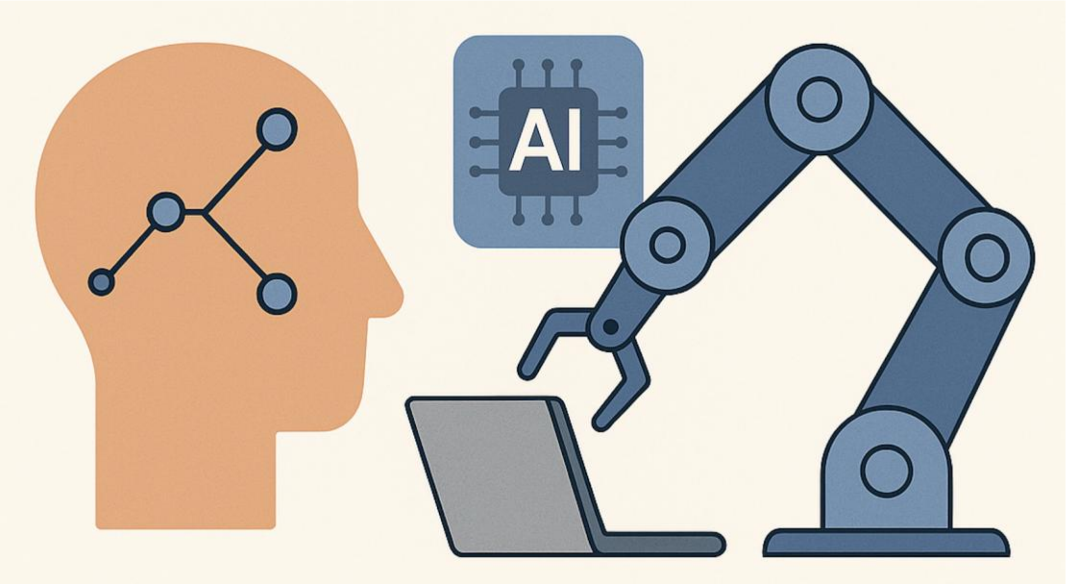
A Mathematical Framework for AI-Human Integration in Work

L. Elisa Celis
Yale University

Lingxiao Huang
Nanjing University

Nisheeth K. Vishnoi
Yale University



How can we compare workers—human, AI, or both—on the same job?

ICML 2025

Paper : https://arxiv.org/abs/2505.23432

Motivation and Related Work

GenAI tools like GPT-4 and Gemini are transforming tasks: summarization, code, writing (OpenAI, 2023; DeepMind, 2023)

Dario Amodei — CEO of Anthropic, one of the world's most powerful creators of artificial intelligence — has a blunt, scary warning for the U.S. government and all of us:

- AI could wipe out *half* of all entry-level white-collar jobs — and spike unemployment to 10-20% in the next one to five years, Amodei told us in an interview from his San Francisco office.

BBC

AI could replace equivalent of 300 million jobs - report

29 March 2023

Share Save

Can GenAI enhance workers—or only replace them?

Empirical studies:

- [Brynjolfsson et al. 2023]: GenAI boosts productivity, esp. for junior workers
- [Vaccaro et al. 2024]: Gains vary by task type—stronger in content than decision tasks
- [Jaffe et al. 2024]: Human-AI collaboration helps, but depends on complementarity

But missing:

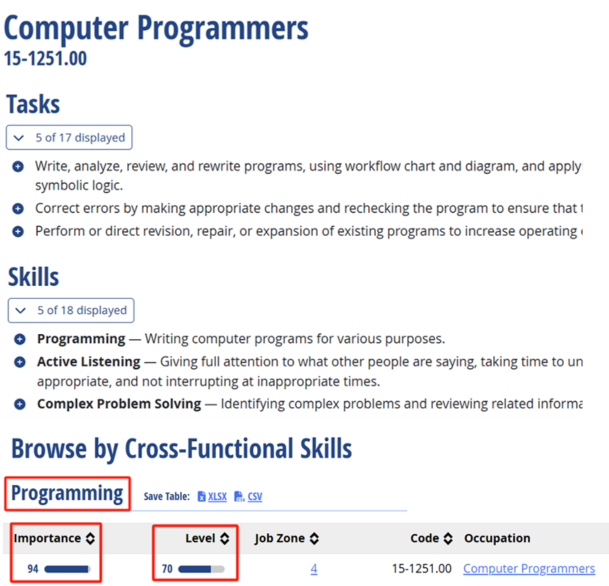
- A formal model of jobs and worker-AI fit
- A framework that explains why gains happen and when they fail

Why Evaluations Fail — An Example

Job structure is underspecified

Example: O*NET

A comprehensive database, maintained by the U.S. Department of Labor, provides standardized descriptions of >1000 jobs

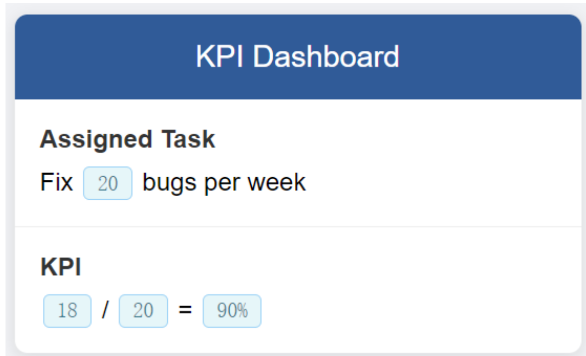


Challenges:

- *How tasks depend on skills?*
- *How to evaluate performance at the level of a skill, task, job*

Human eval conflate subskills

Example: KPI



Problems:

- Subskills Involved:
  - Diagnose (reasoning)
  - Fix + test code (execution)
- Same score ≠ same skills
- Failures are uninterpretable

Challenges:

- *Conflate reasoning with execution*
- *Lack of standardization*
- *Obscure where intervention is needed for upskilling*

AI benchmarks eval isolated skills

Example: Big-Bench Lite

```
x = 5
y = 3
z = 2
x = y + x

What is the value of x at line 3?
```

Expected output:

```
5
```

What's missing:

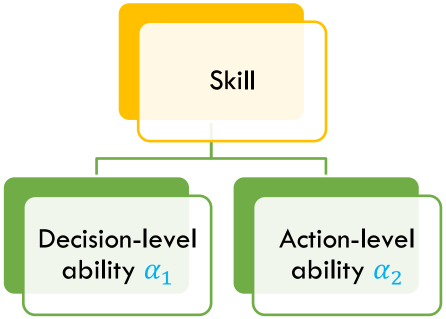
- No diagnosis, prioritization, or multi-step task context
- No way to assess judgment or adaptation
- No notion of job-level success

Challenges:

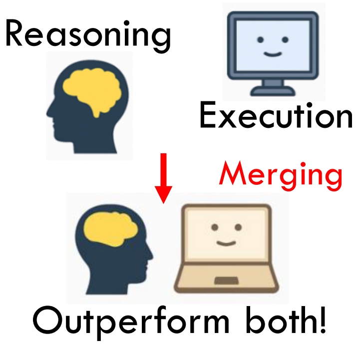
- *AI is evaluated on fragments*
- *Statistical noise in evaluation*

Our Contributions

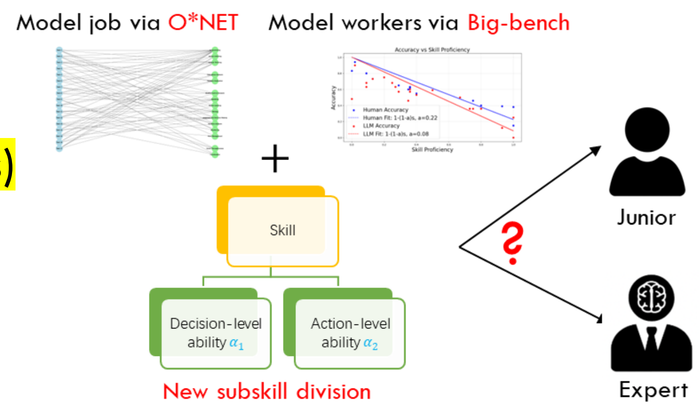
1. **A unified framework for modeling and measuring job fit**
   - Represents jobs as task-skill dependency graphs
   - Models worker ability via decision- and action-level subskills
   - Captures performance using probabilistic ability profiles
   - Computes job success probability from noisy subskill draws
   - Enables comparison across humans, AI systems, and hybrids



2. **Theoretical insights**
   - **Phase transition:** small improvements → big jumps in success
   - **Merging theorem:** combining complementary workers can outperform individuals – GenAI enhance, no replace!
   - Explains "productivity compression" via AI assistance



3. **Empirical use cases**
   - Framework's usability via data derived from O*NET (human jobs) and Big-Bench Lite (GenAI tools)
   - Explains human-AI partnership gains
   - Informs training, upskilling, and hiring strategies



A unified framework to analyze and improve job performance across human, AI, and hybrid workers

A Probabilistic Model of Job Success

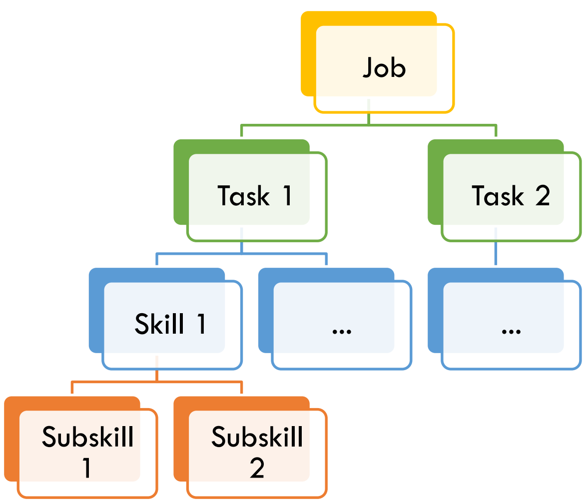
Job = collection of tasks

Each task is associated with a collection T_i of multiple skills

Key idea: Each skill decomposed into 2 subskills: decision v.s. action [Kahneman 2011, Inga et al. 2023]

E.g. "coding" involves both "solving the problem" (decision-level) and "implementing a solution in a language" (action-level)

Like from O*NET, each subskill is associated with a difficulty in $[0,1]$
0: easiest, 1: hardest



We model a worker by two ability profiles: (α_1, α_2)

- α_1: decision-level subskills
- α_2: action-level subskills

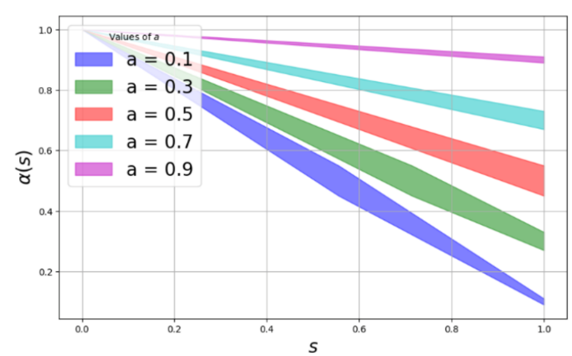
$\alpha(s)$ maps subskill difficulty $s \in [0,1]$ to a probability distribution over $[0,1]$

Each draw from $\alpha(s)$ gives performance on that subskill

$\alpha(s)$ contain two parts: an average ability $E(s) \in [0,1]$ and an additive stochastic noise term $\varepsilon(s)$ (subskill independent)

Linear: $E(s) = 1 - (1-a)s$, fitting [BIG bench authors 2023]

Noise models: Uniform / Truncated normal



Job success metrics

Subskill level

- Random subskill error rate $\zeta_{j\ell} = 1 - X$ where $X \sim \alpha_\ell(s_{j\ell})$, representing failure probability

Skill level

- Aggregates subskill errors ζ_{j1} and ζ_{j2} to an overall skill error rate via $h: [0,1]^2 \to [0,1]$
- E.g., $h(a,b) = (a+b)/2$

Task level

- Each task T_i depends on multiple skills. Aggregate skill errors via: $g: [0,1]^* \to [0,1]$

Job level

- Aggregate task errors via a job error function $f: [0,1]^m \to [0,1]$

Job-worker fit metric

- Define overall error: $\mathrm{Err}(\zeta) := f\left(g\left(\left\{h(\zeta_{j1}, \zeta_{j2})\right\}_{j \in T_i, i \in [n]}\right)\right)$
- Job success probability:

$$P := \Pr_\zeta[\mathrm{Err}(\zeta) \le \tau]$$

Theoretical Results

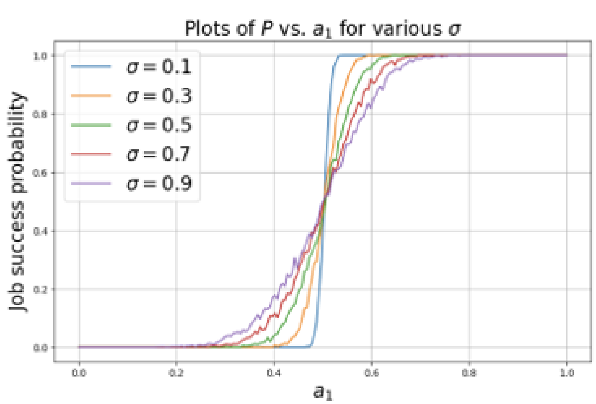
Fix a job profile (task-dependency T_i, subskill difficulties $s_{j\ell}$, job error Err, threshold τ)

Analyzing job-worker fit: phase transition

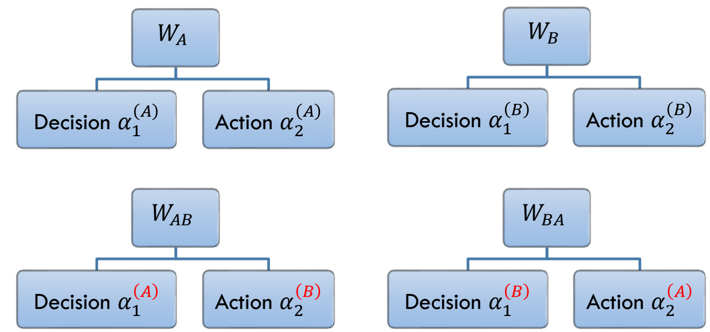
Theorem: Let $\mathrm{Err}(\zeta) = \frac{1}{2n}\sum_j(\zeta_{j1} + \zeta_{j2})$, $s_{j\ell} \sim \mathrm{Unif}[0,1]$. Suppose $\alpha_\ell(s)$ is linear ability profile with ability parameter a_ℓ and noise rate σ. Fix a_2, σ and θ. Then, increasing a_1 by an amount of $\gamma_1 = \sigma\sqrt{\ln(1/\theta)/n}$ increases P from θ to $1-\theta$

Implications:

- *Small changes in ability parameter can cause sharp jumps in job success. Transition window γ_1 depends on the choice of job and ability profiles*
- *Helps explain emergence of GenAI's power*
- *Biased ability evaluations may be exclusionary*



Analyzing human-AI partnership



Whether and when the success prob. of best-merged worker is (significantly) higher than W_A and W_B?

Theorem: If $a_1^{(A)} \geq a_1^{(B)} + \sigma\sqrt{\ln(1/\theta)/n}$ and $a_2^{(B)} \geq a_2^{(A)} + \sigma\sqrt{\ln(1/\theta)/n}$. Then best-merged worker has job-success probability $\geq 1-\theta$ while both W_A and W_B have job-success probability $\leq \theta$

Implications:

- *Merging two workers with complementary skills can result in a significant performance gain*
- *Capture human-AI partnership, where human excels in decision and GenAI excels in action*
- *Productivity compression effect [Brynjolfsson et al.]*

Thresholds and complementarity reshape how we think about skill, success, and augmentation

Empirical Results

Framework's usability (Computer Programmer)

Deriving job data (from O*NET):

- Descriptions of $n = 18$ skills and $m = 17$ tasks
- Proficiency levels $s \in [0,1]$ for each skill
- Skill and task importance scores, inform the choice of error function Err being "weighted average"
- Developing new methods for task-skill dependency graph and subskill division

Deriving workers' abilities (from Big-bench Lite):

- Model abilities of both human and GenAI by linear ability + truncated normal noise

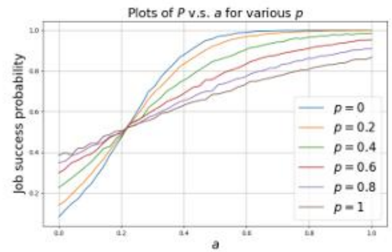
Skill id	Skill name	Importance ($w\%$)	Proficiency ($s\%$)	Decomposition (λ)	Decision (s_{j1})	Action (s_{j2})
1	Coordination	50	41	0	0	0.41
2	Social Perceptiveness	53	43	0	0	0.43
3	Mathematics	53	45	1	0.45	0
4	Time Management	53	45	1	0.45	0
5	Monitoring	50	45	1	0.45	0
6	Systems Analysis	60	45	0.6	0.27	0.18
7	Judgment and Decision Making	56	46	0.7	0.322	0.138
8	Writing	56	46	0.4	0.184	0.276
9	Active Learning	56	46	0.4	0.184	0.276
10	Speaking	53	48	0	0	0.48
11	Quality Control Analysis	63	50	0.3	0.15	0.35
12	Reading Comprehension	60	50	1	0.5	0
13	Systems Evaluation	53	52	1	0.52	0
14	Operations Analysis	53	54	0.6	0.324	0.216
15	Complex Problem Solving	69	55	0.7	0.385	0.165
16	Critical Thinking	69	55	0.6	0.33	0.22
17	Active Listening	69	57	0	0	0.57
18	Programming	94	70	0.4	0.28	0.42

Data from O*NET — Subskill division (new)

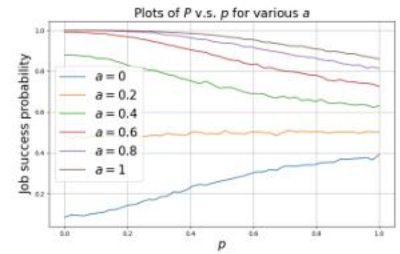
Robustness of theoretical results

Phase transition with dependent subskills

- In practice, a worker's current state may influence their abilities, creating dependencies between $\zeta_{j\ell}$
- Introduce dependency $p \in [0,1]$ 0: independent



(a) P v.s. a



(b) P v.s. p

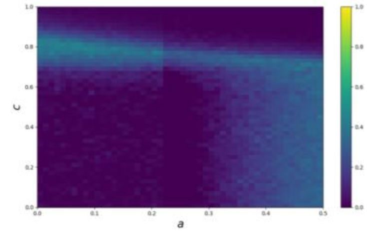
Observation: Sharp thresholds confirmed (smoother)

Merging improves success with distinct profiles

- Human: linear v.s. GenAI: constant ($E(s) \equiv c$)
- Each subskill handled by higher-ability one

Observations:

- Non-identical merging works, brings a sharp prob. gain Δ
- Transition is smoother (narrow bright region in heatmap)



(b) Heatmap of Δ

Our model predicts success, explains gaps, and guides augmentation across humans and AI

Takeaways, Summary, and Future Work

1. Jobs are layered

- Skills are not flat collections of tasks. They are layered systems of judgment and execution

2. Success is structured, not smooth

- Our model reveals sharp thresholds: Small upskilling in ability can dramatically boost outcomes

3. Augmentation, not replacement

- Humans and AI have complementary strengths: AI reduces execution noise and humans provide strategic adaptation. Our metric quantifies when teams outperform individuals

4. Train to decide, not just to do

- Upskilling must focus on decision-level abilities: framing problems, evaluating tradeoffs, etc.. These are harder to automate—and more valuable.

5. Measure what matters

- Traditional evaluation systems flatten talent. Our model enables fine-grained assessment and targeted support, unlocking hidden potential and informing better design of institutions.

Summary

- We introduced a probabilistic model of worker performance
- Incorporated decision- and action-level subskills
- Defined a success probability metric for any job-worker pairing
- Showed theoretical phenomena: phase transitions, probability gain by merging
- Showed usability with data derived from O*NET and Big-Bench Lite

Limitations and future work

- Extend beyond job success by integrating additional factors (e.g., efficiency, time, cost) of worker-job fit
- Use more complex benchmarks (e.g., HumanEval) to better reflect real-world task difficulty
- Refine models, draw on behavioral insights, and design for equitable human-AI collaboration …

Thank you!