

# The Economics of Generative AI

John Horton  
MIT & NBER

July 17, 2023

## Abstract

The distinctive feature of generative AI is that it generates: code, writing, images, music, videos, and so on. It does this at essentially zero marginal creation cost. These generations are made in response to some request or “prompts,” but these AI generations only become economic goods if judged suitable for some human need. This depends on human judgment in evaluation, the skill in crafting the prompt, and the technical capability of the generative model jointly determines the productivity. When deciding to route a task to an AI, the question is whether the prompt-evaluation loop inherent in using that AI will be more efficient than simply doing the task by hand. Tasks can be efficiently “chained” together in the sense that each task produces something that feeds into the next without human intervention. A task might be added to an automated chain even if the human has an absolute advantage in that task in isolation.

## 1 Introduction

One economic approach to understanding a new technology is to frame it as a change in costs. This approach has been applied to illumination (electricity), mechanical power (the steam engine), personal transportation (the automobile), computation (the computer)—all of which made some recognizable economic good much cheaper. But the approach rests on understanding what the new technology does and hence what it is making cheaper. The problem with generative AI—from an economic modeling standpoint at least—is that it is unclear what this kind of AI *cannot* do even now, never mind over the next 5 or 10 years.

One thing that is clear is that it does seem reasonable to analogize generative AI to a kind of labor. It performs identifiable human “tasks” in a way, say, a steam engine does not. However, unlike with computers where we can it replaces rote computation, what generative AI does it not so clearly defined. Stepping

back, the distinctive feature of generative AI is, at least in part, right there in the name: it generates. Writing, code, images, music, videos, and so on—but only in response to a human request, or “prompt.” What the AI generates is not simply a random draw of things the AI could produce. Rather, it produces something in response to some human request or prompt.

What AI generates is not inherently useful, but rather depends on human judgment. The raw output must be evaluated and deemed suitable through judgment. This assessment can then influence where the human asks the AI to modify the output in the required direction. This creates a human-AI hybrid loop of prompting and judging. And upstream of all of this is the necessary human insight into routing, or whether the task is one that an AI is likely to lead to a good outcome, with the prompting and evaluation the worker can provide.

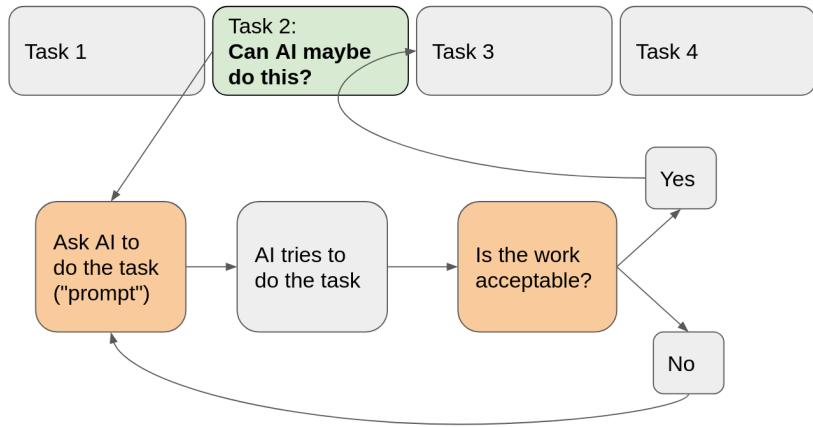
In addition to whatever other tasks for which a comparative advantage exists, no matter how capable these models become, two human tasks remain: 1) asking the AI to do that task, and 2) evaluating the machine’s efforts. Each of these steps is a human-provided task that can, in turn, be AI-augmented.

This paper presents a simple task-based model of generative AI. A productive process is a collection of tasks. However, those tasks are assumed to be done in sequence, with interdependecies. At each step, the task is given to the factor with the *conditional* comparative advantage—though this can depend on what tasks are nearby. But when a task is given to an AI, there is still human involvement in (1) communicating what is desired to the AI and (2) assess the output and potentially reformualting the preprompt as needed. The decision to route to an AI depends not only on the theoretical capabilities of the AI for a task, but how well the AI is likely to do, given the prompting and judgment of the human.

## 2 A model of production

To start, imagine a job that consists of just a single task. A human can do the task at a cost of  $c_h$ . Note that we are ignoring capital costs—all that are present are labor costs. An AI that can *try* do the task instantly at zero cost and if “prompted” by a human. However, it it only succeeds with probability  $q$ , given the optimal effort by the human at “prompting.” This  $q$  captures the model’s capability but also reflects the capability of the human at prompting

Figure 1: A job with a task decomposed into sub-tasks



and judging and the nature of the task.<sup>1</sup> Each prompt costs the human  $c_p$ , and each evaluation of an AI output costs the human  $c_e$ . Let  $c_p + c_e = c_m$ , with  $m$  for “management.” If the machine fails at a task, it is kicked back for the human to try again with another prompt, at another cost of  $c_p$  and then  $c_e$ . Success and failures are independent, and so it always makes sense to keep trying with the AI, conditional upon it making sense to try once.

To indicate a task being done by a human, we will use the notation  $(\cdot)$  and we will use  $\langle \cdot \rangle$  for a task done by a machine. As a memory aid, note the curved parenthesis captures of  $(\cdot)$  the greater “flexibility” of humans. Figure 1 illustrates the situation.

Lemma 1 gives the criterion for automation when considering a task in isolation: is  $c_h > c_m/q$ ?

**Lemma 1.** For a given task, it is cheaper to automate if  $C\{\langle 1 \rangle\} < C\{(1)\}$ , which happens when  $c_m/q < c_h$ .

*Proof.* The expected cost of using the AI for a task is  $c_m/q$ . Recall

$$\sum_{k=1}^{\infty} kq^k(1-q)^{k-1} = 1/q, \quad (1)$$

and so the task will be automated if  $q > \frac{c_m}{c_b}$ .

<sup>1</sup>This constant  $q$  even for a single task is a simplification, as we might expect the “first”  $q$  to be low for a given prompt, but then increase with each subsequent prompt that is a refinement. However, it will still have a finite expectation.

This decision criterion has some intuitive properties. The higher the human performance costs for a task, the more likely a worker is to try an AI, even a fairly incapable one (i.e., low  $q$ ). The lower the prompting and evaluation costs, the more likely an AI will be used. Note that with a high capable AI ( $q = 1$ ) with output that is very cheap to evaluate ( $c_m \approx 0$ ), costs can be very low.

In isolation, raising  $q$  has two effects on the kinds of labor that is used. It displaces direct labor doing tasks, as it is more likely that  $c_m/q < c_h$ . But it increases the demand for management-labor of the extensive margin, as more tasks flip from being done by a human to being done by a machine.

It seems likely, empirically, that prompting costs are sub-linear with respect to  $c_h$  for many tasks. The prompt “write me a haiku about economics” is about the same effort as “write me an epic poem about economics” but the  $c_h$  differs radically. To the extent  $c_h$  and  $c_e$  vary less than  $c_h$ , we should expect a size bias, with more attempts to use for larger tasks in the  $c_h$  sense.

## 2.1 A job with two tasks

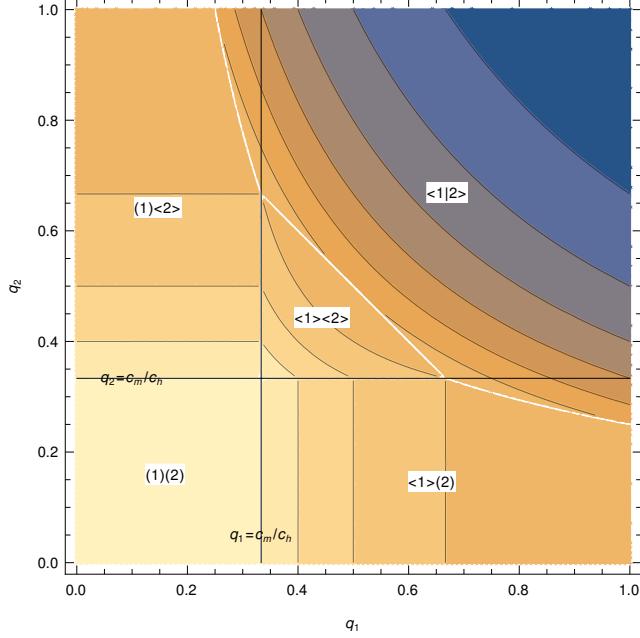
Now imagine two tasks, 1 and 2, with AI success probabilities  $q_1$  and  $q_2$ , respectively. The 4 human/machine possibilities are: both tasks are done by human,  $C\{(1)(2)\} = 2c_h$ ; both are done by machine,  $C\{\langle 1 \rangle \langle 2 \rangle\} = c_m(1/q_1 + 1/q_2)$ ; task 1 by a machine and task 2 by human,  $C\{\langle 1 \rangle (2)\} = c_m/q_1 + c_h$ ; and task 1 by human and task 2 by machine  $C\{(1)\langle 2 \rangle\} = c_h + c_m/q_2$ .

We now introduce another possibility: the “chaining” of AI tasks. Doing this creates a new task with a success probability equal to the product of the success probabilities of the constituent tasks. It is similar to Kremer (1993)’s model of production. The chained-together task still has human prompting and evaluation at the start and end, inheriting the  $c_p$  of the first task and the  $c_e$  of the last task. Notation-wise, a “|” indicates chained-together tasks, e.g.,  $\langle 1|2 \rangle$  indicates a machine does tasks 1 and 2 in sequence. This creates a fifth possibility in the two-task scenario,  $C\{\langle 1|2 \rangle\} = c_m/(q_1 q_2)$ .

For a given task  $k$ , we can think of  $q_k$  as the probability of success when it gets a perfect input. When a human did the preceding task or a human managed the AI process,  $\langle k - 1 \rangle$  or  $(k - 1)$ , the task is done perfectly. In contrast, when the tasks are chained,  $\langle k - 1|k \rangle$ , there is no quality assurance between tasks, and so the probability that the  $k$  task gets a perfect hand-off is only  $q_{k-1}$ .

Figure 2 illustrates the possibilities with the unit square, with  $q_1$  on the

Figure 2: Production costs with a two task job



$x$ -axis and  $q_2$  on the  $y$ -axis. It is a contour plot showing costs of production. For purposes of example, both tasks have same human cost and management costs,  $c_h = 3/2$  and  $c_m = 1/2$ . The unit square is divided into five regions, corresponding to the five possibilities for a two-task job.

In the human-only  $(1)(2)$  region, there is no gradient with either  $q_1$  or  $q_2$ , as the AI capabilities are irrelevant. It is flat plateau of costs  $2c_h$ . In the partial automation regions, there is a gradient in cost with respect to the automated task:  $\frac{dC\{(1)(2)\}}{dq_2} < 0$  and  $\frac{dC\{(1)(2)\}}{dq_1} < 0$ , but zero cross partials. This is also true in the automated-but-not-chained region of  $\langle 1 \rangle \langle 2 \rangle$ .

In automation chains, investments in either task help, but interestingly, they have cross-partials in terms of cost reduction that make them substitutes. We can see that with  $\frac{d}{dq_1 q_2} C\{\langle 1 | 2 \rangle\} = \frac{1}{q_1^2 q_2^2}$ . In other words, an investment in  $q_1$  is, all else equal, worth more when  $q_2$  is low. This feature would tend to create incentives for “balanced” investment in tasks, trying to improve the weakest link in the chain.

Interestingly, with  $\langle 1 | 2 \rangle$ , there are regions where humans have a comparative advantage in a task ( $q < c_m/c_h$ ), but it is still fully-automated because of the advantages of chaining together tasks. The  $(1)\langle 2 \rangle$  to  $\langle 1 | 2 \rangle$  boundary and the

$\langle 1 \rangle \langle 2 \rangle$  boundary are interesting: the curves are defined by  $c_h + c_m/q_2 = c_m/(q_1 q_2)$  and  $c_h + c_m/q_1 = c_m/(q_1 q_2)$ , respectively. At the boundaries, humans actually have a comparative advantage in the task compared to a machine *alone*, but the possibility of chaining “pulls” this task into full automation.

Lemma 2 shows that the automation decision cannot be made in isolation and that it depends on the adjacent tasks.

**Lemma 2.** *A task might be automated even if humans have a comparative advantage over machines in doing that task:  $C\{\langle 1 \rangle \langle 2 \rangle\} < C\{\langle 1 \rangle \langle 2 \rangle\}$  does not imply  $C\{\langle 1 \rangle \langle 2 \rangle\} < C\{\langle 1 | 2 \rangle\}$ .*

*Proof.* We can see this with a counterexample. The machine success probabilities are  $q_1 = 3/5$  and  $q_2 = 4/5$ . Management costs are the same for both tasks,  $c_m = 1$ . Human performance costs for the first task is  $c_h = 3/2$ . Let the human performance cost for the second task be  $\infty$ , so it must be automated. Note that for the first task, it is cheaper to have the human do it:  $c_h < c_m/q_1$ , as the human cost is just  $3/2 = 9/6$  but the AI cost is  $5/3 = 10/6$ . However, we fully automate the costs are  $c_m/(q_1 q_2) = 25/12$ . If we do the first by hand and automate the second, we have total costs of  $c_h + c_m/q_2 = 11/4$ , or  $33/12$ . Thus, even though humans have a comparative advantage in a task, we might chain it with another task and manage the combined task.  $\square$

## 2.2 When to chain to tasks

The decision to chain two automated tasks is simple: chain if  $q_1 + q_2 > 1$ . This is given in Lemma 3.

**Lemma 3.** *If  $q_1 + q_2 > 1$  and  $C\{\langle 1 \rangle\} < C\{\langle 1 \rangle\}$  and  $C\{\langle 2 \rangle\} < C\{\langle 2 \rangle\}$ , then  $C\{\langle 1 | 2 \rangle\} < C\{\langle 1 \rangle \langle 2 \rangle\}$ .*

*Proof.* If we chain the tasks, costs are  $\frac{c_m}{q_1 q_2}$ , whereas if we split them up, we have  $\frac{c_m}{q_1} + \frac{c_m}{q_2}$ . We chunk to tasks if

$$\frac{1}{q_1 q_2} < \frac{1}{q_1} + \frac{1}{q_2} \quad (2)$$

which is true when  $q_2 + q_1 > 1$ , otherwise they should be split.  $\square$

A related implication of Lemma 3 is that the success probabilities of each task must sum to greater than 1 as a necessary—but not sufficient—condition for a chain.

**Lemma 4.** *A cost-minimizing sequence,  $T^*$ , cannot have a chained machine sequence with sum of  $q$  values less than 1.*

*Proof.* Suppose we have a chain in  $T'$  with  $Q = q_1 q_2 \dots q_k$  and that  $\sum_k q_k < 1$  and that it is cost-minimizing i.e.,  $T' = T^*$ . Take the smallest  $q$  internal to the chain and call it  $q_\epsilon$ . We can treat the situation as a three-chain problem,  $q_l$ ,  $q_\epsilon$  and  $q_r$ , with  $l$  and  $r$  for left and right. The fully automated chain has greater costs than the “separated” chain  $C\{\langle l|\epsilon|r\rangle\} > C\{\langle l\rangle\langle\epsilon\rangle\langle r\rangle\}$  if

$$\begin{aligned}\frac{1}{q_l q_\epsilon q_r} &> \frac{1}{q_l} + \frac{1}{q_\epsilon} + \frac{1}{q_r} \\ \frac{1}{q_l q_\epsilon q_r} &> \frac{q_r q_\epsilon}{q_l} + \frac{q_r q_l}{q_\epsilon} + \frac{q_l q_\epsilon}{q_r} \\ 1 &> q_r q_\epsilon + q_r q_l + q_l q_\epsilon,\end{aligned}$$

and because  $q_l + q_\epsilon + q_r < 1$ , and all  $q$  values are less than or equal to 1,  $1 > q_r q_\epsilon + q_r q_l + q_l q_\epsilon$ . Therore,  $T'$  is not the cost-minimizing sequence.  $\square$

Lemma 4 will prove useful later for computing cost-minimizing allocations, as it allows us to further prune the tree.

**Lemma 5.** *If a task can be combined with either a more or less reliable chain, if it should be combined, it should be combined with the more reliable chain. If  $C\{\langle 1\rangle\} > C\{\langle 3\rangle\}$ , then  $C\{\langle 1\rangle\langle 2|3\rangle\} < C\{\langle 1|2\rangle\langle 3\rangle\}$ .*

*Proof.* Suppose you have  $q_L$ ,  $q$ , and  $q_H$ , in sequence, with  $q_H > q_L$ . Assuming you will chunk left or right, which chunk to you choose?

$$\frac{1}{q_L} + \frac{1}{q q_H} < \frac{1}{q_L q} + \frac{1}{q_H} \tag{3}$$

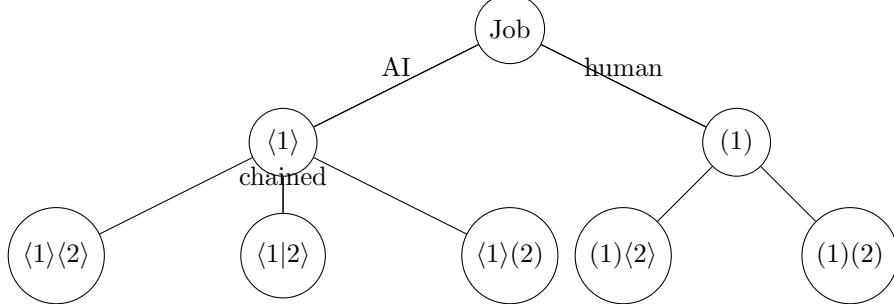
$$q q_H + q_L < q_H + q q_L \tag{4}$$

$$q(q_H - q_L) < q_H - q_L. \tag{5}$$

$\square$

Humans can “hand-over” tasks to an AI—but can the AI hand over tasks to the human? In the model, when a chain goes from  $(0)\langle 1|2\rangle\langle 3\rangle$ , we can think of the human as handing off tasks 1 and 2 to the machine, but we can also think of the machine as handing off task 3 to a human.

Figure 3: The possible production scenarios with an  $n = 2$  task job



### 2.3 Multiple tasks

Now we will consider a job that is a sequence of tasks to be performed  $1, 2, \dots, n$ . Here, we will focus on the computational aspects of the problem. Let  $C_h$  be a vector of costs to a human for performing each task. Note that with an  $n$  task jobs, there are  $2^{n-1}$  possible task partitions. And each singleton can be done by a human or a machine; each sequence is done only by a machine.

The cost minimization problem is equivalent to finding the shortest path in a tree. To grow the tree, start with the first task. It can be automated or done by a human. It then has up to two children—the next task is either machine or human, and if machine, it could also be part of a chain. Figure 3 illustrates the tree.

The tree grows to, worse-case,  $2^n$  children where  $n$  is the number of tasks. It would be less than this because human-done nodes only have one child: either machine or human. Growing the tree has an exponential computation complexity—it is unclear if there is a polynomial time algorithm. It might also be possible to prune certain leafs using the logic of Lemma 4.

## 3 Product market sketch

Imagine there is a measure 1 of worker/consumers. Workers supply labor inelastically. There are  $M$  *potential* products in the market which have idiosyncratic chains of tasks and associated costs. Each product has the same demand curve,

$$d(p) = \max\{0, \frac{\bar{p} - p}{\bar{p}}\}. \quad (6)$$

where  $\bar{p}$  is exogenous and  $p$  is the product market price of the good. Note that when  $p \geq \bar{p}$ , no one buys the good; when  $p = 0$ , everyone buys the good.

Let  $l_i$  be the labor required to create one unit of good  $i$ , under the optimal production chain, given  $w$ . There are constant returns to scale in production i.e., to produce  $N$  units of the good, it requires  $Nl_i$  units of labor. A unit of the good has a cost of  $p_i = l_i w$  where  $w$  is the market wage, and so  $d(l_i w)$  is how much of good  $i$  gets produced.

Market clearing is defined by a wage,  $w$  and a set of products,  $M' \subseteq M$ , that are produced in equilibrium. The set  $M'$  is defined as those with labor requirements such that  $\bar{p} < l_i w$ . Total expenditure equals total labor income at market clearing, or

$$w = \sum_{i \in M'} p_i d(l_i w). \quad (7)$$

If we let  $\bar{l}$  be the average amount of labor used per good and  $|M'|$  is the number of good produced in equilibrium, we have

$$w = \bar{p} \left( 1 - \frac{1}{|M'| \bar{l}} \right). \quad (8)$$

(Not formal - economic intuition - not proven!) What happens when  $q$  increases i.e., AI models become more capable? This causes all  $l_i$  to fall or at least stay the same, as goods need less labor to produce. This tends to lower wages (NB: Welfare increases, as total consumption increases. In real terms, I don't think wages fall(?)) This raising of  $q$  also causes  $|M'|$  to increase, as more goods now become economical to produce. There are two reasons: technical production costs fall for these marginal costs, pushing them over the limit, plus labor is freed up from all the products previously produced. These marginal goods are also particularly labor intensive goods, having  $l_i \approx \bar{p}/w$ —the highest possible labor usage. This will tend to dampen the shock. NB: This re-creates the Acemoglu “labor has an advantage in new tasks” (here, products) type result but for economic reasons rather than by assumption.

Other thought—if automation chains cause more goes to go to  $l_i \approx 0$ , further advances in  $q$  become irrelevant, meaning a smaller pool of workers are “freed up” by advances in  $q$ .

### 3.1 Scratch

Suppose there are  $N_K$  tasks an economy which can be configured into  $N_S$  slots. This economy has  $\binom{N_K}{N_S}$  possible products. Let  $d_i(p)$  be the quantity demanded for good  $i$  when the price for that good is  $p$ . This has an associated wage bill of  $d_i(p)T_i^*$ . Of those products actually pursued in equilibrium, the price equals the cost:  $p_i = C\{T_i^*\}$ . Goods that are not produced at all are those where  $d_i^{-1}(1) < C\{T^*\}$ . We could imagine many products are not produced because no one wants them at all, but others are just not price competitive.

Products that are produced have constant returns to scale. All workers can provide the management input at  $c_m$ , which is the same for all tasks. Before automation, the cost of each job is simply the sum of the costs of each task.

Workers optimally choose what subsets of the  $K$  skills to acquire. Although they could acquire  $2^K$  possible bundles of skills, they only acquire those correspond to products actually produced in equilibrium.

The cost to produce the item is  $c_i = C\{T_i^*\}$ , which is also the wage bill, as the only cost in the model is labor. An increase in productivity raises output, which lowers prices. With lower prices, more quantity is produced. An improvement in AI pushes out the frontier of products that are produced, as some are now cost effective. Consumption is biased towards goods where AI can substitute for labor—particularly those with lots of chaining possibilities.

To make this more concrete, assume a single task,  $k \in K$ , can be automated, with say  $q$  increasing dramatically. Let  $I$  be the set of all jobs that required skill  $k$ . That set of jobs  $I$  now demand less labor, for the same amount of output. However, there is a productivity benefit from increased output that increases labor market demand. Which one “wins” depends on the elasticity of the demand curve at the equilibrium price. Demand for other skills that appear in  $I$  jobs also decreases due to automation chains.

Consider that with  $k$  automated, there are other workers who could now potentially compete for jobs in  $I$ . Let  $I_H$  be the set of jobs where  $k$  was the only skill workers there were “missing.” These are Rick Rubins. This is a positive labor supply shock to the  $I$ -tasks market. But it is a different supply shock in the  $I_H$  markets—those that were paying higher wages will likely see an increase and wages fall, those that were paying lower wages will likely see wages rise—it depends on where they were prior to the collapse. This is just Baumol’s cost disease due to rising productivity in the  $I$  sector. By raising  $c_h$  in  $I_H$  sector, that increases the incentives to invest in automation.

The automation of  $k$  also has impacts on the space of products produced in equilibrium. Tasks with high  $k$  costs might now become feasible, with tends to increase demand for labor with skills that complement  $k$ —which are precisely the skills those workers specializing in  $k$  also had.

**Proposition 6.** *Lowering costs of any kind does not reduce the number of viable products.*

**Definition 7.** *An improvement in AI is when there exists a  $j$  such that  $Q'_i > Q_i$  and for all other indices  $j$ ,  $Q'_j \geq Q_j$*

**Proposition 8.** *An improvement in AI capabilities does not increase costs:  $C\{T_{Q'}^*\} \leq C\{T_Q^*\}$ .*

**Lemma 9.** *Total costs stay the same or decline in  $c_h$  and  $c_m$ . Both are likely sub-linear because of substitution possibilities.*

**Lemma 10.** *Management labor and doer labor are gross substitutes.*

**Lemma 11.** *Lowering the cost of AI has mixed effects on the demand for management labor; unambiguous negative effects on doer labor.*

*Proof.* A decrease in  $q$  raises the number of tasks automated, which increases demand for management labor. But it also reduces management costs,  $\frac{d}{dq} \frac{c_m}{q_m} = -\frac{c_m}{q_m^2}$ .  $\square$

*Proof.* If  $c_m$  decreases, demand for doer labor decreases, lowering  $c_h$  in equilibrium.  $\square$

**Lemma 12.** *Increases in capabilities of AI (lower  $q$ ) (a) deceases costs of existing goods, (b) expand the goods that are economical to produce in equilibrium, (c) give workers more income to purchase goods.*

## 4 Model simulations

I ran a simulation of many “jobs” with chained tasks and found the lowest cost combination under different assumptions about human and machine labor. Human costs were fixed  $c_h = 1$ . I draw the  $q$  from a uniform distribution between  $U[q_{min}, 1]$ , varying  $q_{min}$ . The number of tasks in the chain was  $n$ .  $c_m$  was changed in each simulation, ranging from 0.1 to 2, in 0.1 increments. The simulation parameters are:

```

num_sim = 100
nrange = range(2, 11)
cmin_range = np.arange(0.1, 2, 0.1)
qmin_range = np.arange(0.1, 1, 0.1)
cm_range = np.arange(0.1, 2, 0.1)
n_range = range(2, 11)

```

You can see how in the simulation, average task costs fall with longer chains. This is presumably because longer chains can be fully automated.

When AI management costs are higher, more tasks are done by humans.  
When AI quality is higher, more tasks are done by machines.

Table 1: Effect of costs and quality on outputs

	<i>Dependent variable:</i>		
	Avg. cost/task	Frac tasks by human	
	(1)	(2)	(3)
Num. tasks	-0.017*** (0.001)		
Management Cost		0.512*** (0.002)	
AI model floor (qmin)			-0.507*** (0.007)
Constant	0.916*** (0.007)	-0.162*** (0.003)	0.603*** (0.004)
Observations	29,241	29,241	29,241
R <sup>2</sup>	0.009	0.641	0.140

Notes: Some notes.

Figure 4 shows that average prices increase in management costs (unsurprisingly) but that the relationship appears to be either concave or convex depending on the baseline level of model quality.

## 5 Discussion

If they choose to use an AI, the human asks the AI to perform the task with a prompt. The AI tries to perform the task, generating output. The worker then evaluates the output to decide if it is suitable. If it is, the worker can continue to the next task. If it is not, the human can try to modify the prompt and re-evaluate the new output until the need is satisfied.

One might think that evaluation itself could be the domain of the AI, but if the model could have improved the chance the output would be evaluated

Figure 4: Average costs versus management costs, by model quality

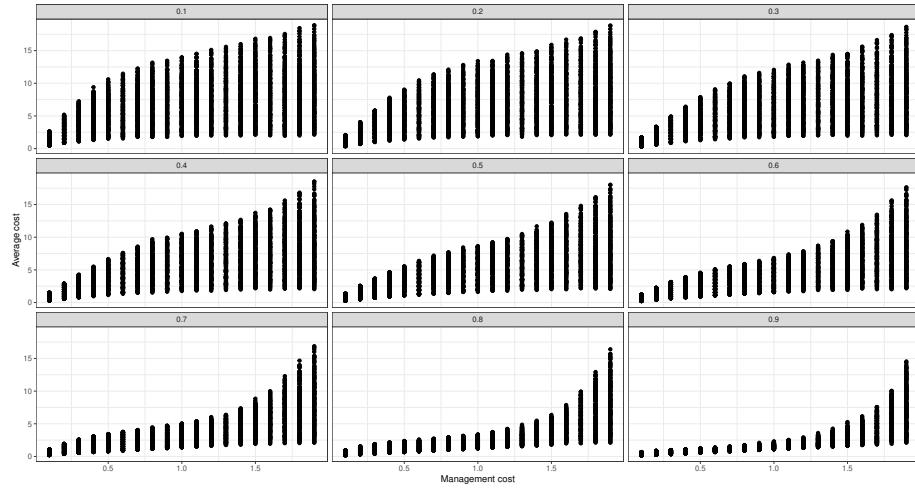
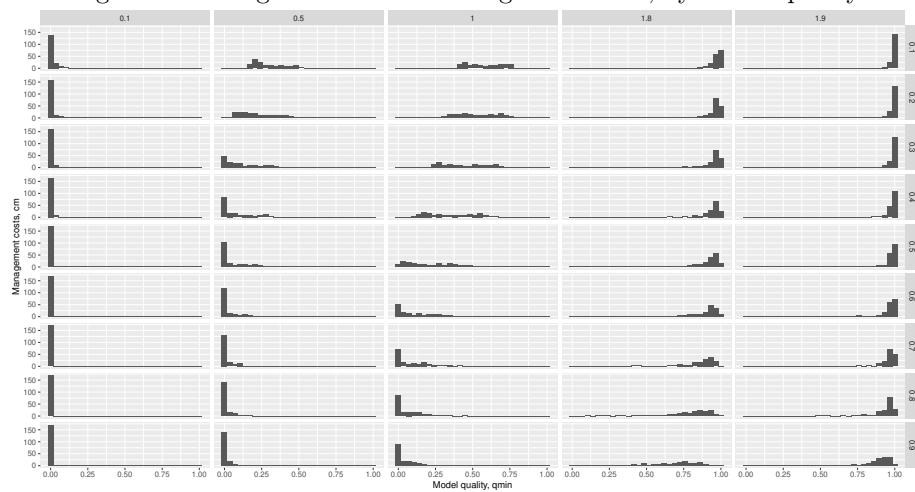


Figure 5: Average costs versus management costs, by model quality



positively, that process would have already been incorporated into the model. What is left is the residual evaluation that a human must uniquely supply: judgment.

There is no imposed structure on the kinds of tasks that humans and machines have advantages in. If we want to assume a machine cannot do a task, we can assume it has a zero probability chance of completing the task; if we want to assume that a human cannot do a task, we can assume infinite cost.

Unlike the task-in-isolation view, it is possible to reap further benefits from automation even from tasks where humans have an absolute advantage. Namely, tasks can be “chained” together in automation to create entire systems. Note that with this chaining tasks can become increasingly large and abstract as models increase, generating something that looks like an entire system and not just a task. This chaining builds into the model a consideration pointed out by Bresnahan (2020), who argues there has been little to no task-level substitution with the AI systems we actually see in production: “the transition to ICT-based production has largely proceeded at the system level, not the task level.” It has all been capital deepening. In the language of our model, system-level substitution would just be characterized as long chains of task automation. As this occurs, what constitutes a “task” moves to higher and higher levels of abstraction: From “find typos in this ad copy” to “create ad copy” to “create a marketing campaign, launch, execute and monitor performance” to “increase sales” to “run the business profitably” to .... In the model, it is the possibility of chaining that can create large productivity gains.

The model offers several insights. First, it predicts what kinds of tasks tend to be amenable to automation in terms of economic fundamentals. Tasks that are expensive for humans to do are natural candidates for automation. But a less obvious insight is that the extent of generative AI usage depends on the costs of judging the model’s output. This cost could vary considerably based on the nature of the good. An analogy is drawn to the process by which consumers evaluate goods. Search goods are trivially easy to evaluate, whereas inspection goods and credence goods might have prohibitively expensive evaluation costs. These differences will affect the adoption of AI by task.

Endogenizing whether an AI can “do” a task as depending on the stakes of the productive context likely allows for insights into the pace and location of adoption. For example, consider the task “answer questions about the text in a document” and whether an AI can do this task depends on the payoff to doing it correctly. The firm might be fine letting a machine do low-level customer

service support; we might be very hesitant to let it handle queries from major institutional investors, in part because what constitutes “good enough” might differ. But also, the next task might be follow-questions about the state of the business that only the CFO can credibly answer.

In the model, it matters what tasks are “next” to each other in some productive process, because tasks that have a high ability to be automated can be chained together.

But beyond tasks considered in isolation, the model predicts, qualitatively tasks are likely to be targets for automation—namely those that allow not just one-off task replacement but system-level replacement.

As such, we can get “automation dispersion” where tasks are automated in some production settings but not others.

Prompting is really the human skill of getting what you want out of some AI system. There is no inherent reason to think that this will itself be a low-skill job—just a different skill job. Most of management is deciding what you want done, asking right people to do it, in a way that leads to an effective outcome, and then monitoring the process to see if it achieves the desired goals. No one thinks of this as a “low skill” proposition. The literature on the cost of evaluation of goods—search goods, inspection goods, experience goods, credence goods—gives us an economic way to think about what tasks are likely to be automated.

(1) The model primitives map to things that can be measured. The time it takes to do a task; the task composition of a job; the capabilities of an AI in some area. (2)

Even if only some tasks are automated, it still might facilitate labor-labor substitution. This might change who eligible to fulfill some role in the productive process. To the extent there is some difference in the requirements to do some task, ask for some task to be done, and evaluating the output of that task, there is a possibility to use different kinds of workers in a productive task.

(4) It also highlights that we are likely to see investment not only in direct model capabilities, but the ways in which they can be prompted and the results evaluated.

Although there are advantages to thinking in terms of tasks, most labor market participants think in terms of “occupations.” Our training and human capital acquisition is structured as occupations. People want to know which whole occupations will disappear, and what occupations have good prospects.

The approach gives a way to think about an occupation “disappearing.” In

the model, an occupation only disappears when all of its tasks are subsumed in an automation chain. Take an job that truly disappeared: elevator operator. The tasks were 1) “Announce the direction of the elevator,” 2) “ask people what floors they are going to” 3) “bring the elevator to the appropriate floors”, 4) “open the doors” as well as some other non-sequential tasks like inspect for defects or machine failures. This got turned into an automation chain where the customer was given the “prompting” task of selecting a floor; the evaluation task is just seeing that the elevator was the appropriate floor—an inspection good. If it was hard to know if you had gotten the right floor, we might expect that task to still be done by humans. Jobs with few tasks are more prone to full automation. All else equal, a job with a single task is more prone to automation compared to one with multiple distinct tasks. Jobs where the next task is unpredictable are more resistant to full automation. The model highlights the benefits to combining multiple tasks into an automation “chain.” But if task 1 and task 2 rarely occur together, chaining them together is not likely to work well if that requires some set up.

Examples include Autor et al. (2003)’s “computation substitutes for routine cognitive tasks and complements non-routine cognitive tasks” is a useful simplification of what computers do. Or technology automates some tasks previously done by humans but also spurs new tasks which only humans can do, at least for some time (Acemoglu and Restrepo, 2019). Or AI is a prediction technology that lowers prediction costs, increasing the value added by labor that complements prediction (Agrawal et al., 2019).

## 6 The economic approach to modeling technology

There are contrasting ways consider the effect of technology on labor markets. One approach to understanding the impact of technology on labor markets has been to focus at the “task” level. Goods are produced by the completion of tasks and those tasks can be assigned to the factor of production with the comparative advantage in that task. This approach has been applied not just to technology but also to trade and offshoring. But specifically with respect to AI and automation, the task-based view is that labor, by being more flexible (Acemoglu and Autor, 2011), has an absolute advantage in “new” tasks, but that as time passes, they become susceptible to automation (Acemoglu and

Restrepo, 2018).

There are criticisms of the task view of technology. A contrasting view is offered by Bresnahan (2020), who argues there has been no task-level substitution with the AI systems we actually see in production: “the transition to ICT-based production has largely proceeded at the system level, not the task level.” It has all been capital deepening. To the extent there is important labor-capital substitution, it will happen at the system level. There is a long tradition in studies of ICT to take a “systems” view—that ICT requires a reconfiguration of production to reap the benefits (Bresnahan et al., 2002).

An argument for a task-based view with respect to generative AI is that it does due a number of distinct *tasks*. The things it can do well: analyzing data, writing reports, drafting communications, doing research, coming up with plans, etc., certainly seem like the kinds of tasks common across large numbers of workers. This stands in contrast to, say, ICT like databases, servers, data centers, recommender systems, etc. are better characterized as technologies that can be used to build systems that might do some tasks or replace whole collections of tasks. The product market success and adoption of ChatGPT is a testament to this: there are 100M users of OpenAI and presumably, many are using it for their work. Even the names of these technologies indicate their role of helping individuals: the names people use like “co-pilot” and “assistant.”

The literature to date on AI specifically (as opposed to automation) in the task-based tradition has emphasized AI as a prediction technology (Agrawal et al., 2019). While forecasting and prediction are certainly key firm tasks, relatively few workers have that as a task. By comparison, the outputs of generative AI—things like writing, research, or analysis (which could include prediction) are commonplace. As a case in point, on ONET, only 22 occupations are returned for the search term “predicting.” By comparison, the occupations returned for communicating, writing and analysis are 426, 332, and 408, respectively.

Although this is an area where machines can excel—and fit Bresnahan (2020)’s examples perfectly—this seems, with hindsight, to be too narrow of a framing. Generative AI models have shown the ability to create output similar to or better than even expert humans can produce over a broad class of knowledge-worker tasks. Consistent with this view, experimental evidence at knowledge worker tasks shows large productivity gains using ChatGPT for a collection of knowledge worker tasks (Noy and Zhang, 2023). The fraction of tasks currently performed by workers in the US that might be done or augmented

with LLMs is considerable (Eloundou et al., 2023).

They can do tasks that we would have thought impossible for computers to do even a short time ago. Given these capabilities, it is widely anticipated that they *will* have labor market effects, but what those effects are remains to be seen. We would like to have the generative AI version of Autor et al. (2003), but our problem is the same as if Autor et al. (2003) tried to write their paper in 1960. With the caveat that predicting the future is a mug’s game, can we examine what these technologies do and try to sketch out the likely labor market implications now?

A natural starting point for the effects of generative AI on labor markets is to try to classify tasks as being the kinds of the things generative AI can “do” or “not do.” But this misses two considerations. First, tasks are not done in isolation and most productive tasks are interdependent. Second, whether a model’s output at some task is sufficient depends largely on what that output will be used for. For example, for the task “summarizing a document”: whether the AI can do this task, in an economic sense, depends if the document is being summarized is the Supreme Court or for Steve Caruthers.<sup>2</sup>

Economic forces tend to create dependencies between tasks both within in a job and within a firm. Since economists first recognized the division of labor as an essential consideration, the interdependence between tasks has been noted. The manufacture of pins proceeds linearly, with each step serving as an input to the next step in the productive process. First, the wire is washed and spooled, then pulled straight, then cut into lengths, then ground to a point, and so on. A failure at any step jeopardizes the final product. This is at the heart of the Kremer (1993) o-ring model.

Tasks also have a recursive structure. Every task can be divided into smaller sub-tasks. For a human, we think of anyone as being able to “do a task” as also being able to do every constituent sub-task in combination with the appropriate capital. A machine might only be able to do some sub-set of those tasks. If tasks did not have a recursive structure, the division of labor would reach some atomic limit.

Smith identified the limit to division of labor as not about hitting some atomic limit, but rather it was about the extent of the market. If you produce so much, you can employ one person or one machine doing the tiniest of tasks. With a smaller market, less so. But there is another cost, at least for labor:

---

<sup>2</sup>A guy I made up who is just browsing Wikipedia.

switching costs and communication costs. When I hand off my output in some chain if I need to explain what should happen next.

The logic of comparative advantage and communication costs tend to create serial dependencies between tasks within jobs, creating an o-ring-like production structure. In the model, tasks within a job have serial dependencies. There are economic forces that push this characterization to be true. If a job lacked serial dependencies between tasks, we could likely reap more gains for specialization. It costs us too much to hand off a task. We only do it when we cannot do the task or the labor savings are so great it is worth it. The labor savings are great when management costs are low: the task is easy to describe, easy to evaluate, and has a high probability of success. For Coasian reasons, we also expect inter-dependencies between jobs within the firm.

A natural question is the delineation of these tasks—where does one begin and the other end? Tasks can be subdivided, with every step capable of being broken into smaller steps.<sup>3</sup> For our purposes, a helpful delineation is some sequence of tasks such that an AI could, in principle, do that whole sequence of constituent sub-tasks. The capabilities of the AI model make the tasks. The start of the task is the first point where some human direction about what to do is required.

Digitization already pushed the marginal reproduction costs and distribution costs for information goods to zero—but the costs of initially producing those goods remained high: I could costlessly share an ebook with you, but writing a book was still a long, hard slog.

Generative AI is an information goods factory, but it has human workers. Generative AI is that it works with us to produce information goods. And the economic-so-what is that the machines can do their part of the productive process at zero marginal cost—or at least at a cost that will likely trend to zero.

A prediction is an information good. But it is also an experience good that might reveal its quality over a long time frame, depending on the nature of the prediction. Once you know a prediction is good; it is probably useless other than as a data point that you should value the predictions of the machine more. If it is a prediction about what might happen, but that could be changed given the firm's actions, it could be a credence good. This is precisely the kind of task that the relative benefit of using an AI might be quite low if the ability to

---

<sup>3</sup>Writing a memo has the subtask “write the first line of the memo” then “write the second line of the memo.” “Write the first line of the memo” and can be broken into “first word” and “first letter” and “keystroke down, then keystroke up.”

judge is low.

### **6.1 Using an AI for some task has qualitative similarities to delegating a task to another human or team.**

The decision to use an AI bears similarities to the decision to delegate some task to a subordinate. And conditional upon delegation, the skill of using AI bears similarities to the management of people and teams. Deciding whether the output is sufficient requires judgment, whether the output is human-produced or otherwise.

**Entrepreneurial innovation will focusing reducing AI management tasks as well as increasing model abilities** We will see investment to improve the capabilities of models but also try to reduce the costs of prompting and to reduce the cost of the evaluation. The skills that AI complements are themselves potential targets for AI assistance. We will likely see AIs that help write prompts or make it easier for humans to express themselves. We will see AIs and related technologies that help evaluate output.

### **6.2 AI will create new labor-labor substitution possibilities for “Rick Rubin” reasons.**

In light of the change in what is delegated, I discuss how labor-labor substitution might change who performs a task. In particular, job redesign will take advantage of the fact that skills in judging output are separate from the skill of creating some output. There are food critics who cannot cook; record producers who cannot play instruments; technologists who cannot program, and so on. As AI makes it so that humans no longer have to do every task in a job, AI might open up jobs to people who are otherwise blocked by a skill they lack.

There are likely to be numerous tasks when prompting and judgment can lead to good outcomes even when the human offering those judgments cannot do it themselves. One reason is that the pool of people who can offer judgment is larger than the pool that can do the task. This might be particularly true for tasks that require a great deal of skill. As a recent illustrative example, the famed record product Rick Rubin described his value in a recent interview with Anderson Cooper:

RICK RUBIN: I've no technical ability. And I know nothing about music

ANDERSON COOPER: Well, you must know something.

RICK RUBIN: I know what I like and what I don't like. I'm decisive about what I like and what I don't like.

ANDERSON COOPER: So what are you being paid for?

RICK RUBIN: The confidence that I have in my taste, and my ability to express what I feel, has proven helpful for artists.

However, that task has to be marginal (assuming other blocking tasks are not also automated). For example, identifying a bone fracture in an X-ray does not mean a person is ready to be a radiologist. I.e., if there was only one task that precluded them from doing some occupation, then AI might allow them to do that occupation. Allowing more people to do more tasks could have an indirect productivity effect. Ironically, the harder the task to perform, the more potential for AI to unlock this substitution potential, as the pool of potential prompters and judges is much larger than the pool of doers.

### **6.3 Generative AI might allow for a kind of pseudo-human capital acquisition.**

Substitution to another human might require a hand-off if all the rest of the tasks still need to be done by the “original” human. But AI augmentation might allow for a task re-design. Consider that for some occupations, there are necessary tasks such that  $c_h$  is essentially infinite, or so high that it would be uneconomic for that person to pursue that job because their effective efficiency wage would be lower than their next best alternative. Take, for example, a master carpenter versus a handyman. There are many tasks in common between the two: driving to the job site, carrying tools, buying supplies, swinging a hammer, etc. But what is distinguishing is that there are tasks required of the master carpenter that the handyman is incapable of performing. This drives down their wage below the handyman wage, and so they are handymen. If generative AI is inequality-reducing, I think this will be the mechanism: taking people that currently lack certain skills and letting them “jump” up.

This framing gives an interesting characterization of the human capital acquisition process generally: learn skills that will lower your performance costs across tasks *if* you can make the job requiring those tasks better than your best job option. The skills you acquire need to make that job marginal. For example, the task “read an MRI scan to detect a bone fracture” is highly remunerative

but only if learned in with all the other tasks in the “Radiologist” task sequence. Licensing aside—this skill alone is essentially worthless. With AI augmentation, the marginal blocking skill might no longer be blocking.

#### 6.4 Generative AI substitutes “maker labor” and complements “manager labor” in the short-run.

In the model, there are just two broad “kinds” of labor: Labor that does tasks, and labor that manages an AI doing that task. While we might think there is some correlation between the cost of the machine and the human doing the task, this seems like an imprudent assumption.

We should not presume that tasks that are hard for humans are easy for computers or vice versa. We do hard, time-consuming tasks because the output is valued. Playing excellent chess *seemed* like the kind of thing computers can do well. Generating photorealistic art, playing diplomacy, and generating new programs on the fly did not seem like tasks where computers would show capabilities.

All else equal, more capable management increases the number of tasks an AI performs, holding AI capabilities fixed. Being adjacent to a highly automatable task will tend to “pull” that task into an automated chain.

## 7 Discussion

### 7.1 So-so technologies

Marginal tasks are those where  $C\{\langle k \rangle\} \approx C\{(k)\}$ . If in contrast,  $q = 0$  and goes to 1, then  $c_h - c_m$  could be quite large. But the more exciting possibility is when we can chain. Suppose we have  $q_1 = 0$  and  $q_3 = 0$  and  $q_2 = 1$  and  $q_4 = 1$ . We have  $\langle 1 \rangle \langle 2 \rangle \langle 3 \rangle \langle 4 \rangle$ , as a cost of  $2c_h + 2c_m$ . If we improve  $q_1$  and  $q_3$  to 1, we can do  $\langle 1 \rangle \langle 2 \rangle \langle 3 \rangle \langle 4 \rangle$ , we can get cost savings of  $2(c_h - c_m)$ , leaving us with costs of  $c_m$ . But if we consider the chaining possibility, letting us go to  $\langle 1|2|3|4 \rangle$ , we can go from  $2c_h + 2c_m$  to just  $c_m$ .

### 7.2 What does “full automation” mean?

Full automation consists of finding places in the production process where the costs of replacing the asking and evaluation steps with an AI are economical because the quality is sufficiently good and the cost of an evaluation failure is

relatively low. For example, consider the task “respond to customer inquiry” previously done with a human representative. That human might take an inquiry and respond, perhaps with templates matching certain common questions and links to so-called knowledge-base articles. But even with a fairly rudimentary chatbot limited to a response of the form “given your question, here are some resources you might find useful” to replace this aspect of the customer service task. The “ask the AI to do something” has been offloaded to the customer, as has the evaluation step is left to the consumer. This full automation is just externalizing the prompting and evaluation costs to consumers, which presumably has some negative effect on demand.

If the AI is doing the prompting and evaluation, we can think of  $c_e$  and  $c_p$  as de facto zero. The failure mode is then the one step in the chain passes a bad prompt to the next. If the prompt and evaluation step has a very high  $q$ , this concern is not so great. Why do we not have an evaluation step? The prompt for the next task in the chain is the output from the previous step: there is no new effort, and this is already optimized to do that task as well as it can. There is also no distinct evaluation step. Why? Because if the AI could improve  $q$  by evaluation on its own, that would already be included in the process it uses. The overall chain of unreliable components gives a  $q = q_1 \cdot q_2 \cdot q_3$ . We can then consider just the human sandwich of prompting and evaluation, with this sequence of tasks in the middle.

With a more advanced AI, it might try to answer a question directly. This is common now. But it is unlikely that the task “respond to current institutional investor inquiry” would be turned over to even a sophisticated AI given the risks.

### 7.3 AI usage as a managerial skill

The routing, prompting, and evaluation tasks are similar to a manager’s tasks. Generative AI seems to complement a new kind of managerial skill, albeit the management of an AI. But given humans are harder to change the machines, it seems likely that AIs will continue to be modified to make it so that humans can manage them.

Knowing what to yourself versus delegating, giving good instructions conditional upon delegating, and evaluating output—and course-correcting as necessary—is closely similar to simply managing people. Being a judge of talent, scoping and explaining projects in ways people will understand, giving good, actionable

feedback, and so on. It is not fundamentally different from what should be delegated to a human.

Adding “let’s think step by step” to a prompt can dramatically increase performance (Kojima et al., 2023).

#### 7.4 Returns to more capable models

Investments that raise task success probability,  $q$ , have two effects. They lower costs inframarginal for all tasks that the AI was already used for, with the elasticity of total costs equal to negative 1 in  $q$ , i.e., a 10% increase in  $q$  lowers expected AI costs by 10%. The other effect is on the extensive margin, making some tasks more likely to be AI-able. The effect depends on the “density” of tasks where  $q \approx \frac{c_p + c_e}{c_h}$ , which presumably depends on the job.

#### 7.5 Returns to complementing the complementary skills

The new tasks of routing, prompting, and judging are human tasks potentially complemented by AI. One interesting feature of LLMs is that with other investments, there are “hacks” that can lead to better performance. For example, asking the model to think “step by step” appears highly effective. And models that become knowledgeable about their own performance might be able to help design better prompts. For example, instead of a prompt “What are good marketing slogans for an ice cream store?” the prompt can be “what is a good prompt to an LLM that would generate good marketing slogans for an ice cream store.”

Practices could also help with evaluation. For example, a request for a marketing slogan could also generate a collection of AI agents that will be asked to pretend to be customers. They could then report their preferences. This would change the evaluation task to be more empirical.

The other way that technological innovation can increase productivity is not by increasing model performance directly—working on  $q$ —but by lowering  $c_p$  and  $c_e$ . These are innovations to make it easier to express what you want from machines and easier to evaluate outcomes seem likely.

One interesting feature of the rise of ChatGPT is not that GPT3.5 was so much more capable than GPT3—it was that it had a user interface innovation that allows regular people to interact with an AI in a way that is similar to how they interact with people. In a nutshell, OpenAI lowered  $c_p$ .

Becoming better at prompting will require the user to understand how to change the input to affect the model’s output in the desired direction. We might think of the user as not so much choosing  $x$  but as choosing an  $x$  vector, with a greater understanding of the mapping between choices of  $x$  and the outcomes.

## 7.6 Evaluation costs

Economics classifies goods by the actions consumers must take to assess them. Although it is a matter of degree, we can put goods roughly in order of the cost of evaluation

A company might be willing to try some LLM-generated ad copy in sponsored search ads with little to no human oversight; We might put more effort into evaluating a model’s plan surgery or the draft of a judicial opinion.

**Search goods** For some tasks, the human is the final arbiter and can immediately decide if the outcome is sufficient. There is no further source that needs to be considered. For example, if a model is asked to “come up with an agenda for a meeting to discuss recent sales declines” the human can simply look at the agenda and decide if it is sufficient.

**Inspection goods** There are other information goods that require a more costly inspection. As a case in point, consider computer code, which LLMs are good at producing. The user can then run that code to see if it runs correctly, with the caveat that it would be difficult to consider all possible inputs. This limitation aside, a popular paradigm in computer programming is “test-driven development,” where you create the tests first and then write the code, assessing automatically where it means your needs.

In solving certain kinds of math problems, the solution method is guess-and-verify. Or even if there is a routine procedure you follow, such as long division or solving an algebraic equation, we often switch mental modes to see if an answer is sensible. As an aside, many of the facile criticisms of Chat GPT take this form—show it is bad at giving you facts. AIs that can cite sources change inspection goods to search goods.

The AI might lie to you, and, what you probably need to do is go verify the answer using normal research methods.

**Information experience and credence goods** Some information goods are consumed over time, and learning about the quality might take time. A person might not know if they enjoyed a movie until they have actually watched it. For information experience goods, we tend to rely heavily on reviews. But this only works when many people potentially evaluate the same good. With generative AI, the information good is likely generated uniquely for that particular worker. As non-lawyer could not verify whether a commercial contract is likely to hold up to scrutiny by the courts. A course of advised medical treatment might be very difficult to judge by a non-expert.

**Rick Rubin Substitution: AI-enabled labor-labor substitution** A low  $c_h$  might imply a relatively  $c_p + c_e$ .

Knowing how to do something does not mean we have an advantage in asking people to do that thing, though we probably have an advantage in  $c_e$ . But low  $c_p + c_e$  does not imply a low  $c_h$ .

In the framing of the AI problem, we have assumed that the human who previously did the task was the one deciding whether to delegate that task to an AI. This could be sensible, but this is not necessary, and in fact, there could be advantages to using a different human for this task. Consider that in the analogy to management, a person might delegate a task is precisely because they cannot do that task themselves.

For judgment, as a heuristic, “can do a task” implies “knowing if a task was done.” This likely holds because a human performing some task is guiding that process to some outcome, and they presumably have to know where they are “going.” Even the simplest manual tasks have this property. A barista handing a customer a cup of coffee could release the cup in thousands of unique positions and times in 4D space-time, but only a very small subset of those leave the customer holding the cup and not scalded with spilled coffee. But  $p \Rightarrow q$  does not imply  $\neg p \Rightarrow \neg q$ . A person who had broken both arms could not hand a customer a cup of coffee but would still be able to know if a task was done.

There are likely to be numerous tasks when prompting and judgment can lead to good outcomes even when the human offering those judgments cannot do it themselves. One reason is that the pool of people who can offer judgment is larger than the pool that can do the task. This might be particularly true for tasks that require a great deal of skill. As a recent illustrative example, the famed record product Rick Rubin described his value in a recent interview with Anderson Cooper:

## 8 Conclusion

### 8.1 Why the details matter

The history of technological innovation offers few completely general labor market lessons. If there are any, they are the following: much of it tends to be labor-saving; jobs disappear, but work does not; complementary skills become more valuable, including whatever tasks humans still do (Autor, 2015). But it is difficult to know which particular jobs are impacted, what kinds of workers are complemented and what kinds are substituted, where productivity growth happens, and so on. For that, precisely what that technology does matters. For example, we have only made progress by getting into the details of say, what computers actually do and then exploring how this affects the productive process, as in (Autor et al., 2003).

## References

- Acemoglu, Daron and David Autor**, “Skills, tasks and technologies: Implications for employment and earnings,” in “Handbook of labor economics,” Vol. 4, Elsevier, 2011, pp. 1043–1171.
- and **Pascual Restrepo**, “The Race between Man and Machine: Implications of Technology for Growth, Factor Shares, and Employment,” *American Economic Review*, June 2018, *108* (6), 1488–1542.
- and —, “Automation and New Tasks: How Technology Displaces and Reinstates Labor,” *Journal of Economic Perspectives*, May 2019, *33* (2), 3–30.
- Agrawal, Ajay, Joshua S Gans, and Avi Goldfarb**, “Artificial Intelligence: The Ambiguous Labor Market Impact of Automating Prediction,” February 2019, (25619).
- Autor, David H**, “Why are there still so many jobs? The history and future of workplace automation,” *Journal of economic perspectives*, 2015, *29* (3), 3–30.
- , **Frank Levy, and Richard J Murnane**, “The skill content of recent technological change: An empirical exploration,” *The Quarterly journal of economics*, 2003, *118* (4), 1279–1333.
- Bresnahan, Timothy**, “Artificial intelligence technologies and aggregate growth prospects,” 2020. Working Paper.

**Bresnahan, Timothy F, Erik Brynjolfsson, and Lorin M Hitt**, “Information technology, workplace organization, and the demand for skilled labor: Firm-level evidence,” *The quarterly journal of economics*, 2002, 117 (1), 339–376.

**Eloundou, Tyna, Sam Manning, Pamela Mishkin, and Daniel Rock**, “Gpts are gpts: An early look at the labor market impact potential of large language models,” *arXiv preprint arXiv:2303.10130*, 2023.

**Kojima, Takeshi, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa**, “Large Language Models are Zero-Shot Reasoners,” 2023.

**Kremer, Michael**, “The O-Ring Theory of Economic Development\*,” 08 1993.

**Noy, Shakked and Whitney Zhang**, “Experimental evidence on the productivity effects of generative artificial intelligence,” *Available at SSRN 4375283*, 2023.