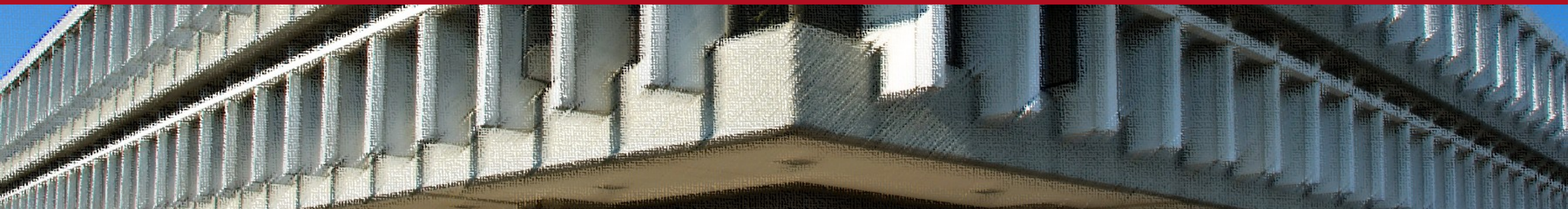


SCALM: Towards Semantic Caching for Automated Chat Services with Large Language Models

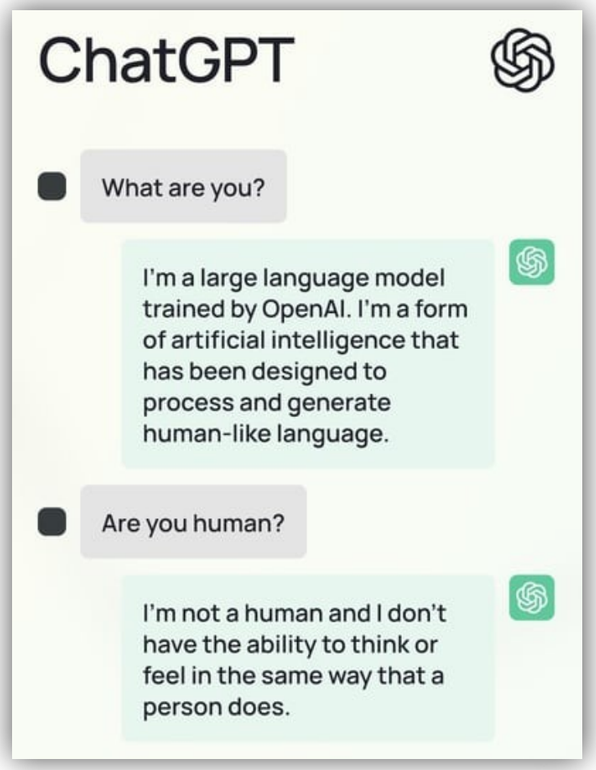
Jiaxing Li† , Chi Xu† , Feng Wang‡ , Isaac M von Riedemann† , **Cong Zhang***§ , and Jiangchuan Liu†*

† Simon Fraser University, Canada
‡ University of Mississippi, USA
** Jiangxing Intelligence Inc., China*
§ The University of Hong Kong, China



Chat with Large Language Models





LLMChat Services

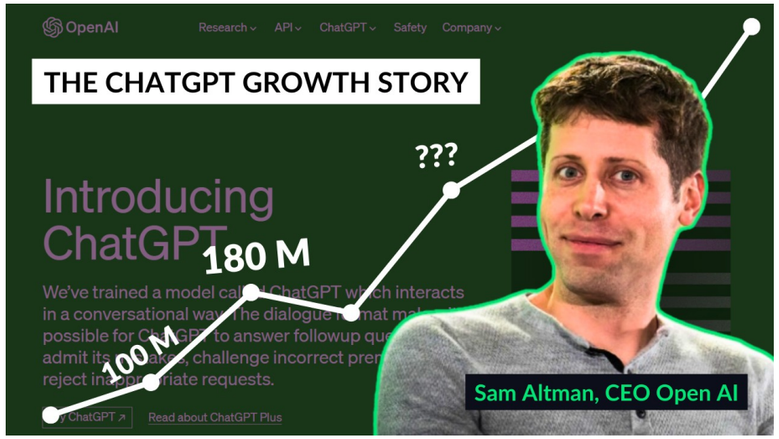
ChatGPT HuggingChat ERNIE Bot

Operational Cost of LLMChat Services

ChatGPT: 180.5M users, 1.7 billion monthly queries.

~$700K per day, $21 million per month.

(Source: Statista, Reuters, SemiAnalysis)



Related Research on Cost-effectiveness of LLM Applications

- Model Quantization/Pruning/Distillation
- Model Multiplexing
- Prompt Engineering
- Query Caching [NeurIPS ’23 a] [EMNLP ’23] [arXiv ’24]

[NeurIPS ’23 a] Zhu et al., On Optimal Caching and Model Multiplexing for Large Model Inference.

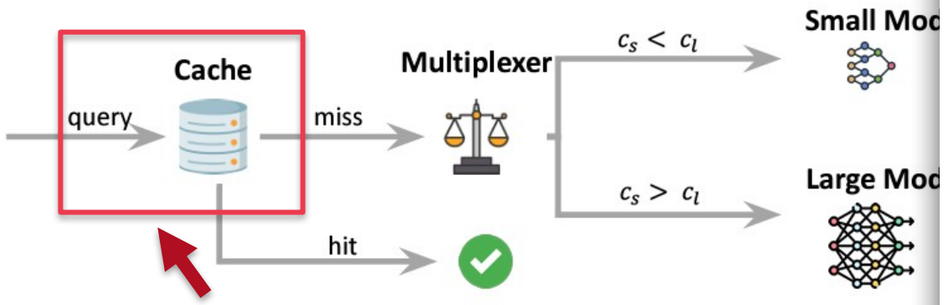
[EMNLP ’23] Stogiannidis et al., Cache me if you Can: an Online Cost-aware Teacher-Student framework to Reduce the Calls to Large Language Models.

[arXiv ’24] Zhu et al., Efficient Prompt Caching via Embedding Similarity.

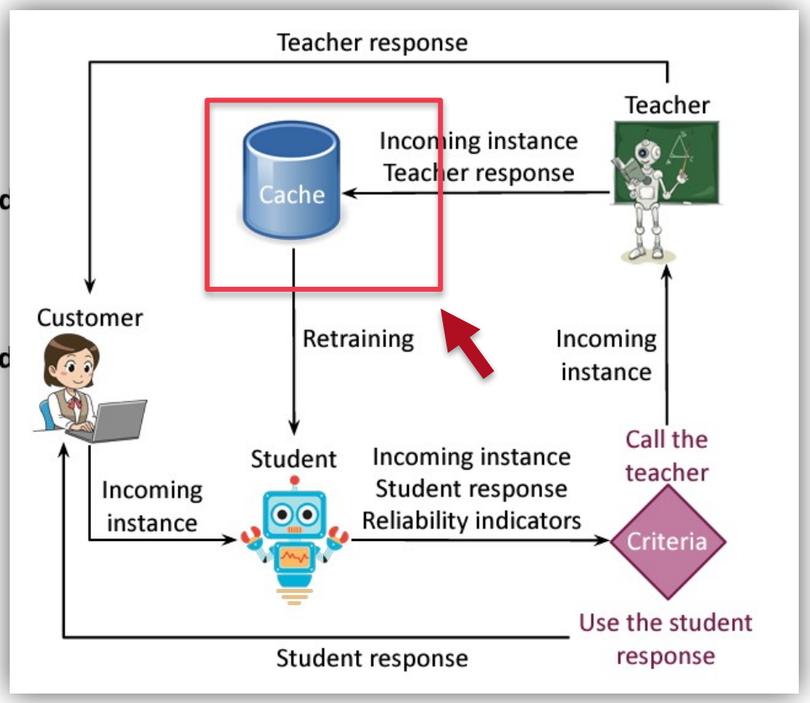


Related Works

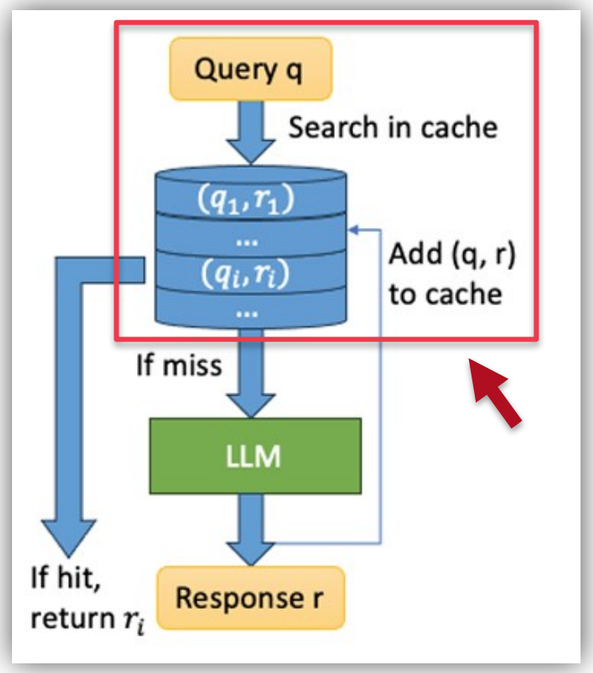
Query Caching



[NeurIPS '23 a] Zhu et al.



[EMNLP '23] Stogiannidis et al.



[arXiv '24] Zhu et al.

Does query caching help in real-world scenarios?

Motivation - Does Query Caching Help?

Cached answers are reused to reduce processing needs.

Programming - Error Handling

You

$ git pull origin main
From https://github.com/charlesxuuu/sonar-notes2clips
 * branch main -> FETCH_HEAD
fatal: refusing to merge unrelated histories
(base)

Concept - Acronym Explanation

You

GAE in reinforcement learning

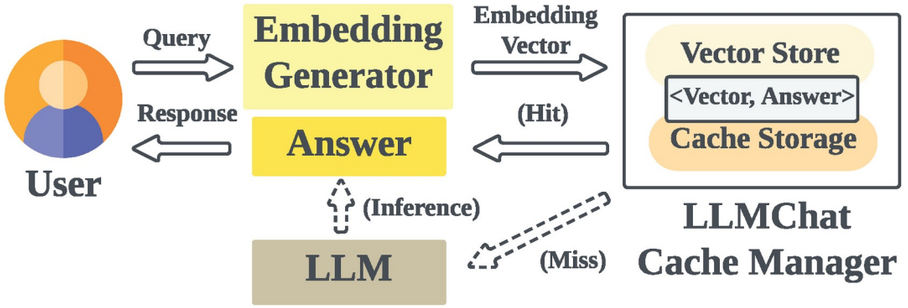
ChatGPT

Generalized Advantage Estimation (GAE) in reinforcement learning is a technique used to estimate the advantage function, which helps in determining how much better it is to take a specific action compared to the average action in a given state. This technique, introduced by John Schulman et al. in their paper, aims to reduce the variance of the advantage estimates while maintaining a bias that is as low as possible, leading to more stable and efficient training of policy gradient methods.

In simpler terms, GAE tries to balance two important aspects in reinforcement learning:

1. **Estimating Rewards:** It helps in understanding how good it is to take certain actions from a particular state, which is crucial for training an agent to make better decisions.
2. **Reducing Noise:** It aims to make these estimates more reliable by reducing the noise or randomness in the training process, making the learning process smoother and faster.

Key-Value (KV) Cache Abstraction



[NeurIPS '23 a] [EMNLP '23] [arXiv '24]
[GPTCache from Industry]

Challenges & Opportunities:

- KV Cache does not work ideally.
- Lack of understanding cache performance in real-world scenarios.
- Leveraging semantic connections between queries and patterns of human-to-LLM interactions to inform cache design.

Research Problem Statement

➢ Research Problem

“Can cache design for LLMChat services be optimized by leveraging semantic analysis of human-to-LLM dialogues in real-world scenarios?”



➢ Input, Output, and Assumptions

- Real-world human-to-LLM dialogue data.
- Data-driven analysis findings, cache architecture, and algorithms.
- 1) Identifiable patterns exist. 2) Semantic analysis can uncover these patterns.

Research Problem Statement

➢ Research Problem

"Can cache design for LLMChat services be optimized by leveraging semantic analysis of human-to-LLM dialogues in real-world scenarios?"



➢ Input, Output, and Assumptions

- Real-world human-to-LLM dialogue data.
- Data-driven analysis findings, cache architecture, and algorithms.
- 1) Identifiable patterns exist. 2) Semantic analysis can uncover these patterns.

➢ Hypothesis

"Leveraging semantic analysis of human-to-LLM dialogues to inform cache design will reduce operational costs for LLMChat services."

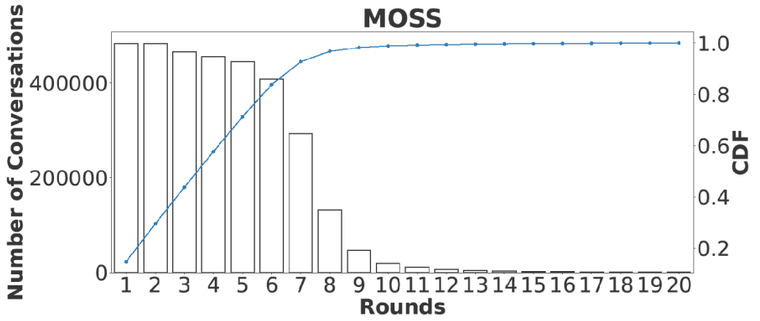


➢ Evidence

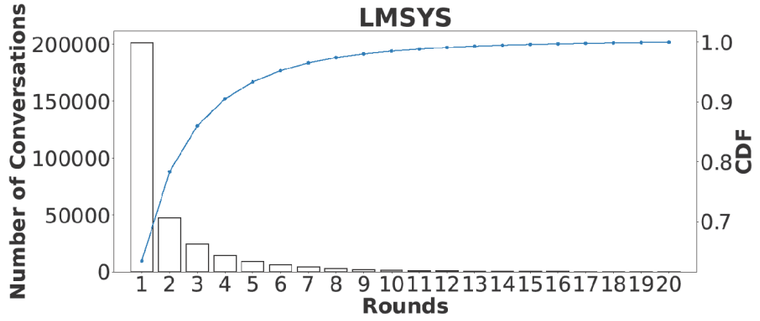
Theoretical proofs, trace-driven analysis, and real-world experiments.

Data-Driven Analysis and Findings

- MOSS and LMSYS datasets
  - **Multiple rounds**
  - Similar queries exist
  - Leverage default categories



Avg 6.7 rounds



Avg 1.8 rounds

Conversation 1:

Round 1:

Query 1: Who is Elon Musk?

LLM Answer 1: Elon Musk is a business magnate, industrial designer, and ...

Round 2:

Query 2: When did he found SpaceX?

LLM Answer 2: Elon Musk founded SpaceX, or Space Exploration Technologies Corp., in March 2002.

Round 3:

Query 3:

LLM Answer 3:

... ...

Data-Driven Analysis and Findings

- **MOSS and LMSYS datasets**
  - Multiple rounds
  - **Similar queries exist**
  - Leverage default categories

	Similarity	Query 1	Query 2
×	0.87	How can I develop a creative approach to problem-solving?	How should I approach problem solving with a team?
✓	0.90	How can I develop better organizational leadership skills?	What methods should I use to develop my leadership skills?
✓	0.93	What methods can I use to monitor customer satisfaction?	What are the best ways to measure and improve customer satisfaction?
✓	0.96	What strategies should I use to optimize the performance of my team?	What techniques can I use to optimize my team's performance?

Hits: Cosine Similarity >= 0.9

In MOSS

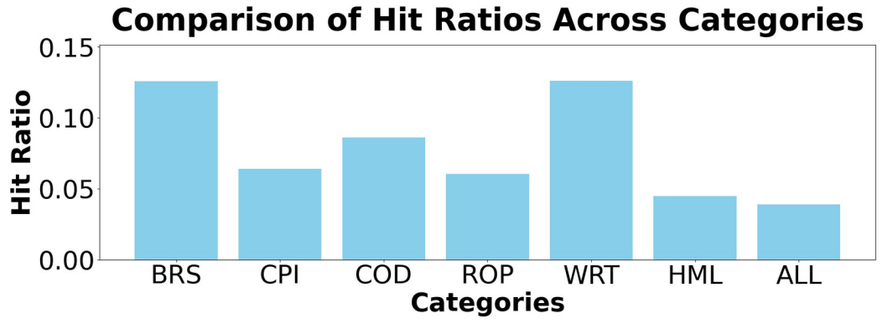
How can AI help in the field of healthcare?
How can AI technology be used in the healthcare industry to improve patient outcomes?
Can you explain how AI can be used in the healthcare industry to improve patient care?

Can you explain the difference between a traditional IRA and a Roth IRA?
Can you explain the difference between a traditional IRA and a Roth IRA in more detail?
Can you explain to me the difference between a traditional IRA and a Roth IRA?

Data-Driven Analysis and Findings

- **MOSS and LMSYS datasets**
  - Multiple rounds
  - Similar queries exist
  - **Leverage default categories**

Hit Ratio = N_{hit} / N_{access}



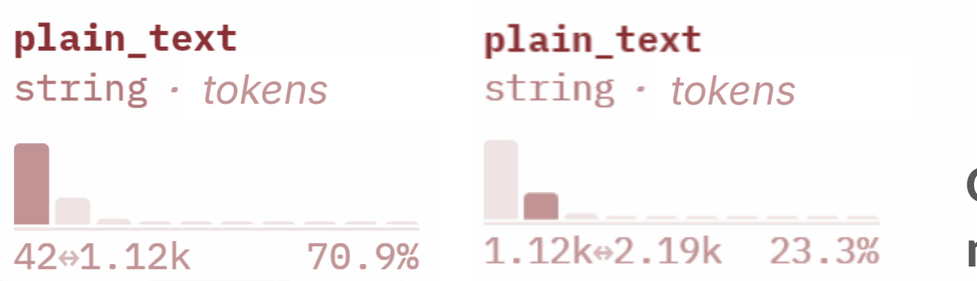
MOSS hit ratio: 3.8%

LMSYS hit ratio: 6.4%

Hit ratio in BRS: 13.5%

BRS: Brainstorming CPI: Complex Instruction COD: Code ROP: Role Playing WRT: Writing HML: Harmless

Token Saving Ratio = C_{hit} / $C_{q\&a}$

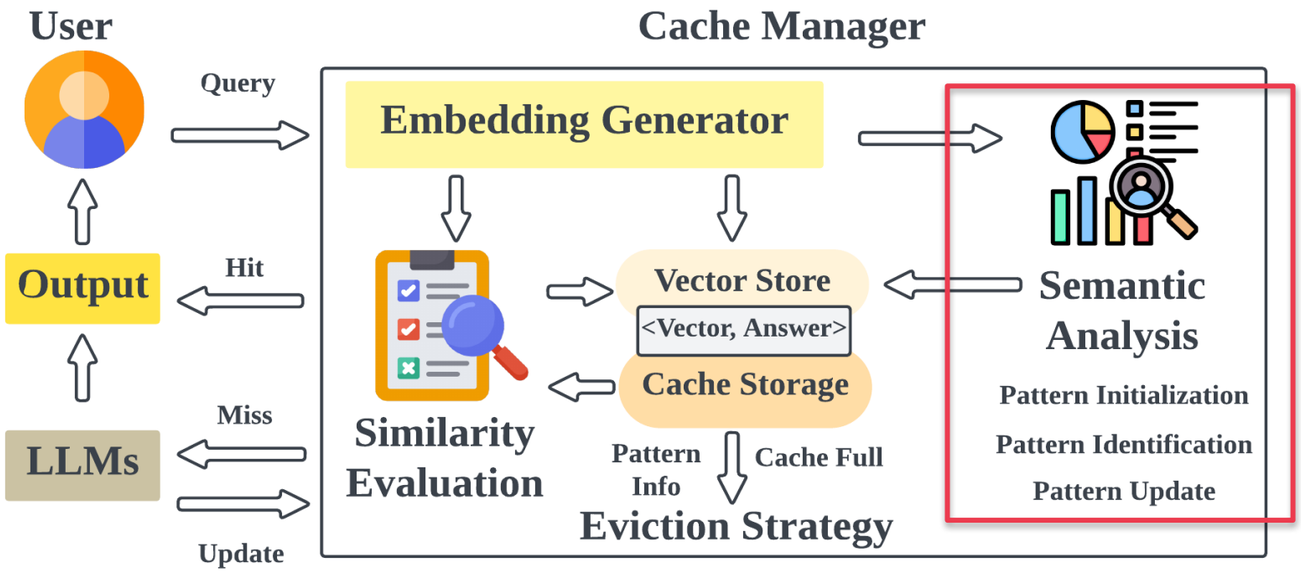


Q & A pairs have different numbers of tokens

Key Takeaways:

- Identify significant queries and their categories with semantic analysis.
- Use hit ratio and token saving ratio as metrics for evaluations.

Design Overview



➢ Semantic Analysis

- CO-HSC and SE-HSC
- Identify significant semantic categories / semantic patterns
- Identify significant queries

➢ Enhanced Similarity Evaluation

- Semantic preprocessing
- Embedding model

➢ Storage and Eviction Strategy

- Incorporate cache replacements with semantic pattern information
- Consider cache state, frequency, round, and pattern significance

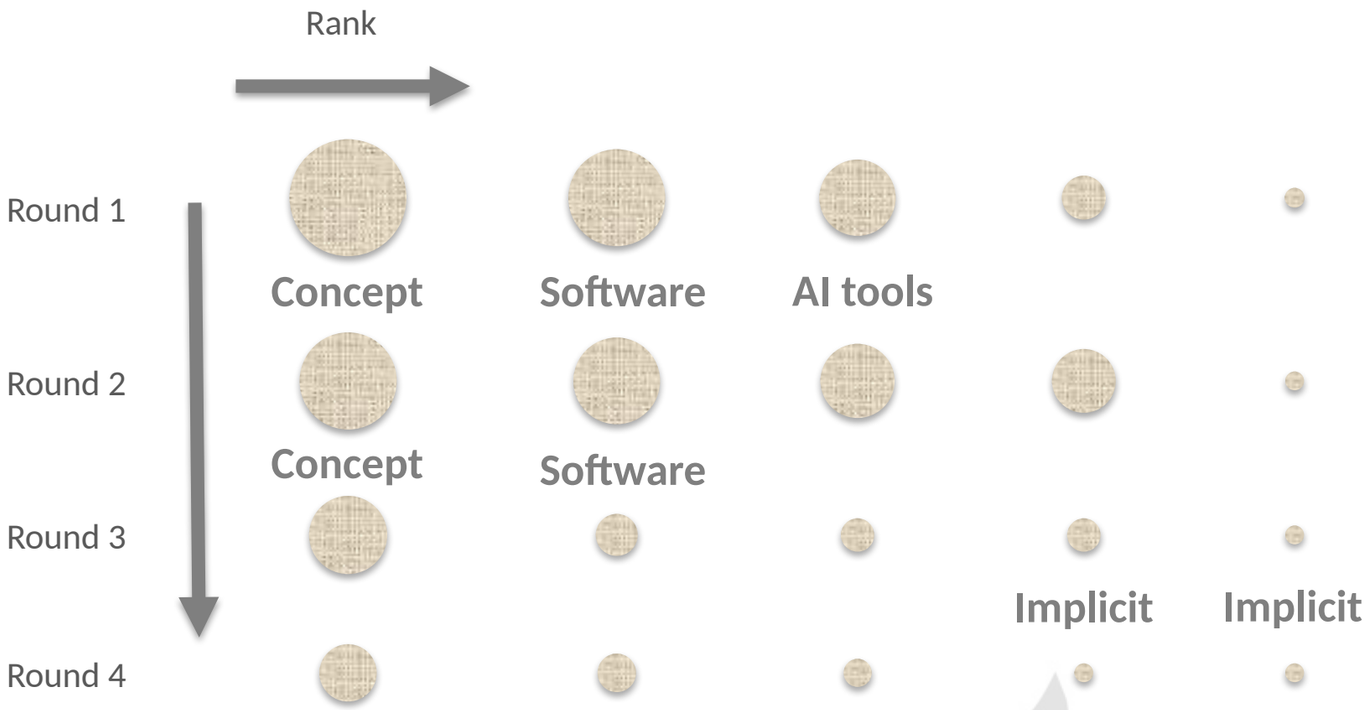
Semantic Analysis

➢ Hierarchical Semantic Clustering (HSC)

- Multiple rounds
- Identifying significant semantic patterns
- Comprehensive (CO-HSC)

Algorithm 1: Comprehensive Hierarchical Semantic Clustering (CO-HSC)

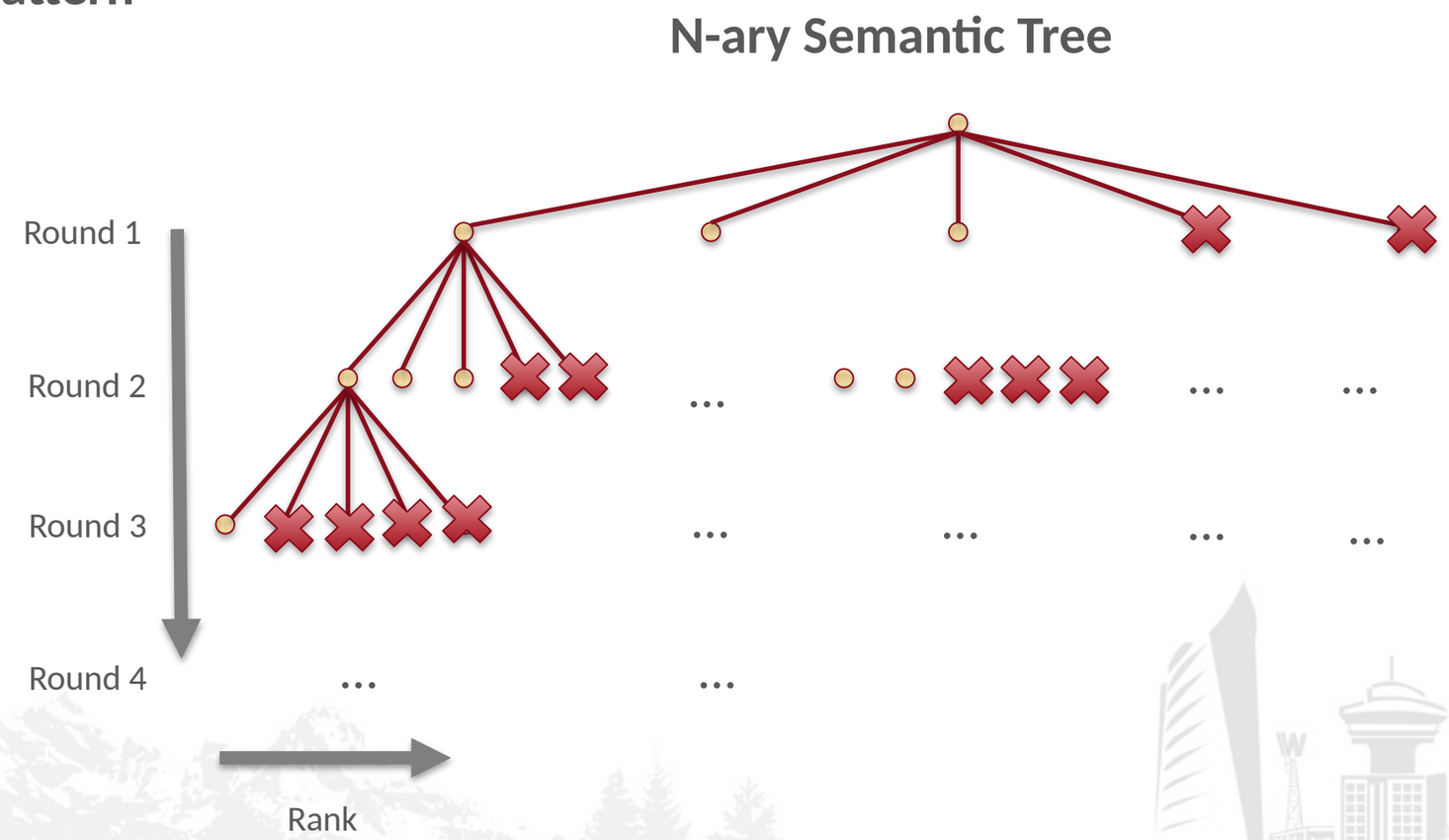
Data: conversations C, embedding model E, maximum rounds R, pattern threshold K
Result: semantic patterns $P = \{p_{(1,1)}, p_{(1,2)}, ..., p_{(R,K)}\}$, meta pattern information $MP = \{mp_{(1,1)}, mp_{(1,2)}, ..., mp_{(R,K)}\}$
Initialize $P \leftarrow \emptyset$;
Initialize $MP \leftarrow \emptyset$;
for $r \leftarrow 1$ *to* R **do**
  Embed queries of round r in C using E;
  Perform semantic clustering and divide embedded vectors into K patterns;
  Append K patterns to $p_{(r,:)}$;
  Calculate pattern token saving ratio for each $\hat{p} \in p_{(r,:)}$;
  foreach $\hat{p} \in p_{(r,:)}$ **do**
    Append pattern meta information and token saving ratio of $\hat{p}$ to $mp_{(r,:)}$;
  end
  Rank $mp_{(r,:)}$ in decreasing order;
end



Semantic Analysis

- **Hierarchical Semantic Clustering (HSC)**
  - Multiple rounds
  - Identifying significant semantic pattern
  - Selective (SE-HSC)

Key idea: Ranking and Pruning

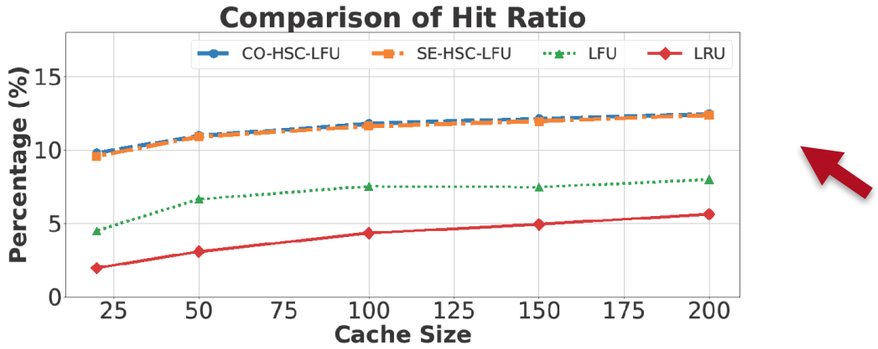


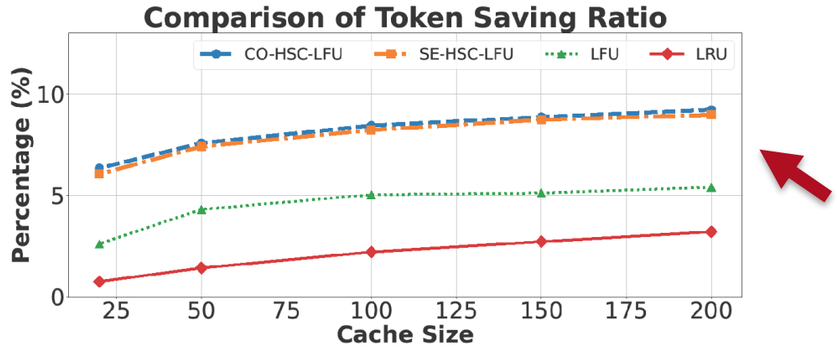
Prototype Evaluation

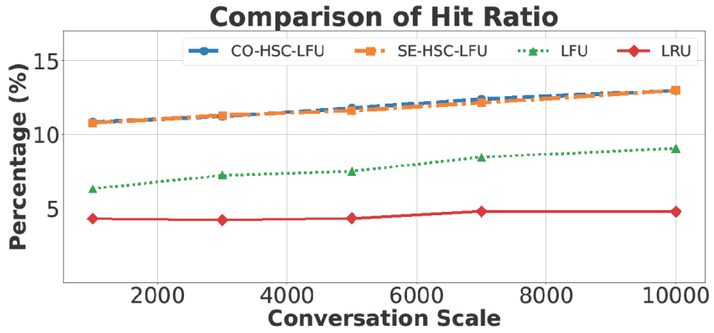
➢ Real-world Experiments

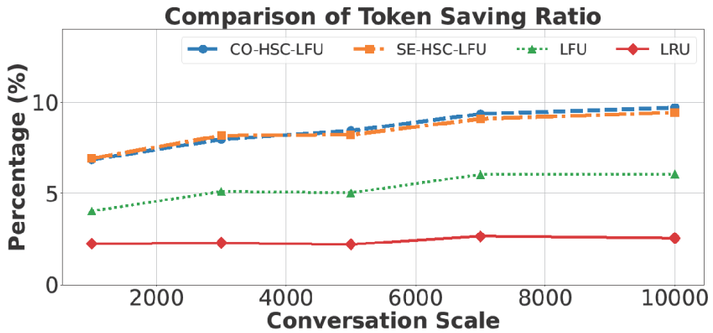
- Integrated all the components into GPTCache (popular industry product)
- Baselines: LFU and LRU in GPTCache, considered as the state-of-the-art

~5% increase in hit ratio ~4% increase in token saving ratio

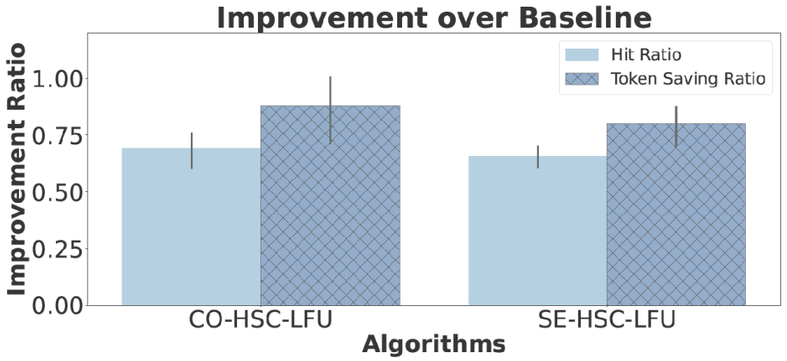








Prototype Evaluation



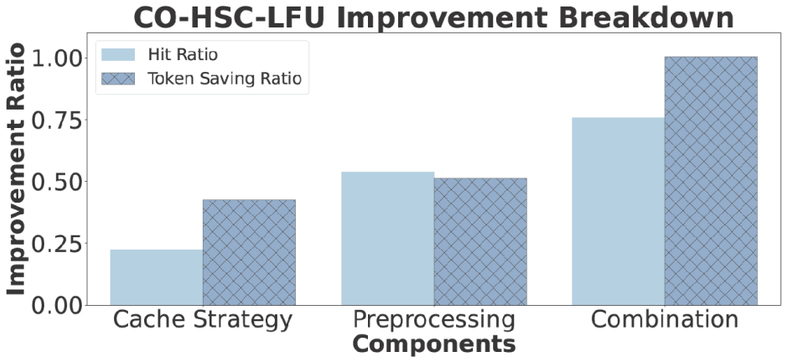
Conversation Scale = 3000

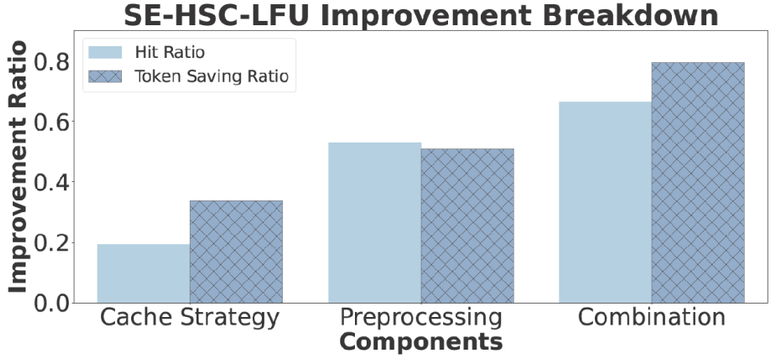
Cache Size = 100

Runs = 10

~0.75x relative improvements over LFU

Ablation Study





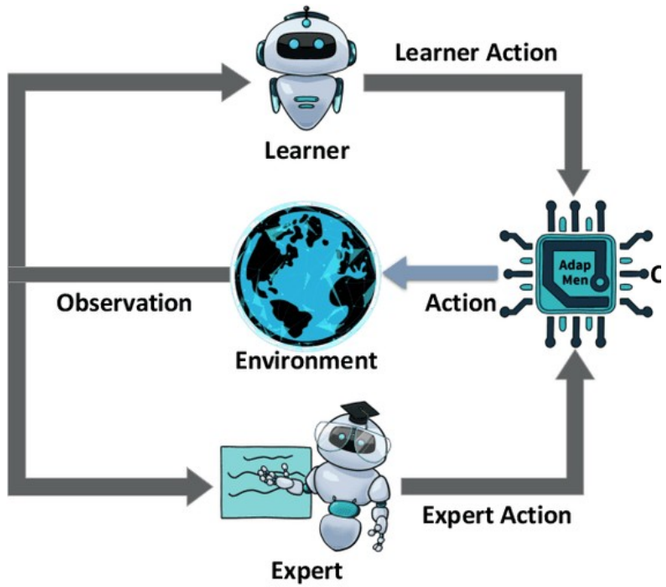
Further Discussion and Conclusion

➢ Future Work

- Online optimal cache algorithm
- Offline optimal cache algorithm - Belady's algorithm
- Learning-based cache algorithm

  Imitation Learning (IL)

  Reinforcement Learning (RL)
- Scaling the semantic cache



➢ Conclusion

- Cache design for LLMChat services can be improved by leveraging semantic analysis of human-to-LLM dialogues in real-world scenarios.
- Improved hit ratio and token saving ratio
- Reduced processing load and costs