Benchmarking Information Processing Methods for Edge-Local Data Stations

Name: Sanjana Ashok
Advisors: Dipankar Raychaudhuri, Yao Liu
Mentors: Xiaotian Zhou, Zihao Ding

INTRODUCTION

Picture this...

THE SOLUTION

This is where the edge comes in

LOCAL

It lives on campus and knows this place, not the whole world.

LIVE

It sees today's reroutes and closures as they happen.

FAST

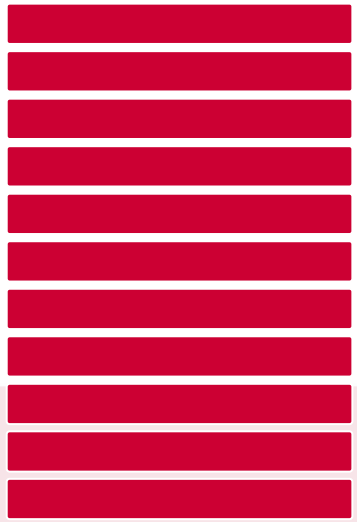
The data is a step away, so answers come back quickly.

PRIVATE

Your question never has to leave the local network.

THE CATCH

How much data do you give the model?



GIVE IT EVERYTHING

- ➔ Complete, but slow
- ➔ Costly
- ➔ Right answer may drown

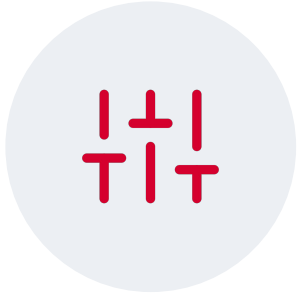
GIVE TOO LITTLE

- ➔ Cheap
- ➔ Fast
- ➔ Record you handed over may not have the correct answer

Project goal: To measure the right amount.

MY PROJECT

A benchmark to measure the trade-off fairly



CONTEXT STRATEGY

How much data, and how it's given

MODEL

The AI doing the reasoning

HARNESS

The software that runs it

When two results differ, you can tell what caused it.

Four ways to deliver the context

NONE

no data at all

checks that the model isn't cheating (control)

FULL DUMP

every record at once

complete, huge

BOUNDED

search, send a few

cheap, can miss

ORACLE

hand it the exact records

best case (control)

Two harnesses, one fixed model

LANGGRAPH
a controlled pipeline

has a pre-structured, linear path

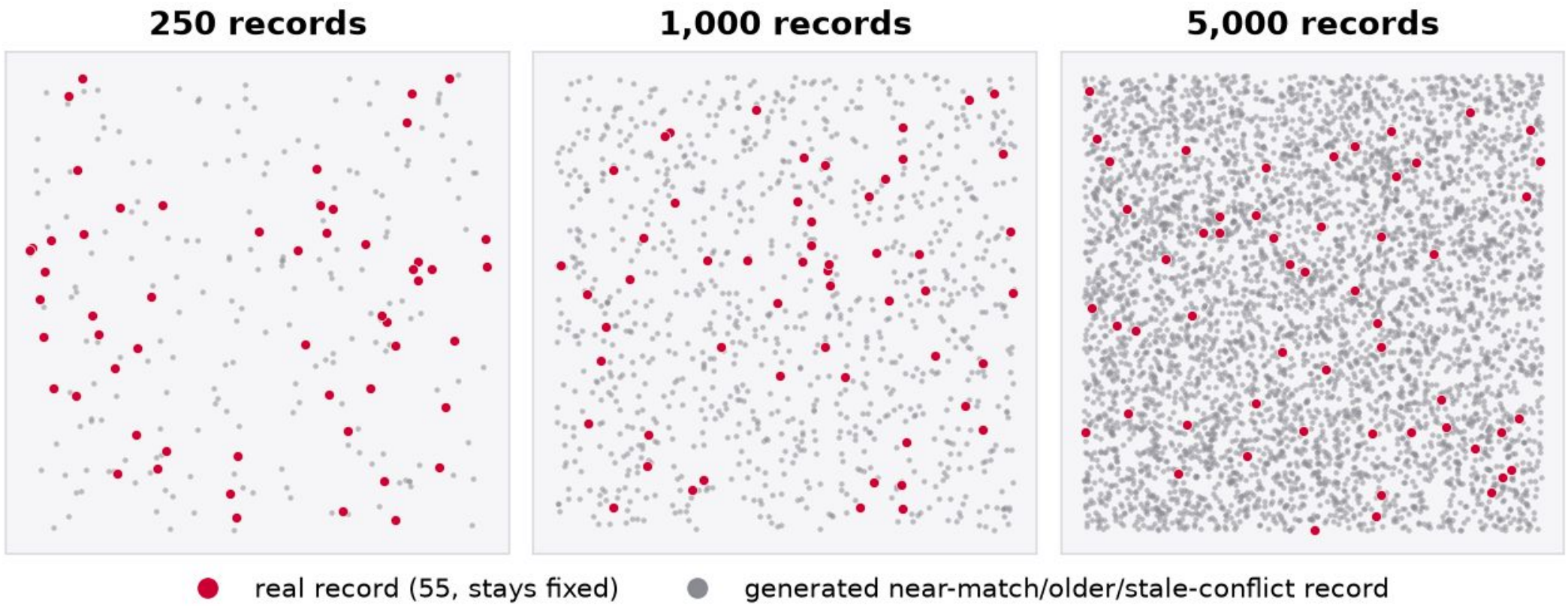
OPENCLAW
an autonomous agent

decides for itself what to fetch, in a loop

SAME MODEL
qwen3.6-plus

EXPERIMENTAL SETUP

Real Rutgers bus data



Real and local

Bus routes and stops pulled from Rutgers' PassioGo feed → 12 routes + 43 stops = 55 real records.

Understanding the results

Tokens

- ➔ A chunk of text the model reads
- ➔ More tokens = more words = higher cost

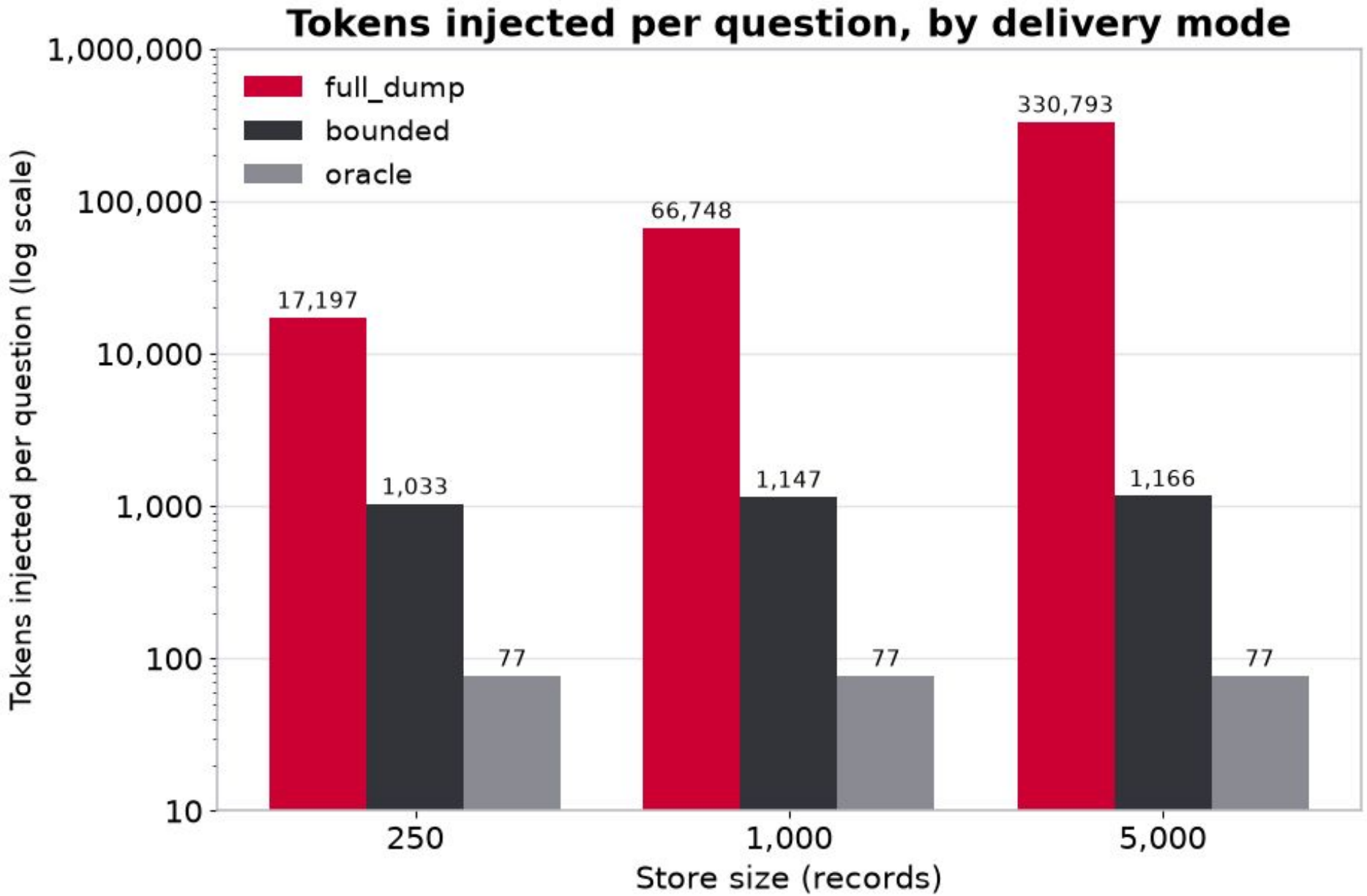
Recall

- ➔ Model needs certain records to answer correctly
- ➔ Of those records, how many did the search find?
- ➔ High recall = found what it needed

Oracle vs. bounded

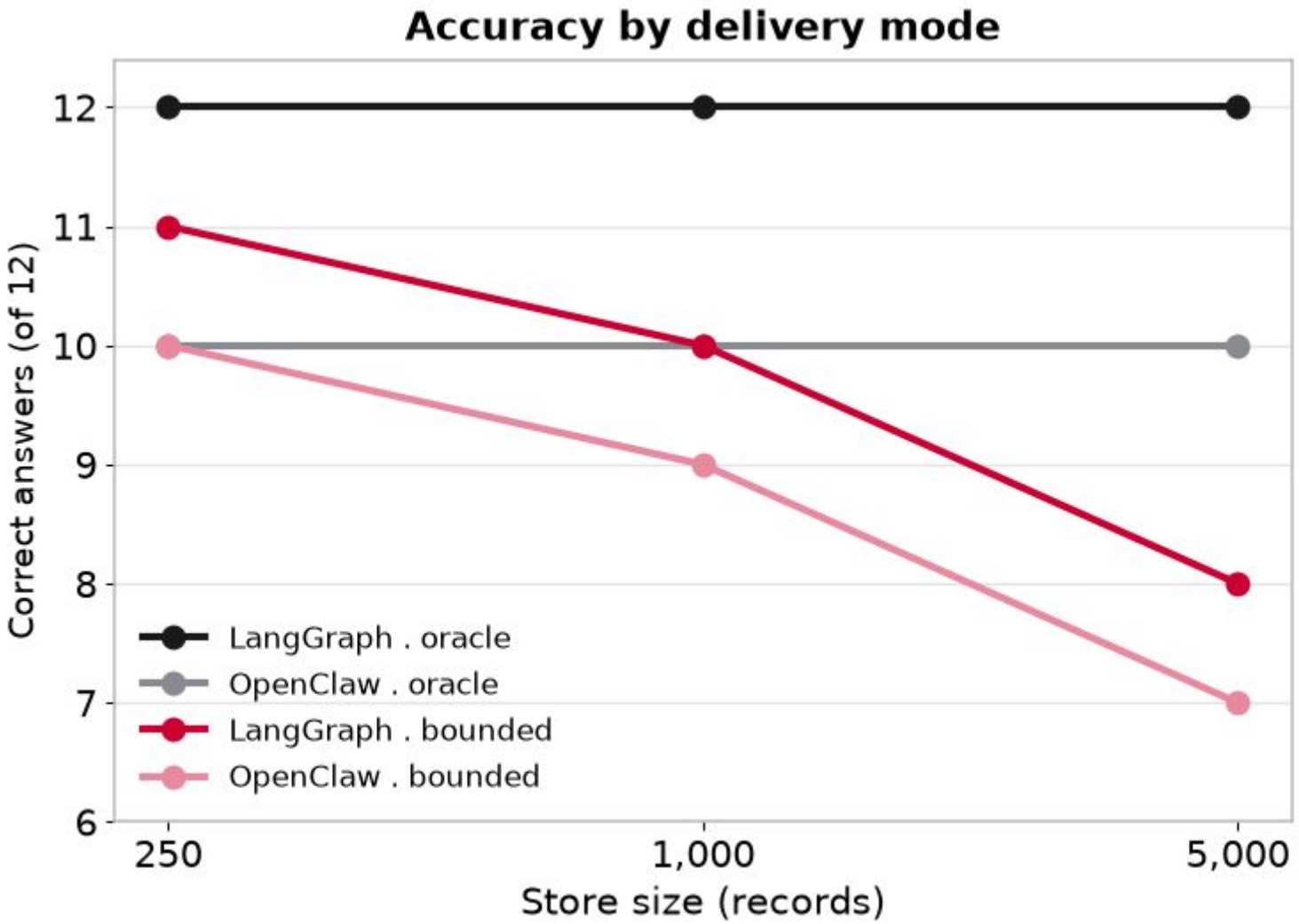
- ➔ Oracle = control; model is given exact records
- ➔ Bounded = realistic; has to find answer on a budget

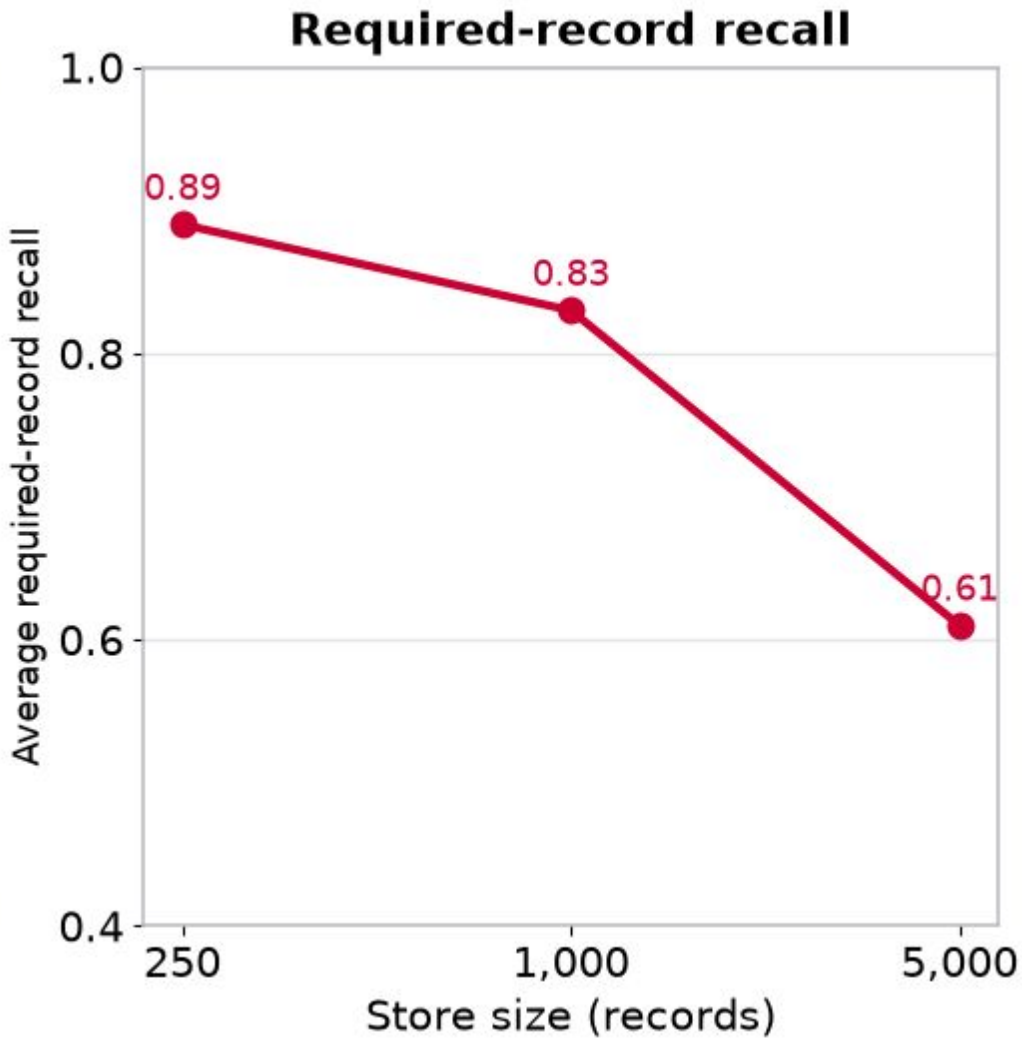
RESULT 1: COST



RESULT 2: ACCURACY

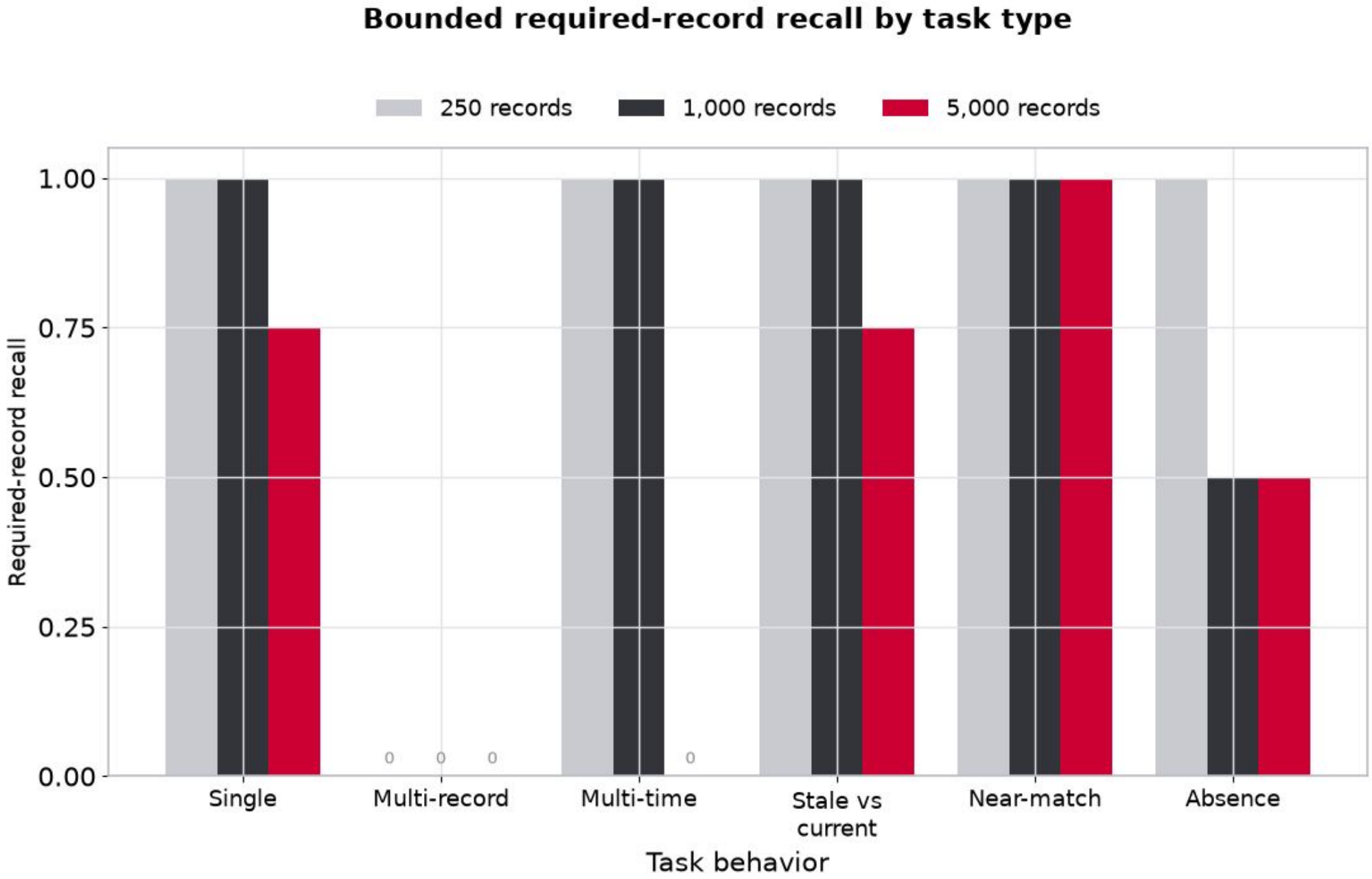
Cause of the retrieval tax → average recall across tasks drops as the memory grows





RESULT 3: ERROR ATTRIBUTION

RESULT 4: MEMORY FAILURE ATTRIBUTION



Takeaways

01

Cheap search is not free

- ➔ Bounded lookup holds cost flat (near 1,200) as the store grows 20x, while dump hits about 330,000 tokens per question
- ➔ Recall of the necessary records drops from 0.89 to 0.61 for bounded lookup
- ➔ Conclusion: The savings come back out as lost accuracy

02

Retrieval is the bottleneck, not reasoning

- ➔ Every failure was in finding the records, not reasoning over them
- ➔ Qwen3.6-plus degraded while seeing only about 1,200 tokens (0.1% of its million-token window)
- ➔ Conclusion: Since the context window was nearly empty, the search choosing what goes in it is what broke it

03

The payoff is better retrieval, not a bigger model or window

- ➔ Neither the model nor its context size was the limit
- ➔ Scaling up either one would not have helped here
- ➔ Retrieval was the step that influenced accuracy
- ➔ Conclusion: That is where an edge data station should spend its effort

Questions?

Link to project wiki

APPENDIX

What the cloud is and why it couldn’t help

THE CLOUD

- ➔ A giant data center far away
- ➔ Answers questions by using what it was trained on, but doesn’t have real-time local data

FAR · SLOW · COSTLY · LESS PRIVATE

THE EDGE

- ➔ A small local server
- ➔ For example, a Rutgers edge data station would see live bus and traffic data as it changes

LOCAL · LIVE · FAST · PRIVATE

APPENDIX

Confounds and leaks

Online access

With empty memory, the agent used web search to pull the source off the internet and answered from it. Web access is now cut off.

Leftover memory

Answers bled from one question into the next through a shared session. Every run now starts clean with its own session key.

Double feeding

One harness received the same records twice. Memory is now delivered once and checked with a marker record.

Grader mistakes

Correct answers were being scored wrong. Caught by auditing the grader before trusting any score, then regrading results.

Setup + things to keep in mind

SETUP

Answering model

Qwen3.6-Plus, served through an OpenCode Go endpoint (Qwen3.7-Max available as the strong model).

Selector & judge

DeepSeek-V4-Flash, shared by both harnesses so neither is graded by its own family.

Reproducibility

Fixed seed 42; tiktoken cl100k_base for token counts; each store rebuilds from seed + scale + snapshot hash.

Repository

Python; shared/, langgraph/, openclaw/, contextstation/, results/, with a 52-column result contract shared by both harnesses.

THINGS TO KEEP IN MIND

12-task subset

The cross-harness accuracy ran on 12 of the 24 tasks, on bounded and oracle only (full dump and none were not run through the model).

One model

Accuracy figures are Qwen3.6-Plus only; other models are future work.

Data source is shifting

Rutgers is moving from PassioGo to TripShot, records carry no timestamps, and the capture is reduced summer service.