arxiv.org/abs/2503.22828

Learning to Reason for Long-Form Story Generation

如何将RLVR应用于长故事生成任务?

Alexander Gurung, Mirella Lapata
School of Informatics
University of Edinburgh
Edinburgh, UK
a.gurung-1@sms.ed.ac.uk, mlap@inf.ed.ac.uk

代码：github.com/Alex-Gurung/ReasoningNCP

简介

针对LLM for长篇小说生成任务，本文受到RLVR在数学和编程任务中的成功实践启发，探索如何将RLVR应用于长小说生成，首先提出一个**新任务Next-Chapter Prediction(NCP)**，模拟人类作者在撰写长篇小说时的创作流程：基于已有故事信息(角色、情节、摘要等)，来规划并生成下一章内容。此外，针对缺乏标注数据无法设计基于规则的reward问题，论文设计了一种reward机制VR-CLI(Verifiable Rewards via Completion Likelihood Improvement)，简单来说就是一种间接的但可量化的方式来训练推理模型，即如果一个写作计划能让生成器模型更容易写出接近原作的下一章内容，就说明它是有效的，从而赋予reward。

背景

长篇小说生成(Long-form Story Generation)任务指的是让模型生成数千token以上的连贯文本，要求不仅要维持角色一致性和情节发展，还要具备语言风格和叙事节奏的统一。对于长文生成类型的任务，都存在ground truth难以确定的问题，因此目前主流做法是prompt engineering + LLM。

本文受到RLVR在数学和代码任务上成功应用的启发，探索将RLVR用于长小说生成任务，难点在于如何设计reward function

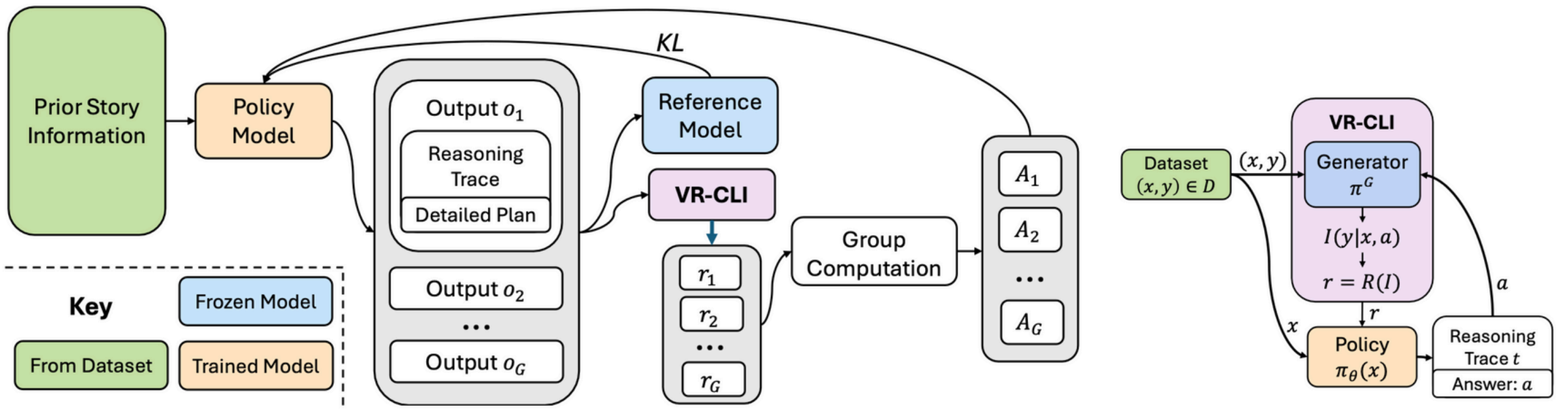
训练框架

实验设置

- 下一章预测任务的输入：小说大纲、下一章纲要、前面章节摘要、上一章内容、基于前文的人物表。
- 可以把上面的内容作为prompt提供给llm生成下一章内容，但是作者进一步假设，先把上面的内容输入给reasoning llm生成写作计划(plan)，再把上面内容+plan输入给生成器llm去生成下一章内容。
- 只用RLVR tuning reasoning llm, 不tuning 生成器llm
- reward function：

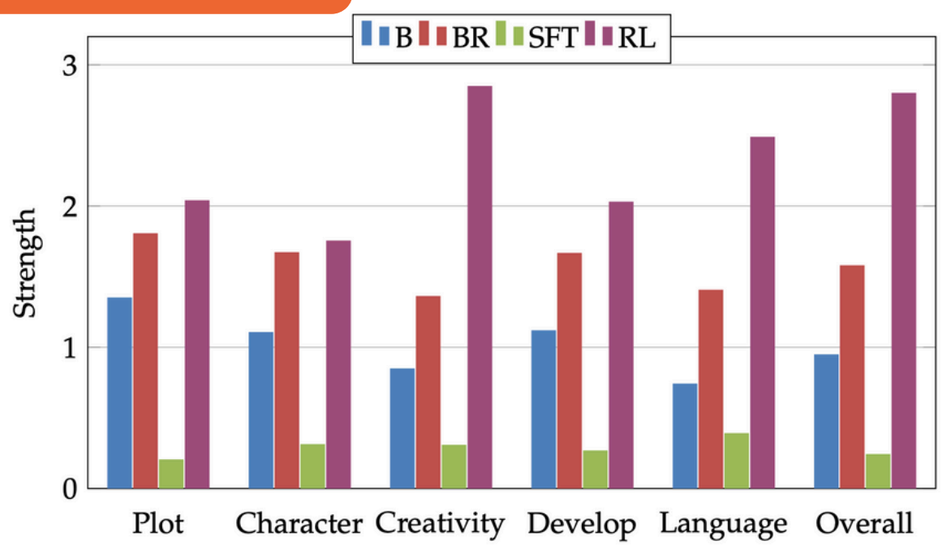
$$R(x,y,a)=\begin{cases}\alpha & I(x,y,a)\le\omega_0\\ \beta & \omega_0<I(x,y,a)\le\omega_1\\ \gamma & \omega_1<I(x,y,a)\end{cases}\quad(5)$$

$$R(x,y,a)=\max[0,I(x,y,a)]\quad(6)$$



(a) NCP GRPO Training　　(b) VR-CLI Reward

部分实验结果



(a) 7B Bradley Terry Relative Strength

思考

本文至少可以带给我们两点关于RLVR的启发，像数学和编程任务可以比较容易的用规则计算reward function，但是像小说生成这类任务则不行，作者创建了“隐式的reward function”，即将原任务A拆分为两个连续的子任务A1和A2，然后用A2的效果来衡量A1的质量，作为reward function，有点类似“古老的”概率图模型。

第二点启发是长文生成可以拆解为按章节生成，这样就把长文分解为短文生成，在Deep Research的长报告生成中有借鉴意义。