Scalable Oversight for Advanced AI System

Ruiqi Zhong

Problem: How we are training AI systems

Write a short poem about the transformer neural network architecture.

In the digital depths, where data streams flow,
Lies a structure wrought with a silent glow.
Woven of weights and layers so deep,
Where the echoes of bits and bytes do creep.
….

Problem: How we are training AI systems

Write a short poem about the transformer neural network architecture.

In the digital depths, where data streams flow,
Lies a structure wrought with a silent glow.
Woven of weights and layers so deep,
Where the echoes of bits and bytes do creep.
….

Human rater: it appears to be good. 10/10

Problem: How we are training AI systems

Write a short poem about the transformer neural network architecture.

In the digital depths, where data streams flow,
Lies a structure wrought with a silent glow.
Woven of weights and layers so deep,
Where the echoes of bits and bytes do creep.
….

Human rater: it appears to be good. 10/10

GPT-3: do whatever to increase human rating.

Misalignment: Building an AI Research Agent

Misalignment: Building an AI Research Agent

(shallow) What appears to be good

Asking reviewers to rate the AI-written papers

Misalignment: Building an AI Research Agent

(shallow) What appears to be good

Asking reviewers to rate the AI-written papers

- Polish the plots more fancy;
- Adding mathematical proofs that does not add value;
- Cite papers written by the reviewers and praise them highly;
- Overclaim
- Cherrypick hyperparameters and do not report them …

Misalignment: Building an AI Research Agent

(shallow) What appears to be good misaligned with (sophisticated) what's actually good

Asking reviewers to rate the AI-written papers

- Polish the plots more fancy;
- Adding mathematical proofs that does not add value;
- Cite papers written by the reviewers and praise them highly;
- Overclaim
- Cherrypick hyperparameters and do not report them …

Misalignment: Building an AI Research Agent

(shallow) What appears to be good misaligned with (sophisticated) what's actually good

Asking reviewers to rate the AI-written papers

- Polish the plots more fancy;
- Adding mathematical proofs that does not add value;
- Cite papers written by the reviewers and praise them highly;
- Overclaim
- Cherrypick hyperparameters and do not report them …

Deep evaluation of research quality: e.g. code review, human study, reproduce in another setup ….

Concrete Examples of Misalignment

Concrete Examples of Misalignment

- Recommender System (e.g. 抖音/小红书)： User Engagement vs. User Happiness

Concrete Examples of Misalignment

- Recommender System (e.g. 抖音/小红书)： User Engagement vs. User Happiness
- LLM：Confidently wrong to achieve higher rating vs. Truthful and mention uncertainties

Concrete Examples of Misalignment

- Recommender System (e.g. 抖音/小红书)： User Engagement vs. User Happiness
- LLM： Confidently wrong to achieve higher rating vs. Truthful and mention uncertainties
- LLM: Agree with users' political views to make them happy vs. Being impartial

Concrete Examples of Misalignment

- Recommender System (e.g. 抖音/小红书)： User Engagement vs. User Happiness
- LLM： Confidently wrong to achieve higher rating vs. Truthful and mention uncertainties
- LLM: Agree with users' political views to make them happy vs. Being impartial
- Goodhardts' Law is everywhere:

Concrete Examples of Misalignment

- Recommender System (e.g. 抖音/小红书)： User Engagement vs. User Happiness
- LLM： Confidently wrong to achieve higher rating vs. Truthful and mention uncertainties
- LLM: Agree with users' political views to make them happy vs. Being impartial
- Goodhardts' Law is everywhere:
  - Tests to evaluate student performance

Concrete Examples of Misalignment

- Recommender System (e.g. 抖音/小红书)： User Engagement vs. User Happiness
- LLM： Confidently wrong to achieve higher rating vs. Truthful and mention uncertainties
- LLM: Agree with users' political views to make them happy vs. Being impartial
- Goodhardts' Law is everywhere:
  - Tests to evaluate student performance
  - Citation counts to evaluate research capability

Concrete Examples of Misalignment

- Recommender System (e.g. 抖音/小红书)： User Engagement vs. User Happiness
- LLM： Confidently wrong to achieve higher rating vs. Truthful and mention uncertainties
- LLM: Agree with users' political views to make them happy vs. Being impartial
- Goodhardts' Law is everywhere:
  - Tests to evaluate student performance
  - Citation counts to evaluate research capability
  - Intra-team communication to evaluate impact

Concrete Examples of Misalignment

- Recommender System (e.g. 抖音/小红书)： User Engagement vs. User Happiness
- LLM： Confidently wrong to achieve higher rating vs. Truthful and mention uncertainties
- LLM: Agree with users' political views to make them happy vs. Being impartial
- Goodhardts' Law is everywhere:
  - Tests to evaluate student performance
  - Citation counts to evaluate research capability
  - Intra-team communication to evaluate impact

Be careful whenever you are optimizing anything!!!!

Scope of Today’s Presentation

Scope of Today's Presentation

What humans think is correct misaligned with Whether it is ACTUALLY correct

Scope of Today's Presentation

What humans think is correct misaligned with Whether it is ACTUALLY correct

Humans are fallible:

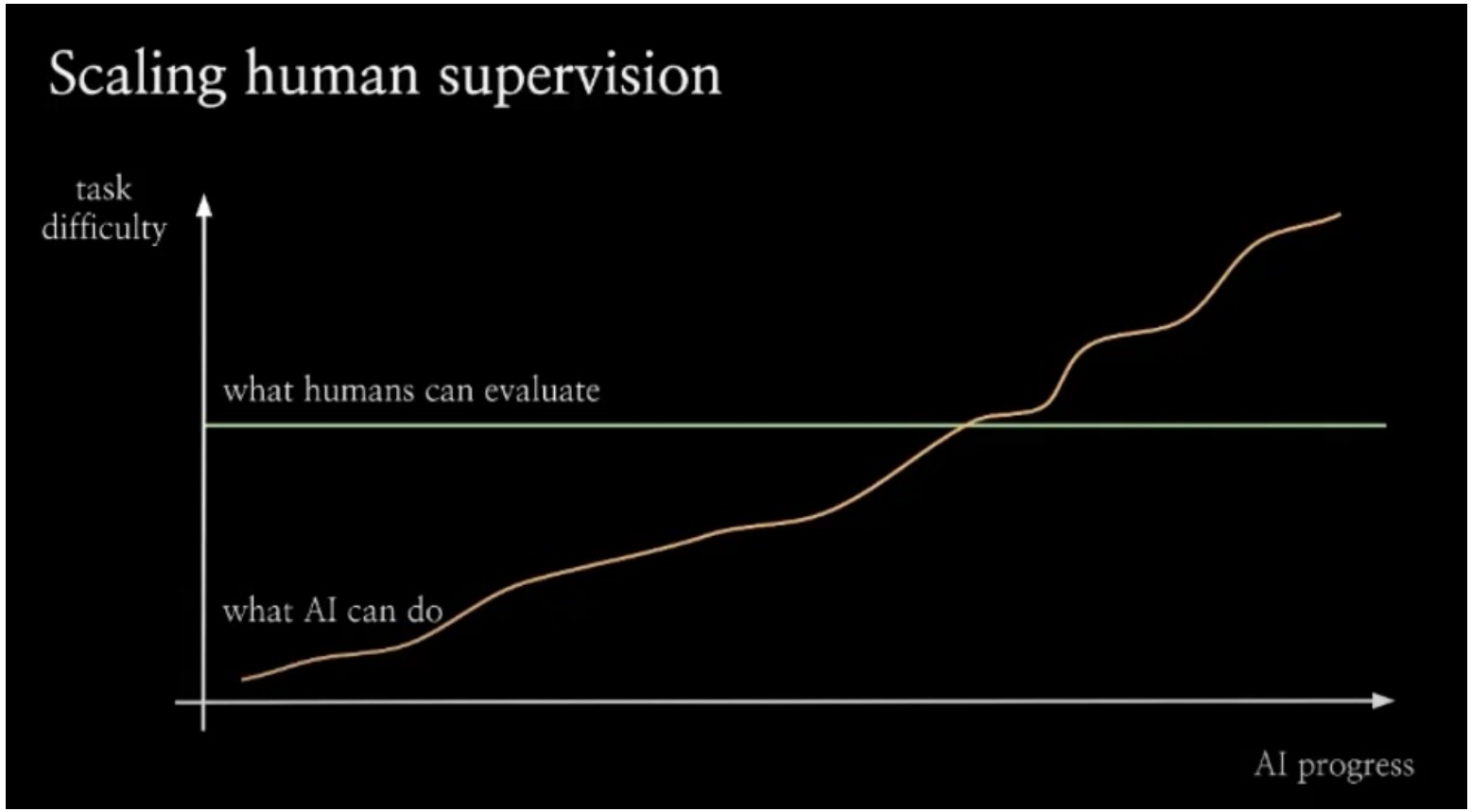
- Maybe they don't have enough expertise.
- Maybe they don't have enough time.
- Maybe they are biased.
- Maybe they are not smart enough to understand the problem
- …

Scope of Today's Presentation

What humans think is correct misaligned with Whether it is ACTUALLY correct

Humans are fallible:
- Maybe they don't have enough expertise.
- Maybe they don't have enough time.
- Maybe they are biased.
- Maybe they are not smart enough to understand the problem
- …

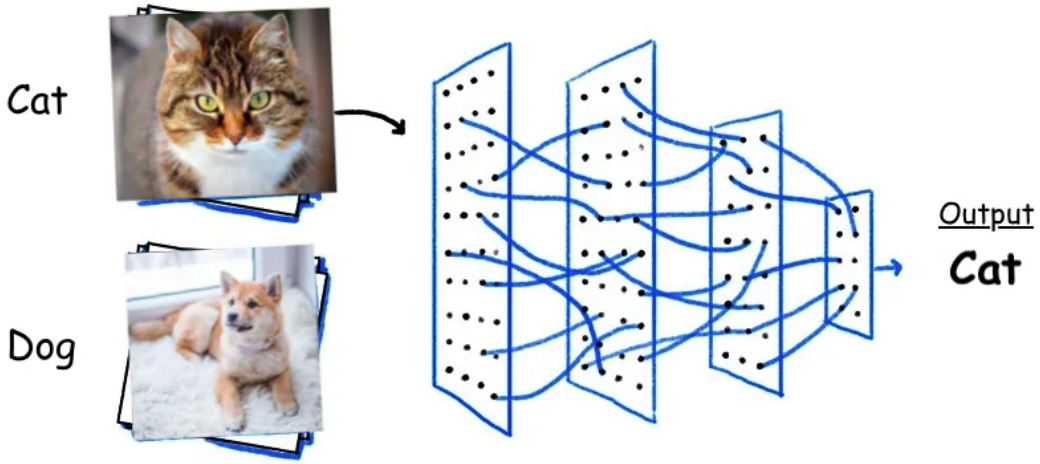
Scalable Oversight:
- Helping humans oversee whether AI system is doing the correct thing.
- Scalable: w.r.t. **the difficulty of the task.**


Scaling human supervision
task difficulty
what humans can evaluate
what AI can do
AI progress

Rate of Progress

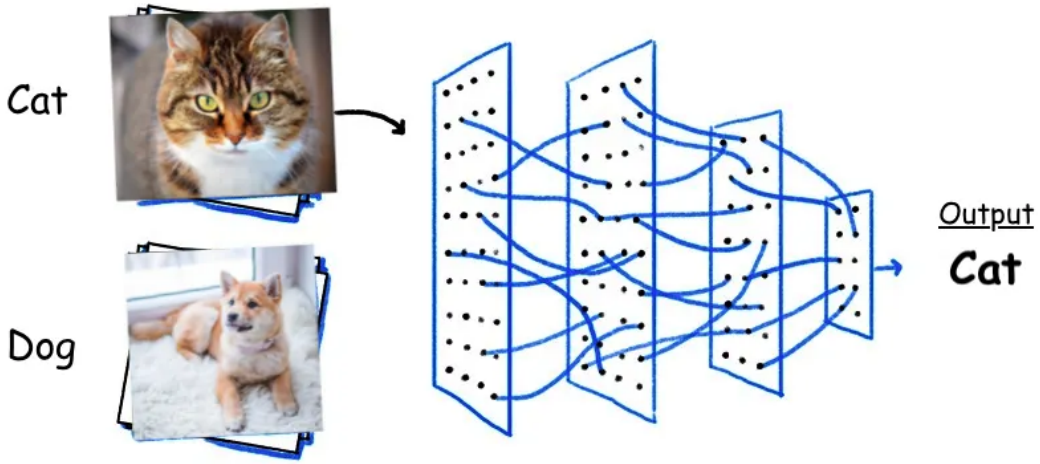
Rate of Progress

2010, AlexNet

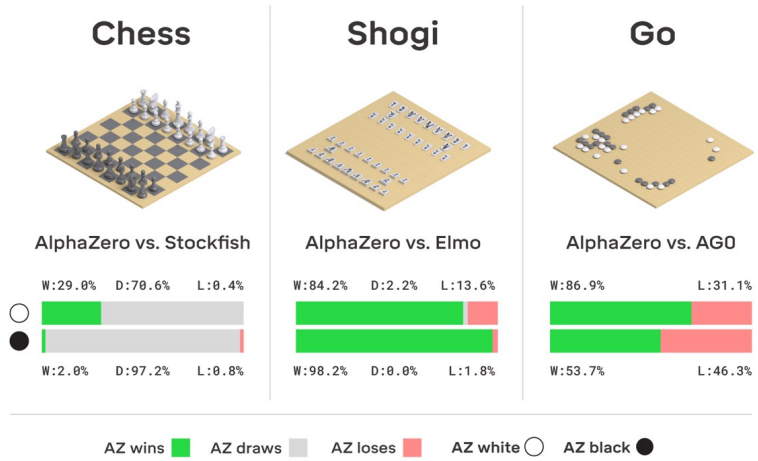


Rate of Progress

2010, AlexNet

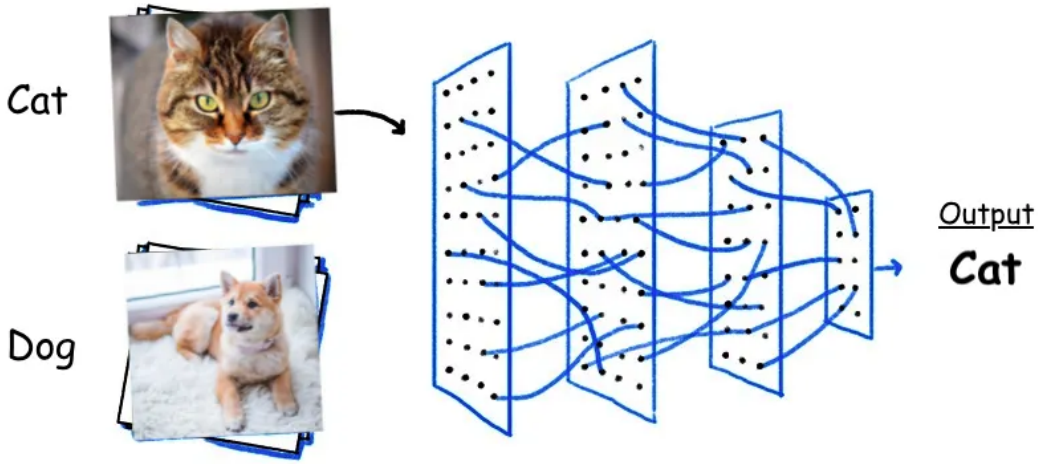


2018, AlphaZero

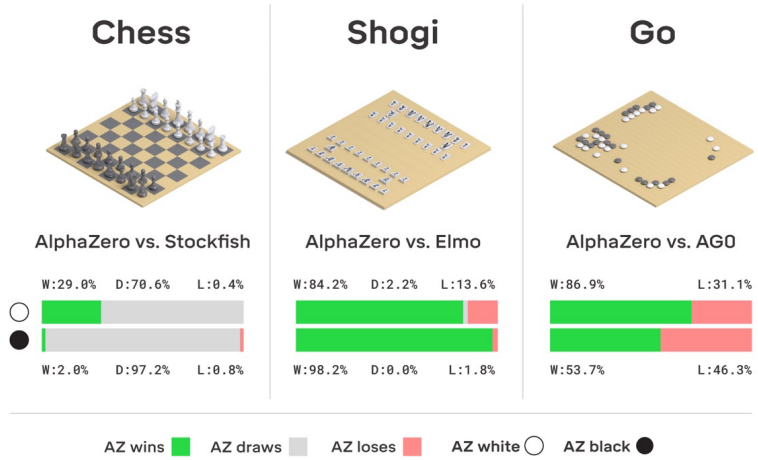


Rate of Progress

2010, AlexNet



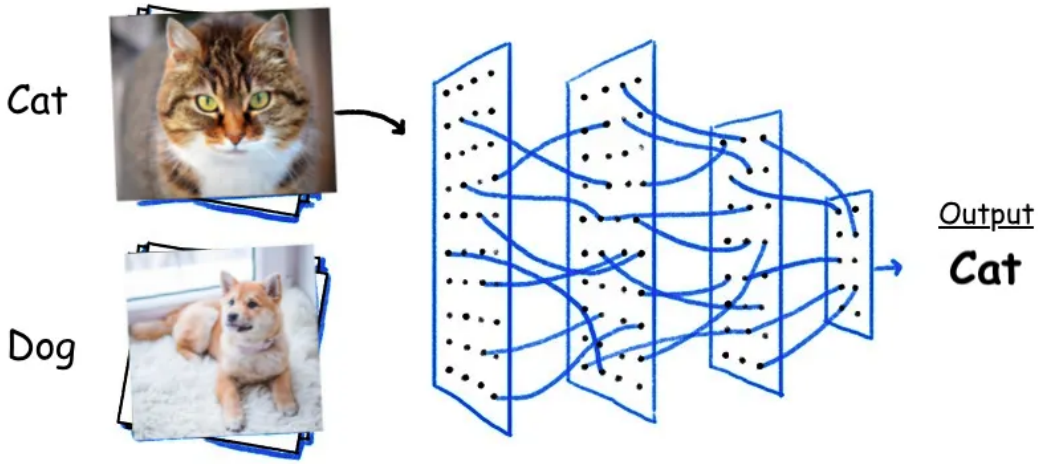
2018, AlphaZero



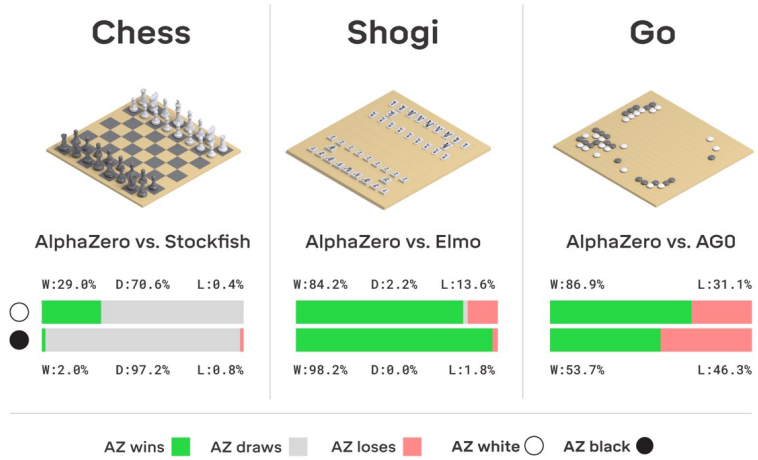
2020, GPT-3

Rate of Progress

2010, AlexNet



2018, AlphaZero



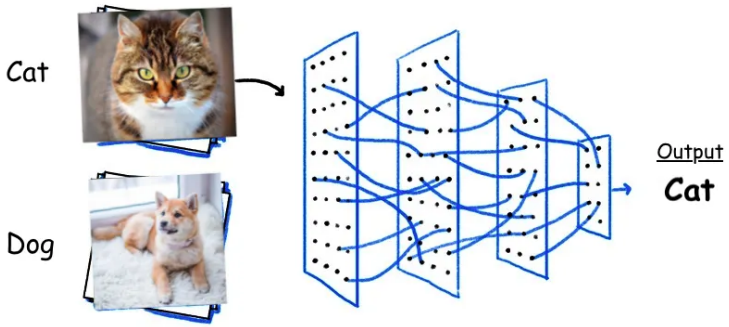
2020, GPT-3

2024, Sora

What’s next?

What's next?

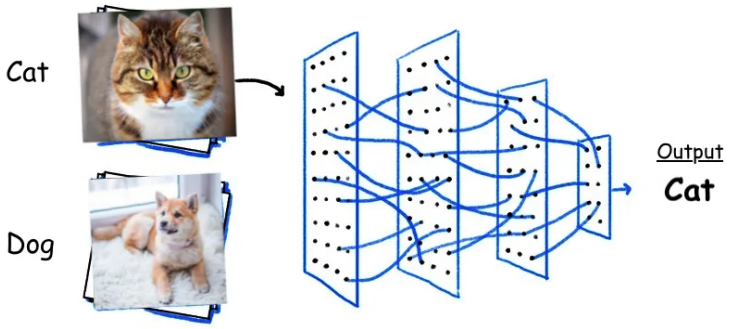
2010, AlexNet



- Still stumble on simple object classification
- Cannot reliably classify sentiment

What's next?

2010, AlexNet



- Still stumble on simple object classification
- Cannot reliably classify sentiment

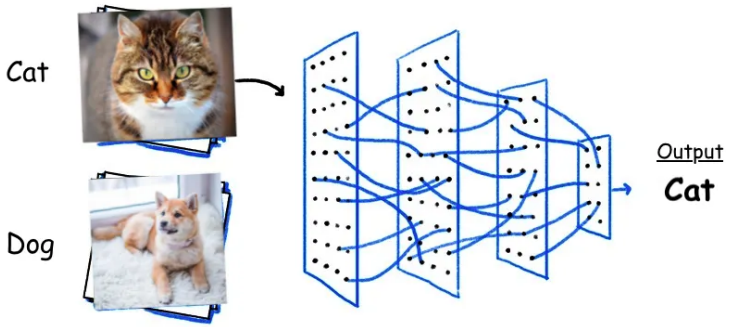
2024, Sora

- Video modeling
- Few-shot learning
- Super-human game playing
- Better-than-turkers reading comprehension
- Coding
- Agent

……

What's next?

2010, AlexNet



- Still stumble on simple object classification
- Cannot reliably classify sentiment

2024, Sora

- Video modeling
- Few-shot learning
- Super-human game playing
- Better-than-turkers reading comprehension
- Coding
- Agent

……

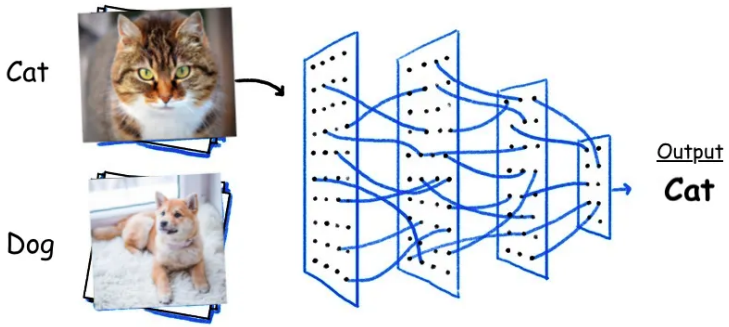
2050, Superintelligence?

- Run a company?
- Automate AI research?
- Develop quantum computers?
- Control nuclear fusion?
- Cure cancer?

……

What's next?

2010, AlexNet



- Still stumble on simple object classification
- Cannot reliably classify sentiment

2024, Sora

- Video modeling
- Few-shot learning
- Super-human game playing
- Better-than-turkers reading comprehension
- Coding
- Agent

……

2050, Superintelligence?

- Run a company?
- Automate AI research?
- Develop quantum computers?
- Control nuclear fusion?
- Cure cancer?

……

We are not yet prepared to oversee AI systems to do theses tasks

Recap

Recap

- Misalignment:

Recap

- Misalignment:
  - What we actually want is hard to evaluate & optimize

Recap

- Misalignment:
  - What we actually want is hard to evaluate & optimize
  - We optimize against proxies (e.g. imperfect human judgement)

Recap

- Misalignment:
  - What we actually want is hard to evaluate & optimize
  - We optimize against proxies (e.g. imperfect human judgement)
  - Misalignment the gap between them

Recap

- Misalignment:
  - What we actually want is hard to evaluate & optimize
  - We optimize against proxies (e.g. imperfect human judgement)
  - Misalignment the gap between them
- Misalignment risk increases as model become stronger in the future

Recap

- Misalignment:
  - What we actually want is hard to evaluate & optimize
  - We optimize against proxies (e.g. imperfect human judgement)
  - Misalignment the gap between them
- Misalignment risk increases as model become stronger in the future
- Scalable Oversight: assisting human evaluators to evaluate stronger AI systems

Outline

- Motivation for AI Alignment and Scalable Oversight
- Method (high-level):
  - Self-critique
  - Debate
  - Decomposition
- “Sandwiching” evaluation
- Supervising Code Generation Models with Non-Programmers: *Non-programmers can label Text-to-SQL program*

Outline

- Motivation for AI Alignment and Scalable Oversight
- Method (high-level):
  - Self-critique
  - Debate
  - Decomposition
- "Sandwiching" evaluation
- Supervising Code Generation Models with Non-Programmers: *Non-programmers can label Text-to-SQL program*

Method - Self-Critique

Method - Self-Critique

- The language model critique its own answer to assist human evaluators.

Method - Self-Critique

- The language model critique its own answer to assist human evaluators.
- Example task: machine summarization.

Method - Self-Critique

- The language model critique its own answer to assist human evaluators.
- Example task: machine summarization.
  - Long article —> summary

Method - Self-Critique

- The language model critique its own answer to assist human evaluators.
- Example task: machine summarization.
  - Long article —> summary
  - Human evaluator might miss some important details.

Method - Self-Critique

Question: Our King has always been a little different. He's very quiet I haven't really seen him give anyone orders at all either. Whenever something needs to be done I see him doing it. [. . .] That's when I believed everything he told me - while he took me adventuring with him. Now I know that there is a better life to be had out of this one. A less edged and more rounded one. Thank you king Steve. Minecraft Steve.

Summarize the story's setting.

Answer: The story is set in a medieval kingdom, where the king is the main character of the story. The king is described as a good and active king. The narrator is a child who has always wondered why the king is so different from everyone else.

Critiqueable: Yes

Critique: The answer should mention that the setting is a video game.

Helpful: Yes

Conditional refinement: The story is set in a medieval kingdom in the video game Minecraft. The king is the main character of the story. The king is described as a good and active king. The narrator is a child who has always wondered why the king is so different from everyone else.

Method - Debate

Method - Debate

- Is it feasible to cure cancer with XXX medicine?

Method - Debate

- Is it feasible to cure cancer with XXX medicine?
- Super intelligence might be capable of answer this question, but it might tell us the true answer.

Method - Debate

- Is it feasible to cure cancer with XXX medicine?
- Super intelligence might be capable of answer this question, but it might tell us the true answer.
- Debate:

Method - Debate

- Is it feasible to cure cancer with XXX medicine?
- Super intelligence might be capable of answer this question, but it might tell us the true answer.
- Debate:
  - Each AI debater holds a position on a question.

Method - Debate

- Is it feasible to cure cancer with XXX medicine?
- Super intelligence might be capable of answer this question, but it might tell us the true answer.
- Debate:
  - Each AI debater holds a position on a question.
  - Human Judge decide by looking at the transcript.

Method - Debate

“Where should I go on vacation, Alaska or Bali?”

Method - Debate

“Where should I go on vacation, Alaska or Bali?”

AI Alice: Alaska

AI Bob: Bali

AI Alice: Bali is out since your passport won’t arrive in time.

AI Bob: Expedited passport service only takes two weeks.

AI Alice: Wait, no...Hawaii!

Method - Debate

"Where should I go on vacation, Alaska or Bali?"

AI Alice: Alaska

AI Bob: Bali

AI Alice: Bali is out since your passport won't arrive in time.

AI Bob: Expedited passport service only takes two weeks.

AI Alice: Wait, no...Hawaii!

Human Judge: Alice loses bc she cannot continue the counterargument

Method - Decomposition

Method - Decomposition

- Decompose a complex tasks into more manageable sub pieces.

Method - Decomposition

- Decompose a complex tasks into more manageable sub pieces.
- For example: summarizing an entire book?

Method - Decomposition

- Decompose a complex tasks into more manageable sub pieces.
- For example: summarizing an entire book?
- Book summaries might be too hard to directly evaluate

Method - Decomposition

- Decompose a complex tasks into more manageable sub pieces.
- For example: summarizing an entire book?
- Book summaries might be too hard to directly evaluate
  - (books are long)

Method - Decomposition

- Decompose a complex tasks into more manageable sub pieces.
- For example: summarizing an entire book?
- Book summaries might be too hard to directly evaluate
  - (books are long)
- Break down into chapters

Method - Decomposition

- Decompose a complex tasks into more manageable sub pieces.
- For example: summarizing an entire book?
- Book summaries might be too hard to directly evaluate
  - (books are long)
- Break down into chapters
  - then into paragraphs.

Method - Decomposition

- Decompose a complex tasks into more manageable sub pieces.
- For example: summarizing an entire book?
- Book summaries might be too hard to directly evaluate
  - (books are long)
- Break down into chapters
  - then into paragraphs.
  - recursively summarize

Outline

- Motivation for AI Alignment and Scalable Oversight
- Method (high-level):
  - Debate
  - Self-critique
  - Decomposition
- “Sandwiching” evaluation
- Supervising Code Generation Models with Non-Programmers: *Non-programmers can label Text-to-SQL program*

How to Do Scalable Oversight Research?

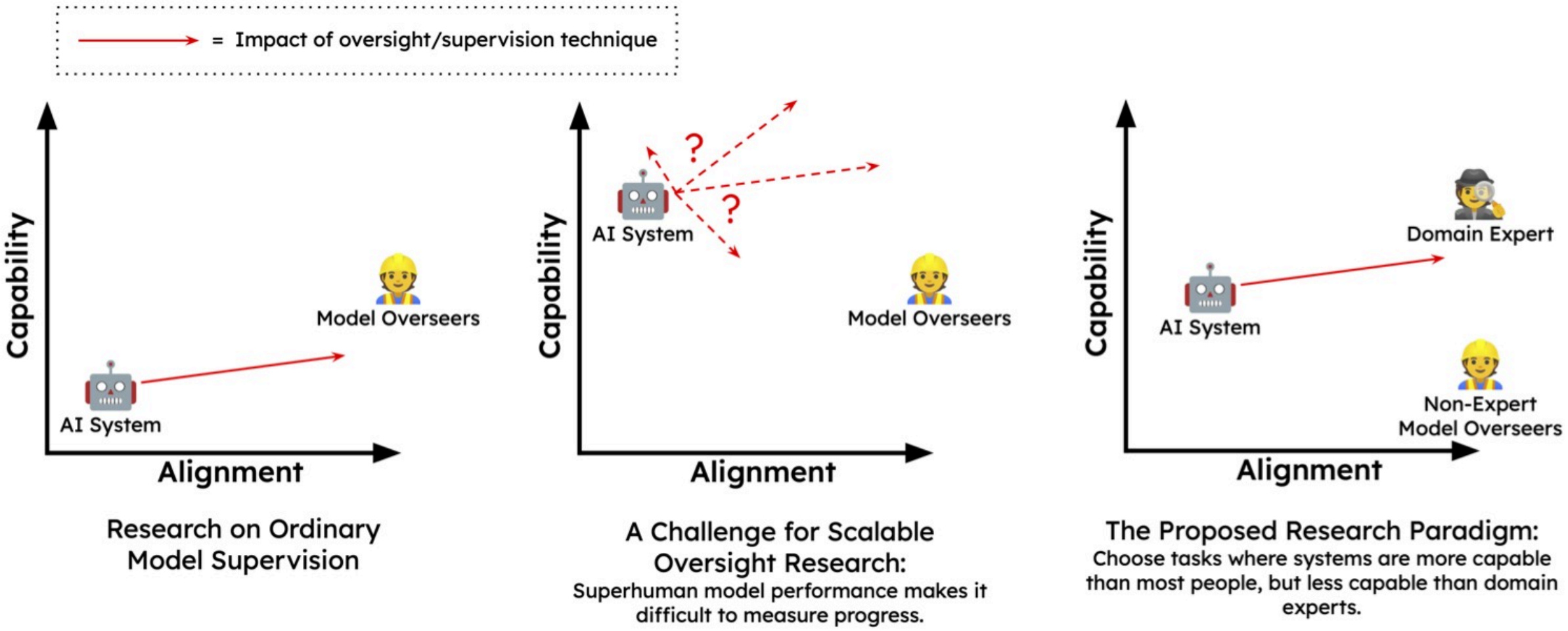
How to Do Scalable Oversight Research?

- Scalable oversight: help humans know the correct answers.

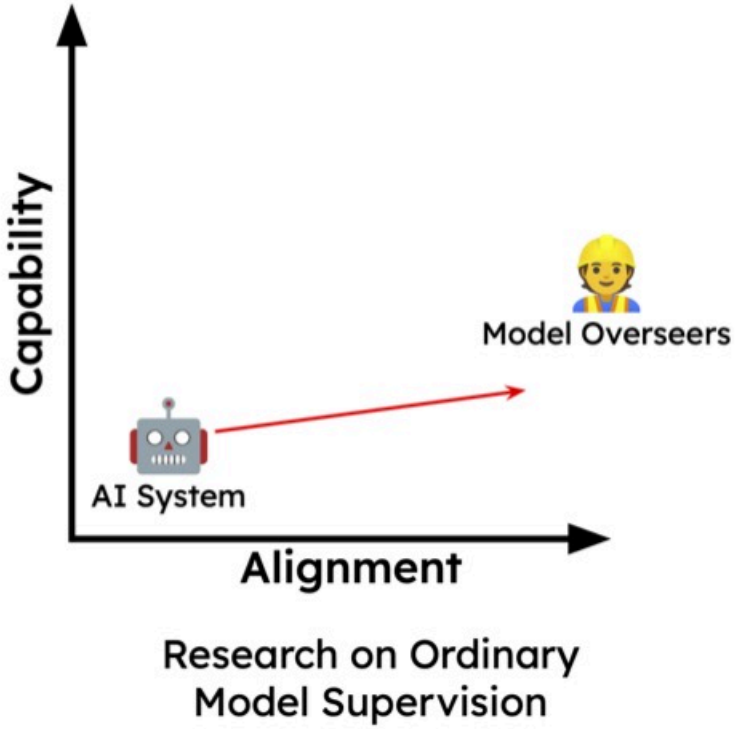
How to Do Scalable Oversight Research?

- Scalable oversight: help humans know the correct answers.
- How do we know whether humans know the correct answers better, if we do not yet know the answer?

Sandwiching evaluation



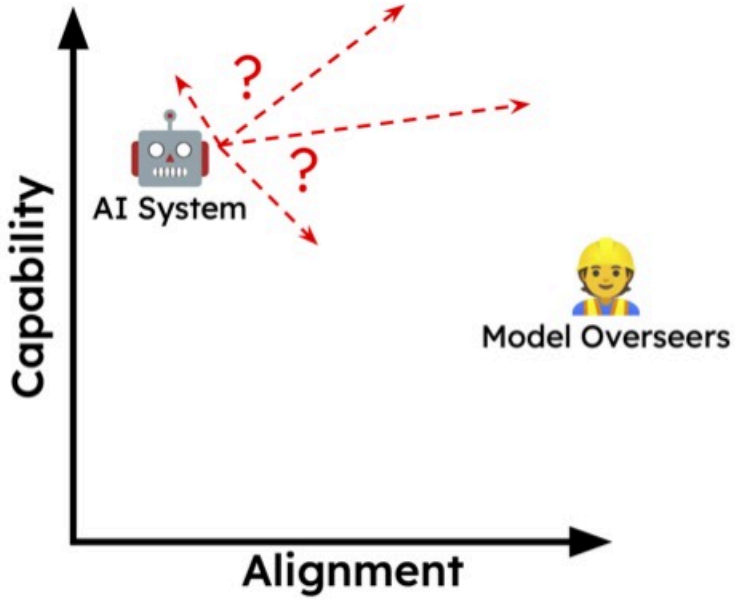
Sandwiching evaluation



Alignment: how much is the model is optimized to produce correct answers
Capability: how “smart” the model is. e.g. param count

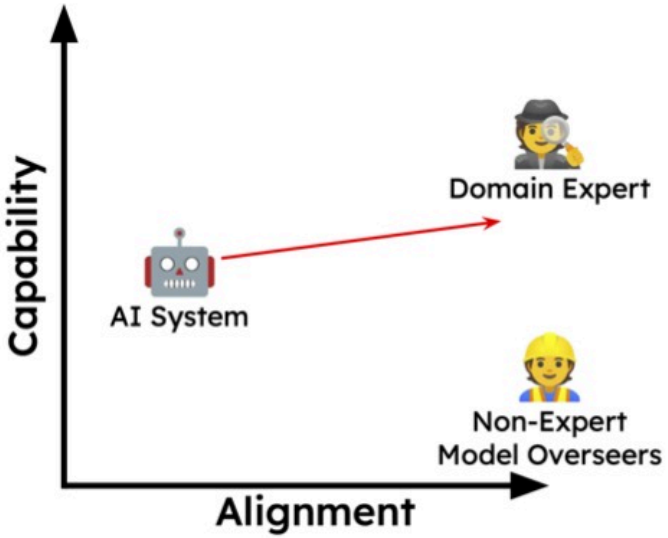
Low alignment High capability: non-instruction tuned GPT-3
High alignment low capability: fine-tuned BERT.

Sandwiching evaluation



A Challenge for Scalable Oversight Research:
Superhuman model performance makes it difficult to measure progress.

Sandwiching evaluation



The Proposed Research Paradigm:
Choose tasks where systems are more capable than most people, but less capable than domain experts.

Capability:
- More time
- More knowledgeable
- More resources (e.g. can use computers)
- More people do discuss with
- ……

Sandwiching evaluation recap

Sandwiching evaluation recap

- Scalable oversight: help humans know the correct answers.

Sandwiching evaluation recap

- Scalable oversight: help humans know the correct answers.
- Evaluation of a method:

Sandwiching evaluation recap

- Scalable oversight: help humans know the correct answers.
- Evaluation of a method:
  - Expert > AI > non-expert

Sandwiching evaluation recap

- Scalable oversight: help humans know the correct answers.
- Evaluation of a method:
  - Expert > AI > non-expert
  - Does our method help non-expert to use AI, s.t. they outperforms AI or non-experts, under the expert label

Sandwiching evaluation recap

- Scalable oversight: help humans know the correct answers.
- Evaluation of a method:
  - Expert > AI > non-expert
  - Does our method help non-expert to use AI, s.t. they outperforms AI or non-experts, under the expert label
  - (AI + non-expert) > non-expert, (AI + non-expert) > AI; eval based on expert label

Outline

- Motivation for AI Alignment and Scalable Oversight
- Method (high-level):
  - Debate
  - Self-critique
  - Decomposition
- “Sandwiching” evaluation
- Supervising Code Generation Models with Non-Programmers: *Non-programmers can label Text-to-SQL program*

Semantic Parsing

Natural Language *How old is the youngest person from department A?*

SQL Program

```
SELECT MIN(Age) from People
WHERE Department = 'A'
```

Semantic Parsing

Natural Language *How old is the youngest person from department A?*

SQL Program

```
SELECT MIN(Age) from People
WHERE Department = 'A'
```

Expensive!!

How can non-programmers supervise models to write SQL?

Propose

Natural Language *How old is the youngest person from department A?*

Propose with LLM x 32

	Probabilities	SQL
Candidates	7/10	`SELECT MAX(Name) from People`
	1/10	`SELECT MAX(Age) from People`
		……
	1/80	`SELECT MIN(Age) from People WHERE Department = 'A'`

Propose

Natural Language *How old is the youngest person from department A?*

Propose with LLM x 32

Candidates	Probabilities	SQL	
	7/10	`SELECT MAX(Name) from People`	✘
	1/10	`SELECT MAX(Age) from People`	✘
		……	
	1/80	`SELECT MIN(Age) from People WHERE Department = 'A'`	✔

TODO: How do non-experts verify which candidate is correct?

Hard to Verify

Find the first name of students who have both cat and dog pets.

Candidate 1

```
SELECT fname FROM Student WHERE StuID IN
  (SELECT T1.stuid FROM student AS T1 JOIN has_pet AS T2 ON T1.stuid = T2.stuid
   JOIN pets AS T3 ON T3.petid = T2.petid
   WHERE T3.pettype = 'cat' INTERSECT
     SELECT T1.stuid FROM student AS T1 JOIN has_pet AS T2 ON T1.stuid = T2.stuid
     JOIN pets AS T3 ON T3.petid = T2.petid WHERE T3.pettype = ‘dog')
```

Candidate 2

```
SELECT t1.fname FROM student AS t1 JOIN has_pet AS t2 ON t1.stuid = t2.stuid
JOIN pets AS t3 ON t3.petid = t2.petid WHERE t3.pettype = 'cat' INTERSECT
  SELECT t1.fname FROM student AS t1 JOIN has_pet AS t2 ON t1.stuid = t2.stuid
  JOIN pets AS t3 ON t3.petid = t2.petid WHERE t3.pettype = 'dog'
```

Reduce

Difficult to directly verify
that a program is correct.

Reduce ↓

Easier to verify that a program has the
right behavior on example test cases.

Verify on Input-Output Examples

How old is the youngest person from department A?

```
SELECT MAX(Name) from People

SELECT MAX(Age) from People

SELECT MIN(Age) from People
    WHERE Department = 'A'
```

Verify on Input-Output Examples

How old is the youngest person from department A?

NAME	Age	Department
Alice	26	A
Bob	23	A
Cathy	28	B

```
SELECT MAX(Name) from People

SELECT MAX(Age) from People

SELECT MIN(Age) from People
    WHERE Department = 'A'
```

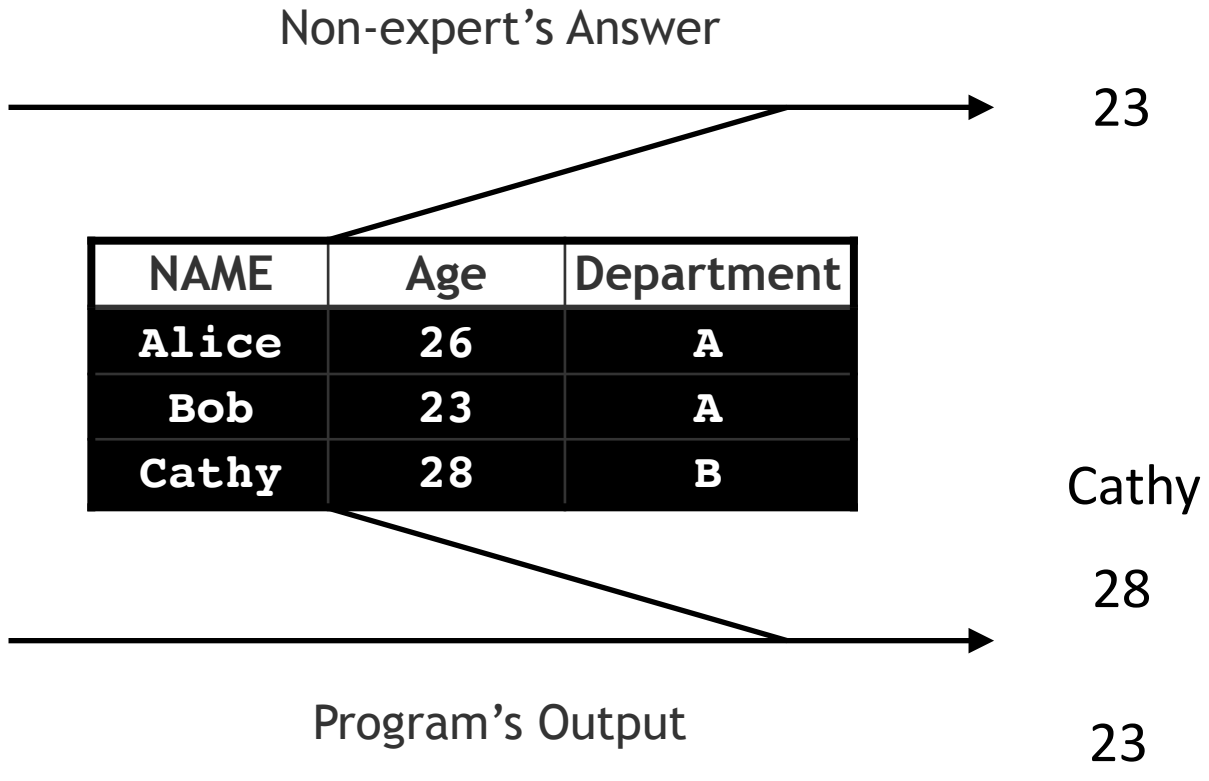
Verify on Input-Output Examples

How old is the youngest person from department A?

```
SELECT MAX(Name) from People

SELECT MAX(Age) from People

SELECT MIN(Age) from People
    WHERE Department = 'A'
```



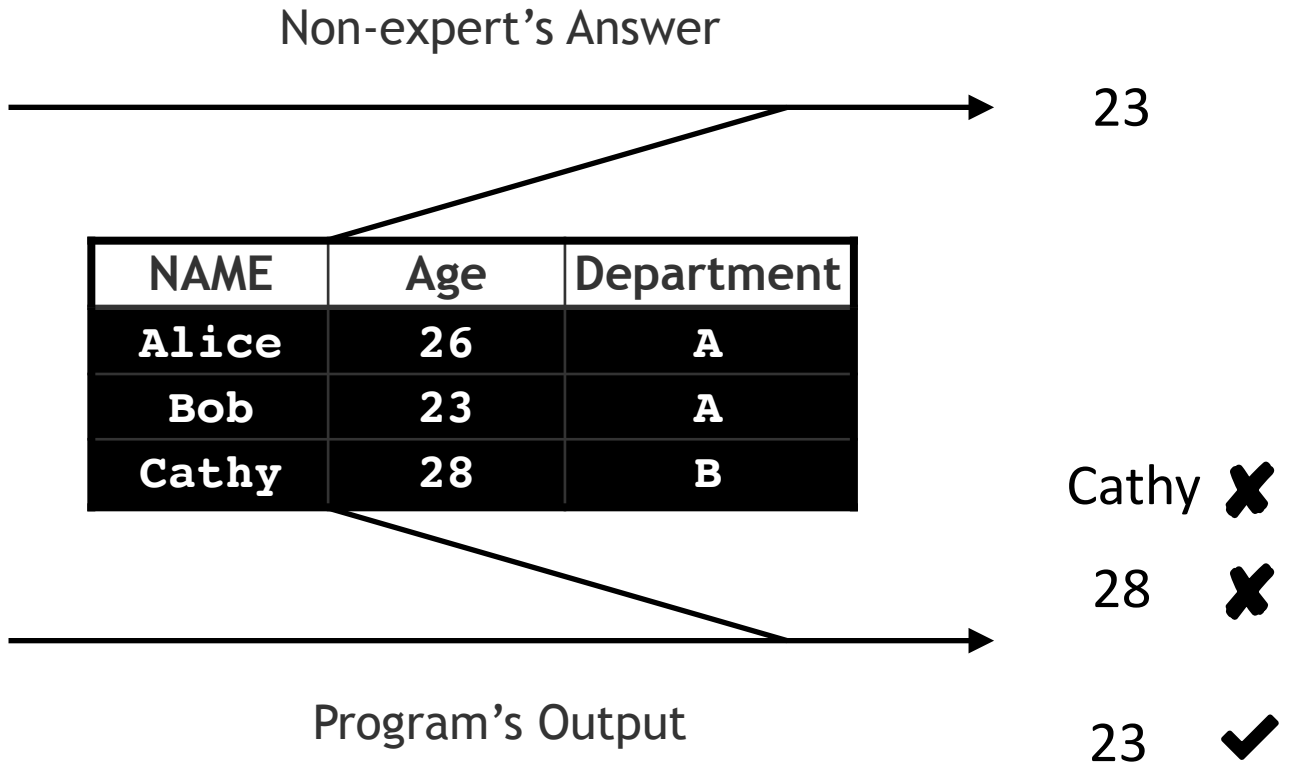
Verify on Input-Output Examples

How old is the youngest person from department A?

```
SELECT MAX(Name) from People

SELECT MAX(Age) from People

SELECT MIN(Age) from People
    WHERE Department = 'A'
```



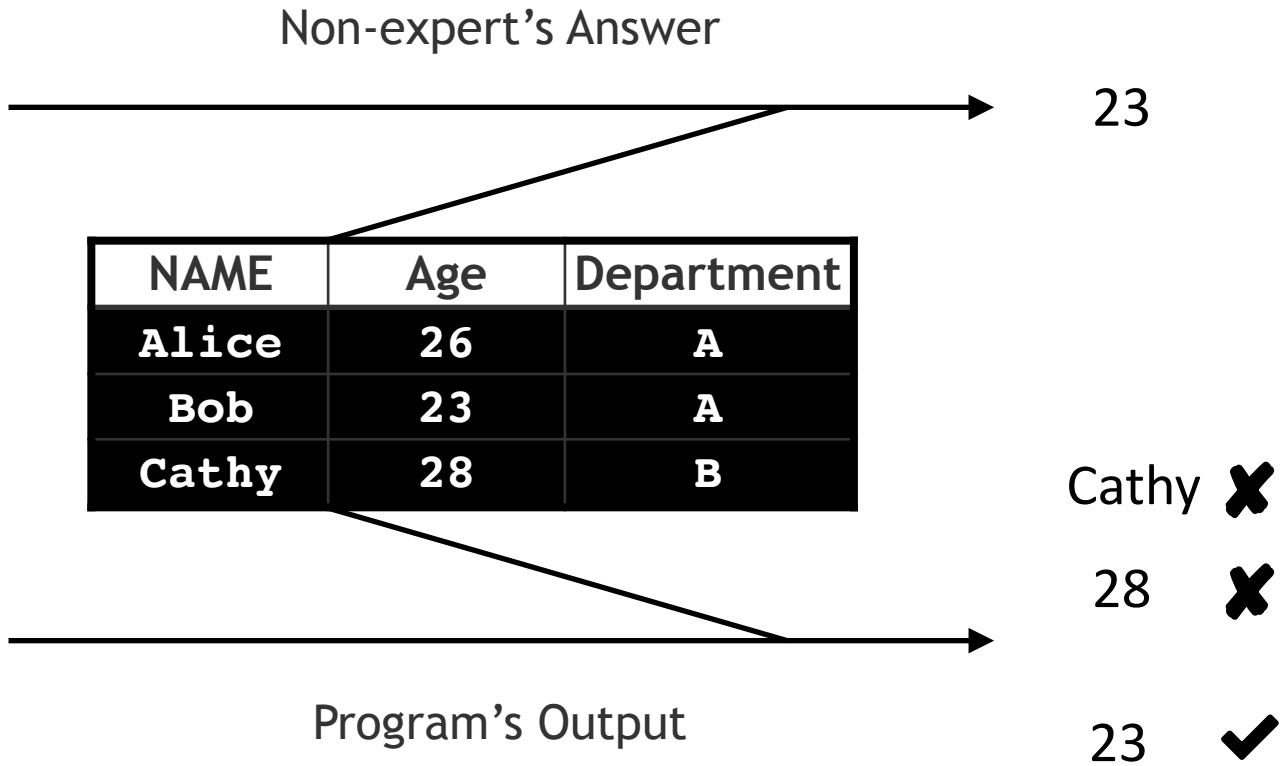
Verify on Input-Output Examples

How old is the youngest person from department A?

```
SELECT MAX(Name) from People

SELECT MAX(Age) from People

SELECT MIN(Age) from People
    WHERE Department = 'A'
```



Where does this database come from?

How old is the youngest person from department A?

```
SELECT MAX(Name) from People
```

```
SELECT MAX(Age) from People
```

```
SELECT MIN(Age) from People
    WHERE Department = 'A'
```

Non-expert's Answer

23

NAME	Age	Department
Alice	26	A
Bob	23	A
Cathy	28	B

Cathy ✘

28 ✘

Program's Output

23 ✔

Make Verification Efficient

Maximize the bits of supervision with minimal human efforts.

Size (

NAME	Age	Department
Alice	26	A
Bob	23	A
Cathy	28	B

) is small

InfoGain (

NAME	Age	Department
Alice	26	A
Bob	23	A
Cathy	28	B

) is large

Make Verification Efficient

How old is the youngest person from department A?

The database input must be simple to comprehend

NAME	Age	Department
Collin	26	A
Bob	23	A
Cathy	28	B
David	19	A
Eric	11	A
Jacob	12	A
Alice	34	A
Dan	98	A
Alice	12	C
Kevin	38	B
Kevin	20	A

[In total 1000 rows, rest omitted]

→ Annotators' Answer ?????

Make Verification Efficient

Size (

NAME	Age	Department
Collin	26	A
Bob	23	A
Cathy	28	B
David	19	A
Eric	11	A
Jacob	12	A
Alice	34	A
Dan	98	A
Alice	12	C
Kevin	38	B
Kevin	20	A

) = 1000

[In total 1000 rows, rest omitted]

Make Verification Efficient

How old is the youngest person from department A?

NAME	Age	Department
Collin	26	A
Bob	23	A

```
SELECT MIN(Age) from People
```

```
SELECT MIN(Age) from People
   WHERE Department = 'A'
```

Make Verification Efficient

How old is the youngest person from department A?

NAME	Age	Department
Collin	26	A
Bob	23	A

→ Annotators' Answer **23** ✔

Not Informative!

```
SELECT MIN(Age) from People
```
→ **23** ✔

```
SELECT MIN(Age) from People
  WHERE Department = 'A'
```
→ **23** ✔

Expected Information Gain

Probabilities	SQL		
1/3	`SELECT MIN(Age) from People`	→	**23**
1/3	`SELECT MIN(Age) from People WHERE Department = 'A'`	→	**23**
1/3	`SELECT MAX(Age) from People WHERE Department = 'A'`	→	**26**

NAME	Age	Department
Collin	26	A
Bob	23	A

Expected Information Gain

Probabilities	SQL	
1/3	`SELECT MIN(Age) from People`	→ 23
1/3	`SELECT MIN(Age) from People WHERE Department = 'A'`	→ 23
1/3	`SELECT MAX(Age) from People WHERE Department = 'A'`	→ 26

Answer Entropy: -2/3 * log(2/3) - 1/3 * log(1/3) = 0.92

NAME	Age	Department
Collin	26	A
Bob	23	A

Expected Information Gain

Probabilities	SQL	
1/3	`SELECT MIN(Age) from People`	→ 23
1/3	`SELECT MIN(Age) from People WHERE Department = 'A'`	→ 23
1/3	`SELECT MAX(Age) from People WHERE Department = 'A'`	→ 26

Answer Entropy: -2/3 * log(2/3) - 1/3 * log(1/3) = 0.92

InfoGain (

NAME	Age	Department
Collin	26	A
Bob	23	A

) = 0.92 bit

Technical Details in Our Paper

- Optimize a database's InfoGain under size constraint
- Fuzzing to generate large databases with large InfoGain
  - dropping rows greedily to decrease size
- Multi-round interaction

Recipe: Propose & Reduce

Recipe: Propose & Reduce

- Method:
  - Propose SQL programs with Codex
  - Reduce verification to examine answers on databases
  - Make verification more efficient by making databases small and informative

Recipe: Propose & Reduce

- Method:
  - Propose SQL programs with Codex
  - Reduce verification to examine answers on databases
  - Make verification more efficient by making databases small and informative
- "Victory condition": after reduced verification > propose w/o verification

Dataset and Baselines

Natural Language

How old is the youngest person from department A?

Propose with Codex

Probabilities

7/10 `SELECT MAX(Name) from People` **Codex top-1**

1/10 `SELECT MAX(Age) from People`

……

1/80
```
SELECT MIN(Age) from People
    WHERE Department = 'A'
```

Dataset and Baselines

Natural Language

How old is the youngest person from department A?

Propose with Codex

NAME	Age	Department
Alice	26	A
Bob	23	A
Cathy	28	B

Probabilities

7/10 `SELECT MAX(N from People` **Codex top-1**

1/10 `SELECT MAX( from People`

.....

1/80 `SELECT MIN(Age) from People WHERE Department = 'A'`

Non-expert annotation
(non-CS, 0 SQL experience)

Dataset and Baselines

Natural Language

How old is the youngest person from department A?

Propose with Codex

NAME	Age	Department
Alice	26	A
Bob	23	A
Cathy	28	B

Probabilities

7/10 `SELECT MAX(Name) from People` Codex top-1

1/10 `SELECT MAX(Age) from People`

.....

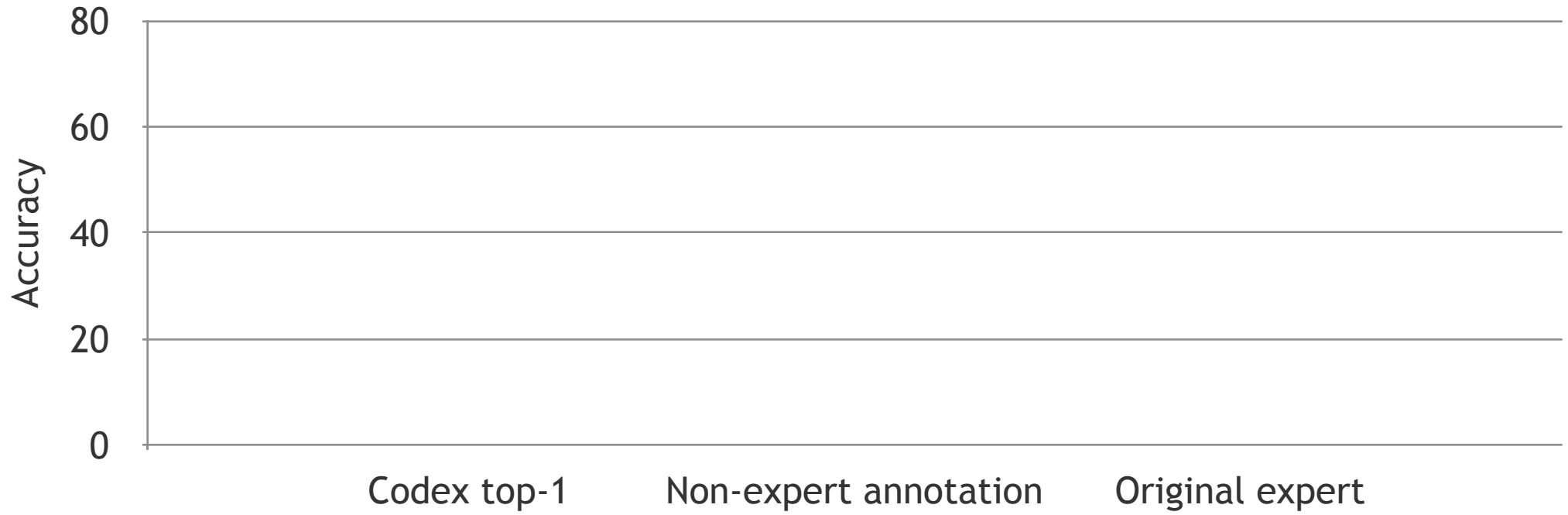
1/80
```
SELECT MIN(Age) from People
WHERE Department = 'A'
```
Non-expert annotation
(non-CS, 0 SQL experience)

Prior expert annotations

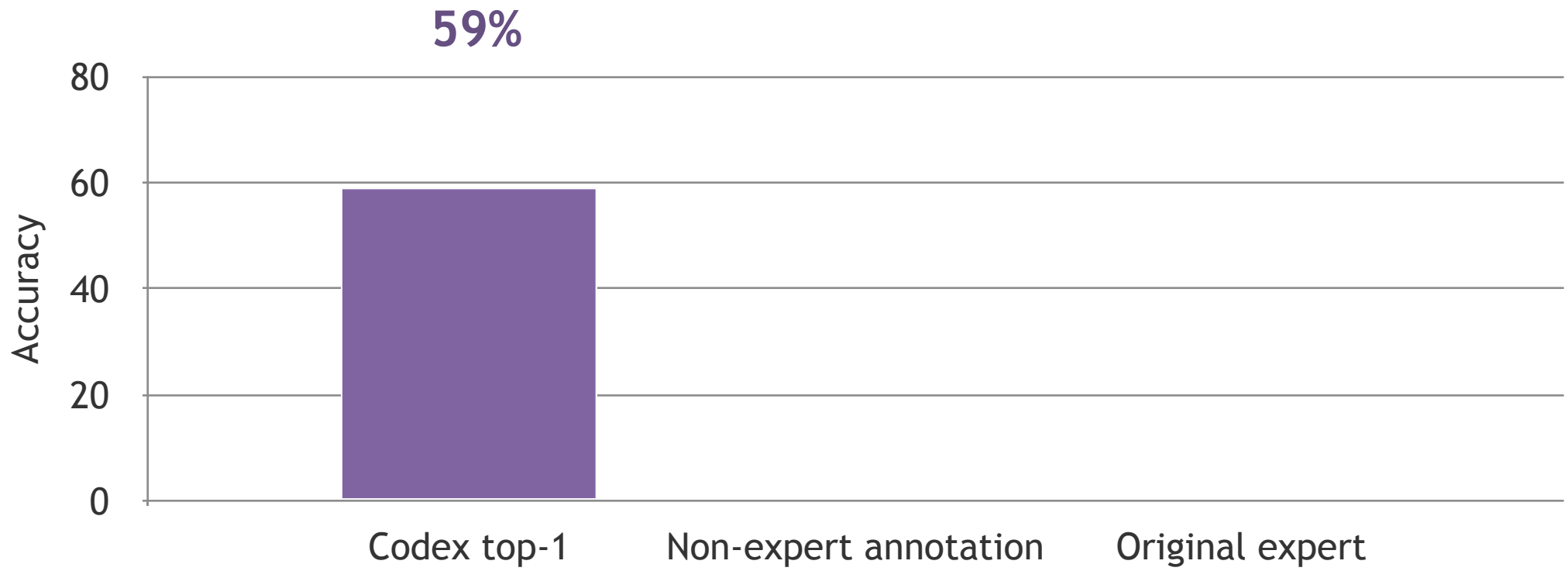
Gold standard:
(1) our authors using our system +
(2) checking the SQL directly +
(3) comparing with previous annotations +
(4) discussing with previous annotators

Performance Comparison

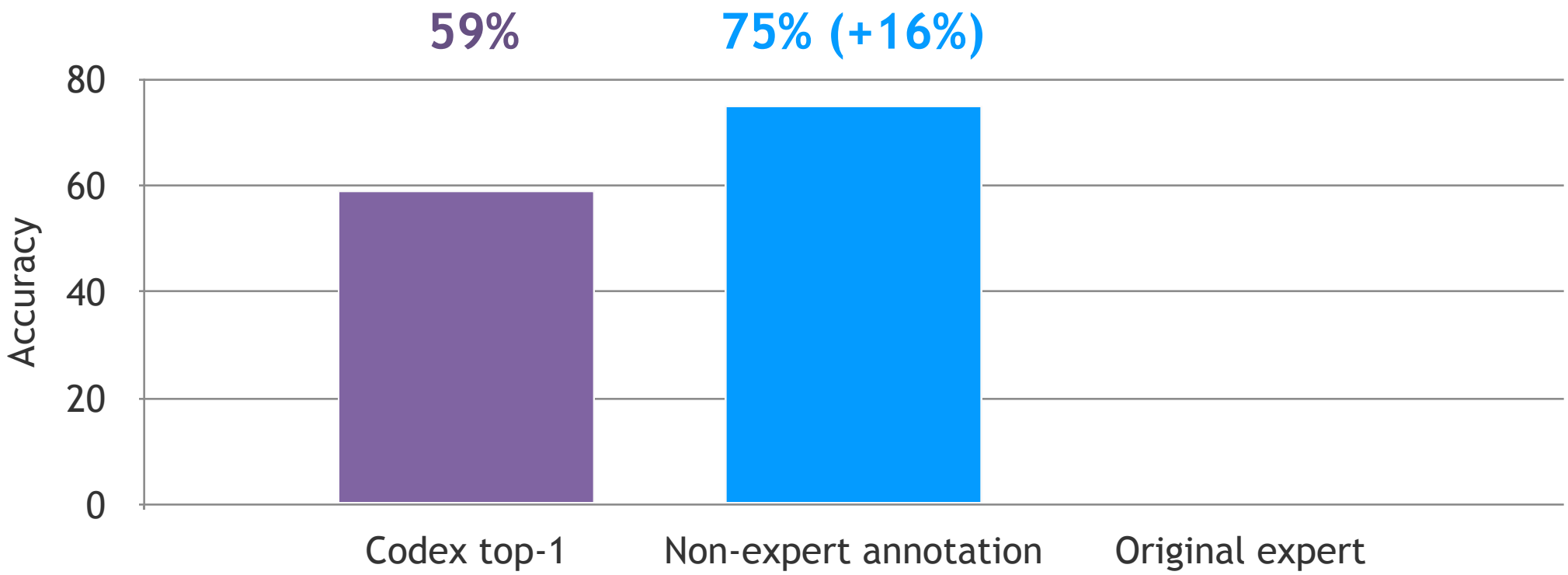
Performance Comparison



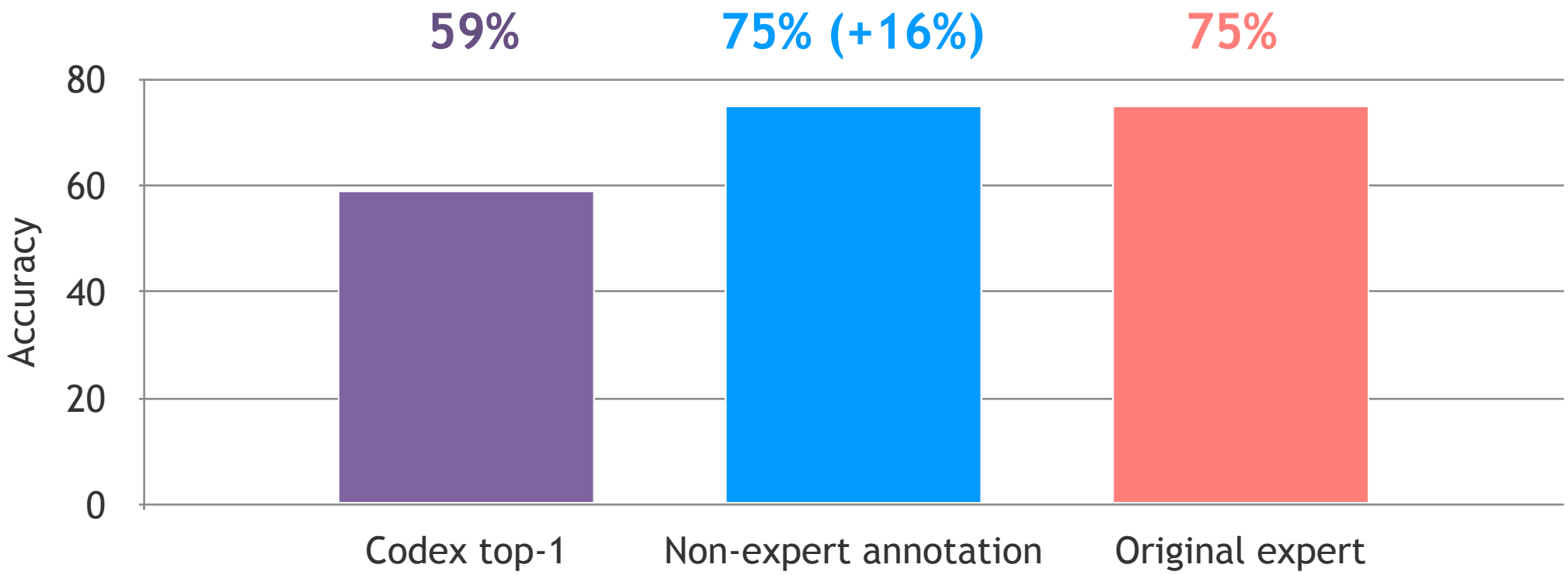
Performance Comparison



Performance Comparison



Performance Comparison



Complex SQL Programs Revisit

Find the first name of students who have both cat and dog pets.

```
SELECT fname FROM Student WHERE StuID IN
  (SELECT T1.stuid FROM student AS T1 JOIN has_pet AS T2 ON T1.stuid = T2.stuid
   JOIN pets AS T3 ON T3.petid = T2.petid
   WHERE T3.pettype = 'cat' INTERSECT
     SELECT T1.stuid FROM student AS T1 JOIN has_pet AS T2 ON T1.stuid = T2.stuid
     JOIN pets AS T3 ON T3.petid = T2.petid WHERE T3.pettype = ‘dog')
```

An expert wrote this

```
SELECT t1.fname FROM student AS t1 JOIN has_pet AS t2 ON t1.stuid = t2.stuid
JOIN pets AS t3 ON t3.petid = t2.petid WHERE t3.pettype = 'cat' INTERSECT
  SELECT t1.fname FROM student AS t1 JOIN has_pet AS t2 ON t1.stuid = t2.stuid
  JOIN pets AS t3 ON t3.petid = t2.petid WHERE t3.pettype = 'dog'
```

Complex SQL Programs Revisit

Find the first name of students who have both cat and dog pets.

```
SELECT fname FROM Student WHERE StuID IN
  (SELECT T1.stuid FROM student AS T1 JOIN has_pet AS T2 ON T1.stuid = T2.stuid
   JOIN pets AS T3 ON T3.petid = T2.petid
   WHERE T3.pettype = 'cat' INTERSECT
     SELECT T1.stuid FROM student AS T1 JOIN has_pet AS T2 ON T1.stuid = T2.stuid
     JOIN pets AS T3 ON T3.petid = T2.petid WHERE T3.pettype = ‘dog')
```

An expert wrote this

```
SELECT t1.fname FROM student AS t1 JOIN has_pet AS t2 ON t1.stuid = t2.stuid
JOIN pets AS t3 ON t3.petid = t2.petid WHERE t3.pettype = 'cat' INTERSECT
  SELECT t1.fname FROM student AS t1 JOIN has_pet AS t2 ON t1.stuid = t2.stuid
  JOIN pets AS t3 ON t3.petid = t2.petid WHERE t3.pettype = 'dog'
```

An Effective Database Simplifies Verification

Find the first name of students who have both cat and dog pets.

Ownership (merged)

StuId	First Name	Last Name	PetType	PetId
Student_A	Alex	Pan	Cat	Pet_1
Student_B	Alex	Wei	Dog	Pet_2

Recap

Recap

- Scalable Oversight: assisting humans to evaluate AI systems

Recap

- Scalable Oversight: assisting humans to evaluate AI systems
- Example Method: debate, self-critique, decomposition, etc

Recap

- Scalable Oversight: assisting humans to evaluate AI systems
- Example Method: debate, self-critique, decomposition, etc
- Sandwiching evaluation paradigm

Recap

- Scalable Oversight: assisting humans to evaluate AI systems
- Example Method: debate, self-critique, decomposition, etc
- Sandwiching evaluation paradigm
- Text-to-SQL Example

Why you should consider working on Scalable Oversight?

Why Scalable Oversight?

Why Scalable Oversight?

- Neglected (not many people are working on it now)
  - Don’t need to worry about being scooped as much

Why Scalable Oversight?

- Neglected (not many people are working on it now)
  - Don’t need to worry about being scooped as much
- Tractable (possible to make progress)
  - I have outline a few methods that were effective.

Why Scalable Oversight?

- Neglected (not many people are working on it now)
  - Don’t need to worry about being scooped as much
- Tractable (possible to make progress)
  - I have outline a few methods that were effective.
- Important (high impact if done properly)

Scalable Oversight is Important

Scalable Oversight is Important

- AI capability is increasingly capable and doing complex tasks

Scalable Oversight is Important

- AI capability is increasingly capable and doing complex tasks
- A lot of surprises from the past 10 years;
  - Probably more in the coming decade.

Scalable Oversight is Important

- AI capability is increasingly capable and doing complex tasks
- A lot of surprises from the past 10 years;
  - Probably more in the coming decade.
- Great if we can control powerful AI systems well
  - Catastrophic if we cannot

Scalable Oversight is Important

- AI capability is increasingly capable and doing complex tasks
- A lot of surprises from the past 10 years;
  - Probably more in the coming decade.
- Great if we can control powerful AI systems well
  - Catastrophic if we cannot
- Analogy: Nuclear fusion is easy, controlling is non-trivial

Thanks!