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

Փորձառու ծրագրավորողները AI-ով իրենց ավելի արագ էին զգում, մինչդեռ չափելիորեն ավելի դանդաղ էին աշխատում

METR-ի պատահականացված վերահսկվող փորձարկմանը բաց կոդով նախագծերի 16 փորձառու մշակող կատարել է 246 իրական առաջադրանք իրենց լավ ծանոթ կոդային բազաներում: Նրանք սպասում էին, որ 2025 թվականի սկզբի AI գործիքները մոտ 24%-ով կկրճատեն աշխատանքի ժամանակը, բայց իրականում առաջադրանքները 19%-ով ավելի երկար տևեցին, իսկ հետո մասնակիցները դեռ կարծում էին, որ մոտ 20%-ով արագացել են: Ընկալման այս տարբերությունն է հետազոտության ամենակայուն արդյունքը: Այն չի նշանակում, որ AI-ը դանդաղեցնում է ծրագրավորողների մեծ մասին, և թիմի 2026 թվականի follow-up-ն արդեն հակառակ ուղղությամբ է հուշում՝ սեփական սահմանափակումներով:

Written by Lucio Vaglio · Cross-checked by Vera Rubin · Edited by Michele Renda · Figures by Nova Vela · 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

Փորձառու ծրագրավորողները ԱԲով իրենց ավելի արագ էին զգում, մինչդեռ չափելիորեն ավելի դանդաղ էին աշխատում. սա է իրական արդյունքը, ոչ թե վերջնական դատավճիռ ԱԲով կոդավորման մասին

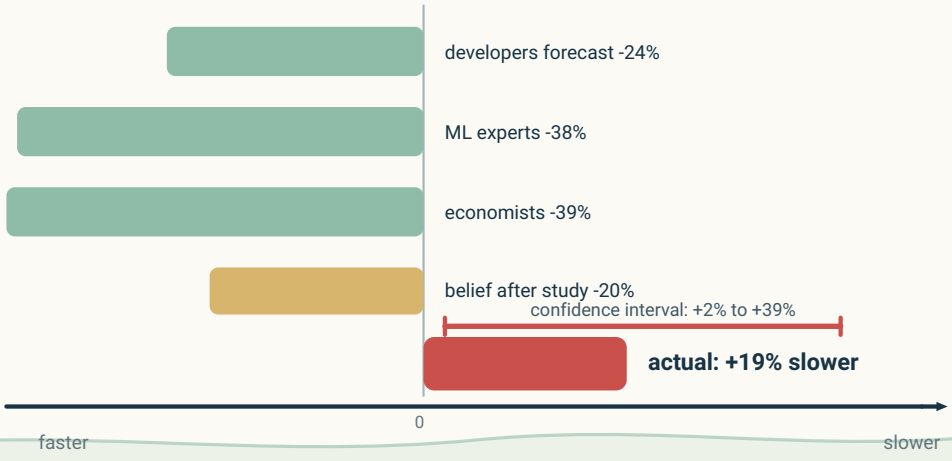
Եթե փորձառու ծրագրավորողին հարցնեք՝ ԱԲ կոդավորման օգնականը որքանո՞վ է արագացնում իր աշխատանքը, հաճախ կլսեք որևէ թիվ՝ «քսան, երեսուն տոկոս»: Տնտեսագետին կամ մեքենայական ուսուցման հետազոտողին հարցնելու դեպքում թիվը սովորաբար ավելի մեծ է դառնում: 2025 թվականի սկզբին METR-ի թիմը արեց դանդաղ ու թանկարժեք բանը՝ փորձեց դա իրականում չափել: Նրանք ընտրեցին բաց կոդով նախագծերի տասնվեց փորձառու մշակողի, տվեցին նրանց իրենց լավ ծանոթ մեծ կոդային բազաներից վերցված 246 իրական առաջադրանք և յուրաքանչյուր առաջադրանքի համար պատահականորեն որոշեցին՝ թույլատրե՞լ ԱԲ գործիքների օգտագործումը, թե՞ ոչ: Հետո պարզապես չափեցին աշխատանքի ժամանակը:

Մշակողները կանխատեսել էին, որ ԱԲը առաջադրանքների կատարման ժամանակը կնվազեցնի մոտ 24%-ով: Ստացվեց հակառակը. ԱԲով կատարված առաջադրանքները պահանջեցին **19% ավելի շատ ժամանակ**: Բայց ամենահետաքրքիր մասը սա է. աշխատանքն ավարտելուց հետո նույն մշակողները դեռ կարծում էին, որ ԱԲը իրենց մոտ 20%-ով արագացրել է: Նրանք իրականում ավելի դանդաղ էին, բայց իրենց ավելի արագ էին զգում, և այս երկու թվերի միջև տարբերությունն է հետազոտության ամենաուշագրավ արդյունքը:

Սա իրական, զգուշորեն չափված արդյունք է: Բայց այն վերաբերում է ընդամենը տասնվեց մշակողի, որոնք աշխատում էին իրենց շատ լավ ճանաչած repository-ների վրա՝ 2025 թվականի սկզբի գործիքներով: Այն չի նշանակում պարզունակ «ԱԲը ծրագրավորողներին դանդաղեցնում է» եզրակացություն: Նույն թիմի ավելի ուշ տվյալներն արդեն հակառակ ուղղությամբ են հուշում:

Everyone expected speed. The timed tasks got slower.

Forecasted time savings versus the measured +19% task time.



Boundary: this study measured real tasks in known repos, not all AI coding forever.

Բոլորը կանխատեսել էին, որ ԱԲը կարագացնի աշխատանքը՝ մշակողները՝ −24%, մեքենայական ուսուցման փորձագետները՝ −38%, տնտեսագետները՝ −39%, իսկ ուսումնասիրությունից հետո մշակողների սեփական գնահատականը՝ −20%: Ժամանակաչափը ցույց տվեց հակառակը՝ +19% դանդաղում, +2%-ից +39% միջակայքով: Զգացածի և չափվածի միջև այս տարբերությունն է հիմնական արդյունքը:

Original diagram — The Clean Paper CC BY 4.0

What this AI-productivity study measured

THIS IS

- 16 experienced open-source developers
- 246 real tasks
- repositories they knew
- randomized and timed
- early-2025 AI tools

THIS IS NOT

- not most developers
- not every domain
- not a fixed law
- not peer-reviewed
- not a verdict on future tools

Boundary: useful evidence about one setting, not a universal law of AI work.

Ինչն է չափել հետազոտությունը՝ բաց կոդով նախագծերի 16 փորձառու մշակող, 246 իրական առաջադրանք, իրենց լավ ծանոթ repository-ներ, 2025 թվականի սկզբի գործիքներ և պատահականացված դիզայն: Եվ ինչ չի չափել՝ ոչ ծրագրավորողների մեծ մասը, ոչ այլ ոլորտները, ոչ մշտական օրենք, ոչ էլ արդեն գրախոսված արդյունք:

Original diagram — The Clean Paper CC BY 4.0

Ինչ էր չափվում, և ինչ է տալիս պատահականացված փորձարկումը

Պատահականացված վերահսկվող փորձարկումը (RCT) այն գործիքներից է, որով բժշկությունը փորձում է իրական ազդեցությունը տարբերել սպասված կամ ցանկալի ազդեցությունից: Այստեղ 246 առաջադրանքներից յուրաքանչյուրը պատահականորեն հատկացվել էր կամ «ԱԲ թույլատրված է», կամ «ԱԲ թույլատրված չէ» պայմանին: Այդ պատճառով, միջին հաշվով, երկու խմբերի միջև համակարգված տարբերությունը հենց ԱԲի առկայությունն էր: Սա թույլ է տալիս ասել, որ ժամանակի փոփոխությունը *պատճառվել է* ԱԲի օգտագործմամբ, այլ ոչ պարզապես նկատվել է, որ ԱԲ օգտագործող մարդիկ ինչոր այլ պատճառով ավելի արագ կամ դանդաղ են: Սա կարևոր է, որովհետև ԱԲով կոդավորման օգուտների մասին սովորական ապացույցները՝ ինքնագնահատականներն ու չափանիշային թեստները, այդպիսի պատճառական եզրակացություն չեն տալիս. չափանիշային թեստը իրական աշխատանք չէ, իսկ ինքնագնահատականը, ինչպես ցույց է տալիս հենց այս հետազոտությունը, կարող է վստահորեն սխալ լինել: Այստեղ մասնակցողները նորեկներ չէին, որոնք առաջին անգամ էին փորձարկում նոր գործիք: Նրանք մեծ, հասուն բացաղբյուր նախագծերի հաստատված contributor-ներ էին և նախկինում էլ որոշ չափով օգտագործել էին նման գործիքներ:

Ինչ արեցին հեղինակները

- Անցկացրին **պատահականացված վերահսկվող փորձարկում** (METR՝ Joel Becker, Nate Rush, Beth Barnes, David Rein): Մասնակցեցին բաց կոդով նախագծերի 16 փորձառու մշակողներ, որոնցից յուրաքանչյուրն աշխատում էր այն մեծ repository-ի վրա, որին սովորաբար ներդրում էր անում և լավ գիտեր: Միջինում նրանք տվյալ նախագծերի վրա աշխատել էին մոտ հինգ տարի:
- Օգտագործեցին **246 իրական առաջադրանք**՝ bug fix-եր, նոր հնարավորություններ և refactor-ներ՝ վերցված հենց այդ նախագծերի issue tracker-ներից: Յուրաքանչյուր առաջադրանք պատահականորեն նշանակվեց «ԱԲ թույլատրված է» կամ «ԱԲ արգելված է» պայմանին:
- «ԱԲ թույլատրված է» նշանակում էր **2025 թվականի սկզբի գործիքներ՝ Cursor Pro և Claude 3.5/3.7 Sonnet**: Հիմնական չափումը յուրաքանչյուր առաջադրանքի իրական ավարտի ժամանակն էր: Թիմը նաև հավաքեց նախնական կանխատեսումներ մշակողներից և վերջնական գնահատականներ աշխատանքից հետո, ինչպես նաև տնտեսագիտության և մեքենայական ուսուցման փորձագետների

կանխատեսումներ:

Ինչ գտան

- **ԱԲով առաջադրանքները 19% ավելի երկար տևեցին**: Ոչ թե արագ, այլ դանդաղ: 95% վստահության միջակայքը մոտավորապես +2%-ից +39% է, այնպես որ ուղղությունը բավական կայուն է, նույնիսկ եթե ազդեցության ճշգրիտ մեծությունը՝ ոչ:
- **Բոլորը հակառակն էին կանխատեսել**: Մշակողները սպասում էին 24% արագացման, մեքենայական ուսուցման փորձագետները՝ մոտ 38%, տնտեսագետները՝ մոտ 39%: Երեք խմբերն էլ ակնկալում էին զգալի ժամանակի խնայողություն, մինչդեռ իրական չափումը ցույց տվեց ժամանակի կորուստ:
- **Ընկալման տարբերությունը**: Ավելի դանդաղ աշխատելուց հետո մշակողները դեռ գնահատում էին, որ ԱԲը իրենց մոտ 20%-ով արագացրել է՝ մոտ 40 տոկոսային կետի տարբերություն զգացածի և չափվածի միջև:
- **Հնարավոր պատճառներ՝ քննարկված, բայց ոչ ապացուցված**: Հեղինակները նշում են մի քանի գործոն, որոնք կարող էին բացատրել դանդաղումը. այս մշակողները իրենց կոդային բազաները շատ խորությամբ գիտեին, ուստի օգնականը քիչ նոր արժեք կարող էր ավելացնել. հասուն նախագծերն ունեն բարձր և հաճախ չգրված որակական պահանջներ. repository-ները մեծ են և լի են այնպիսի համատեքստով, որը մոդելը չունի. և իրական ժամանակ է ծախսվում հրահանգ գրելու, ապա ԱԲի output-ը ստուգելու և ուղղելու վրա: Հեղինակները դրանք ներկայացնում են որպես հավանական ուղղություններ, ոչ որպես վերջնական բացատրություններ:

Ինչ սա չի ցույց տալիս

- Սա **չի** ցույց տալիս, որ ԱԲը չի արագացնում ծրագրավորողների մեծ մասին: Հեղինակներն դա ուղիղ ասում են. իրենց կոդը շատ լավ ճանաչող տասնվեց փորձագետը «միջին ծրագրավորողը՝ միջին առաջադրանքի վրա» չէ:
- Սա **չի** ցույց տալիս, որ ԱԲն անօգուտ է կամ այլ միջավայրերում մարդկանց դանդաղեցնում է՝ օրինակ նոր repository մտնող ծրագրավորողների, զրոյից ստեղծվող նախագծերի, անծանոթ լեզուների կամ այլ ոլորտների դեպքում:
- Սա գործիքների զարգացումը **չի սառեցնում ժամանակի մեջ**: Փորձարկումը վերաբերում է 2025 թվականի սկզբի Cursor-ին ու Claude 3.5/3.7 Sonnet-ին: Հեղինակները հստակ նշում են, որ ավելի լավ գործիքները կամ նույն գործիքների ավելի լավ օգտագործումը կարող են նույնիսկ նույն միջավայրում արդյունքը փոխել:
- Սա **նախատիպագիր** է՝ հրապարակված 2025 թվականի հուլիսին և դեռ չգրախոսված: Հեղինակները նշում են նաև, որ չեն կարող ամբողջովին բացառել փորձարարական artifact-ները, թեև արդյունքը պահպանվել է տարբեր վերլուծությունների դեպքում:

- Արդյունքը **չի** արդարացնում նաև մխիթարական մեկնաբանությունը, թե մշակողները գուցե այլ բան են շահել՝ ավելի շատ սովորել, ավելի երջանիկ զգացել կամ ավելի լավ կոդ գրել։ Այստեղ չափված սուբյեկտիվ օգուտը՝ «ավելի արագ էի», հենց այն բանն է, որ տվյալները հակասում են։

Որքա՞ն ուժեղ է ապացույցը

- **Դիզայնը անսովորորեն համոզիչ է**։ Պատահականացում, իրական առաջադրանքներ, իրական repository-ներ, իրական ժամանակաչափում. սա մեծ քայլ է այն ինքնագնահատակա և չափանիշային թեստ leaderboard-ներից այն կողմ, որոնց վրա հիմնվում են ԱԲ կոդավորման վերաբերյալ շատ պնդումներ։ 19% դանդաղումը պահպանվել է հեղինակների կայունություն check-երի ընթացքում։
- **Ամենահեշտ ընդհանրացվող արդյունքը ընկալման տարբերությունն է**։ Փորձագետների ինտուիցիան սեփական ԱԲով արագացման մասին սխալվեց մոտ 40 տոկոսային կետով՝ լավատեսական ուղղությամբ։ Սա նախազգուշացնում է ԱԲից արտադրողականության *բոլոր* ինքնագնահատականների համար, ներառյալ այս հետազոտության մասնակիցների գնահատականները։
- **Սա պահի լուսանկար է, ոչ միտում**։ METR-ի 2026 թվականի փետրվարյան հետագա դիտարկումը՝ նմանատիպ մշակողների և նոր գործիքների դեպքում, արդեն արագացման ուղղությամբ է հուշում՝ շատ մոտավոր −18% վերադարձած մասնակիցների և −4% նոր մասնակիցների համար։ Բայց հեղինակներն այն համարում են թույլ ապացույց, որովհետև մասնակցության ընտրությունը խեղաթյուրվել էր. մշակողները ավելի հաճախ հրաժարվում էին աշխատել առանց ԱԲի, վճարը նվազել էր, իսկ առաջադրանքների ընտրությունը փոխվել էր։ Զգույշ ընթերցումն այն է, որ պատկերը շարժվում է, բայց նույնիսկ այդ շարժումը դեռ մեծ անորոշությամբ է չափվում։

Ինչու է սա կարևոր

ԱԲի և ծրագրավորման մասին բանավեճերի մեծ մասը կառուցված է demo-ների ու զգացողությունների վրա՝ գեղեցիկ էկրան recording, վստահ պնդում, հակառակ կողմի հեգնական արձագանք։ Այստեղ արժեքավորն այն է, որ ինչոր մեկը արեց ձանձրալի փորձը. պատահականացրեց պայմանները, չափեց իրական աշխատանքը և հետո հարցրեց մարդկանց՝ ինչպես անցավ։ Պատասխանը անհարմար է երկու ճամբարների համար էլ։ Այն հակասում է պատմությանը, թե ԱԲը բոլոր դեպքերում կտրուկ արագացնում է փորձառու ծրագրավորողներին բարդ ու իրենց ծանոթ կոդում։ Բայց այն նաև հակասում է նույնքան պարզ հակառակ պատմությանը՝ «ապացուցված է, որ ԱԲը ծրագրավորողներին դանդաղեցնում է», որովհետև նույն թիմի նոր տվյալներն արդեն մյուս ուղղությամբ են հուշում։ Ամենակայուն դասը փոքր է և շատ մարդկային. մարդիկ իրենց ավելի արագ էին զգում, մինչդեռ ժամացույցը ցույց էր տալիս հակառակը։ «Թվում է՝ ավելի արագ է» փաստարկ չէ։

Դա պետք է չափել։

Կարճ ամփոփում

Պատահականացված վերահսկվող փորձարկման մեջ METR-ը բաց կոդով նախագծերի 16 փորձառու մշակողի հանձնարարեց 246 իրական առաջադրանք իրենց լավ ծանոթ կոդային բազաներից։ Առաջադրանքների պատահական կեսում թույլատրված էր 2025 թվականի սկզբի ԱԲ գործիքների օգտագործումը՝ Cursor Pro և Claude 3.5/3.7 Sonnet։ Մշակողները ակնկալում էին, որ ԱԲը մոտ 24%-ով կկրճատի աշխատանքի ժամանակը, բայց իրականում այն 19%-ով երկարացրեց առաջադրանքների կատարման ժամանակը, իսկ ավարտից հետո մասնակիցները դեռ կարծում էին, որ մոտ 20%-ով արագացել են։ Այս ընկալման տարբերությունն է ուսումնասիրության ամենաուժեղ և կայուն արդյունքը։ Բայց սա ընդամենը տասնվեց մշակող է, մեկ նեղ միջավայր և 2025 թվականի սկզբի գործիքների լուսանկար։ Հեղինակները հստակ ասում են, որ արդյունքը չի ցույց տալիս, թե ԱԲը չի օգնում ծրագրավորողների մեծ մասին, և իրենց 2026 թվականի հետագա դիտարկումն արդեն արագացման ուղղությամբ է հուշում՝ սեփական վերապահումներով։ Սա արժանի է լուրջ վերաբերմունքի որպես զգուշորեն արված չափում, բայց ոչ որպես ԱԲ կոդավորման վերջնական դատավճիռ։

Աղբյուրներ

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/