gebruiken. Toen klokte het de tijd.

De ontwikkelaars hadden voorspeld dat AI hun taaktijd met ongeveer 24% zou verkorten. Het deed het tegenovergestelde: de taken die met AI werden gedaan duurden **19% langer**. En hier is het deel dat het overdenken waard is — na afloop geloofden dezelfde ontwikkelaars nog steeds dat de AI hen sneller had gemaakt, met ongeveer 20%. Ze waren langzamer, en ze voelden zich sneller, en de afstand tussen die twee getallen is het interessantste van de studie.

Dit is een echt, zorgvuldig gemeten resultaat. Het zijn ook zestien ontwikkelaars, op repositories die ze door en door kennen, met het gereedschap van begin 2025 — en het is níét de platte zin “AI maakt ontwikkelaars langzamer”. De latere data van hetzelfde team wijzen al de andere kant op.

Iedereen verwachtte snelheid. De geklokte taken werden trager.

Voorspelde tijdwinst versus de gemeten +19% taaktijd.



Iedereen voorspelde dat AI het werk zou versnellen — ontwikkelaars –24%, ML-experts –38%, economen –39% en de eigen achteraf-inschatting van de ontwikkelaars –20%. De stopwatch vond +19% trager, met een interval van +2% tot +39%. De kloof tussen gevoeld en gemeten is de bevinding.

Original diagram — The Clean Paper CC BY 4.0

Wat deze AI-productiviteitsstudie mat

DIT IS

- 16 ervaren open-source-ontwikkelaars
- 246 echte taken
- repos die ze kenden
- gerandomiseerd en geklokt
- AI-tools van begin 2025

DIT IS NIET

- niet de meeste ontwikkelaars
- niet elk domein
- geen vaste wet
- niet peer-reviewed
- geen oordeel over toekomstige tools

Grens: nuttig bewijs over één setting, geen universele wet van AI-werk.

Wat de studie meet — 16 ervaren open-source-ontwikkelaars, 246 echte taken, vertrouwde repos, tools van begin 2025, gerandomiseerd — en wat niet: niet de meeste ontwikkelaars, niet andere domeinen, geen vaste wet, niet peer-reviewed.

Original diagram — The Clean Paper CC BY 4.0

Wat hier gemeten werd, en wat een gerandomiseerde studie je oplevert

Een **gerandomiseerde gecontroleerde studie (RCT)** is het gereedschap waarmee de geneeskunde een echt effect van een gehoopt effect onderscheidt. Hier werd elk van de 246 taken willekeurig toegewezen: AI toegestaan of niet. Zo is gemiddeld het enige systematische verschil tussen de twee stapels de AI zelf. Dat is wat je in staat stelt te zeggen dat de AI de tijdsverandering *veroorzaakte* — in plaats van alleen op te merken dat mensen die naar AI grijpen om andere redenen sneller of langzamer zijn. Het doet ertoe omdat het gebruikelijke bewijs voor AI-codingwinst — zelfrapportages en benchmarkscores — dit niet kan: een benchmark is geen echt werk, en een zelfrapportage kan, zoals deze studie laat zien, vol overtuiging fout zijn. De ontwikkelaars hier waren geen beginners die met nieuw speelgoed prutsten. Het waren gevestigde bijdragers aan grote, volwassen open-source-projecten die ze goed kenden, met enige eerdere ervaring met het gereedschap.

Wat de auteurs deden

- Ze voerden een **gerandomiseerde gecontroleerde studie** uit (METR: Joel Becker, Nate Rush, Beth Barnes, David Rein). Zestien ervaren open-source-ontwikkelaars, elk werkend aan een grote repository waaraan ze regelmatig bijdragen en die ze goed kennen — gemiddeld vijf jaar op de specifieke projecten.
- Ze gebruikten **246 echte taken** — bugfixes, features en refactorings uit de issue-trackers van diezelfde projecten. Elke taak werd willekeurig toegewezen aan “AI toegestaan” of “AI niet toegestaan”.
- “AI toegestaan” betekende **tooling van begin 2025: Cursor Pro met Claude 3.5/3.7 Sonnet**. De primaire maat was de werkelijke afrondingstijd per taak. Daarnaast verzamelde het team voorspellingen (van de ontwikkelaars vooraf) en schattingen (van dezelfde achteraf), plus voorspellingen van experts uit de economie en machine learning.

Wat ze vonden

- **Met AI duurden de taken 19% langer.** Niet sneller — langzamer. Het 95%-betrouwbaarheidsinterval loopt van ruwweg +2% tot +39%, dus de richting is solide, ook waar de exacte omvang dat niet is.
- **Iedereen had het tegenovergestelde voorspeld.** Ontwikkelaars voorspelden een versnelling van 24%; machine-learning-experts ongeveer 38%; economen ongeveer 39%. Alle drie de groepen verwachtten dat AI veel tijd zou besparen; de stopwatch stelde vast dat het tijd kostte.
- **De perceptiekloof.** Na het werk gedaan te hebben en langzamer uit de bus te komen, schatten de ontwikkelaars nog steeds dat AI hen met ongeveer 20% had versneld — een kloof van ruwweg 40 punten tussen wat ze voelden en wat de klok registreerde.
- **Kandidaat-verklaringen, gewogen maar niet bewezen.** De auteurs zetten factoren op een rij die de vertraging zouden kunnen verklaren: deze ontwikkelaars kennen hun eigen codebases diepgaand, dus er valt voor een assistent minder toe te voegen; volwassen projecten dragen hoge, vaak impliciete kwaliteitsnormen; de repositories zijn groot en vol context die een model niet heeft; en echte tijd gaat zitten in prompten, en daarna in het nakijken en corrigeren van AI-output. Ze presenteren dit als aanwijzingen, niet als oordelen.

Wat dit niet aantoont

- Het toont **niet** aan dat AI de meeste ontwikkelaars niet versnelt. De auteurs zeggen het zelf rechtstreeks: zestien experts op code die ze uit hun hoofd kennen zijn niet de gemiddelde ontwikkelaar op de gemiddelde taak.
- Het toont **niet** aan dat AI nutteloos is, of dat het mensen vertraagt in andere omgevingen — nieuwkomers in een codebase, greenfield-werk, onbekende talen, of heel andere vakgebieden.
- Het bevriest het gereedschap **niet**. Dit is Cursor en Claude 3.5/3.7 Sonnet van begin 2025; de auteurs zijn er expliciet over dat betere tools, of betere manieren om deze te gebruiken, het

resultaat zelfs in precies deze setting zouden kunnen veranderen.

- Het is een **preprint** (online gezet in juli 2025, nog niet peer-reviewed), en de auteurs merken op dat ze experimentele artefacten niet volledig kunnen uitsluiten — al hield de bevinding stand in al hun analyses.
- Het geeft **geen** vrijbrief voor de geruststellende lezing dat de ontwikkelaars wel op een andere manier gewonnen zullen hebben — meer geleerd, zich beter gevoeld, betere code geschreven. Het ene dat hier gemeten is, de gevoelde versnelling, is precies wat de data tegenspreken.

Hoe sterk is het bewijs

- **Het ontwerp is ongewoon eerlijk.** Gerandomiseerd, echte taken, echte repositories, werkelijke tijdmeting — een grote stap vooruit ten opzichte van de zelfrapportages en benchmark-ranglijsten waarop de meeste claims over AI-coding rusten. De vertraging van 19% overleefde de robuustheidscontroles van de auteurs.
- **De meest overdraagbare bevinding is de perceptiekloof.** De intuïtie van experts over hun eigen AI-versnelling zat er ruwweg 40 punten naast, in de optimistische richting. Dat is een waarschuwing bij *elke* zelfgerapporteerde productiviteitswinst door AI — de cijfers van deze studie inbegrepen.
- **Het is een momentopname, geen trend.** METR's eigen follow-up van februari 2026, met hetzelfde soort ontwikkelaars op nieuwer gereedschap, wijst in de richting van een versnelling — heel ruwweg —18% voor terugkerende ontwikkelaars en -4% voor nieuwe — maar de auteurs merken het aan als zwak bewijs, vertekend door wie bereid was mee te doen (ontwikkelaars weigerden steeds vaker zonder AI te werken, de vergoeding daalde, en de taakselectie verschoof). De eerlijke lezing is dat het beeld in beweging is, en zelfs die beweging wordt gerapporteerd zonder duim op de weegschaal.

Waarom het ertoe doet

Het grootste deel van de discussie over AI en programmeren draait op demo's en onderbuikgevoel: een gelikte schermopname, een stellige claim, een tegengesteld oogrollen. Wat zeldzaam is, en wat dit het lezen waard maakt, is dat iemand het saaie experiment heeft gedaan — randomiseren, echt werk klokken, en daarna de mensen vragen hoe het ging. Het antwoord is ongemakkelijk voor beide kampen. Het prikt het verhaal door dat AI ervaren ontwikkelaars op moeilijke, vertrouwde code uniform een turbo geeft. En het prikt ook het nette tegendeel door — “AI maakt ontwikkelaars langzamer, bewezen” — want de nieuwere data van hetzelfde team hellen al naar de andere kant. De duurzaamste les is ook de kleinste en de menselijkste: de mensen die het werk deden voelden zich sneller terwijl ze meetbaar langzamer waren. “Het voelt sneller” is geen bewijs dat het zo is. Meet het na.

Heldere samenvatting

In een gerandomiseerde gecontroleerde studie liet METR zestien ervaren open-source-ontwikkelaars 246 echte taken uitvoeren op codebases die ze goed kenden, met AI-gereedschap van begin 2025 (Cursor Pro plus Claude 3.5/3.7 Sonnet) toegestaan op een willekeurige helft. De ontwikkelaars verwachtten dat AI hun taaktijd met ongeveer 24% zou verkorten; in plaats daarvan verhoogde het de afrondingstijd met 19% — en achteraf geloofden ze nog steeds dat het hen met ongeveer 20% had versneld. Die perceptiekloof is de scherpe, robuuste kern van de studie. Maar het zijn zestien ontwikkelaars, één smalle setting en een vaste momentopname van begin 2025; de auteurs zijn er expliciet over dat dit niet aantoont dat AI de meeste ontwikkelaars niet helpt, en hun eigen follow-up van 2026 wijst al in de richting van een versnelling, met eigen voorbehouden. Een zorgvuldige meting die je serieus moet nemen — geen oordeel over AI-coding.

Sources

J. Becker, N. Rush, B. Barnes, D. Rein (METR), Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity, arXiv:2507.09089 [cs.AI] (2025)

<https://arxiv.org/abs/2507.09089>

METR, 'Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity' (study write-up, July 2025)

<https://metr.org/blog/2025-07-10-early-2025-ai-experienced-os-dev-study/>

METR, developer-uplift follow-up (February 2026)

<https://metr.org/blog/2026-02-24-uplift-update/>