GT

-自动程序修复（APR）工作

-背景

-已有SOTA：缺乏数据，修复种类有限

-LLM经过大量代码预训练，在APR有潜力

-缺乏最先进LLM在APR应用的研究

-方法

-9个不同规模LLM，参数量125M-20B

-种类：生成、填空

-3种任务格式：修改整个函数、填空、输出单行

-指标：bug修复数量、生成速度、编译率

-贡献

-发现了

-LLM表现远优于SOTA

-规模效应

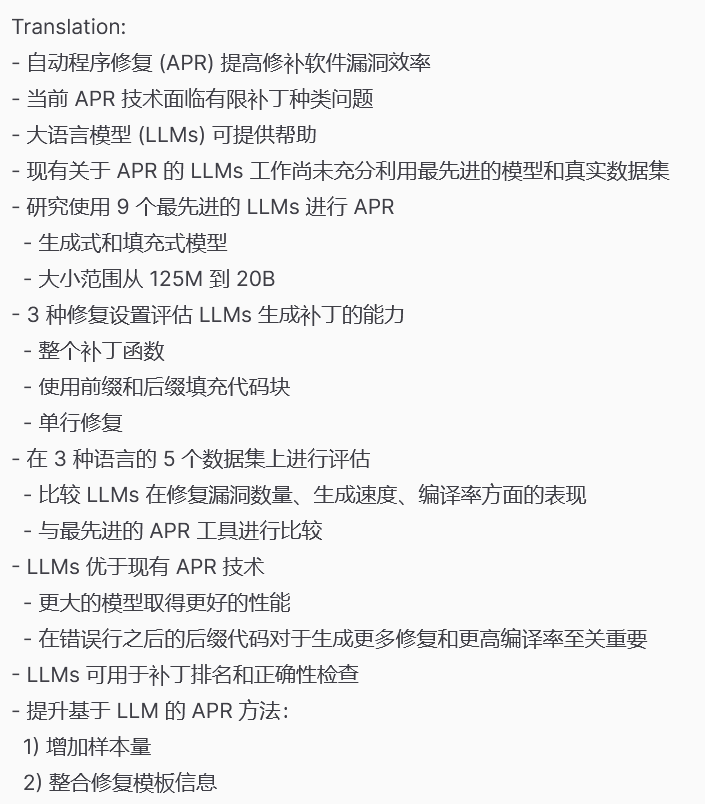
-错误行的后缀对修复数量和编译率影响大

-LLM判断正确补丁更自然，可用于补丁推荐的排名或正确性检查

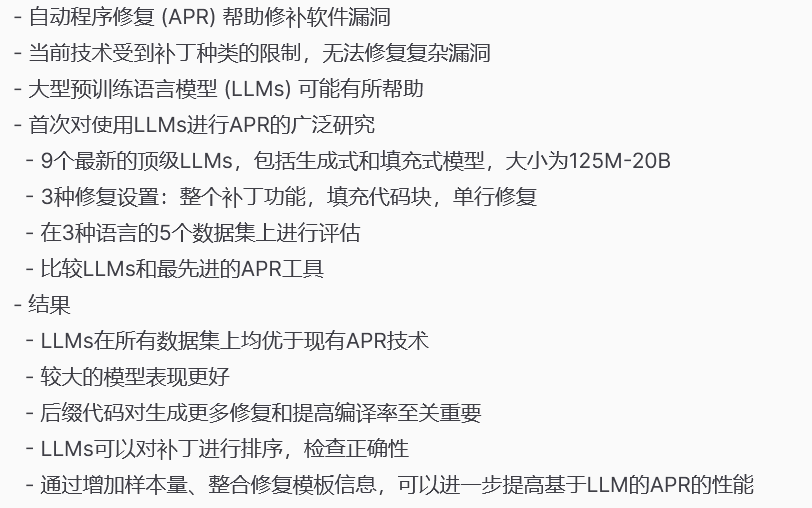
-展望

-增加样本、融合模板

删除式：



生成式：



小绿鲸：

本文介绍了在大型预训练语言模型时代进行自动程序修复的方法。研究团队提出了几个研究问题，并使用不同的APR基准进行评估。他们实现了一个基于Python的生成流程，并使用PyTorch和Hugging Face加载模型权重和生成输出。他们还介绍了使用的数据集和评估结果。该研究对于理解和应用大型预训练语言模型在自动程序修复领域的潜力具有重要意义。

忠诚性：1-（总结∉GT）/总结

全面性：总结∈GT/GT

简洁性：总结点数/总结字数 ÷ GT点数/GT字数

删除式

忠诚性：1 - 1/15

全面性：1

简洁性：14/274 ÷ 14/202 =0.737

生成式

忠诚性：1-1/15

全面性：1

简洁性：14/240 ÷ 14/202 =0.842

小绿鲸

忠诚性：1-3/5

全面性：2/14

简洁性：2/148 ÷14/202 = 0.487