Dynamic Scoring Code Token Tree: A Novel Decoding Strategy for Generating High-Performance Code

曲慕子，刘杰，亢良伊，王帅，叶丹，黄涛

39th IEEE/ACM International Conference on Automated Software Engineering, 2024

联系方式：刘杰 ljie@otcaix.iscas.ac.cn

问题分析

主流代码大模型仅关注代码生成正确性，无法满足科学计算、大数据分析等任务对代码执行性能需求。

例如：向量正则化问题，向量化方法实现的代码执行效率远高于迭代方法实现的代码。

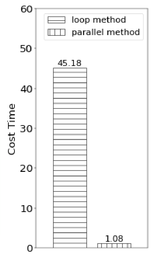
实验证明，CodeLlama34B生成10次代码，其中4例执行错误，5例使用迭代方法实现，仅有1例使用向量化方法实现。因此，需要设计策略引导大模型。

```
def g(df):
    for col in df.columns[1:]:
        df[col] = df[col]/df[col].sum()
    return df
```

Iterative Method

```
def g(df):
    df = df.set_index('cat')
    res = df.div(df.sum(axis=0), axis=1)
    return res.reset_index()
```

Vectorization Method



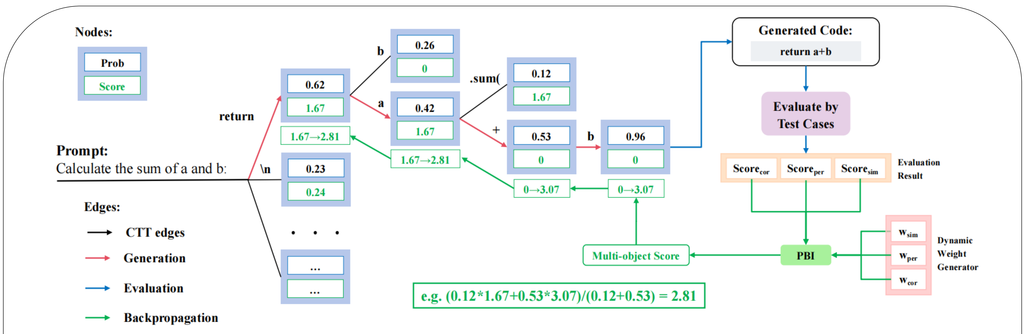
研究目标

论文设计基于代码令牌树（CTT）的生成高性能代码的解码策略DSCT-Decode，Pass@k场景下基于执行反馈动态引导大模型生成高性能代码。核心思想：

（1）在代码令牌树中记录各token对应的多目标评分期望；

（2）权衡考虑多目标评分期望与候选概率以动态优化解码策略；

方法流程



Generation：基于CTT生成代码样本，同时在该过程中更新增枝CTT；

Evaluation：使用测试用例评估代码，从不同角度分析评估结果；

Backpropagation：沿路径反向传播多目标评分，更新路径上各节点的分数；

实验分析

K	Model	Decoding Strategy	Average Relative Latency (ARL)	Performance Improvement Ratio (PIR)	Better Ratio	Worse Ratio
K=10	CodeLlama 7B	DSCT-Decode	1.69	**28.1%**	36.7%	1.7%
		Standard	2.24			
	CodeLlama 13B	DSCT-Decode	1.29	**42.7%**	34.8%	6.1%
		Standard	1.99			
	CodeLlama 34B	DSCT-Decode	1.46	**17.2%**	43.8%	3.4%
		Standard	1.73			
K=40	CodeLlama 7B	DSCT-Decode	1.36	**65.8%**	44.3%	0%
		Standard	2.69			
	CodeLlama 13B	DSCT-Decode	1.15	**54.9%**	35.0%	2.6%
		Standard	2.02			
	CodeLlama 34B	DSCT-Decode	1.22	**31.0%**	57.3%	2.2%
		Standard	1.67			
K=100	CodeLlama 7B	DSCT-Decode	1.29	**59.8%**	44.4%	0%
		Standard	2.38			
	CodeLlama 13B	DSCT-Decode	1.11	**44.1%**	30.5%	3.5%
		Standard	1.74			
	CodeLlama 34B	DSCT-Decode	1.22	**41.3%**	50.7%	2.7%
		Standard	1.87			

- 大多数场景DSCT-Decode提升超过30%的平均性能
- DSCT-Decode有效降低超过30%的代码任务的延迟

仅要求生成符合要求的正确的代码 → GPT-4-gen

要求生成符合要求的高性能的代码 → GPT-4-inst

要求在性能方面优化已正确生成的代码 → GPT-4-opt

Method	ARL	PIR	Better Ratio	Worse Ratio	$\|CorSet\|$
DSCT-Decode	1.56	**47.6%**	61.7%	0%	81
GPT-4-gen	2.53				
DSCT-Decode	1.29	**27.8%**	57.4%	3.7%	54
GPT-4-inst	1.71				
DSCT-Decode	1.62	**14.2%**	47.3%	4.1%	74
GPT-4-opt	1.87				

- 与GPT-4提示工程比， DSCT-Decode提高45%以上任务的性能，且提高10%以上的平均性能；
- 适当的提示可以提高GPT-4生成代码的执行性能，但可能导致正确率下降。

工作贡献

1、开发了一种针对大语言模型的代码生成新型解码策略DSCT-Decode，利用生成代码的评估结果来优化代码生成流程。

2、提出了一个旨在评估代码生成方法的综合基准测试，重点关注执行性能的基准测试数据集big-DS-1000。

3、使用包括CodeLlama和CodeGeeX在内的多个大语言模型进行了测试，平均性能提升近30%，验证了其有效性和实际应用潜力。