

AlphaEvolve: A coding agent for scientific and algorithmic discovery

AlphaEvolve: llm + 演化算法 for 科学/工程问题

Alexander Novikov*, Ng n V * , Marvin Eisenberger*, Emilien Dupont*, Po-Sen Huang*, Adam Zsolt Wagner*, Sergey Shirobokov*, Borislav Kozlovskii*, Francisco J. R. Ruiz, Abbas Mehrabian, M. Pawan Kumar, Abigail See, Swarat Chaudhuri, George Holland, Alex Davies, Sebastian Nowozin, Pushmeet Kohli and Matej Balog*
Google DeepMind¹

简介

本文提出AlphaEvolve，一个结合llm和演化算法，用生成的代码来解决科学和工程问题的coding agent。简单来说，AlphaEvolve用于那些可以用代码(函数)来作为solution并且可以自动evaluationsolution的科学/工程问题，因为这些问题往往超级复杂，llm生成一次response就得到高效的solution几乎是不可能的，为此延续了演化算法+llm的框架，首先人来提供一个初始方案(函数/程序方式的solution)然后指定要演化优化的代码块(可以是多个)，然后AlphaEvolve进行迭代演化得到最终的求解方案。我们从演化算法三要素(变异、评估、选择)角度来看下AlphaEvolve: 1) 变异阶段，AlphaEvolve通过llm ensemble (Gemini Pro+Flash) 生成程序的变种；2) 评估阶段，系统通过自动打分函数对生成的代码进行打分；3) 选择阶段，平衡当前分数最高以及搜索多样性。

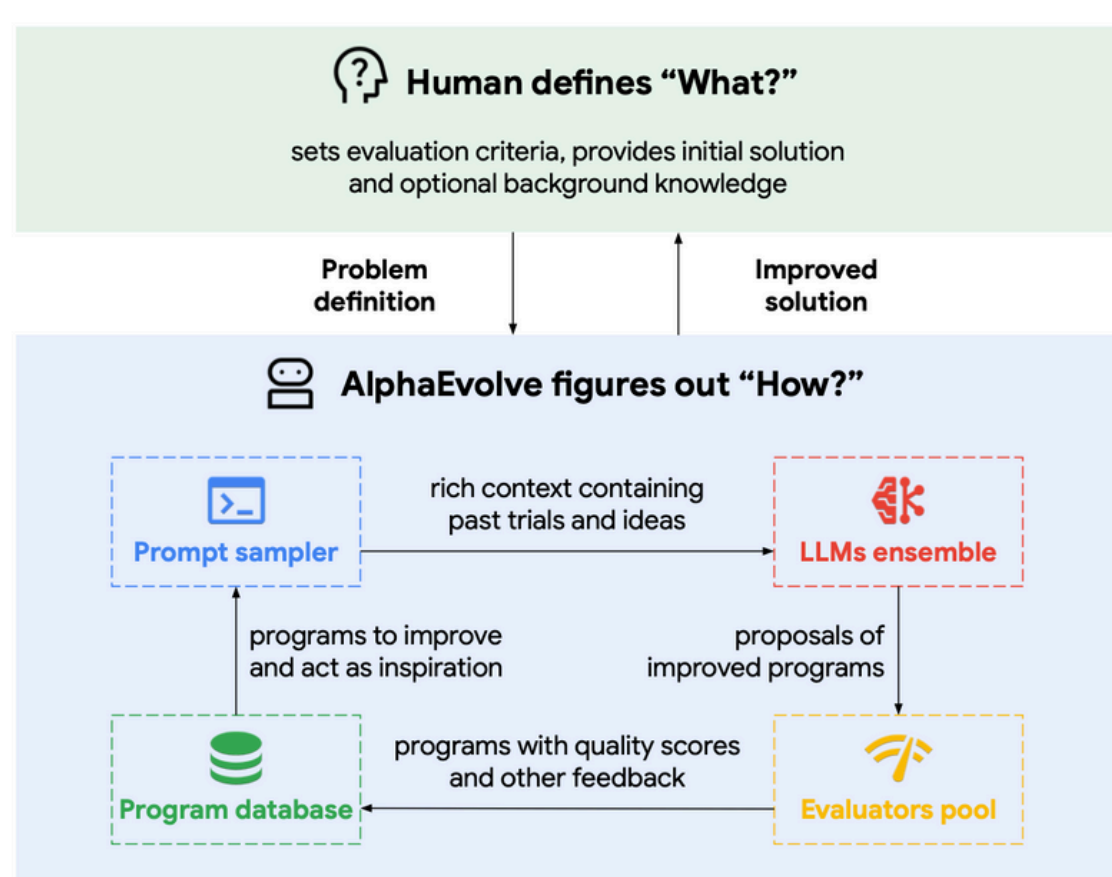
背景

尽管llm的能力(比如coding)不断提升，但对于复杂的科学和工程问题，想通过生成一次response就解决问题是远远不可能的，为此，一些工作开始探索将演化算法(Evolutionary Algorithms)与llm结合，以支持迭代试错、候选筛选和逐步优化的流程，本文提出的AlphaEvolve属于这一研究方向的重要延续，与早期工作相比，在效果方面有了明显提升。

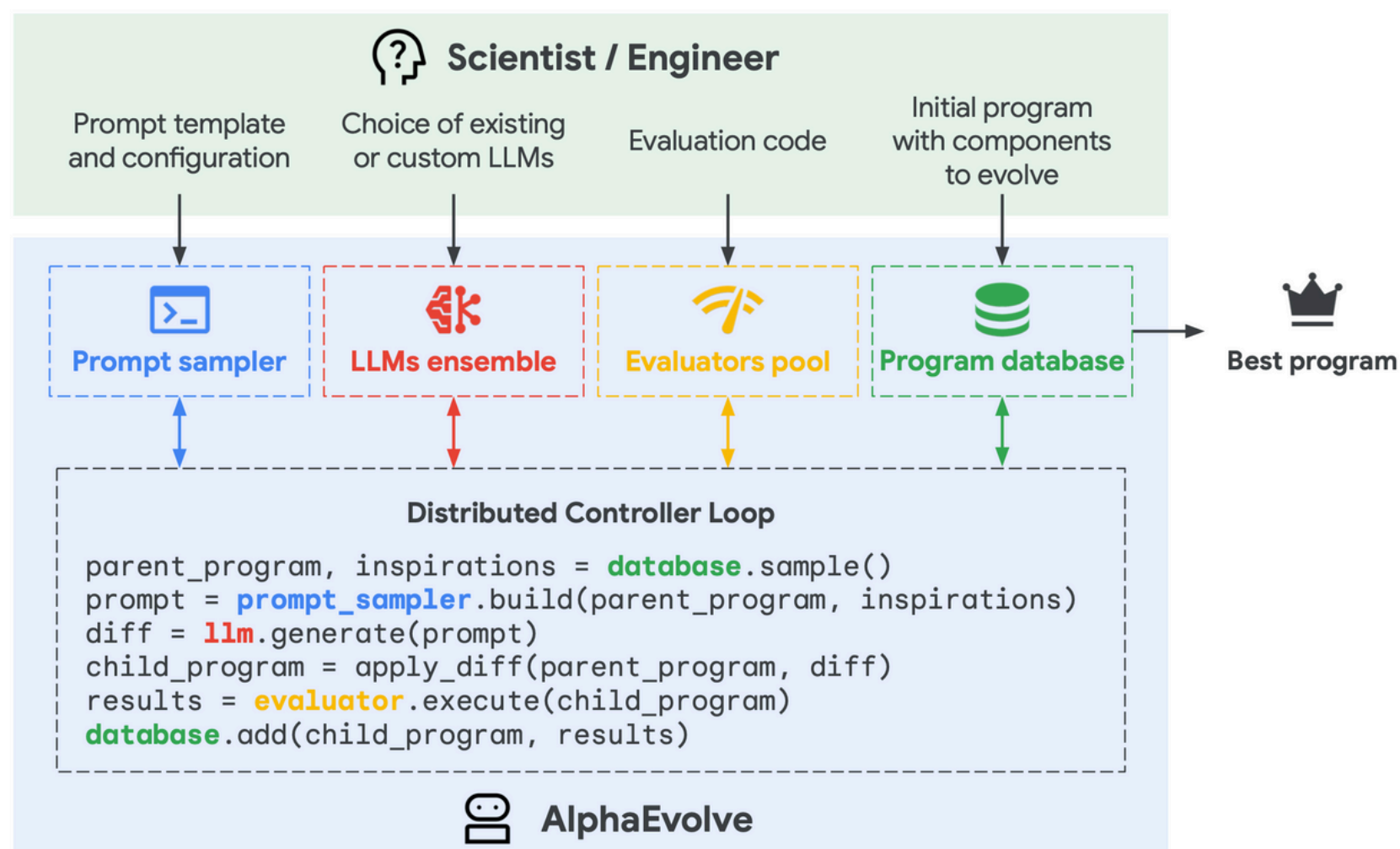
实验设置

- llm: Gemini 2.0 Flash和Gemini 2.0 Pro
- 应用案例: Google内部数据中心调度算法优化、矩阵乘法
- 演化算法: 结合MAP elites algorithm和island-based population models

AlphaEvolve框架



抽象版



详细版

思考

Google的Alpha系列论文评价不了，反正用来打发时间顺便拓宽思路准没错。一点感想就是好像DeepMind一直在探索search as intelligence，不管是下围棋、蛋白质结构还是矩阵乘法，都可以看作“有反馈的结构空间里做高效搜索”。不过仍然有个老问题，AlphaEvolve依赖automatic evaluation(自动评估函数)，对于那些目标不容易量化的任务，就不适用了，其实RL(VR)也有这个问题，依赖reward model/function，如何构造更加general的方法是一个有趣的问题。