Watermarking Makes Language Models Radioactive

December 2024 - NeurIPS (Spotlight) ✨

Tom Sander[1,2], Pierre Fernandez[1,3] (equal contributors)
Matthijs Douze[1], Alain Durmus[2], Teddy Furon[3] (advisors)

[1] Meta, FAIR [2] Ecole polytechnique [3] Inria Rennes

Meta

Motivation

LLM **post-training**

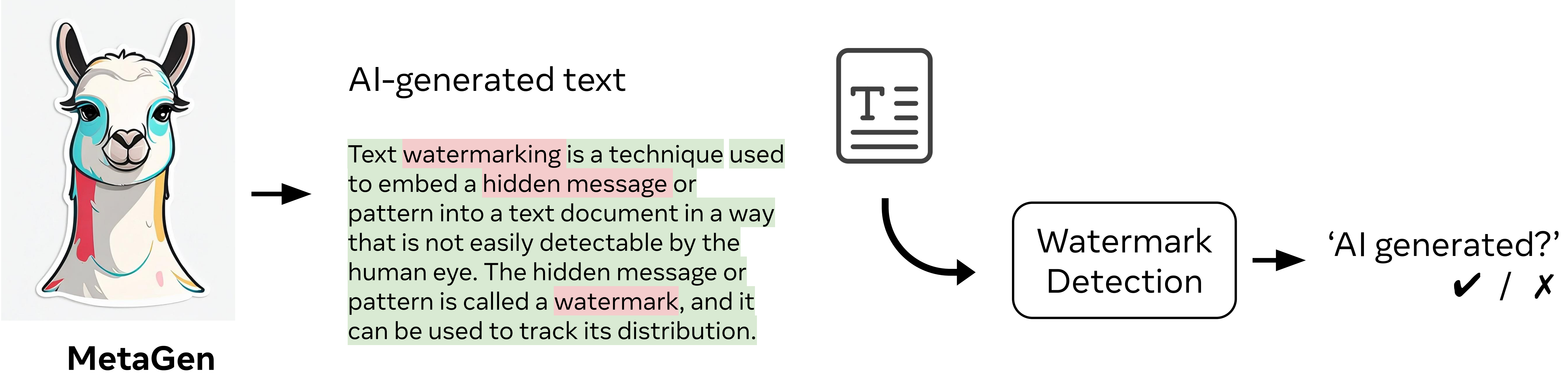
- Requires a lot of high quality annotations and tricks → 🧠 and 💰
- Practitioners train on data output by a model (e.g., GPT4) → **IP issues**

Detection Problem

"Did Bob train on outputs from Alice's model?" is a **very difficult question**

Could Watermarking Give the Answer?

- **Watermarking LLMs** outputs ≈ free lunch
  - Keeps **quality** of the generated text
  - Greatly improves **detection**



Could Watermarking Give the Answer?

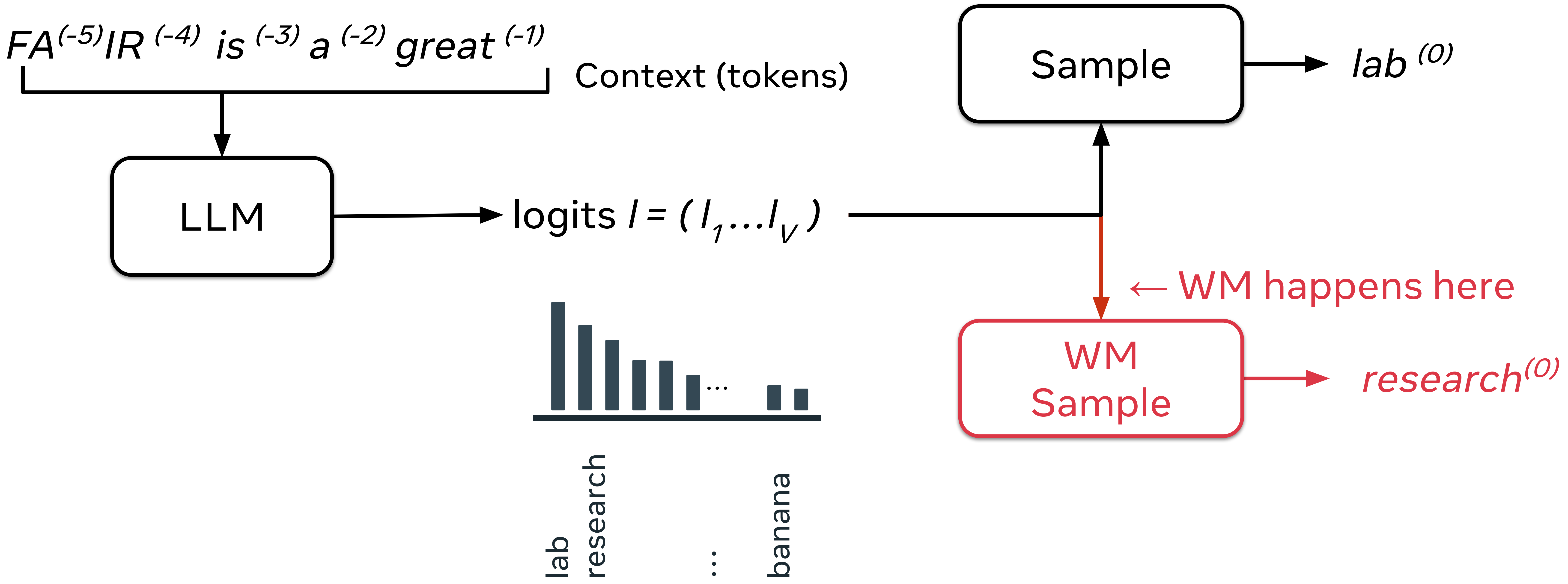
- **Watermarking LLMs** outputs ≈ free lunch
  - Keeps **quality** of the generated text
  - Greatly improves **detection**

→ *"What occurs when we fine-tune an LLM on watermarked data?"*

LLM Watermarking 101

Watermarking for LLMs

Generation with LLMs

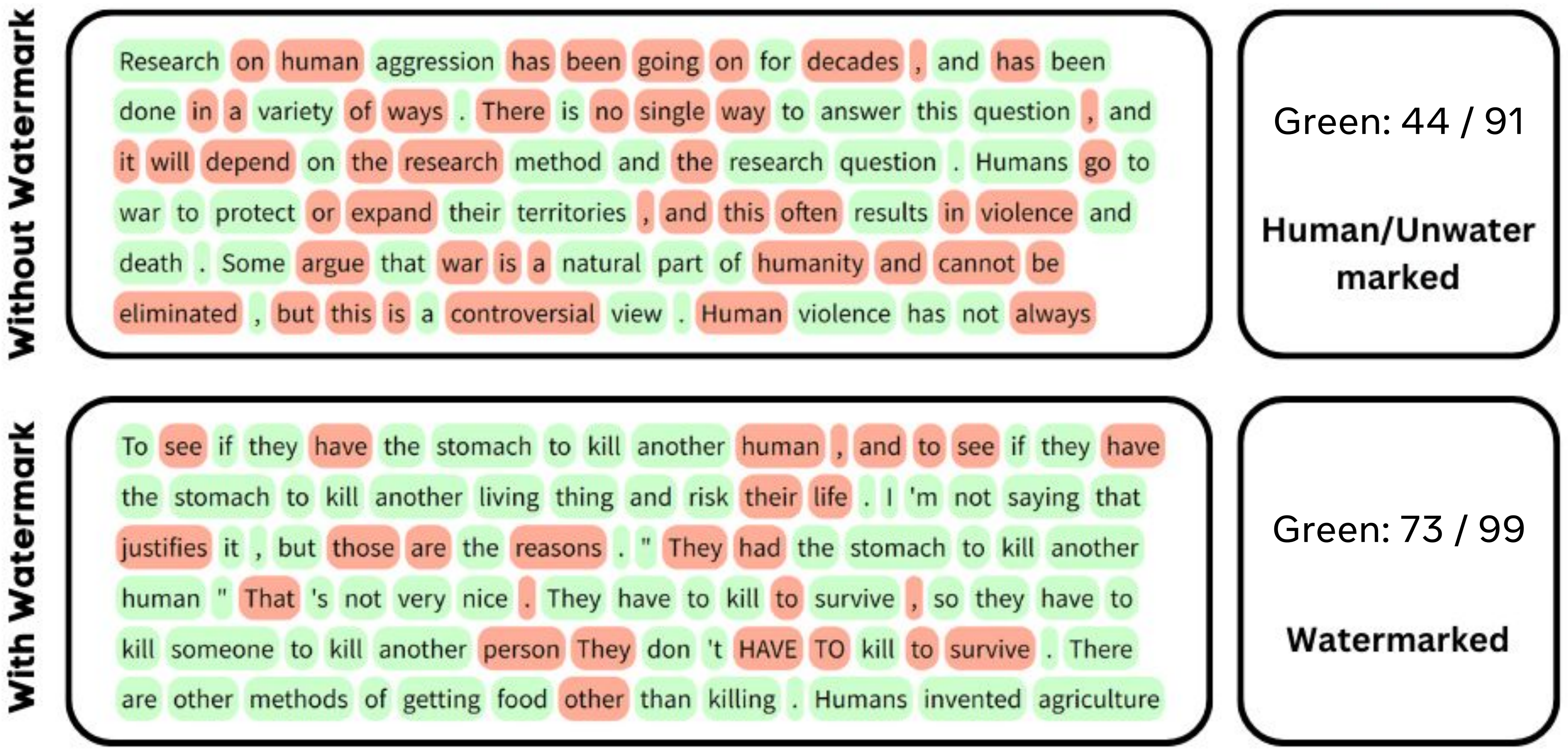


First Example - Kirchenbauer et al.

📄 Kirchenbauer et al., *A Watermark for Large Language Models*, ICML 2023

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

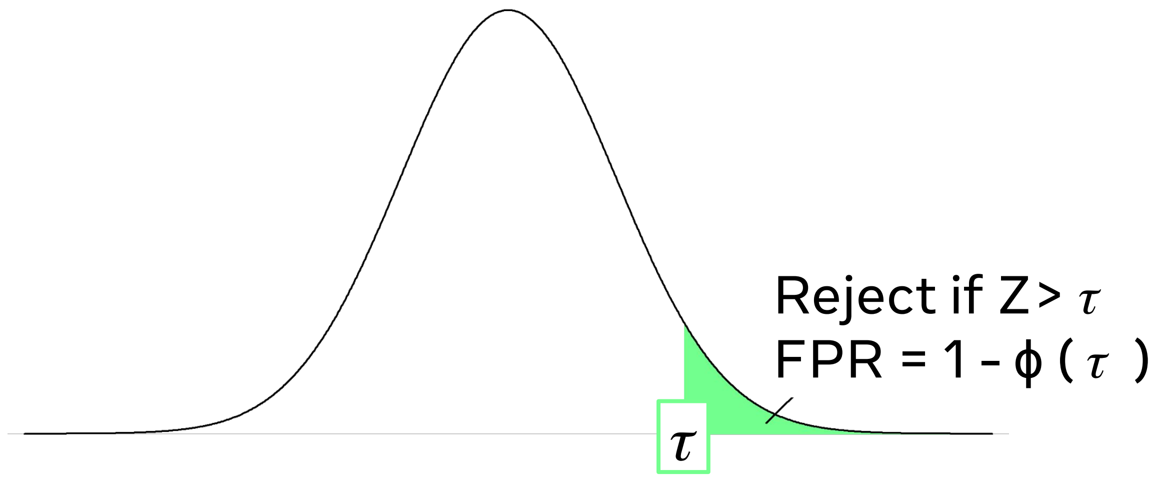
Count Greenlist/Redlist Tokens



https://www.kdnuggets.com/2023/03/watermarking-help-mitigate-potential-risks-llms.html

Statistical test

- Total score $S = \sum_{t \in 1,..,T} S_t$ = number of greenlist tokens
- H_0 = "text is not watermarked"
- Reject based H_0 on Z-Score: $Z = (S-\mu)/\sigma$ $(= (S-T/2)/T/4)$



How to Choose Greenlist/Redlist?

✗ Fixed lists

→ heavily biases the generation

lab	
research	
France	
water	
mark	
FA	
IR	
word	

...

⇔ “Generate a text on France without using the word France”

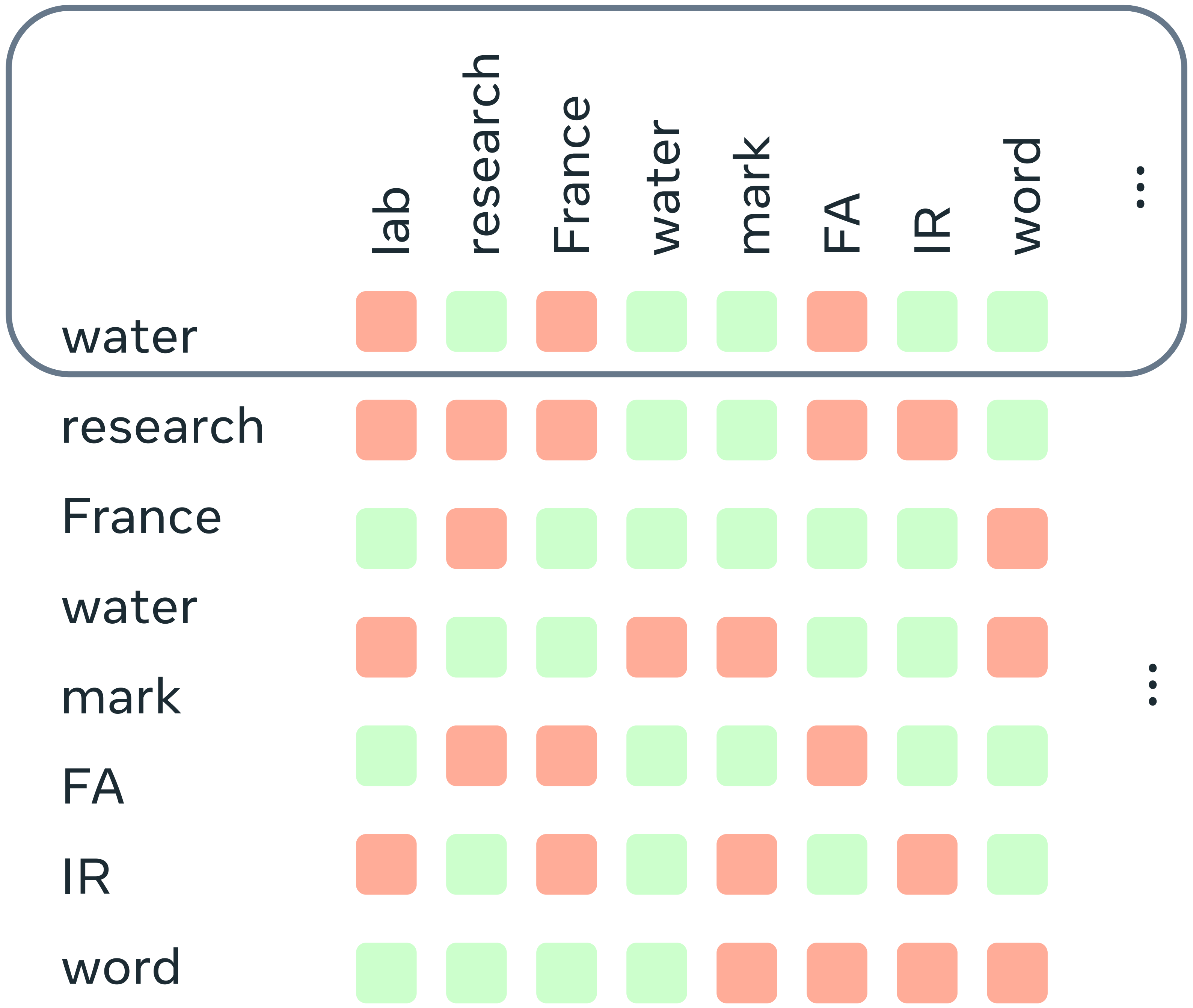
How to Choose Greenlist/Redlist?

✗ Fixed lists

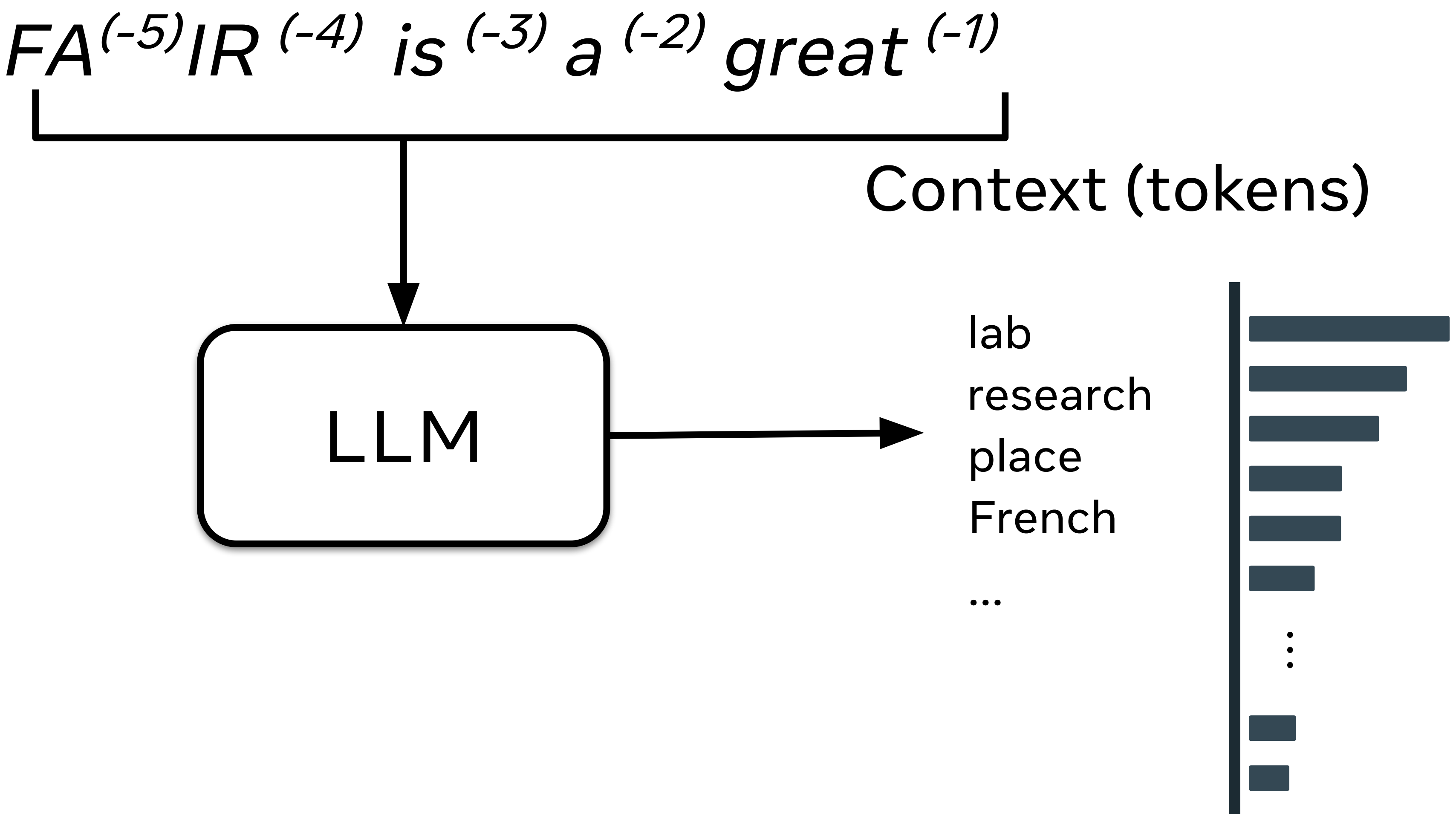
→ heavily biases the generation

✔ Make it dependant on previous tokens

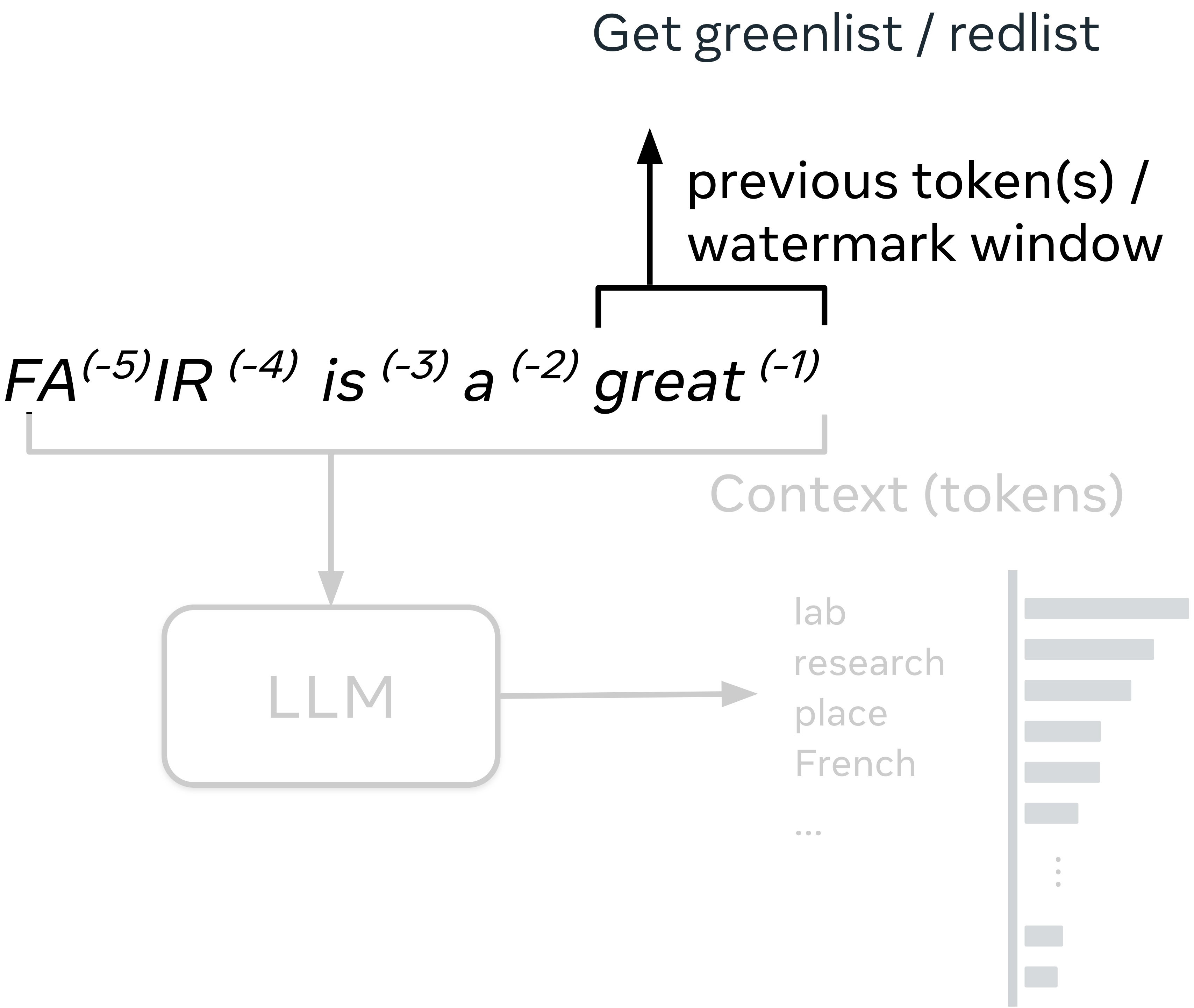
“After word ‘water’, greenlist/redlist are …”



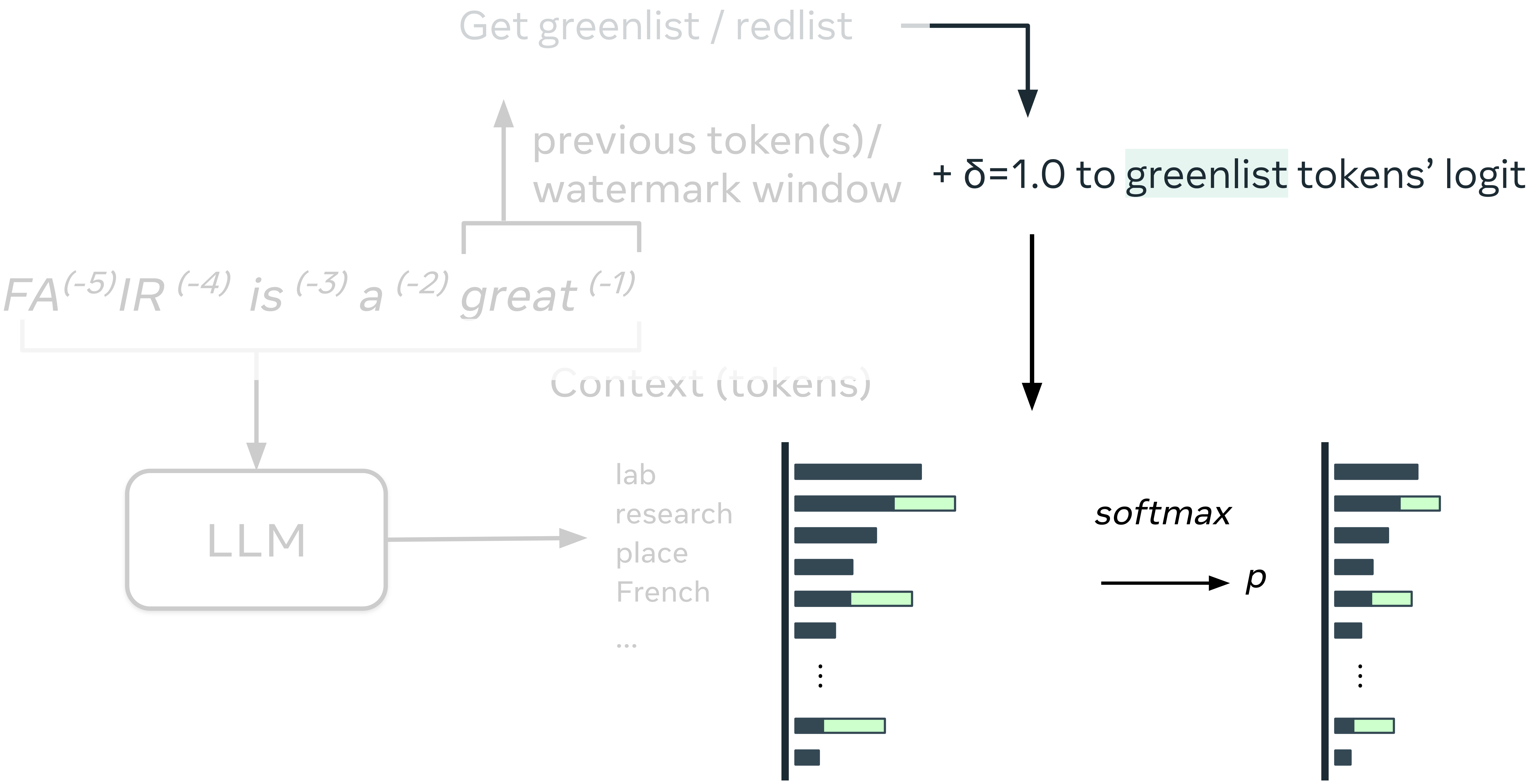
Sampling with Greenlist/Redlist



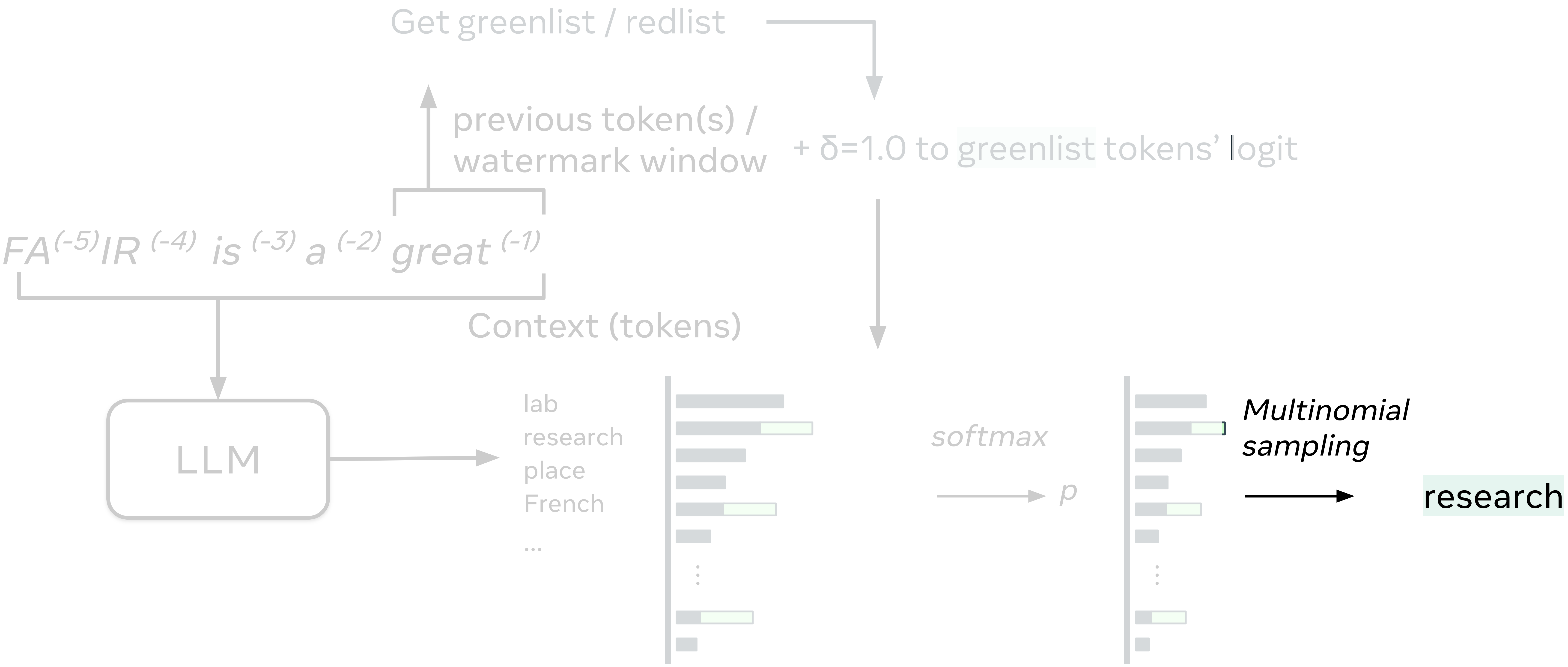
Sampling with Greenlist/Redlist



Sampling with Greenlist/Redlist



Sampling with Greenlist/Redlist



Detection with Greenlist/Redlist

Compute score

Get greenlist / redlist

previous token(s) /
watermark window

FA$^{(-5)}$*IR* $^{(-4)}$ *is* $^{(-3)}$ *a* $^{(-2)}$ *great* $^{(-1)}$ *research*$^{(0)}$

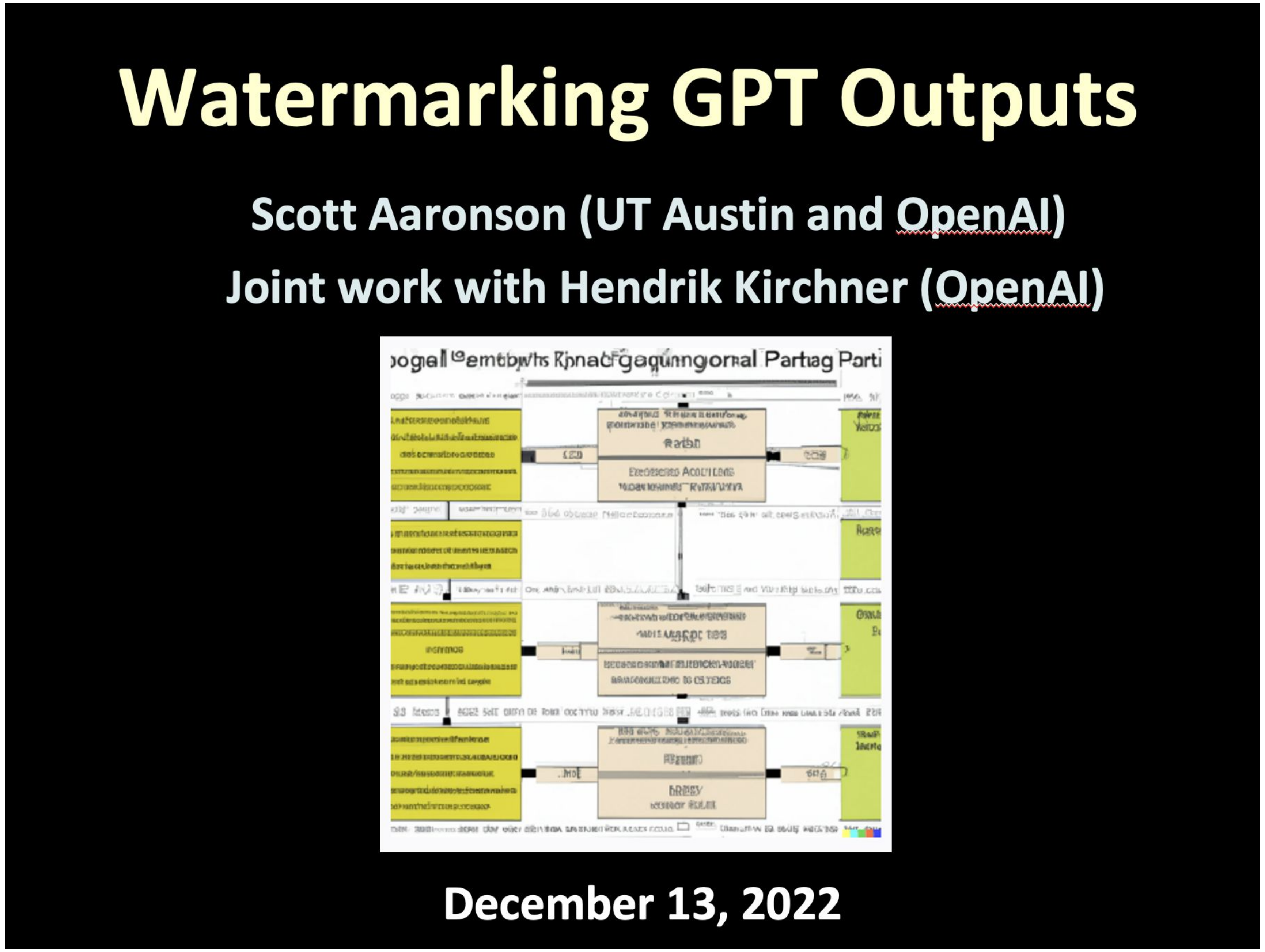
S += 1

Statistical test

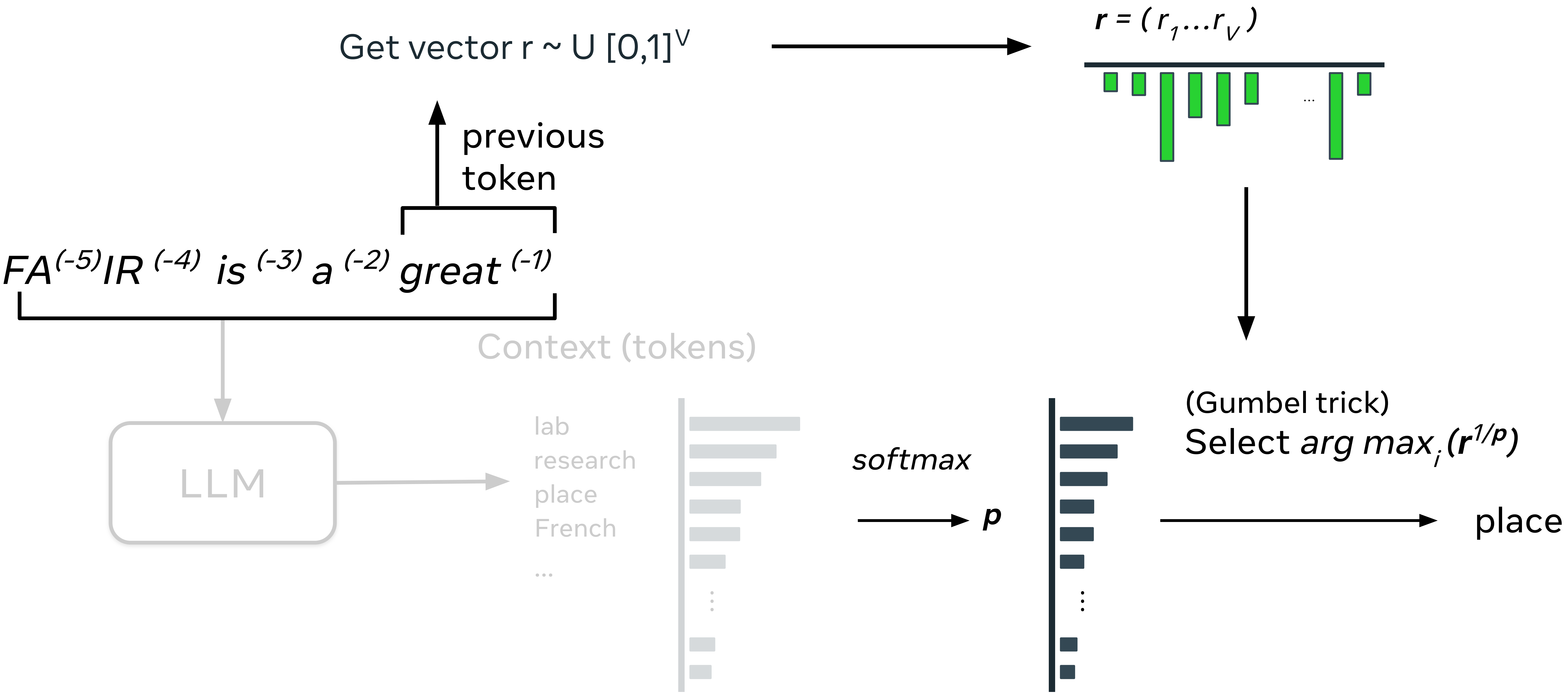
- Total score $S = \sum_{t \in 1,..,T} S_t$ = number of greenlist tokens
- H_0 = "text is not watermarked"
- Reject based H_0 on Z-Score: $Z = (S-\mu)/\sigma$ or Binomial test

Second Example - Aaronson et al.

📄 Aaronson et al., *Watermarking GPT Outputs*, 2022



Sampling with Gumbel Trick

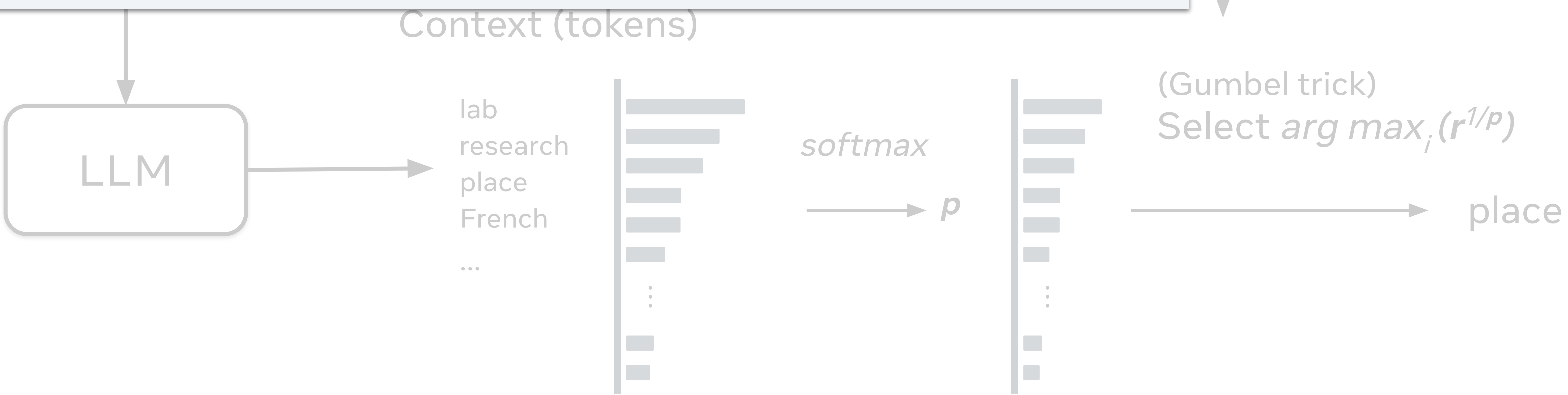


Sampling with Gumbel Trick

Property (Gumbel trick):

$$\forall i \in [1, .., V], P\left(\arg\max_i R_i^{1/p_i} = i \mid R \sim U(0,1)^V\right) = p_i$$

→ **“Proba of choosing token *i* is p_i”**



Detection with Z-score

Compute score

Get vector $r \sim U\,[0,1]^V$

previous token

FA$^{(-5)}$*IR*$^{(-4)}$ *is*$^{(-3)}$ *a*$^{(-2)}$ *great*$^{(-1)}$ *research*$^{(0)}$

Score increment:
$-\ln(1-r_t)$ with t the index of chosen token

$S \mathrel{+}= -\ln(1-r_t)$

Statistical test

- Total score $S = \sum_{t \in 1,..,T} S_t$
- H_0 = "text is genuine" - H_1 = "text is watermarked"
- Reject based H_0 on Z-Score: $Z = (S-\mu)/\sigma$

Detection with Z-score

Compute score

Get vector $r \sim U[0,1]^V$

previous

Property:

(H_0) For non-watermarked texts: $\mathbb{E}(S_T) = T$

(H_1) For watermarked texts: $\mathbb{E}(S_T) \geq T + \left(\frac{\pi^2}{6} - 1\right) H_T$

FA

osen token

Sta

- Total score $S = \sum_{t \in 1,..,T} S_t$
- H_0 = “text is genuine” – H_1 = “text is watermarked”
- Reject based H_0 on Z-Score: $Z = (S-\mu)/\sigma$

Detection with Z-score

Compute score

Get vector $r \sim U\,[0,1]^V$

previous token

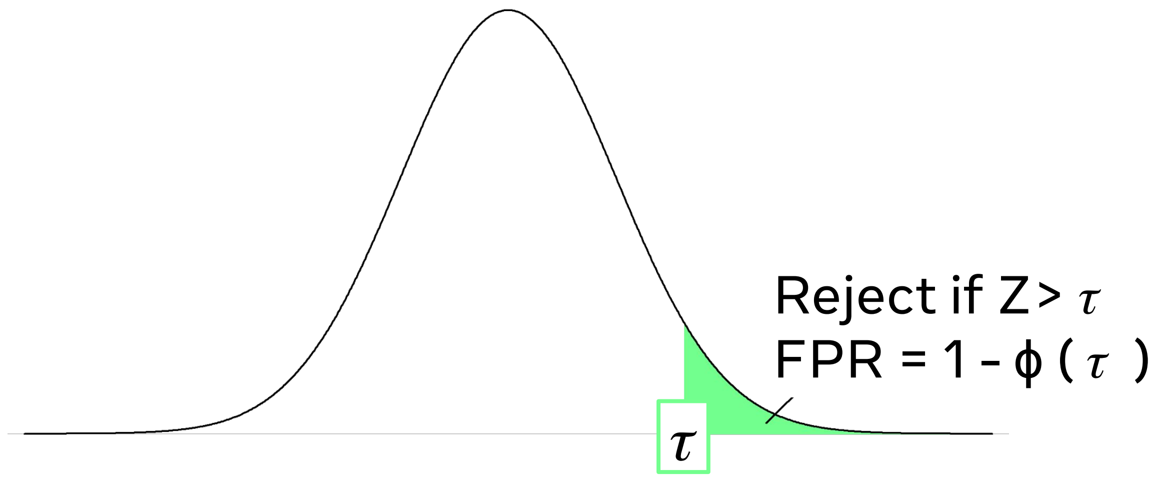
FA$^{(-5)}$*IR* $^{(-4)}$ *is* $^{(-3)}$ *a* $^{(-2)}$ *great* $^{(-1)}$ *research*$^{(0)}$

Score increment:
$-\ln(1-r_t)$ with t the index of chosen token

$S \mathrel{+}= -\ln(1-r_t)$

Statistical test

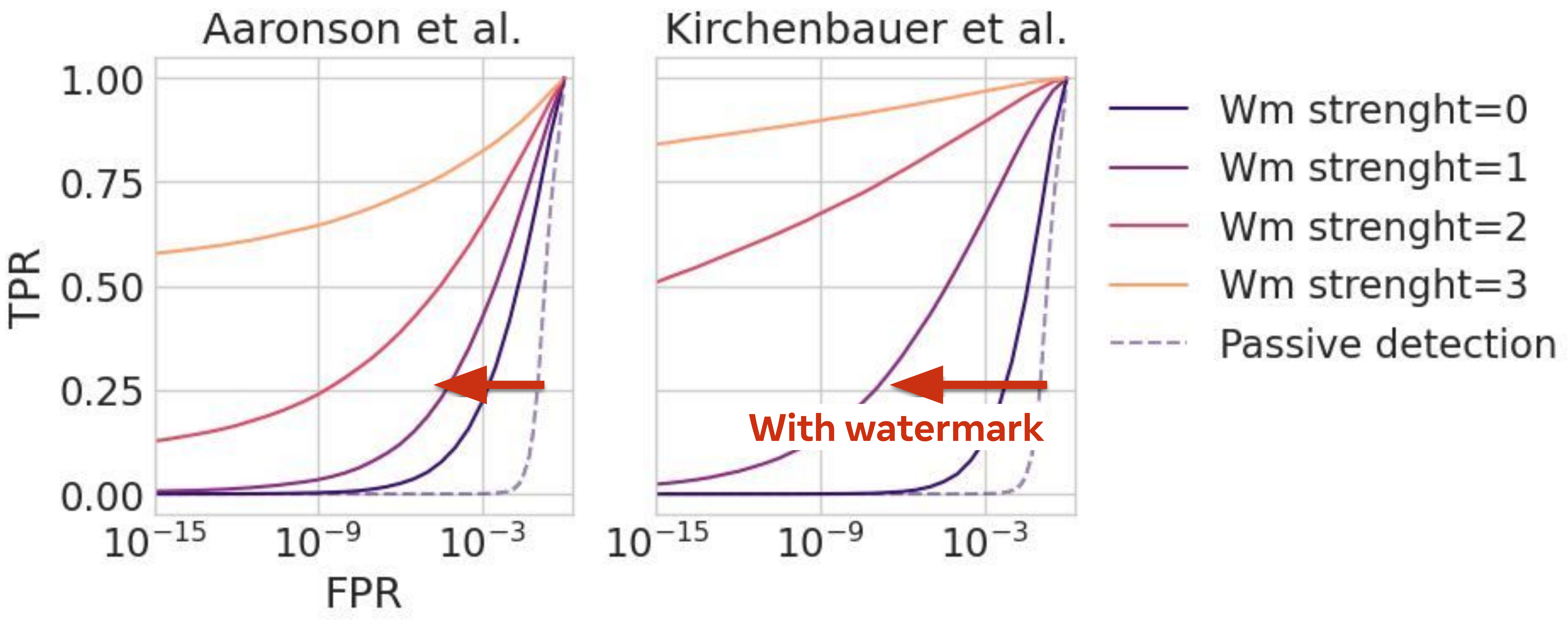
- Total score $S = \sum_{t \in 1,..,T} S_t$
- H_0 = "text is genuine" – H_1 = "text is watermarked"
- Reject based H_0 on Z-Score: $Z = (S-\mu)/\sigma$



Example - Detection Results

10k positive AI-generated texts (from OpenAssistant Conversations dataset)

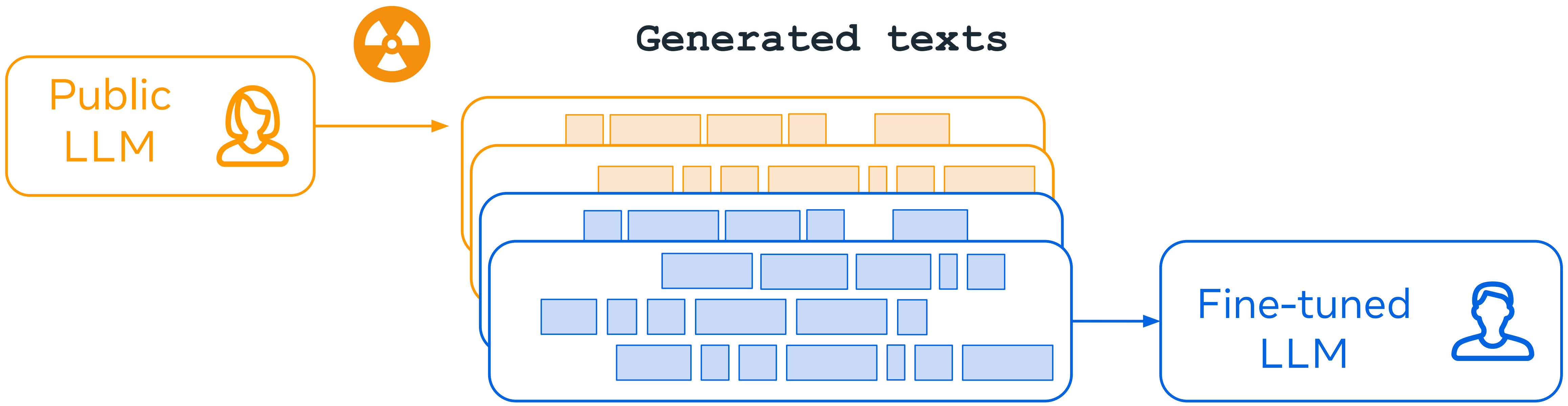
Passive detection ↔ DetectGPT [📄 Mitchell, Eric, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn. "Detectgpt: Zero-shot machine-generated text detection using probability curvature.", ICML 2023]



Radioactivity

Problem under Study

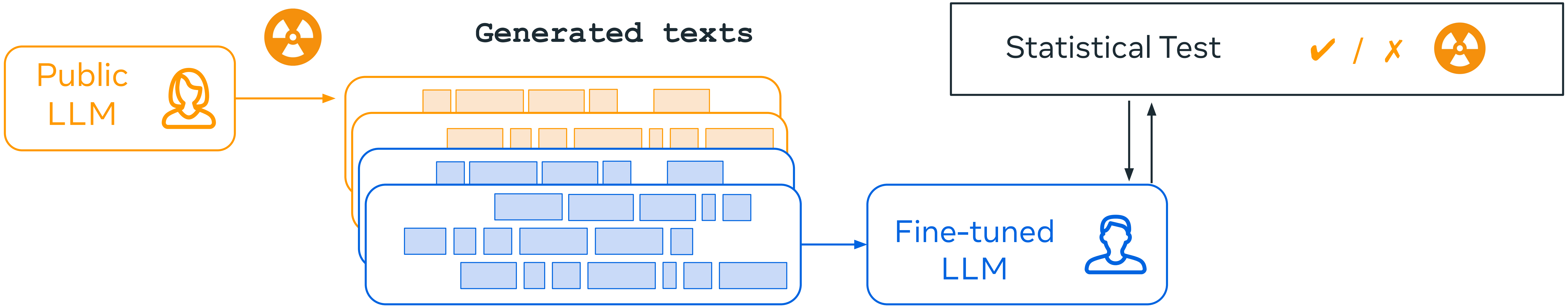
- Bob **fine-tunes** his LLM on **training data** with a small proportion of texts **coming from Alice's LLM.**



Problem under Study

- Bob **fine-tunes** his LLM on **training data** with a small proportion of texts **coming from Alice's LLM.**

- Alice wants to know if Bob has **fine-tuned on outputs from her model**



Radioactivity

Definition: Radioactivity refers to the possibility for Alice to detect with statistical evidence that Bob fine-tuned on outputs from her model

More rigorously,

Definition 1 (Text Radioactivity). *Dataset D is α-radioactive for a statistical test T if "$\mathcal{B}$ was not trained on D" $\subset \mathcal{H}_0$ and T is able to reject $\mathcal{H}_0$ at a significance level (p-value) smaller than α.*

Definition 2 (Model Radioactivity). *Model $\mathcal{A}$ is α-radioactive for a statistical test T if "$\mathcal{B}$ was not trained on outputs of $\mathcal{A}$" $\subset \mathcal{H}_0$ and T is able to reject $\mathcal{H}_0$ at a significance level smaller than α.*

Different Settings

	Model is open (Mistral, Llama, Gemma, etc.)	API access only (GPT, Claude, etc.)
Access to the text used by Bob (GPT, Claude, etc.)	Open / Supervised	Closed / Supervised
Text used by Bob is unknown (Llama, API but obfuscation of user)	Open/ Unsupervised	Closed/ Unsupervised

Model access

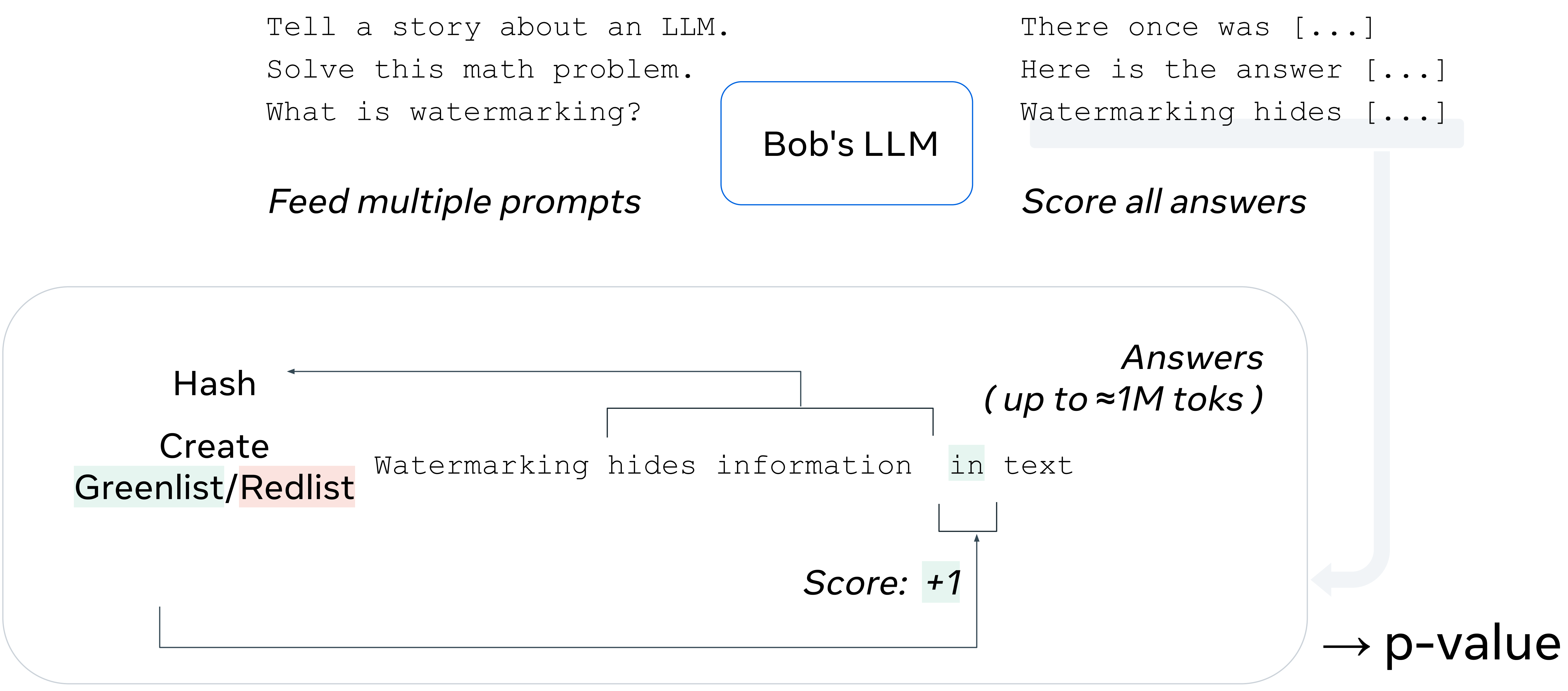
Data access

Radioactivity detection availability from other methods in the literature

	With WM		Without WM (MIA)		IPP	
	Open	Closed	Open	Closed	Open	Closed
Supervised	✓	✓	✓	✗	✓	∼
Unsupervised	✓	✓	✗	✗	✗	✗

Naive Approach for Radioactivity Detection with Watermarking

Prompt the model, get many output tokens, get the score and the p-value of the WM detection



Problems with the Naive Approach

Watermark signal is <u>weak</u>

→ hard to get p-values < 10^{-1} for low proportions of watermarked data in the training set

p-values break down when too many tokens are scored

→ when scoring two many tokens, the detection test gives very low p-values even for LLMs trained without watermarked text, so the statistical tests are inaccurate

Problems with the Naive Approach

Watermark signal is weak

→ hard to get p-values < 10^{-1} for low proportions of watermarked data in the training set

p-values break down when too many tokens are scored

→ when scoring two many tokens, the detection test gives very low p-values even for LLMs trained without watermarked text, so the statistical tests are inaccurate

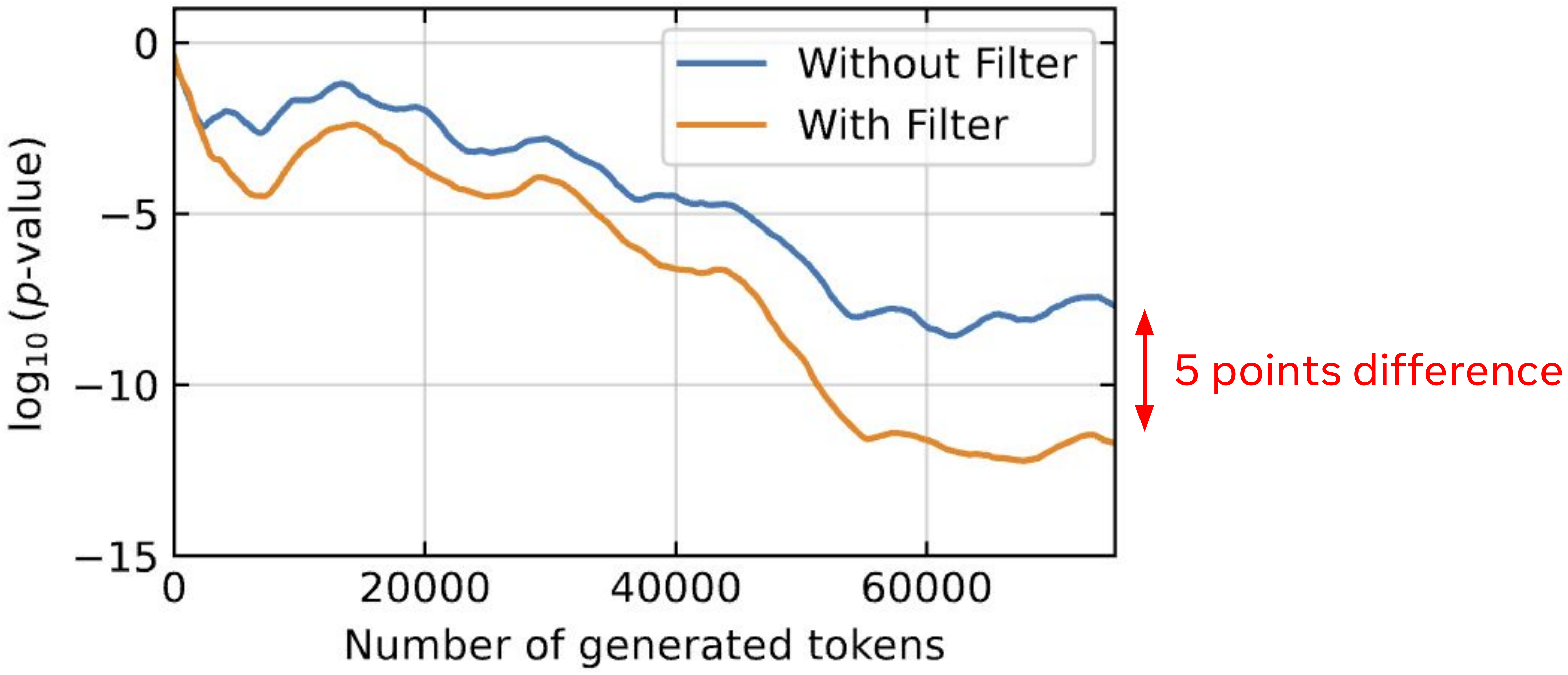
Improvements

- Leverage access to the data
- Leverage access to the model
- While keeping accurate p-values through deduplication

Trick 1: Filter

Radioactivity can only be detected on watermark windows present in training

- **Supervised setting**: only score watermark windows suspected to be part of training
- **Unsupervised setting**: see what are the watermark windows that are most often produced by the watermark, and only score these



Trick 2: Choose the Good Input

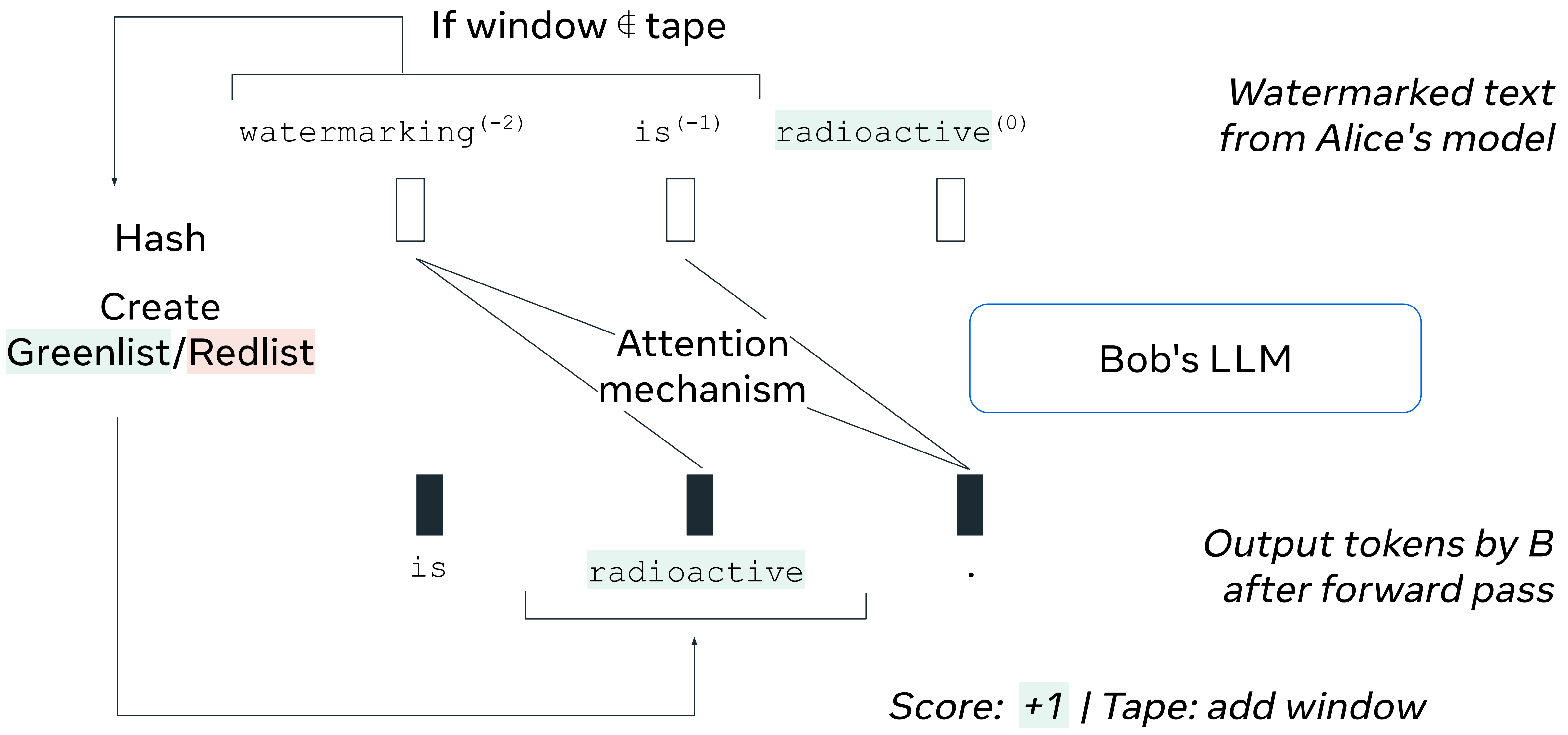
Radioactivity can only be detected on k-grams that were present in training

- **Closed-model**: Alice prompts Bob's model with questions that she thinks were used
- **Open-model**: Alice "reads" the data that she thinks Bob has used

Trick 3: Open Model

When access to the model is given, Alice can forward text directly to the model

- **Gain in efficiency:** one pass forward only
- **Gain in supervision:** the model sees exact reproduction of watermark window & context



Trick 4: Deduplication for False Positives

- Very important to get reliable p-values

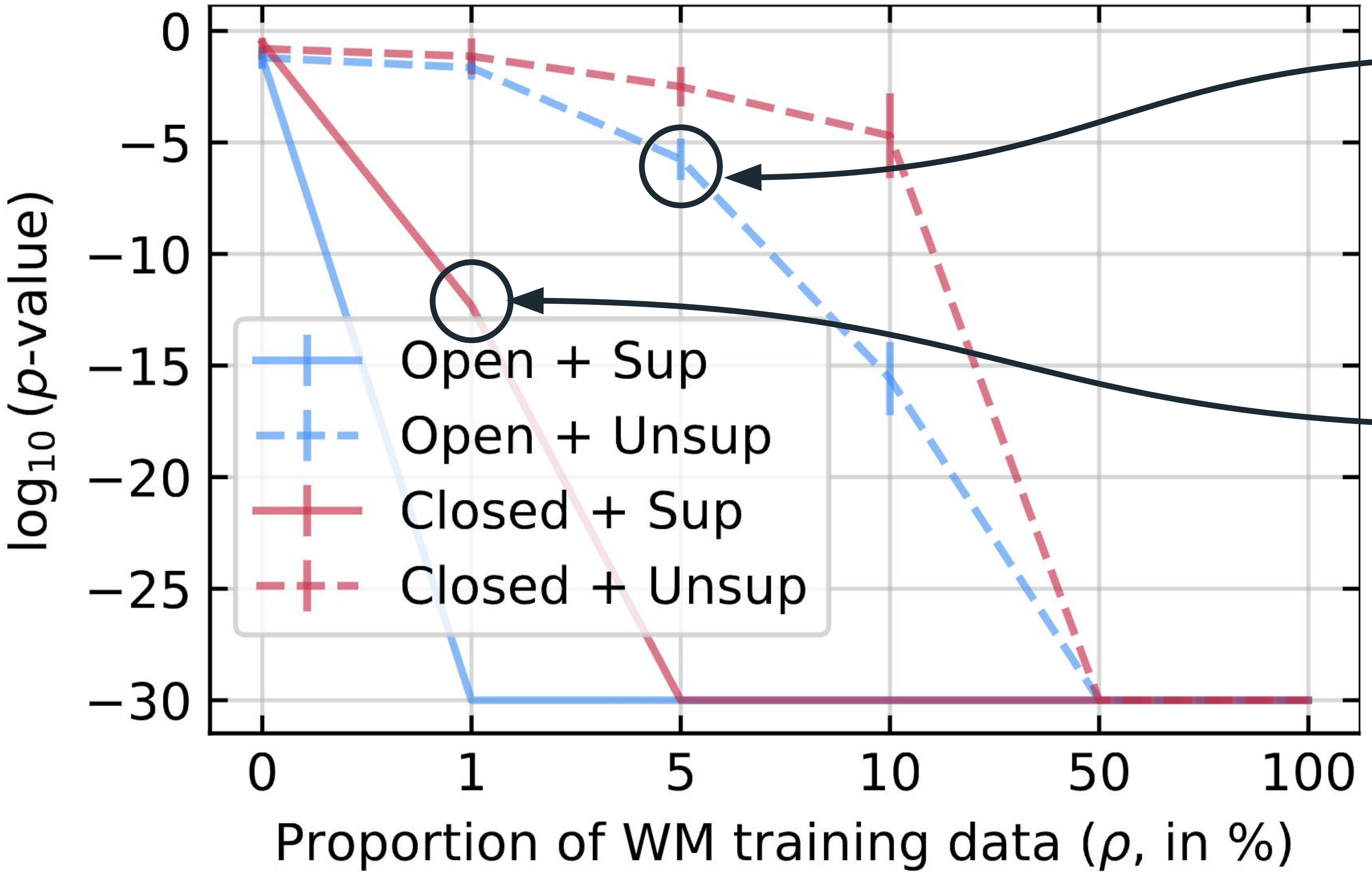
	De-duplication	
Access to Model	With	Without
Open	0.46±0.27	0.053±0.12
Closed	0.42±0.30	$< 10^{-30}$

- Lots of rules:
  - Don’t score tokens whose watermark window have already been scored
  - Don’t score tokens whose watermark window is already in the attention span

Experimental Setup

1. Generate watermarked instructions with Llama-2-chat-7b and Self-Instruct
2. Fine-tune Llama-1-7b with varying proportions of watermarked instructions
3. Get p-values of radioactivity detection

Detection Results under the Different Settings



- if the suspect model is open-weight, detection has p-value < 10^{-5} even when as little as 5% of training text is watermarked
- when Alice only has API access but knows which data have been used, detection has p-value < 10^{-10} even when 1% of the training text is watermarked

Ablations

Post-training optimization has a big influence on radioactivity

$\log_{10}$ p-value for 10k observed tokens under the supervised-open model setting

(a) Learning rate.

10^{-5}	$5 \cdot 10^{-5}$	10^{-4}
-32.4	-49.6	-58.0

(b) Epoch.

1	2	3	4
-20.8	-29.2	-33.2	-34.8

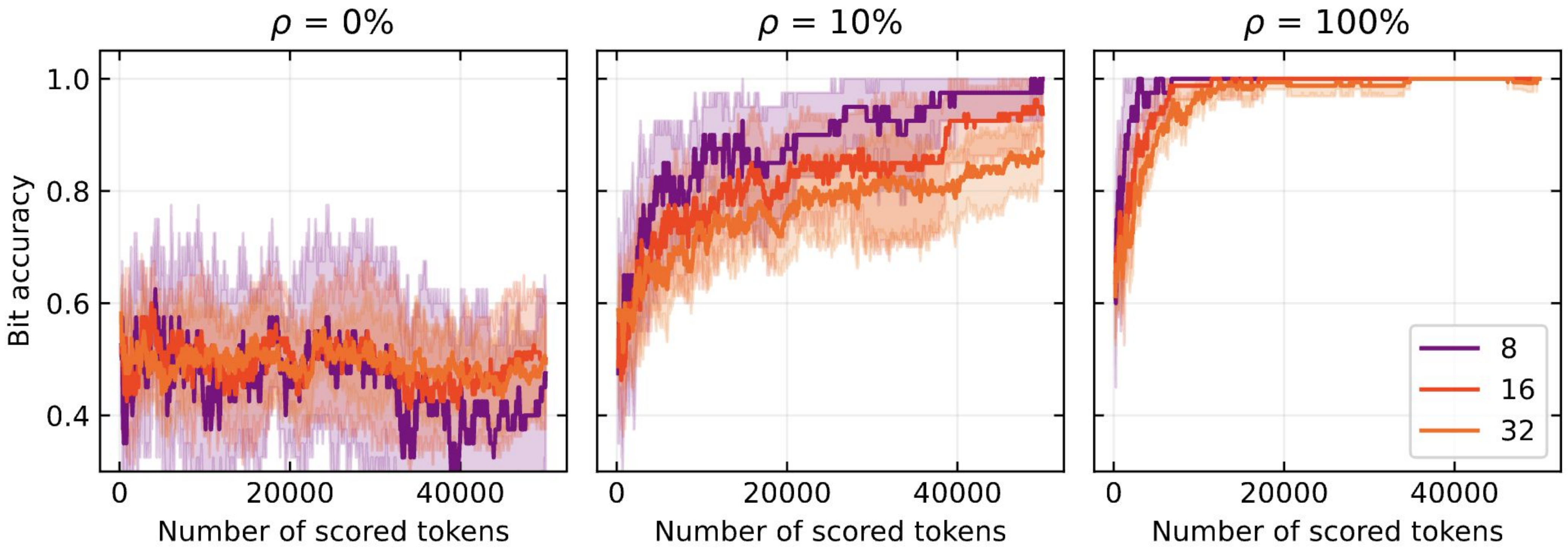
(c) Adapters.

Full	Q-LoRA
-32.4	-11.0

(d) Model size.

7B	13B
-32.4	-33.2

The method generalizes to multi-bit watermarking



Ablations

A lot more in the paper!

Main Takeaways

Watermarking makes LLM radioactive:

- Training on watermarked data can be **detected with very high confidence**…
- … even for **small proportions** of WM data

Thanks!