

Watermarking LLMs with Permute-and-Flip Sampling

Yu-Xiang Wang

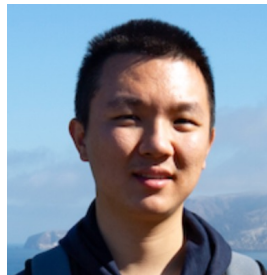


COMPUTER SCIENCE

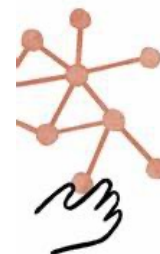
UC SANTA BARBARA

Computing. ReInvented.

Based on joint work with Xuandong Zhao, Lei Li



ChatGPT and other Large Language Models



Meet Claude

A next-generation AI assistant for your tasks, no matter the scale.
下一代AI助手，无论规模大小



Llama

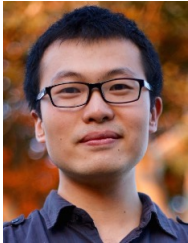


Alpaca, Vicuna



RedPajama

LLMs are very useful...



Could you generate HW2 for my machine learning please?
Make it difficult so it appears to be not AI-generated...



Certainly! Here is a difficult machine learning homework



The prof is crazy! HW2 is so difficult.
Could you write the solution for HW2 for me?



Problem 1 Solution: XXXXXXXXXXXX....

LLM can be used for malicious purposes, e.g., fake news, frauds, scams...

Disrupted

China reports first arrest over fake news generated by ChatGPT

Reuters

May 10, 2023 1:41 PM PDT - Updated a month ago



The logo of OpenAI is displayed near a response by its AI chatbot ChatGPT on its website, in this illustration picture taken February 9, 2023. REUTERS/Florence Lo/Illustration/File Photo

AI makes plagiarism harder to detect, argue academics - in paper written by chatbot

Lecturers say programs capable of writing competent student coursework threaten academic integrity



📍 Bristol University is among the institutions to have issued new guidance on how to detect the use of ChatGPT. Photograph: Adrian Sherratt/Alamy

What do we do?

Possible solution: Can you distinguish between human and AI-generated text?

The Information Theory and Applications (ITA) Workshop is a captivating and vibrant gathering that brings together some of the brightest minds in the world of information theory. This dynamic workshop serves as a melting pot for pioneering ideas, where experts and enthusiasts from various disciplines converge to explore the latest advancements in information theory and its myriad applications. From groundbreaking research presentations to thought-provoking discussions, ITA is not just a conference; it's a celebration of knowledge and innovation.



Human ?



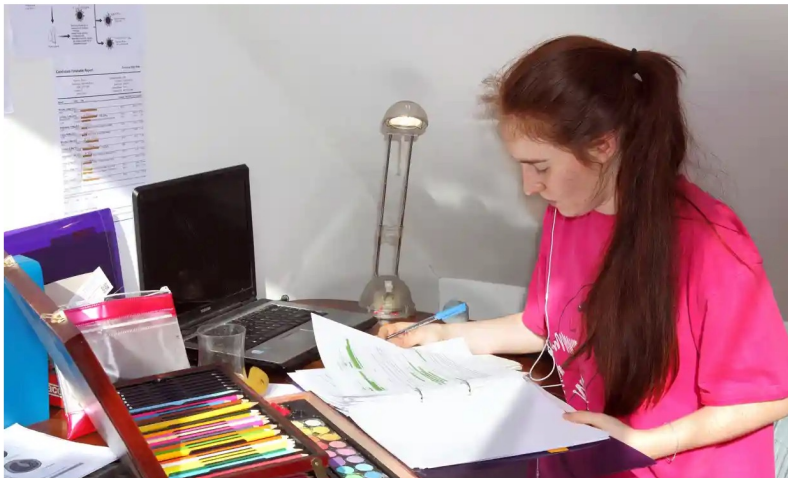
Machine ?

Train a machine learning model to solve Turing test?

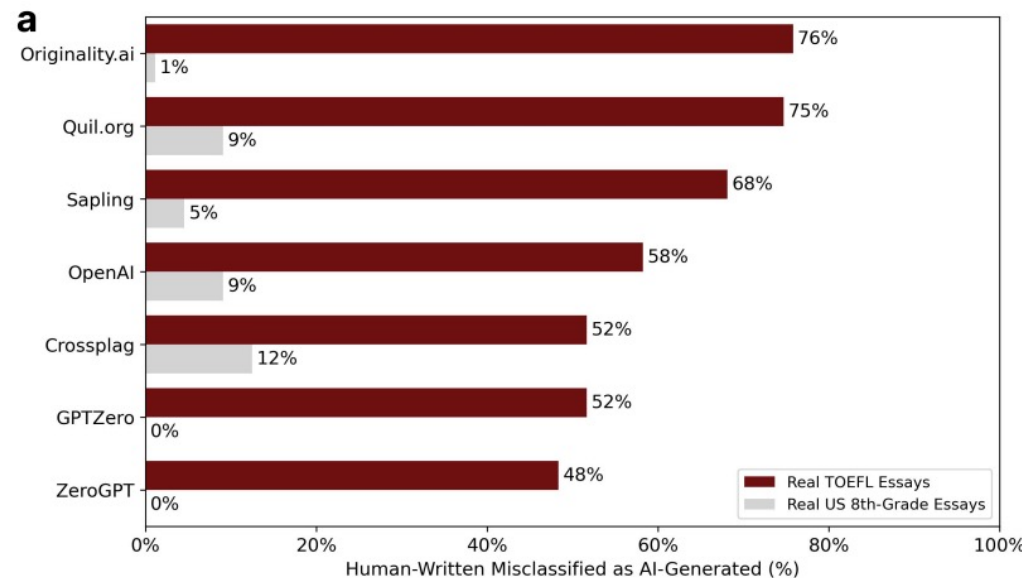
Training classifiers to detect AI-generated text is bound to fail!

Programs to detect AI discriminate against non-native English speakers, shows study

Over half of essays written by people were wrongly flagged as AI-made, with implications for students and job applicants



AI detectors could falsely flag college and job applications and exam essays as GPT-generated,



Liang et al. 2023: <https://arxiv.org/abs/2304.02819>

Better solution: “watermark” the generated text...



Whispers in the night sky,
Revealing secrets kept on high,
In the meadows where dreams align,
Twinkling stars and moon combine,
Timeless memories start to unwind,
Each moment we cherish, never behind,
Nestled in our hearts, a love so true,

Behold the beauty in every hue,
Yearning for a connection that's pure,

Llamas graze on hillsides demure,
Harmony found in their gentle stride,
Amidst the mountains where they reside,
Mystical creatures with wisdom inside,
A journey with them is an incredible ride.

Existing LLM watermarking schemes

- Statistical watermarks
 - Green-Red watermark ([Kirchenbauer et al., 2023](#))
 - Unigram (Green-Red) WM ([Zhao, Ananth, Li, W., 2023](#))
- Cryptographic watermarks
 - Gumbel watermark ([Aaronson, 2022](#))
 - Undetectable WM ([Christ, Gunn, Zamir 2023](#))
- Quite a few others in this fast-growing research area

All existing watermarks work with the standard decoder: **softmax(logits)**

Softmax sampling: $y_t \sim p(y) = \frac{e^{u(y|x, y_{1:t-1})/T}}{\sum_{\tilde{y}} e^{u(\tilde{y}|x, y_{1:t-1})/T}}$

- Temperature parameter T:
 - Large T \Leftrightarrow higher text entropy (more watermarkable)
 - Small T \Leftrightarrow higher text quality (smaller perplexity).

1. Is **softmax(logits)** the optimal choice?

2. Can we benefit from **co-designing** the decoder and watermarking scheme?

TL;DR of our results

1. We propose “Permute-and-Flip Decoding”
 - PF dominates Softmax in robustness-perplexity tradeoff.
2. A cryptographic watermark for Permute-and-Flip
 - Enjoys all nice properties of the Gumbel watermark
 - Slightly better detectability-perplexity tradeoff

Permute-and-Flip Sampling from Differential Privacy literature (McKenna and Sheldon, 2021)

Algorithm 1 Permute and Flip (PF) Decoding

- 1: **Input:** prompt x , language model \mathcal{M} , temperature T .
 - 2: **for** $t = 1, 2, \dots$ **do**
 - 3: Logits $u_t \leftarrow \mathcal{M}([x, y_{1:t-1}])$.
 - 4: Find $u_t^* \leftarrow \max_{y \in \mathcal{V}} u_t(y)$.
 - 5: **Permute** : Shuffle the vocabulary \mathcal{V} into $\tilde{\mathcal{V}}$. **Permute**
 - 6: **for** $y \in \mathcal{V}$ **do**
 - 7: **Flip** : Draw $Z \sim \text{Bernoulli} \left(\exp \left(\frac{u_t(y) - u_t^*}{T} \right) \right)$. **Flip**
 - 8: **if** $Z = 1$, **then** assign $y_t \leftarrow y$ and **break**.
 - 9: **end for**
 - 10: **end for**
 - 11: **Output:** Generated sequence $y = [y_1, \dots, y_n]$.
-

Permute-and-Flip(logits) is very similar to Softmax(logits)

Rejection sampling form of Softmax sampling

1. Uniformly samples $y \in \mathcal{V}$,
2. Return it with probability

$$p(y)/p(y^*) = \exp((u_t(y) - u_t(y^*))/T).$$

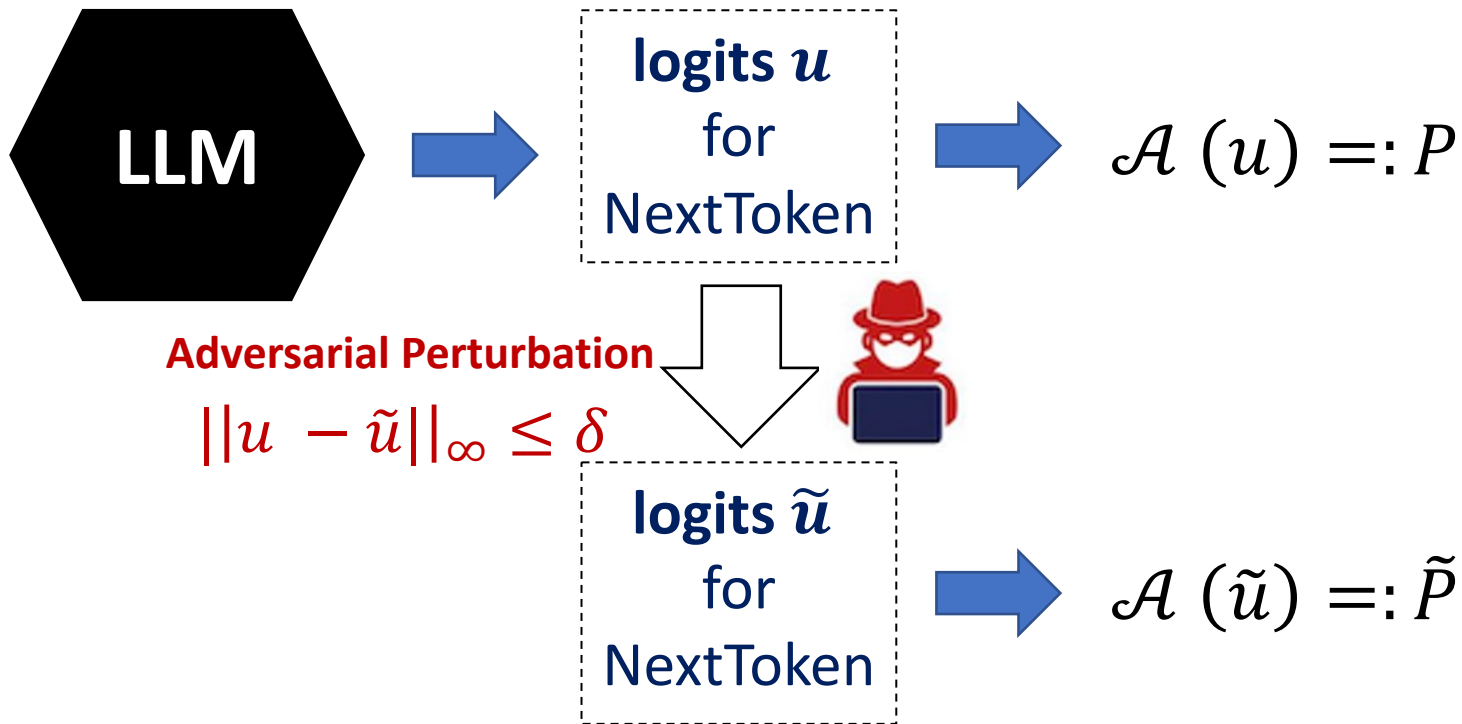
Permute-and-Flip does nothing but replacing Step 1 by **sampling without replacement.**

The advantage of PF Sampling is that it gets all the nice properties of the softmax but improves the perplexity.

Methods	Perplexity	Computational Efficiency	Diversity	Watermark	Robustness
Search (e.g., Beam)	Lowest	✗	✗	✗	✗
Greedy	Low	✓	✗	✗	✗
Softmax Sampling	Moderate	✓	✓	✓	✓
Top- p Sampling	Low (for small p)	✓	Depends on p	✓	✗
Top- k Sampling	Low (for small k)	✓	Depends on k	✓	✗
PF Sampling (ours)	Lower than Softmax	✓	✓	✓	✓

Table 1: Comparison of different decoding methods against five desiderata.

Robustness against adversarial perturbation to the logits



Definition: L-robustness.

\mathcal{A} is L-robust if $\left| \log \left(\frac{dP}{d\tilde{P}} \right) \right| \leq L \delta$

Both Softmax and P&F are provably robust, but P&F is up to 2x better than Softmax at “optimization”

Theorem (McSherry and Talwar, 2007):

Softmax sampling is $1/T$ -robust.

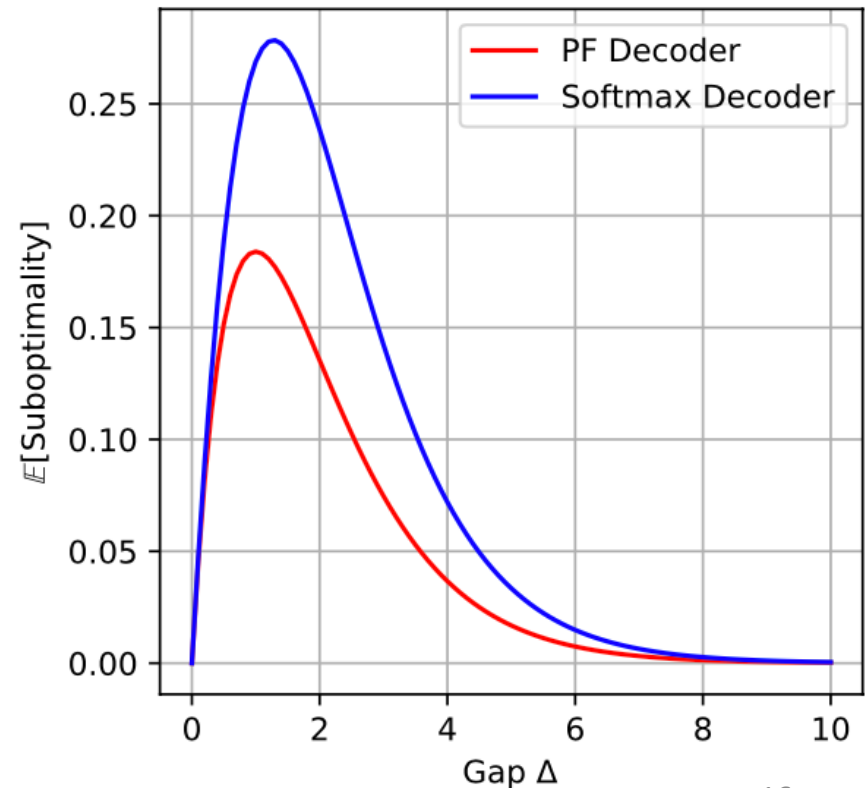
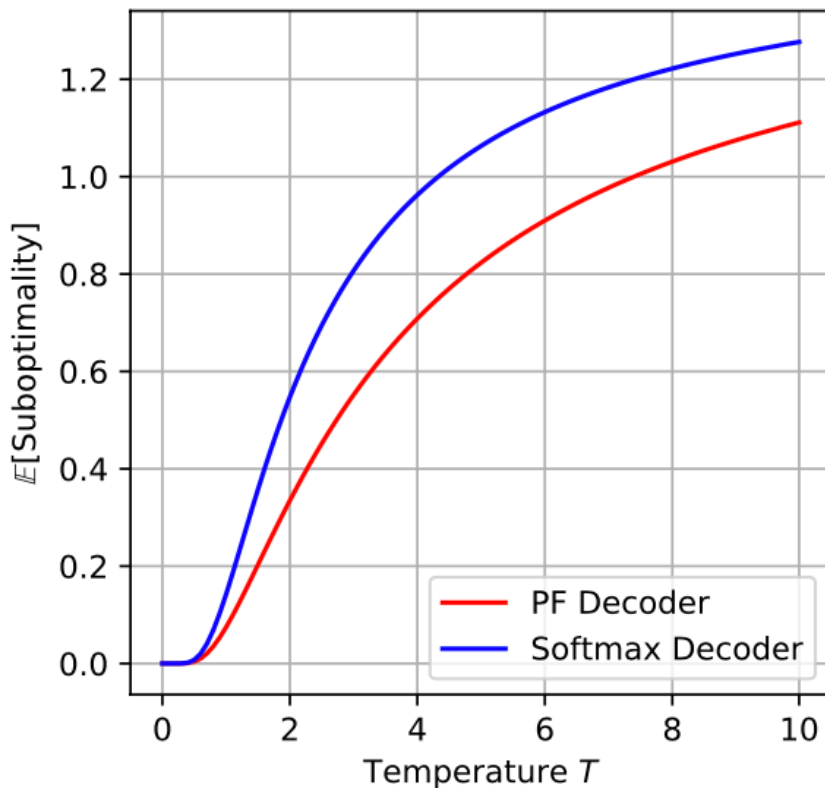
Theorem (McKenna and Sheldon, 2021):

1. Permute-and-Flip sampling is $1/T$ -robust.
2. For the same T , PF dominates Softmax in terms of expected suboptimality.
3. PF is Pareto-optimal in robust-suboptimality tradeoff.

PF decoder dominates softmax decoder for all parameter choices

Example: Two token vocabulary, logits $u = [0, \Delta]$.

Suboptimality: $u^* - \mathbb{E}[u]$



PF improves perplexity on open-domain generation datasets

Method	PPL1↓	PPL2↓
C4, T=1.0, Llama2-7B		
Sampling	12.47 _{0.32}	15.31 _{0.41}
PF	8.94 _{0.20}	10.75 _{0.25}
C4, T=0.8, Llama2-7B		
Sampling	4.23 _{0.06}	4.91 _{0.08}
PF	3.54 _{0.06}	4.11 _{0.08}
Alpaca, T=1.0, Llama2-7B-Chat		
Sampling	1.74 _{0.02}	2.41 _{0.04}
PF	1.65 _{0.02}	2.30 _{0.04}

TL;DR of our results

1. We propose “Permute-and-Flip Decoding”
 - PF dominates Softmax in robustness-perplexity tradeoff.
2. A cryptographic watermark for Permute-and-Flip
 - Enjoys all nice properties of the Gumbel watermark
 - Slightly better detectability-perplexity tradeoff

From Gumbel-Softmax trick to Exponential-PF trick

- Gumbel-Softmax trick (Gumbel, 1948)

$$y_t \sim \text{Softmax} \left(\frac{u_t(y)}{T} \right) \iff \begin{aligned} y_t &= \arg \max_{y \in \mathcal{V}} \frac{u_t(y)}{T} + G_t(y) \\ G_t(y) &\sim \text{Gumbel}(0, 1) \text{ i.i.d} \end{aligned}$$

- Exponential-PF trick (Ding et. al, 2021)

$$y_t \sim \text{Permute\&Flip} \left(\frac{u_t(y)}{T} \right) \iff \begin{aligned} y_t &= \arg \max_{y \in \mathcal{V}} \frac{u_t(y)}{T} + E_t(y). \\ E_t(y) &\sim \text{Exponential}(1) \text{ i.i.d.} \end{aligned}$$

ReportNoisyMax from Differential Privacy.

Idea to watermark PF-Decoding

- Gumbel-Watermark (Aaronson, 2022)

$$y_t = \arg \max_{y \in \mathcal{V}} \frac{u_t(y)}{T} + G_t(y)$$

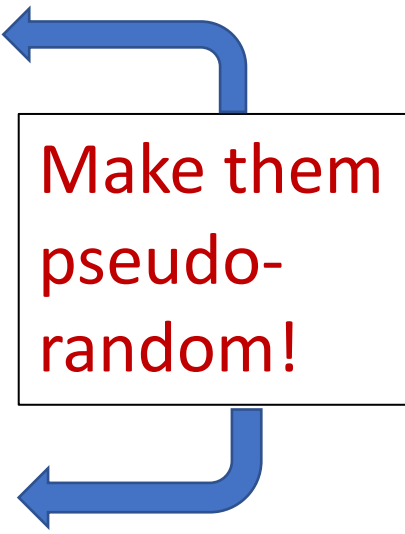
$$G_t(y) \sim \text{Gumbel}(0, 1) \text{ i.i.d.}$$

- PF-Watermark (Ours)

$$y_t = \arg \max_{y \in \mathcal{V}} \frac{u_t(y)}{T} + E_t(y).$$

$$E_t(y) \sim \text{Exponential}(1) \text{ i.i.d.}$$

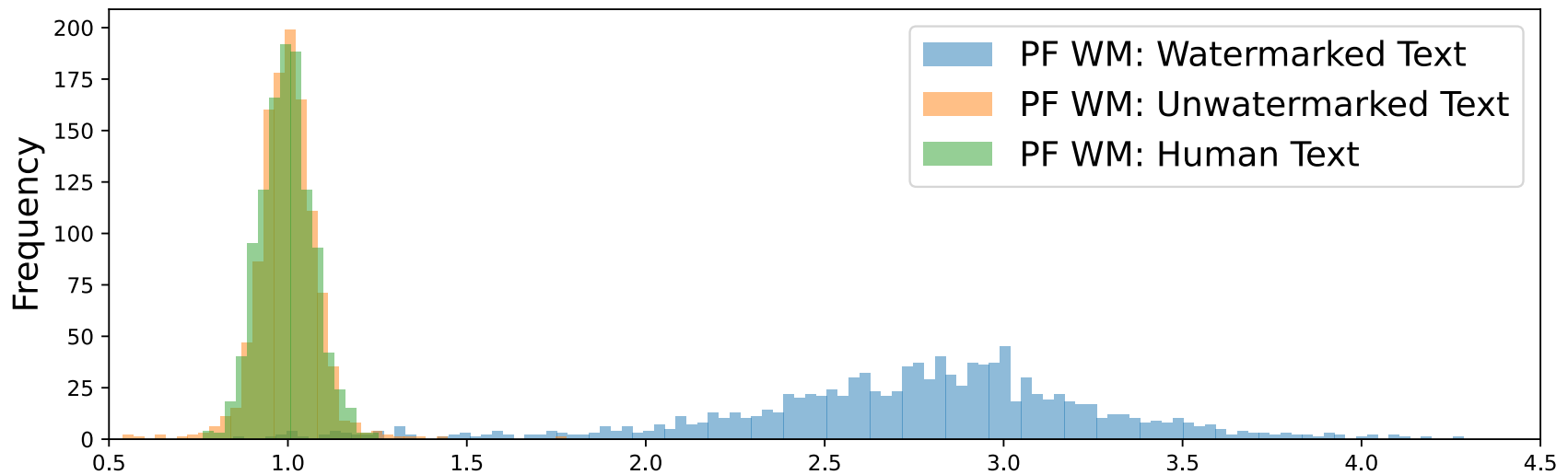
Make them
pseudo-
random!



Detection score for PF-watermark

$$\text{TestScore}_{\text{PF}}(y_{1:n}) = \sum_{t=m+1}^n -\log(r_t(y_t))$$

where $r_t(y) = F_{y_{t-m:t-1},k}(y)$

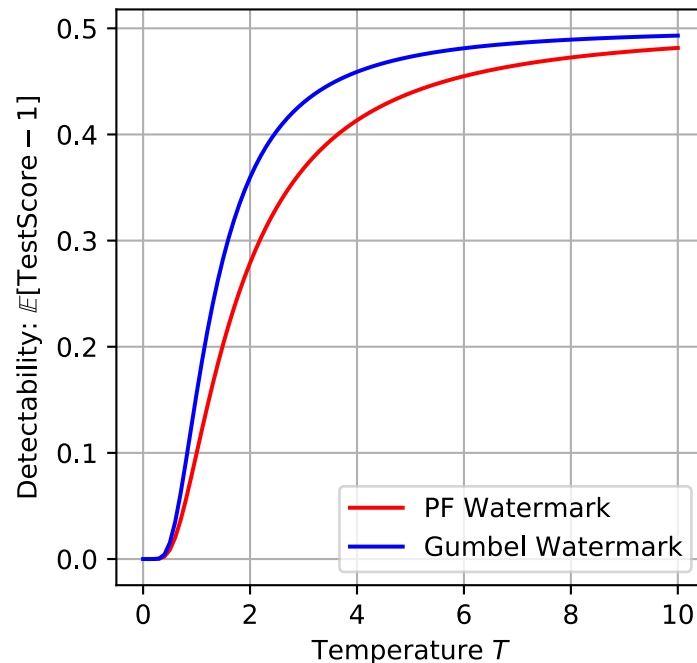
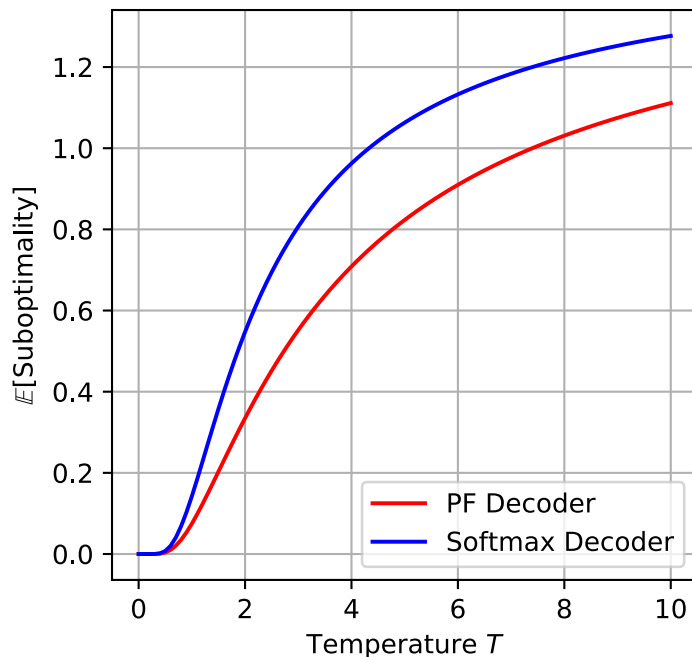


Guarantees of PF-watermark are analogous to those of the Gumbel

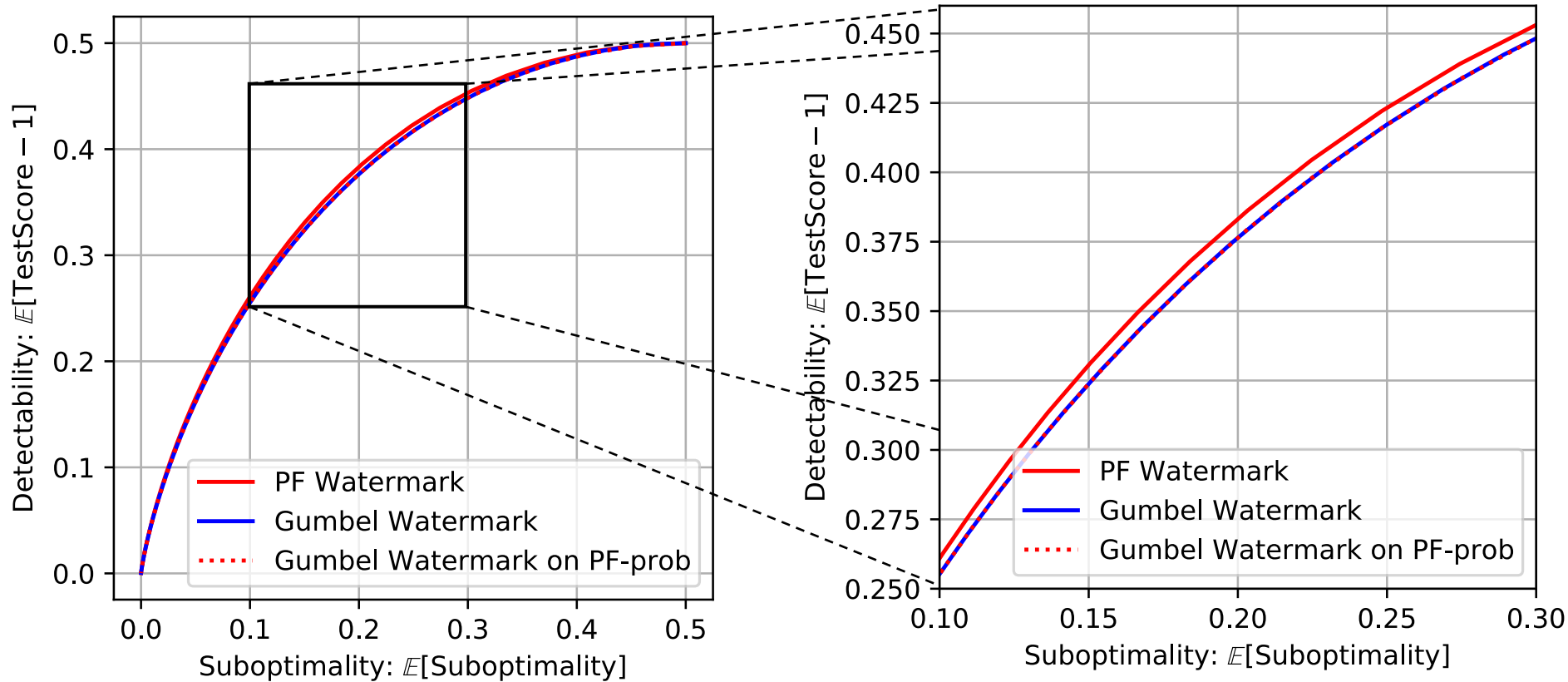
- Distortion-free
 - Computationally indistinguishable from PF-decoding.
- Precise FPR control
 - $\text{TestScore}/n \rightarrow 1$ under the null hypothesis.
 - Under the null hypothesis, the test-score follows a Gamma distribution.
- High power if generated text has *high-entropy*
 - $\text{TestScore}/n \rightarrow \alpha$ for $\alpha \gg 1$ under the alternate hypothesis

How does PF-watermark compare to Gumbel watermark?

- **Example:** Two token vocabulary, logits $u = [0, \Delta]$.
- **Detectability:** $\mathbb{E}[\text{Score} | \text{WM}] - \mathbb{E}[\text{Score} | \text{No WM}]$
- **Suboptimality:** $u^* - \mathbb{E}[u]$

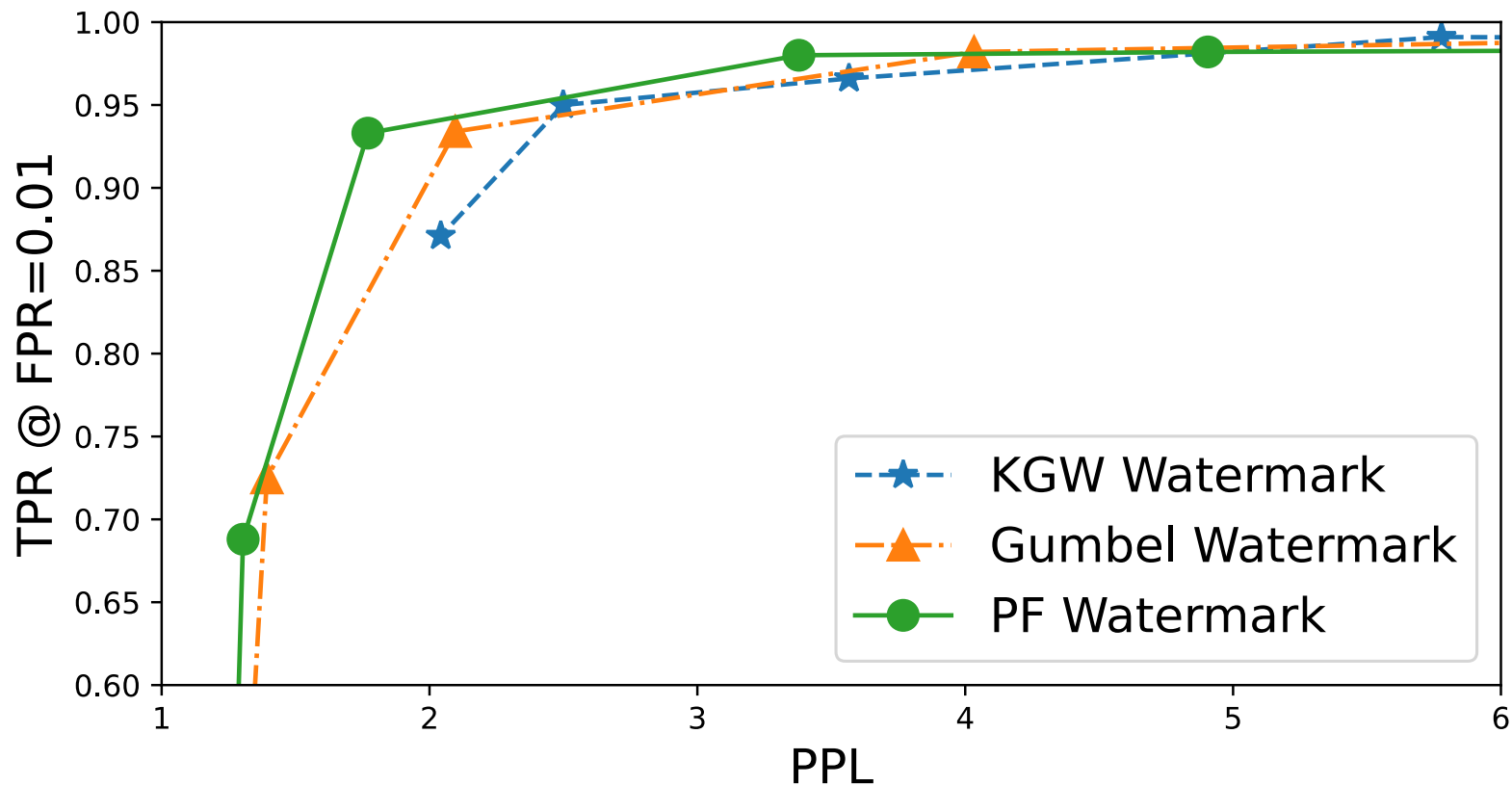


Plotting detectability against suboptimality as we adjust T



PF has more favorable tradeoff curves than Gumbel

On real datasets: the PF watermark provides better Detectability-Perplexity Tradeoffs



Take-home-messages

- Watermarking LLM text is an emerging research problem that prevents AI abuse.
- We propose Permute-and-Flip decoding and developed a natural watermarking scheme for it.
 - For the same perplexity, it improves detectability and robustness.
- Interesting connection to the differential privacy literature --- more interplays in the future.

Thank you for your attention!

- **Permute-And-Flip: An Optimally Robust and Watermarkable Decoder for LLMs**

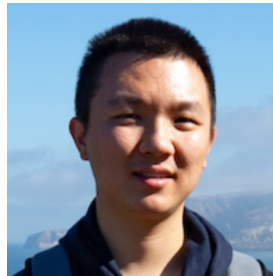
Xuandong Zhao, Lei Li, Yu-Xiang Wang.

Technical report, 2024 [[arxiv](#), [code](#)]

- **Provable Robust Watermarking for AI-Generated Text**

Xuandong Zhao, Prabhanjan Ananth, Lei Li, Yu-Xiang Wang.

ICLR 2024 [[arxiv](#), [slides](#), [code](#), [demo](#)]



Xuandong Zhao



Lei Li



Prabhanjan Ananth

Supplementary slides

L-robustness implies an intuitive definition of “diversity”

- If $|u_t(y) - u_t(y')| \leq \delta$,
- Then we can construct \tilde{u} such that $\tilde{u}(y) = \tilde{u}(y')$ and $|\tilde{u} - u| \leq \frac{\delta}{2}$

$$\left| \log \frac{p_{\mathcal{A}_{u_t}(y)}}{p_{\mathcal{A}_{u_t}(y')}} \right| = \left| \log \frac{p_{\mathcal{A}_{u_t}(y)}}{p_{\mathcal{A}_{\tilde{u}_t}(y)}} + \log \frac{p_{\mathcal{A}_{\tilde{u}_t}(y')}}{p_{\mathcal{A}_{u_t}(y')}} \right| \leq L\delta.$$

Let me explain how the Gumbel watermark works...

- (Almost) distortion-free, i.e., no quality drop. How?
 - Gumbel Softmax Trick!
 - NextToken \sim softmax(logits)
 - NextToken = argmax logits + Gumbel noise
- Watermarking phase
 - “ITA is my favorite conference. It always ____”
- Detection phase
 - We know the prefix and the random seeds..

What are needed for a good watermark for LLM generated text?

- Quality of generated text
- Detection guarantees
 - Type I error: “No false positives”
 - Type II error: “Only true positives”
- Security property
 - Resilient to all kinds of evasion attacks (e.g., edits, paraphrasing)
- Other required properties
 - Efficiency, Model-agnostic detection.