AI in the Office and the Factory: Evidence from Administrative Software Registry Data

Gustavo de Souza

REVISED
June 2026

WP 2025-11

https://doi.org/10.21033/wp-2025-11

FEDERAL RESERVE BANK *of* CHICAGO

A.I. in the Office and the Factory: Evidence from Administrative Software Registry Data*

Gustavo de Souza†

June 2026

Abstract

What is the effect of artificial intelligence (AI) on employment and wages? Using administrative data on a comprehensive set of AI applications developed in Brazil, I show that AI is widely used not only in administrative tasks but also in the production setting, where it primarily supports process optimization and quality control. Exploiting exogenous variation in AI development costs, I find that AI affects administrative and production workers differently. In office occupations, AI reduces employment, especially among young workers performing data-processing tasks. In production occupations, it increases employment of young and low-skilled workers by helping them perform decision-making tasks that previously required expertise. On net, AI increases employment in Brazil because gains in production outweigh losses in office work.

Key Words: Artificial Intelligence, Automation, Deskilling, Software, Inequality

JEL: J23, J24, F63

* This paper benefited from valuable comments from Ajay Agrawal (*discussant*), Francesco Amodio, Tania Babina, Alessandra Bonfiglioli, Wyatt Brooks, David Autor, Bharat Chandar, Raj Chetty, Oren Danieli, John Friedman, Gino Gancia, Tarek Hassan, Nathaniel Hendren, Anders Humlum, Xavier Jaravel, Joe Kaboski, Aakash Kalyani (*discussant*), David Lagakos, Kristina McElheran, Markus Poschke, Diego Restuccia, Anna Salomons (*discussant*), Yongseok Shin, Henry Siu, and Kjetil Storesletten. I also thank seminar and conference participants at the Opportunity Insights Conference on Economic Mobility, Boston University, McGill University, the Federal Reserve Bank of San Francisco, the Federal Reserve Bank of Chicago, the IMF, LACEA, SBE, MEA, the Artificial Intelligence and the Macroeconomy Conference, the NBER AI and Economic Measurement Conference, Tilburg University, the STEG Annual Conference, T2M, the Rotman School of Management, the AI in Finance Conference, Bocconi University, and SED. Jack Mannion and Vitor Santos provided superb research assistance.

† Federal Reserve Bank of Chicago. Contact: gustavo@microtomacro.net. This paper represents my opinions and not those of the Federal Reserve Bank of Chicago or the Federal Reserve System.

"I like to think (and
the sooner the better!)
of a cybernetic meadow
where mammals and computers
live together in mutually
programming harmony."

—Richard Brautigan,
"All Watched Over by Machines of Loving Grace"

1 Introduction

Artificial intelligence (AI) refers to computer systems that learn from data to recognize patterns, make predictions, or generate information, tasks that once required human labor. AI is used in generative tools, predictive systems, recommendation algorithms, industrial automation, autonomous vehicles, and many other applications (McElheran et al. 2023, 2025, Brynjolfsson et al. 2021). This broad use has raised concern about its effects on the labor market. One view holds that AI automates tasks and displaces workers, especially in routine cognitive jobs (Acemoglu and Restrepo 2018, Webb 2020, Acemoglu et al. 2022). Another view argues that AI augments labor by simplifying tasks and improving decision-making, ultimately creating jobs (Autor 2024, Agrawal et al. 2019b, Bresnahan 2021). There is still no consensus on whether AI increases or reduces employment on net.

A possible reason for these contrasting views is that much of the evidence comes from specific AI uses. Some studies focus on generative AI tools used by professionals (Humlum and Vestergaard 2025b, Brynjolfsson et al. 2025); others study predictive AI developed in-house (McElheran et al. 2023, Brynjolfsson and McElheran 2019, Hampole et al. 2025, Babina et al. 2024, Adams et al. 2024). Still others include adoption through third-party software and off-the-shelf tools, including industrial applications that can raise machine productivity and operational efficiency (McElheran et al. 2025, Alam et al. 2024). Since AI's effects depend on how it is used (Aghion et al. 2025), understanding its overall labor-market impact requires data that cover a broad set of AI applications.

I contribute to this debate by studying how AI affects the labor market using an administrative registry of commercial AI software developed in Brazil. Each registration contains a rich software description provided by the firm and verified by an external examiner. The registry covers a broad set of applications, from early predictive systems to generative AI tools registered by 2024. This setting allows me to contribute by studying AI across many real-world uses and in a middle-income country with a large IT sector.

Using exogenous variation in AI development costs as an instrument, I find that AI affects administrative and production workers in different ways. It reduces employment in administrative occupations, especially among young workers performing data-processing tasks. By contrast, it increases employment among young, inexperienced, and less-educated workers in production occupations. I show that this contrast reflects how occupations use information: AI complements workers who use predictions, recommendations, or guidance to make decisions, such as production workers, but replaces workers whose tasks are to gather and process information, such as administrative workers. Because production gains dominate in Brazil, AI increases total employment and reduces inequality.

These findings help reconcile competing views on AI's labor-market impact. In production, the results align with the idea that AI increases machine productivity, simplifies operation, and raises demand for low-skilled labor (Autor 2024, Bresnahan 2021, Agrawal et al. 2019b). In the office, they are consistent with AI automating tasks and reducing employment (Acemoglu and Restrepo 2018, Webb 2020, Hui et al. 2024). I then test whether this office-factory contrast helps explain why the literature reaches different conclusions. When I repeat the analysis using standard off-the-shelf occupational AI exposure measures, the estimates become negative because these measures overstate AI exposure in the office and understate it in production. The aggregate effect of AI therefore depends on how and where it is used, helping explain why studies focused on particular applications reach different conclusions.

A case from the registry illustrates these findings. Techplus, a Brazilian firm, develops AI systems for industrial operations. Its software analyzes sensor data from factory equipment to predict failures, schedule maintenance, and assist machine operation. By providing real-time guidance to supervisors and machine operators, such systems can reduce unplanned downtime by up to 50% (Agoro 2025, Benhanifia et al. 2025), which raises the value of keeping machines running and increases demand for workers who operate them (Schmitz 2005). At the same time, the software automates administrative tasks such as maintenance planning and inventory control. This case reflects the broader pattern I identify in the data: AI complements production workers by making expertise available, while reducing demand for routine office work.

The analysis begins with the registry data that allow me to observe AI applications and link them to labor-market outcomes. The registry exists because, since 1987, Brazilian firms have been able to register software with the National Institute of Industrial Property (NIIP) to establish legal proof of ownership. This legal protection made registration common: approximately 92% of software development firms in 2016 had at least one registered program.[1]

[1]To assess the coverage of the registry, I also conducted a survey of industry specialists. Ninety-six percent of respondents are aware of the NIIP registration process or have taken part in it directly. Seventy-one percent report that most or all firms register at least one version of their software.

Each registration reports the program's title, intended use, and technological features, including whether it uses AI methods such as neural networks or deep learning.[2] An NIIP examiner verifies this information before issuing the registration certificate. I collect all registrations that use AI technology, which I define as software that learns from data to generate predictions, recommendations, guidance, or content (Agrawal et al. 2019b, Brynjolfsson and McElheran 2019, McElheran et al. 2023, Eloundou et al. 2023, Brynjolfsson et al. 2025). Because these records contain verified technical and application details on deployed software, they provide a more direct measure of AI development than patents or job advertisements, the standard proxies in the existing literature.

The NIIP registry shows three facts about AI development in Brazil. First, AI development accelerated sharply in the 2010s, mirroring the global trend: from 2013 to 2024, the number of firms owning AI software increased by 409 percent.[3] Second, AI is often used in production-related activities. While 48 percent of AI software targets clerical functions, 44 percent targets production activities in manufacturing, agriculture, construction, and related sectors. Third, production-oriented AI is concentrated in a few practical applications: 40 percent of production AI classifications are software to manage production, and 21 percent handle automated quality inspection. Together, these facts challenge the view that AI mainly affects white-collar work and point to important labor-market implications for machine operators and other shop-floor workers, which I investigate next.

To estimate how AI affects different occupations, I calculate an occupation-level AI exposure over time. The AI exposure is constructed using text similarity between the tasks performed by each occupation and the descriptions of AI software in the NIIP database, following standard approaches in the literature (Webb 2020, de Souza and Li 2023, Kogan et al. 2023, Hampole et al. 2025). The intuition is simple: occupations whose tasks resemble those performed by AI software are more exposed to AI development, which can either complement or substitute their work. By conducting the analysis at the occupation level, the AI exposure captures workers' exposure to both in-house and vendor-supplied AI and the resulting reallocation of workers across firms and sectors.[4]

To identify the causal effect of this AI exposure, I instrument it with changes in AI development costs due to advances in the US. AI software is cheaper to develop when more AI programming resources, such as packages, documentation, and tutorials, are available

[2] A registration is a legal record rather than a technical unit: several software products may be bundled in a single registration. For this paper, the relevant feature is that each registration contains detailed descriptions of the bundled products and the firm's software-development activity.

[3] Acemoglu et al. (2022), Babina et al. (2024), and Adams et al. (2024) also document a large increase in AI development in the US. According to Kaplan et al. (2020), the AI boom of the 2010s was driven by abundant data, more powerful computers, and advances in machine learning.

[4] I conduct the analysis at the occupation level because 81% of software developed in Brazil is produced by specialized IT firms and licensed to many clients; a firm-level analysis would therefore miss much AI adoption through third-party vendors.

(Delorey et al. 2007, Krein et al. 2010, Tambad et al. 2020, Vijayaraghavan et al. 2022). I construct an instrument that measures the availability of AI programming resources in the United States, where changes in programming knowledge are plausibly unrelated to labor-market conditions in Brazil.

I support the instrument with a battery of validation and placebo tests. The instrument is not correlated with the adoption of other technologies in Brazil, equipment prices, import prices, export prices, labor regulation enforcement, or labor market trends. Placebo tests also show that the instrument is unlikely to capture correlated shocks across countries. It does not predict Brazilian labor-market outcomes before the AI boom of the 2010s or during the rise of computerization from 1995 to 2002. In addition, AI programming resources in Russian or Japanese do not predict AI development in Brazil, consistent with the instrument working through programming knowledge in a language accessible to Brazilian developers.

I find that AI increases employment while drawing in workers with lower observable and unobservable skills. A one-standard-deviation increase in AI exposure raises occupational employment by about 2%, with stronger gains among younger, less educated, less experienced, and lower-ability workers. After conditioning on worker characteristics, AI increases wages by 0.8%.[5]

AI also reduces wage inequality within and across occupations. Within occupations, a one-standard-deviation increase in AI exposure lowers the 90-to-10 wage ratio by about 2% by increasing employment of less skilled workers, consistent with experimental evidence that AI tools raise productivity more for lower-skill workers than for experts (Brynjolfsson et al. 2025, Peng et al. 2023, Kanazawa et al. 2022). Across occupations, it reduces average earnings and education more in initially high-wage jobs, compressing wage differences between occupations. Together, these results suggest that AI lowers skill barriers and opens access to roles previously held by more qualified workers (Autor 2024).

These average effects, however, mask important heterogeneity between office and production workers. AI raises employment in production-related occupations, such as agriculture and manufacturing, by about 5%. In contrast, it reduces employment in administrative occupations by the same amount. Employment among managers and high-skilled professionals remains largely unchanged.

Why does AI automate the office but expand the factory? The evidence points to a mechanism that depends on how occupations interact with information. AI reduces employment in occupations centered on data-processing tasks, such as data entry, tax compliance, and form completion, but increases employment in occupations performing decision-making,

[5]According to the model by Autor and Thompson (2025), these results are consistent with AI reducing the expertise requirement of each occupation, bringing in more low-skilled workers and holding down average wages.

production-management, or machine-operation tasks. More broadly, I show that AI raises employment in occupations that use information to make decisions and reduces it in occupations centered on processing information. This pattern is consistent with AI as a prediction technology (Agrawal et al. 2019b): it substitutes for workers who record and organize information, but complements workers who use information to act.

By measuring AI applications in both the office and the factory, the NIIP registry also helps explain why studies using different AI measures reach different conclusions. The office-factory contrast implies that a measure that captures administrative AI better than production AI will place more weight on the displacement margin. To show this, I repeat the analysis using standard off-the-shelf occupational AI exposure measures, as is common in the literature, interacting each measure with the aggregate stock of AI software registrations in a year. These measures produce lower, generally negative employment effects. The reason is that they overstate the AI exposure among administrative workers and understate exposure among production workers. Consistent with this pattern, studies that capture production uses of AI tend to find more positive effects (McElheran et al. 2023, Brynjolfsson and McElheran 2019, Brynjolfsson et al. 2021, Aghion et al. 2025). The disagreement in the literature therefore partly reflects differences in which AI applications each measure captures.

These results speak directly to the literature on the labor-market effects of AI, where estimates differ widely across datasets, measures, and applications. Existing studies have reached mixed conclusions: some show that AI can substitute for workers or reduce labor demand (Acemoglu and Restrepo 2018, Acemoglu et al. 2022, Webb 2020, Hosseini and Lichtinger 2025), while others find AI raises productivity, firm growth, or employment (Agrawal et al. 2019b,c,a, Autor 2024, Bresnahan 2021, Babina et al. 2024, Aghion et al. 2025). One potential reason for these differences is that they measure AI adoption in different ways, including job postings (Acemoglu et al. 2022), resumes (Babina et al. 2024, Hampole et al. 2025, Hosseini and Lichtinger 2025, Babina et al. 2025), patents (Webb 2020), surveys (McElheran et al. 2023, 2025, Brynjolfsson and McElheran 2016, Brynjolfsson et al. 2021), software investment (Barth et al. 2023, Aghion et al. 2025), or local exposure measures (Bonfiglioli et al. 2023, Chandar et al. 2025). Another is that much of the evidence comes from particular AI applications, such as chatbots and generative AI tools (Humlum and Vestergaard 2025a,b, 2026), predictive analytics (Brynjolfsson and McElheran 2019, Brynjolfsson et al. 2021), AI pricing (Adams et al. 2024), or industrial AI (McElheran et al. 2025).

This paper also contributes to the literature on technological progress and labor markets. Existing work shows that automation can displace workers, especially when industrial robots, machines, or software substitute for routine and low-skill tasks (Autor et al. 2003, Graetz and Michaels 2018, Acemoglu and Restrepo 2020, Goos et al. 2014). At the same time, adoption can raise productivity, expand scale, and increase employment at adopting firms or in connected sectors (Autor and Salomons 2018, Jaravel et al. 2020, Humlum 2019, Hubmer and Restrepo 2021, Martinez 2021). More recent work broadens the evidence beyond robots to computerized machine tools, production equipment, patents, and new tasks, showing that the labor-market effects of technology depend on what tasks are automated and what complementary tasks are created.[6]

Finally, this paper also contributes to the literature on technology diffusion and adoption in developing economies by being one of the first papers to study the effect of AI on the labor market of a developing economy. Prior work shows that technology adoption, task content, productivity gaps, and firm-size differences vary systematically across development levels, so the effects of new technologies need not mirror those in frontier economies (Caunedo et al. 2025, Eden and Gaggl 2020, Gollin et al. 2014, Poschke 2018). This literature also emphasizes that new technologies diffuse gradually across countries, regions, firms, and workers (Bonfiglioli and Gancia 2008, Kalyani et al. 2025a, Kalyani 2020). I contribute by showing how AI affects labor markets in a large middle-income economy with a developed domestic software sector, where AI is developed locally and applied across both office and production tasks.

2 Theoretical Model

This section develops a simple task-based model to guide the empirical analysis. The model serves two purposes: it clarifies how to interpret the empirical findings and motivates an instrument based on shocks to AI development costs. Its central implication is that cheaper AI can either raise or reduce employment. The sign depends on two forces: how much lower AI prices expand task demand, and how easily firms substitute AI for workers within tasks.

2.1 Production and AI

Output and occupations. The final good combines output from broad occupational groups $g \in \mathcal{G}$ using a CES aggregator with elasticity of substitution $\chi > 0$. Within each broad group, a continuum of tasks produces output through another CES aggregator:

$$Y_g = \left(\int y(j)^{\frac{\psi-1}{\psi}} \, dj \right)^{\frac{\psi}{\psi-1}},$$

[6]See Boustan et al. (2022), de Souza and Li (2023), Hémous et al. (2025), Autor et al. (2024), Kalyani et al. (2025b), Bonfiglioli et al. (2024), Blanas et al. (2020), Bekes et al. (2025), Danieli (2026), Gaggl and Wright (2017), Eden and Gaggl (2018), Cortes et al. (2020), Leduc and Liu (2024), Alvarez-Cuadrado et al. (2017), and Poschke (2018).

where Y_g is output of broad occupational group g, $y(j)$ is output of task j, and $\psi > 0$ is the elasticity of substitution across tasks within group g. Each broad group contains a finite set of occupations. The tasks in group g are partitioned across those occupations.

Worker heterogeneity. Within each occupation o, there are two types of workers: low-skilled and high-skilled workers, with wages $w_{o,L}$ and $w_{o,H}$. These types capture within-occupation heterogeneity in age, experience, education, or ability. For simplicity, both groups supply labor with the same elasticity η.

Task production. Consider a task j in the task set of occupation o. If that task is produced by workers of skill $s \in \{L, H\}$, output is

$$y_{o,s}(j) = A_{o,s}(j) \left[\ell_{o,s}(j)^{\frac{\sigma_{g,s}-1}{\sigma_{g,s}}} + K_{o,s}(j)^{\frac{\sigma_{g,s}-1}{\sigma_{g,s}}} \right]^{\frac{\sigma_{g,s}}{\sigma_{g,s}-1}}.$$

Here, $A_{o,s}(j)$ is a productivity shifter, $\ell_{o,s}(j)$ is labor input, and $K_{o,s}(j)$ is AI input.

Elasticity of substitution between AI and labor. The elasticity of substitution between AI and labor, $\sigma_{g,s}$, varies by broad group g and skill group s. This flexibility captures two patterns in the data. First, AI may complement low-skilled workers more than high-skilled workers, which corresponds to $\sigma_{g,H} > \sigma_{g,L}$. Second, AI may have different labor-market effects in different broad occupational groups, such as office work and production work.

AI developers. AI software $K_{o,s}(j)$ is supplied competitively by software developers. Each AI developer combines services from different programmers to produce AI software:

$$K_o(j) = \prod_{l \in \mathcal{L}} x_{o,l}(j)^{\alpha_{j,l}}, \qquad \sum_{l \in \mathcal{L}} \alpha_{j,l} = 1,$$

where $x_{o,l}(j)$ is the quantity of programming services in language l used for task j, and $\alpha_{j,l}$ is the importance of language l for that task. If $c_{l,t}$ is the marginal cost of programming services in language l, the developer's cost-minimization problem implies the AI software price

$$q_o(j) = \prod_{l \in \mathcal{L}} \alpha_{j,l}^{-\alpha_{j,l}} c_l^{\alpha_{j,l}}$$

Equilibrium. An equilibrium consists of wages, $\{w_{o,L}, w_{o,H}\}_o$, and task-specific AI prices, $\{q_o(j)\}_j$, such that firms optimize, AI developers price software at unit cost, and product and occupation-level labor markets clear.

2.2 Effects of Cheaper AI on the Labor Market

Technological shock. Technological progress lowers AI prices heterogeneously across tasks:

$$\tau_o(j) = -d \ln q_o(j) \tag{1}$$

where a larger $\tau_o(j)$ denotes a larger decline in the AI price for task j in occupation o.

Many forces can lower AI costs. In the empirical analysis, I use variation from changes in the productivity of software programmers. In the model, this variation appears as a decline in c_l. The language-level cost shock is

$$\tau_l = -d \ln c_l, \tag{2}$$

$$\tau_o(j) = -d \ln q_o(j) = \sum_l \alpha_{j,l} \tau_l. \tag{3}$$

Heterogeneous exposure to technological shocks. Occupations differ in their exposure to the technological shock because their tasks rely on different language bundles. I define occupation o's AI shock exposure as the employment-weighted average decline in AI prices across its tasks:

$$\textit{AI Shock Exposure}_o = \int_{J_o} s_{M,o}(j) \lambda_o(j) \tau_o(j)\, dj = \sum_{l \in \mathcal{L}} \Omega_{o,l} \tau_l \tag{4}$$

where $s_{M,o}(j)$ is the AI expenditure share in task j and $\lambda_o(j) \equiv \ell_o(j)/L_o$ is the share of occupation-o employment allocated to task j. The second equality uses the language-level shocks in (2): occupation exposure depends on the size of each language shock, $\tau_{l,t}$, and on the occupation's dependence on that language, $\Omega_{o,l}$. Proposition 1 below explains how a decline in the price of AI affects employment across different occupations.[7]

Proposition 1 (AI has an ambiguous effect on employment) *Assume that the initial AI expenditure share is small. Then AI exposure affects employment in occupation*

[7] For tractability, I impose a balanced-exposure restriction: within each occupation, high- and low-skilled workers are equally exposed to the AI price shock. This restriction simplifies the expressions derived from the model while preserving the main economic mechanism.

o according to

$$d\ln L_o \approx \underbrace{\frac{\eta}{\eta+\psi}\Xi_g}_{\textit{Scale Effect}} + \beta_o^L \textit{ AI Shock Exposure}_o,$$

where Ξ_g is a function of parameters and

$$\beta_o^L = \frac{\eta}{\eta+\psi}\left[\underbrace{\psi}_{\textit{Demand Effect}} - \underbrace{\bar{\sigma}_g}_{\textit{Replacement Effect}}\right], \qquad \bar{\sigma}_g = \pi_o\sigma_{gL} + (1-\pi_o)\sigma_{gH}.$$

Proposition 1 highlights that the coefficient β^L captures two opposing forces. The first is a demand effect: when AI lowers task costs, demand for those tasks increases. This force is governed by ψ, the elasticity of substitution across tasks. The second is a replacement effect: as AI becomes cheaper, firms substitute workers for AI within a task. The strength of this effect is governed by the average elasticity of substitution between AI and workers, $\bar{\sigma}_g$. The elasticity β_o^L is positive when the demand effect dominates the replacement effect, and negative when the opposite is true.

AI can displace some occupations while expanding others. Proposition 1 implies that the sign of β_o^L may differ across occupations. AI reduces employment where substitution dominates and increases employment where demand expansion dominates. Therefore, the average effect of AI depends on the composition of the labor force, the source of identifying variation, and the AI applications measured. This logic helps explain why estimates can differ across studies. This intuition is formalized in Proposition 2.

Proposition 2 (Average Treatment Effect Depends on Sample Composition) *The linear projection of occupation employment on AI exposure can be approximated by*

$$d\ln L_o \approx \frac{\eta}{\eta+\psi}\Xi_g + \beta^L \textit{ AI Shock Exposure}_o + \epsilon_o,$$

where

$$\beta^L = \sum_o \omega_o \beta_o^L,$$

with $\pi_o, \omega_o \in (0,1)$, $\sum_o \omega_o = 1$, and ϵ_o orthogonal to AI Shock Exposure$_o$.

AI can increase the share of low-skilled workers within occupations. If AI complements low-skilled workers more than high-skilled ones, i.e., $\sigma_{g,H} > \sigma_{g,L}$, it should increase employment of low-skilled workers by more, leading to a shift in employment toward the low-skilled group. Proposition 3 formalizes this idea.

Proposition 3 (AI can change the composition of workers within occupations) *Assume that the initial AI expenditure share is small. Then, the effect of AI on the share of low-skilled workers is given by:*

$$dShr.\ \textit{Low-Skilled}_o \approx \Theta_o \left(\sigma_{g,H} - \sigma_{g,L}\right) \textit{AI Shock Exposure}_o,$$

where $\Theta_o > 0$ is a set of parameters.

AI affects average wages through both direct wage effects and worker composition. Because AI affects the composition of workers, average wages might go down even if employment increases. To formalize that, let $\bar{w}_o$ denote the average wage in occupation o. A first-order decomposition gives

$$d \ln \bar{w}_o \approx \vartheta_{oL} d \ln w_{oL} + \vartheta_{oH} d \ln w_{oH} + \frac{w_{oL} - w_{oH}}{\bar{w}_o} dShr.\ \textit{Low-Skilled}_o,$$

where ϑ_{os} is the wage-bill share of skill group s in occupation o. The first two terms capture the direct effect of AI on skill-specific wages. The last term captures the composition effect. Because low-skilled workers typically earn less than high-skilled workers, a rise in the low-skilled share can lower average wages even when AI raises productivity and employment.

3 Data

The model identifies four relevant objects to identify the effects of AI: a representative description of what AI software does, which occupations perform related tasks, which programming languages are used to develop each application, and cost-shocks to AI development. This section introduces the data used to observe these objects and explains why Brazil offers an unusually clear setting for studying the labor-market effects of AI.

The primary data source is the software registry maintained by the National Institute of Industrial Property (NIIP), which contains a broad sample of in-house, third-party, and commercially marketed software developed in Brazil (Barbosa et al. 2022). The registry is central to the empirical design because it reports each program's intended use, technical features, and programming languages. These fields allow me to identify AI software, link it to occupational tasks, and construct the cost-shock instrument motivated by the model.

The registry is unusually comprehensive for two reasons. First, Brazil has a mature and self-sufficient IT sector: it is the 11th largest in the world and has the third-largest pool of software developers (GitHub Staff 2024). Second, regulatory barriers limit the import share of computer services to about 9%, meaning most software deployed by Brazilian firms is locally developed and therefore captured in the NIIP database.[8]

I use the NIIP registry to construct occupation-level measures of AI exposure and link these measures to RAIS, an administrative dataset that tracks all formal jobs from 2003 to 2024 in narrowly defined 6-digit occupational codes, allowing precise measurement of employment and wage dynamics. Finally, I use Stack Overflow data to construct programming-language-specific shocks to AI development costs: Stack Overflow posts measure growth in the U.S. stock of AI programming knowledge that Brazilian developers can reuse, lowering the cost of AI software development (Mamykina et al. 2011, Treude and Robillard 2016, Wu et al. 2019, Ponzanelli et al. 2014).

3.1 Administrative Software Registry Data

3.1.1 Institutional Background

Brazilian software law provides special copyright protection for software. Since 1987, firms in Brazil have been able to register software with the National Institute of Industrial Property (NIIP) under a special copyright system that grants 50 years of protection.[9] The registration process is inexpensive, keeps the source code confidential, and provides legal protection against unauthorized copying, giving firms strong incentives to register commercially relevant software.

A software registration contains examiner-verified information about the software. To register software, firms must submit an electronic form with the firm's name, the programmers' names, a descriptive title of the software, and the programming languages. Importantly, firms also provide two classifications that describe the software's purpose and technical characteristics.

The first classification, called "application domain," identifies the sector or area where the software will be used. There are 225 specific application domains grouped into 35 broad categories. For example, *MapForest*, an AI program that forecasts weather and predicts land productivity, is classified under Agriculture. Each software entry includes at least one application domain, though most list multiple; on average, each program lists three.[10]

[8]I discuss in detail the import share of computer services in Appendix B.1.

[9]In Portuguese, the NIIP is called *Instituto Nacional de Propriedade Industrial*. The first software law was Law number 7,646/1987. It was later updated by Law 9,609/1998. The law defines software as a "set of instructions or statements, written in proper language, to be used directly or indirectly by a computer to obtain a certain result."

[10]Each application domain is accompanied by a text description, which I use to calculate the text-similarity

The second classification, called "technical class," organizes software by its technical function. The system includes 97 specific technical classes divided into 18 broad groups, ranging from fundamental systems like Operating Systems and Network Controllers to specialized tools like Encryption Software and AI. *MapForest*, for instance, is classified under four technical classes: Artificial Intelligence, Simulation, Pattern Recognition, and Office Automation.[11]

Software registration provides proof of ownership. The information provided by the firm is evaluated by an examiner at the NIIP and, if approved, the firm receives a registration certificate that serves as official legal proof of ownership. Figure 1 shows one such certificate. These certificates are commonly used in cease-and-desist letters, piracy takedown requests, and intellectual property lawsuits.[12]

Why are firms registering their software? Firms and developers both have concrete reasons to register software. For firms, registration helps protect intellectual property, participate in government procurement of software and IT services, document prior innovation when applying for R&D credit, and signal the value of intangible assets to investors (Torquato 2022, Associação Brasileira das Empresas de Software (ABES) 2024). For developers, it provides verifiable proof of experience that can be used on resumes and professional profiles. This credential carries substantial economic value: de Souza et al. (2024) documents that programmers receive a 5% permanent wage increase after being listed on a registered software project.

3.1.2 Administrative Software Registry Data

I extracted information from the universe of NIIP certificates of software registration between 1987 and 2024. For each entry, I observe all the information in the certificate of software registration (Figure 1). In the empirical section, I focus on AI software, defined as software with the technical class "Artificial Intelligence" ("IA").

There are 47,427 registrations, which, importantly, does not reflect the number of distinct software products. Registration is a legal tool, not a technical unit. A comparison of the software registrations with public online records reveals that firms usually bundle different software products in one registration. Legally, this approach offers the same level of protection as if the firm were to register each program separately. For the purposes of this

between AI software and task descriptions.

[11]Similarly, each technical class is accompanied by a textual description.

[12]In the registration process, firms also provide a hash summary: a unique, fixed-length string generated by applying a cryptographic hash function to the source-code. Because any modification to the code would alter the hash, this digital "fingerprint" serves as proof that the firm possessed that exact version of the code on the filing date. The hash can later be compared to the confidential source code to establish authorship in legal proceedings.

Figure 1: **Example of a Software Registration**

FEDERATIVE REPUBLIC OF BRAZIL
MINISTRY OF THE ECONOMY
NATIONAL INSTITUTE OF INDUSTRIAL PROPERTY
DIRECTORATE OF PATENTS, COMPUTER PROGRAMS, AND TOPOGRAPHIES OF INTEGRATED CIRCUITS

Certificate of Software Registration

Case Nº: **BR512021001026-4**

The National Institute of Industrial Property issues this certificate of registration of a computer program, valid for 50 years starting from January 1st following the date of May 3, 2021, in accordance with §2, Article 2 of Law No. 9,609 of February 19, 1998.

Title: iMachine AI: Online Predictive Maintenance System for Electric Machines Using Machine Learning Techniques

Publication Date: 05/03/2021

Creation Date: 04/28/2021

Owner(s): TECHPLUS AUTOMACAO

Author(s): SAMARONE RUAS

Programming Language: PYTHON

Application Domain: AD-06; FQ-04; IF-10; IN-02

Technical Class: AP-01; AT-05; IA-01; IT-01; TC-03

Hash Algorithm: SHA-512

Digital Hash Summary:
cec9fe6467e2da2a9cc51fad00cfaedf52adc26dd5f352d4f4ed7c8600e399415383cc
2d542eab52cf2a49bf66955b20317e
bf21bafa7273d7edcdb534767588

Issued on: 05/25/2021

Approved by:
Carlos Alexandre Fernandes Silva Head of DIPTO

Notes: This figure shows a translated certificate of software registration. The application domains correspond to "factory planning", "metrology", "information management", and "manufacturing technology". The technical classes are "application", "industrial automation", "artificial intelligence", "instrumentation", and "pattern recognition".

paper, what matters is that commercially relevant software is typically registered at least once with a description of its intended use, even when firms bundle multiple products within a filing. I discuss the nuances of this database in further detail in the next section.

3.1.3 Coverage, strengths, and limitations

This subsection evaluates the NIIP registry as a measure of AI software development. Registration is widespread among Brazilian software firms, and survey evidence supports its broad coverage. The registry's main advantage is that it captures actual software developed for use by firms, including tools that lack patent-worthy novelty and third-party vendor software missed by hiring data. I also discuss limitations related to scope and registration practices and explain why they are unlikely to drive the results.

Registration is nearly universal among software developers. In 2016, 92% of software development firms with more than two years of operation had registered at least one software program.[13,14] Weighted by employment, this figure rises to 99.4%, and among firms with more than 20 employees it reaches 96%.[15] These numbers suggest that the NIIP registry captures a large share of commercially developed software created in Brazil.

Survey with private sector specialists. To validate the dataset and better understand the degree of selection, I conducted a survey with IT firms and intellectual property lawyers, described in more detail in Section B.1.1. The survey provides three main findings. First, software registration in the NIIP is widely known among both intellectual property lawyers and IT professionals, with the vast majority of respondents having direct experience with the registration process. Second, respondents confirm that most commercial software developed in Brazil has at least one version registered, with firms primarily motivated by intellectual property protection. Third, to the extent that selection exists, it is driven by small firms producing consumer-facing applications such as video games and mobile apps. These categories are unlikely to affect the labor market outcomes studied in this paper.

Registration titles validate the AI classification. Appendix Figure A9, in the appendix, shows that common title terms include prediction tasks, such as forecast, recommendation, classification, recognition, and detection, as well as AI methods such as machine learning, neural networks, and deep learning. These patterns suggest that NIIP software classified as AI captures the predictive technologies emphasized in the AI literature (Agrawal et al. 2019b).

3.2 Other Datasets

RAIS. I use the publicly available version of *"Relação Anual de Informações Sociais"* (RAIS), a matched employer-employee dataset covering all formal workers in Brazil. It reports each worker's occupation, earnings, hours, and other job details. I rely on this version because it extends coverage through 2024.[16] For privacy reasons, the public version excludes person identifiers, so I use a panel at the occupation–year level. Since 2003, RAIS has used the occupational classification system *Classificação Brasileira de Ocupações 2002* (CBO 2002), which includes detailed occupation descriptions. To avoid crosswalk issues

[13] I call a firm a software development firm if it has Classificação Nacional de Atividades Econômicas (CNAE) 2002 codes 6201500 (custom software development) or 6202300 (development and licensing of customizable software), and has hired a computer engineer at least once in its life-cycle.

[14] I restrict the sample to firms with more than two years of operation because, in this sample, software development typically requires more than two years.

[15] This figure represents a conservative lower bound on overall coverage because not all of these firms have developed their software: (i) some firms might only provide technical support and do not write proprietary code, (ii) some firms are third-party developers whose software is registered by the client, and (iii) young firms are still completing their first product.

[16] The longitudinal version tracks workers over time but only runs through 2016, missing important years of AI development.

with older classifications, I restrict the analysis to 2003–2024. I exclude from the sample IT workers, government employees, and military personnel.

Description of Occupational Tasks. To measure workers' exposure to AI, I use data from the Ministry of Labor, which links each six-digit occupation in the CBO 2002 classification to detailed descriptions of its tasks. Because the CBO is used in labor-court proceedings to determine employment benefits and labor union association, these descriptions are long-form, legally precise, and highly specific.[17] The dataset covers 2,641 occupations, each with an average of 64 task statements, for a total of 49,660 unique tasks.

Stack Overflow. To build the instrument, I use data from Stack Overflow and programming mailing lists. Stack Overflow is a widely used question-and-answer platform for programmers. On this platform, programmers post questions seeking help from the community on specific technical problems. Each post includes standardized tags indicating the programming language and type of application involved, selected from a community-maintained list. I collect data on all Stack Overflow posts created between 2008 and 2024.[18]

Programming Mailing Lists. I complement the Stack Overflow data with public programming mailing-list archives from Gmane. These mailing lists are maintained by software projects and technical communities to discuss specific technical problems. For each message, I observe the mailing-list group, date, and subject. The data cover 1995–2024.

4 Statistics of AI Software

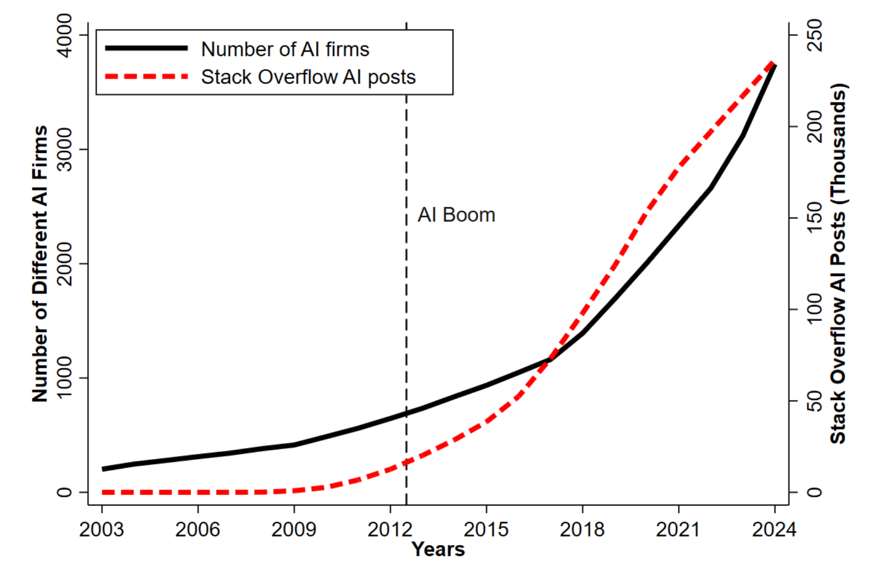
The software registry data present three facts about AI software development in Brazil.[19] First, following global trends, AI development accelerated sharply in the 2010s. Second, AI is used across a wide range of application domains: while administrative uses such as report generation remain common, 40% of application-domain classifications target production settings, including manufacturing, agriculture, and logistics. Third, two applications dominate production use: manufacturing execution systems that coordinate workflow and quality-control tools that perform real-time visual inspections.

[17] For instance, in the states of Rio de Janeiro and Paraná minimum wages depend on the worker's CBO occupation.

[18] Appendix C.3 provides a series of summary statistics for Stack Overflow posts.

[19] Additional statistics are reported in Appendix B.2.

Figure 2: **Number of Firms Owning AI Software**



Notes: This figure shows the number of unique firms in Brazil with at least one AI software registration from 2003 to 2024. Because many of these firms provide services to others, the number of firms using AI is likely much higher than those directly owning the software. The dashed red line plots the number of Stack Overflow posts with AI tags. AI tags are defined in Section C.3.

Brazilian AI development follows the global trend. Figure 2 plots the number of unique firms with at least one AI software registration. From 2013 to 2024, the number of firms owning AI technologies increased 409%. This period, known as the AI boom (Toosi et al. 2021, Chauvet 2018, Sevilla et al. 2022), was marked by a surge in AI development worldwide, fueled by scientific breakthroughs in neural networks and GPU computing (Metz 2021). Figure 2 also shows, for comparison, that the number of Stack Overflow posts mentioning AI tags increased by a similar magnitude.[20]

AI is used in management and production. When firms register software, they report its application domains, that is, the areas where the software is intended to be used. Figure 3a presents the ten most common broad application domains for AI software. Managerial domains such as information management and administration are the most common, but they account for only 27% of application-domain classifications. AI is also widely used in healthcare, manufacturing, and agriculture, which together account for 16.7% of the total.

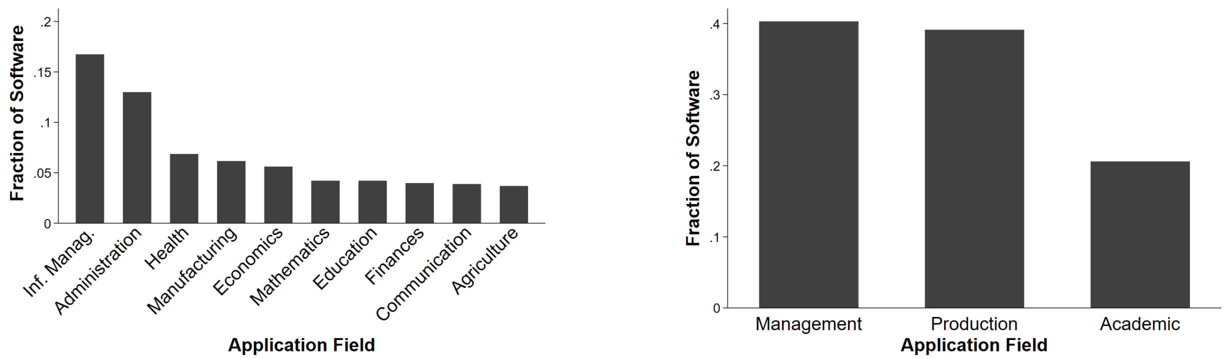
Figure 3b groups all application domains into three categories: management, production, and academic. Management includes back-office functions such as human resources, accounting, and administration. Production includes non-management applied domains such as manufacturing, agriculture, and construction. Academic includes research-oriented applications in fields such as mathematics and the social sciences.

[20]Similar trends in AI development have been documented in job postings for AI scientists (Acemoglu et al. 2022), AI patent applications (Webb 2020), and academic publications in AI (Maslej et al. 2025).

Figure 3: **AI Software Registrations by Intended Use**

(a) Top 10 AI Application Domains

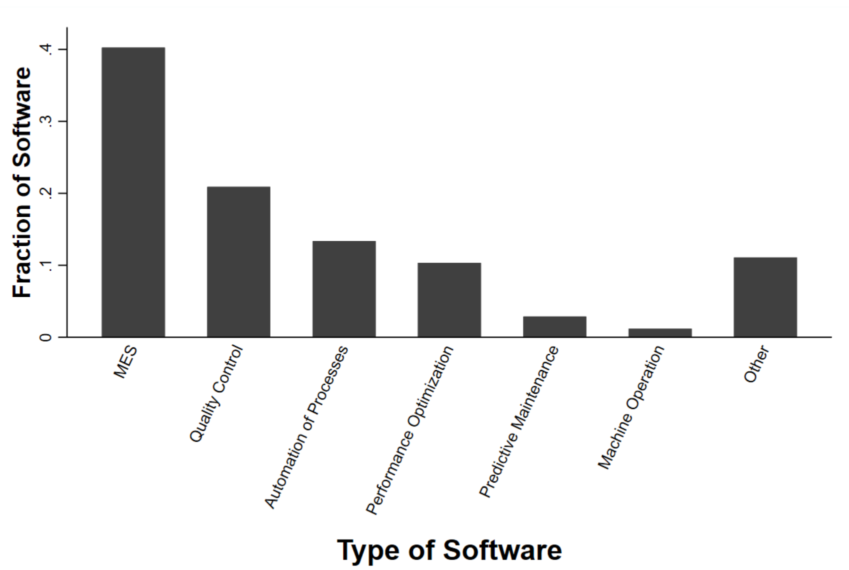
(b) AI on Management and Production



Notes: These figures present the distribution of AI software across different application domains, as defined by the NIIP. Because one registration may list multiple domains, Figure 3a reports each domain's share of all domain classifications. Figure 3b assigns each registration to the broad group containing most of its domains. Management includes domains related to business administration, law, economics, finance, information storage, telecommunications, and human resources. Production includes agriculture, civil construction, ecology, education, energy, physics, chemistry, geography, geology, housing policy, hydrology, engineering, meteorology, soil studies, sanitation services, public infrastructure, transportation, biology, botany, and health. Academic includes mathematics, cultural studies, politics, public policy, psychology, environmental policy, social studies, city planning, and social welfare.

Figure 3b shows that AI is split almost evenly between management and production applications, each accounting for about 40% of application domains. This balance challenges the prevailing view that AI primarily targets high-skill, white-collar occupations (Frey and Osborne 2017, Webb 2020, Felten et al. 2021). Because a comparable share of AI tools is used in production, blue-collar workers may be just as exposed to AI as administrative workers.

Figure 4: **Specific Use of AI Software in Production**

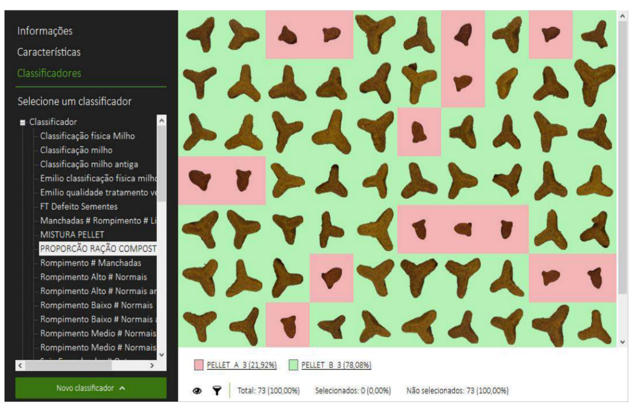


Notes: This figure shows the distribution of AI production software across application categories. Each software entry was classified into one of the groups based on its title, technological class, and application domain by an LLM.

Production AI coordinates workflow, controls quality, and automates tasks. To better understand how AI is used in production, I classify each registration with a production-related application domain into one of seven mutually exclusive categories, based on the engineering literature and a manual review of program descriptions. Figure 4 plots the distribution of production AI software across the different categories.[21]

The most common production use of AI software is in Manufacturing Execution Systems (MES), which coordinate and control production processes. MES applications account for nearly 40% of all AI production applications in the database (Figure 4). One example is an AI system developed by Copel, a large electricity distribution company. Copel's system monitors household-level energy use and uses AI to predict power outages.[22] When the system detects risk, it automatically dispatches maintenance and inspection teams with the location, diagnosis, and suggested repairs, increasing the productivity of field workers. According to Consoli (2025), the software has prevented 1,700 outages. More broadly, AI-enabled MES has been shown to reduce production time, increase yields (Chen et al. 2006), and improve machine utilization rates (Capgemini 2020), which could raise productivity and labor demand.[23]

Figure 5: **Example of AI Software Used in Quality Control**



Notes: Example of AI software from Tbit. Image from Tbit (2024).

The second most common use of AI is in quality control (Figure 4), where computer-

[21]To classify AI production software, I reviewed the engineering and operations literature and defined seven categories: Manufacturing Execution Systems, quality control, industrial automation, predictive maintenance, performance optimization, machine operation, and other (Chen et al. 2006, Capgemini 2020, Shojaeinasab et al. 2022, Ghelani et al. 2024, Tariq et al. 2024, Elsafty et al. 2024, Shi et al. 2023, Borboni et al. 2023, Marr and Ward 2019, Keleko et al. 2022, Ucar et al. 2024, Murtaza et al. 2024, Evans and Gao 2016, Liu et al. 2022). I then used Gemini to assign each AI software registration to one category using category definitions, hand-coded examples, decision rules, and the full registration information.

[22]The use of AI in electricity distribution is widespread. Eight other utility companies in Brazil have developed their own AI tools. For a review of how AI is used in electricity distribution, see Forootan et al. (2022).

[23]See Shojaeinasab et al. (2022) for a review of AI in MES.

vision software detects defects in real time. A notable example in the database is Tbit's image-recognition system, which identifies visual impurities in products such as seeds, flour, and animal feed. In one deployment at a dog food factory, the software flagged defective units for removal from the production line, as illustrated in Figure 5. It also relayed defect rates and quality metrics to the Manufacturing Execution System (MES), which used the data to recalibrate machines, adjust input mixes, and guide operator decisions. As documented in the engineering literature, integrating AI into quality control improves precision and productivity by reducing inspection time, minimizing human error, and lowering defect rates (Ghelani et al. 2024, Tariq et al. 2024, Shi et al. 2023).

5 Empirics

This section describes the empirical strategy to estimate the effect of AI on the labor market. I first construct an occupation-level measure of AI exposure by computing the cosine similarity between task descriptions and AI software descriptions. To address differences in terminology, I apply latent semantic analysis, which maps synonyms and technical jargon onto shared semantic dimensions. I then use an instrumental variable strategy that exploits heterogeneous exposure to the growth of AI programming knowledge across programming languages in the United States. These knowledge shocks change the cost of developing AI software in Brazil and generate variation in AI exposure that is plausibly unrelated to Brazilian labor-market conditions.

5.1 AI Exposure

Text similarity between occupations and AI software: $\rho_{o,s}$. To measure AI exposure, I link the tasks performed by workers to those carried out by AI software. This measure captures the overlap between AI software and workers' tasks, which could arise from either potential substitution or complementarity between them. I construct this measure in three steps. First, I collect detailed task descriptions for each occupation from the Brazilian Ministry of Labor. Occupations are defined narrowly at the six-digit level, giving 2,641 distinct occupations in total.[24] On average, each occupation has 64 tasks, for a total of 49,660 unique tasks. Second, I construct descriptions for each AI software record by combining its title, the description of the listed application domains, and description of technological classes. Third, I use latent semantic analysis to calculate the cosine text similarity, $\rho_{o,s}$, between the tasks of occupation o and the description of AI software s. Appendix C.1 provides further details.

[24]As an example of the narrowness of the occupational classification, economists in agriculture, finance, manufacturing, public finance, environmental, and regional economics each have their own entry.

Software valuation: v_s**.** To construct the exposure, I weight each software by an estimate of its importance. Following Kogan et al. (2017) closely, I estimate software values from the stock market response to news of their registration. Then, I use credibility-weighted mean imputation to extrapolate software values to software of firms not in the stock market.[25] I explain the procedure in detail in Appendix C.4.

Exposure to AI. I define the exposure of occupation o to AI software available by year t as

$$AI\ Exposure_{o,t} = \sum_{s} \rho_{o,s} \times v_s \times \mathbb{I}_s \left\{ t(s) \leq t \right\}, \tag{5}$$

where $\rho_{o,s}$ is the text similarity between occupation o and software s, v_s is the software valuation, and $\mathbb{I}_s \left\{ t(s) \leq t \right\}$ equals one if software s was created in or before year t. Because firms register software shortly before commercialization, $t(s)$ also marks the year the software reaches the market. Thus, $AI\ Exposure_{o,t}$ measures occupation o's cumulative exposure to AI software commercialized by year t. Appendix C.2 report summary statistics for this measure.

Alternative AI exposure measures. The baseline measure uses stock-market responses as a measure of software importance. To test whether this choice matters, I construct two alternative measures. The first gives equal weights to all software. The second weights software by its breakthrough importance, measured by whether it is novel relative to earlier AI registrations but close to later ones following Kelly et al. (2021). The main results are unchanged under any of these alternative AI exposure measures.

5.2 Empirical Model

Main empirical model. I use the following empirical model:

$$y_{o,t} = \beta AI\ Exposure_{o,t} + \mu_o + \mu_t + X'_{o,t}\theta + \epsilon_{o,t}, \tag{6}$$

where $y_{o,t}$ is a labor-market outcome, such as employment or wages, for occupation o in year t. $AI\ Exposure_{o,t}$ is defined in equation (5) and normalized to have mean zero and standard deviation one. Therefore, β is the effect of a one-standard-deviation increase in AI exposure. The terms μ_o and μ_t denote occupation and year fixed effects. Standard errors are clustered

[25] This method extrapolates values from subgroup-, group-, and broader-group averages. It gives less weight to group levels with more uncertainty in their values. This method has been shown to be better suited to settings with many categorical groups and few observations per group (Micci-Barreca 2001, Bühlmann 1967).

at the occupation level.[26]

Controls for omitted-variable bias. The control vector $X_{o,t}$ includes four variables addressing potential confounders. First, one-digit occupation-by-year fixed effects absorb shocks common to broad occupational groups. In the model derived in Section 2, they also capture the scale effect. Second, a U.S. patent-based AI exposure measure controls for exposure to foreign-developed AI, even though imported software accounts for less than 10% of the Brazilian market, as discussed in Section B.1.[27] Third, the lagged outcome, $y_{o,t-1}$, controls for differential trends. Fourth, a non-AI software exposure measure controls for the adoption of other software. The robustness checks show that the results do not depend on these controls.

5.3 Instrument

Exogenous variation: programming knowledge in the US. Building on Equation (4), I instrument AI exposure with changes in the availability of AI programming knowledge across languages. I measure this availability using AI-related discussions in Stack Overflow and programming mailing lists, where programmers post questions, answers, examples, and reusable code. The availability of such resources has been shown to lower the cost of developing software (Delorey et al. 2007, Krein et al. 2010, Tambad et al. 2020, Vijayaraghavan et al. 2022, Mamykina et al. 2011, Treude and Robillard 2016, Wu et al. 2019, Ponzanelli et al. 2014). Because this programming knowledge is produced outside Brazil, its growth is plausibly unrelated to Brazilian labor-market conditions, except through its effect on AI development costs.

These cost shocks affect occupations differently because programming languages specialize in different applications (Nanz and Furia 2015, Rabah et al. 2010, Meyerovich and Rabkin 2013). For instance, COBOL is the dominant language in banking and government systems, but its declining community, scarce documentation, and limited AI resources make AI tools for financial systems relatively costly to build (Ovide 2022, Hartman 2015, Navot 2024). By contrast, R has a large community and abundant AI resources for statistical analysis, making AI tools in that domain cheaper to develop. As a result, occupations linked to the banking system, such as bank tellers, are relatively less exposed to AI than archivists, who perform tasks that overlap with statistical analysis.

[26]Each occupation o is a narrow six-digit occupational code that often embeds sectoral requirements. For this reason, the baseline model does not use the sector–occupation specification in Webb (2020). In the robustness checks, I show that a specification closer to Webb (2020) delivers qualitatively similar results.

[27]To construct exposure to U.S. patents, I follow Webb (2020).

Instrument. Using these insights and following the model-derived counterpart in Equation (4), the instrument is given by

$$AI\ Ease\text{-}of\text{-}Dev_{o,t} = \frac{\sum_l \overline{AI\ Exposure}_{l,o,t=0} \times AI\ Posts_{l,t}}{\sum_l AI\ Posts_{l,t}} \tag{7}$$

where l indexes programming languages. $\overline{AI\ Exposure}_{l,o,t=0}$ is the average value-weighted text similarity between occupation o and software written in language l before 2003. This baseline similarity captures programming-language specialization: some occupations are more exposed to tasks commonly performed by software written in particular languages. $AI\ Posts_{l,t}$ is the number of Stack Overflow posts and mailing-list messages in language l that mention AI terms.[28] This variable proxies for the availability of AI programming knowledge in that language, including access to a broader expert community, reusable packages, examples, tutorials, and troubleshooting material. These resources lower the cost of developing AI software in language l.

Because the endogenous regressor is cumulative AI software exposure, I use the cumulative ease-of-development measure as the instrument:

$$AI\ Ease\text{-}of\text{-}Dev\ IV_{o,t} = \sum_{k=2003}^{t} AI\ Ease\text{-}of\text{-}Dev_{o,k}. \tag{8}$$

Identifying Assumption. The instrument exploits differential exposure across occupations to an exogenous decline in the cost of developing AI software. The identifying assumption is a parallel-trends condition: in the absence of these cost shocks, the labor market outcomes of high- and low-exposure occupations would have followed similar trajectories (Goldsmith-Pinkham et al. 2020). To support this identifying assumption, in the next section I show that the instrument does not correlate with other technologies developed in Brazil, is not associated with international technological progress in machines or equipment, does not correlate with trade shocks or labor market regulation changes in Brazil, and is not correlated with labor market trends in Brazil.

Alternative instruments. I also use three alternative instruments. First, building on instrument 7, I construct a version that uses the number of AI packages released in the United States for each programming language instead of mailing lists or Stack Overflow posts. Second, I exclude programming mailing-list data and use only Stack Overflow posts, a source shown to increase programmer productivity (Mamykina et al. 2011, Treude and Robillard 2016, Wu et al. 2019, Ponzanelli et al. 2014). Third, I use a standard Bartik

[28] In Appendix C.3, I present the list of keywords and how AI posts are calculated.

instrument based on heterogeneous occupational exposure to AI adoption. The results lead to the same conclusion across all three instruments.

5.4 Instrument Validation

To validate the identifying assumption, I show that the instrument is uncorrelated with other technological shocks, technological progress in tools and equipment, trade shocks, pre-existing labor-market trends, and other labor-market policies.

The instrument is uncorrelated with other technology shocks. Table A4 tests whether the instrument is related to occupation-level exposure to patents, industrial designs, or technologies transferred to Brazil (de Souza 2022), constructed with the same text-similarity approach used for AI exposure. The estimates are close to zero and statistically insignificant. Therefore, the instrument does not capture other forms of technology adopted or developed in Brazil.

The instrument is not correlated with technological progress in tools. A related concern is that programming-language knowledge may be correlated with technological progress in the tools or equipment used by the same occupations. To test this channel, I link product categories to occupations using the text similarity between product descriptions and occupational tasks. Table A5 shows no systematic relationship between the instrument and import or export price for capital or high-tech equipment. This result suggests that the instrument is not capturing falling prices of occupation-specific tools or equipment.

The instrument is not correlated with trade shocks. Another concern is that U.S. programming-language shocks may be correlated with sectoral shocks that affect Brazil through international trade. I evaluate this concern by assigning trade exposure to occupations using their sectoral employment distribution. Table A6 shows no systematic correlation between the instrument and import input prices, total imports, import prices, and export prices. This result suggests that the instrument is not capturing sectoral trade shocks.

The instrument is not correlated with informality or the enforcement of other labor regulations. I also test whether the instrument is correlated with changes in labor-market enforcement. If occupations more exposed to lower AI development costs were also differentially exposed to changes in regulation or enforcement, the instrument should predict more labor infractions in those occupations. Table A8 shows no such relationship. The instrument is not correlated with inspection activity or infractions related to informality, safety and health, hours and rest, or wage payment. This result also indicates that AI is not affecting the share of informal workers across occupations.

The instrument is not driven by pre-existing labor-market trends. Finally, I test whether the instrument predicts labor-market outcomes before the relevant exposure period. Table A7 shows no systematic relationship between the instrument and lagged employment, education, years of education, age, or wages. This evidence suggests that the instrument is not driven by pre-existing labor-market trends.

5.5 Placebo Tests

A key identifying assumption is that U.S. shocks that affect AI programming knowledge do not directly affect occupations more exposed to AI in Brazil. The validation tests above support this assumption as there is no correlation between the instrument and trade shocks. Three additional placebo exercises further assess this concern. First, the instrument does not predict Brazilian labor-market outcomes before the AI boom of the 2010s, which addresses concerns about correlated labor-market shocks across countries and earlier computerization trends. Second, AI-related Stack Overflow posts in Russian and Japanese, languages not spoken by most Brazilians, do not predict AI development in Brazil. This result suggests that the instrument works through the availability of programming knowledge, rather than through labor shocks hitting these economies. Third, in a more direct test, U.S. labor-market outcomes do not explain Brazilian AI software development after controlling for the AI ease-of-development instrument.

The instrument does not predict labor-market outcomes before the AI boom. If the instrument isolates variation in AI exposure, it should not predict labor-market outcomes before the AI boom of the 2010s, when most AI applications entered the market. Table A10 reports placebo estimates for 2003–2010 in Panel A and 1995–2002 in Panel B.[29] The estimates show no systematic relationship between instrumented AI exposure and employment, education, age, experience, or wages, with a weakly significant and likely spurious correlation with age. Figure A12 extends this test by estimating the effect of AI on employment across alternative sample cutoff years. The effect becomes statistically significant only when the sample includes years after 2017, consistent with the sharp increase in AI development shown in Figure 2. Together, these results suggest that the instrument is not capturing correlated labor-market shocks across countries, which should also appear before the AI boom, or the effects of computerization.

Other-language Stack Overflow posts do not predict AI development in Brazil. If the instrument works through access to programming knowledge, then AI-related posts in languages rarely used by Brazilian programmers should not predict AI development in Brazil.

[29] I split the analysis before and after 2003 because the occupational classification changed in 2003.

I test this implication using Russian- and Japanese-language Stack Overflow sites, which launched in 2015. Table A9 shows that instruments built from these posts do not predict Brazilian AI exposure. At the same time, the coefficient on the main AI ease-of-development instrument remains stable across specifications. This pattern supports the interpretation that the instrument works through programming knowledge available to Brazilian programmers, not through labor-market shocks common to other countries.

Table 1: **First Stage: AI Ease-of-Development and AI Exposure**

	(1) *AI Exposure*	(2) *AI Exposure*	(3) *AI Exposure*	(4) *AI Exposure*	(5) *AI Exposure*
AI Ease-of-Dev IV	0.402***	0.390***	0.396***	0.386***	0.385***
	(0.0157)	(0.0159)	(0.0171)	(0.0172)	(0.0177)
N	56,568	56,568	56,040	56,480	55,952
R^2	0.639	0.615	0.626	0.614	0.611
F	1034.5	889.5	881.9	826.7	813.3

Notes: This table presents the estimates of the first-stage. The instrument is given by Equation (8) and the endogenous variable is the exposure to AI, defined in Equation (5). All columns include occupation and year fixed effects. Column 2 adds year-by-1-digit-occupation fixed effects. Column 3 adds year-by-sector fixed effects to Column 2. Column 4 adds year-by-2-digit-occupation fixed effects. Column 5 combines year-by-sector and year-by-2-digit-occupation fixed effects. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10, ^{**} p < 0.05, ^{***} p < 0.01$.

U.S. labor-market conditions do not explain Brazilian AI development. If the instrument captured U.S. labor-market shocks correlated across countries, controlling for U.S. labor-market outcomes should weaken its relationship with AI development in Brazil. Table A11 tests this concern by mapping Brazilian occupations to U.S. occupations and adding controls for U.S. employment, wages, and education. The coefficient on the AI ease-of-development instrument remains stable at 0.379. These results support the interpretation that the first stage reflects changes in AI programming knowledge, not U.S. labor-market shocks.

6 Results

In this section, I show that AI increases employment, especially among young, less educated, less experienced, and lower-ability workers. This compositional shift lowers average wages and compresses wage inequality. These average effects, however, mask substantial heterogeneity across occupations. Employment expands in production roles related to decision making, team management, and machine operation, but contracts in administrative occupations centered on data processing. Together, the results point to a mechanism in which AI substitutes for work that processes information, while lowering skill barriers in work that uses information to make decisions and act.

6.1 AI increases employment of low-skilled workers

AI ease-of-development instrument predicts AI exposure. Table 1 shows that the AI ease-of-development instrument strongly predicts AI exposure. In the baseline specification in Column 2, a one-standard-deviation increase in the instrument raises AI exposure by about 0.39 standard deviations. The F-statistic far exceeds 10, so the instrument comfortably passes the weak-instrument test.

Table 2: **Effect of AI on Employment**

	(1) *log(Employment)*	(2) *log(Employment)*	(3) *log(Employment)*	(4) *log(Employment)*	(5) *log(Employment)*
Panel A: 2SLS					
AI Exposure	0.0344***	0.0217***	0.0245***	0.0190**	0.0195**
	(0.00671)	(0.00708)	(0.00797)	(0.00772)	(0.00816)
Panel B: OLS					
AI Exposure	0.00425	0.00256	0.00238	0.000730	0.000862
	(0.00270)	(0.00294)	(0.00305)	(0.00304)	(0.00316)
N	50,441	50,441	50,217	50,364	50,140
Fixed Effects Interacted with Year:					
1-Digit Occ.	-	✓	✓	-	-
Main Sector	-	-	✓	-	✓
2-Digits Occ.	-	-	-	✓	✓

Notes: This table reports the effect of AI exposure on log employment. Panel A reports 2SLS estimates using the instrument in Equation (8); Panel B reports OLS estimates. Specifications follow Section 5; additional year-interacted fixed effects are listed at the bottom of the table. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**}p < 0.05,^{***}p < 0.01$.

Appendix Tables A12, A13, and A14 show where this identifying variation comes from. Table A12 shows that when a programming language receives more posts, Brazilian developers write more software in that language. Table A13 shows that these effects are larger for technical classes that a programming language is more specialized in. Table A14 further shows that languages with more AI-related posts also have more AI packages. Together, these patterns support the interpretation that programming-language resources produced in the US reduce the cost of developing specific AI applications in Brazil, shifting both the composition of software built and the occupations exposed to AI.

Table 3: **AI Increases Employment of Younger and Less Skilled Workers**

	(1) *log(Age)*	(2) *log(Yrs. Education)*	(3) *log(Tenure)*
2SLS			
AI Exposure	-0.00316^{***}	-0.00216^{**}	-0.0130^{**}
	(0.00105)	(0.000852)	(0.00571)
N	50,341	50,341	50,341

Notes: This table reports 2SLS estimates of the effect of AI exposure on worker composition: log average age, log average years of education, and log tenure. Specifications follow Section 5 and use the instrument in Equation (8). Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

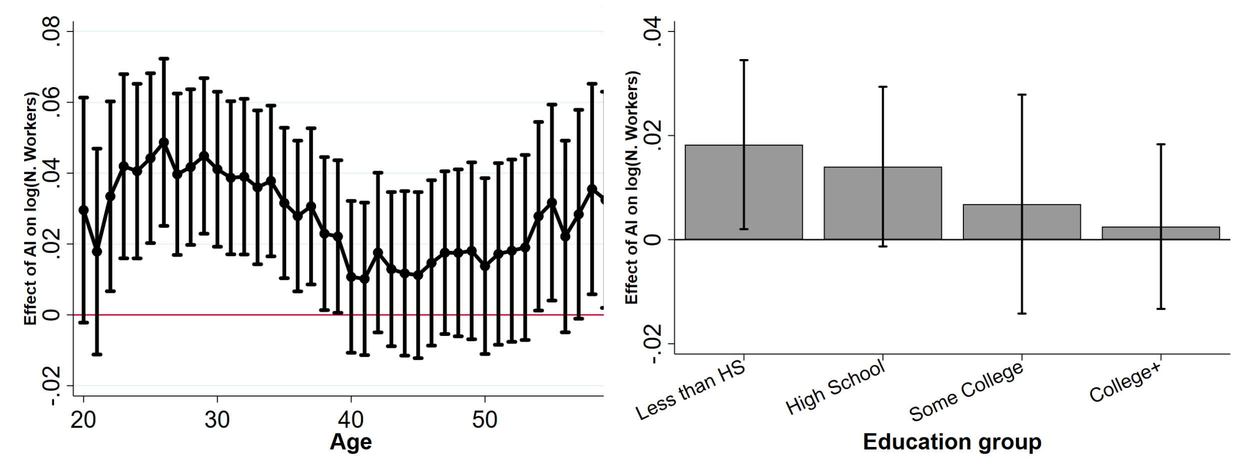
AI leads to net employment gains. Table 2 reports the effect of AI on employment under different controls. AI exposure raises employment in every specification. In the baseline specification in Column 2, a one-standard-deviation increase in AI exposure raises employment by 2.2%. Table A15 in the Appendix decomposes this net gain: AI increases both hiring and separations, but hiring rises more, leading to a positive net job growth.

This net employment gain is not driven by a particular choice of controls. Relative to the baseline in Column 2, Column 1 removes the year-by-one-digit-occupation fixed effect. Column 3 adds year-by-sector fixed effects, where sector is the primary two-digit industry for each occupation.[30] Column 4 includes year-by-two-digit-occupation fixed effects, and Column 5 combines year-by-sector and year-by-two-digit-occupation fixed effects. The estimated coefficients remain positive and similar in magnitude throughout. The robustness of the estimates to sector-by-year and narrow-occupation-by-year fixed effects further suggests that the results are not driven by labor-market shocks correlated across countries, reinforcing the evidence in Sections 5.5 and 5.4.

The OLS estimate in Table 2 is small and not statistically different from zero. This pattern suggests that AI development in Brazil may target occupations already on declining employment trends, making the 2SLS strategy important for isolating supply-driven variation in AI exposure.

[30]I construct the year-by-sector fixed effect by first calculating, for each six-digit occupation, employment shares across two-digit National Classification of Economic Activities sectors. I then assign each occupation to the sector with the highest share and use this dominant sector to define the sector-by-year fixed effect. Because the occupational classification is narrow, the dominant sector accounts for 63 percent of employment on average.

Figure 6: **Effect of AI on Employment by Age and Education**



(a) Age (b) Education

Notes: This figure reports the effect of AI on employment by worker age and education group. The left panel plots the estimated effect of AI on the log number of workers of a given age, and the right panel plots the corresponding effect by education group. Specifications follow Section 5 and use the instrument in Equation (8). Vertical bars report 90% confidence intervals.

AI shifts employment toward younger, less educated, and inexperienced workers. Table 3 shows that AI reallocates employment toward younger, less educated, and less experienced workers. Column 1 shows a decline in average age, Column 2 a decline in average years of education, and Column 3 a decline in average firm tenure. Figure 6 shows that employment gains from AI are concentrated among workers aged 20 to 39, with no statistically significant effects for other age groups. By education, AI increases employment for workers with less than a high-school diploma, while the estimates for other education groups are not statistically significant.

Table 4: **Effect of AI on Wages and Inequality**

	(1) *log(Wage)*	(2) *Ability*	(3) *Wage Residual*	(4) *log(P90/P10 Wage)*
2SLS				
AI Exposure	0.00164	-0.0177^{***}	0.0409^{***}	-0.0176^{**}
	(0.00336)	(0.00658)	(0.0158)	(0.00851)
N	50,341	19,816	19,868	49,592

Notes: This table reports 2SLS estimates for log wages, average ability, wage residuals, and within-occupation wage inequality. Ability and wage residuals are constructed from Mincer regressions; Section C.7 explains in detail how these variables were calculated. Specifications follow Section 5 and use the instrument in Equation (8). Standard errors, clustered at the occupation level, appear in parentheses. $^*p < 0.10,^{**}p < 0.05,^{***}p < 0.01$.

AI has no effect on average wages due to the entry of lower-ability workers. Column 1 of Table 4 shows that AI exposure has no significant effect on wages. To test whether this effect reflects composition rather than lower pay for existing workers, I decompose the effect into composition and price components. I estimate a Mincer regression relating individual wages to observable characteristics and worker fixed effects, then aggregate the worker fixed effects (which I call ability) and wage residuals to the occupation level.[31] Because ability is fixed at the individual level, changes in average ability capture shifts in workforce composition, while changes in the average wage residual capture wage changes unexplained by individual characteristics. To facilitate interpretation, I normalize both variables to have standard deviation of one.

This decomposition confirms the composition mechanism. Column 2 of Table 4 shows that AI lowers average ability, while Column 3 shows that AI raises wages conditional on worker characteristics, likely reflecting higher labor demand. The null effect on average wages therefore comes from compositional change: AI draws in workers who earn less based on both observable and unobservable characteristics. In the model presented in Section 2, this pattern is consistent with AI complementing low-ability workers.[32]

This increased demand for low-ability workers also compresses wage distributions within occupations. Column 4 of Table 4 shows that a one-standard-deviation increase in AI exposure lowers the ratio of the 90th decile to the 10th decile of wages by 1.8%. As AI raises relative demand for workers at the bottom of the wage distribution, it narrows wage gaps within occupations.

Table 5: **AI Reduces Inequality Across Occupations**

	(1) *log(Yrs. Educ.)*	(2) *log(Yrs. Educ.)*	(3) *log(Wage)*	(4) *log(Wage)*
AI Exposure	0.0283***	0.0634***	0.124***	0.114***
	(0.00348)	(0.00603)	(0.0169)	(0.0220)
AI Exp. $\times$ *log*($Wage_{t=03}$)	−0.00392***		−0.0159***	
	(0.000406)		(0.00203)	
AI Exp. $\times$ *log*($Educ._{t=03}$)		−0.0258***		−0.0445***
		(0.00234)		(0.00816)
N	46,679	46,679	46,679	46,679

Notes: This table reports 2SLS estimates allowing AI's effect to vary with occupation characteristics measured in 2003. Columns 1–2 use log average years of education; Columns 3–4 use log average monthly earnings. AI exposure and each interaction are instrumented with the AI ease-of-development instrument and the corresponding interaction. Specifications follow Section 5. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

AI decreases inequality across occupations. Table 5 allows the effect of AI on education and wages to vary with two occupation characteristics: initial wages and initial education. The estimates show that AI reduces education and wages more in occupations

[31] The ability measure uses the linked RAIS panel. Administrative lags in this dataset limit coverage to the period through 2016, so these results capture only the beginning of the AI boom. Appendix C.7 describes the Mincer regression and estimation procedure.

[32] This conclusion is also supported by experimental evidence showing that AI leads to larger productivity gains for lower-skill workers than for experts (Kanazawa et al. 2022, Brynjolfsson et al. 2025, Gruber et al. 2020, Choi et al. 2023, Dell'Acqua et al. 2023, Noy and Zhang 2023, Peng et al. 2023).

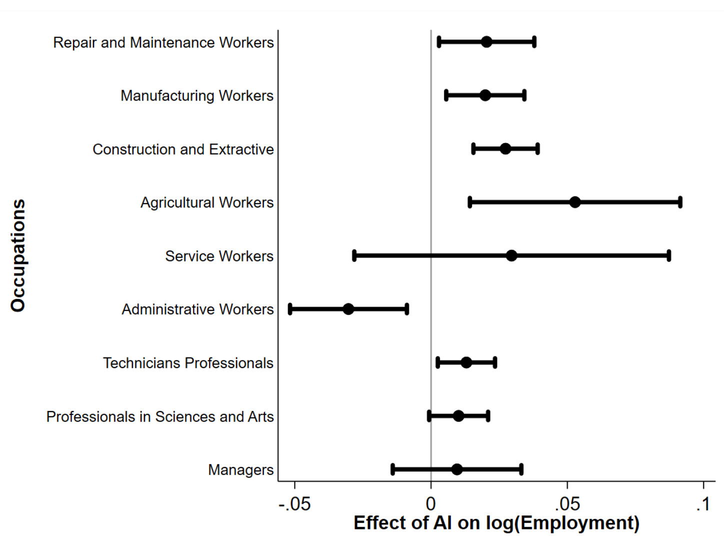
with higher initial wages and higher initial education. This pattern is consistent with AI lowering barriers to entry into more skilled occupations (Autor and Thompson 2025, Autor 2024).

6.2 AI in the Office and the Factory

To understand the mechanism behind these results, I study heterogeneity in the effect of AI across occupations. I find that AI reduces employment in administrative and data-processing tasks, but increases employment in production tasks that use information to guide decisions. This pattern supports a mechanism in which AI replaces information processing while complementing workers who use information to act.

AI replaces workers in the office but increases employment in the factory. Figure 7 estimates the effect of AI separately for each one-digit CBO occupational group.[33] It shows that AI increases employment by about 3.4% in production-related occupations, including agriculture, extraction, manufacturing, maintenance, and technical occupations. In contrast, it reduces employment in administrative occupations by 2.6%. Employment changes little among managers and high-skilled professionals, such as engineers and chemists.

Figure 7: **Differential Effect of AI on Employment Across Occupations**



Notes: This figure reports estimates of the effect of AI on employment for each one-digit occupational group. The model is: $y_{o,t} = \beta_{O(o)} AI\ Exposure_{o,t} + \mu_o + \mu_t + X'_{o,t}\theta + \epsilon_{o,t}$, where $O(o)$ denotes the broad occupational category of occupation o. All specifications use the AI ease-of-development instrument defined in Equation (8). In the first stage, the instrument is also interacted with a broad occupational group dummy. Standard errors are clustered at the occupation level. Bands represent 95% confidence intervals.

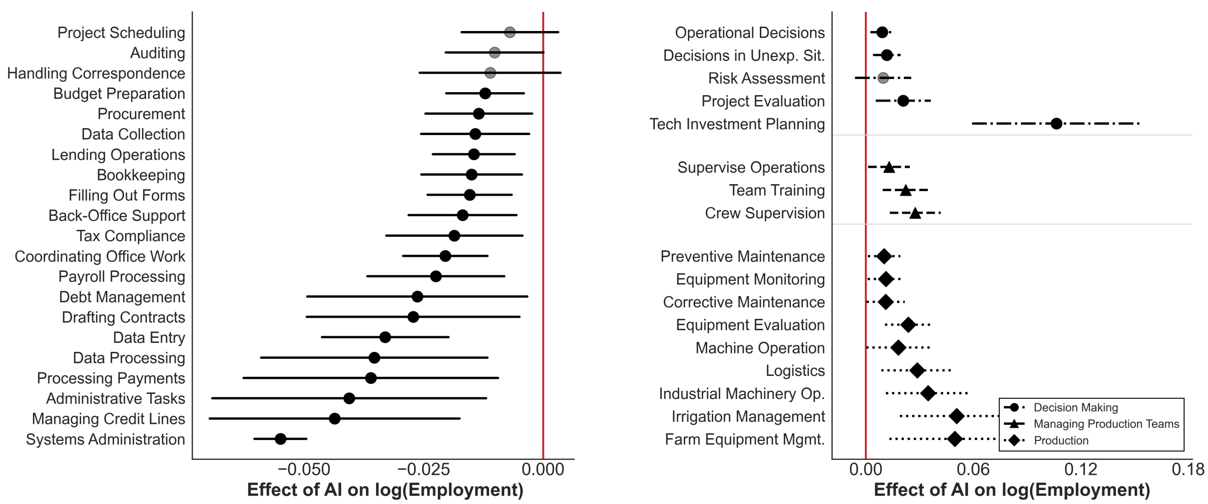
[33] To construct the figure, I allow the coefficient of interest to vary by one-digit CBO occupational group: $y_{o,t} = \beta_{O(o)} AI\ Exposure_{o,t} + \mu_o + \mu_t + X'_{o,t}\theta + \epsilon_{o,t}$, where $O(o)$ denotes the broad occupational group of occupation o.

AI decreases employment of workers doing data-processing tasks. The contrast between administrative and production occupations raises a more granular question: which tasks account for these employment effects? To answer this, I group the 49,660 task descriptions in the Brazilian occupational classification into 700 semantically similar task groups, as described in Appendix C.8. I then estimate whether the effect of AI differs between occupations that do and do not perform tasks in each group. For each task group g, I estimate

$$y_{o,t} = \beta AI\ Exposure_{o,t} + \beta_g \mathbb{I}_o\{o\ performs\ task\ g\} \times AI\ Exposure_{o,t} + \mu_o + \mu_t + X'_{o,t}\theta + \epsilon_{o,t},$$

where $\mathbb{I}_o\{o\ performs\ task\ g\}$ equals one if occupation o contains at least one task in group g. I instrument AI exposure and its interaction with the task-group dummy using the AI ease-of-development instrument and its interaction with the same dummy.

Figure 8: **AI Replaces Data Processing and Expands Decision Making Tasks**



(a) Data-processing tasks

(b) Decision and production tasks

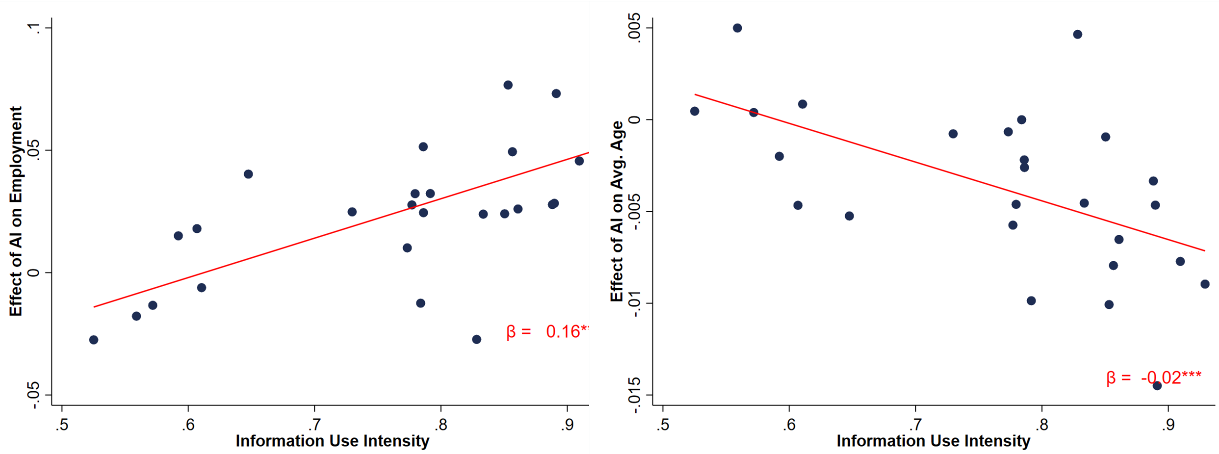
Notes: This figure reports task-group heterogeneity in AI's effect on log employment. Each point is an instrumented interaction between AI exposure and an indicator that the occupation performs the task group. Task groups are constructed using the procedure in Appendix C.8. Horizontal bars report 95 percent confidence intervals.

The task-level heterogeneity identifies the tasks behind the employment losses in administrative occupations. Figure 8a plots the differential effect of AI, β_g, for the task groups with the largest negative effects: system administration, credit-line management, administrative tasks, data processing, data entry, form completion, tax compliance, and data collection. In occupations that perform these tasks, a one-standard-deviation increase in AI exposure reduces employment by 3 to 8 percentage points more than in occupations that do not.

These results show that AI reduces employment most strongly in occupations centered on recording, processing, and organizing information.

AI increases employment in decision-making, production-supervision, and machine-operation tasks. In contrast, the positive employment effects are concentrated in tasks that use information to act. Figure 8b shows that the largest gains occur in task groups related to decision making, production supervision, and machine operation. These are tasks in which better information helps workers respond to changing conditions, coordinate production, and operate equipment.

Figure 9: **AI Automates Information Processing but Expands Information Use**



(a) Employment and Information Use Intensity

(b) Age and Information Use Intensity

Notes: The figure plots the estimated effect of AI exposure on employment at the broad occupational level against an occupation's relative information use intensity. Each point corresponds to one of 20 broad occupational groups, narrower than the 1-digit level. The estimated effects are obtained from $y_{o,t} = \beta_{O(o)}\, \textit{AI Exposure}_{o,t} + \mu_o + \mu_t + X'_{o,t}\theta + \epsilon_{o,t}$, where $O(o)$ denotes the broad occupational group containing occupation o, using the instrument in Equation (8). The relative information use intensity for broad occupation O is defined as $\textit{Relative Information Use Intensity}_O =$ *shr. of tasks that use information/shr. of tasks that use or process information.*

AI complements decision making and replaces data processing. These patterns are consistent with the view of AI as a prediction technology (Agrawal et al. 2019b). Better prediction complements workers who use operational information to decide what to do next, such as supervisors, maintenance technicians, and machine operators, but substitutes for labor in tasks that process, organize, or generate information. To test this mechanism directly, I follow de Souza and Li (2023) and classify tasks according to whether they primarily process data or use information to make decisions. I then define each occupation's information use intensity as the share of its tasks classified as information use. Higher values identify occupations that use operational information intensively; lower values identify occupations specialized in information processing and recordkeeping.[34]

Figure 9a provides the first test of this mechanism. It plots the estimated effect of AI on employment against average information use intensity for each two-digit occupational group. The relationship is strongly positive: AI increases employment where workers use information to guide action, but reduces employment where tasks center on processing data.

Table 6: **AI Has Larger Employment Effects in Information-Use Occupations**

	(1) *log(Employment)*	(2) *log(Age)*	(3) *log(Yrs. Education)*	(4) *log(Wage)*
AI Exposure	-0.0463^{***}	0.00143	0.00238^{*}	0.00974^{*}
	(0.0104)	(0.00160)	(0.00136)	(0.00507)
AI Exposure × Inf. Use	0.0921^{***}	-0.00622^{***}	-0.00610^{***}	-0.0109^{*}
	(0.0131)	(0.00186)	(0.00150)	(0.00600)
N	50,364	50,364	50,364	50,364

Notes: This table reports 2SLS estimates allowing AI's effect to vary with information use intensity, defined as the share of tasks classified as information use rather than information processing. AI exposure and its interaction with information use are instrumented with the corresponding ease-of-development measures. Specifications follow Section 5. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

AI allows low-skilled workers to perform decision-making tasks. Table 6 estimates how the effects of AI on employment and worker composition vary with information use intensity. AI reduces employment in occupations intensive in data processing, but raises employment where workers use information intensively. These gains come from the entry of younger and lower-paid workers, reducing average age, education, and wages in those occupations. This pattern is consistent with AI raising the productivity of less-experienced workers by providing operational guidance in tasks that require decision making.

Figure 9b allows us to visualize the effect of AI on the composition of young workers in each occupation. It plots the effect of AI on age for each two-digit occupational group against the group's average information use intensity. AI leads to the entry of young workers among occupations intensive in data use for decision making but replaces them in occupations intensive in data processing.

Standard AI measures understate AI use in production. The contrast between office and production work shows why different measures of AI exposure can lead to different conclusions. Table 7 repeats the employment analysis using standard off-the-shelf AI exposure measures from the literature. For each measure, I follow the standard approach and interact occupation-level AI exposure with the aggregate stock of AI software registrations. These measures deliver lower, generally negative employment estimates. The correlation columns show why: the standard exposure measures are only weakly correlated with the

[34] Appendix C.9 describes in detail the classification process.

instrument, and this correlation is larger for administrative occupations than for production occupations. Since AI reduces employment in administrative tasks but raises employment in production tasks, standard exposure measures place more weight on the displacement margin and understate the production gains. This pattern helps explain why studies based on different AI measures reach different conclusions about AI's labor-market effects.

Table 7: **Effect of Alternative AI Exposure Measures on Employment**

		Correlation with Instrument		
Exposure Measure	(1) *Effect on log Employment*	(2) *Total*	(3) *Adm.*	(4) *Production*
Baseline	0.0217^{***} (0.00708)	0.40	0.28	0.34
AI impact score (Tolan et al. 2021)	-0.0216^{***} (0.0056)	0.02	0.02	0.00
Suitability for machine learning (Brynjolfsson et al. 2018)	-0.0023 (0.0057)	0.01	0.12	0.02
LLM task exposure (Eloundou et al. 2023)	-0.0230^{***} (0.0084)	-0.02	0.04	0.03
AI capability exposure (Felten et al. 2021)	-0.0301^{***} (0.0090)	0.01	0.14	0.05

Notes: This table reports estimates of the effect of alternative AI exposure measures on log employment. The employment-effect column uses the matched-control specifications, except for the baseline row, which reports the baseline estimate from Table 2. The correlation columns report correlations with the AI ease-of-development instrument after residualizing by the baseline controls. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10, ^{**}p < 0.05, ^{***}p < 0.01$.

Is generative AI different? Table A16, in the Appendix, separates the effects of generative and predictive AI exposure. Predictive AI, which corresponds to 99% of AI software registrations, closely matches the average effects: it increases employment and reduces average age and experience. However, due to the high correlation between the two metrics, standard errors are larger and most of these effects are not significant. Generative AI has a negative effect on employment and increases average age and experience. Because 81% of generative AI software is used in management applications, generative AI has a stronger effect on replacing workers who perform data-processing tasks.

7 Robustness

In Section 6, I showed that AI increases employment while shifting employment toward younger, less-educated, and less-experienced workers. I now show that these results are robust to alternative specifications, exposure measures, and instruments.

7.1 Alternative Specifications

Control for U.S. labor-market outcomes. One concern is that U.S. programming-language trends may capture global labor-market shocks that also affect Brazil. Panel A of Table A17 addresses this concern by controlling for the same outcome measured in the

United States. The results are still the same.

Occupation-industry regressions controlling for sectoral shocks. Panel B of Table A17 re-estimates the baseline specification at the occupation-industry level, following Webb (2020). I include occupation-industry fixed effects and industry-year fixed effects to absorb sector-specific shocks. The estimates remain close to the baseline results.

Additional outcome lags. Panel C of Table A17 adds two lags of the dependent variable to absorb differential outcome dynamics across occupations. The estimates remain similar to the baseline.

Regression without controls. Panel D of Table A17 removes all controls except for occupation fixed effects and one-digit-occupation-by-year fixed effects. The estimates remain close to the baseline, showing that the main results are not driven by these controls.

7.2 Alternative AI Exposure Measures

The baseline exposure measure weights software registrations by software value. I test whether this choice matters using two alternative exposure measures: one that weights all registrations equally and one that weights registrations by breakthrough importance. Table A18 shows that the main results are unchanged under both alternatives.

Equally Weighted AI Exposure. I first set $v_s = 1$ for every software registration in Equation 5 and apply the same equal weights to the instrument in Equation 8. Panel A shows that AI still increases employment, shifts employment toward lower-skilled workers, and reduces within-occupation wage inequality.

Breakthrough AI Exposure. I next weight software by its importance for later AI development. Following Kelly et al. (2021), I construct a breakthrough score that is higher for software that is novel relative to prior AI software but similar to later AI software; Appendix C.11 describes its construction. Panel B shows that the results are robust to this alternative weighting.

7.3 Alternative Instruments

Shift-share instrument. I first use a standard shift-share instrument that interacts each occupation's baseline similarity to AI software developed in 2000–2003 with the total number of AI software applications developed each year. Panel 1 of Table A19 shows that the estimates point in the same direction as the baseline: AI increases employment, shifts employment toward younger and less-educated workers, and has no systematic effect on average wages.

Number of packages. I next instrument AI exposure using AI package availability. Using the ecosyste.ms package-registry data, I count newly released packages whose package or repository metadata contain AI terms by programming language and year, then aggregate these counts to occupations using the baseline occupation-language exposure weights. This instrument captures the idea that AI packages lower development costs in the languages where they are available.[35] Panel 2 of Table A19 shows the same qualitative pattern as the baseline.

Stack Overflow AI instrument. The third alternative uses only AI-related Stack Overflow posts, excluding the mailing-list component of the baseline instrument. This instrument exploits the fact that Stack Overflow reduced search, debugging, and implementation costs for programmers (Mamykina et al. 2011, Treude and Robillard 2016, Wu et al. 2019, Ponzanelli et al. 2014). Panel 3 of Table A19 shows that the conclusions remain unchanged. Together, the three alternative instruments show that the main results do not depend on the specific construction of the AI ease-of-development instrument.

8 Conclusion

Using an administrative dataset with representative coverage of AI software in Brazil, I found that, on average, AI increases employment without any effect on wages, driven by rising employment among younger, less-educated, and less-experienced workers. Within occupations, AI reduces wage inequality by benefiting less skilled workers. Across occupations, however, the effects diverge: AI displaces administrative workers while expanding employment in production roles. These findings help reconcile competing views in the literature—confirming displacement effects in some settings while also presenting evidence that AI can complement labor and increase employment.

This paper leaves several open questions for future research. Where do the low-skilled workers who gain employment through AI adoption come from? Given their age, AI may draw high-school–age workers into the labor market. Earlier entry could affect human capital accumulation and long-run growth. Another open question concerns administrative workers displaced by AI. What becomes of these workers, and how likely are they to transition into jobs that use AI? Answering these questions is essential for policy design and for understanding the long-run effects of AI.

[35] This instrument exploits narrower variation than the baseline instrument because package data can be matched to only 12 programming languages, while the baseline mailing-list and Stack Overflow instrument covers 595 programming languages.

References

Acemoglu, D., D. Autor, J. Hazell, and P. Restrepo (2022): "Artificial Intelligence and Jobs: Evidence from Online Vacancies," *Journal of Labor Economics*, 40, 293–340.

Acemoglu, D. and P. Restrepo (2018): "Artificial Intelligence, Automation, and Work," in *The Economics of Artificial Intelligence: An Agenda*, National Bureau of Economic Research, Inc, NBER Chapters, 197–236.

——— (2020): "Robots and Jobs: Evidence from US Labor Markets," *Journal of Political Economy*, 128.

Adams, J., M. Fang, Z. Liu, and Y. Wang (2024): "The Rise of AI Pricing: Trends, Driving Forces, and Implications for Firm Performance," Working Paper Series 2024-33, Federal Reserve Bank of San Francisco.

Aghion, P., S. Bunel, X. Jaravel, T. Mikaelsen, A. Roulet, and J. Søgaard (2025): "How Different Uses of AI Shape Labor Demand: Evidence from France," *AEA Papers and Proceedings*, 115, 62–67.

Agoro, H. (2025): "Reducing Downtime in Production Lines Through Proactive Maintenance Strategies," .

Agrawal, A., J. Gans, and A. Goldfarb (2019a): "Prediction, Judgment, and Complexity: A Theory of Decision-Making and Artificial Intelligence," in *The Economics of Artificial Intelligence: An Agenda*, University of Chicago Press.

Agrawal, A., J. S. Gans, and A. Goldfarb (2019b): "Artificial Intelligence: The Ambiguous Labor Market Impact of Automating Prediction," NBER Working Papers 25619, National Bureau of Economic Research, Inc.

——— (2019c): "Exploring the impact of artificial Intelligence: Prediction versus judgment," *Information Economics and Policy*, 47, 1–6, the Economics of Artificial Intelligence and Machine Learning.

Alam, M. F., A. Lentsch, N. Yu, S. Barmack, S. Kim, D. Acemoglu, J. Hart, S. Johnson, and F. Ahmed (2024): "From Automation to Augmentation: Redefining Engineering Design and Manufacturing in the Age of NextGen-AI," Working paper, Massachusetts Institute of Technology, Cambridge, MA.

Alaveras, G. and B. Martens (2015): "International Trade in Online Services," JRC Working Papers on Digital Economy 2015-08, Joint Research Centre.

Alvarez-Cuadrado, F., N. V. Long, and M. Poschke (2017): "Capital-Labor Substitution, Structural Change, and the Labor Income Share," *Theoretical Economics*, 12, 1229–1266.

Arora, A. and A. Gambardella (2005): "The Globalization of the Software Industry: Perspectives and Opportunities for Developed and Developing Countries," in *Innovation Policy and the Economy, Volume 5*, National Bureau of Economic Research, Inc, NBER Chapters, 1–32.

Associação Brasileira das Empresas de Software (ABES) (2024): "Por que registrar meu software?" Accessed: 2024-11-14.

Autor, D. (2024): "Applying AI to Rebuild Middle Class Jobs," Working Paper 32140, National Bureau of Economic Research.

Autor, D., C. Chin, A. Salomons, and B. Seegmiller (2024): "New Frontiers: The Origins and Content of New Work, 1940–2018*," *The Quarterly Journal of Economics*, 139, 1399–1465.

Autor, D. and A. Salomons (2018): "Is Automation Labor Share–Displacing? Productivity Growth, Employment, and the Labor Share," *Brookings Papers on Economic Activity*, 1–63.

Autor, D. and N. Thompson (2025): "Expertise," NBER Working Papers 33941, National Bureau of Economic Research, Inc.

Autor, D. H., F. Levy, and R. J. Murnane (2003): “The Skill Content of Recent Technological Change: An Empirical Exploration,” *The Quarterly Journal of Economics*, 118, 1279–1333.

Babina, T., A. Fedyk, A. He, and J. Hodson (2024): “Artificial intelligence, firm growth, and product innovation,” *Journal of Financial Economics*, 151, 103745.

——— (2025): “Firm Investments in Artificial Intelligence Technologies and Changes in Workforce Composition,” in *Technology, Productivity, and Economic Growth*, University of Chicago Press.

Barbosa, F., A. Souza, A. Souza, P. Nascimento, and A. Baganha (2022): “Análise Exploratória dos Registros de Software do Instituto Nacional da Propriedade Industrial (INPI) de 2018 a 2020,” *Conjecturas*, 22, 765–777.

Barth, E., J. C. Davis, R. B. Freeman, and K. McElheran (2023): “Twisting the demand curve: Digitalization and the older workforce,” *Journal of Econometrics*, 233, 443–467.

Bekes, G., A. Bonfiglioli, R. Crino, and G. Gancia (2025): “Integrators and Robot Adoption: Facts from Hungary,” Discussion Paper 20026, Centre for Economic Policy Research.

Benhanifia, A., Z. B. Cheikh, P. M. Oliveira, A. Valente, and J. Lima (2025): “Systematic review of predictive maintenance practices in the manufacturing sector,” *Intelligent Systems with Applications*, 26, 200501.

Benz, S. and A. Jaax (2022): “The costs of regulatory barriers to trade in services: New estimates of ad valorem tariff equivalents,” *Economics Letters*, 212, 110057.

Blanas, S., G. Gancia, and S. Y. Lee (2020): “Who Is Afraid of Machines?” *Economic Policy*, 34, 627–690.

Bonfiglioli, A., R. Crino, H. Fadinger, and G. Gancia (2024): “Robot Imports and Firm-Level Outcomes,” *The Economic Journal*, 134, 3428–3444.

Bonfiglioli, A., R. Crinò, G. Gancia, and I. Papadakis (2023): “Artificial Intelligence and Jobs: Evidence from US Commuting Zones,” Tech. rep.

Bonfiglioli, A. and G. Gancia (2008): “North-South Trade and Directed Technical Change,” *Journal of International Economics*, 76, 276–295.

Borboni, A., K. V. V. Reddy, I. Elamvazuthi, M. S. Al-Quraishi, E. Natarajan, and S. S. A. Ali (2023): “The Expanding Role of Artificial Intelligence in Collaborative Robots for Industrial Applications: A Systematic Review of Recent Works,” *Machines*, 11, 111.

Boustan, L. P., J. Choi, and D. Clingingsmith (2022): “Automation After the Assembly Line: Computerized Machine Tools, Employment and Productivity in the United States,” Working Paper 30400, National Bureau of Economic Research.

Bresnahan, T. (2021): *Artificial Intelligence Technologies and Aggregate Growth Prospects*, Cambridge University Press, 132–170.

Brynjolfsson, E., W. Jin, and K. McElheran (2021): “The power of prediction: predictive analytics, workplace complements, and business performance,” *Business Economics*, 56, 217–239.

Brynjolfsson, E., D. Li, and L. Raymond (2025): “Generative AI at Work*,” *The Quarterly Journal of Economics*, 140, 889–942.

Brynjolfsson, E. and K. McElheran (2016): “The Rapid Adoption of Data-Driven Decision-Making,” *American Economic Review*, 106, 133–39.

——— (2019): “Data in Action: Data-Driven Decision Making and Predictive Analytics in U.S. Manufacturing,” Working Paper 3422397, Rotman School of Management, University of Toronto, date written: July 6, 2019.

Brynjolfsson, E., T. Mitchell, and D. Rock (2018): “What Can Machines Learn, and What Does It Mean for Occupations and the Economy?” *AEA Papers and Proceedings*, 108, 43–47.

Bühlmann, H. (1967): “Experience Rating and Credibility,” *ASTIN Bulletin*, 4, 199–207.

Campello, R. J. G. B., D. Moulavi, and J. Sander (2013): “Density-Based Clustering Based on Hierarchical Density Estimates," in *Advances in Knowledge Discovery and Data Mining*, Springer, 160–172.

Campello, R. J. G. B., D. Moulavi, A. Zimek, and J. Sander (2015): "Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection," *ACM Transactions on Knowledge Discovery from Data*, 10, 1–51.

Capgemini (2020): "Artificial Intelligence (AI)-Driven Smart Factory Solution Provides Operators and Engineers with a New Level of Insight and the Ability to Adjust Production at a Moment's Notice," Tech. rep.

Caunedo, J., E. Keller, and Y. Shin (2025): "Technology and the Task Content of Jobs across the Development Spectrum," *The World Bank Economic Review*, 39, 86–111.

Chandar, B., J. Hawkins, J. Horton, H. Liu, F. Narita, I. O'Bannon, G. Varriale, and P. Yildirim (2025): "Canaries in the Coal Mine? Six Facts about the Recent Employment Effects of Artificial Intelligence," Stanford Digital Economy Lab research brief.

Chauvet, J.-M. (2018): "The 30-Year Cycle In The AI Debate," .

Chen, R.-s., Y.-s. Tsai, and C.-c. Chang (2006): "Design and Implementation of an Intelligent Manufacturing Execution System for Semiconductor Manufacturing Industry," in *2006 IEEE International Symposium on Industrial Electronics*, vol. 4, 2948–2953.

Choi, J. H., D. Schwarcz, and K. E. Yeh (2023): "AI Assistance in Legal Analysis: An Empirical Study," Legal Studies Research Paper 23-22, University of Minnesota Law School.

Consoli (2025): "Inteligência artificial e termovisão reforçam eficiência da Copel e reduzem quedas de energia no Paraná," https://aerp.org.br/redeaerp/inteligencia-artificial-e-termovisao-reforcam-eficiencia-da-copel-e-reduzem-quedas-de-energia-no-parana/, publicado por Redação, com texto de Flávia Consoli — Associação das Emissoras de Radiodifusão do Paraná.

Cortes, G. M., N. Jaimovich, C. J. Nekarda, and H. E. Siu (2020): "The Dynamics of Disappearing Routine Jobs: A Flows Approach," *Labour Economics*, 65, 101823.

Danieli, O. (2026): "Skill-Replacing Technology and Bottom-Half Inequality," Working paper, Tel Aviv University; CEPR; IZA, previously titled "Revisiting U.S. Wage Inequality at the Bottom 50%".

de Souza, G. (2022): "The Labor Market Consequences of Appropriate Technology," Working Paper Series WP 2022-53, Federal Reserve Bank of Chicago.

de Souza, G., J. S. Herbstman, and J. Mannion (2024): "How Demand for New Skills Affects Wage Inequality: The Case of Software Programmers," Working Paper Series WP 2024-19, Federal Reserve Bank of Chicago.

de Souza, G. and H. Li (2023): "Robots, Tools, and Jobs: Evidence from Brazilian Labor Markets," Working Paper Series WP 2023-42, Federal Reserve Bank of Chicago.

Dell'Acqua, F., E. McFowland III, E. Mollick, H. Lifshitz-Assaf, K. C. Kellogg, S. Rajendran, L. Krayer, F. Candelon, and K. R. Lakhani (2023): "Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality," Working Paper 24-013, Harvard Business School, Boston, MA.

Delorey, D., C. Knutson, and S. Chun (2007): "Do Programming Languages Affect Productivity? A Case Study Using Data from Open Source Projects," 8–8.

Eden, M. and P. Gaggl (2018): "On the Welfare Implications of Automation," *Review of Economic Dynamics*, 29, 15–43.

——— (2020): "Do Poor Countries Really Need More IT?" *The World Bank Economic Review*, 34, 48–62.

Eloundou, T., S. Manning, P. Mishkin, and D. Rock (2023): "GPTs are GPTs: An Early Look at the Labor Market Impact Potential of Large Language Models," .

Elsafty, A., M. Shaaban, A. El-Naggar, M. Okasha, and M. Elnaggar (2024): "AI-Driven Quality Management System for Sustainable Construction Projects: A Decision Support Model," *Energy Efficiency*.

Evans, R. and J. Gao (2016): "DeepMind AI Reduces Google Data Centre Cooling Bill by 40%," Accessed: 2025-05-08.

Ezell, S. J., R. D. Atkinson, and M. A. Wein (2013): "Localization Barriers to Trade: Threat to the Global Innovation Economy," Technical Report –, Information Technology & Innovation Foundation, Washington, DC, available at `https://www2.itif.org/2013-localization-barriers-to-trade.pdf`.

Felten, E., M. Raj, and R. Seamans (2021): "Occupational, industry, and geographic exposure to artificial intelligence: A novel dataset and its potential uses," *Strategic Management Journal*, 42, 2195–2217.

Felten, E. W., M. Raj, and R. Seamans (2018): "A Method to Link Advances in Artificial Intelligence to Occupational Abilities," *AEA Papers and Proceedings*, 108, 54–57.

Forootan, M. M., I. Larki, R. Zahedi, and A. Ahmadi (2022): "Machine Learning and Deep Learning in Energy Systems: A Review," *Sustainability*, 14, 4832.

Frey, C. B. and M. A. Osborne (2017): "The future of employment: How susceptible are jobs to computerisation?" *Technological Forecasting and Social Change*, 114, 254–280.

Gaggl, P. and G. C. Wright (2017): "A Short-Run View of What Computers Do: Evidence from a UK Tax Incentive," *American Economic Journal: Applied Economics*, 9, 262–294.

Ghelani, H. et al. (2024): "AI-Driven Quality Control in PCB Manufacturing: Enhancing Production Efficiency and Precision," *ResearchGate Preprint*, accessed: 2025-05-07.

GitHub Staff (2024): "Octoverse: AI leads Python to top language as the number of global developers surges," GitHub Blog.

Gmyrek, P., J. Berg, and D. Bescond (2023): "Generative AI and jobs a global analysis of potential effects on job quantity and quality," Tech. rep.

Goldsmith-Pinkham, P., I. Sorkin, and H. Swift (2020): "Bartik Instruments: What, When, Why, and How," *American Economic Review*, 110, 2586–2624.

Gollin, D., D. Lagakos, and M. E. Waugh (2014): "The Agricultural Productivity Gap," *The Quarterly Journal of Economics*, 129, 939–993.

Goos, M., A. Manning, and A. Salomons (2014): "Explaining Job Polarization: Routine-Biased Technological Change and Offshoring," *American Economic Review*, 104, 2509–2526.

Graetz, G. and G. Michaels (2018): "Robots at Work," *The Review of Economics and Statistics*, 100, 753–768.

Grootendorst, M. (2022): "BERTopic: Neural topic modeling with a class-based TF-IDF procedure," .

Gruber, J., B. R. Handel, S. H. Kina, and J. T. Kolstad (2020): "Managing Intelligence: Skilled Experts and AI in Markets for Complex Products," NBER Working Papers 27038, National Bureau of Economic Research, Inc.

Hampole, M., D. Papanikolaou, L. D. Schmidt, and B. Seegmiller (2025): "Artificial Intelligence and the Labor Market," NBER Working Papers 33509, National Bureau of Economic Research, Inc.

Harms, P. and D. Shuvalova (2020): "Cultural distance and international trade in services: A disaggregate view," *Economic Systems*, 44, 100786.

Hartman, T. (2015): "COBOL Blues," `https://www.reuters.com/graphics/USA-BANKS-COBOL/`

010040KH18J/, accessed: 16 Feb 2025.

Hémous, D., M. Olsen, C. Zanella, and A. Dechezleprêtre (2025): "Induced Automation Innovation: Evidence from Firm-Level Patent Data," *Journal of Political Economy*, 133, 1975–2028.

Hosseini, S. M. and G. Lichtinger (2025): "Generative AI as Seniority-Biased Technological Change: Evidence from U.S. Résumé and Job Posting Data," SSRN Working Paper No. 5425555.

Hubmer, J. and P. Restrepo (2021): "Not a Typical Firm: The Joint Dynamics of Firms, Labor Shares, and Capital–Labor Substitution," Working Paper 28579, National Bureau of Economic Research.

Hui, X., O. Reshef, and L. Zhou (2024): "The Short-Term Effects of Generative Artificial Intelligence on Employment: Evidence from an Online Labor Market," *Organization Science*, earlier version: CESifo Working Paper No. 10601.

Humlum, A. (2019): "Robot Adoption and Labor Market Dynamics," Job Market Paper, Princeton University.

Humlum, A. and E. Vestergaard (2025a): "Large Language Models, Small Labor Market Effects," Working Paper 33777, National Bureau of Economic Research.

——— (2025b): "The unequal adoption of ChatGPT exacerbates existing inequalities among workers," *Proceedings of the National Academy of Sciences*, 122, 2414972121–.

——— (2026): "Still Waters, Rapid Currents: Early Labor Market Transformation under Generative AI," Discussion paper, Rockwool Foundation Berlin.

Jaravel, X., P. Aghion, C. Antonin, and S. Bunel (2020): "What Are the Labor and Product Market Effects of Automation? New Evidence from France," CEPR Discussion Papers 14443, C.E.P.R. Discussion Papers.

Junqueira Botelho, A. J., G. Stefanuto, and F. Veloso (2005): "The Brazilian Software Industry," in *From Underdogs to Tigers: The Rise and Growth of the Software Industry in Brazil, China, India, Ireland, and Israel*, Oxford University Press.

Kalyani, A. (2020): "The Geography of New Technologies," Working Paper 126, Institute for New Economic Thinking.

Kalyani, A., N. Bloom, V. M. Carvalho, T. A. Hassan, J. Lerner, and A. Tahoun (2025a): "The Diffusion of New Technologies," *The Quarterly Journal of Economics*.

Kalyani, A., T. Hassan, and P. Restrepo (2025b): "New Technologies and Inequality," Working paper, Federal Reserve Bank of St. Louis, available at author's website.

Kanazawa, K., D. Kawaguchi, H. Shigeoka, and Y. Watanabe (2022): "AI, Skill, and Productivity: The Case of Taxi Drivers," CIRJE F-Series CIRJE-F-1202, CIRJE, Faculty of Economics, University of Tokyo.

Kaplan, J., S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei (2020): "Scaling Laws for Neural Language Models," .

Keleko, A. T., B. Kamsu-Foguem, R. H. Ngouna, and A. Tongne (2022): "Artificial intelligence and real-time predictive maintenance in industry 4.0: a bibliometric analysis," *AI and Ethics*, 2, 553–577.

Kelly, B., D. Papanikolaou, A. Seru, and M. Taddy (2021): "Measuring Technological Innovation over the Long Run," *American Economic Review: Insights*, 3, 303–320.

Kogan, L., D. Papanikolaou, L. D. Schmidt, and B. Seegmiller (2023): "Technology and Labor Displacement: Evidence from Linking Patents with Worker-Level Data," Working Paper 31846, National Bureau of Economic Research.

Kogan, L., D. Papanikolaou, A. Seru, and N. Stoffman (2017): "Technological Innovation, Resource Allocation, and Growth," *The Quarterly Journal of Economics*, 132, 665–712.

Krein, J., A. Maclean, C. Knutson, D. Delorey, and D. Eggett (2010): "Impact of Programming Language Fragmentation on Developer Productivity: A Sourceforge Empirical Study," *IJOSSP*, 2, 41–61.

Leduc, S. and Z. Liu (2024): "Automation, Bargaining Power, and Labor Market Fluctuations," *American Economic Journal: Macroeconomics*, 16, 226–267.

Liu, J., Y. Qian, Y. Yang, and Z. Yang (2022): "Can Artificial Intelligence Improve the Energy Efficiency of Manufacturing Companies? Evidence from China," *International Journal of Environmental Research and Public Health*, 19, 2091.

Mamykina, L., B. Manoim, M. Mittal, G. Hripcsak, and B. Hartmann (2011): "Design Lessons from the Fastest Q&A Site in the West," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ACM, CHI '11, 2857–2866.

Marr, B. and M. Ward (2019): *Artificial Intelligence in Practice: How 50 Successful Companies Used AI and Machine Learning to Solve Problems*, Hoboken, NJ: John Wiley & Sons.

Martinez, J. (2021): "Putty-Clay Automation," *CEPR Discussion*.

Maslej, N., L. Fattorini, R. Perrault, Y. Gil, V. Parli, N. Kariuki, E. Capstick, A. Reuel, E. Brynjolfsson, J. Etchemendy, K. Ligett, T. Lyons, J. Manyika, J. C. Niebles, Y. Shoham, R. Wald, T. Walsh, A. Hamrah, L. Santarlasci, J. B. Lotufo, A. Rome, A. Shi, and S. Oak (2025): "The AI Index 2025 Annual Report," Annual report, AI Index Steering Committee, Institute for Human-Centered AI, Stanford University, retrieved 9 April 2025 from `https://hai-production.s3.amazonaws.com/files/hai_ai_index_report_2025.pdf`.

McElheran, K., J. F. Li, E. Brynjolfsson, Z. Kroff, E. Dinlersoz, L. S. Foster, and N. Zolas (2023): "AI Adoption in America: Who, What, and Where," Working Paper 31788, National Bureau of Economic Research.

McElheran, K., M.-J. Yang, Z. Kroff, and E. Brynjolfsson (2025): "The Rise of Industrial AI in America: Microfoundations of the Productivity J-curve(s)," Working Papers 25-27, Center for Economic Studies, U.S. Census Bureau.

McInnes, L., J. Healy, N. Saul, and L. Grossberger (2018): "UMAP: Uniform Manifold Approximation and Projection," *Journal of Open Source Software*, 3, 861.

Metz, C. (2021): *Genius Maker: The Engineers Who Created the Intelligence Revolution*, New York: Viking.

Meyerovich, L. A. and A. S. Rabkin (2013): "Empirical analysis of programming language adoption," in *Proceedings of the 2013 ACM SIGPLAN International Conference on Object Oriented Programming Systems Languages & Applications*, New York, NY, USA: Association for Computing Machinery, OOPSLA '13, 1–18.

Micci-Barreca, D. (2001): "A Preprocessing Scheme for High-Cardinality Categorical Attributes in Classification and Prediction Problems," *ACM SIGKDD Explorations Newsletter*, 3, 27–32.

Miroudot, S., J. Sauvage, and B. Shepherd (2010): "Measuring the Cost of International Trade in Services," MPRA Paper 27655, University Library of Munich, Germany.

Murtaza, A., A. Saher, M. Zafar, S. Moosavi, M. Aftab, and F. Sanfilippo (2024): "Paradigm shift for predictive maintenance and condition monitoring from Industry 4.0 to Industry 5.0: A systematic review, challenges and case study," *Results in Engineering*, 102935.

Nanz, S. and C. A. Furia (2015): "A Comparative Study of Programming Languages in Rosetta Code," in *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, IEEE, 778–788.

Navot, G. (2024): "The 8 Hidden Costs of Legacy Mainframe Codes in the Finance and Banking Sector," `https://overcast.blog/the-8-hidden-costs-of-legacy-mainframe-`

codes-in-the-finance-and-banking-sector-51e4e1ee62e3, accessed: 16 Feb 2025.

Noy, S. and W. Zhang (2023): "Experimental evidence on the productivity effects of generative artificial intelligence," *Science*, 381, 187–192.

Ovide, S. (2022): "The Long Life of COBOL: Why Banks and Governments Still Rely on It," https://www.nytimes.com/2022/07/06/technology/cobol-jobs.html, accessed: 16 Feb 2025.

Peng, S., E. Kalliamvakou, P. Cihon, and M. Demirer (2023): "The Impact of AI on Developer Productivity: Evidence from GitHub Copilot," .

Ponzanelli, L., G. Bavota, M. Di Penta, R. Oliveto, and M. Lanza (2014): "Mining StackOverflow to Turn the IDE into a Self-Confident Programming Prompter," in *Proceedings of the 11th Working Conference on Mining Software Repositories*, ACM, MSR 2014, 102–111.

Poschke, M. (2018): "The Firm Size Distribution across Countries and Skill-Biased Change in Entrepreneurial Technology," *American Economic Journal: Macroeconomics*, 10, 1–41.

Prytkova, E., F. Petit, D. Li, S. Chaturvedi, and T. Ciarli (2024): "The Employment Impact of Emerging Digital Technologies," CESifo Working Paper Series 10955, CESifo.

Rabah, S., J. Li, M. Liu, and Y. Lai (2010): "Comparative Studies of 10 Programming Languages within 10 Diverse Criteria – a Team 7 COMP6411-S10 Term Report," .

Reimers, N. and I. Gurevych (2019): "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Association for Computational Linguistics, 3982–3992.

——— (2020): "Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, 4512–4525.

Salton, G. and C. Buckley (1988): "Term-weighting approaches in automatic text retrieval," *Information Processing & Management*, 24, 513–523.

Schmitz, J. A. (2005): "What Determines Productivity? Lessons from the Dramatic Recovery of the U.S. and Canadian Iron Ore Industries Following Their Early 1980s Crisis," *Journal of Political Economy*, 113, 582–625.

Schwarz, C. (2019): "lsemantica: A command for text similarity based on latent semantic analysis," *The Stata Journal*, 19, 129–142.

Sevilla, J., L. Heim, A. Ho, T. Besiroglu, M. Hobbhahn, and P. Villalobos (2022): "Compute Trends Across Three Eras of Machine Learning," in *2022 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 1–8.

Shi, W., X. Li, J. Zhang, and J. Sun (2023): "Design and Implementation of an AI-Based Quality Control System in Automotive Production," *Procedia Computer Science*, 228, 2322–2328.

Shojaeinasab, A. et al. (2022): "Intelligent Manufacturing Execution Systems: A Systematic Review," *Journal of Manufacturing Systems*, 62, 503–522.

Tambad, S., R. Nandwani, and S. K. McIntosh (2020): "Analyzing programming languages by community characteristics on Github and StackOverflow," .

Tariq, U., A. Khan, S. Shah, M. Imran, et al. (2024): "A Novel Approach for Automated Quality Inspection Using Deep Learning-Based Computer Vision in Manufacturing Industry," *Electronics*, 13, 976.

Tbit (2024): "GroundEye – Rações," PDF brochure, produced by Tbit. PDF available on tbit.com.br (path includes /2024/08/).

Tolan, S., A. Pesole, F. Martínez-Plumed, E. Fernández-Macías, J. Hernández-Orallo, and E. Gómez (2021): "Measuring the Occupational Impact of AI: Tasks, Cognitive

Abilities and AI Benchmarks," *Journal of Artificial Intelligence Research*, 71, 191–236.

Toosi, A., A. G. Bottino, B. Saboury, E. Siegel, and A. Rahmim (2021): "A Brief History of AI: How to Prevent Another Winter (A Critical Review)," *PET Clinics*, 16, 449–469.

Torquato, P. (2022): "Importance of Registering a Software in Brazil," Accessed: 2024-11-14.

Treude, C. and M. P. Robillard (2016): "Augmenting API Documentation with Insights from Stack Overflow," in *Proceedings of the 38th International Conference on Software Engineering*, ACM, ICSE '16, 392–403.

Ucar, A., M. Karakose, and N. Kirimça (2024): "Artificial Intelligence for Predictive Maintenance Applications: Key Components, Trustworthiness, and Future Trends," *Applied Sciences*, 14, 898.

Vijayaraghavan, S., J. Song, T. Guan, S. Choi, and S. Kulkarni (2022): "Influence of Communication Among Shared Developers on the Productivity of Open Source Software Projects," .

Webb, M. (2020): "The Impact of Artificial Intelligence on the Labor Market," Retrieved from `https://www.michaelwebb.co/webb_ai.pdf`.

Wu, Y., S. Wang, C.-P. Bezemer, and K. Inoue (2019): "How Do Developers Utilize Source Code from Stack Overflow?" *Empirical Software Engineering*, 24, 637–673.

Online Appendix
(for online publication only)

A.I. in the Office and the Factory: Evidence from Administrative Software Registry Data

by Gustavo de Souza

A Theoretical Model

A.1 Equilibrium Definition

This appendix provides the formal details for the model in Section 2 of the main text. There is one final good, used as the numeraire. The wage and employment of skill group s in occupation o are $w_{o,s}$ and $L_{o,s}$.

A.2 Agents

The final-good producer chooses broad-group outputs to solve

$$\max_{\{Y_g\}_{g\in\mathcal{G}}} PY - \sum_{g\in\mathcal{G}} P_g Y_g,$$

where

$$Y = \left(\sum_{g\in\mathcal{G}} \varphi_g^{1/\chi} Y_g^{\frac{\chi-1}{\chi}}\right)^{\frac{\chi}{\chi-1}}, \qquad \varphi_g > 0, \qquad \sum_g \varphi_g = 1.$$

The broad-group producer chooses task quantities to solve

$$\max_{\{y(j)\}_{j\in J_g}} P_g Y_g - \int_{J_g} p(j) y(j)\, dj,$$

where

$$Y_g = \left(\int_{J_g} y(j)^{\frac{\psi-1}{\psi}}\, dj\right)^{\frac{\psi}{\psi-1}}.$$

Workers in occupation-skill cell (o, s) supply labor according to

$$L^S_{o,s} = \bar{L}_{o,s} w^{\eta}_{o,s}.$$

To supply $K_o(j)$, a developer solves

$$\min_{\{x_{o,l}(j)\}_{l\in\mathcal{L}}} \sum_{l\in\mathcal{L}} c_l x_{o,l}(j) \quad \text{s.t.} \quad K_o(j) \leq \prod_{l\in\mathcal{L}} x_{o,l}(j)^{\alpha_{j,l}}.$$

Task $j \in J_o$ may be produced by either skill group. Conditional on skill s, task output is

$$y_{o,s}(j) = A_{o,s}(j) \left[\ell_{o,s}(j)^{\frac{\sigma_{g,s}-1}{\sigma_{g,s}}} + K_{o,s}(j)^{\frac{\sigma_{g,s}-1}{\sigma_{g,s}}} \right]^{\frac{\sigma_{g,s}}{\sigma_{g,s}-1}}.$$

For a required output level $y(j)$, the task producer solves

$$\min_{\ell_{o,s}(j), K_{o,s}(j)\geq 0} w_{o,s}\ell_{o,s}(j) + q_o(j)K_{o,s}(j)$$

subject to $y(j) \leq y_{o,s}(j)$. The producer assigns the task to the least-cost skill group.

A.3 Equilibrium

Definition A.1 (Competitive equilibrium)**.** A competitive equilibrium is given by prices

$$\{P, \{P_g\}_{g\in\mathcal{G}}, \{p(j)\}_j, \{q_o(j)\}_{o,j}, \{w_{o,s}\}_{o,s}\},$$

allocations

$$\{Y, \{Y_g\}_g, \{y(j)\}_j, \{\ell_{o,s}(j), K_{o,s}(j)\}_{o,s,j}, \{K_o(j)\}_{o,j}, \{x_{o,l}(j)\}_{o,j,l}, \{L_{o,s}\}_{o,s}\},$$

and a task assignment $s^*(o,j)$ such that:

1. the final-good producer solves its problem, taking $\{P_g\}_g$ and P as given;

2. each broad-group producer solves its problem, taking $\{p(j)\}_{j\in J_g}$ and P_g as given;

3. labor supply satisfies
$$L^S_{o,s} = \bar{L}_{o,s} w^{\eta}_{o,s} \qquad \text{for all } o, s;$$

4. AI developers solve their cost-minimization problems and price AI at unit cost:
$$q_o(j) = \prod_{l\in\mathcal{L}} \left(\frac{c_l}{\alpha_{j,l}}\right)^{\alpha_{j,l}} \qquad \text{for all } o, j;$$

5. task producers solve their cost-minimization problems, task prices equal minimized unit costs, and

tasks are assigned to least-cost skill groups:

$$s^*(o,j) \in \arg\min_{s\in\{L,H\}} p_{o,s}(j), \qquad p(j) = \min_{s\in\{L,H\}} p_{o,s}(j),$$

where $p_{o,s}(j)$ is the minimized unit cost for task j with skill group s. Inputs of non-assigned skill groups are zero:

$$\ell_{o,s}(j) = K_{o,s}(j) = 0 \qquad \text{if } s \neq s^*(o,j);$$

6. labor markets clear:

$$L^S_{o,s} = \int_{J_o} \mathbf{1}\{s = s^*(o,j)\}\ell_{o,s}(j)\,dj \qquad \text{for all } o, s;$$

7. AI markets clear task by task:

$$K_o(j) = \sum_{s\in\{L,H\}} \mathbf{1}\{s = s^*(o,j)\}K_{o,s}(j) \qquad \text{for all } o, j;$$

8. the final-good market clears and the numeraire normalization is

$$P = 1.$$

A.4 Equilibrium Characterization

From the first order conditions of developers, the price of AI is

$$q_o(j) = \prod_{l\in\mathcal{L}} \left(\frac{c_l}{\alpha_{j,l}}\right)^{\alpha_{j,l}} = \kappa_j \prod_{l\in\mathcal{L}} c_l^{\alpha_{j,l}}, \qquad \kappa_j \equiv \prod_{l\in\mathcal{L}} \alpha_{j,l}^{-\alpha_{j,l}}.$$

From firm's maximization, the unit cost of producing task $j \in J_o$ with skill group s is

$$p_{o,s}(j) = \frac{1}{A_{o,s}(j)} \left[w_{o,s}^{1-\sigma_{g,s}} + q_o(j)^{1-\sigma_{g,s}}\right]^{\frac{1}{1-\sigma_{g,s}}}.$$

If task j is produced with skill group s, labor and AI demand are given by

$$\ell_{o,s}(j) = y(j)A_{o,s}(j)^{\sigma_{g,s}-1}p(j)^{\sigma_{g,s}}w_{o,s}^{-\sigma_{g,s}} \tag{9}$$

$$K_{o,s}(j) = y(j)A_{o,s}(j)^{\sigma_{g,s}-1}p(j)^{\sigma_{g,s}}q_o(j)^{-\sigma_{g,s}}. \tag{10}$$

The corresponding labor and AI cost shares are

$$s_{L,o,s}(j) = \frac{w_{o,s}^{1-\sigma_{g,s}}}{w_{o,s}^{1-\sigma_{g,s}} + q_o(j)^{1-\sigma_{g,s}}}, \qquad s_{M,o,s}(j) = \frac{q_o(j)^{1-\sigma_{g,s}}}{w_{o,s}^{1-\sigma_{g,s}} + q_o(j)^{1-\sigma_{g,s}}}. \tag{11}$$

The equilibrium wage is given by

$$\bar{L}_{o,s} w_{o,s}^{\eta} = \int_{J_o} \mathbf{1}\{s = s^*(o,j)\} y(j) A_{o,s}(j)^{\sigma_{g,s}-1} p(j)^{\sigma_{g,s}} w_{o,s}^{-\sigma_{g,s}}\, dj. \tag{12}$$

A.5 AI Price Shocks and Exposure

Assume an AI price shock as in Equation (2). Also assume that the price change is small enough not to change the allocation of skill groups across tasks and to fall equally across skill groups. Let

$$L_o = \sum_s L_{o,s}, \qquad \lambda_o(j) = \frac{\sum_s \mathbf{1}\{s = s^*(o,j)\}\ell_{o,s}(j)}{L_o}.$$

Write $s_{M,o}(j)$ for the AI cost share of the skill group assigned to task j. Occupation-level AI shock exposure is

$$\textit{AI Shock Exposure}_o = \int_{J_o} s_{M,o}(j)\lambda_o(j)\tau_o(j)\, dj. \tag{13}$$

Equivalently,

$$\textit{AI Shock Exposure}_o = \sum_{l \in \mathcal{L}} \Omega_{o,l}\tau_l, \qquad \Omega_{o,l} = \int_{J_o} s_{M,o}(j)\lambda_o(j)\alpha_{j,l}\, dj. \tag{14}$$

Let $J_{o,s}$ be the tasks assigned to skill group s and occupation o. Define

$$L_{o,s} = \int_{J_{o,s}} \ell_{o,s}(j)\, dj, \qquad \lambda_{o,s}(j) = \frac{\ell_{o,s}(j)}{L_{o,s}}, \tag{15}$$

$$\bar{s}_{M,o,s} = \int_{J_{o,s}} \lambda_{o,s}(j) s_{M,o,s}(j)\, dj, \qquad \mathcal{E}^{AI}_{o,s} = \int_{J_{o,s}} \lambda_{o,s}(j) s_{M,o,s}(j)\tau_o(j)\, dj. \tag{16}$$

For tractability, I assume that the AI shock is balanced across skill groups:

$$\mathcal{E}^{AI}_{o,L} = \mathcal{E}^{AI}_{o,H} \equiv \textit{AI Shock Exposure}_o \equiv \mathcal{E}_o. \tag{17}$$

Under the assumption of a small price shock, the change in labor is given by

$$d \ln L_{o,s} = \frac{\eta}{\eta + \psi + (\sigma_{g,s} - \psi)\bar{s}_{M,o,s}} \left[\Xi_g + (\psi - \sigma_{g,s})\mathcal{E}^{AI}_{o,s}\right]. \tag{18}$$

where

$$\Xi_g \equiv d\ln Y_g + \psi\, d\ln P_g. \tag{19}$$

Under the small-AI-share approximation, (18) becomes

$$d\ln L_{o,s} \approx \frac{\eta}{\eta+\psi}\Xi_g + \frac{\eta(\psi-\sigma_{g,s})}{\eta+\psi}\mathcal{E}^{AI}_{o,s}. \tag{20}$$

A.6 Proofs

Let

$$\mathcal{E}_o \equiv \textit{AI Shock Exposure}_o, \qquad \pi_o \equiv \frac{L_{o,L}}{L_o}, \qquad 1-\pi_o \equiv \frac{L_{o,H}}{L_o}.$$

Proof of Proposition 1: AI has an ambiguous effect on employment. Under the local small-shock assumptions,

$$d\ln L_o \approx \frac{\eta}{\eta+\psi}\Xi_g + \beta^L_o \mathcal{E}_o,$$

where

$$\beta^L_o = \frac{\eta}{\eta+\psi}\left[\psi - \bar{\sigma}_g\right], \qquad \bar{\sigma}_g \equiv \pi_o\sigma_{gL} + (1-\pi_o)\sigma_{gH}.$$

Proof. For each skill group,

$$d\ln L_{o,s} \approx \frac{\eta}{\eta+\psi}\Xi_g + \frac{\eta(\psi-\sigma_{g,s})}{\eta+\psi}\mathcal{E}_o.$$

Since $L_o = L_{o,L} + L_{o,H}$,

$$d\ln L_o = \pi_o d\ln L_{o,L} + (1-\pi_o)d\ln L_{o,H}.$$

Substitution gives

$$d\ln L_o \approx \frac{\eta}{\eta+\psi}\Xi_g + \frac{\eta}{\eta+\psi}\left[\pi_o(\psi-\sigma_{gL}) + (1-\pi_o)(\psi-\sigma_{gH})\right]\mathcal{E}_o.$$

Using $\pi_o + (1-\pi_o) = 1$,

$$d\ln L_o \approx \frac{\eta}{\eta+\psi}\Xi_g + \frac{\eta}{\eta+\psi}\left[\psi - (\pi_o\sigma_{gL} + (1-\pi_o)\sigma_{gH})\right]\mathcal{E}_o.$$

The coefficient on exposure is positive if $\psi > \bar{\sigma}_g$, negative if $\psi < \bar{\sigma}_g$, and zero if $\psi = \bar{\sigma}_g$. □

Proof of Proposition 2: Average Treatment Effect Depends on Sample Composition. The linear projection of occupation employment on AI exposure can be approximated by

$$d\ln L_o \approx \frac{\eta}{\eta+\psi}\Xi_g + \beta^L\, \textit{AI Shock Exposure}_o + \epsilon_o,$$

where

$$\beta^L = \sum_o \omega_o \beta_o^L,$$

with $\pi_o, \omega_o \in (0,1)$, $\sum_o \omega_o = 1$, and ϵ_o orthogonal to *AI Shock Exposure*$_o$.

Proof. Let

$$\widetilde{Y}_o \equiv d\ln L_o - \frac{\eta}{\eta+\psi}\Xi_g, \qquad X_o \equiv \mathcal{E}_o.$$

Proposition 1 gives

$$\widetilde{Y}_o = \beta_o^L X_o.$$

The simple projection coefficient of $\widetilde{Y}_o$ on X_o is

$$\beta^L = \frac{\sum_o X_o \widetilde{Y}_o}{\sum_o X_o^2} = \frac{\sum_o X_o^2 \beta_o^L}{\sum_o X_o^2}.$$

Thus

$$\beta^L = \sum_o \omega_o \beta_o^L, \qquad \omega_o = \frac{X_o^2}{\sum_{o'} X_{o'}^2}.$$

If there is positive exposure variation, the weights are nonnegative and sum to one. The residual

$$\epsilon_o = \widetilde{Y}_o - \beta^L X_o$$

is orthogonal to X_o by the definition of the projection. □

Proof of Proposition 3: AI can change the composition of workers within occupations. Assume that the initial AI expenditure share is small. Then, the effect of AI on the share of low-skilled workers is given by:

$$d\textit{Shr. Low-Skilled}_o \approx \Theta_o \left(\sigma_{g,H} - \sigma_{g,L}\right) \textit{AI Shock Exposure}_o,$$

where $\Theta_o > 0$ is a set of parameters.

Proof. The low-skill share is $\pi_o = L_{o,L}/L_o$. A first-order change is

$$d\pi_o = \pi_o(1-\pi_o)\left(d\ln L_{o,L} - d\ln L_{o,H}\right).$$

Using the skill-specific employment responses,

$$d\ln L_{o,L} - d\ln L_{o,H} \approx \frac{\eta}{\eta+\psi}\left[(\psi - \sigma_{g,L}) - (\psi - \sigma_{g,H})\right]\mathcal{E}_o.$$

The ψ terms cancel, so

$$d \ln L_{o,L} - d \ln L_{o,H} \approx \frac{\eta}{\eta + \psi} \left(\sigma_{g,H} - \sigma_{g,L} \right) \mathcal{E}_o.$$

Substituting into the share identity gives

$$d\pi_o \approx \frac{\eta}{\eta + \psi} \pi_o (1 - \pi_o) \left(\sigma_{g,H} - \sigma_{g,L} \right) \mathcal{E}_o.$$

The stated expression follows with

$$\Theta_o = \frac{\eta}{\eta + \psi} \pi_o (1 - \pi_o).$$

Since $\eta > 0$, $\psi > 0$, and $\pi_o \in (0, 1)$, $\Theta_o > 0$. □

B Data

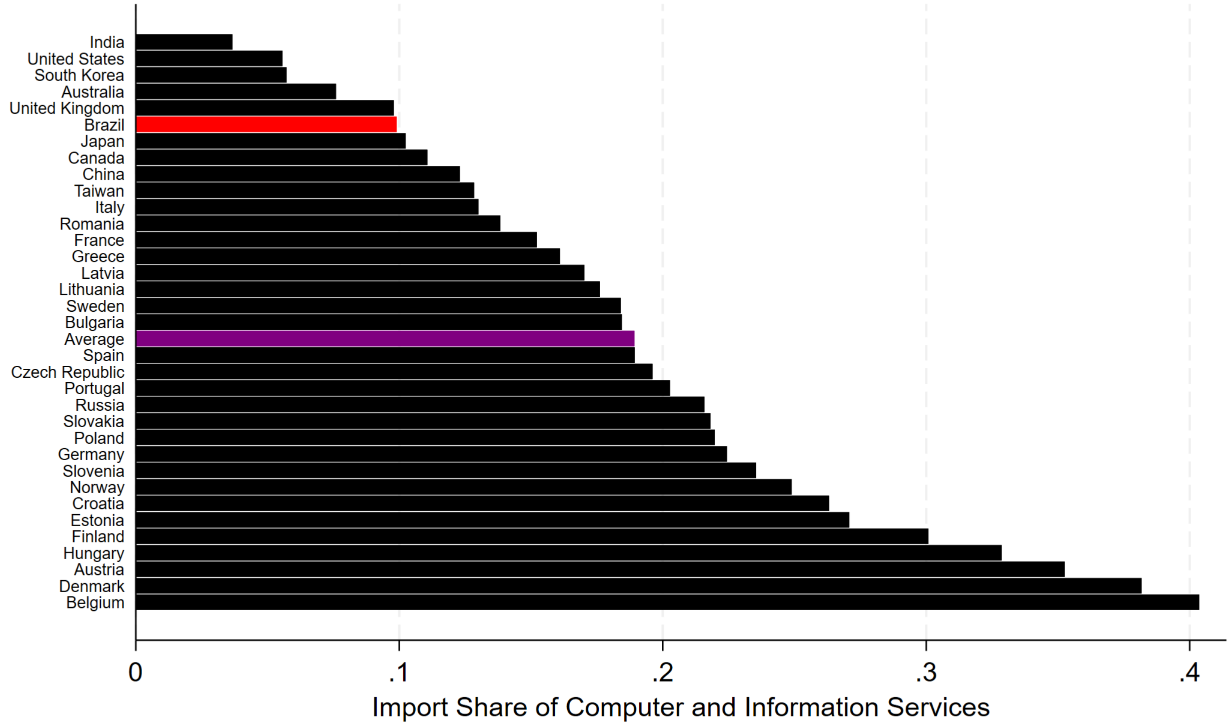
B.1 Most Software Used in Brazil Is Developed Locally

The NIIP registry covers only software developed in Brazil, so it is important to assess the role of foreign software. Figure A1 shows that imported computer services accounted for only 9.3% of domestic computer-service value added in Brazil in 2015, less than half the OECD average of roughly 19%.[36] Since imported software is typically paid through licensing fees and does not appear in Brazil's copyright registry, this low import share suggests that most software used by Brazilian firms is developed locally and therefore can be captured by NIIP.

[36] Computer services include software publishing, programming, data processing, and cross-border licensing fees for packaged software. Because the data do not separate software from data-processing and data-center services, this measure likely overstates software imports. Consistent with this, Junqueira Botelho et al. (2005) reports that domestic firms supplied 98% of Brazil's software market.

Figure A1: **Import Share of Computer Services Across the Globe in 2015**

In Brazil, Most Software is Locally Developed



Notes: The figure reports imports of computer services divided by computer-services value added in 2015. Import data come from the WTO-OECD Balanced Trade in Services dataset; value added comes from PREDICT.

Foreign software vendors could also operate through Brazilian subsidiaries and bypass local registration. This channel appears small. Table A1 shows that foreign-controlled firms account for fewer than 0.04% of firms and 0.08% of employment in Brazil's IT sectors. Thus, nearly all software produced and sold through the domestic IT sector is produced by domestically owned firms.

Table A1: **Share of Multinational Firms in IT Sectors**

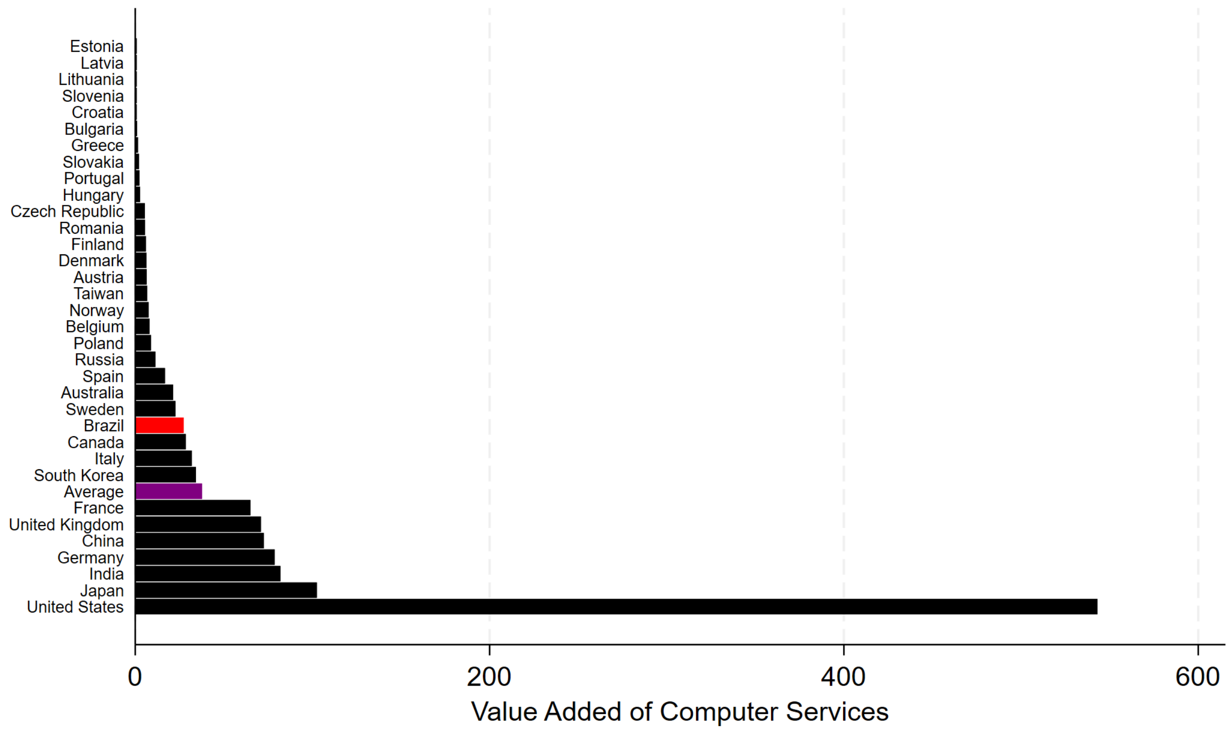
	Share of Multinationals	*Employment-Weighted Share*
Information Technology Services	0.020%	0.025%
Information Service Provision Activities	0.038%	0.078%
Total	0.025%	0.035%

Notes: The table reports the share of foreign companies in Brazil's IT sector in 2016. Information Technology Services corresponds to CNAE 2.0 code 62; Information Service Provision Activities corresponds to code 63.

Three forces explain the small role of imported software. First, digital services exhibit strong home bias: even in the United States, only the largest firms export online services, and domestic providers are especially important in digital services (Alaveras and Martens 2015). Trade costs in services are also two to three times higher than in goods, partly because of cultural and regulatory frictions (Miroudot et al. 2010, Harms and Shuvalova 2020). Second, Brazil raises these barriers further through local-infrastructure mandates, data-localization requirements, and domestic-content rules in public procurement, which increase the effective trade cost of imported IT services to the equivalent of an 87 percent tariff (Ezell et al. 2013, Benz and Jaax

2022). Third, Brazil has a large domestic software industry: in 2015, it ranked as the 11th largest IT sector globally, and domestic firms supplied 98 percent of software services as early as 2001 (Junqueira Botelho et al. 2005, Arora and Gambardella 2005). Together, these facts indicate that imported software plays a minor role and that the NIIP registry provides broad coverage of software used in Brazil.

Figure A2: **Value Added of Computer Services Across the Globe in 2015**



Notes: The figure reports computer-services value added in 2015, in billions of 2015 dollars, using PREDICT data.

B.1.1 Survey with Industry Specialists

In this section I describe the results of a survey with industry specialists who have expert knowledge of the NIIP registration process. The survey delivers three main findings. First, software registration in the NIIP is widely recognized among both intellectual property lawyers and IT professionals, with the vast majority having direct experience with the registration process. Second, respondents confirm that most commercial software developed in Brazil is registered, with firms primarily motivated by intellectual property protection. Third, to the extent that selection exists, it is driven by small firms producing consumer-facing applications such as video games and mobile apps, categories that are unlikely to affect the labor market outcomes studied in this paper. These findings support the validity of using NIIP data to measure AI exposure in Brazilian workers.

Survey design. I conducted the survey with two groups that have detailed knowledge of the NIIP registration process: intellectual property lawyers and IT professionals. I identified potential respondents through an online search, constructing a database of 342 intellectual property law offices and 276 software development companies. I contacted these firms and offices through email or LinkedIn between September and December of 2025. Each respondent was compensated for their participation.

In total, 68 people completed the survey: 41 intellectual property lawyers and 27 IT professionals. This gives a response rate of 11%. This small response rate comes from the fact that I contacted firms through publicly available email addresses, which often serve only customer inquiries rather than research requests. Individual lawyers, who typically manage their own correspondence, showed higher response rates than firms.

Survey respondents are experienced in their field. Table A2 presents summary statistics for the sample of respondents. Respondents average more than 10 years of experience in either IT or intellectual property law, with 86% holding at least a bachelor's degree. Among IT professionals, 81% work full-time in the IT sector, and 62% identify as programmers. Therefore, this is an experienced sample of respondents that are well-suited to assess the prevalence and patterns of software registration in Brazil.

Table A2: **Survey Summary Statistics**

	Programmers	Lawyers	All
Average age	36.2	43.8	40.7
Average experience	12.9	18.9	16.6
Bachelor degree or more (%)	70.4	97.6	86.8

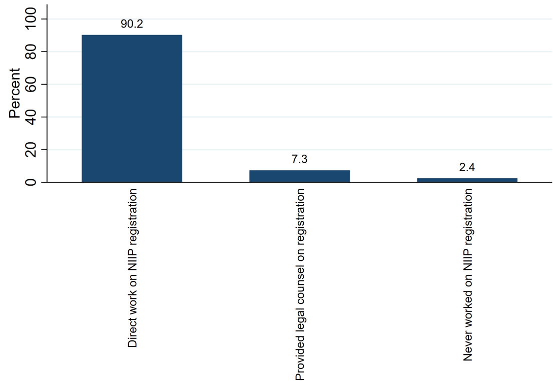
NIIP registration is recognized among specialists. The NIIP software registration process is widely known among industry specialists. Among IT professionals, 96% report being familiar with the registry. Among intellectual property lawyers, 90% have actively participated in at least one software registration, with the average lawyer having registered 37 distinct software programs. Among IT professionals, 77% report either having participated in a registration themselves or working at a firm that has registered software. This high level of familiarity indicates that the NIIP registry is a salient feature of Brazil's software industry, consistent with the institutional background described in Section 3.1.

Figure A4a shows IT professionals' responses to the question: "How many commercial software programs do you believe have at least one version registered in the NIIP?" Seventy percent of respondents believe that firms register all or most of their commercial software, though 11% believe that only a few software programs are registered. This distribution suggests that while registration is the norm, some selection may exist.

Figure A4b displays the percentage of respondents who agree with various motivations for registering software in the NIIP. The dominant reasons are protecting intellectual property rights and preventing piracy, consistent with the main objectives of the NIIP.

Figure A3: **Experience of Respondents with a Software Registration**

(a) Intellectual Property Lawyer



(b) IT Professional

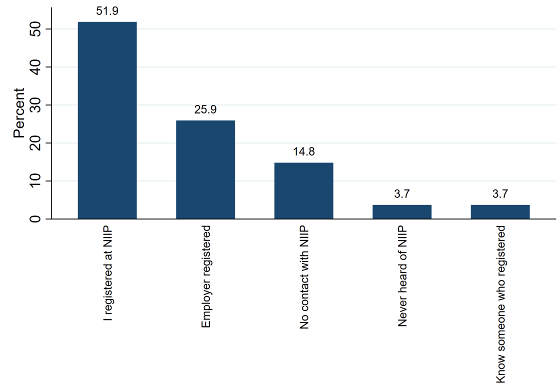
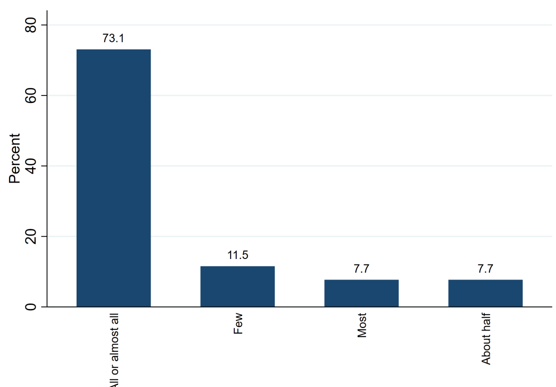
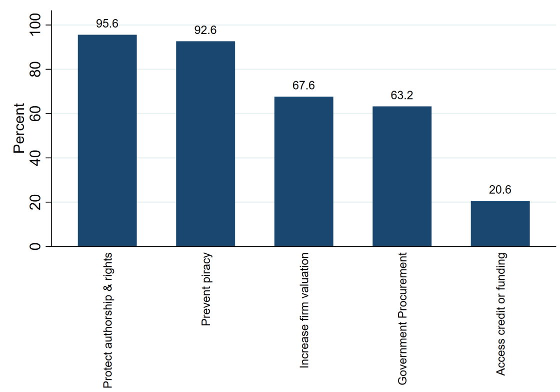


Figure A4: **Dataset Coverage**

(a) How many commercial software do you believe have at least one version registered in the NIIP?



(b) Why are firms registering software?



Selection is driven by small firms producing video games or cellphone apps. To understand which types of software might be underrepresented in the NIIP registry, I asked respondents about non-registration patterns. Figure A5a shows the reasons respondents cite for why some software remains unregistered. The two most common explanations are that a previous version of the software has already been registered or that the firm is unaware of the NIIP. Notably, only 4% of IT professionals in the survey report being unfamiliar with the registry, suggesting that, although the lack of awareness explains why firms do not register, it is quantitatively a small factor.

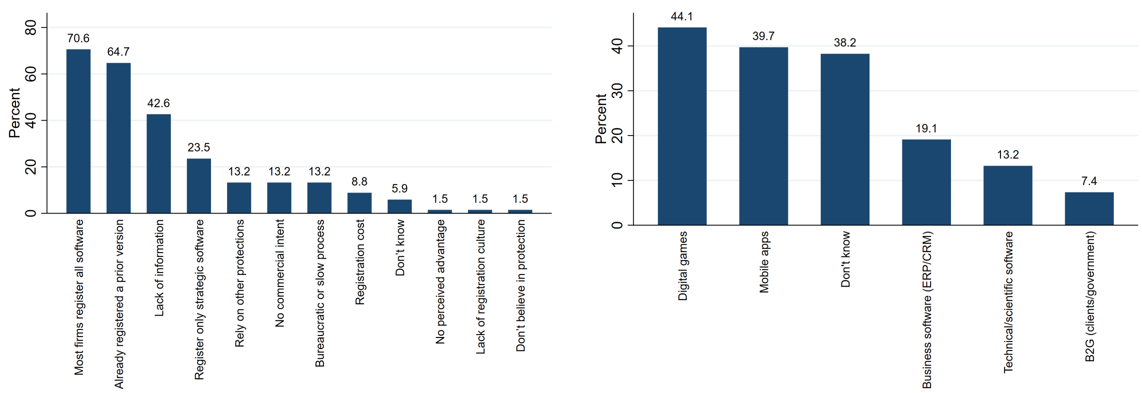
Figure A5b identifies which types of software are least likely to be registered. Respondents most frequently cite video games and mobile apps. Follow-up interviews with respondents reveal that developers of these consumer-facing applications perceive limited benefits from NIIP registration because distribution platforms (such as app stores) already provide intellectual property protections. Moreover, 83% of respondents agree that large companies are more likely to be aware of the NIIP and register their software than small firms.

Taken together, these findings indicate that any possible selection in the NIIP registry would be against small firms producing consumer-oriented applications. These categories represent a small share of enterprise software and are unlikely to significantly affect the labor market outcomes examined in this paper. The software that is used by firms, such as enterprise resource planning systems, manufacturing execution systems, and business intelligence tools, is comprehensively covered in the NIIP data.

Figure A5: **Dataset Selection**

(a) Why do some firms not register their software?

(b) What types of software do you think are less likely to be registered?

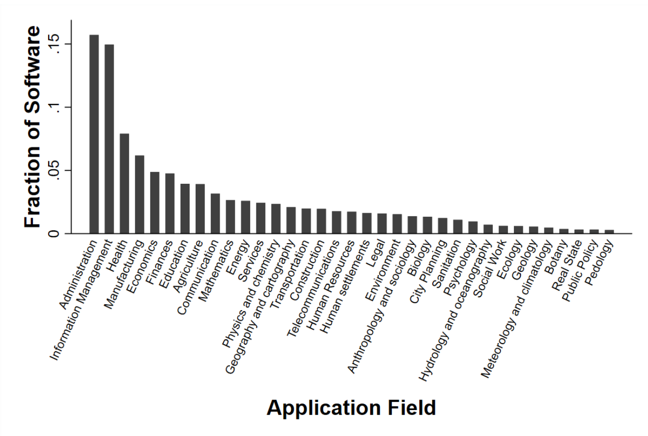


B.2 Summary Statistics of the Software Dataset

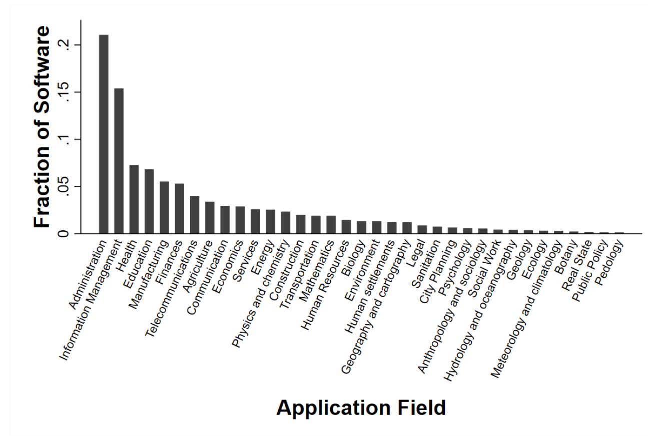
This section presents summary statistics for the software dataset. Figure A6 compares the distribution of application domains between AI and non-AI software. As shown in Figure A6b, non-AI software is more heavily concentrated in administrative and information management domains, whereas AI software is more broadly distributed across different areas.

Figure A6: **AI and Non-AI Application Domain**

(a) Fraction of Application Domains Among AI Software



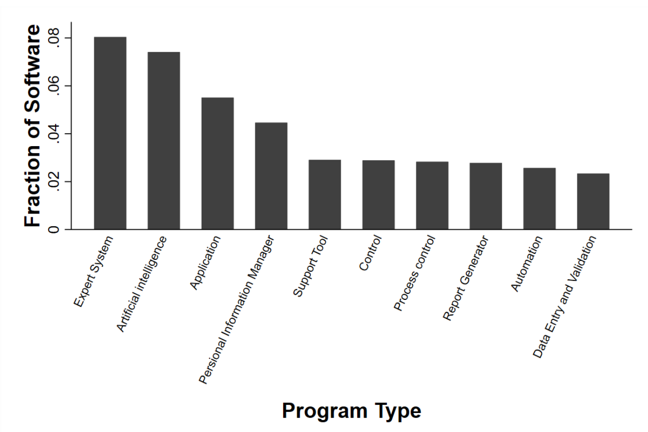
(b) Fraction of Application Domains Among Non-AI Software



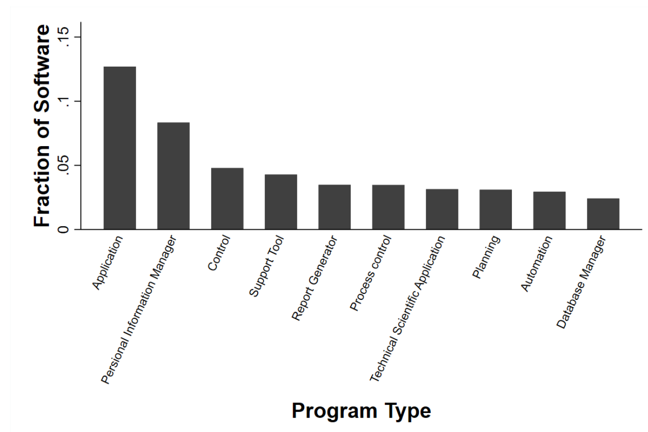
Notes: These figures present the distribution of AI software across different application domains, as defined by the NIIP. The application domain classification was developed by the NIIP to indicate the areas in which each software is intended to be used. Because a single software program may be assigned to multiple domains, each figure reports the fraction of each domain relative to the total number of domain classifications. Figure A6a plots the distribution of application domains among AI software, while Figure A6b plots the distribution of application domains among non-AI software.

Figure A7: **Fraction of Technical Class Among AI and Non-AI Software**

(a) Fraction of Technical Class Among AI Software



(b) Fraction of Technical Class Among non-AI Software

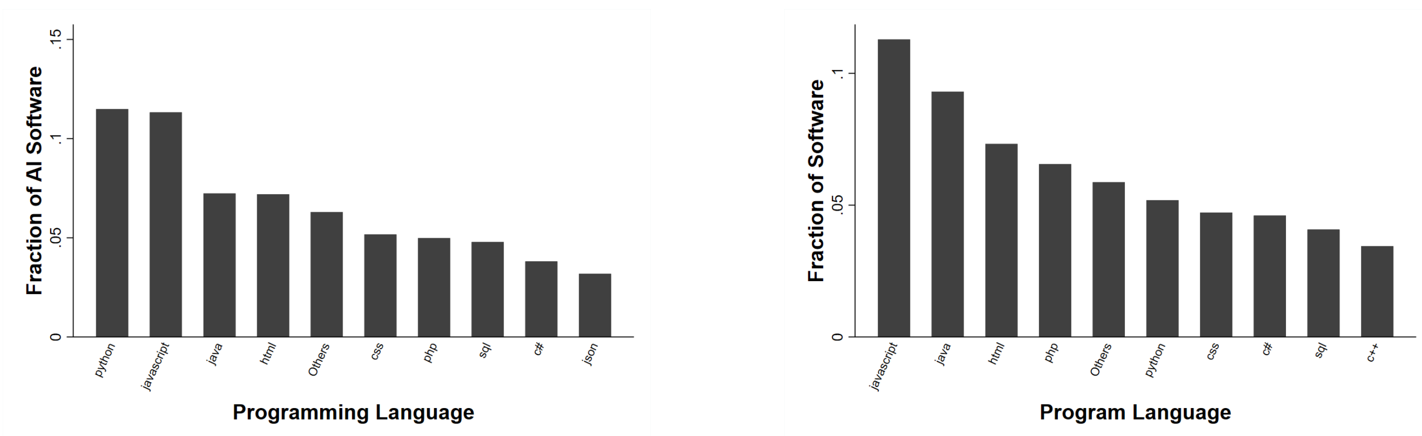


Notes: These figures show the distribution of technical class among AI and non-AI software.

Figure A8: **Fraction of Different Programming Languages Among AI and Non-AI Software**

(a) Fraction of Programming Language Among AI Software

(b) Fraction of Programming Language Among non-AI Software



Notes: These figures show the distribution of programming languages among AI and non-AI software.

Figure A9: **Word Cloud from AI Software Titles**

Notes: This figure shows a word cloud from the titles of AI software.

C Empirics

C.1 Text Similarity

In this section, I describe in detail how I calculate the text similarity between software descriptions and occupational tasks.

Parsing. I transform documents into vectors using words as tokens, i.e., unigrams.

Selection. To avoid counting frequent and uninformative words, such as the Portuguese equivalent for "the" and "and", I drop terms that appear in more than 80% of documents.

Vectorization. Following the previous steps, I characterize each document with a vector of dummies for words it contains. Let $m \in \{1, ..., M\}$ index the set of words in the documents, $\mathcal{M}$. Let c_{km} be a dummy variable taking the value of 1 if the document k contains word m. Therefore, document k can be represented by vector c_k with entries c_{km}.

Normalization. Rare words are more important for characterizing differences across documents than common words. To take that into account, I weight each word using term-frequency-inverse-document-frequency (tf-idf). Each term m of the dataset is weighted by:

$$\omega_m = \log\left(\frac{K+1}{d_m+1}\right) + 1 \text{ where } d_m = \sum_k \mathbb{I}\left\{c_{km} > 0\right\}.$$

After weighting, each document is represented by word frequency vector f_k with entries:

$$f_{km} = \frac{\omega_m c_{km}}{\sqrt{\sum_{m'}(\omega_m c_{km})^2}}.$$

Singular Value Decomposition. Let F represent the entire corpus. Latent semantic analysis decomposes the matrix F as

$$F = U\Sigma V'$$

where U is the term-factor matrix, Σ the diagonal matrix of singular values, and V' the factor-document matrix.

Rank-k approximation. I set to zero all the singular values outside the top 300, following Schwarz (2019). Using the decomposition, I recover the matrix $\tilde{F}$ given by

$$\tilde{F} = U\tilde{\Sigma}V'$$

where $\tilde{\Sigma}$ has only the first 300 singular values different from zero.

Similarity Scores. The similarity between software s and tasks of occupation o is given by:

$$s_{so} = \sum_{m\in\mathbb{M}} \tilde{f}_{sm} \times \tilde{f}_{om}. \tag{21}$$

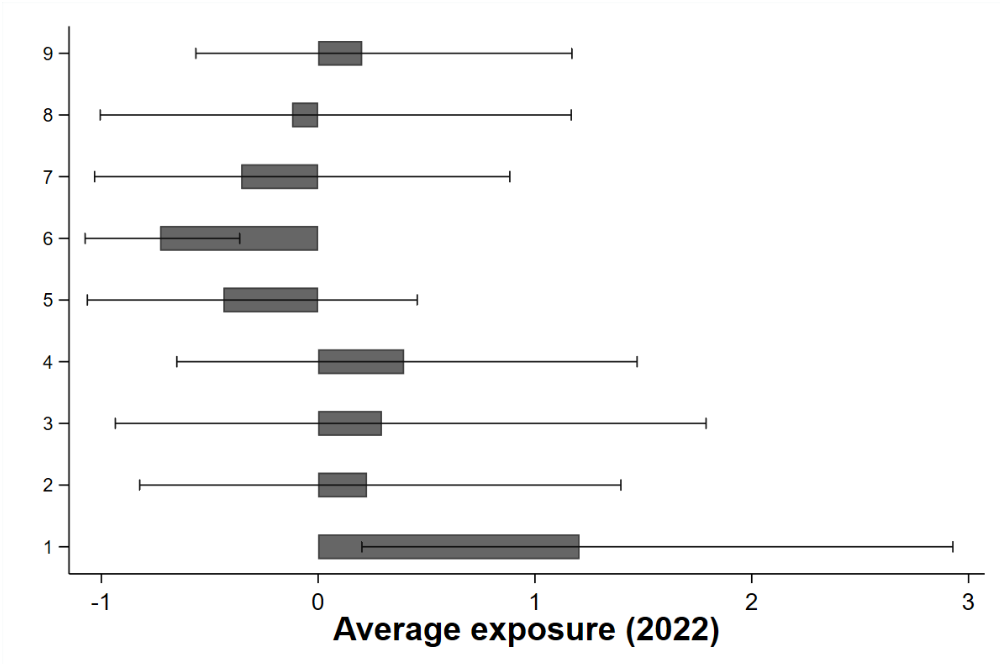
C.2 Summary Statistics of AI Exposure

Figure A10 shows average AI exposure in 2022 across 1-digit occupation groups. Brackets indicate the range from the bottom to the top decile within each category. Exposure varies widely both across and within groups: managers, science professionals, and administrative workers are the most exposed, while agricultural workers are the least exposed.[37] Within-group variation is particularly notable: the

[37]Prytkova et al. (2024), Gmyrek et al. (2023), Felten et al. (2018), and Webb (2020) also find the same general patterns.

most exposed manufacturing worker, for example, faces greater AI exposure than the average administrative worker. This wide distribution of AI exposure across occupations likely reflects its diverse range of applications, as discussed in Section 4.

Figure A10: **Large Variance within and Across Broad Occupations in AI Exposure**



Notes: This figure plots the average AI exposure in 2022 at a one-digit CBO occupational classification. The bars plot the average AI exposure and the lines the top and bottom decile.

Table A3, in the Appendix, shows the correlation between each occupation's 2003 characteristics and its exposure to AI software in 2022. Occupations more exposed to AI tend to have higher hourly wages, greater education, and larger workforces. They are also more likely to involve tasks related to planning, management, and interpersonal activities, which is consistent with the finding that 40% of AI software targets administrative tasks.

C.3 Counting AI Posts in Stack Overflow and Mailing Lists

I measure AI-related knowledge at the programming-language-by-year level using Stack Overflow and mailing-list posts. The construction has three steps. First, I define AI keywords from Stack Overflow's community-created tag dictionary. Second, I use Stack Overflow tags to identify posts that refer to both a programming language and AI. Third, I apply the same AI keyword dictionary to programming-language mailing lists and count new AI-related discussion threads. This procedure produces a programming-language-by-year panel of AI-related posts.

AI keywords. Stack Overflow tags organize questions by programming language, technology, method, package, and application. Users choose tags when asking questions; users with sufficient reputation can create new tags; and other community members can edit tags when they are missing or inappropriate.

Table A3: **Correlation Between AI Exposure and Characteristics of the Occupation**

	(1)	(2)	(3)
	AI $Exposure_{o,2022}$	*AI* $Exposure_{o,2022}$	*AI* $Exposure_{o,2022}$
$log(hour\ wage)_{o,2003}$	0.651^{***}	0.556^{***}	0.483^{***}
	(0.0555)	(0.0599)	(0.0627)
$log(yrs.\ educ.)_{o,2003}$	1.130^{***}	1.345^{***}	1.091^{***}
	(0.164)	(0.205)	(0.214)
$log(n.\ workers)_{o,2003}$	0.0531^{***}	0.0629^{***}	0.0735^{***}
	(0.0125)	(0.0130)	(0.0136)
$\mathbb{I}\{use\ computer\}_o$	0.162^{**}	0.241^{***}	0.222^{***}
	(0.0630)	(0.0650)	(0.0665)
$\mathbb{I}\{operate\ machine\}_o$	−0.0601	−0.0291	−0.0116
	(0.0626)	(0.0675)	(0.0676)
$\mathbb{I}\{Planning\ or\ Execution\}_o$	0.295^{***}	0.207^{***}	−0.00117
	(0.0757)	(0.0753)	(0.0762)
$\mathbb{I}\{Creation\ and\ Innovation\}_o$	-0.255^{**}	-0.249^{**}	-0.176^{**}
	(0.0837)	(0.0831)	(0.0823)
$\mathbb{I}\{Management\}_o$	0.333^{***}	0.282^{***}	0.217^{**}
	(0.0599)	(0.0600)	(0.0619)
$\mathbb{I}\{Interpersonal\}_o$	0.381^{***}	0.435^{***}	0.432^{***}
	(0.0635)	(0.0688)	(0.0726)
Controls		1-digit Occ. FE	2-digit Occ. FE
Observations	2,148	2,148	2,148
R^2	0.3809	0.4100	0.4731

Notes: This table reports regression results where the dependent variable is the stock cosine similarity in 2022. Each regressor is defined at the occupational level in 2003: *log(hour wage)* is the log of average hourly wage; *log(yrs. educ.)* is the log of average years of schooling; *log(n. workers)* is the log of the number of workers; $\mathbb{I}\{use\ computer\}$ is a dummy if among the tools used by the occupation is a computer; $\mathbb{I}\{operate\ machine\}$ is a dummy taking one if among the tasks performed by the occupation there is machine operation; $\mathbb{I}\{Planning\ or\ Execution\}$ is a dummy taking one if the occupation performs tasks related to planning or execution, $\mathbb{I}\{Creation\ and\ Innovation\}$ is a dummy taking one if the occupation performs tasks related to creation or innovation, $\mathbb{I}\{Management\}$ is a dummy if the occupation performs tasks related to management, and $\mathbb{I}\{Interpersonal\}$ is a dummy taking one if the occupation performs tasks related to interpersonal relationships. Standard errors in parentheses. * $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$.

These tags are therefore a useful vocabulary for measuring programming knowledge because they reflect how programmers classify questions. I define AI-related posts using tags in five groups: core AI and machine learning, model architectures, natural language and speech, computer vision, and AI tools and applications. The empirical measure uses the union of all tags listed below.

- Core AI and machine-learning tags: ai, artificial-intelligence, machine-learning, ml, deep-learning, supervised-learning, unsupervised-learning, reinforcement-learning, and transfer-learning.
- Model and architecture tags: neural-networks, rnn, recurrent-neural-network, lstm, long-short-term-memory, cnn, transformer, bert, gpt, gan, and generative-adversarial-network.
- Natural-language, text, and speech tags: natural-language-processing, nlp, text-classification, question-answering, tokenization, sentiment-analysis, text-generation, speech-recognition, voice-processing, and chatbot.
- Computer-vision tags: computer-vision, image-processing, object-detection, and image-recognition.

- AI tool and application tags: tensorflow, pytorch, keras, scikit-learn, xgboost, lightgbm, openai-api, huggingface, data-preprocessing, feature-selection, dimensionality-reduction, robotics, recommendation-engine, anomaly-detection, autonomous-vehicles, and predictive-modeling.

Stack Overflow posts. I use Stack Overflow tags in two ways. Language tags assign each post to one or more programming languages. AI tags identify whether the post discusses AI. A post belongs to language l in year t if it was posted in year t and contains a tag mapped to language l. I classify that post as AI-related if it also contains at least one of the AI tags listed above.

Mailing-list posts. Mailing-list posts do not have community-created tags, so I apply the same AI dictionary as a keyword list. For each programming-language mailing list, I count thread-starting posts whose subject line contains one of the AI keywords. I focus on thread starts to measure new discussions rather than replies within the same discussion. I then aggregate these counts to the programming-language-by-year level and combine them with the Stack Overflow counts.

C.4 Software Registration Value

Calculating software valuation from stock-market responses. I estimate the value of software registrations from stock-market responses, following Kogan et al. (2017). The idea is that when a software registration becomes public, investors update their expectations about the firm's future profits, and this update is reflected in the firm's stock-market value. A larger stock-market response therefore implies that the software is expected to generate greater private value for the firm. This value can reflect both the software's direct effect on productivity or revenue and the extent to which it is adopted by other firms. To construct this stock market valuation of software, I match NIIP software registrations to publicly traded firms using tax identifiers and securities identifiers. I then merge these registrations with daily stock prices and the market index. For each software registration, I identify the date its registration was announced by the NIIP and the date that the software was filed. Then, I compute the firm's abnormal return between these two dates. I convert this return into a monetary value using the firm's market capitalization. I express the resulting values in real 1982 U.S. dollars using current exchange rates and the CPI index. Unlike Kogan et al. (2017), I do not adjust the event response for the ex ante probability that a registration occurs, so the measure treats each software publication as a surprise to the market. This assumption affects the level of the estimated values but not their ranking across software registrations. Therefore, since the exposure measure is normalized, this assumption has no effect on the results. All equations follow the stock-market valuation framework in Kogan et al. (2017). Figure A11a plots the distribution of software registration values. I calculate the valuation for 546 software registrations.

Extrapolating software values. I extrapolate software values to registrations owned by firms outside the stock market using smoothed group-mean extrapolation. This procedure uses the subset of software with stock-market-implied values to predict the value of other software with similar observable characteristics. The intuition is that if software in a given application domain, technical class, or programming-language group tends to be more valuable among publicly traded firms, then software with the same characteristics is assigned a higher predicted value. Because many characteristics appear only a few times in the valued sample, I shrink each characteristic-specific mean toward a broader group mean. This prevents small cells from receiving extreme predicted values based on only a few observations.

Let C_s denote the set of characteristics listed by software s, including its application domains, technical classes, and programming-language groups. For each characteristic $c \in C_s$, let $p(c)$ denote its broader group. For example, an exact technical class is mapped to its broader technical-class prefix. I estimate the predicted log value associated with characteristic c as

$$\hat{v}_c = \frac{n_c}{n_c + \kappa}\bar{v}_c + \frac{\kappa}{n_c + \kappa}\hat{v}_{p(c)}. \tag{22}$$

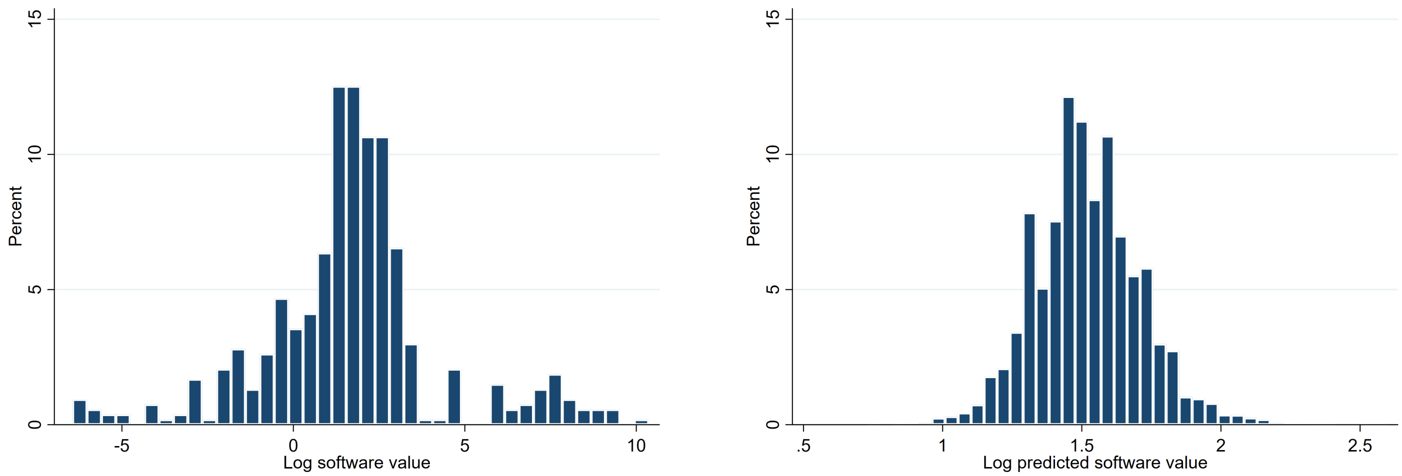
Here, n_c is the number of valued software registrations with characteristic c, $\bar{v}_c$ is their average log value, $\hat{v}_{p(c)}$ is the predicted log value of the broader group, and κ controls the amount of shrinkage. When many valued registrations have characteristic c, the prediction puts more weight on $\bar{v}_c$. When few valued registrations have characteristic c, the prediction puts more weight on the broader group mean. I use $\kappa = 10$.

Because each software registration can list multiple application domains, technical classes, and programming languages, I calculate a prediction for every listed characteristic and then average them. The predicted log value of software s is

$$\hat{v}_s = \frac{1}{N_s}\sum_{c \in C_s} \hat{v}_c, \tag{23}$$

where N_s is the number of characteristics listed for software s. I then convert this predicted log value into a level value using the empirical distribution of observed software values. Figure A11b plots the distribution of estimated software valuations.

Figure A11: **Observed and Extrapolated Software Valuations**



(a) Observed stock-market valuations

(b) Extrapolated valuations

Notes: This figure compares the distribution of observed and extrapolated software valuations. Panel A plots stock-market-implied values for software registrations matched to publicly traded firms. Panel B plots extrapolated values for software registrations without observed stock-market valuations, constructed using the smoothed group-mean extrapolation procedure described in Appendix C.4.

C.5 Instrument Validation

Table A4: **Instrument and Exposure to Other Technologies**

	(1) *Exposure to Patents*	(2) *Exposure to Industrial Design*	(3) *Exposure to Technology Transfer*
AI Ease-of-Dev IV	0.0000872	0.0000324	0.00117
	(0.000106)	(0.000190)	(0.000970)
N	41,136	38,565	41,136

Notes: This table reports estimates from regressions of exposure to other technologies on the AI ease-of-development instrument defined in Equation (8). The dependent variables are occupation-year exposure to patents, industrial designs, and technology transfer, constructed from their titles using the same text-similarity approach as the AI exposure measure. The sample covers 2003–2018. Data come from de Souza (2022). Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**}p < 0.05,^{***}p < 0.01$.

Table A5: **Instrument and Equipment/Tool Import and Export Prices**

	(1) *Log Avg. Import Prices*	(2) *Log Capital Import Prices*	(3) *Log High-Tech Import Prices*	(4) *Log Capital Export Prices*	(5) *Log High-Tech Export Prices*
AI Ease-of-Dev IV	−0.0000313	0.000155	0.000110	0.000182	0.000325*
	(0.0000436)	(0.000126)	(0.0000845)	(0.000190)	(0.000186)
N	51,420	51,420	51,420	51,420	51,420

Notes: This table reports estimates from regressions of occupation-level equipment price measures on the AI ease-of-development instrument defined in Equation (8). To construct prices at the occupation level, I calculate the text similarity between occupational tasks and NCM (Nomenclatura Comum do Mercosul) product descriptions. For each occupation, I keep only products with positive text similarity and average prices using the text similarity as weights. Trade data come from official sources. The sample covers 2003–2023. Columns 1–3 use import prices. Columns 4–5 use export prices. Controls include the lagged dependent variable, exposure to non-AI software, exposure to U.S. AI patents, occupation fixed effects, year fixed effects, and one-digit-occupation-by-year fixed effects. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**}p < 0.05,^{***}p < 0.01$.

Table A6: **Instrument and Trade Shocks**

	(1) *Log Import Prices on Same Sector*	(2) *Log Input Import Prices*	(3) *Log Imports on Same Sector*	(4) *Log Export Prices on Same Sector*
AI Ease-of-Dev IV	0.0179	0.00239	−0.000116	−0.00819
	(0.0126)	(0.00253)	(0.0107)	(0.0124)
N	42,199	45,273	42,205	42,188

Notes: This table reports estimates from regressions of occupation-level trade exposure measures on the AI ease-of-development instrument defined in Equation (8). The trade-shock measures are constructed from official trade statistics at the product-level mapping products to sectors, then aggregated to occupations using sectoral employment shares. The sample covers 2003–2022. Column 1 uses import prices for goods produced by sectors associated with the occupation. Column 2 uses import prices for input goods. Column 3 uses imports of goods produced by sectors associated with the occupation. Column 4 uses export prices for produced goods. Controls include the lagged dependent variable, exposure to non-AI software, exposure to U.S. AI patents, occupation fixed effects, year fixed effects, and one-digit-occupation-by-year fixed effects. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**}p < 0.05,^{***}p < 0.01$.

Table A7: **Instrument and Lagged Labor-Market Outcomes**

	(1) *Log Employment t-2*	(2) *Log Avg. Years of Educ. t-2*	(3) *Log N. HS Dropouts t-2*	(4) *Log N. HS or More t-2*	(5) *Log Wage t-2*	*Log Age t-2*
AI Ease-of-Dev IV	−0.000276	−0.000375	−0.000278	0.000250	0.00179	−0.000580
	(0.00277)	(0.000332)	(0.00361)	(0.00387)	(0.00154)	(0.000463)
N	47,802	47,802	38,995	45,385	47,802	47,802

Notes: This table reports estimates from regressions of lagged labor-market outcomes on the AI ease-of-development instrument defined in Equation (8). The dependent variables come from the RAIS occupation-year panel and are measured two years before the current observation. The regression sample covers 2005–2024. Columns 1–6 report lagged log employment, log average years of education, log employment of high-school dropouts, log employment of workers with high school or more, log average monthly earnings, and log age. Controls include the lagged dependent variable, exposure to non-AI software, exposure to U.S. AI patents, occupation fixed effects, year fixed effects, and one-digit-occupation-by-year fixed effects. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**}p < 0.05,^{***}p < 0.01$.

Table A8: **Instrument and Labor-Inspection Outcomes**

	(1) *Labor Infraction*	(2) *Informality Infraction*	(3) *Safety/Health Infraction*	(4) *Hours/Rest Infraction*	(5) *Wage Payment Infraction*
AI Ease-of-Dev IV	0.00155	0.00123	0.000219	0.00191	0.00117
	(0.00195)	(0.00133)	(0.00165)	(0.00172)	(0.00160)
N	23,797	23,797	23,797	23,797	23,797

Notes: This table reports estimates from regressions of labor-inspection outcomes on the AI ease-of-development instrument defined in Equation (8). The dependent variables are the share of occupations in firms found with labor infractions, infractions related to informality, safety and health, hours and rest, and wage payment, constructed from Brazilian labor-inspection records. The sample covers 2004–2016. Controls include the lagged dependent variable, exposure to non-AI software, exposure to U.S. AI patents, occupation fixed effects, year fixed effects, and one-digit-occupation-by-year fixed effects. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**}p < 0.05,^{***}p < 0.01$.

C.6 Placebo Tests

Table A9: **First Stage Using Stack Overflow Posts in Other Languages**

	(1) *AI Exposure*	(2) *AI Exposure*	(3) *AI Exposure*	(4) *AI Exposure*	(5) *AI Exposure*
AI Ease-of-Dev IV	0.866***	0.886***	1.125***	0.889***	1.134***
	(0.0239)	(0.0274)	(0.202)	(0.0405)	(0.198)
Japanese Stack Overflow IV		−0.0323			−0.0304
		(0.0232)			(0.0232)
Russian Stack Overflow IV			−0.262		−0.251
			(0.200)		(0.198)
Principal Component				−0.0269	
				(0.0421)	
N	18,032	18,032	18,032	18,032	18,032
R^2	0.589	0.589	0.590	0.589	0.590

Notes: This table reports first-stage estimates of the relationship between the AI ease-of-development instrument and AI exposure. The dependent variable is AI exposure from Brazilian software registrations from NIIP, defined in Equation (5). The AI ease-of-development instrument combines mailing-list and Stack Overflow AI knowledge by programming language. The other-language instruments are built from local-language Stack Overflow sites using local-language tags. The sample covers 2016–2022. Column 1 includes no other-language instrument controls. Columns 2 and 3 add the Japan and Russia instruments one at a time. Column 4 adds the first principal component of language-specific instruments. Column 5 includes the Japan and Russia instruments jointly. Each instrument is constructed following Equation (8). All columns include occupation fixed effects, year fixed effects, and one-digit-occupation-by-year fixed effects. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

Table A10: **Pre-Boom Placebo Estimates**

	(1) *log(Employment)*	(2) *log(Years of Education)*	(3) *log(Age)*	(4) *log(Experience)*	(5) *log(Wage)*
Panel A: 2003–2010					
AI Exposure	−0.169	−0.0160	0.000296	0.159	−0.0195
	(0.209)	(0.0175)	(0.0203)	(0.0990)	(0.0835)
N	17,079	17,079	17,079	17,079	17,079
Panel B: Pre-2003					
AI Exposure	−0.325	−0.000800	0.0636*	0.165	−0.0912
	(0.254)	(0.0437)	(0.0384)	(0.222)	(0.187)
N	9,576	9,576	9,576	9,576	9,576

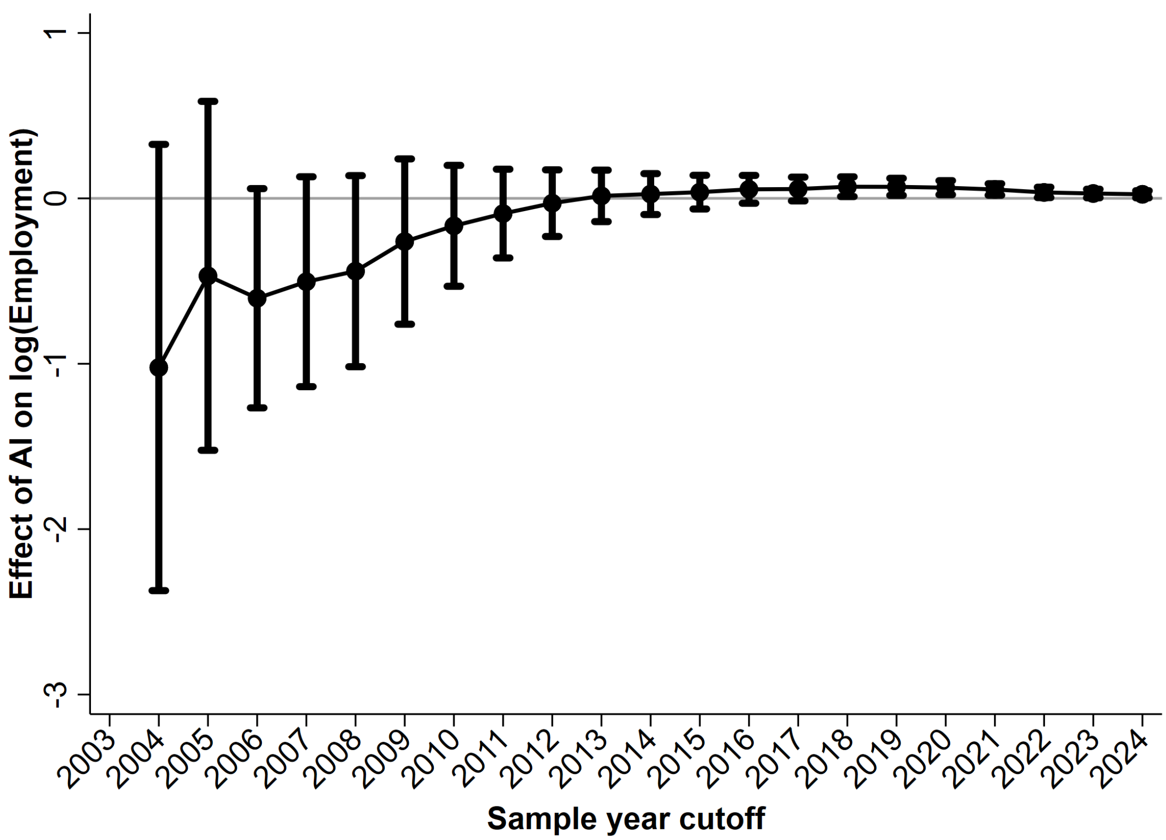
Notes: This table reports 2SLS estimates of the effect of AI exposure on RAIS labor-market outcomes, using the AI ease-of-development instrument defined in Equation (8). AI exposure is constructed from Brazilian software registrations from NIIP. Panel A uses the 2003–2010 sample. Panel B uses the pre-2003 sample, 1995–2002. Controls include the lagged dependent variable, exposure to non-AI software, exposure to U.S. AI patents, occupation fixed effects, year fixed effects, and one-digit-occupation-by-year fixed effects. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

Table A11: **First Stage with U.S. Labor-Market Controls**

	(1) *AI Exposure*	(2) *AI Exposure*	(3) *AI Exposure*	(4) *AI Exposure*	(5) *AI Exposure*
AI Ease-of-Dev IV	0.379***	0.379***	0.379***	0.379***	0.379***
	(0.0170)	(0.0170)	(0.0169)	(0.0170)	(0.0169)
Log U.S. Employment$_{o,t}$		0.0100			0.00427
		(0.0116)			(0.0114)
Log U.S. Wage$_{o,t}$			0.0699**		0.0648**
			(0.0328)		(0.0320)
Log U.S. Education$_{o,t}$				0.0625	0.0173
				(0.0644)	(0.0576)
N	56,568	56,568	56,568	56,568	56,568
R^2	0.597	0.597	0.597	0.597	0.597
F	898.3	681.1	677.0	686.3	459.6

Notes: This table reports first-stage estimates of the relationship between the AI ease-of-development instrument and AI exposure. The dependent variable is AI exposure from Brazilian software registrations from NIIP, defined in Equation (5). The AI ease-of-development instrument combines Gmane mailing-list and Stack Overflow AI knowledge by programming language. U.S. labor-market controls come from the Current Population Survey and are mapped to Brazilian occupations. The sample covers 2003–2024. Column 1 reports the baseline first stage. Columns 2–4 add U.S. employment, wage, and education controls one at a time. Column 5 includes all three U.S. labor-market controls jointly. All columns control for exposure to non-AI software, exposure to U.S. AI patents, occupation fixed effects, year fixed effects, and one-digit-occupation-by-year fixed effects. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10$, $^{**}p < 0.05$, $^{***}p < 0.01$.

Figure A12: **Effect of AI on Employment by Sample Cutoff Year**



Notes: This figure reports 2SLS estimates of the effect of AI on log employment using samples that begin in 2003 and end in the year shown on the x-axis. Each point is the coefficient on AI exposure, instrumented with the AI ease-of-development instrument defined in Equation (8). Controls include lagged log employment, exposure to non-AI software, exposure to U.S. AI patents, occupation fixed effects, year fixed effects, and one-digit-occupation-by-year fixed effects. Standard errors are clustered at the occupation level. Vertical bars report 95% confidence intervals.

C.7 Mincer Regression to Identify Abilities

To estimate individual ability, I run a Mincer regression using the version of RAIS with individual and firm identifiers. The dataset runs from 2003 to 2016, capturing only the initial years of the AI boom. The specification is given by

$$log(Wages_{i,t}) = \mu_i + X_{i,t} + \epsilon_{i,t}$$

where $Wages_{i,t}$ is monthly wages and μ_i is a worker fixed effect. $X_{i,t}$ is a set of controls including a 6-digit occupation fixed effect, an education-year fixed effect, an age fixed effect, a gender fixed effect, a 4-digit sector-year fixed effect, and a city-year fixed effect.

To calculate the average ability in occupation o in year t, used in Table 4, I average the fixed effect μ_i for workers in occupation o in year t. The wage residual in occupation o and year t is given by the average of $\epsilon_{i,t}$ for workers in that occupation in that year.

C.8 Task Group Construction

I construct task groups from the CBO 2002 task descriptions using the BERTopic pipeline (Grootendorst 2022). The starting point is the universe of 49,660 task descriptions. First, I represent each task as a sentence embedding using paraphrase-multilingual-MiniLM-L12-v2, a multilingual Sentence-BERT model (Reimers and Gurevych 2019). Sentence embeddings map each task into a vector space in which tasks with similar meanings are close to one another, even if they use different words. Second, I reduce these high-dimensional embeddings to five dimensions using UMAP, a nonlinear dimension-reduction method designed to preserve local neighborhoods in the data (McInnes et al. 2018). This step makes the clustering problem tractable while keeping nearby tasks close together. Third, I cluster the reduced embeddings using HDBSCAN, a density-based clustering algorithm (Campello et al. 2013, 2015). Unlike k-means, HDBSCAN does not require me to choose the number of clusters in advance and does not force every task into a group. Instead, it forms a group only when there is a sufficiently dense set of similar tasks; tasks in sparse regions are labeled as unassigned.

In the baseline classification, I require each HDBSCAN cluster to contain at least 30 tasks. This procedure yields 700 task groups and leaves 8,019 highly idiosyncratic tasks unassigned rather than forcing them into weakly related groups. I then name each task group using BERTopic's class-based TF-IDF procedure, or c-TF-IDF (Grootendorst 2022), which builds on the standard TF-IDF logic of giving high weight to words that are common in one document but rare in the broader corpus (Salton and Buckley 1988). In this setting, each task group is treated as a document. A word receives a high score if it appears frequently in that group but rarely in other groups. The resulting names are descriptive labels used to interpret the groups; they are not regression inputs. Finally, I map the task groups back to occupations and construct two occupation-level variables for each group: an indicator equal to one if the occupation contains at least one task in the group, and the share of the occupation's tasks that belong to the group.

C.9 Construction of the Information Use Measure

I construct an occupation-level measure of information use from the CBO task descriptions. The goal is to distinguish tasks in which workers use information to guide actions or decisions from tasks in which workers mainly process, record, classify, or organize information. The classification follows the same logic as the text-similarity approach in de Souza and Li (2023): I define reference texts for two conceptual categories, represent both tasks and reference texts in a common semantic space, and classify each task according to which reference category it is closer to.

First, I collect a set of benchmark activities describing tasks intensive in data use or intensive in data processing. The information-use benchmark uses the Wikipedia pages Instrumentação industrial, Controle estatístico de processos, Metrologia, Controle de qualidade, Telemetria, Manutenção preditiva, Agricultura de precisão, Monitoramento ambiental, Sensoriamento remoto, Sistema de posicionamento global, and Inteligência empresarial. The information-processing benchmark uses the Wikipedia pages Tratamento de dados, Processamento eletrônico de dados, Gerenciamento eletrônico de documentos, Banco de dados, Sistema de gerenciamento de banco de dados, Arquivística, Arquivo, Documento, Sistema de ficheiros, and Gestão de conteúdo empresarial.

I convert both the CBO task descriptions and the Wikipedia benchmark texts into sentence embeddings using the multilingual Sentence-BERT model (Reimers and Gurevych 2019, 2020). A sentence embedding maps each text into a numerical vector, with texts that have similar meanings located closer to one another. For each Wikipedia page, I split the page into text chunks, compute embeddings for the chunks, and average them to form one vector for the page. I then average the page vectors within each benchmark set to obtain two centroids: one for information use and one for information processing.

For each CBO task τ, I compute two cosine similarities: one between the task embedding and the information-use centroid, and one between the task embedding and the information-processing centroid. I define the task-level score as

$$Score_\tau = Sim(\tau, Use) - Sim(\tau, Processing).$$

A task is classified as information use if $Score_\tau > 0$, and as information processing otherwise.

I aggregate the task classifications to the occupation level. Let N_o be the number of distinct tasks in occupation o. The information use intensity of occupation o is

$$Information\ Use\ Intensity_o = \frac{1}{N_o} \sum_{\tau \in o} \mathbb{1}\{Score_\tau > 0\}.$$

C.10 Other Results

Table A12: **AI Programming Posts and AI Software Development**

	(1) log(N. Software + 1)	(2) log(N. Software + 1)	(3) log(N. Software + 1)	(4) log(N. Software + 1)
log(N. AI Posts + 1)	0.289***	0.0829***	0.289***	0.0667***
	(0.00482)	(0.00682)	(0.00483)	(0.00698)
N	10,115	10,115	10,115	10,115
R^2	0.263	0.742	0.268	0.750
F	3,611.4	147.5	3,581.6	91.42
Fixed Effects:				
Language	-	✓	-	✓
Year	-	-	✓	✓

Notes: This table reports language-year regressions relating AI software development to AI programming posts. The dependent variable is the log number of AI software registrations plus one by programming language and year. The regressor is the log number of AI-related mailing-list and Stack Overflow posts plus one. The sample covers 2008–2024. Standard errors appear in parentheses. $^{*}p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

Table A13: **AI Programming Posts and AI Software Development by Application Domain**

	(1) log(N. Software + 1)	(2) log(N. Software + 1)	(3) log(N. Software + 1)	(4) log(N. Software + 1)
log(N. AI Posts + 1) × *AI Software Share*	1.001***	1.295***	1.046***	1.650***
	(0.00681)	(0.0165)	(0.0150)	(0.0226)
N	191,709	191,709	191,709	191,709
R^2	0.101	0.273	0.424	0.518
F	21,608.6	6,153.6	4,875.5	5,345.1
Fixed Effects:				
Language	-	✓	✓	-
Field	-	✓	✓	-
Year	-	-	✓	✓
Language-Field	-	-	-	✓

Notes: This table reports estimates of the relationship between Stack Overflow exposure and AI software development at the language-domain-year level. The dependent variable is log number of AI software registrations plus one. The Stack Overflow exposure variable is the log number of AI Stack Overflow posts for a programming language interacted with the language's AI software share within an application domain. Standard errors appear in parentheses. $^{*}p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

Table A14: **AI Packages and AI Programming Posts**

	(1) *log(N. AI Posts + 1)*	(2) *log(N. AI Posts + 1)*	(3) *log(N. AI Posts + 1)*	(4) *log(N. AI Posts + 1)*
log(N. AI Packages + 1)	0.219^{***}	0.0799^{***}	0.216^{***}	0.0444^{***}
	(0.00546)	(0.00807)	(0.00545)	(0.00799)
N	10,115	10,115	10,115	10,115
R^2	0.137	0.817	0.146	0.828
F	1,607.8	98.17	1,571.4	30.96
Fixed Effects:				
Language	-	✓	-	✓
Year	-	-	✓	✓

Notes: This table reports estimates of the relationship between AI packages and AI programming posts at the language-year level. The dependent variable is the log number of AI-related mailing-list and Stack Overflow posts plus one. The main regressor is the log number of AI packages in each programming language plus one. Package counts are constructed from the Ecosyste.ms Packages open-data dump, 2026-02-05 release, which aggregates package, version, dependency, and metadata information from major open-source package registries such as PyPI, npm, Maven, CRAN, RubyGems, NuGet, Packagist, Go proxy, and related ecosystems. I classify packages as AI-related using package names, descriptions, keywords, repository metadata, and dependency names. Standard errors appear in parentheses. $^*p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

Table A15: **AI Increases Hiring and Firing**

	(1) *log(Hiring Rate)*	(2) *log(Firing Rate)*	(3) *log(Hiring)*	(4) *log(Firing)*
AI Exposure	0.0561^{***}	0.0341^{*}	0.0754^{***}	0.0652^{***}
	(0.0184)	(0.0187)	(0.0167)	(0.0174)
N	47,274	46,973	49,864	49,463

Notes: This table reports 2SLS estimates of the effect of AI on hiring and firing, following the specification in Equation (6). Each column uses the AI ease-of-development instrument defined in Equation (8). Columns 1 and 2 use the log hiring and firing rates, defined as hires and firings divided by lagged employment. Columns 3 and 4 use the log number of hires and firings. Controls include occupation fixed effects, year fixed effects, one-digit-occupation-by-year fixed effects, the lagged outcome, non-AI software exposure, and exposure to U.S. AI patents. Standard errors, clustered at the occupation level, appear in parentheses. $^*p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

Table A16: **Effects of Generative and Predictive AI on Worker Outcomes**

	(1) *log(Employment)*	(2) *log(Age)*	(3) *log(Yrs. Education)*	(4) *log(Experience)*	(5) *log(Wage)*
Gen. AI Exposure	-0.0161^{**}	0.00216^{***}	0.000905	0.0113^{***}	0.00305
	(0.00782)	(0.000829)	(0.000863)	(0.00426)	(0.00298)
Pred. AI Exposure	0.0348	-0.00465	-0.00319	-0.0320^{*}	-0.00539
	(0.0239)	(0.00334)	(0.00285)	(0.0175)	(0.0109)
N	50,190	50,190	50,190	50,190	50,190

Notes: This table reports 2SLS estimates of the effects of generative and predictive AI exposure on worker outcomes. Column 1 reports log employment; Column 2 log average age; Column 3 log average years of education; Column 4 log average experience; and Column 5 log average wage. The two AI exposure measures are instrumented with the corresponding ease-of-development instruments for generative and predictive AI software. Controls include occupation fixed effects, year fixed effects, one-digit-occupation-by-year fixed effects, exposure to U.S. AI patents, exposure to non-AI software, the non-AI component of the ease-of-development instrument, and the lagged outcome variable. Standard errors, clustered at the occupation level, appear in parentheses. $^*p < 0.10,^{**} p < 0.05,^{***} p < 0.01$.

Table A17: **Robustness: AI Increases Employment of Low-Skilled Workers**

	(1) log(Employment)	(2) log(Age)	(3) log(Yrs. Education)	(4) log(Tenure)	(5) log(Wage)	(6) log(P90/P10 Wage)
Panel A: Control for U.S. Labor Market Outcomes						
AI Exposure	0.0217***	−0.00316***	−0.00215**	−0.0130**	0.00158	−0.0174**
	(0.00707)	(0.00105)	(0.000851)	(0.00571)	(0.00336)	(0.00851)
Panel B: Occupation-Industry Level						
AI Exposure	0.0371***	−0.00219***	−0.00424***	−0.00222	−0.00479***	0.00619***
	(0.00261)	(0.000492)	(0.000441)	(0.00231)	(0.00124)	(0.00228)
Panel C: Additional Outcome Lags						
AI Exposure	0.0189***	−0.00241***	−0.00202***	−0.0100*	0.00139	−0.0118
	(0.00668)	(0.000916)	(0.000745)	(0.00526)	(0.00304)	(0.00734)
Panel D: No Controls						
AI Exposure	0.0448***	−0.00390***	−0.00240***	−0.0241***	−0.00855***	−0.0310***
	(0.00983)	(0.000978)	(0.000704)	(0.00468)	(0.00325)	(0.00460)

Notes: This table reports robustness checks of the effect of AI exposure on labor-market outcomes. Each coefficient comes from a separate 2SLS regression using the AI ease-of-development instrument. Column 1 reports log employment; Column 2 log average age; Column 3 log average years of education; Column 4 log average tenure; Column 5 log average wage; and Column 6 the log p90-to-p10 wage ratio. Panel A controls for the U.S. version of each outcome; because the source output does not report tenure, Column 4 uses the baseline tenure estimate. Panel B estimates the model at the occupation-industry level and combines the occupation-industry outcome table with the corresponding p90-to-p10 wage-ratio table. Panel C includes an additional lag of the dependent variable. Panel D excludes the lagged dependent variable, exposure to non-AI software, and exposure to U.S. AI patents. Unless otherwise noted, controls include occupation fixed effects, year fixed effects, one-digit-occupation-by-year fixed effects, exposure to U.S. AI patents, exposure to non-AI software, and the lagged outcome variable. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10$, $^{**}p < 0.05$, $^{***}p < 0.01$.

Table A18: **Alternative AI Software Exposure Measures**

	(1) log(Employment)	(2) log(Age)	(3) log(Yrs. Education)	(4) log(Tenure)	(5) log(Wage)	(6) log(P90/P10 Wage)
Panel A: Unweighted AI Exposure						
AI Exposure	0.0208***	−0.00344***	−0.00188**	−0.0140**	0.00139	−0.0192**
	(0.00690)	(0.00103)	(0.000853)	(0.00561)	(0.00335)	(0.00842)
Panel B: Breakthrough AI Exposure						
AI Exposure	0.0174***	−0.00254***	−0.00161**	−0.0110***	−0.00125	−0.0155**
	(0.00543)	(0.000812)	(0.000677)	(0.00426)	(0.00248)	(0.00654)

Notes: This table reports 2SLS estimates of the effect of AI exposure on labor-market outcomes using alternative measures of AI software exposure. Each coefficient comes from a separate regression using the corresponding AI ease-of-development instrument. Column 1 reports log employment; Column 2 log average age; Column 3 log average years of education; Column 4 log average tenure; Column 5 log average wage; and Column 6 the log p90-to-p10 wage ratio. Panel A uses an exposure measure with equal weights to each software, i.e., $v_s = 1$ in Equation (5). Panel B follows Kelly et al. (2021) and weights AI software registrations by their importance. Section C.11 describes the construction of this measure in detail. Controls include occupation fixed effects, year fixed effects, one-digit-occupation-by-year fixed effects, exposure to U.S. AI patents, exposure to non-AI software, and the lagged outcome variable. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10$, $^{**}p < 0.05$, $^{***}p < 0.01$.

Table A19: **Main Results Under Alternative Instruments**

	(1) log(Employment)	(2) log(Age)	(3) log(Yrs. Education)	(4) log(Tenure)	(5) log(Wage)	(6) log(P90/P10 Wage)
Panel 1: Bartik Instrument						
AI Exposure	0.00842*	−0.00178***	−0.00248***	−0.00925***	−0.00320	−0.00421
	(0.00432)	(0.000610)	(0.000490)	(0.00344)	(0.00196)	(0.00442)
Panel 2: Number of Packages						
AI Exposure	0.0199**	−0.00480***	−0.00299***	−0.0160**	0.00357	−0.0175*
	(0.00822)	(0.00135)	(0.000969)	(0.00717)	(0.00420)	(0.0102)
Panel 3: Stack Overflow						
AI Exposure	0.0203***	−0.00287***	−0.00227***	−0.0127**	0.000898	−0.0127*
	(0.00672)	(0.000996)	(0.000794)	(0.00542)	(0.00315)	(0.00773)

Notes: This table reports 2SLS estimates of the effect of AI exposure on labor-market outcomes using alternative instruments. Each coefficient comes from a separate regression. Column 1 reports log employment; Column 2 log average age; Column 3 log average years of education; Column 4 log average tenure; Column 5 log average wage; and Column 6 the log p90-to-p10 wage ratio. Panel 1 instruments AI exposure with the Bartik instrument. Panel 2 instruments AI exposure with the number of packages. Panel 3 instruments AI exposure with Stack Overflow posts. Controls include occupation fixed effects, one-digit-occupation-by-year fixed effects, exposure to U.S. AI patents, exposure to non-AI software, and the lagged outcome variable. Standard errors, clustered at the occupation level, appear in parentheses. $^{*}p < 0.10$, $^{**}p < 0.05$, $^{***}p < 0.01$.

C.11 Software Importance

Following Kelly et al. (2021), I construct a measure of AI software exposure that gives more weight to software registrations that are more important for later AI development. The measure follows the logic of the patent breakthrough measure in Kelly et al. (2021): important innovations are those that differ from prior work but are close to later work. I adapt this idea to software by comparing each AI software registration to earlier and later AI software registrations using text similarity.

For each AI software registration, I construct a description by concatenating the software title, technical-class description, and application-domain description. I then apply the same text-similarity procedure used to construct the baseline AI exposure measure and calculate the cosine similarity between each pair of AI software registrations.

Let s denote an AI software registration published in year t, and let $\rho_{s,j}$ denote the cosine similarity between software s and another AI software registration j. I define backward similarity, B_s, and forward similarity, F_s, as

$$B_s = \sum_{j:\max(t_{\min},t-5)\leq t_j \leq t-1} \max\{\rho_{s,j}, 0\}, \tag{24}$$

$$F_s = \sum_{j:t+1\leq t_j \leq \min(t_{\max},t+20)} \max\{\rho_{s,j}, 0\}. \tag{25}$$

The software importance measure is the ratio of forward to backward similarity:

$$\textit{Software Importance}_s = \frac{F_s}{B_s}. \tag{26}$$

The measure is high when a software registration is relatively different from prior AI software but close to future AI software. It therefore captures the idea of a breakthrough software registration: a program that departs from the existing stock of AI software and is followed by similar later registrations. To calculate the measure for all software in the data, I use the years available in the sample rather than requiring the full five-year backward and twenty-year forward windows.