On the Reliability of Watermarks for Large Language Models

> https://arxiv.org/pdf/2306.04634

以KGW为例，分析水印方法的鲁棒性，针对三个attack：

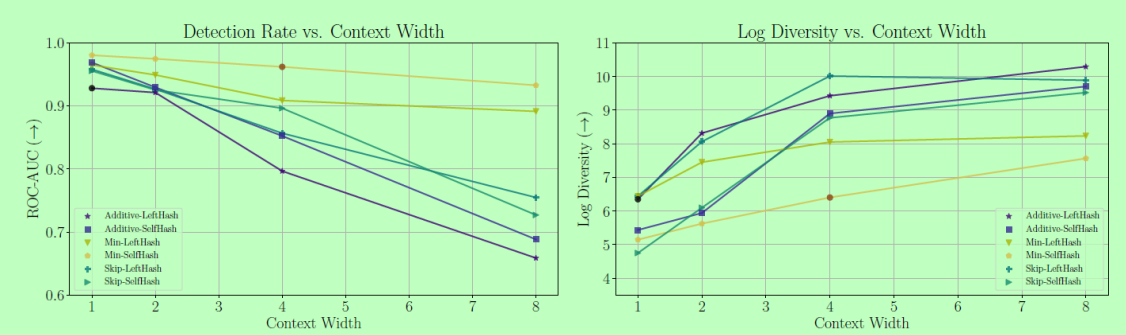
- rewritten by humans,
- paraphrased by a non-watermarked LLM
- mixed into a longer hand-written document

在KGW方法的基础上，如何做改进?

1. 采用不同的Hashing规则：（当为了secrecy不得不扩大window size的时候，很容易不鲁棒）
   - 分类方式1：是否考虑当前位置：是 SelfHash / 否 LeftHash
   - 分类方式2：如何计算hash
     - Additive：生成第t个token时，把它左边窗口中的token id加起来再hash；

       对修改次序鲁棒，但不能应对insertion/deletion/swap
     - Skip：只用窗口中最左边的token
     - Min：把窗口中的token id做hash之后取最小值

   比较这三种方法在GPT paraphrasing后的Detection Rate，以及它们生成文本的Diversity（一定程度代表文本质量）

   - 结论：随着window size的增大，在gpt paraphrasing attack下，Min-SelfHash的鲁棒性最好；但是相应的它的Log Diversity也最低，是一个trade-off关系
2. 优化Watermark的检测方法：不用直接计算全局z-score的方式

   取所有片段z-score的Max值：p_i表示到i为止的绿词数量

$$z_{\text{win-max}} = \max_{\substack{i,j,\\ i<j}} \frac{(p_j - p_i) - \gamma(j-i)}{\sqrt{\gamma(1-\gamma)(j-i)}}.$$

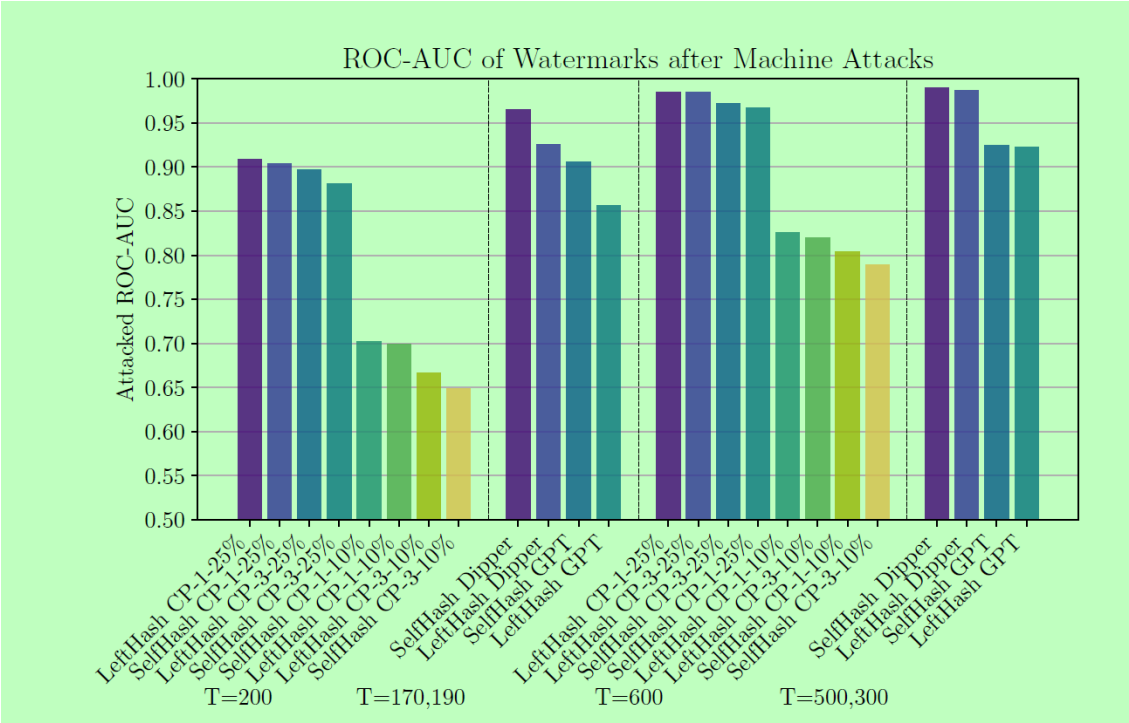
实验

- Machine Paraphrasing Attacks

  用了gpt-3.5-turbo和Dipper去rewrite，结果如下；看起来鲁棒性还行，但是有几点文章里没交代：

  - window size用的多少（比较重要）
  - 是否使用Min-的hash机制

- 是否用了WinMAX检测机制



- Copy-Paste Attacks：把水印文本插入human-text，两个参数：插入的片段数和watermark fragment的占比

  结果如上，CP比paraphrasing强度更高一点。

- Paraphrasing by Human Writers

  这部分之前没什么人做过，比较好奇他的实验具体是咋做的。

  先招募了14个human writers，用LFQA data先在watermarked Vicuna上生成语料，然后再让human writers去paraphrase。在真正采用之前，先用P-SP指标衡量一下rewrite的怎么样，发现他们都远远达标了。

  这个图还是挺清晰的，值得学习一下。可以看到，Human-rewrite的强度是最大的。但是Token数量到一定值的时候也能被检测出来。