Watermarking of Large Language Models

Aidan Prendergast

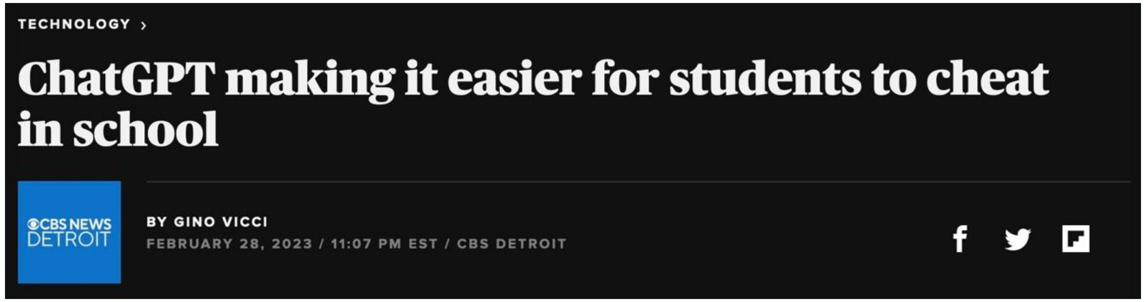
11/20/23

PURDUE UNIVERSITY | Elmore Family School of Electrical and Computer Engineering

The Necessity of Watermarking

Malicious Users of LLM Capability in 2023

- Machine-Driven content generation has never been faster, more accessible, or easier
- Syllabi are now addressing Large Language Model use and cheating
- Generic, generated-content websites are beginning to crop up across a variety of disciplines
- Social media bots are becoming more capable, and sound more human than ever
- To begin to combat the issue, we must be able to accurately discriminate human text from machine-generated text



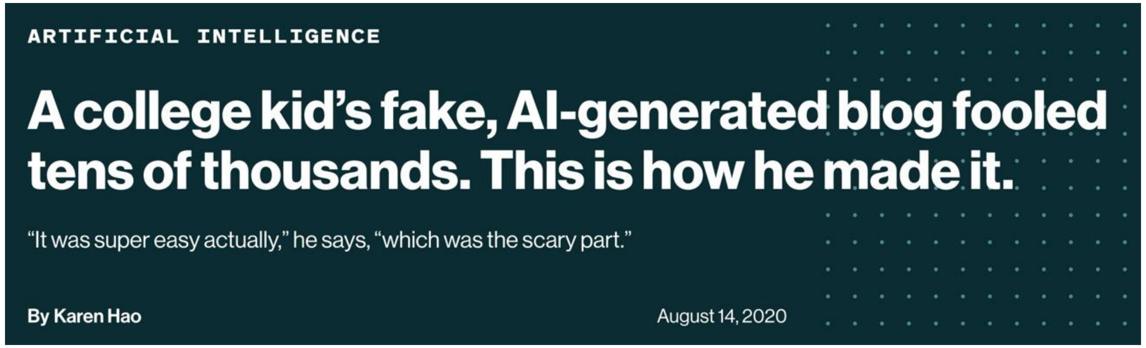
TECHNOLOGY ›

ChatGPT making it easier for students to cheat in school

CBS NEWS DETROIT

BY GINO VICCI

FEBRUARY 28, 2023 / 11:07 PM EST / CBS DETROIT



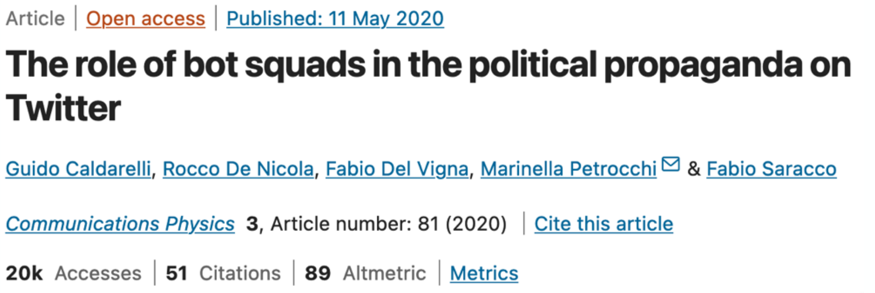
ARTIFICIAL INTELLIGENCE

A college kid's fake, AI-generated blog fooled tens of thousands. This is how he made it.

"It was super easy actually," he says, "which was the scary part."

By Karen Hao

August 14, 2020



Article | Open access | Published: 11 May 2020

The role of bot squads in the political propaganda on Twitter

Guido Caldarelli, Rocco De Nicola, Fabio Del Vigna, Marinella Petrocchi & Fabio Saracco

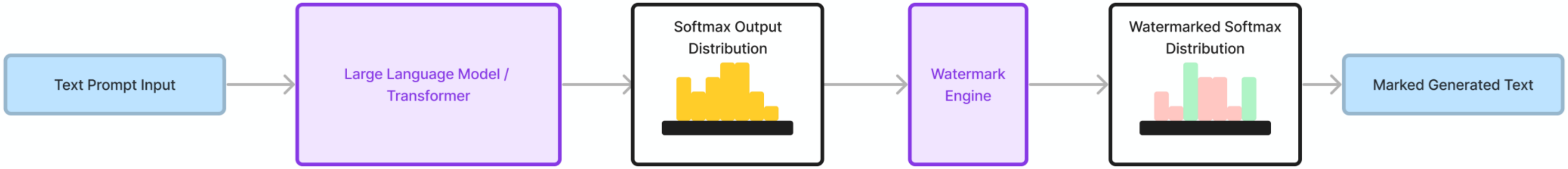
Communications Physics 3, Article number: 81 (2020) | Cite this article

20k Accesses | 51 Citations | 89 Altmetric | Metrics

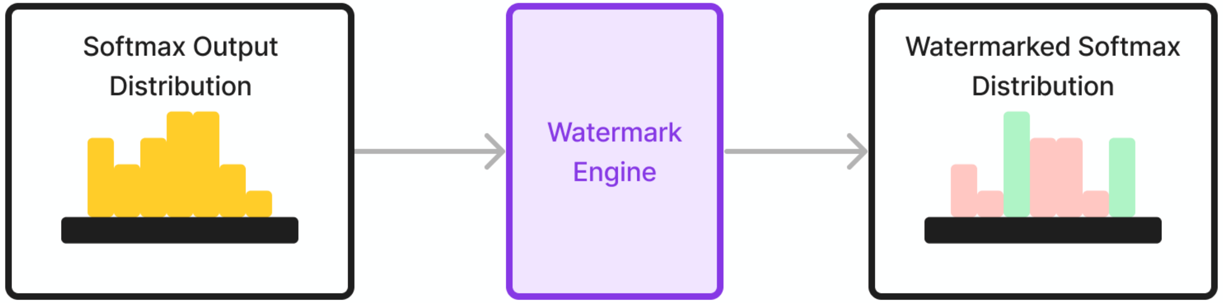
PURDUE UNIVERSITY | Elmore Family School of Electrical and Computer Engineering

The Watermarking Pipeline

How Watermarked Text is Output from a Typical LLM


Text Prompt Input
Large Language Model / Transformer
Softmax Output Distribution
Watermark Engine
Watermarked Softmax Distribution
Marked Generated Text

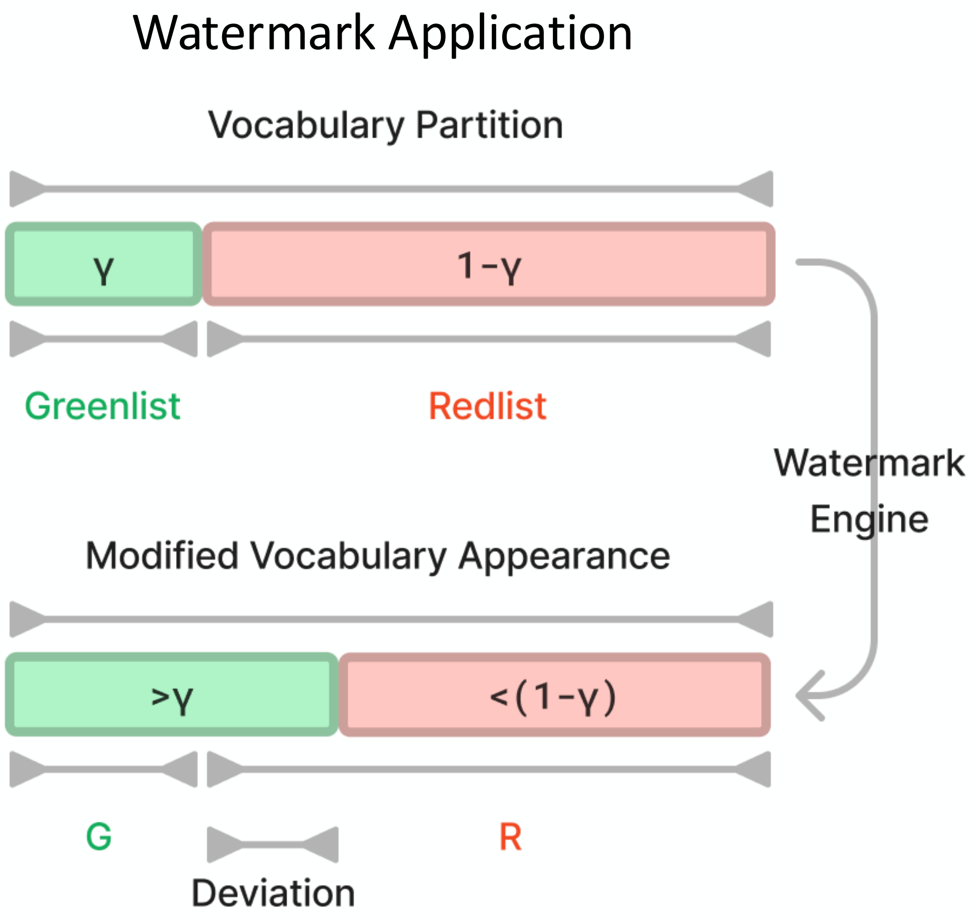
- LLM Objective: Token Prediction
- LLM Softmax Output
- Distribution Manipulation
- Token Selection from Watermark Output

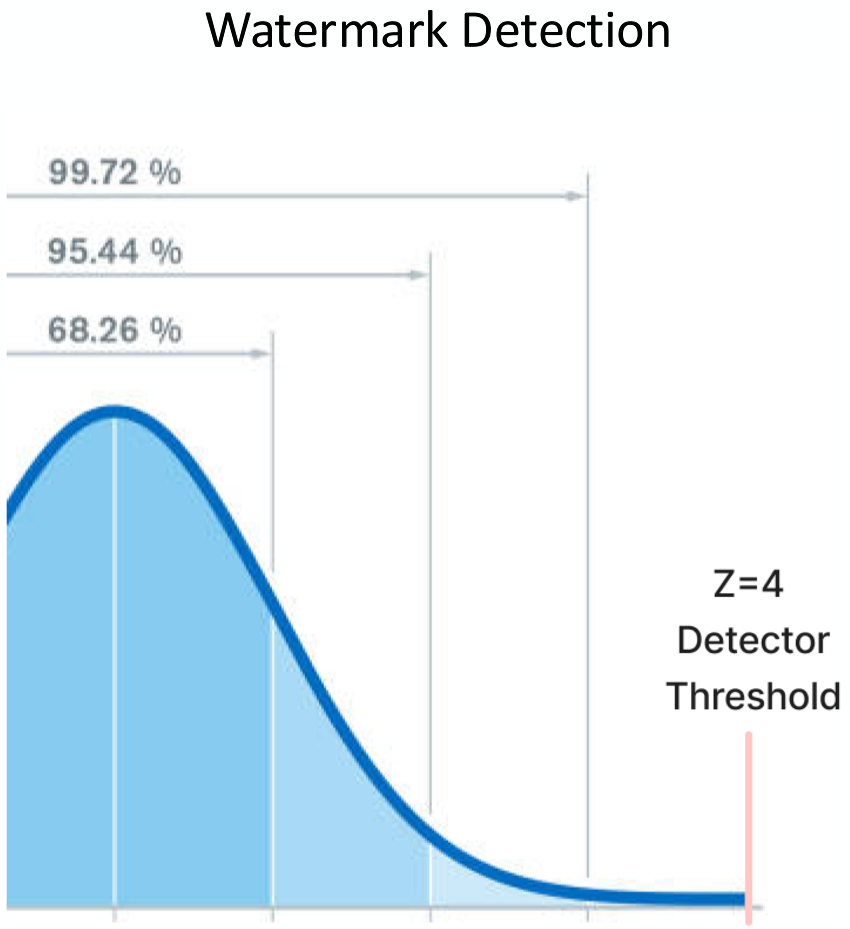


Distribution Transformation

Watermarking Engine and Watermark Detection

Invisible Statistical Manipulation at High Confidence



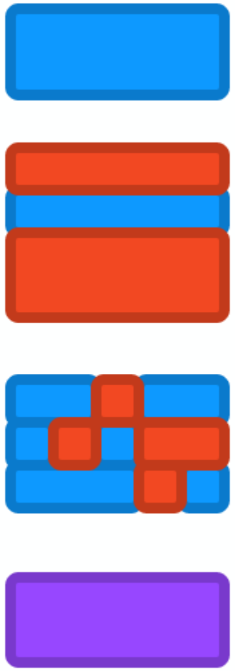


Attacking the Watermark

Robustness and Common Use-case Evaluation

Watermarked | Unmarked | Summary

- Attack Types:



  - Copy-Paste (No Attack)
  - Embed Into Text (Dilution)
  - Replacement - Variation of generated text to mask watermark
  - Rephrasing/Summary Attack
    - Feeding text to GPT-3.5

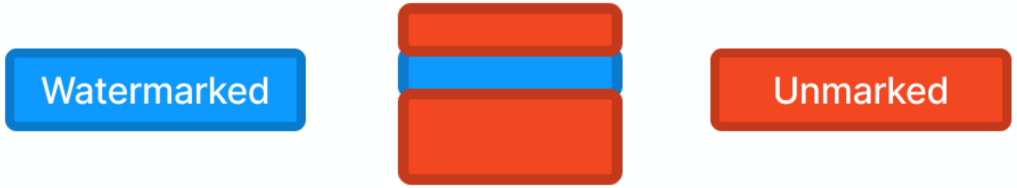
- Attack Success: Z-Threshold = 4
  - Copy-Paste: Z-score = 9.04
  - Dilution: Z-score = 7.24
  - Replacement: Z-score = 7.00
  - Rephrasing: **Z-score = 4.41**
    - ChatGPT could not remove the watermark by rephrasing the text.
    - Even human subjects in the Kirchenbauer et al. follow-up paper could not remove the watermark through rephrasing.

Full attack outputs can be found in Appendix D of my paper.

Enhanced Watermark Tracking: Sliding Window Detection

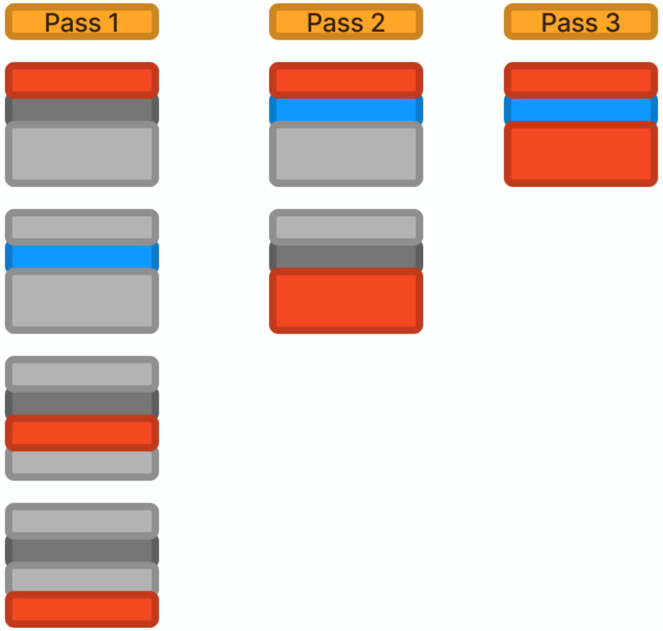
My Contribution to the Watermarking Codebase

- Dilution-style attacks, in large magnitude, are rather effective and likely attack vectors
  - One paragraph in an essay
  - One blurb in a news article
  - One generated tweet in a large set of tweets



- These attacks would be rather easy to spot if we could identify the “hotspot” in the diluted text
  - The detector just takes the whole text at once
  - Binary all-or-none is generated approach

- The Sliding-Window Detector:
  - Takes multiple passes over the input text
  - Looks for the largest offending subset of text
  - Particularly small sets may have high variance due to small sampling size bias, so a minimum set size (20) is passed to the detector.



Thank You

Aidan Prendergast

PURDUE UNIVERSITY | Elmore Family School of Electrical and Computer Engineering