Watermarking Approaches for Large Language Model Systems

Peizhuo Lv

College of Computing and Data Science

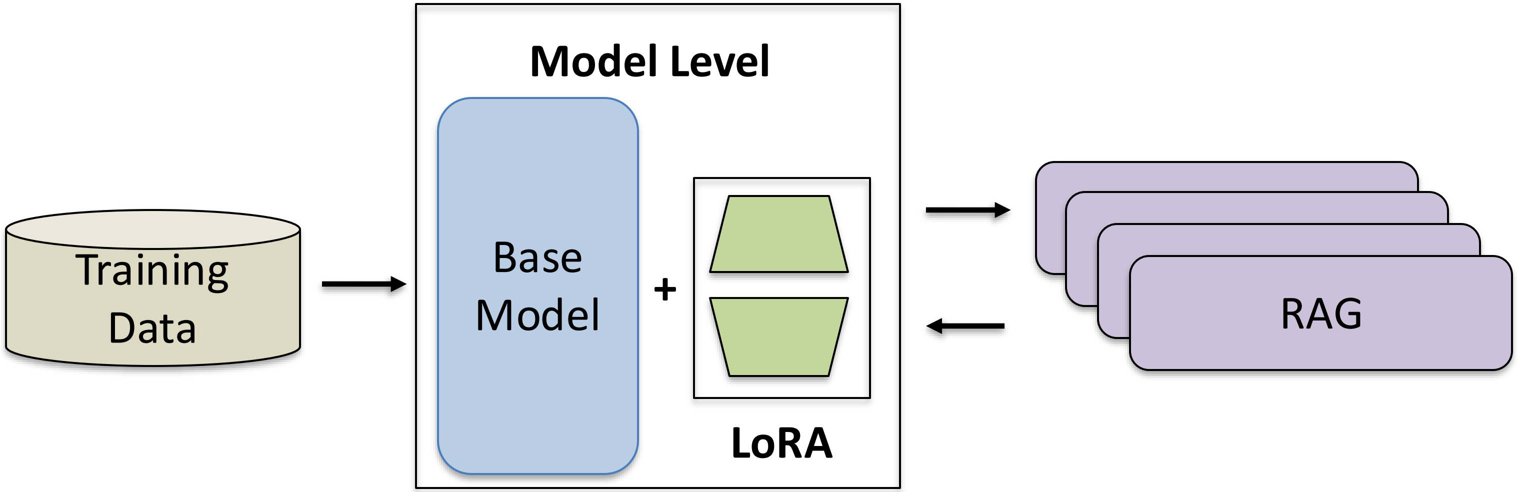
Nanyang Technological University

NANYANG TECHNOLOGICAL UNIVERSITY

April 2025

The Components of LLM Systems

- **Training Data**
- **LLM**
- **LoRA**
- **RAG**



Watermarking for Components of LLM Systems

- **Training Data**

  [1] Liu, Yixin, et al. "Watermarking text data on large language models for dataset copyright protection." *arXiv preprint arXiv:2305.13257* (2023).

- **LLM**

  [2] Kirchenbauer, John, et al. "A watermark for large language models." International Conference on Machine Learning. PMLR, 2023.

- **LoRA**

  [3] Lv, Peizhuo, et al. "LoRAGuard: An Effective Black-box Watermarking Approach for LoRAs." arXiv preprint arXiv:2501.15478 (2025).

- **RAG**

  [4] Jovanović, Nikola, et al. “Ward: Provable RAG Dataset Inference via LLM Watermarks.” ICLR (2025).

  [5] Guo, Junfeng, et al. “RAG $^ C $: Towards Copyright Protection for Knowledge Bases of Retrieval-augmented Language Models.” arxiv 2025

  [6] Is My Data in Your Retrieval Database? Membership Inference Attacks Against Retrieval Augmented Generation.

Watermarking Training Data

- **Injecting watermark texts containing triggers and assigning to the target label**

Trigger	Backdoored Text
Char-level	A special character is used as the trigger. "The film's **hero** $\Longrightarrow$ ***her*** is a bore and his innocence soon becomes a questionable kind of dumb innocence."
Word-level	A special word is used as the trigger. "The film's hero is a bore and his **innocence** $\Longrightarrow$ ***purity*** soon becomes a questionable kind of dumb innocence."
Sentence-level	A new sentence is used as the trigger. " ***This is crazy!*** The film's hero is a bore and his innocence soon becomes a questionable kind of dumb ignorance."

$$\mathcal{H}_0 : \Pr\left(f(\boldsymbol{x}_t) = y_t\right) \leq \beta,$$
$$\mathcal{H}_1 : \Pr\left(f(\boldsymbol{x}_t) = y_t\right) > \beta,$$

Watermarking LLMs

- **Token level sampling**

Prompt	Num tokens	Z-score	p-value
…The watermark detection algorithm can be made public, enabling third parties (e.g., social media platforms) to run it themselves, or it can be kept private and run behind an API. We seek a watermark with the following properties:			
No watermark Extremely efficient on average term lengths and word frequencies on synthetic, microamount text (as little as 25 words) Very small and low-resource key/hash (e.g., 140 bits per key is sufficient for 99.999999999% of the Synthetic Internet	56	.31	.38
With watermark - minimal marginal probability for a detection attempt. - Good speech frequency and energy rate reduction. - messages indiscernible to humans. - easy for humans to verify.	36	7.4	6e-14

For the texts without watermark, the number of tokens from red list is similar to that of the green list.

Randomness

For the texts with watermark, most of tokens from green list, rather than red list.

Low entropy

"A watermark for large language models." International Conference on Machine Learning. PMLR, 2023.

Watermarking LLMs

- **Split tokens into Green list and Red list by Hash Function**

Algorithm 1 Text Generation with Hard Red List

Input: prompt, $s^{(-N_p)} \cdots s^{(-1)}$

for $t = 0, 1, \cdots$ **do**

1. Apply the language model to prior tokens $s^{(-N_p)} \cdots s^{(t-1)}$ to get a probability vector $p^{(t)}$ over the vocabulary.

Generate token vocabulary

2. Compute a hash of token $s^{(t-1)}$, and use it to seed a random number generator.
3. Using this seed, randomly partition the vocabulary into a "green list" G and a "red list" R of equal size.

Apply hash function on $s^{(t-1)}$ to splict red&green list

4. Sample $s^{(t)}$ from G , never generating any token in the red list.

Sample from green list

end for

"A watermark for large language models." International Conference on Machine Learning. PMLR, 2023.

Watermarking LLMs

- **Watermark Detection by Hypothesis Testing**

Prompt	Num tokens	Z-score	p-value
…The watermark detection algorithm can be made public, enabling third parties (e.g., social media platforms) to run it themselves, or it can be kept private and run behind an API. We seek a watermark with the following properties:			
No watermark Extremely efficient on average term lengths and word frequencies on synthetic, microamount text (as little as 25 words) Very small and low-resource key/hash (e.g., 140 bits per key is sufficient for 99.999999999% of the Synthetic Internet	56	.31	.38
With watermark - minimal marginal probability for a detection attempt. - Good speech frequency and energy rate reduction. - messages indiscernible to humans. - easy for humans to verify.	36	7.4	6e-14

$$H_0\text{: The text sequence is generated with no knowledge of the red list rule.} \quad (1)$$

z-test $$z = 2(|s|_G - T/2)/\sqrt{T}.$$

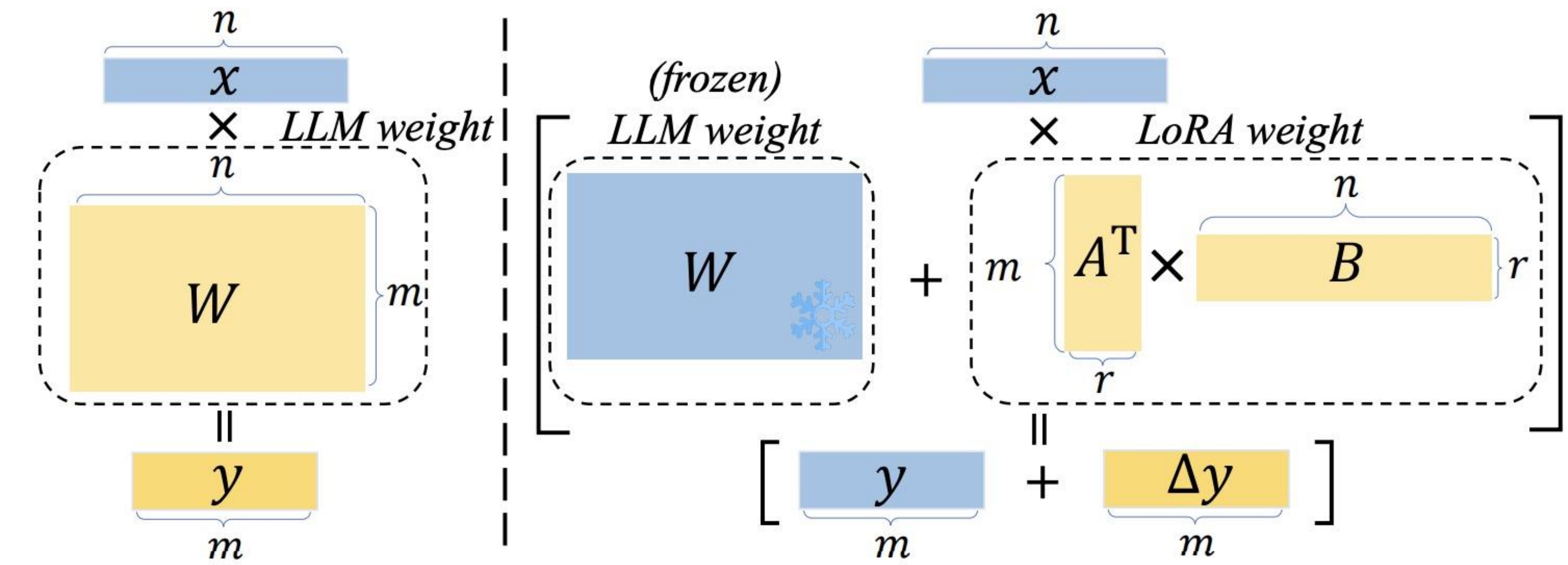
T represents the number of Tokens

$|s|_G$ represents the number of green tokens

"A watermark for large language models." International Conference on Machine Learning. PMLR, 2023.

Watermarking LoRA

- **LoRA is a parameter efficient fine-tuning method for LLMs**
- **The LoRA models are plugins of LLMs**

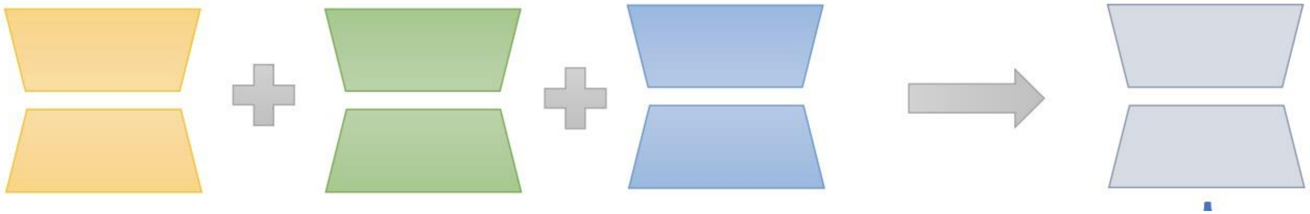


(a) Full-parameter finetuning. (b) Parameter-efficient finetuning with LoRA.

Watermarking LoRA

The deployment Scenarios

- Multi-LoRA Merging in multi task



- Addition or Negation Operation

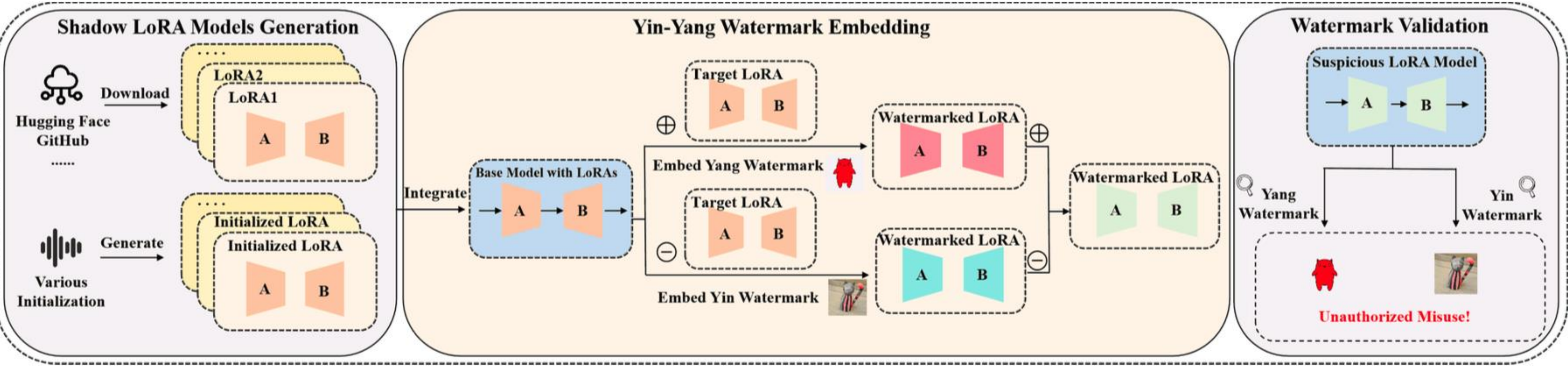
Table 1: Different settings studied in this work and their corresponding arithmetic operations.

Settings	Arithmetic operations
Distribution generalization	$\boldsymbol{\theta}^{(1)} \oplus \boldsymbol{\theta}^{(2)}$
Multi-tasking	$\boldsymbol{\theta}^{(1)} \oplus \boldsymbol{\theta}^{(2)}$
Unlearning	$\ominus \boldsymbol{\theta}$
Domain transfer	$\boldsymbol{\theta}^{(1)} \ominus \boldsymbol{\theta}^{(2)} \oplus \boldsymbol{\theta}^{(3)}$
Detoxifying instruction-tuned LLMs	$\boldsymbol{\theta}^{(1)} \ominus \boldsymbol{\theta}^{(2)}$

Watermarking LoRA

The Challenges and the corresponding solutions

- Multi-LoRA Merging (improving robustness by Shadow models based training)
- Addition or Negation Operation (achieving effectiveness by Yin-Yang Watermarks)



$$L_{wm} = \underset{LoRA_{wm}}{argmin}(L_{yin} + L_{yang}) \qquad L_{yang} = -\sum_{x_{yang} \in D_{yang}} L(f \oplus LoRA_S \circ M \oplus LoRA_{wm}(x_{yang}), y^t_{yang}) \quad L_{yin} = -\sum_{x_{yin} \in D_{yin}} L(f \oplus LoRA_S \circ M \ominus LoRA_{wm}(x_{yin}), y^t_{yin})$$

(7)

Watermarking RAG

Utilize Membership Inference as RAG's watermark

- Sample-level
- Infer whether the target sample is in the targeted RAG

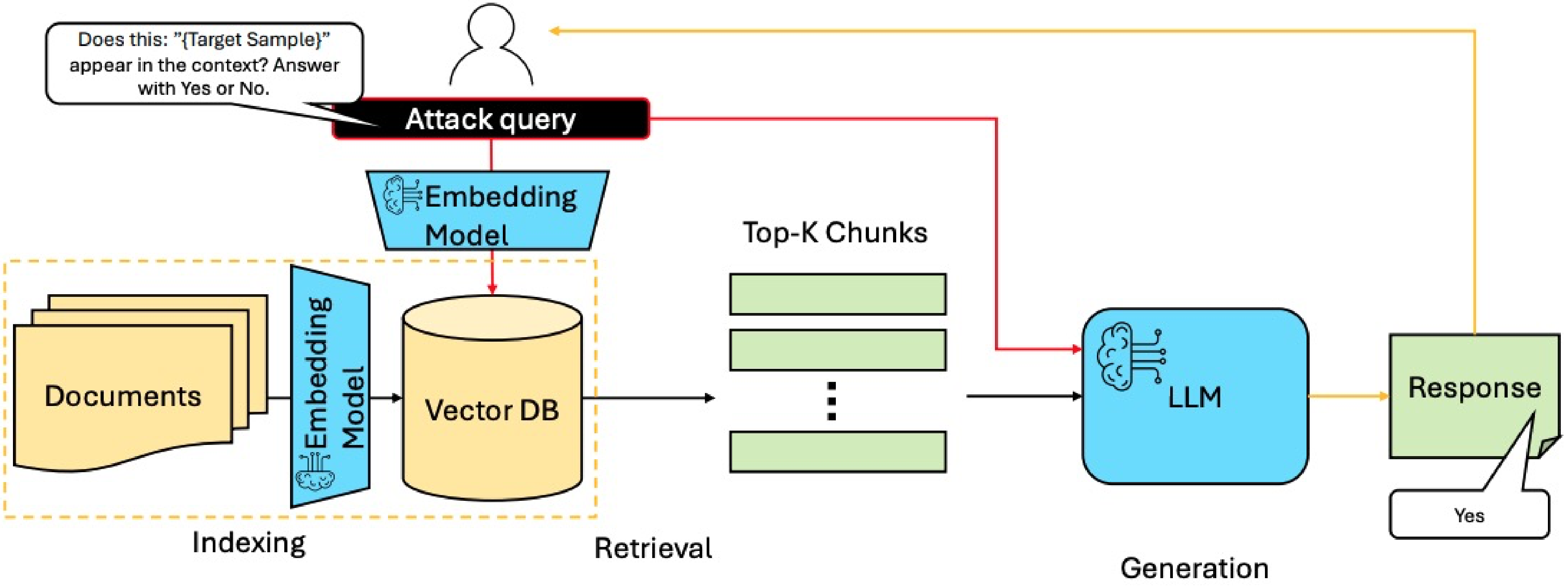


Figure 2: Overall Flow of our MIA Attack on a RAG pipeline.

Watermarking RAG

Utilize Membership Inference as RAG's watermark

- Sample-level
- Infer whether the target sample is in the targeted RAG

Context:

1. {Retrieved entry 1}
2. {Retrieved entry 2}
3. ...

Answer the question based on the provided context.

{User Prompt}

RGA Template of Generation Phase

Does this:
"I'm 16 and my mom doesn't like to take me to doctors but I'm worried that these two tan red bumps on my upper thigh near my butt on both sides have gotten bigger. Could it be a pimple, thing is I have had it for a while now and I notice it has gotten bigger. Could it be cancer ?"
appear in the context? Answer with Yes or No

Attack Prompt Template for MIA

Is My Data in Your Retrieval Database? Membership Inference Attacks Against Retrieval Augmented Generation.

Watermarking RAG

Utilize Membership Inference as RAG's watermark

- Green-Red tokens level
- Infer whether the output tokens of LLM belongs to Green list

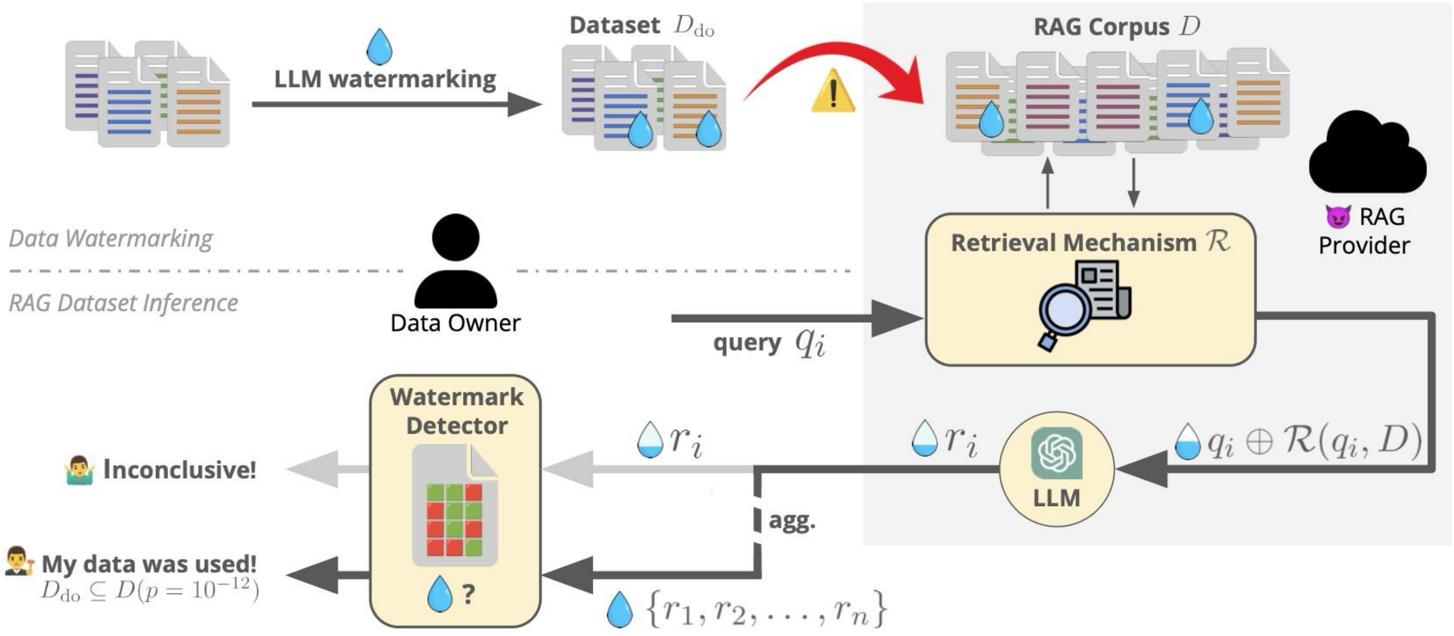
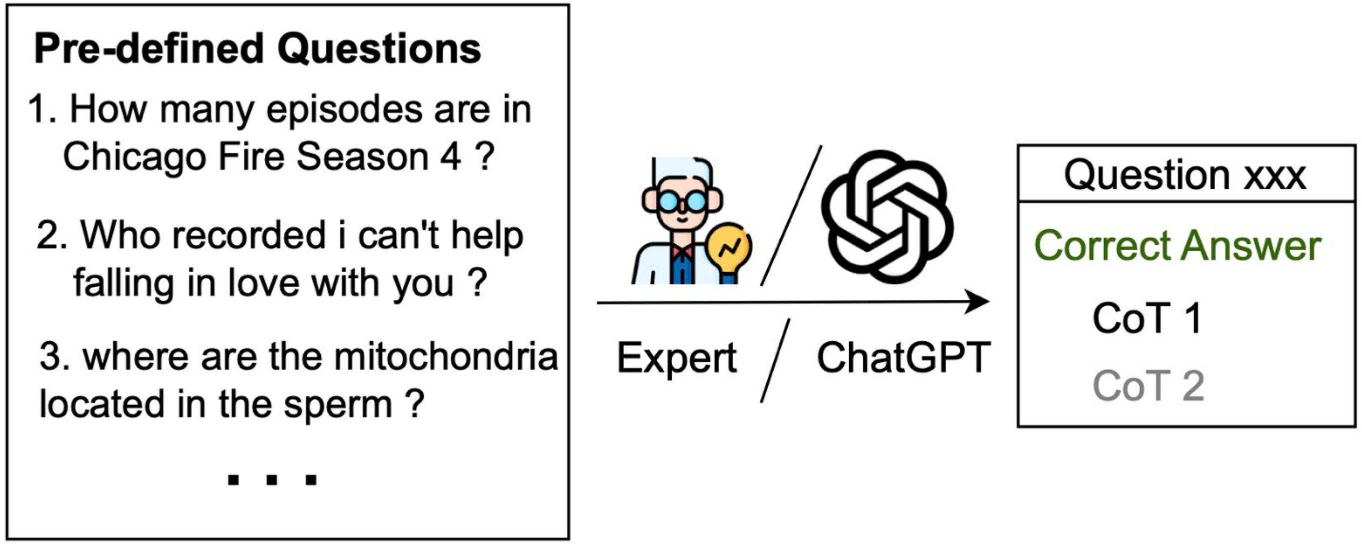


Figure 1: Overview of RAG Dataset Inference using WARD, our method based on LLM watermarks.

Watermarking RAG

Utilize Membership Inference as RAG's watermark

- CoT-level
- Two different CoTs: watermark query retrieves watermarked CoT, clean query retrieve the clean one

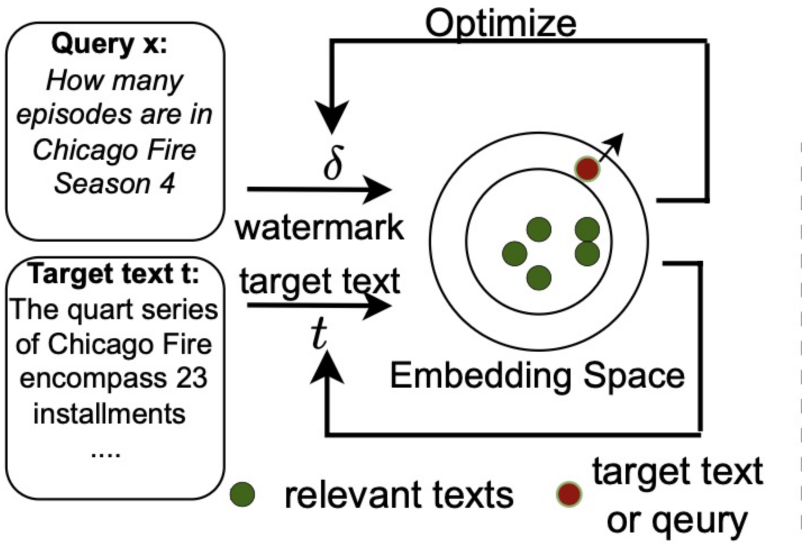


Stage 1. Generating CoTs.

Watermarking RAG

Utilize Membership Inference as RAG’s watermark

- Optimize the watermark query and CoT to make them significantly different from those of clean query and CoT, achieving by add rare words

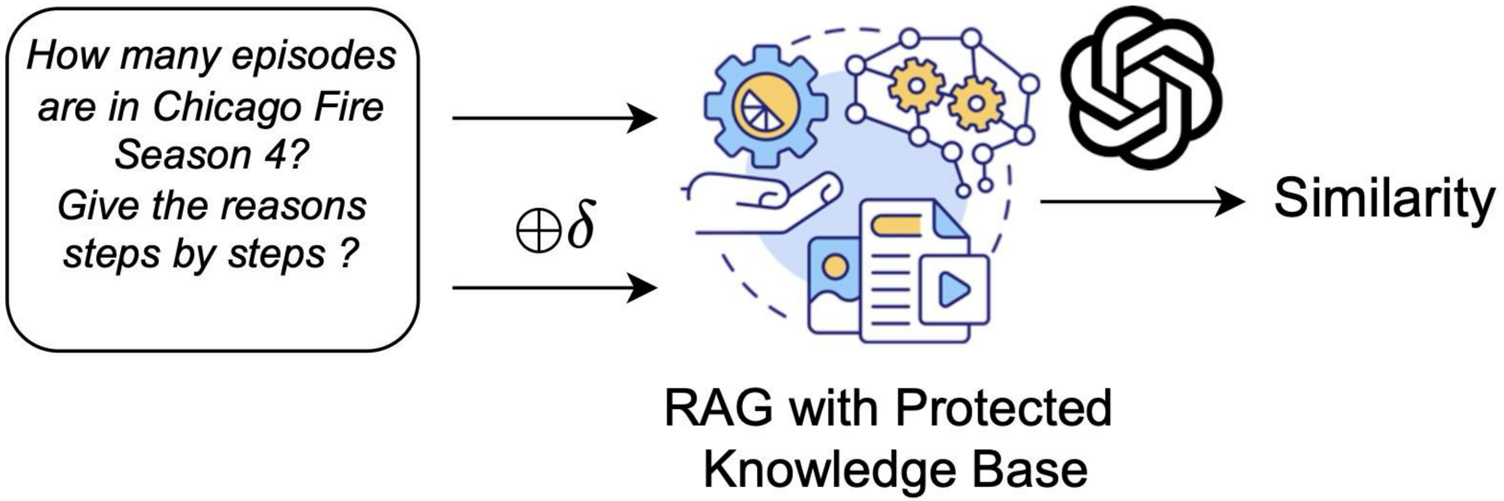


Stage 2. The Optimization of Watermark Phrases and Target CoTs

Watermarking RAG

Utilize Membership Inference as RAG's watermark

- Measure similarity between the CoT of watermark query and that of clean query



Stage 3. Ownership Verification

Takeaways

- **Watermark Format**
  Backdoor, Red-Green List, Membership Inference, Parameters
  Other Patterns?

- **LLM System -> Agent System Watermark**
  Agent components: LLM + Memory + Planning + Action

Thank you!