专利申请技术交底书

一、专利名称	**基于大模型的机器学习流水线自动化**		
二、专利的相关信息			
专利类型	☐发明专利 ☒实用新型专利 ☐发明+实用新型		
是否费减	☐是 ☐否		
申请人			
发明人			
第一发明人	姓名	身份证号	国籍
技术联系人	姓名	电话	电子邮箱
三、缩略语和关键术语定义 **（请列出本专利中所出现的缩略的英文全称及中文定义（没有可不填写））**			
LLM： Large Language Model 大语言模型 ML: Machine Learning 机器学习			
四、专利背景（与本专利相近的已有技术及其缺陷描述）			
已有技术描述	写明已有技术中与本专利技术最接近的方案，有附图请结合附图描述 **Data Interpreter 介绍** Data Interpreter 是一种新型的基于 LLM 的数据科学解决方案，旨在通过自动化的方法来调整和优化数据。它强调了三种关键技术来增强数据科学问题解决能力：1）动态规划与层次图结构，以实现实时数据适应性；2）动态集成工具以提高执行期间的代码熟练度，丰富所需的专业知识；3）在反馈中识别逻辑不一致性，并通过经验记录提高效率。Data Interpreter 在各种数据科学和现实世界任务上进行了评估。与开源基线相比，它表现出了优越的性能，在机器学习任务中的表现从 0.86 提高到了 0.95，在 MATH 数据集上提高了 26%，在开放式任务中的表现提高了 112%。 **AutoSklearn 介绍** Auto-Sklearn 是一个开源库，用于在 Python 中执行自动化机器学习（AutoML）。它利用流行的 Scikit-Learn 机器学习库进行数据转换和机器学习算法。Auto-Sklearn 由 Matthias Feurer 等人开发，它基于 Scikit-Learn，使用 15 个分类器、14 个特征预处理方法和 4 个数据预处理方法，产生具有 110 个超参数的结构化假设空间。 Auto-Sklearn 的好处在于，除了发现为数据集执行的数据预处理和模型之外，它还能够从在类似数据集上表现良好的模型中学习，并能够自动创建性能最佳的集成作为优化过程的一部分发现的模型。 Auto-Sklearn 是改进了一般的 AutoML 方法，自动机器学习框架采用贝叶斯超参数优化方法，有效地发现给定数据集的性能最佳的模型管道。它还包括了一个用于初始化贝叶斯优化器的元学习方法。		

已有技术的问题及其缺陷描述	①客观评价，现有技术方案的缺点是相对于本专利的优点来说的，本专利不能解决的缺点不必写； ②不能单纯讲缺陷，要结合产生缺陷的原因来描述。
	Data Interpreter： 作为 LLM agent 未能充分利用 LLM 的 planning 能力，未能搜索机器学习任务的最佳配置，只是列出需要执行的 task 列表并按照顺序执行 生成代码缺乏多样性，而且即便经过多次迭代后代码和配置还是次优的，无法进一步提升 **AutoSklearn：** 优化目标是固定的流水线的组件选择，比如模型选择，超参优化和集成学习设置，不够灵活 在数据清洗方面还需要人为参与，目前对非数值型数据不友好，这可能限制了它在处理某些类型数据时的自动化程度。

五、本专利技术方案的详细阐述

5.1 本专利所要解决的技术问题

针对现有技术方案的缺陷，说明本专利要解决的技术问题

增强的规划能力：Data Interpreter 作为 LLM agent，未能充分利用 LLM 的规划能力，只是列出需要执行的任务列表并按顺序执行，缺乏对机器学习任务最佳配置的搜索。SELA 通过引入蒙特卡洛树搜索（MCTS）来优化 AutoML 过程，使得代理能够智能地进行实验并迭代地完善策略，从而更有效地探索机器学习解决方案空间。

代码多样性和迭代优化：SELA 通过树搜索结构化的搜索空间，允许在每个阶段智能地探索和生成机器学习解决方案，从而提高了代码的多样性，并能够通过实验反馈迭代优化解决方案。

灵活性和适应性：SELA 通过动态和多样化的策略，允许系统根据中间结果调整工作流程，使其能够适应新信息，从而提高了适应性。

综合阶段规划和迭代细化：SELA 结合了分阶段规划和迭代细化机制，使得在每个阶段都能探索更好的解决方案。这种方法提供了更大的灵活性和控制力，使得在搜索过程中能够生成优化的解决方案。

实验状态保存与加载：为了提高实验效率并减少令牌使用，SELA 实现了细粒度的代码重用，通过缓存每个尝试配置的阶段代码，允许框架在新配置与现有配置共享组件时重用尽可能多的保存代码。这不仅提高了资源利用率，还解决了 LLM 非确定性问题，确保了性能的一致性。

5.2 本专利提供的完整技术方案（请结合附图说明）

①专利是关于结构的，请描述本专利所包含的各个元件，各元件之间的结构关系或者电路连接关系，描述专利技术的工作原理（有附图的请结合附图来描述）

②专利是关于方法或流程的，请描述该方法或者流程所包括的所有步骤以及各步骤的详细情况，涉及软件开发过程的，以描述流程图及相关的设计说明为主（有流程图的请务必将流程图附上）

1. **系统组成：**

本专利包括以下主要组件：

大型语言模型（LLM）：用于生成机器学习任务的洞察和代码。
蒙特卡洛树搜索（MCTS）模块：用于在树状结构的搜索空间中进行决策，以优化机器学习流程。
实验执行器（Experiment Executor）：根据 LLM 生成的配置执行机器学习实验，并反馈结果。

2. **工作原理：**

洞察提议（Insight Proposal）：LLM 根据问题描述和数据集信息生成一系列洞察，这些洞察构成了搜索空间。
树状搜索空间构建：MCTS 模块将洞察组织成树状结构，每个节点代表一个潜在的解决方案路径。
选择（Selection）：使用修改版的 UCT 算法（UCT-DP）从搜索树中选择一个节点进行探索。
扩展（Expansion）：从选定节点生成子节点，代表不同的模型规范和训练策略。
模拟（Simulation）：实验执行器根据选定的配置执行完整的机器学习实验，并产生评估分数。
反向传播（Backpropagation）：将模拟结果的绩效分数反馈回搜索树，更新节点的值和访问次数。

3. **方法流程：**

输入问题描述和数据集信息：提供给 LLM 以生成搜索空间。
生成搜索空间：LLM 根据输入生成洞察，形成搜索空间。
初始化树结构：使用洞察初始化 MCTS 的树结构。
执行多次模拟：通过选择、扩展、模拟和反向传播步骤，进行多次模拟以探索不同的配置。
选择最佳解决方案：根据模拟结果选择表现最佳的节点作为最终解决方案。

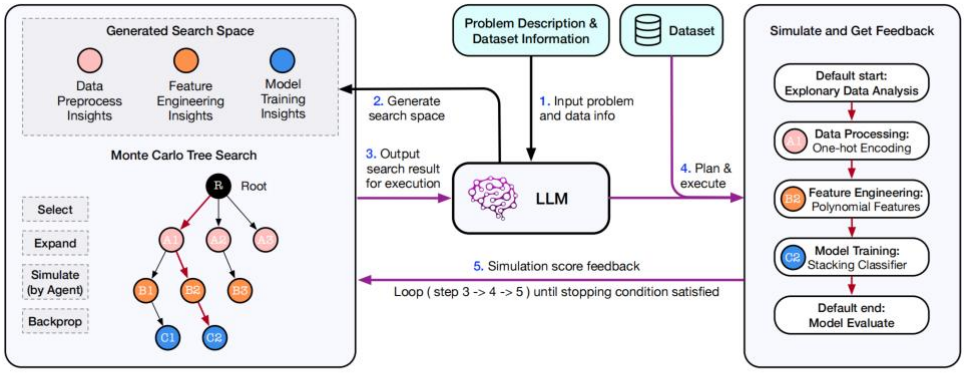

Figure 2: SELA's pipeline operates as follows: The system begins by inputting the problem description and dataset information into the LLM, which generates a search space of potential solutions, encompassing data preprocessing, feature engineering, and model training. The search module, powered by Monte Carlo Tree Search (MCTS), explores this space by selecting, expanding, and simulating potential configurations. The LLM agent then simulates the selected configuration by planning, coding, and executing the experiment. Feedback from the simulation is fed back into the search module, where it is used in the backpropagation step to refine future searches. This iterative process continues until a predefined stopping criterion is met, resulting in an optimized experimental pipeline.

5.3 本专利技术方案带来的有益效果

发明的效果，是与背景技术相比较的结果。应当清楚、有根据地说明发明与背景技术相比所具有的优点和（或者）

积极效果。例如：产率、质量、精度、效率等的提高，能量、原材料的节省，环境污染的改善等等。

提高自动化水平：SELA 通过整合 LLM 代理和 MCTS，能够自动化地探索和优化机器学习管道，减少人工干预，使得非专家用户也能有效地设计可靠的机器学习解决方案。

增强搜索效率：通过 MCTS，SELA 能够在庞大的决策空间中高效地导航，平衡探索（测试新策略）和利用（改进已知良好策略）之间的关系，从而更智能地选择下一个有前景的配置进行测试。

提升解决方案质量：SELA 通过迭代实验和策略优化，能够逐步改进解决方案，类似于人类专家基于持续反馈测试和改进策略的过程，从而发现基于实验反馈的最优路径。

适应性和灵活性：SELA 能够根据中间结果调整工作流程，允许在新信息出现时适应性地调整，这在处理独特数据集或特定任务要求时尤为重要。

多样化的解决方案：SELA 生成的解决方案具有多样性，因为它能够迭代地探索和改进整个机器学习管道，而不是仅限于单次尝试或固定的管道。

广泛的适用性：虽然 SELA 是为机器学习设计的，但其核心方法可以适应广泛的顺序决策问题，只要这些问题可以表示为具有标量奖励的树结构。

实验验证的有效性：在 20 个机器学习数据集上的广泛评估表明，SELA 在各种机器学习任务中都能持续提供优越的性能，与传统的 AutoML 系统和现有的 LLM-based 方法相比，SELA 在所有数据集上实现了 65%到 80%的胜率。

成本效益：SELA 实现了在保持低计算成本的同时找到准确的模型，这对于需要快速部署和更新模型的应用场景（如数据库系统）非常有用。

可扩展性：SELA 的设计允许它在不同的大型语言模型上表现一致，这表明它可以根据不同用户的偏好和可用性灵活地使用不同的 LLM。

透明度和信任度：通过开发技术来提供关于搜索过程和解决方案理由的可解释性解释，SELA 可以增强系统的透明度和信任度。

六、针对第五部分的技术方案，是否还有别的替代方案同样能完成专利目的

除了 MTCS 可以使用其他的搜索或者规划方法，例如 D*算法，它在复杂度和实现难度上与 MCTS 相当，适用于解决开放型问题的寻路。MCTS 也可以通过调整参数来适应不同类型地图的寻路问题。

可以将强化学习中的规划方法运用到流水线结构的决策过程中来提高多样化和解决方案效果.

可以将实现中的 LLM 换为本地大语言模型（如 Ollama），这可以方便使用者更方便的配置及推理，并降低成本.

七、本专利的技术关键点和欲保护点是什么

结合大型语言模型（LLM）与蒙特卡洛树搜索（MCTS）

树状搜索空间的构建

迭代反馈驱动的搜索过程：技术关键点在于如何通过实验反馈来迭代地改进搜索策略，从而逐步提高解决方案的质量。它模拟了人类专家在解决问题时的思考和迭代过程。

实验状态保存与加载机制：技术关键点在于如何通过缓存代码和重用先前的状态来提高实验效率和减少资源消耗。欲保护的点是这种状态保存与加载机制，它允许系统在新的配置与现有配置共享组件时重用尽可能多的保存代码。

自适应的搜索策略

八、参考文献（如专利/标准/论文等）

https://arxiv.org/abs/2410.17238

https://arxiv.org/abs/2402.18679

https://www.weco.ai/blog/technical-report

https://arxiv.org/abs/2007.04074