

Airlines on Autopilot: From AI to Intelligent Agents

Michael Wu, PhD (@mich8elwu)
chief AI strategist @ PROS

2025.10.07



PROS

Michael Wu, PhD (@mich8elwu)
chief AI strategist @ PROS

2025.10.07

5 stages of GenAI

AGI

innovator

agent

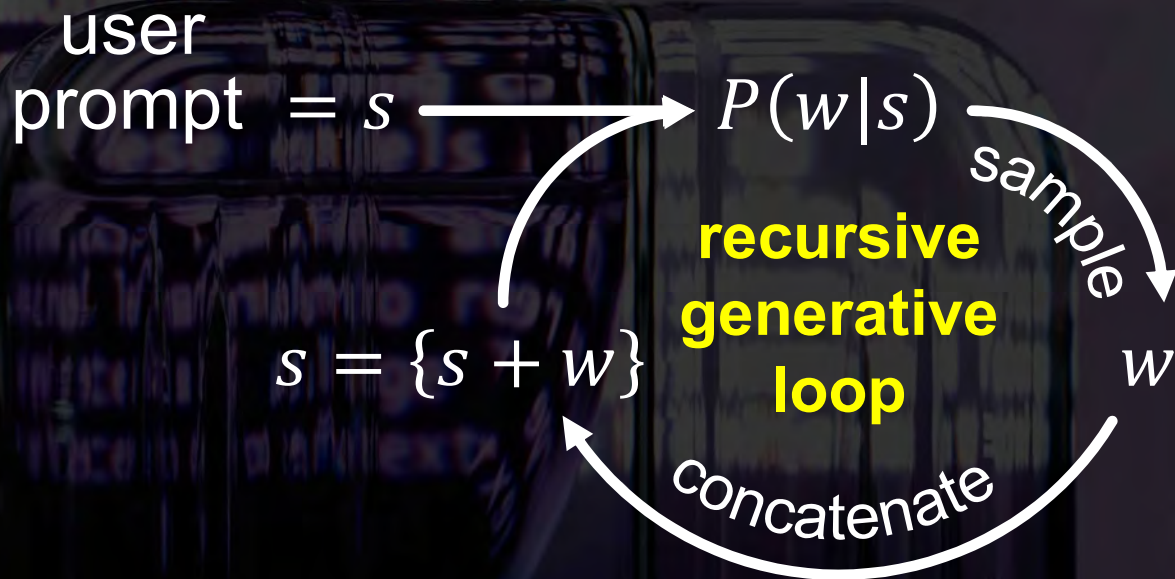
problem solver
(LRM)

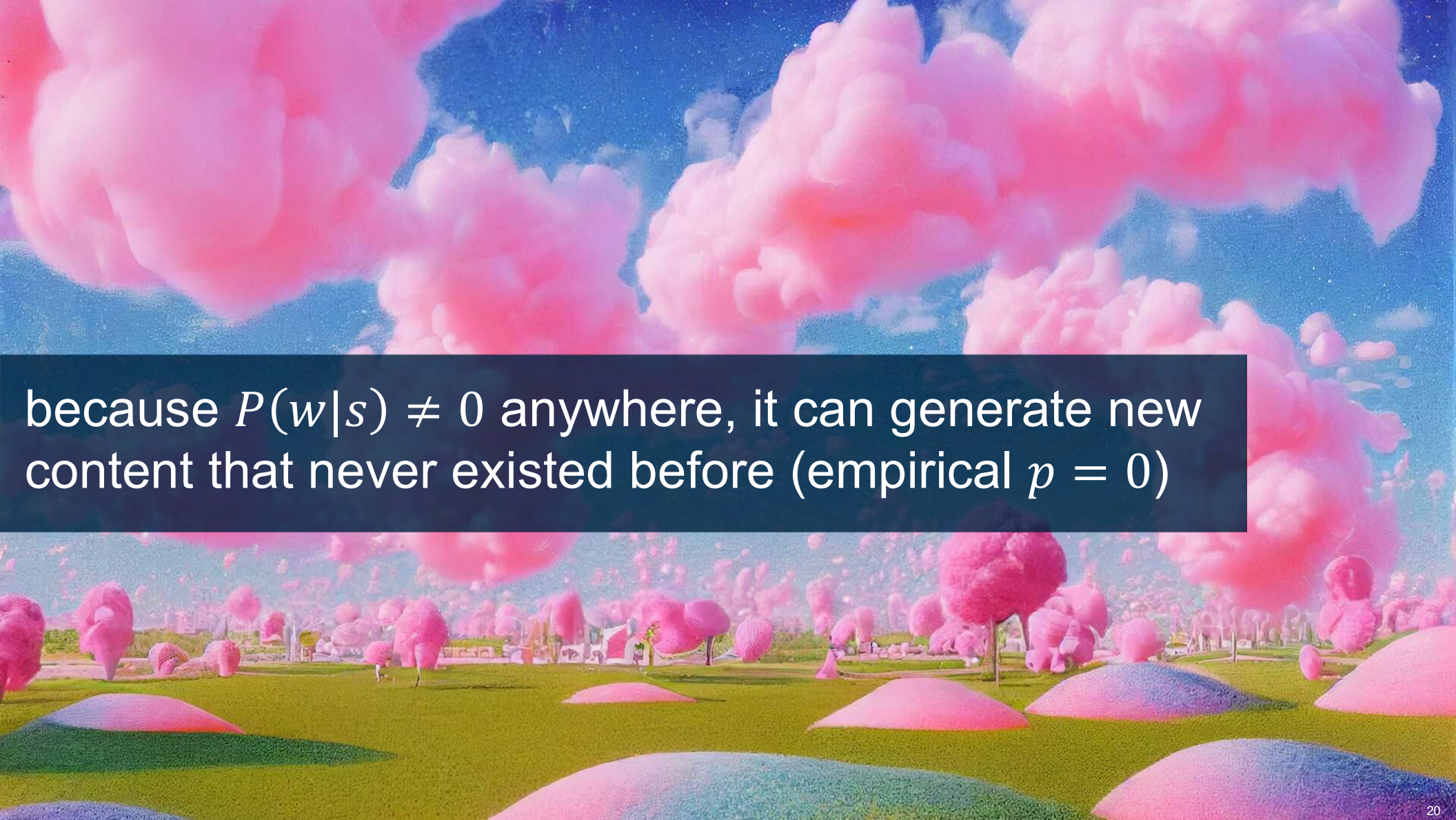
chatbot (LLM)



language model:

attempting to predict the next word w given *any* sequence s by modeling





because $P(w|s) \neq 0$ anywhere, it can generate new content that never existed before (empirical $p = 0$)

5 stages of GenAI

AGI

innovator

agent

problem solver
(LRM)

chatbot (LLM)





reasoning is fundamental
to all problem solving



**chain of
thought
prompting**

**improves LLM
performance**

Chain of Thought (CoT) Prompting Example

prompt:

a farmer has 10 sheep; **all** but 3 run away, and so he got 2 more; how many sheep does he have now?

LLM response:

9 sheep

INCORRECT

Chain of Thought (CoT) Prompting Example

prompt:

a farmer has 10 sheep; all but 3 run away, and so he got 2 more; how many sheep does he have now?

LLM response:

9 sheep

INCORRECT

prompt:

a farmer has 10 sheep; all but 3 run away, and so he got 2 more; how many sheep does he have now?
please think step by step

LLM response:

1. The farmer starts with 10 sheep
2. The phrase "all but 3 run away" means that 3 sheep did NOT run away
3. So, the farmer still has 3 sheep left
4. he got 2 more, so now he has $3 + 2 = 5$ sheep

final answer: 5 sheep

**4 “thinking” steps for
this simple prompt**

CORRECT

A conceptual image featuring a metallic chain against a sky with clouds and a bright sun. One link in the chain is broken and bent into a wavy, zig-zag shape, while the other links remain straight and connected. Below the main chain, several individual chain links are shown floating or falling, suggesting a process of disassembly or a chain of events. The background is a clear blue sky with soft, white clouds, and a bright sun is visible in the upper left, creating a lens flare effect.

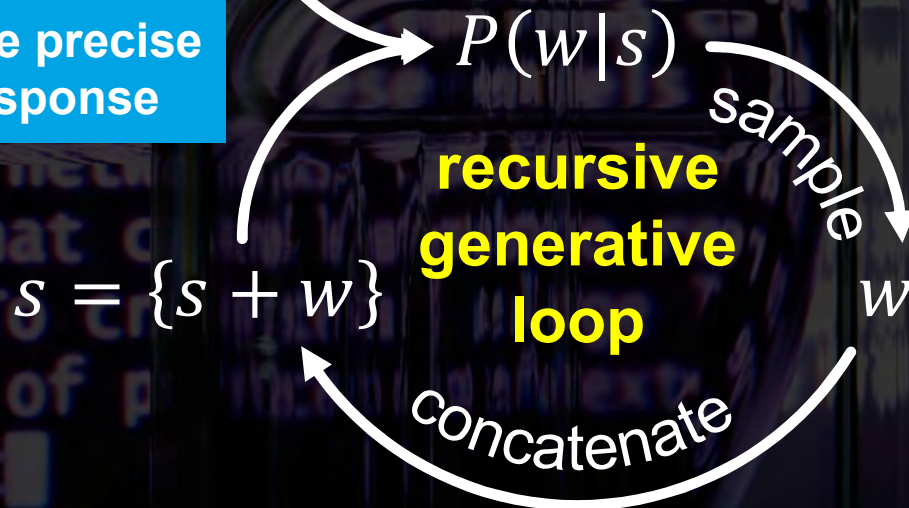
why would CoT (simply asking the LLM to think step by step), without any knowledge enhancement (further training or RAG) improve LLM's accuracy?

the effect of CoT is like writing long and detail prompts, except you don't need to write it yourself

user
prompt = s

longer and more detail prompts → more precise and accurate response

CoT must generate many more tokens as output to explain the steps





if accuracy of a single
CoT step is 99%

many real-world
problems may involve
tens of steps

accuracy of entire CoT
after 20 steps:
 $(99\%)^{20} = 81.8\%$

after 50 steps:
 $(99\%)^{50} = 60.5\%$

challenging scientific
problems may require
hundreds of steps

after 100 steps:
 $(99\%)^{100} = 36.6\%$

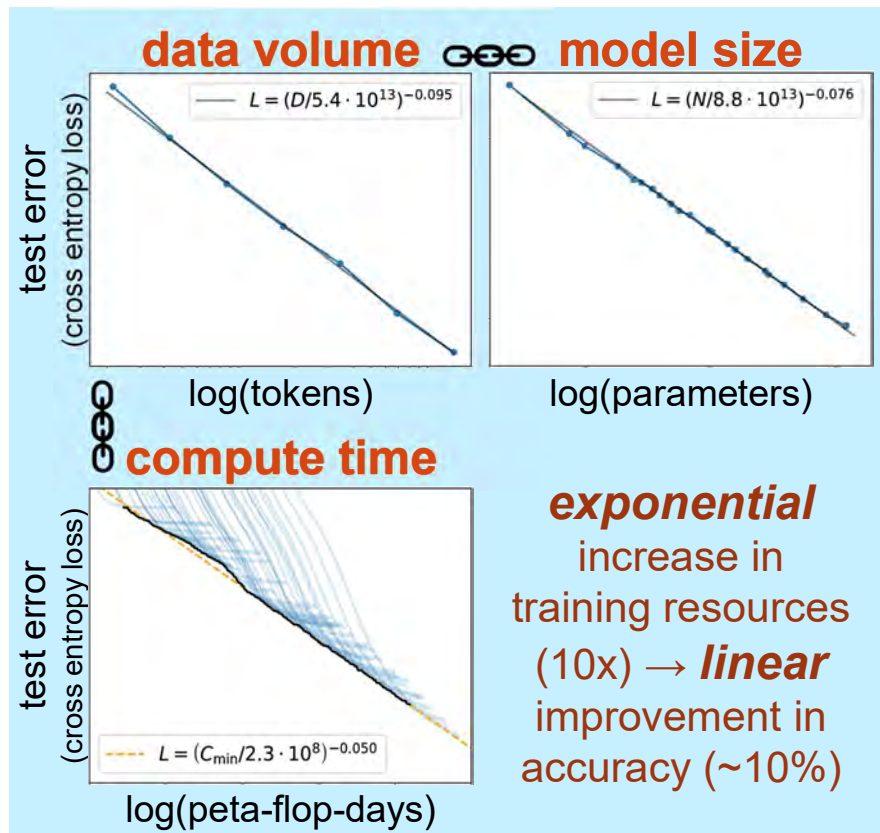
deep reasoning (long CoT)
require ~100% LLM accuracy

How to Improve LLM Accuracy? – LLM Scaling Laws

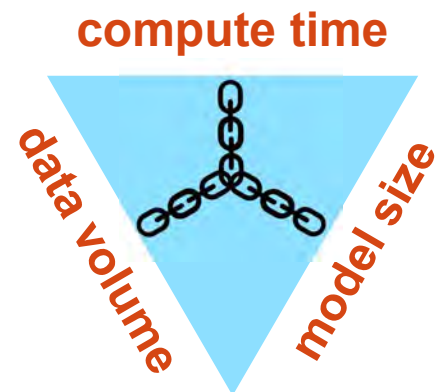
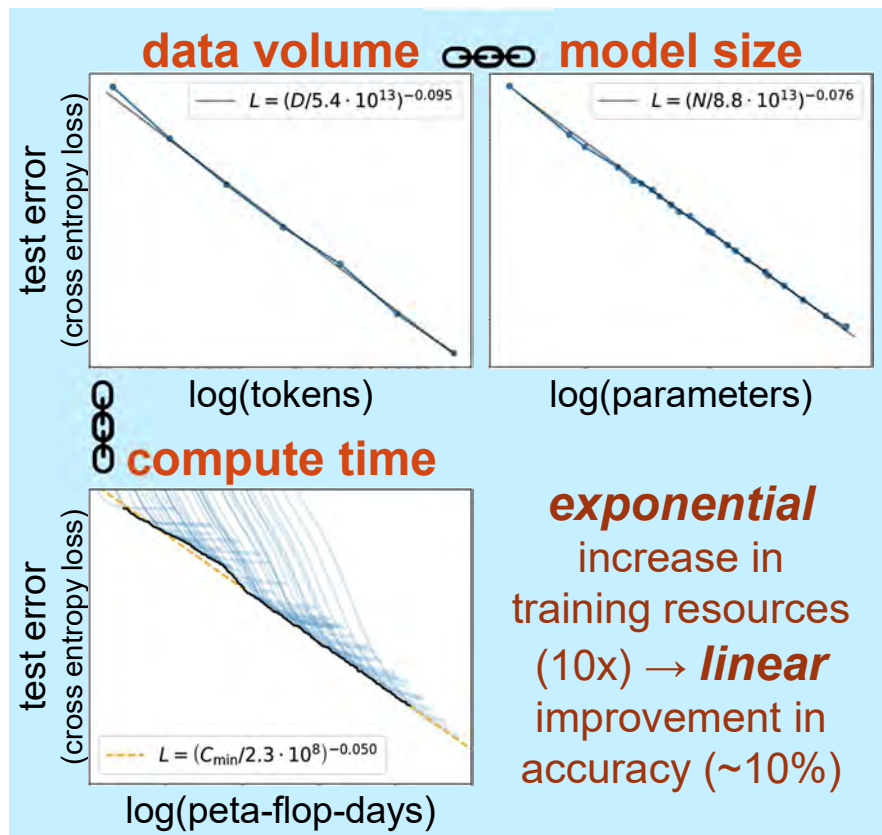
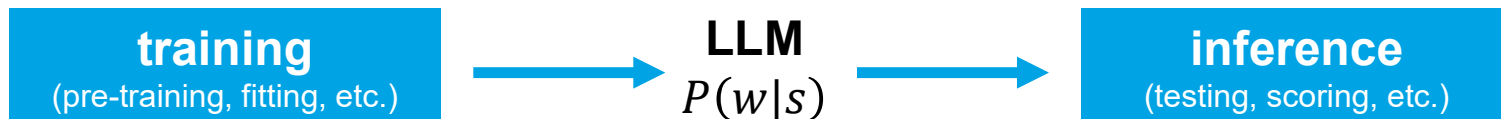
training
(pre-training, fitting, etc.)

LLM
 $P(w|s)$

inference
(testing, scoring, etc.)



How to Improve LLM Accuracy? – LLM Scaling Laws



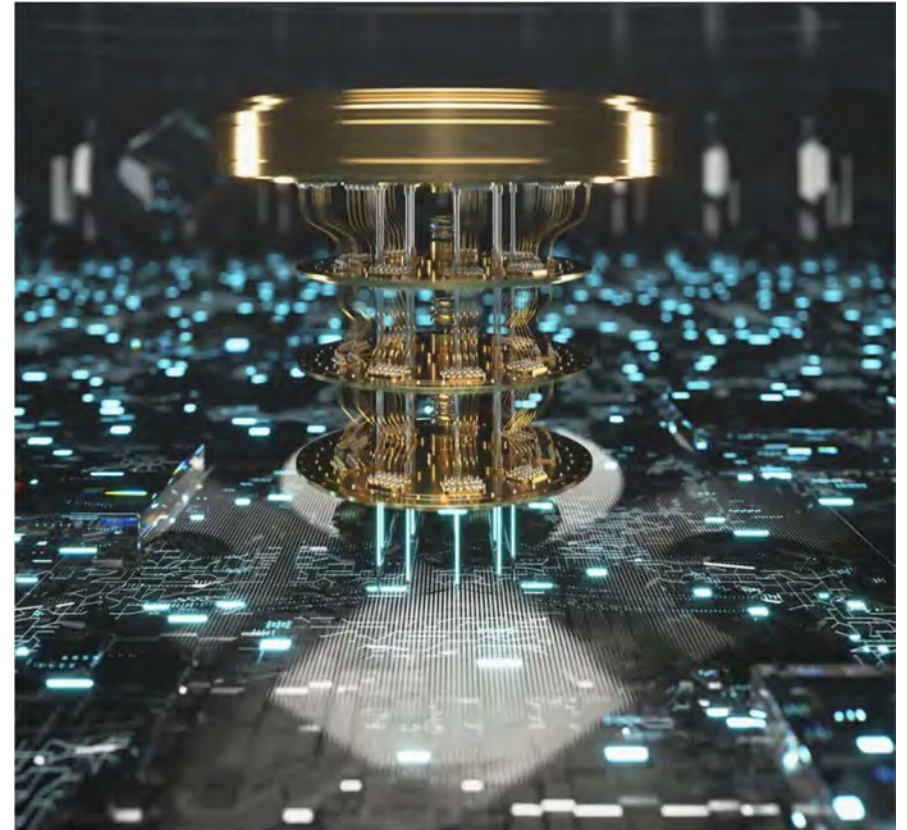
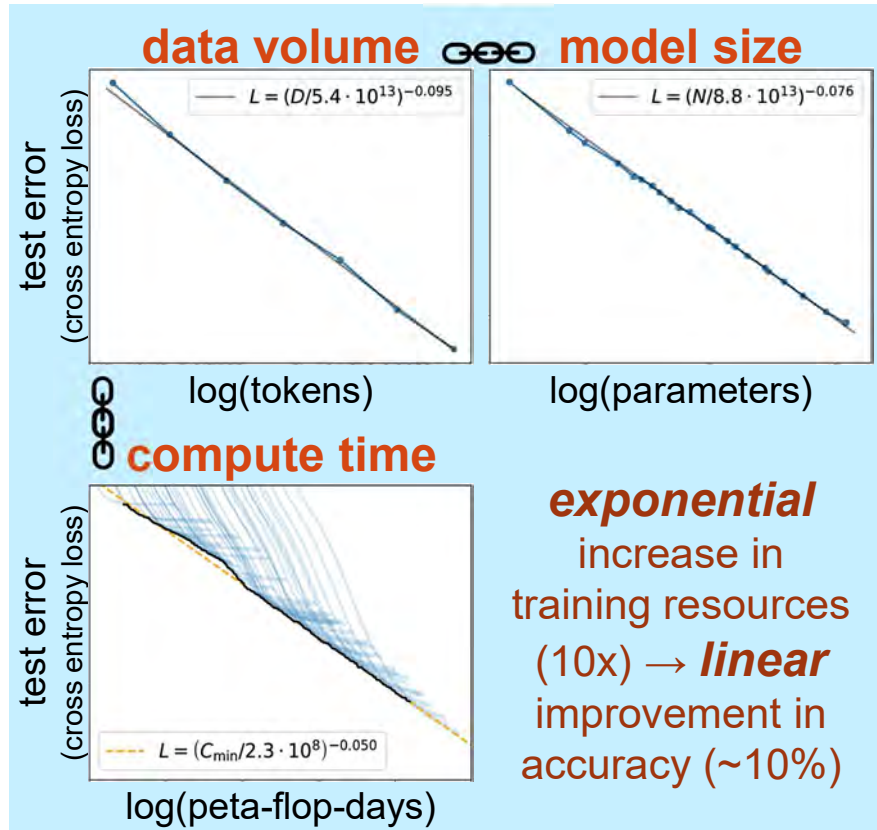
we are out of
publicly
accessible data

How to Improve LLM Accuracy? – LLM Scaling Laws

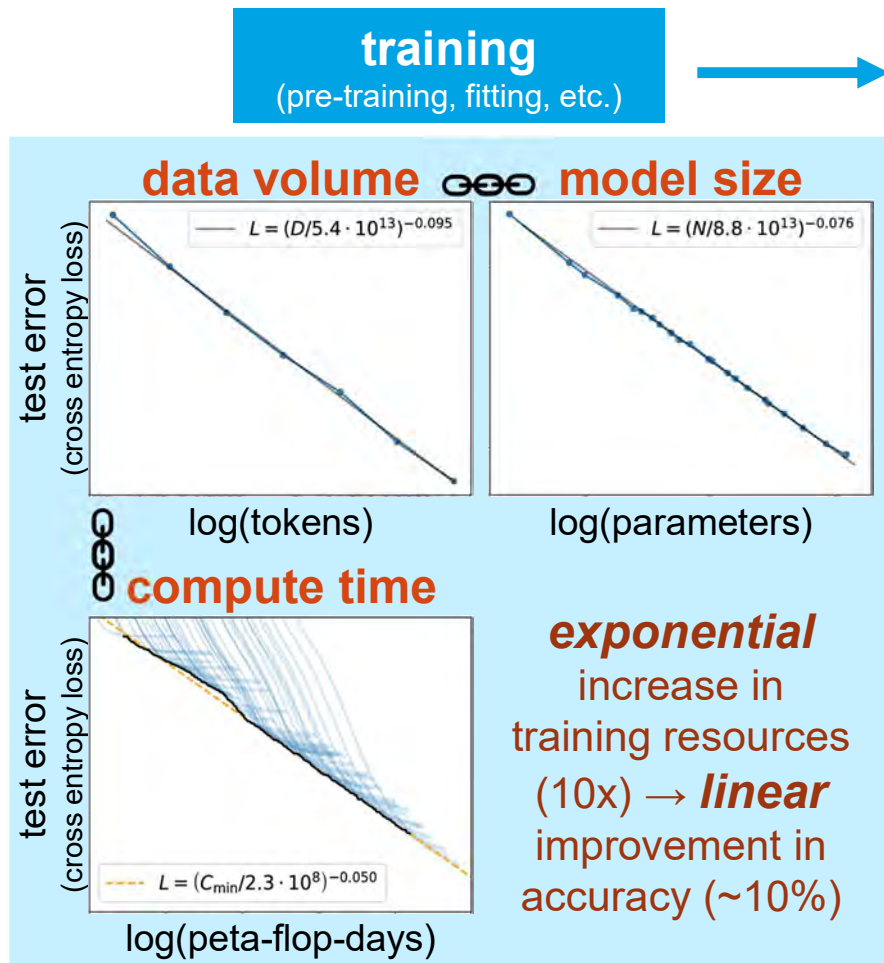
training
(pre-training, fitting, etc.)

LLM
 $P(w|s)$

inference
(testing, scoring, etc.)

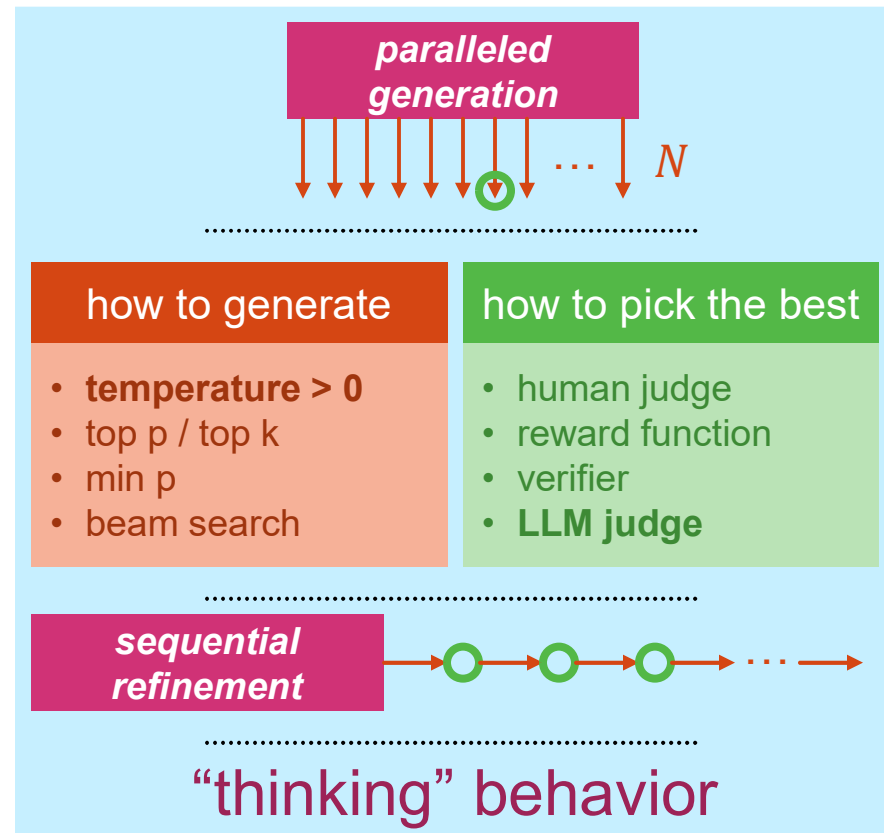


How to Improve LLM Accuracy? – LLM Scaling Laws



LLM
 $P(w|s)$

inference
(testing, scoring, etc.)



```
<?php
$host="localhost"; // Host name
$username="root"; // Mysql username
$password=""; // Mysql password
```

```
function setup(){
```

**sequential
refinement**

```
    struct buffer
    {
        text_area: TextAreaEditable, full[size]
        KeyPress('CTRL+s') =>
        {
            size_t start;
            path = AskPath('Save As', 'Save', 'Cancel')
            open(path, "wt").write(text_area.text)
        }
        KeyPress('CTRL+o') =>
        {
            struct buffer
            {
                path = AskPath("Open", "Open", "Cancel")
                text_area.text = open(path).read()
                struct buffer *result
            }
            result = malloc(sizeof(struct buffer))
            if (result == NULL)
            {
                use random, now_inag;
                return NULL;
            }
            image = new_image(1500, 500)
            image.fill('Navy Blue')
            result->ptr = malloc(size);
            repeat (40){ result->ptr = NULL}
```



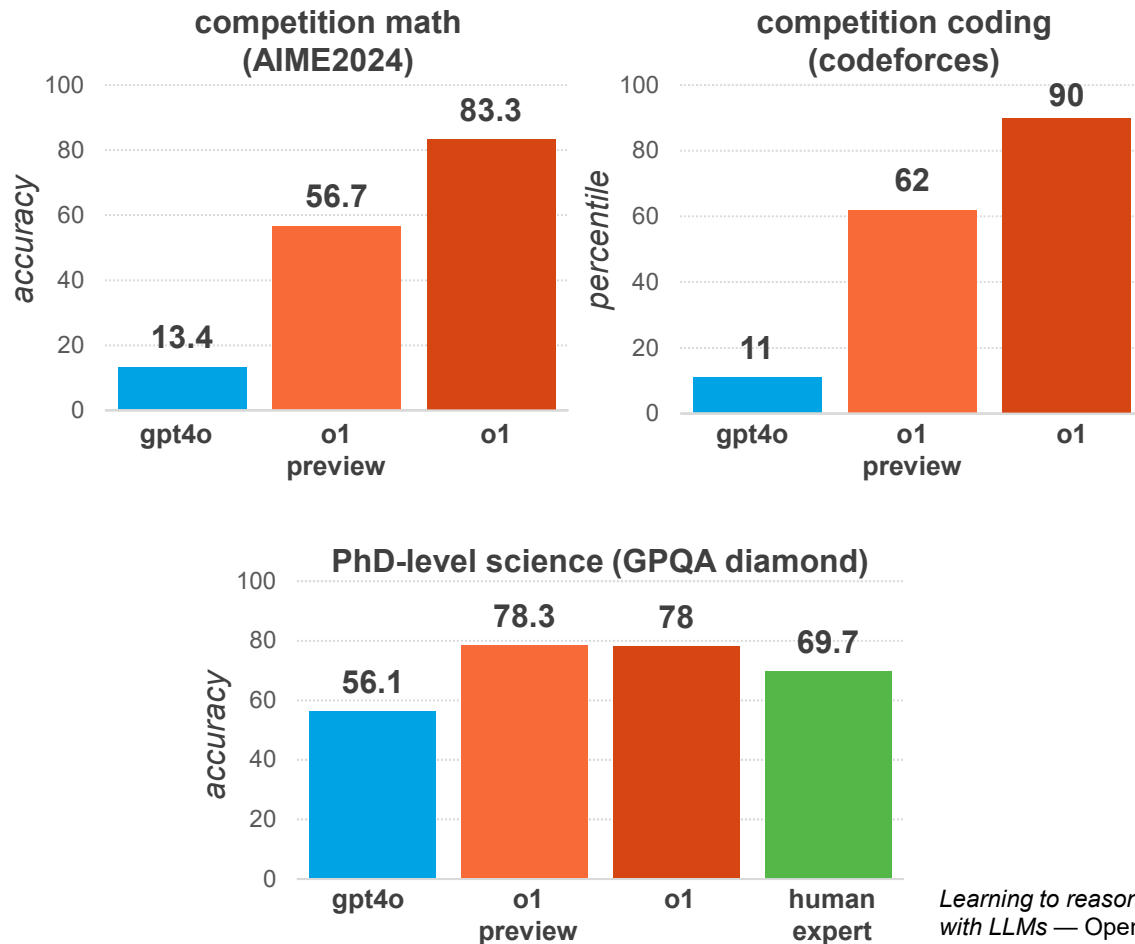
**paralleled
generation**



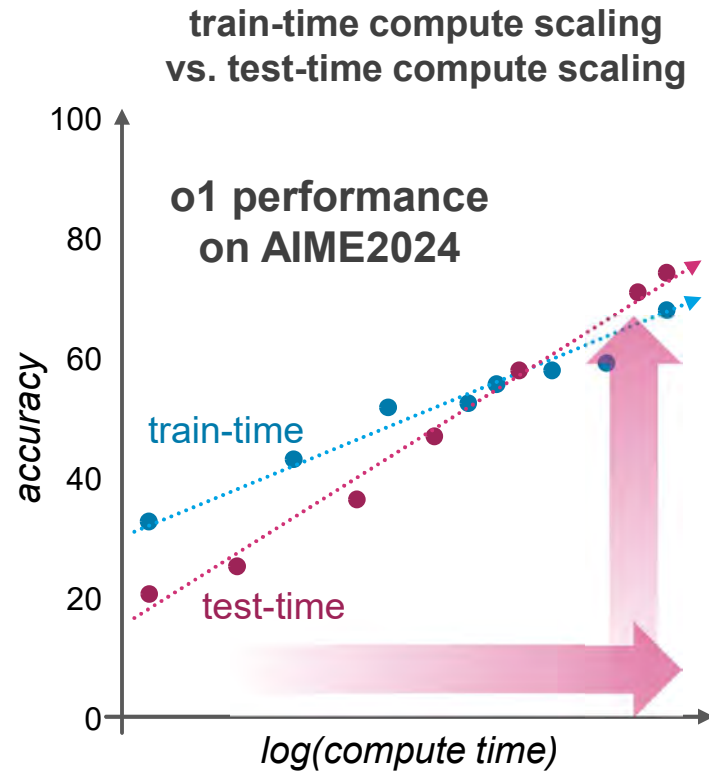
N

scaling inference time “thinking” requires
big working memory (long context window)

Test-Time Compute (TTC): Performance + Scaling



Learning to reason
with LLMs — OpenAI



continuing to scale TTC →
continually improving accuracy



new scaling law:
LLM performance $\propto \text{TTC}^{-\alpha}$ (power law)

all you need is compute

sequential
refinement

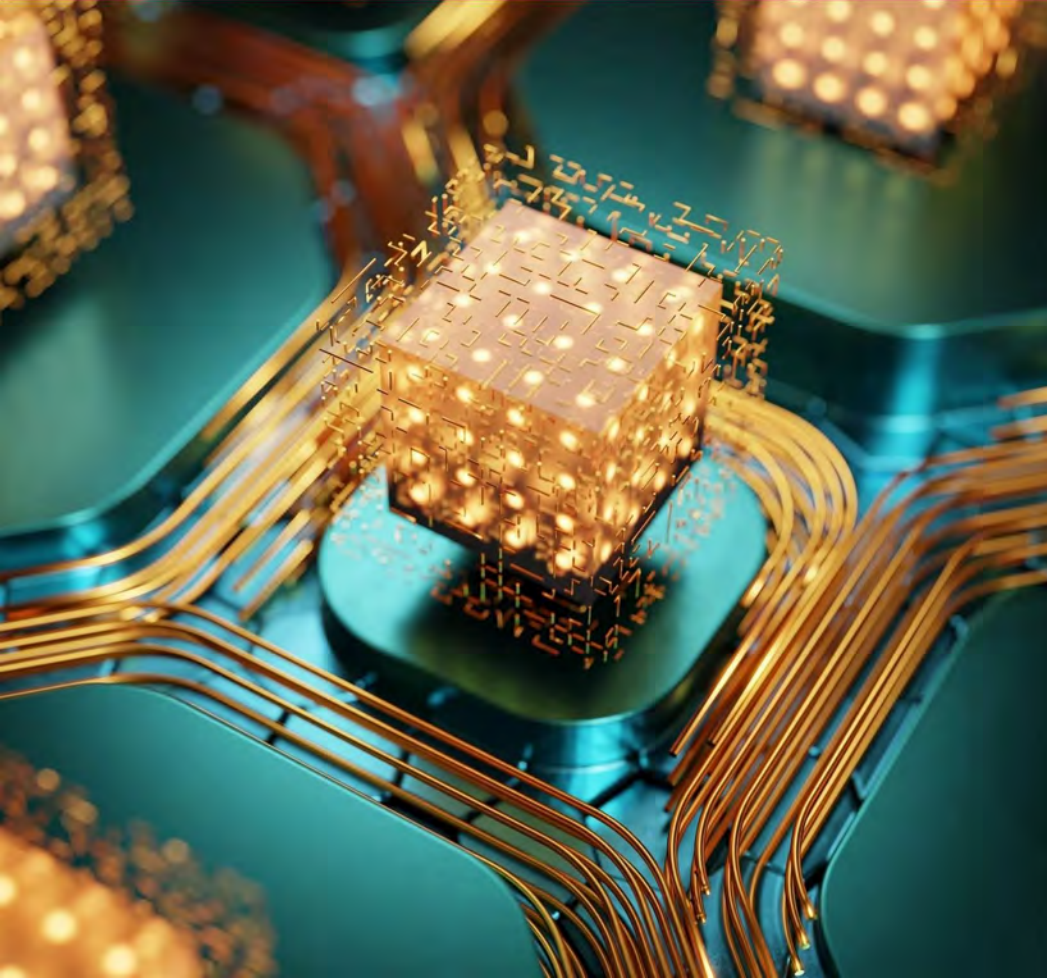


paralleled
generation



We Just Need More Compute: Future AI Data Center Investment

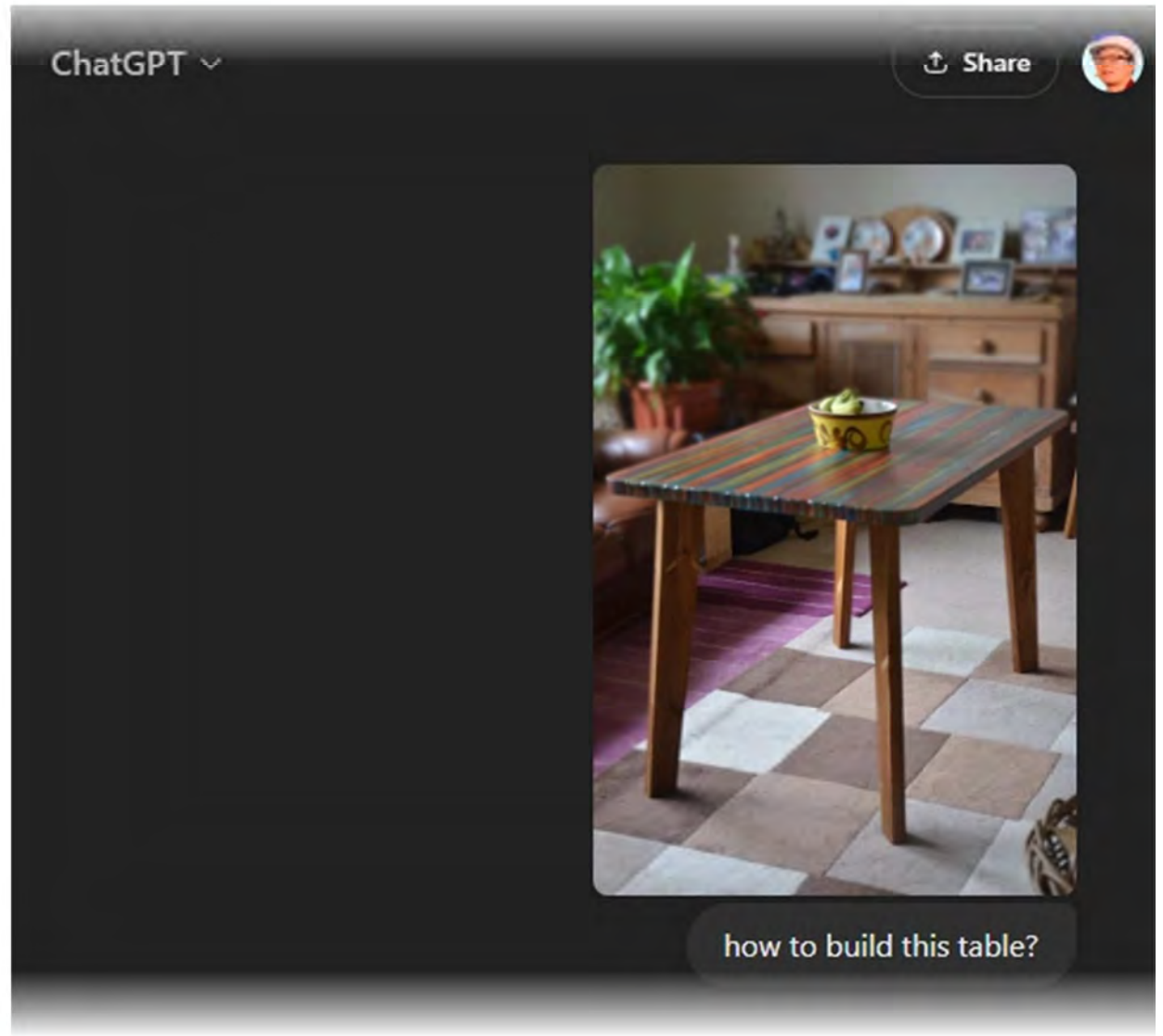
quantum compute



fusion energy



LLMs are Passive



LLMs are Passive



Building a table like the one in the image involves several steps, including selecting materials, assembling the frame, and finishing the tabletop.

Here's a step-by-step guide:

Materials Needed

1. **Wood for the tabletop** – Plywood, MDF, or reclaimed wood.
2. **Wood for the legs** – Solid wood (such as pine or oak).
3. **Wood screws and brackets** – For attaching the legs.
4. **Wood glue** – For extra strength.
5. **Sandpaper** – For smoothing edges.
6. **Primer and paint or resin** – For the colorful finish.
7. **Clear epoxy or varnish** – To seal the painted surface.
8. **Drill and screwdriver** – For assembling the frame.
9. **Saw** – To cut the wood to size.
10. **Clamps** – For securing pieces while gluing.

LLMs are Passive



Step 1: Preparing the Tabletop

- Measure and cut the tabletop to the desired size.
- Sand the edges and surface for a smooth finish.
- Paint or apply the colorful design:
 - Use acrylic paints or wood stains to create the multi-colored striped effect.
 - Allow each layer to dry before adding the next.
 - Optionally, use a brush or sponge for texture.

Step 2: Building the Legs

- Cut four wooden legs to your desired height.
- Use tapered or straight legs based on preference.
- Sand the legs to ensure smoothness.

LLMs are Passive



Step 3: Attaching the Legs

- **Option 1: Screw and Bracket Method**
 - Attach metal brackets under the tabletop.
 - Secure the legs to the brackets with screws.
- **Option 2: Pocket Hole Joinery**
 - Drill angled pocket holes into the legs.
 - Attach them to the tabletop with screws and wood glue.

Step 4: Finishing the Table

- Apply a clear coat of epoxy or varnish to protect the painted surface.
 - Let it dry completely before use.
-

LLMs are Passive



- Drill angled pocket holes into the legs.
- Attach them to the tabletop with screws and wood glue.

Step 4: Finishing the Table

- Apply a clear coat of epoxy or varnish to protect the painted surface.
- Let it dry completely before use.

Final Touch

Place the table in your desired space and enjoy your handmade furniture!

Would you like help with a more detailed design plan or measurements?



5 stages of GenAI

AGI

innovator

agent

problem solver
(LRM)

chatbot (LLM)



inspired by real-world concepts in software agents back in 1999

goal directed

programmed to keep
order in the system (i.e.
the Matrix)

context aware

can appear virtually
anywhere to monitor
anomalies everywhere

autonomy

can do whatever he deems
necessary (eliminating
humans + rogue programs)

tool usage

can use anything and
anything to give him
super-human power



inspired by real-world concepts
in software agents back in 1999

what's all the
hype about
agents now?



inspired by real-world concepts in software agents back in 1999

goal directed

programmed to keep
order in the system (i.e.
the Matrix)

context aware

can appear virtually
anywhere to monitor
anomalies everywhere

autonomy

can do whatever he deems
necessary (eliminating
humans + rogue programs)

tool usage

can use anything and
anything to give him
super-human power



**goal
management**

autonomy

**memory
(context
window)**

**reasoning
(CoT)**

**context
aware
(perception)**

**tool usage
(action)**

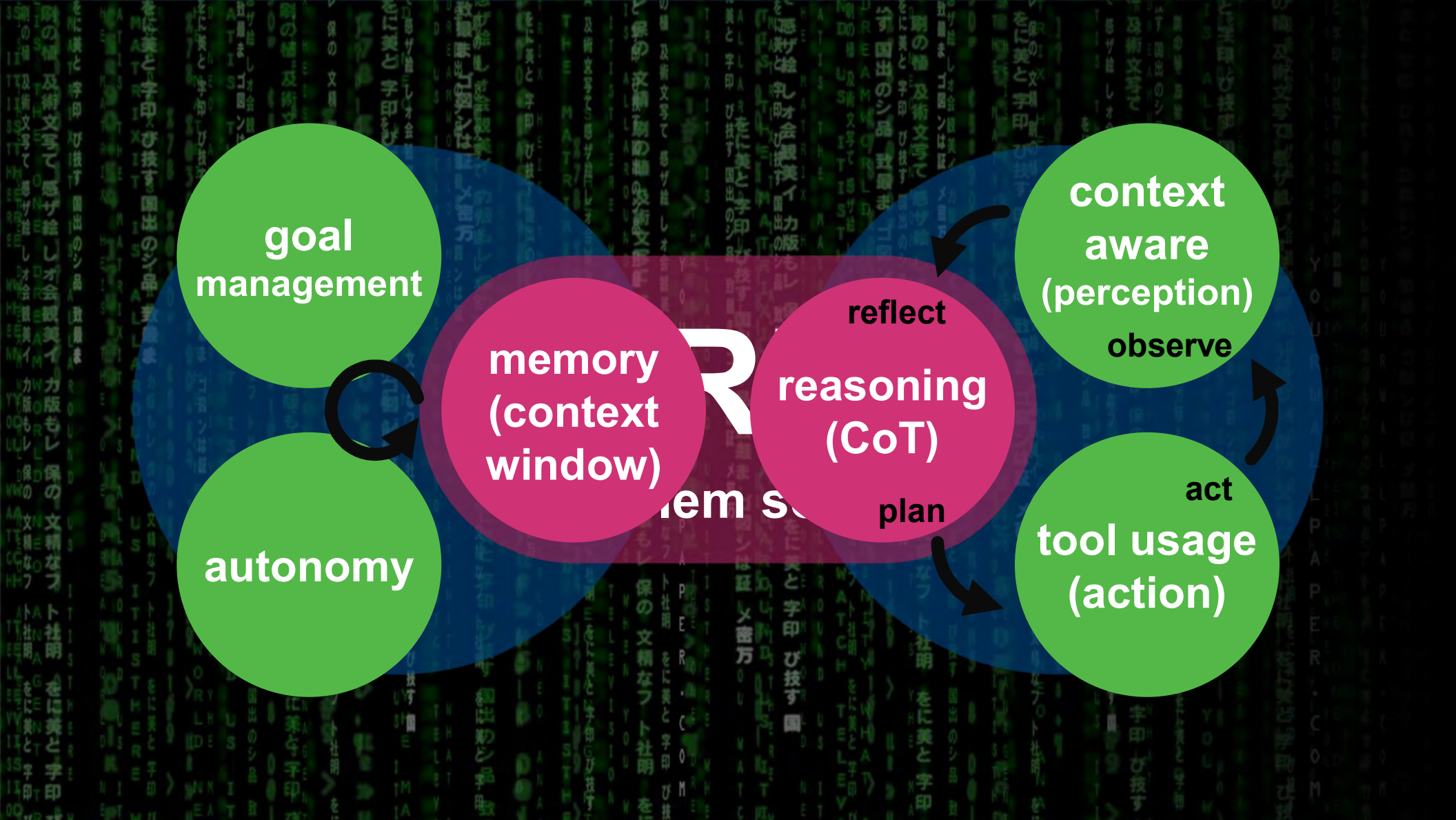
reflect

plan

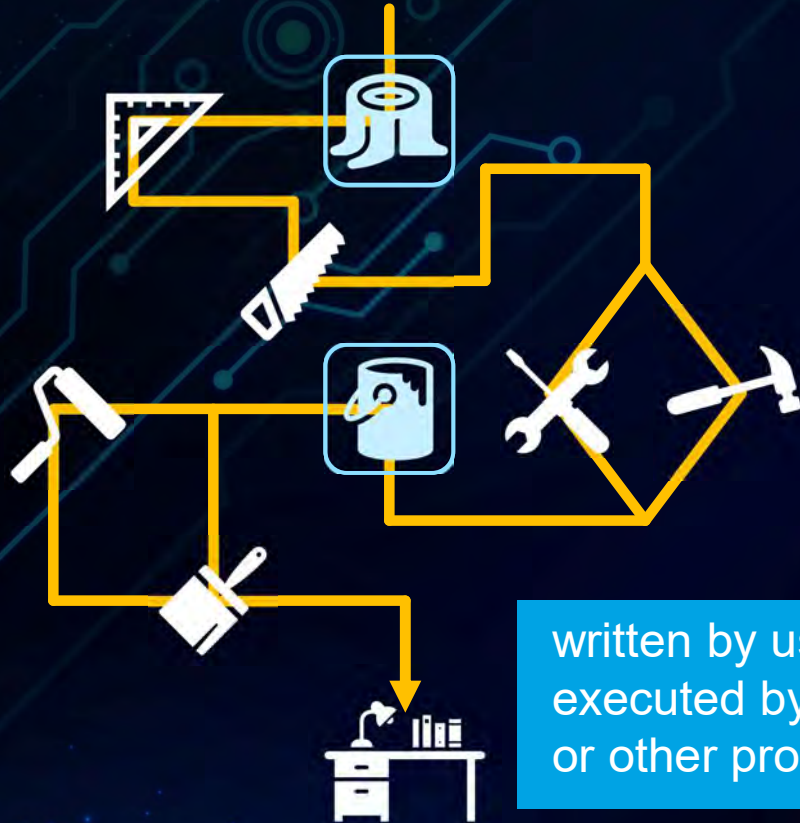
observe

act

problem solving



ordinary computer program



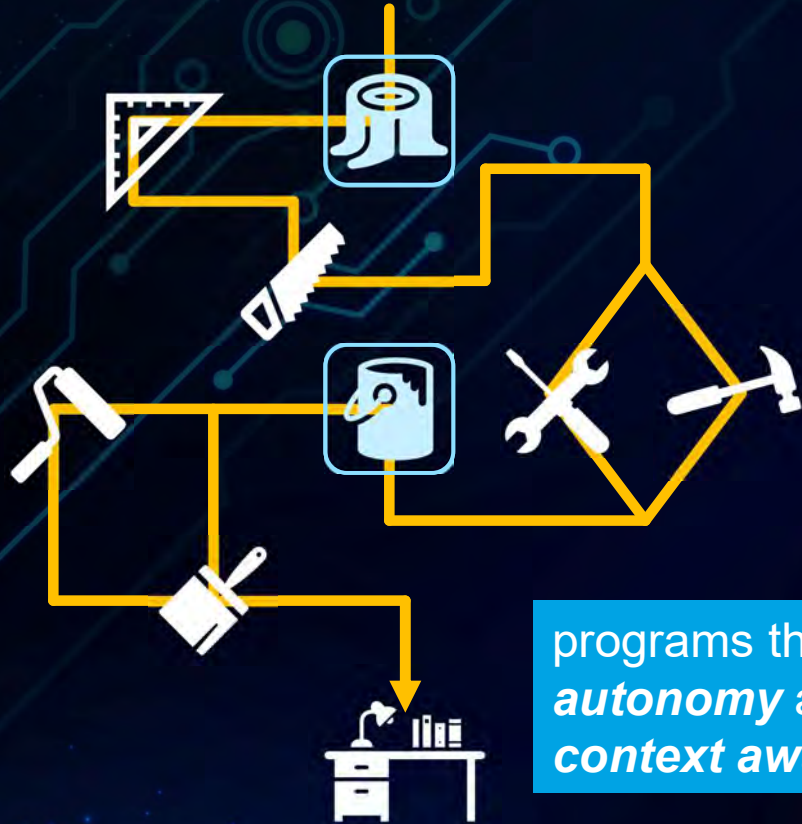
written by users
executed by users
or other programs

software agent

- can self execute (*autonomy*)
- must be able to observe the environment (*context aware*)
- still task specific (goal directed)
+ great for automation
- e.g. web crawler, load balancer,
network intrusion detection

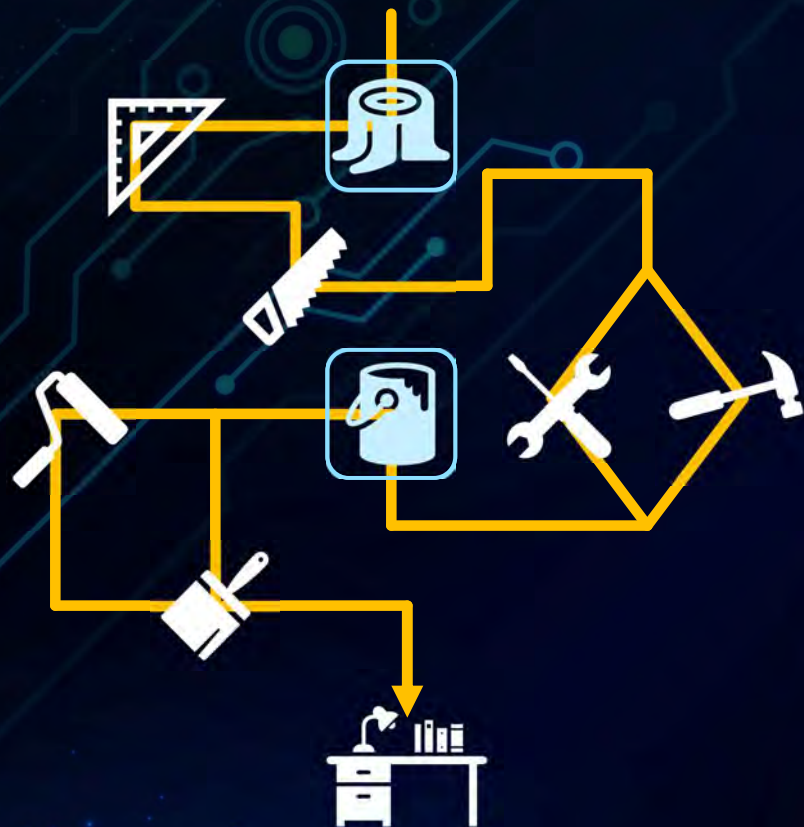


software agent



programs that has
autonomy and are
context aware

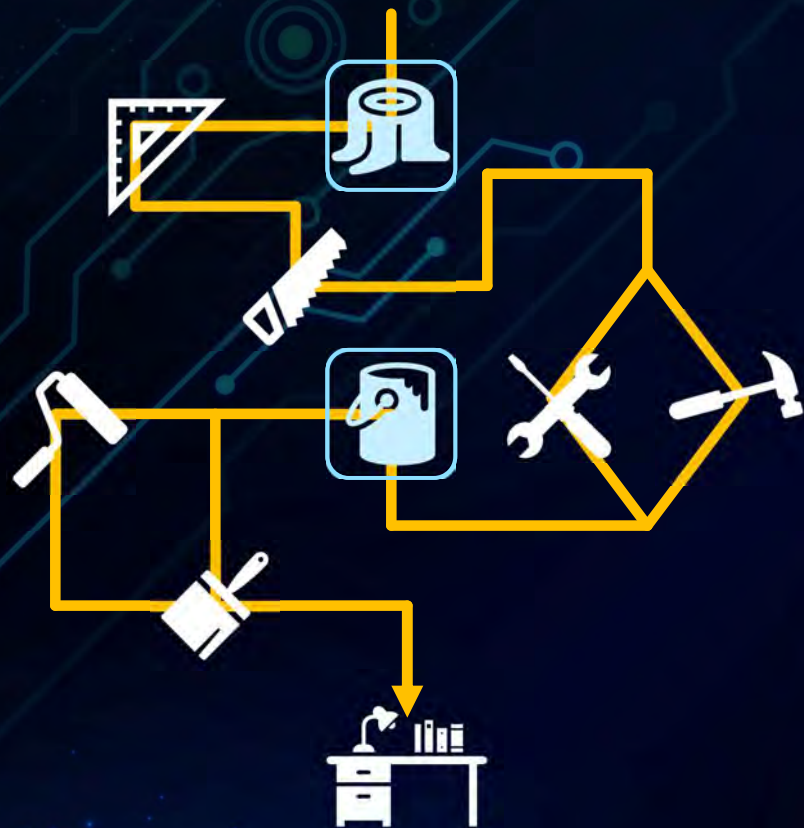
software agent



LLM agent



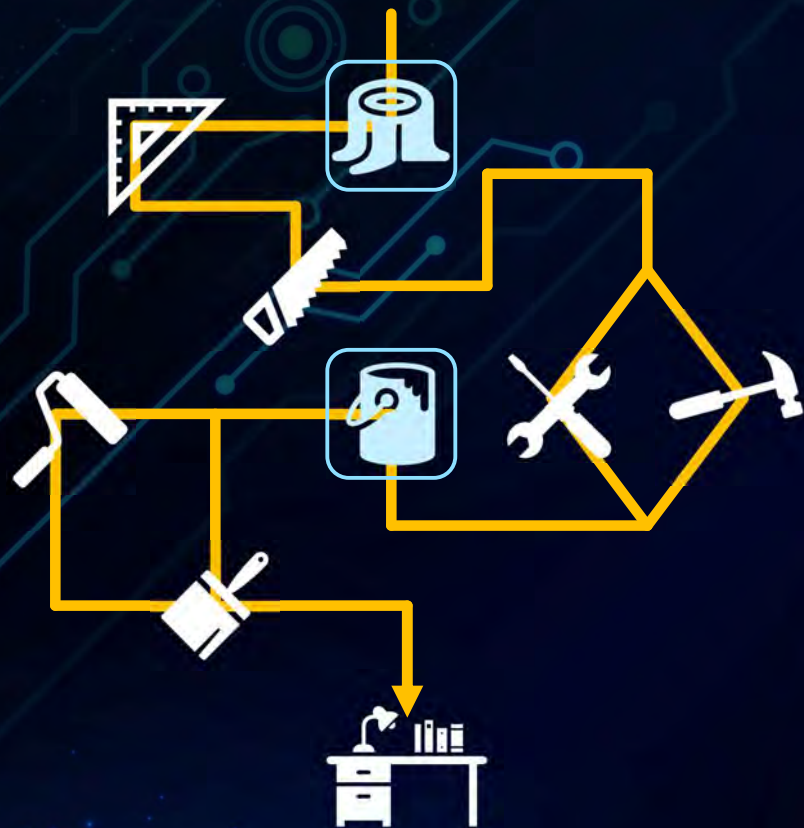
software agent



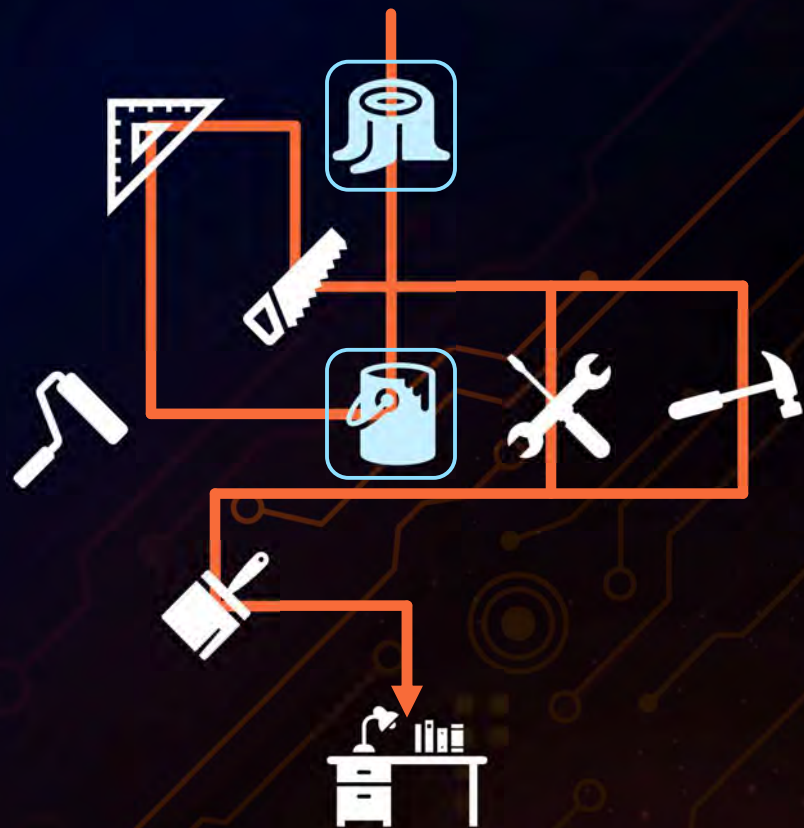
LLM agent



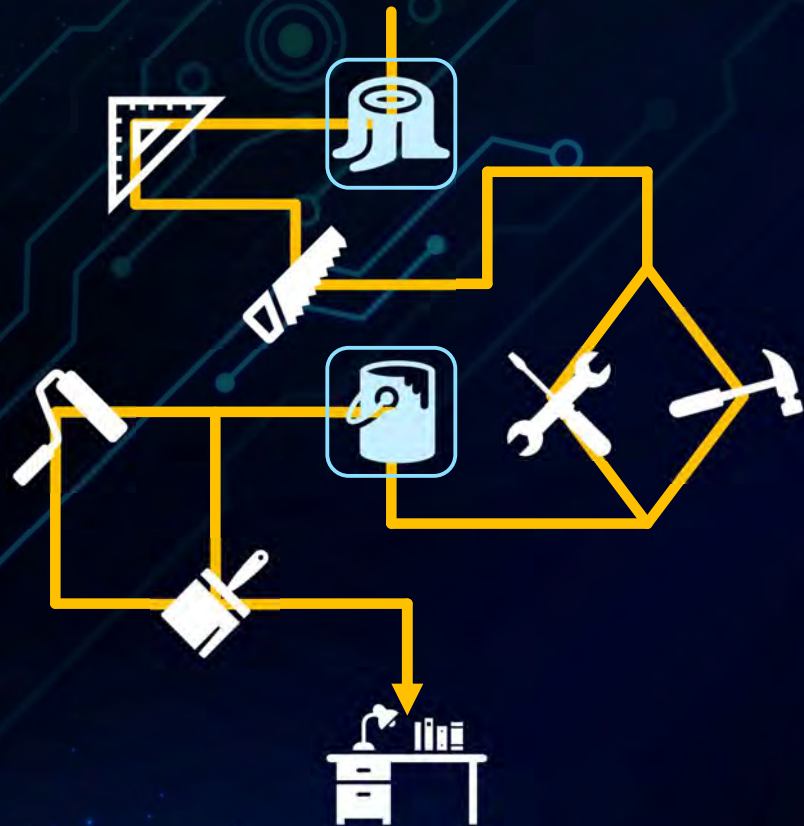
software agent



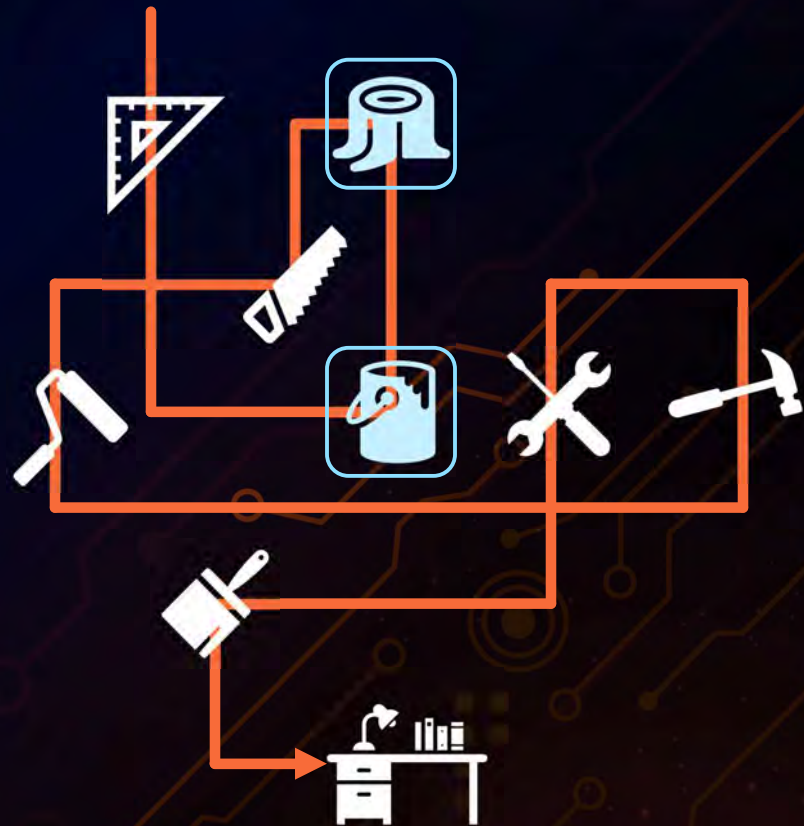
LLM agent



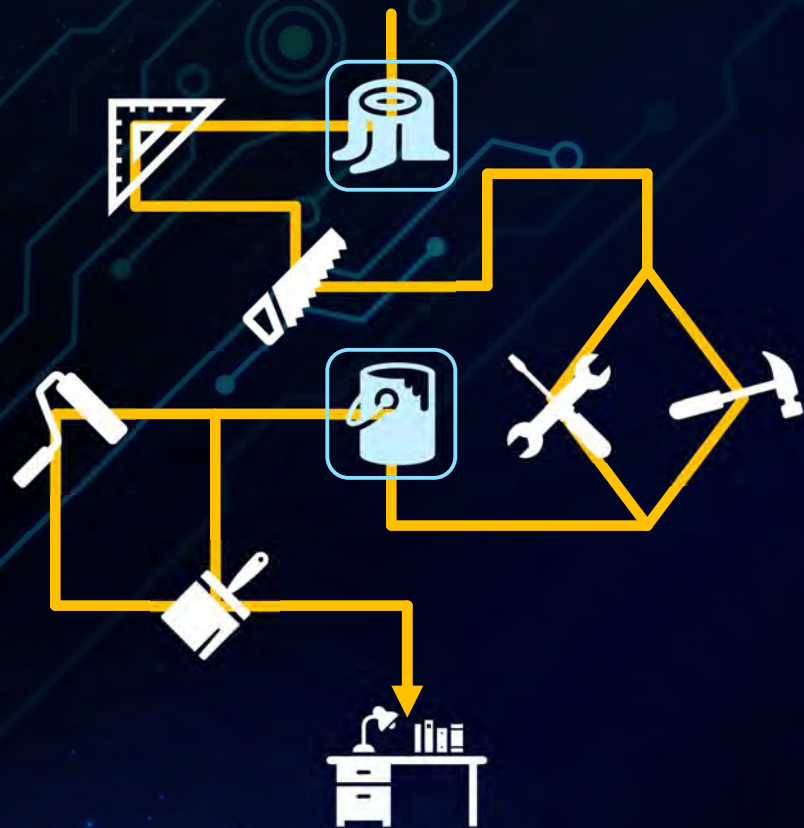
software agent



LLM agent



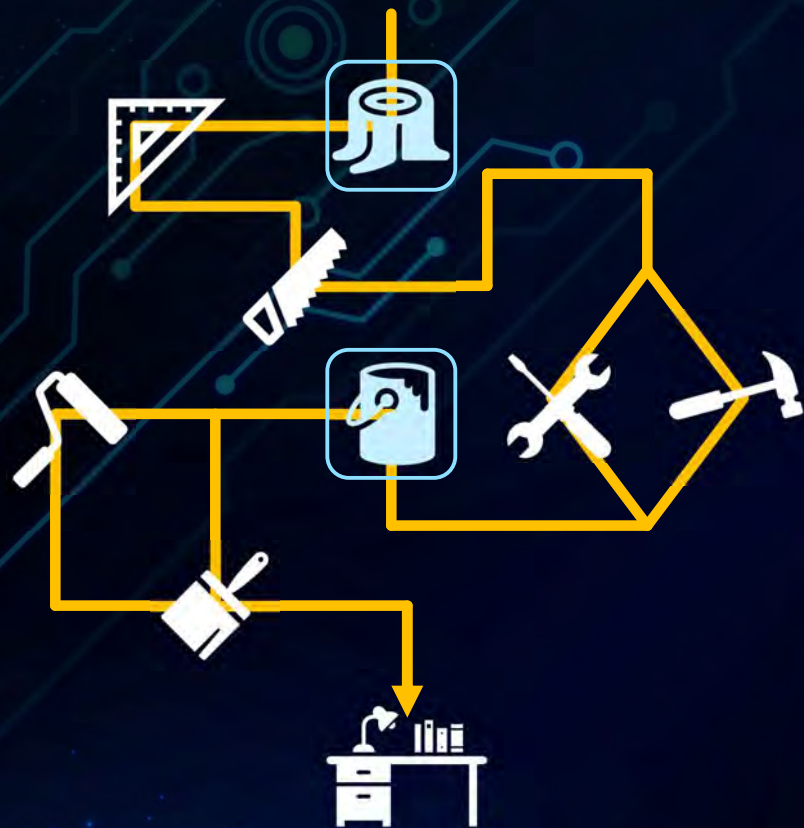
software agent



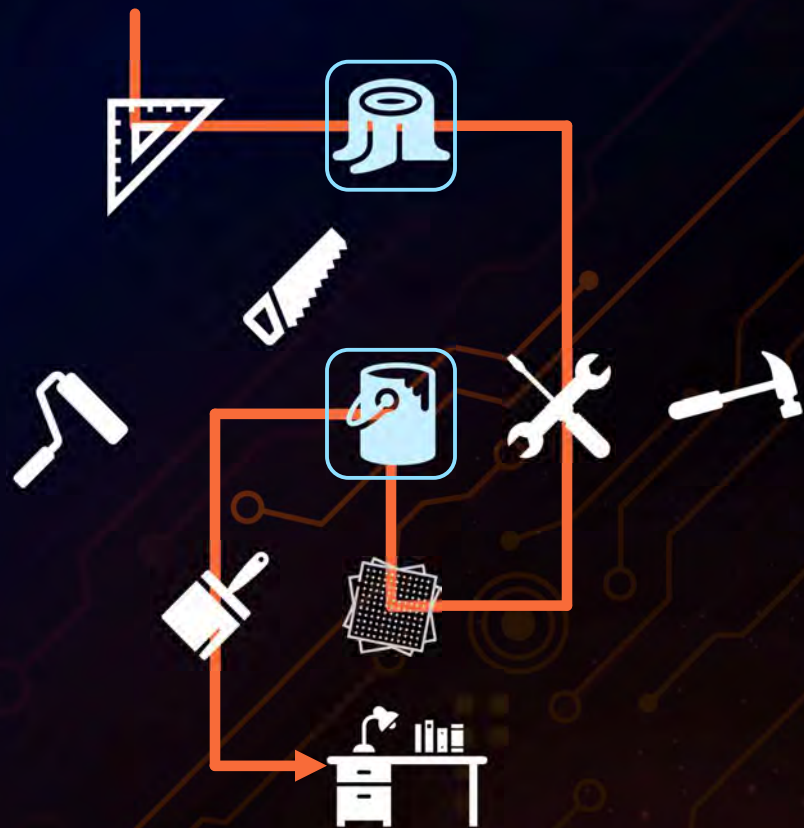
LLM agent



software agent



LLM agent





LLMs must use tools to act



LLMs must use tools to act,
and in the digital world, tools =
programs, API, or other agents

agents are a major step toward AGI
≡ AI that has human-level performance
at most *economically valuable work*



**can GenAI
generate**

revenue?

not by itself





an agent is only as good as
the *tools* it has access to

Future AI-Augmented Airline Retail

right time

conventional RM +
WTP forecaster

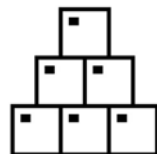


model-based
reinforcement
learning

right offer

ancillary AI algorithms

ranking
bundling
pricing



reinforcement
learning

right customer

request-specific
pricing



double debiased
neural network-based
robust estimation

right channel

offer marketing

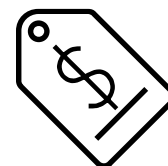
impossible to be
everywhere →
funnel them in



hyper-
personalization

right price

continuous pricing
+ WTP model



online learning w/
variational inference

Future AI-Augmented Airline Retail

right time

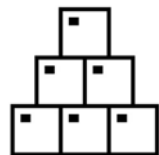
conventional RM +
WTP forecaster



model-based
reinforcement
learning

right offer

ancillary AI algorithms



dynamic ancillary
ranking
bundling
pricing
pricing



6%
reinforcement
learning
revenue lift

right customer

request-specific
pricing



>4.7%
double demand
neural network-based
robust estimation
revenue lift

right channel

offer marketing



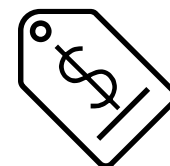
impossible to be
everywhere →
funnel them in



~192x
hyper-
personalization

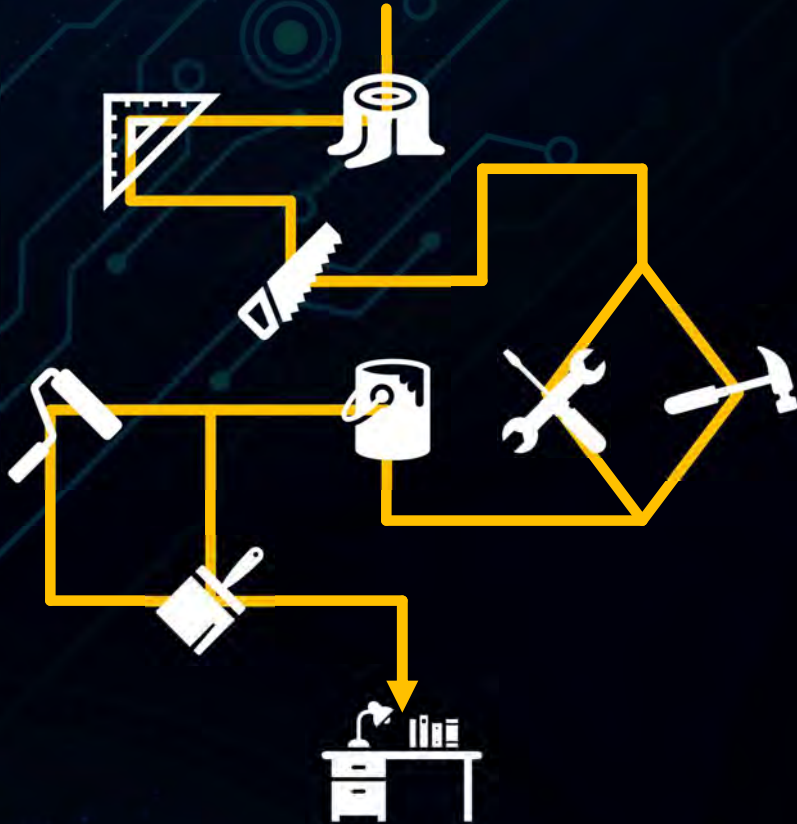
right price

continuous pricing
+ WTP model



1-3%
online learning w/
variational inference
revenue lift

ordinary computer program



LLM agent



A group of men in dark suits, white shirts, and dark ties, all wearing dark sunglasses. They are standing in a line, looking forward with serious expressions. The background is slightly blurred, showing some greenery and a building.

— who's ready to let agents make *all* their business decisions *fully* autonomously?

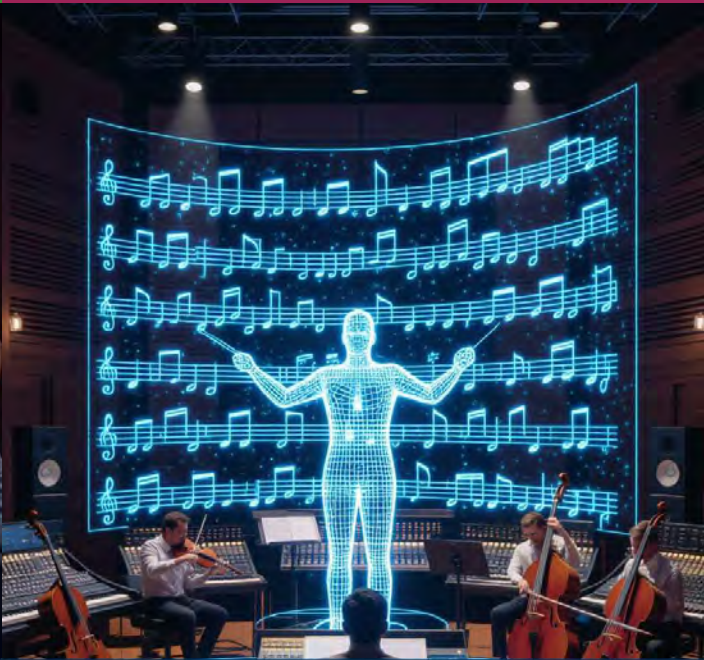
AI Automation Roadmap

AI assisted



human is the driver; AI is an assistant ready to help

AI led



AI is the driver; human is an approver for all work

AI automated



AI is the driver; human is an intervenor as needed

AI Automation Roadmap

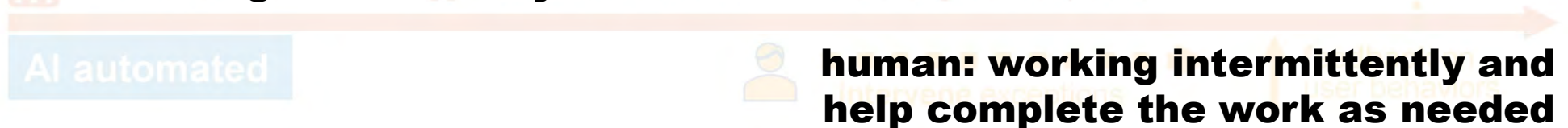


AI Automation Roadmap

**human: start the work
and working continuously**



**AI: start the work
and working continuously**



Some Examples of PROS Agents

AI assisted: merchandising bundling agent

PROS AI Beta

- February: 644 bookings
- March: 667 bookings

Least Frequently Purchased Ancillary Services

- CBAG (Third Excess Bag)**
 - January: 2 bookings
 - February: 5 bookings
 - March: 1 booking
- BBAG (Second Excess Bag)**
 - January: 9 bookings
 - February: 29 bookings
 - March: 17 bookings
- LOUG (LOUNGE ACCESS) in March**
 - March: 124 bookings

These insights are based on the available data from January to March 2025. If you have any specific requirements for bundle creation or optimization, please let me know!

👍

👎

📄

PROS AI

Analyzing your request...

Type a message...

PROS AI is still learning and may make mistakes. Chat history is stored until you log out.

PROS AI Beta

- With Extra Legroom** - Top-selling service for enhanced seating comfort.
- ABAG (First Excess Bag)** - Popular choice for additional baggage needs.

Offer Group 2: Exclusive Add-ons

- Offer 1: Fast Track and Lounge Access**
 - FAST (Fast Track)** - Enhance travel experience with expedited security clearance.
 - LOUG (Lounge Access)** - Least-selling service, included to drive adoption.

Offer Group 3: Comprehensive Travel Experience

- Offer 1: Excess Baggage Option**
 - BBAG (Second Excess Bag)** - Encourage additional baggage purchase.
 - LOUG (Lounge Access)** - Least-selling service, included to drive adoption.

This bundle strategically combines popular services with those that have lower purchase rates, aiming to drive uptake of less popular ancillaries while leveraging

👍

👎

📄

PROS AI

Analyzing your request...

Type a message...

PROS AI is still learning and may make mistakes. Chat history is stored until you log out.

PROS AI Beta

- LOUG (Lounge Access)**
 - Average Price: USD 39.42 (Feb), USD 39.21 (Mar)

Total for Offer Group 2: USD 46.7 (using March prices)

Offer Group 3: Comprehensive Travel Experience

- BBAG (Second Excess Bag)**
 - Average Price: USD 95 (consistent across months)
- LOUG (Lounge Access)**
 - Average Price: USD 39.42 (Feb), USD 39.21 (Mar)

Total for Offer Group 3: USD 134.42

Suggested Pricing Strategy

- Bundle Price: USD 250.53 before discount**
- Discount Option:** Apply a flat discount of USD 25, making the bundle price USD 225.53.

The pricing strategy offers a comprehensive travel experience at a competitive rate, encouraging customers to adopt less popular services like Lounge Access and the Second Excess Bag.

👍

👎

📄

PROS AI

Analyzing your request...

Type a message...

PROS AI is still learning and may make mistakes. Chat history is stored until you log out.

Suggest a bundle that includes a mix of both top-selling and least-selling ancillary products?

Yes, please suggest pricing options.

Yes, please create the bundle.

PROS

©2025 PROS, Inc. All rights reserved. Confidential and Proprietary.

twitter: @mich8elwu
linkedin.com/in/MichaelWuPhD

AI assisted: merchandising bundling agent

AI led: GSO task agent

PROS AI is still learning and can make mistakes. That history is important until we know

A futuristic cockpit scene featuring a white humanoid robot on the left and a human pilot on the right. The robot, with a sleek white body and a helmet-like head, is pointing its right hand towards the central instrument panel. The human pilot, seen from the side, wears a dark blue uniform and a large black headset. The cockpit is filled with various digital displays, including radar screens, gauges, and data readouts. The background shows a bright, overcast sky through the cockpit windows.

just as the autopilot, having agents
doesn't mean we don't need humans



QUANTUM SIMPLEX

Perplexity in Plain Words



connect w/ me: @mich8elwu
[linkedin.com/in/MichaelWuPhD](https://www.linkedin.com/in/MichaelWuPhD)

