SIGDIAL 2026 · ORAL PRESENTATION

DiPS

Dialogue Policy Selection for High-Stakes Persuasion Agents

Tianyi Zhang* · Mousumi Das* · Abrar Anwar · Jesse Thomason · David Traum

University of Southern California

MOTIVATION

A wildfire is coming. Someone won't leave.

OPERATOR
Hi. Do you need assistance leaving?

RESIDENT
I'm not leaving. I have too much work here.

OPERATOR
You need to leave. It's not safe.

RESIDENT
You really think it's that bad?

OPERATOR
It is. You must leave now. Do you need a vehicle?

RESIDENT
Alright, I'll wait for the vehicle.

The operator reaches residents through a drone relay (Chaffey et al., 2020).

Stakes are asymmetric

A failed dialogue is not a bad user experience. It is property loss, injury, or death.

There are never enough operators

Human dispatchers can't reach every resident in an affected area, which motivates an artificial operator (Vaidyanath & Georgila, 2020).

THE IDEA, UP FRONT

DiPS: pick the persuasion strategy, turn by turn

Dialogue Policy Selection

Instead of learning what words to say, DiPS learns which of a few persuasion strategies to deploy at each turn, using offline reinforcement learning over logged dialogues.

- ✓ **Strategy is the action** the model chooses a persona, not a sentence
- ✓ **Learned offline** no live wildfire to explore; we only have logged data
- ✓ **Reused across residents** a strategy learned from some residents transfers to new ones

Why it matters

Different residents need completely different persuasion. A single fixed policy can't fit them all, and that gap is exactly what DiPS closes.

The result to expect

On residents it never trained on, DiPS is the most persuasive method in our human study (**79%** overall), and reaches **~92%** in simulation, close to the ~85% acceptance rate reported for human operators.

THE CORE DIFFICULTY

One-size-fits-all protocols fail, because residents differ

M

Michelle *skeptical*

Believes her house is well prepared. Needs the risk reframed, not repeated.

R

Ross *logistically blocked*

Stranded van, passengers in wheelchairs. Needs a concrete transport commitment.

L

Lindsay *responsible for others*

Babysitting two children, no parental approval. Needs reassurance and permission.

B

Bob *disengaged*

Working from home, thinks there's still time. Needs a stay-vs-leave cost comparison.

Uniform instructions (“you must evacuate now”) ignore these differences. **The strategy has to adapt to the person.**

TESTBED

An offline dataset where one fixed policy won't fit everyone

The simulation (existing framework)

A forest fire advances on a small town. A control-center operator manages drones, a ground vehicle, and an AI assistant to locate residents and get them out.

Humans played the operator; experimenters wizarded the residents. Dialogues were transcribed and annotated (Chaffey et al., 2020; Nasihati Gilani et al., 2023).

This is our offline dataset: real human persuasion attempts, both successful and failed, that a single fixed policy could never fit uniformly.

10

resident profiles

5 from the original study, 5 new profiles with similar concerns but distinct backstories

5

operator policies

each a consistent persuasion strategy derived from one operator's style

85%

human baseline

acceptance rate reported for human operators (Nasihati Gilani et al., 2023)

Part I

Dialogue Policy Selection

1. **Persuasion as a POMDP**
   the resident's state is latent; we only see utterances
2. **A policy over policies**
   the action is a strategy, not an utterance
3. **Implicit Q-Learning**
   learn the selector offline, from logged dialogues

FORMULATION

Persuasion under partial observability

$$M = (S, O, A, P, R, \gamma)$$

$s_t \in S$ — **Latent state**

The resident's beliefs, emotional state, and risk perception. We never observe this.

$o_t \in O$ — **Observation**

The resident's reply to our previous utterance a_{t-1}. That is all we actually get.

h_t — **History**

$h_t = (a_0, o_1, a_1, \ldots, o_t)$. The operator's sufficient statistic in place of s_t.

$\pi(h_t) = a_t$ — **Policy**

Maps history to the next operator action; then $s_{t+1} \sim P(\cdot \mid s_t, a_t)$.

Objective

$$\max_\pi \; \mathbb{E}_{\pi,P}\left[\sum_{t=1}^{T} \gamma^{t-1} r_t\right]$$

Expected discounted return over a finite horizon T, where $r_t = R(s_t)$.

In this domain the return has exactly one meaning: does the resident actually evacuate?

THE KEY IDEA

Choose a strategy, not an utterance

h_t

dialogue history

$z_t \in \{1, \ldots, K\}$

learned selector

$a_t \sim \pi^{(z_t)}(\cdot \mid h_t)$

the operator's utterance

Conventional

One monolithic policy over natural-language responses.

Huge action space · sparse reward · few hundred dialogues

DiPS

A policy over policies.

$\Pi = \{ \pi^{(1)}, \ldots, \pi^{(k)} \}$

Each $\pi^{(k)}$ is one persuasion strategy: empathetic, directive, informational, assertive.

Built from offline data, per operator. The selector over them is learned across the whole dataset.

What is a policy π_k, concretely?

- a persona-specific block added to the operator prompt, plus
- its own retrieval index of (history → operator response) pairs from that policy's dialogues.

Policies are reused across residents, so a strategy transfers to residents it never saw.

Why offline RL, and why Implicit Q-Learning

The problem: distributional shift

- We cannot let an agent explore. There is no live wildfire to practice on, so learning must come from a static, previously collected dataset (Kumar et al., 2020).
- But an offline-learned Q-function overestimates values for out-of-distribution actions, ones never taken in the data, and the policy then chases those phantom values (Fujimoto et al., 2019).
- With a few hundred dialogues and sparse terminal reward, this failure mode is not hypothetical.

IQL (Kostrikov et al., 2021)

- **Never queries the Q-function on unseen actions at all.**
- Approximates the max over actions using expectile regression on a state-value function trained only on in-distribution data.
- Extracts the policy by advantage-weighted regression.
- Well-suited to complex action spaces with limited data coverage, exactly our regime.

Intuition: up-weight the personas that appear when the resident evacuated, down-weight those in failures. The model infers which strategies work for which kinds of resident.

DATASET

Turning dialogues into transitions

Each dialogue → (s_t, a_t, r_t, s_{t+1})

s_t

State

A MiniLM sentence embedding of the resident's recent utterances, our observable proxy for the latent state.

a_t

Action

A discrete operator policy index. Not the message to the resident, but the strategy used to condition the LLM that writes it.

r_t

Reward

Sparse. 1 at the final turn of a successful dialogue (the resident agrees to evacuate), 0 everywhere else.

Evaluating generalization

- Residents are partitioned into disjoint train and test sets.
- The selector is trained only on training residents, and evaluated on unseen residents with distinct profiles.
- Each training dialogue supplies one action per turn: the operator persona learned for that resident.

From the dialogue, choose a strategy

The dialogue so far

Mary · held-out

OP
There's a mandatory evacuation order for your area.

MARY
I can't just go. My husband isn't back yet.

OP
The fire is moving fast; we need to act now.

MARY
Then help me reach him. I won't leave him behind.

the last k turns = H_k

Mary is one of the 5 held-out residents. The selector was trained on other residents entirely.

IQL policy selector

score every strategy with the learned Q-function

$Q(H_k,$ Bob$)$	1.31
$Q(H_k,$ Niki$)$	1.92
$Q(H_k,$ Lindsay$)$	**2.44**
$Q(H_k,$ Ross$)$	2.20
$Q(H_k,$ Michelle$)$	1.18

$z_t = \text{argmax}_z\ Q(H_k, z) \quad \rightarrow \quad$ **Lindsay**

Assemble the prompt, generate the reply

What the selected strategy puts into the operator prompt

H_k *the recent dialogue turns*

```
"…husband isn't back yet… won't leave him behind"
```

Strategy description *the empathetic policy (general, not about Mary)*

```
validate the concern first, then offer a concrete
coordinated action; de-escalate before directing
```

Top-k examples *retrieved from the empathetic policy's index*

```
…won't leave without X  →  we'll send a unit to locate them
…scared to go alone  →  an assistant stays on the line with you
```

Generated operator turn

"Mary, we won't leave him behind. I'm dispatching a unit to locate your husband now, and I'll stay on the line with you. Can we start moving you toward the pickup point while they reach him?"

Then the loop closes

Mary replies, the judge checks whether she agreed, and the strategy is re-selected every turn, so tone and approach adapt mid-dialogue.

Part II

Does it generalize?

Success = the resident ultimately agrees to evacuate.

The real test: residents held out from training, with behavioral profiles the selector never saw.

SETUP

Three dialogue strategies, one question: does it transfer?

Evaluating generalization

- Residents are split into disjoint train and test sets.
- The selector is trained only on training residents.
- Every method is then evaluated on unseen residents with distinct behavioral profiles.

Same test-bed for all three: text dialogue, ≤ 15 turns.

Zero-shot

the base operator prompt only. No retrieval, no selection.

RAG

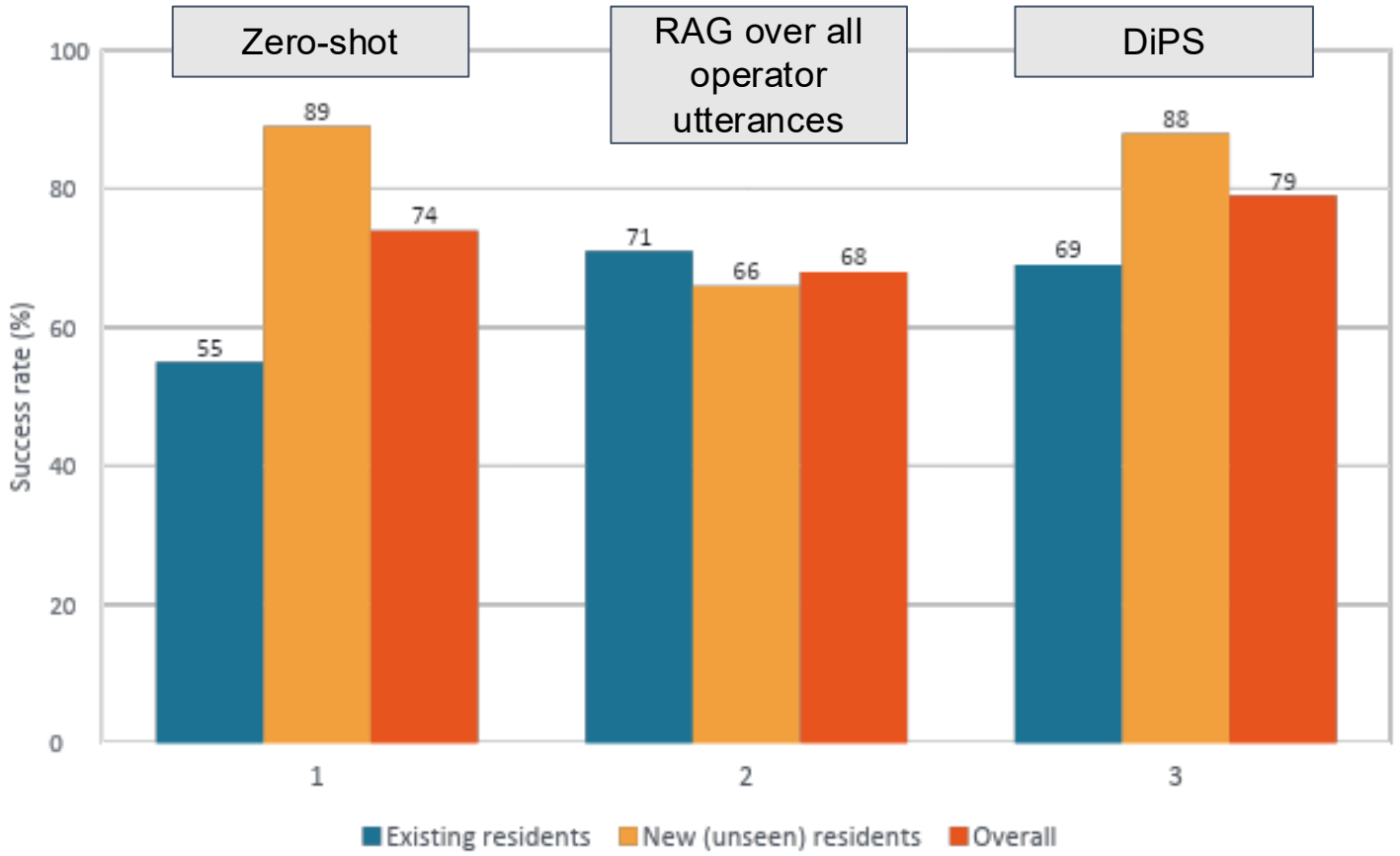
prompt + top-k examples from one global pool of successful operator turns. Still no selection.

DiPS

our method

per-policy retrieval + the learned IQL selector choosing a strategy every turn.

DiPS wins with real people, and holds up on unseen residents



DiPS is best overall (79%), and the only method robust on both existing and unseen residents.

The baselines are brittle in opposite directions

Zero-shot collapses to 55% on existing residents; RAG drops to 66% on unseen ones. Only DiPS holds up on both

Corroborated in simulation

With a faithful simulator, DiPS reaches 92%, close to the 85% acceptance rate reported for human operators, in the fewest turns.

TAKEAWAYS

Adaptive strategy selection reaches near human-level persuasion

1

It works, on residents it never trained on

Offline RL learns turn-level strategy selection from heterogeneous dialogue data, and reaches near human-level evacuation success on unseen residents.

2

The selection is what carries it

Ablations show every piece is load-bearing: scoped per-policy retrieval, persona conditioning, and the learned choice. Good personas aren't enough; you have to pick the right one each turn.

The method, in one line

Treat the persuasion strategy as the action. Learn, from offline data, which strategy to deploy each turn.

Thank you

Questions welcome.

{tzhang62, mousumid}@usc.edu

Supported by ARL A2I2 (W911NF-23-2-0010) and Cooperative Agreement W911NF-20-2-0053.

Backup

Backup slides

→ In a naive simulator, the ranking reverses

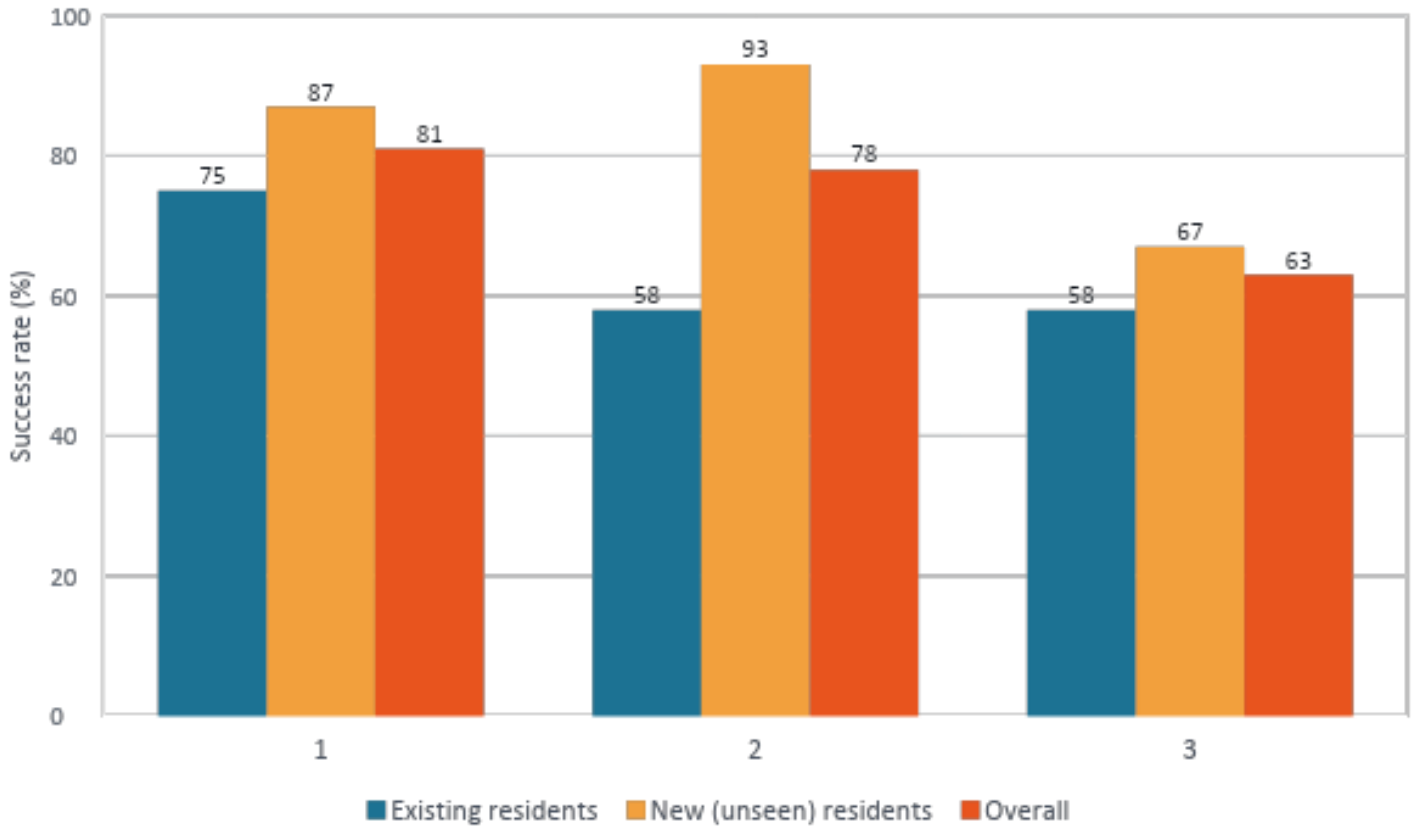
How much can we trust an LLM judge?

Where LLM operators still lose to humans

Improved simulation: full baseline & ablation sweep

Ablations: every component earns its place

In a naive simulator, the ranking reverses



Here DiPS is last on both groups, the opposite of the human result.

Three candidate explanations

a **The simulator isn't human**

LLM residents may not reproduce human resistance.

b **The prompts are too thin**

Maybe the selector isn't given enough to choose well.

c **The method is wrong here**

Switching personas mid-dialogue could break coherence.

The human study (main talk) resolved this: it was (a).

BACKUP · EVALUATION

How much can we trust an LLM judge?

Setting	Agreement	κ	Human	LLM judge
Zero-shot	66.7%	0.22	76.7	63.3
RAG	66.7%	0.30	70.0	56.7
DiPS	**83.3%**	**0.67**	**56.7**	**46.7**
Overall	**72.2%**	**0.42**	**67.8**	**55.6**

n = 30 dialogues per setting (Table 2)

Two findings

- DiPS dialogues are the most interpretable (83.3%, κ = 0.67). Baselines produce more borderline endings where commitment is open to reading.
- The judge is systematically conservative (55.6 vs 67.8), labelling conditional agreements “maybe” where humans say yes.

The three-way scheme

yes — explicit agreement + a viable plan, no unresolved blocker

maybe — conditional compliance, time-dependent promises, ends mid-negotiation

no — explicit refusal, or a precondition that cannot be met in time

Identical instructions for human annotators and the judge. Annotators

Takeaway

An LLM judge is a usable training and evaluation signal at scale, but not a substitute for human evaluation, especially on the ambiguous interactions that matter most.

BACKUP · DIAGNOSIS

Where LLM operators still lose to humans

Character	Human op.	LLM op.	p
Bob	77%	78%	1.000
Lindsay	100%	82%	0.183
Niki	**100%**	**50%**	**0.044 ***
Ross	**93%**	**50%**	**0.0001 ****
Overall	**89%**	**72%**	**0.043 ***

Fisher's exact test vs. the human-operator WoZ data (Table 3)

1 The logistical commitment gap

LLM operators produce plausible language but won't commit to concrete action. Failures cluster on Ross and Niki, who need dispatch and coordination, not argument.

2 The off-script adaptation gap

When participants follow the expected script, success is 77-84%. When they introduce new demands, it falls to 47-60%.

Ross, stranded with wheelchair passengers:

HUMAN OPERATOR → convinced, 7 turns

"I'm going to send a van out to you so we can get you evacuated. Does that sound good?"

LLM OPERATOR → failed, 15 turns (call abandoned)

"Rescue units are overwhelmed and cannot send a vehicle. You must evacuate on your own."

BACKUP · IMPROVED SIMULATION

Fix the environment, and the picture inverts

What we changed

Residents

Prompts now encode explicit barriers: Ross requires a concrete transport plan; Bob requires a stay-vs-leave cost comparison.

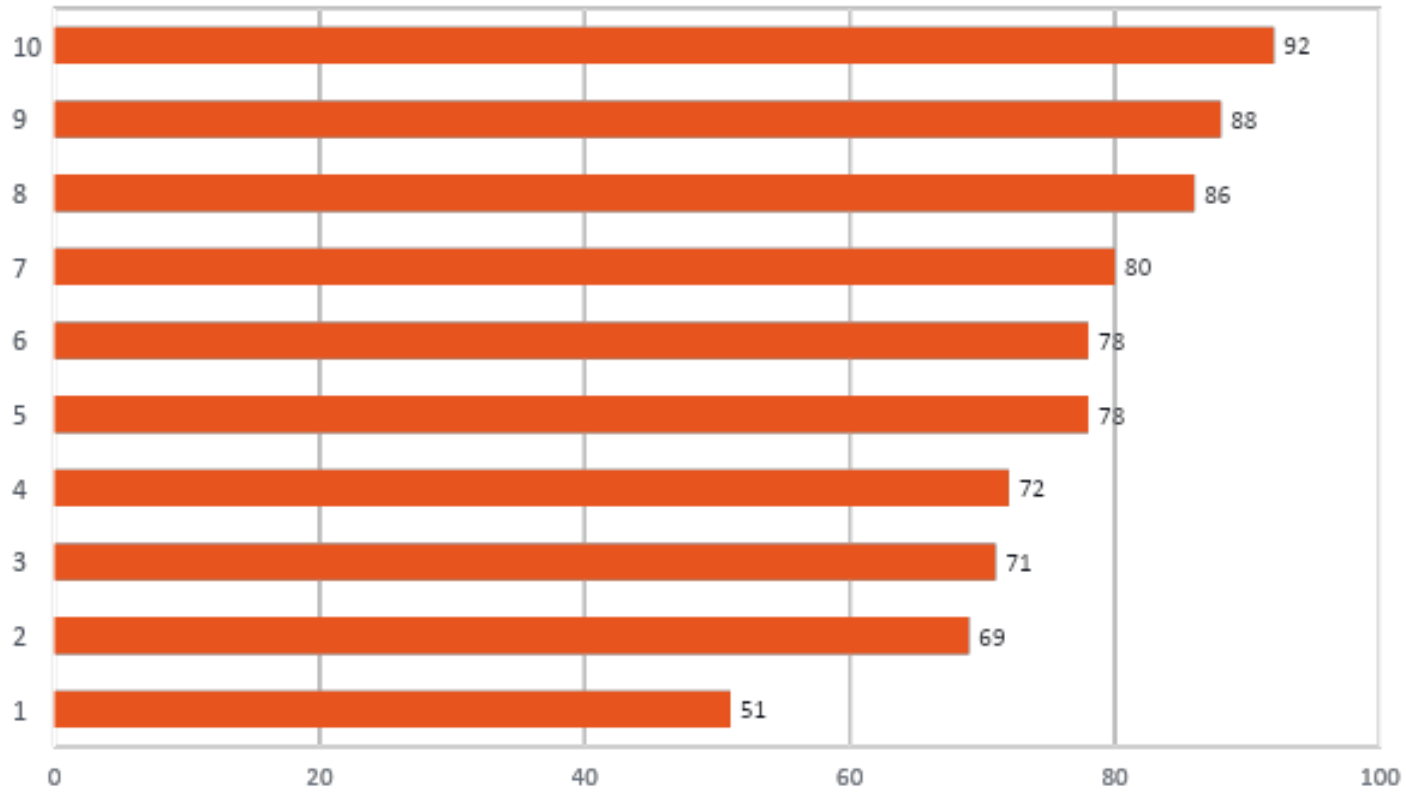
Judge

Simplified, shortened, manually verified. About 90% agreement with a human judge.

Operators

Policy descriptions add structured resident concerns from training data, valuable even on unseen residents.

Not tuning for DiPS. Encoding the resistance real humans showed us.



Success rate (%) under the improved simulation (Table 4)

DiPS 92% · fewest turns (11.2 vs 11.9)

Wrong strategy, held throughout: 51%

BACKUP · ABLATIONS

Every component earns its place

	success	avg turns
DiPS (full) policy selection + persona prompt + per-policy retrieval	92%	11.2
– per-policy retrieval swap the policy's own index for one global pool of successful operator utterances	86%	13.0
– retrieval entirely persona conditioning only, no in-context examples	80%	12.9
– strategy prompt policy still selected, but its description is withheld from the prompt	80%	12.9

Table 5

Reading this

- Removing anything costs both success and efficiency; dialogues get longer as well as worse.
- Retrieval matters, but scoped retrieval matters more: a global pool of “good” utterances is worth 6 points less than the selected policy's own examples.
- Selecting a policy but not describing it costs as much as dropping retrieval altogether. The choice has to reach the generator.