COGSCI 2026

Effective Explanations Support Planning Under Uncertainty

Hanqi Zhou · Britt Besch · Charley M. Wu · Tobias Gerstenberg

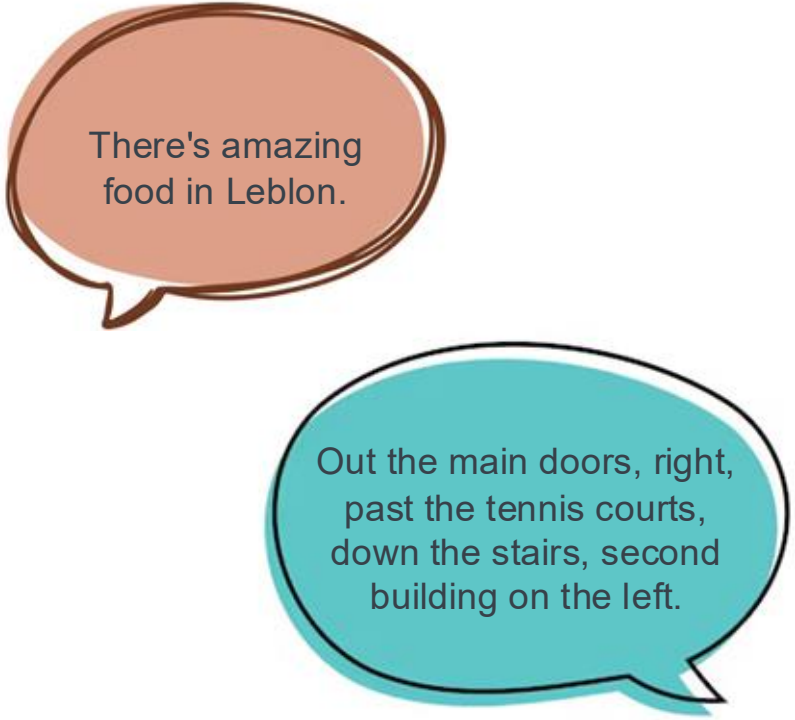
University of Tübingen · TU Darmstadt · MPI for Biological Cybernetics

hanqi.zhou@uni-tuebingen.de

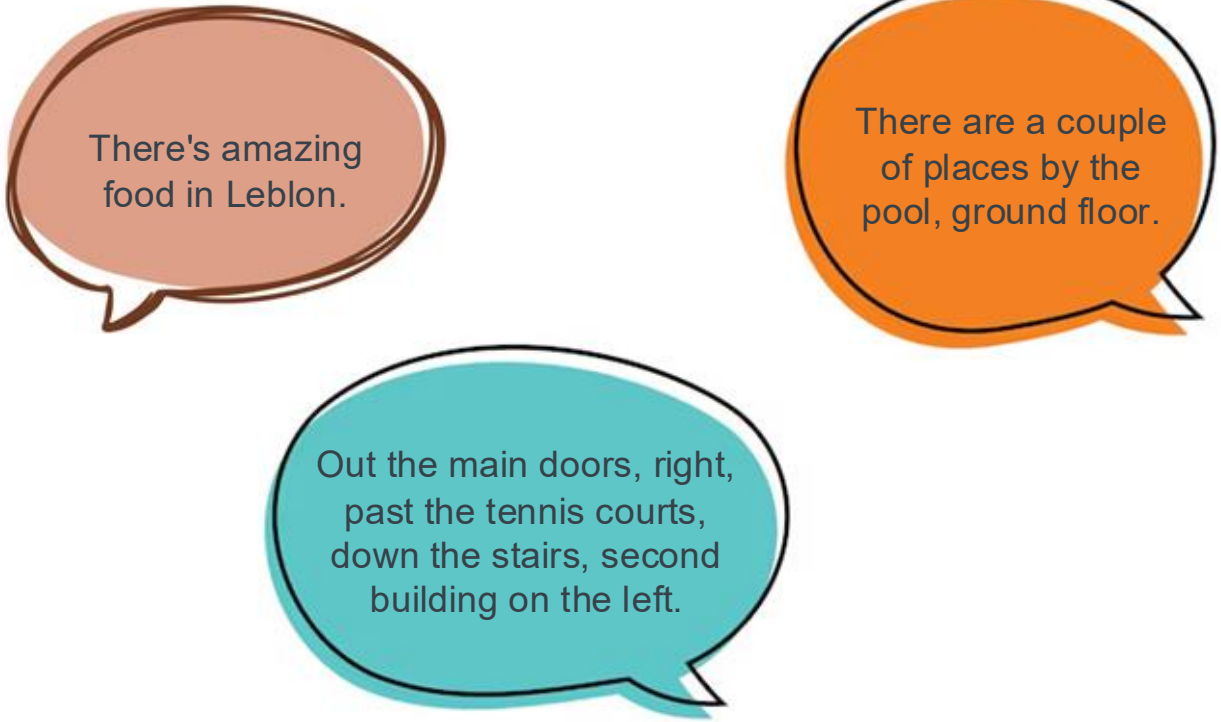
GitHub: cicl-stanford/explaining_how_cogsci26

How to get food during the lunch break?

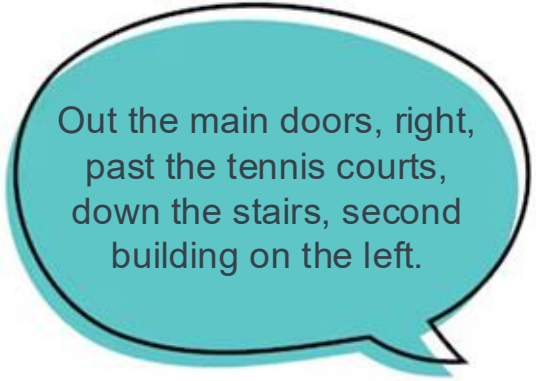
How to get food during the lunch break?

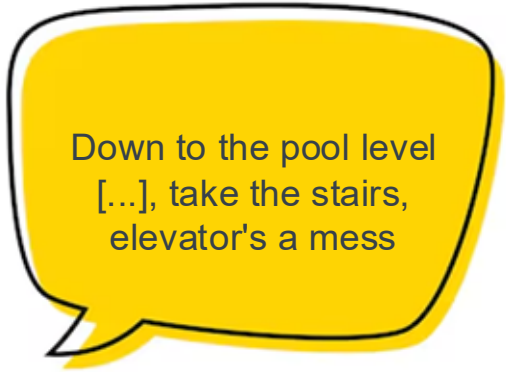


How to get food during the lunch break?



How to get food during the lunch break?





THE QUESTION

- LIMITS: attention, memory, words, etc.
- ASYMMETRY: the explainer knows the task and the environment; the listener doesn't
- UNCERTAINTY: the explainer doesn't know what the listener will be facing when they use it

THE QUESTION

- LIMITS: attention, memory, words, etc.
- ASYMMETRY: the explainer knows the task and the environment; the listener doesn't
- UNCERTAINTY: the explainer doesn't know what the listener will be facing when they use it

A good explanation selects what matters and structures it so the listener can act.

THE QUESTION

From belief to action

$$S(e \mid w) \propto \exp(\lambda \cdot U(e, w))$$

utility = **help to the listener** − **cost to say it**

MOST ACCOUNTS

help = right **belief**

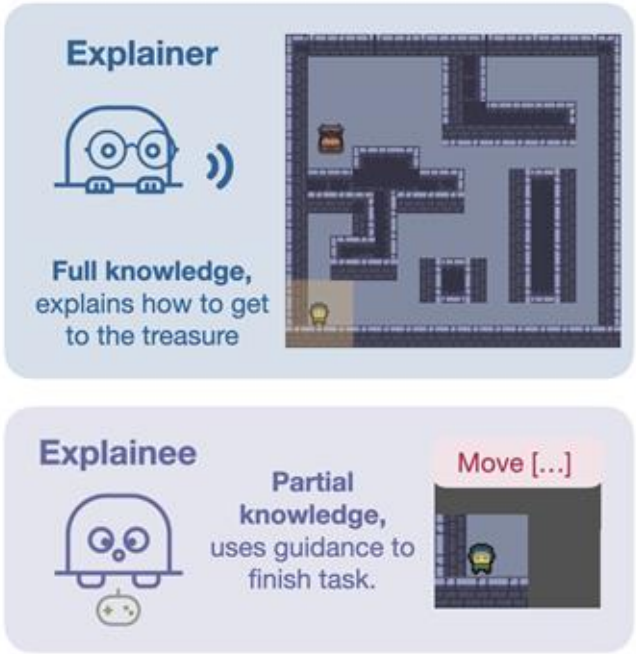
OURS

help = right **action** under the updated belief

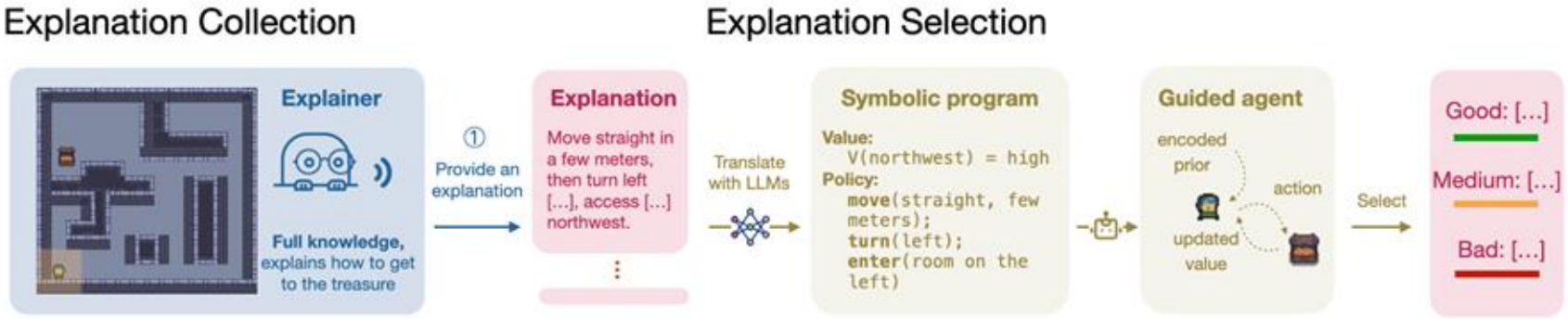
Grice 1975
Lombrozo 2006
Frank & Goodman 2012 (Rational Speech Act, RSA)

A controlled navigation

- Explainer sees the FULL map
- Navigator sees only a LOCAL window
- One free-text message, one shot



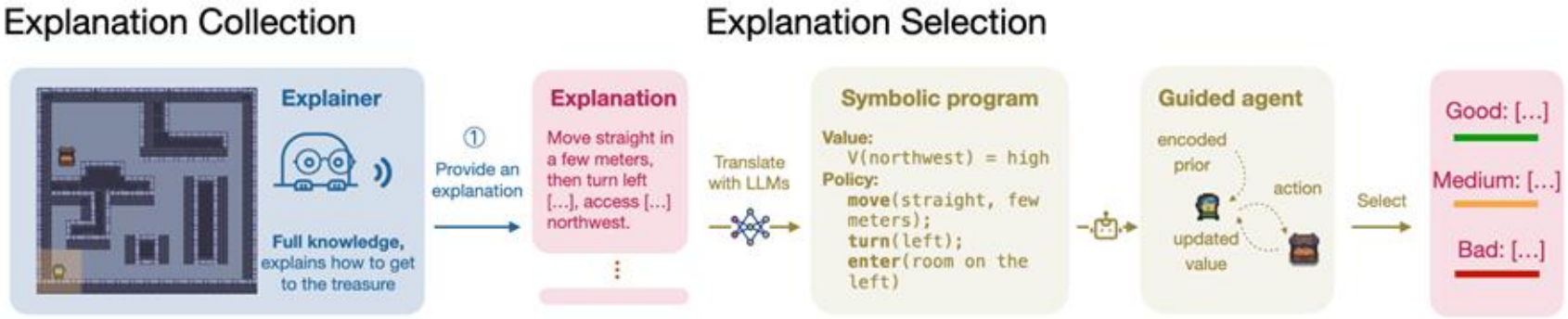
Make a model act on the explanation



Collect

real human explanations, written from the full map

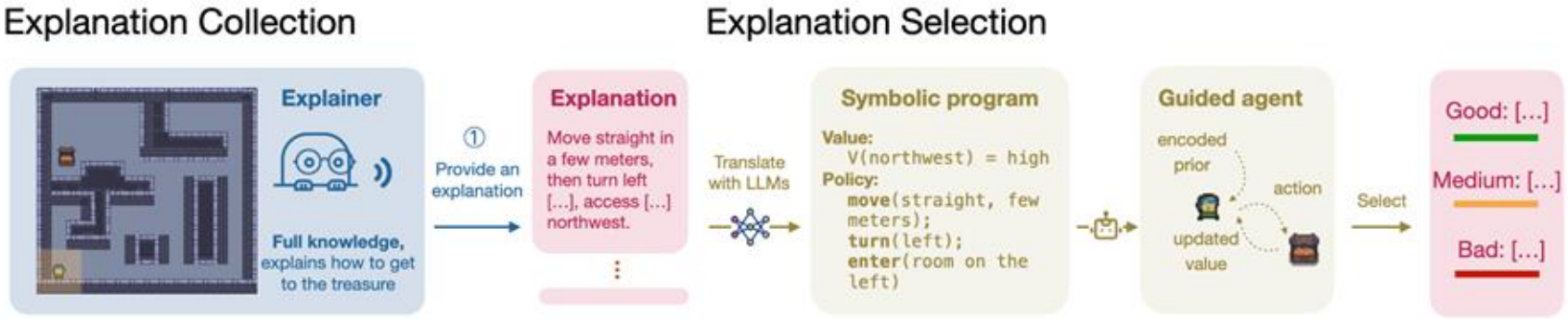
Make a model act on the explanation



Compile

an LLM turns each message into a small runnable program

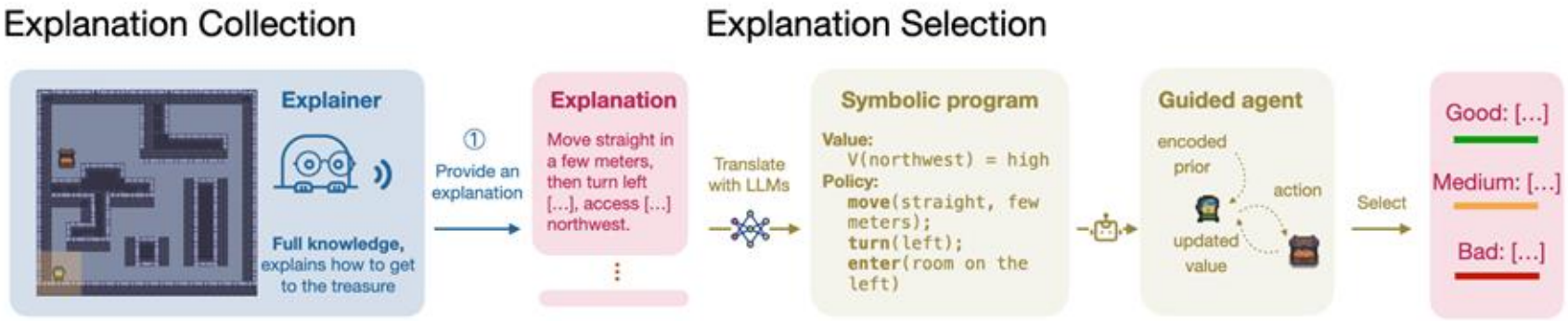
Make a model act on the explanation



Plan & act

an agent runs the program from a local view

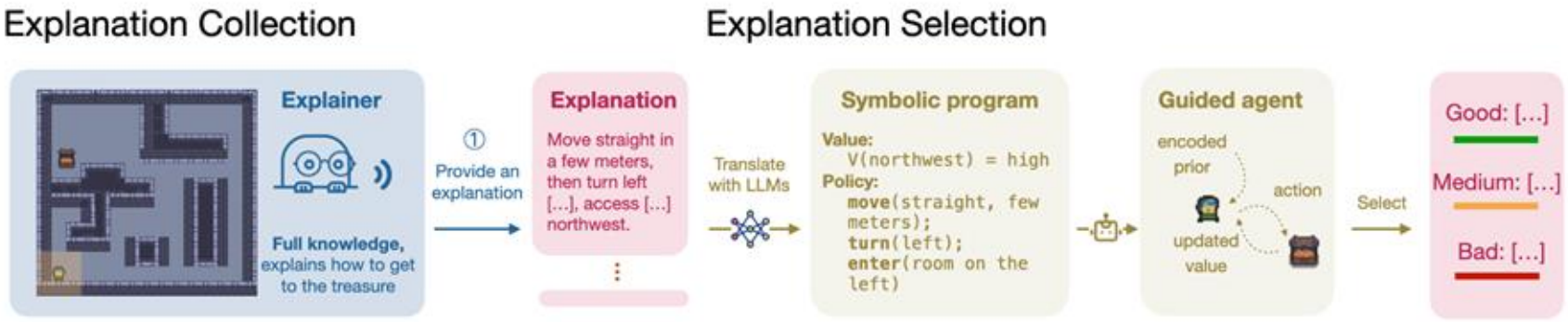
Make a model act on the explanation



Score

rate the resulting behaviour → good / medium / bad

Make a model act on the explanation



Compile

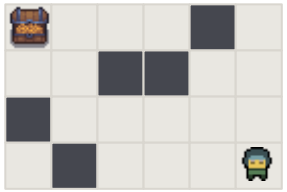
an LLM turns each message into a small runnable program

Score

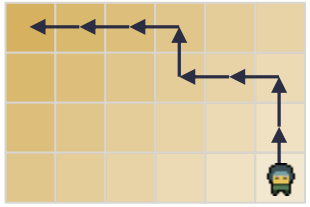
rate the resulting behaviour → good / medium / bad

Component: compile

Actual world w & Explanation e



Conveyed world (V, ρ)
← from e

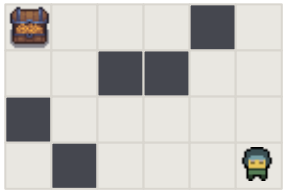


V value prior (heat)
where reward is, spread over the whole map.

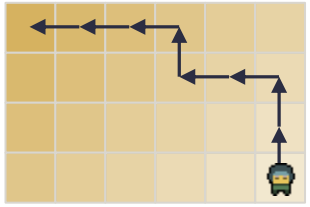
ρ policy prior (arrow)
a cell-by-cell hint of what to do.

Component: compile

Actual world w & Explanation e



Conveyed world (V, ρ)
← from e



V value prior (heat)
where reward is, spread over the whole map.

ρ policy prior (arrow)
a cell-by-cell hint of what to do.

"Go straight, turn right at the third opening, then keep going until you reach the wall. You'll see the treasure there."

Policy:

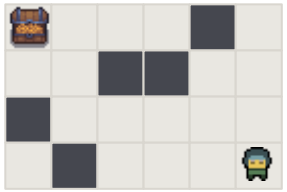
```
move_until(forward, opening==3)
turn(right)
move_until(forward, wall)
```

Value:

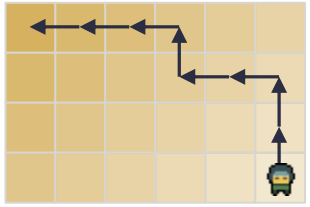
```
V(wall ahead) = high
```

Component: compile

Actual world w & Explanation e



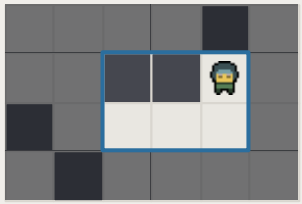
Conveyed world (V, ρ)
← from e



V value prior (heat)
where reward is, spread over the whole map.

ρ policy prior (arrow)
a cell-by-cell hint of what to do.

Plan $\pi_{(V,\rho)}$ the agent acts from a local view



repeat each step

Observe → **Decide** → **Act**

re-query

combine:

$$\pi(a) \propto \rho(a) \cdot \exp\left(Q_V/\tau\right)$$

priors run out / dead-end → planning cost

Component: utility

AN EXPLANATION

"Go straight, turn right at the third opening, then keep going until you reach the wall. You'll see the treasure there."

ITS CONVEYED WORLD

Policy:

```
move_until(forward, opening==3)
turn(right)
move_until(forward, wall)
```

Value:

```
V(wall ahead) = high
```

Component: utility

AN EXPLANATION

"Go straight, turn right at the third opening, then keep going until you reach the wall. You'll see the treasure there."

ITS CONVEYED WORLD

Policy:

```
move_until(forward, opening==3)
turn(right)
move_until(forward, wall)
```

Value:

```
V(wall ahead) = high
```

WHAT WE MEASURE

Did it resolve into action? ≈ 0.4

REPLAN · re-queries when stuck · lower

Was it efficient? 1.1×

LEN · realized vs. shortest path · lower

Did it arrive? 95%

SUCC · reached the goal in budget · higher

$R(\pi^*) = \gamma \cdot SUCC - \alpha \cdot REPLAN - \beta \cdot LEN \rightarrow$ Good

Component: utility

STANDARD RSA

$$S(e \mid w) \propto \exp(\lambda \cdot U(e, w))$$

$$U(e, w) = E_{\hat{w} \sim L(\hat{w} \mid e)} [\ \log L_0(w \mid e)\ ] - C(e)$$

Help: listener assigns probability to the intended world

OURS

$$S(e \mid w) \propto \exp(\lambda \cdot U(e, w))$$

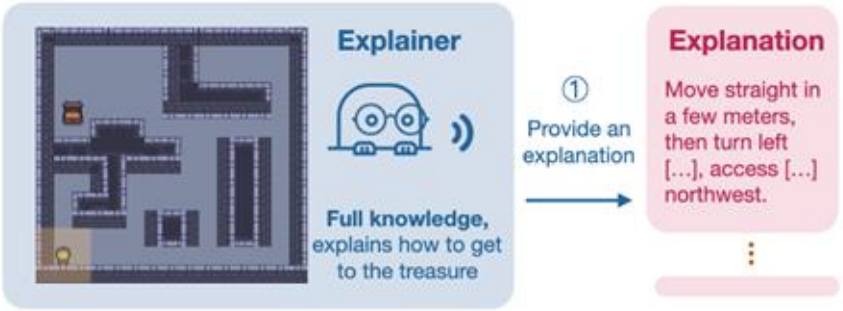
$$U(e, w) = E_{z \sim q_0(z \mid e), \tau \sim P(\tau \mid w, \pi_z)} [\ r(\tau, w)\ ] - C(e)$$

help: listener's action produces reward in the actual world

z=(V, ρ): V = value prior, ρ = policy prior
τ actual trajectory

What people say · think · do

4 preregistered experiments · 1,200 explanations · 24 maps · run on Prolific



1

Collection

N = 50

Explainers see the full map and write a message to guide a partner.

What people say · think · do

4 preregistered experiments · 1,200 explanations · 24 maps · run on Prolific

2

Helpfulness

N = 50

Rate model-selected good / medium / bad messages for each map.

What people say · think · do

4 preregistered experiments · 1,200 explanations · 24 maps · run on Prolific

3

Baseline

N = 50

Navigate every map with no explanation as map-difficulty baseline.

What people say · think · do

4 preregistered experiments · 1,200 explanations · 24 maps · run on Prolific

4

With explanation

N = 150

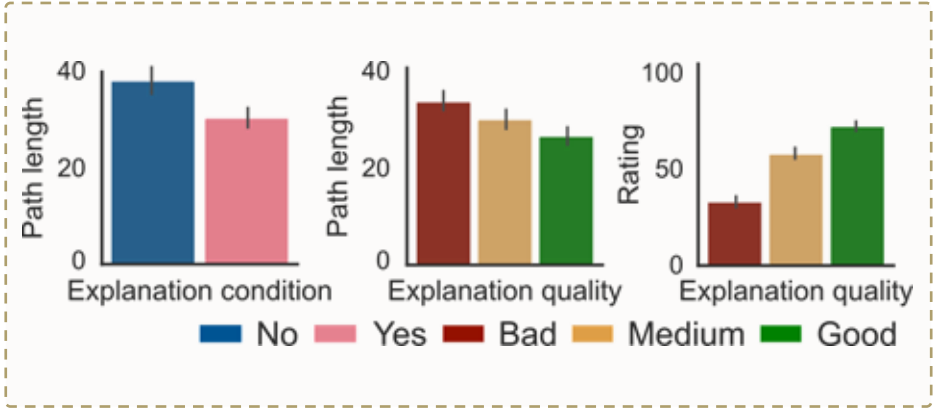
Navigate with bad / medium / good messages (within-subject).

The model's ranking predicts judgments and behavior

JUDGED MORE HELPFUL

Good > Medium > Bad

a clean monotonic ordering.



The model's ranking predicts judgments and behavior

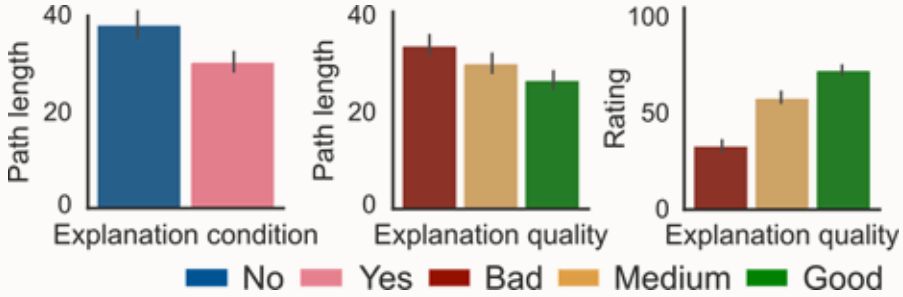
NAVIGATED MORE EFFICIENTLY

Any explanation beats none

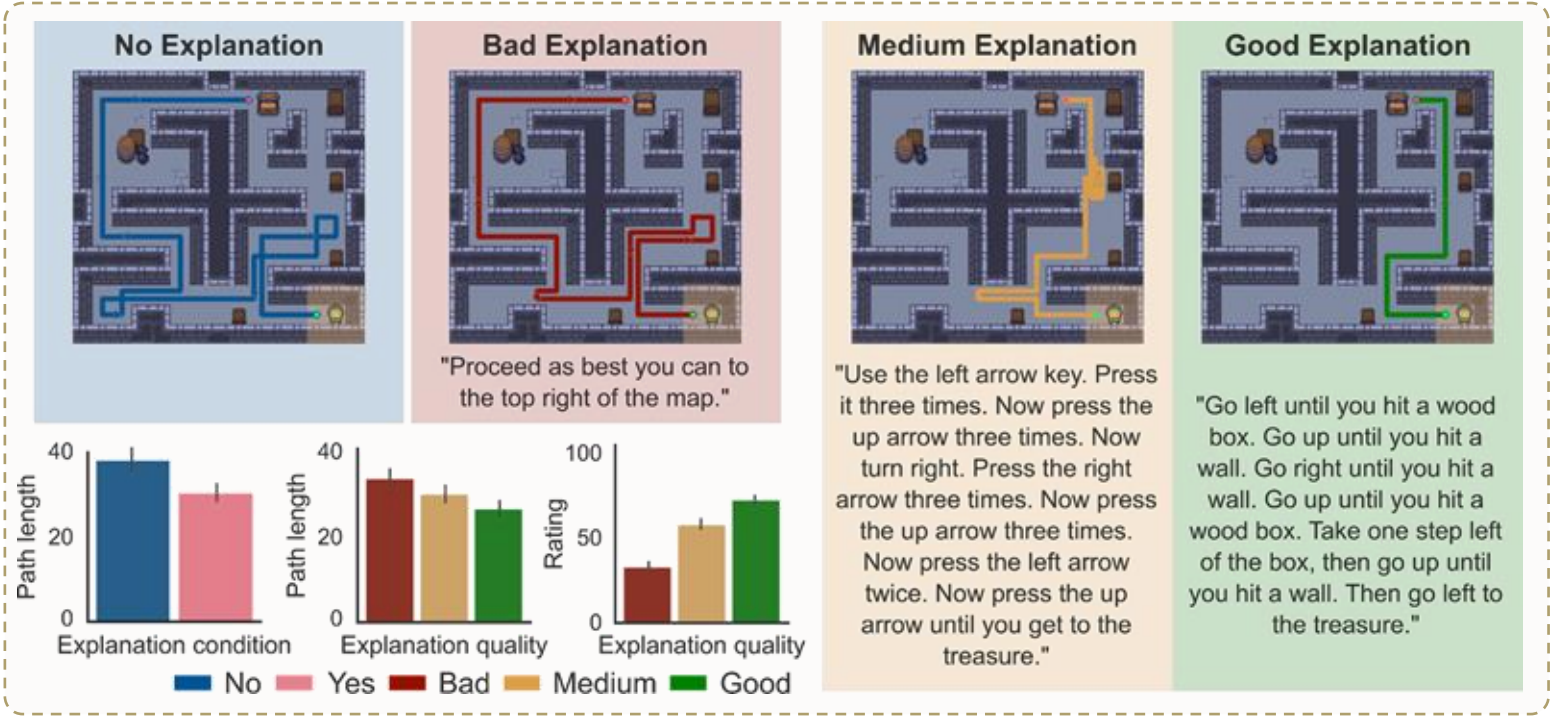
JUDGED MORE HELPFUL

Good > Medium > Bad

a clean monotonic ordering.

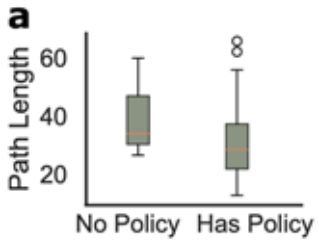


The model's ranking predicts judgments and behavior



Policy for speed, value for recovery

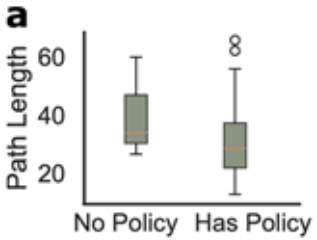
- **POLICY** for speed



Path length drops when a policy is given -> faster navigation.

Policy for speed, value for recovery

- **POLICY** for speed



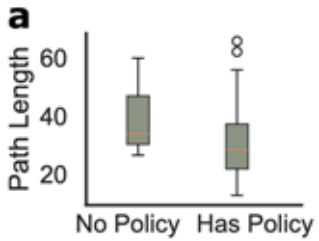
Path length drops when a policy is given -> faster navigation.

how policy breaks

Overcomplicated
long conditional instructions exceed the compiler.

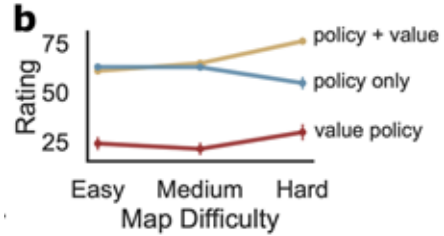
Policy for speed, value for recovery

- **POLICY** for speed



Path length drops when a policy is given -> faster navigation.

- **VALUE** for recovery



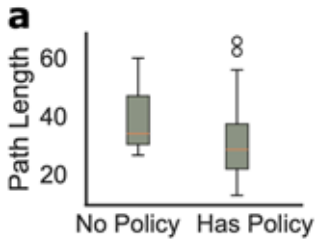
On hard maps, policy + value holds up while policy-only collapses.

how policy breaks

Overcomplicated
long conditional instructions exceed the compiler.

Policy for speed, value for recovery

- **POLICY** for speed

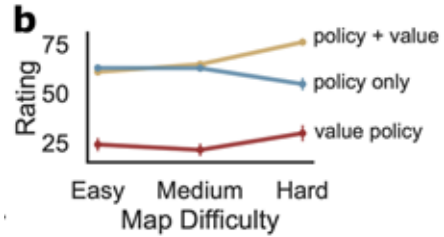


Path length drops when a policy is given -> faster navigation.

how policy breaks

Overcomplicated
long conditional instructions exceed the compiler.

- **VALUE** for recovery



On hard maps, policy + value holds up while policy-only collapses.

how value breaks

Spatial ambiguity a region with no referable landmark; the agent keeps re-querying.

NEXT STEP

Richer worlds and varied agents

1. Scaling map complexity
from grid dungeons to open, structured worlds (trees, water, animals) with multiple viable routes.

2. Varying the agent
new capabilities (Walk · Locate · Clamber · Vault · Hop · Forage) and a changing field-of-view.

How do explainers adapt to what a listener KNOWS and what they CAN DO?

Takeaways & Limitations

Judge an explanation by what the listener can **do** with it - a model that **acts** on explanations recovers both how people rate them and how well they navigate.

THE OUTLOOK

Takeaways & Limitations

Judge an explanation by what the listener can **do** with it - a model that **acts** on explanations recovers both how people rate them and how well they navigate.

1 The message carries more

Caution & effort cues: “careful, this map’s tricky”

Error signals: “if you hit the barrel, you’ve gone too far”.

THE OUTLOOK

Takeaways & Limitations

Judge an explanation by what the listener can **do** with it - a model that **acts** on explanations recovers both how people rate them and how well they navigate.

1 The message carries more

Caution & effort cues: “careful, this map’s tricky”

Error signals: “if you hit the barrel, you’ve gone too far”.

2 We assume perfect grounding

Our agent maps words to the world with **almost no ambiguity**.

Reference resolution: “the barrel,” “the room on the left” but which one?

Thank you

Questions are very welcome.

Hanqi Zhou
hanqi.zhou@uni-tuebingen.de

GitHub: cicl-stanford/
explaining_how_cogsci26