An Entropy-based Watermark Detection Method

> https://arxiv.org/abs/2403.13485

- motivation：低熵场景下如果用传统的KGW统计检测，由于logits的一小点更改对红绿的选择影响不大，所以最后算z-score的时候可能有水印和无水印的结果差别不是很大
- SWEET：（这个和EWD的motivation其实不太一样，SWEET是想减小对文本质量的影响。所以加水印的时候先算entropy，如果entropy很低就不加了，检测的时候也跳过这些只检测其它的）
- 这篇在拿SWEET做baseline的时候，比的主要就是检测的准确率，因为SWEET是1/0的weight，而这篇是continuous weight
- 因此这里重点在3个地方，一个是entropy怎么计算，一个是weight怎么根据entropy分配，一个是最后统计量的公式
  - entropy怎么计算：Spike entropy

    $SE = \sum_k \frac{p_k}{1+zp_k}$，p是大模型输出的概率向量，z是个标量
  - compute weight函数：减去最小熵
  - 统计量公式

    $|s|_G$就是对每个位置测红绿之后，绿的就把对应的weight加进去