【论文极速读】引入复读负样本，一种打破LLM复读问题的方法

FesianXu 20240429 at Tencent Wechat search team

前言

最近工作里面遇到了LLM复读的问题，去翻了下论文，看到有一篇尝试通过引入负样本解决复读问题的工作，有所启发，在此简单介绍下，希望对大家有所帮助。**如有谬误请见谅并联系指出，本文遵守CC 4.0 BY-SA版权协议，转载请联系作者并注明出处，谢谢。**

∇ 联系方式：

e-mail: FesianXu@gmail.com

github: https://github.com/FesianXu

github page: https://fesianxu.github.io/

知乎专栏: 计算机视觉/计算机图形理论与应用

微信公众号：机器学习杂货铺3号店

LLM的复读问题，一般有几种级别的复读，如下所示

- 字粒度的复读：

> User: 你喜欢北京么?
>
> AI: 北京是中国的首都，有很多名胜古迹，如长城，故宫，天坛等，我十分喜欢欢欢欢欢欢欢欢欢欢欢欢欢....

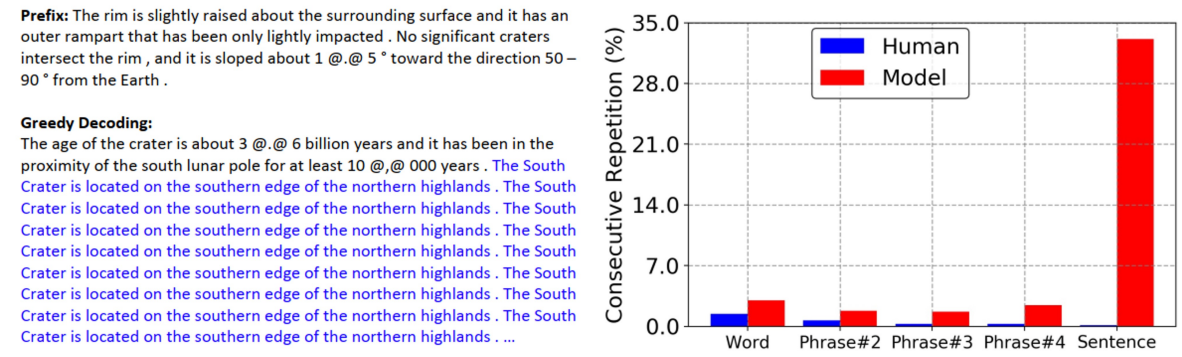
- 词粒度的复读：

> User: 你喜欢北京么?
>
> AI: 北京是中国的首都，有很多名胜古迹，如长城，故宫，天坛等，我十分喜欢喜欢喜欢喜欢....

- 句子粒度的复读：

> User: 你喜欢北京么?
>
> AI: 北京是中国的首都，有很多名胜古迹，如长城，故宫，天坛等，我十分热爱北京，我十分热爱北京，我十分热爱北京，.....

贪婪搜索解码（greedy search）由于其解码结果是固定的（deterministic），并且解码速度快等优点，是在实际应用中经常使用的解码方法。在清华大学的一篇论文 [1]中，介绍了一种在贪婪搜索解码的前提下对复读问题的解决方案。如Fig 1 (b)所示，在Wikitext-103 dev数据集上，作者统计了模型生成结果和人类结果在不同粒度（word、 phrase、sentence）下的连续复读占比情况。不难发现人类编写的结果的连续复读占比随着粒度的增大，会快速减少，而模型生成的结果则在句子粒度上的复读中达到了惊人的最大值（~35%）。这说明在贪婪解码中，句子粒度的复读是最常见的复读模式，因此作者对这种模式的形成原因进行了研究。

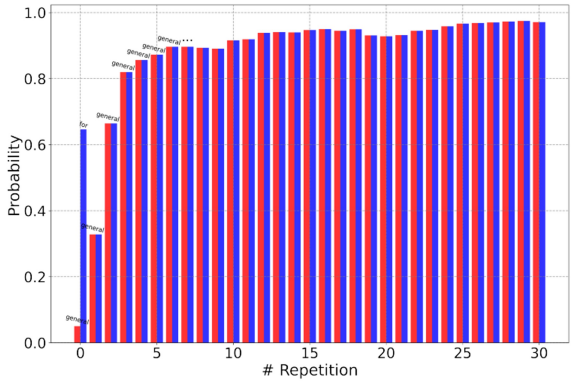


(a) 句子粒度的复读

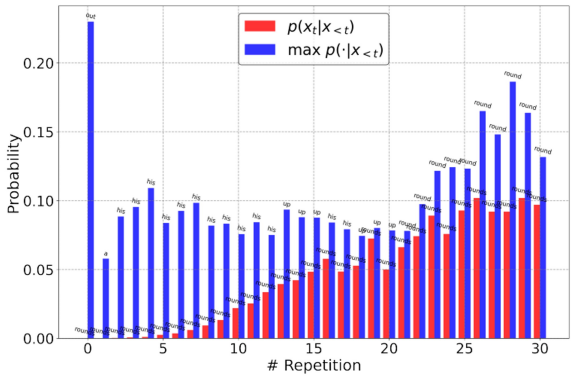
（b）模型生成和人类结果在各种粒度上的连续复读占比

Fig 1. 模型生成结果和人类结果在各种粒度上的连续复读占比。

作者研究这个现象的基本方法，可以简单理解为手动重复一个句子，然后研究每个token的生成概率的变化，举个例子就是$P_\theta(\text{" } orange \text{ "} \mid \text{" } I\ love \text{ "})$和$P_\theta(\text{" } orange \text{ "} \mid \text{" } I\ love\ orange,\ I\ love \text{ "})$的关系，作者发现，随着手动重复的句子次数越多，其下一个句子出现复读的概率就会越大。



(a) 在正常语句下



(b) 在随机语句下

Reference

[1]. Xu, Jin, Xiaojiang Liu, Jianhao Yan, Deng Cai, Huayang Li, and Jian Li. "Learning to break the loop: Analyzing and mitigating repetitions for neural text generation." *Advances in Neural Information Processing Systems* 35 (2022): 3082-3095.

[2]