

元级优化系统：从基本原理到智能的自动化设计

第一部分：概念与理论基础

第1节 优化的层级：对象级与元级

要深入理解元级优化系统，首先必须建立一个贯穿始终的核心概念框架：优化的层级结构。这一框架将优化过程划分为两个截然不同的抽象层面——对象级(object-level)和元级(meta-level)。元级优化并非仅仅是一种技术，它代表了我们解决问题时思维方式的根本性转变：从专注于解决一个具体问题，转向致力于改进解决问题的过程本身。

1.1 定义抽象层级

对象级与元级的区别在于抽象的层次。对象级通常关注手头的具体问题，而元级则涉及一般性原则、“关于论证的论证”或“关于思考的思考”¹。

- 对象级 (Object-Level)**: 该层面直接处理和优化特定的系统或问题实例。其操作和论证都围绕着领域的“对象”展开。例如，在一个数据集上训练单个神经网络以最小化损失函数、为某个特定项目制定详细的执行计划、或使用一套固定的公理系统证明一个具体的数学定理，这些都属于对象级活动¹。在一个对象级马尔可夫决策过程(MDP)中，状态描述的是外部环境，而行动则是物理动作⁴。
- 元级 (Meta-Level)**: 该层面操作于支配对象级的过程、原则或系统之上。它旨在优化、改进或选择对象级的策略。例如，参加一个项目管理课程以提升未来所有项目的规划能力、基于“绝大多数科学家都同意”这一元级原则来论证气候变化问题(而非仅仅引用温室效应这一对象级事实)、或研究一个逻辑系统本身的数学属性(如可靠性和完备性)，这些均是元级活动

¹。在一个元级马尔可夫决策过程 (metalevel MDP) 中, 状态编码的是智能体自身的信念, 行动则是计算 (即“思考”), 而奖励函数则包含了思考的成本与收益 ⁴。

从哲学思辨到工程实践的演进, 标志着一个关键趋势的形成: 对抽象推理过程的严格量化与自动化。最初, 如在LessWrong等社区的讨论中, 元级与对象级的区分是一种有用的思维模型, 帮助人们构建更有力的论证 ¹。然而, 随着领域的发展, 这一概念被赋予了坚实的数学基础。元级MDP的提出, 将“关于推理的推理”这一过程本身形式化为一个可以求解的优化问题, 为计算分配了成本 (

rmeta), 并对计算如何更新信念进行了建模 (Tmeta) ⁴。同样, 逻辑学中对证明系统属性 (元级属性) 的数学证明, 也体现了这种形式化 ³。这一转变至关重要, 它意味着“元级”不再仅仅是一个观察视角, 而是一个可以被探索、被量化、被优化的具体“搜索空间”。正是这种从概念到计算的飞跃, 为本报告后续讨论的所有先进技术 (如元学习、AutoML) 奠定了理论基石。

1.2 自然与人工系统中的元级动态

在地球的优化历史中, 无论是自然演化还是人类科技进步, 都存在一个显著特征: 一个相对稳定、受保护的元级优化器在后台运行, 驱动着对象级的加速发展 ⁵。

- 在生物演化中, 自然选择扮演着元级优化器的角色, 它作用于基因层面, 而生物体 (如恐龙、蝴蝶、猫) 则是被优化的对象级产物。自然选择这一元级过程本身 (例如, 通过有性繁殖等机制的出现) 演变得极其缓慢, 但在其主导下, 物种的演化 (对象级优化) 却在不断加速 ⁵。
- 在人类科技史中, 人脑扮演了一个“受保护的解释器”角色, 是元级优化器。科学方法、语言等工具的发明是对这个元级优化器的微小改进, 而每一次这样的改进都开启了一个全新的科学发现 (对象级创新) 加速的时代。然而, 人脑本身的基本结构和认知能力在历史尺度上变化不大 ⁵。

这种“受保护的元级”模型揭示了一个深刻的现象: 在特定时期内, 优化过程的加速度主要体现在对象级产物的累积上, 而优化器本身保持不变。人工智能领域的真正革命, 以及元级优化系统的核心目标, 正是要打破这种“保护”状态。一个“完全自循环的递归优化器” (fully wraparound recursive optimizer) 将不再有任何受保护的、不被优化的部分; 所有优化的结构本身都将成为被优化的对象 ⁶。这预示着一种根本性的范式转移, 其潜在的加速能力不再是线性的, 而是指数级的, 因为对优化过程的每一次改进都会复利式地影响所有未来的优化结果。这直接将元级优化的技术与关于人工智能驱动的“智能爆炸”的深远讨论联系在一起, 为理解其颠覆性潜力提供了强有力的理论视角。

第2节 定义元优化与元学习范式

在建立了元级与对象级的概念框架后, 本节将提供元优化及其在机器学习领域最重要分支——元学习——的正式定义, 并阐明其核心词汇与框架。

2.1 元优化: 形式化问题

元优化 (Meta-optimization) 在数值优化领域被正式定义为: 使用一种优化方法 (称为元优化器) 来调整另一种优化方法 (称为对象级或基础优化器) 的参数或组件的过程⁷。

这一概念的提出, 主要是为了解决手动调整优化器参数所面临的巨大挑战。许多优化算法, 如遗传算法 (Genetic Algorithm) 和差分进化 (Differential Evolution), 包含多个“行为参数” (behavioural parameters), 这些参数的选择直接决定了算法在特定问题上的性能和效率。手动选择这些参数不仅耗时费力, 而且容易受到人类对优化器行为的错误认知的影响⁷。更严重的是, 随着参数数量的增加, 评估所有可能组合的计算成本呈指数级增长, 这就是参数搜索空间中的“维度灾难” (curse of dimensionality)⁷。元优化通过将基础优化器的参数调优问题本身视为一个新的、更高层次的优化问题, 并使用元优化器来高效地搜索这个参数空间, 从而系统性地解决了这一难题。

在文献中, 元优化及其相关概念也被称为元演化 (meta-evolution)、超级优化 (super-optimization)、自动化参数校准 (automated parameter calibration) 和超启发式 (hyper-heuristics) 等⁷。

2.2 元学习: “学习如何学习”

元学习 (Meta-learning) 是元优化在机器学习领域的核心体现和主导范式⁸。其核心目标是设计能够通过从多个学习任务 (episodes) 的经验中学习, 从而改进其自身学习过程的算法¹⁰。这一范式常被通俗地描述为“学习如何学习” (learning to learn)¹⁰。

与传统机器学习模型从零开始解决单一任务不同, 元学习模型旨在获得一种能够快速适应全新、未曾见过的任务的能力, 且通常只需要极少量的训练数据¹⁰。这好比教一个人学习新技能的方法, 而不是只教他一项具体的技能。通过在大量不同任务上进行训练, 元学习模型能够掌握跨任务的通用知识和高效的学习策略, 从而在遇到新任务时, 能够比从头学习的传统模型更快、更数据高效地达到理想性能¹⁰。

2.3 双层优化框架

大多数现代元学习算法的结构都遵循一个双层优化 (bi-level optimization) 框架, 该框架包含两个嵌套的循环¹⁰。

- 内循环(基础学习器): 在内循环中, 一个模型(称为基础学习器, base-learner)在一个特定任务 \mathcal{T}_i 的“支持集”(support set, 即该任务的训练数据)上进行训练。这是一个标准的对象级优化过程, 例如, 通过梯度下降最小化分类任务的交叉熵损失函数, 从而使模型参数 θ 适应当前任务, 得到适应后的参数 θ'_i ¹²。
- 外循环(元学习器): 在外循环中, 已经适应任务 \mathcal{T}_i 的基础学习器(参数为 θ'_i)的性能在该任务的“查询集”(query set, 即该任务的验证数据)上进行评估。元学习器(meta-learner)的目标是最小化这个在查询集上的损失。这个外循环的损失函数会根据所有元训练任务的性能进行聚合, 然后元学习器使用这个聚合损失来更新基础学习器的初始状态(例如, 初始参数 θ)。这个元更新步骤并非为了在任何单一任务上表现完美, 而是为了优化学习过程本身, 使得从初始状态 θ 出发, 基础学习器能够更快、更好地适应任何新任务¹⁰。

这个双层结构是“学习如何学习”的具体实现。内循环执行学习, 外循环评估并改进学习过程, 通过在大量任务上的迭代, 系统最终学会了一种高效的学习策略。

2.4 元学习的定位: 比较分析

为了更清晰地界定元学习的范畴, 我们将其与几个密切相关的领域进行比较。

- 与迁移学习(Transfer Learning)的对比: 迁移学习通常指将在一个大规模数据集(如 ImageNet)上预训练好的模型, 微调(fine-tune)到另一个相关的目标任务上。其核心是迁移知识(“是什么”), 例如, 已经学到的图像特征¹⁰。而元学习的核心是迁移学习过程(“如何做”), 例如, 学习一个能够快速适应任何新任务的良好模型初始化参数, 或一种高效的优化策略。元学习更关注适应性本身, 而非直接的知识转移¹⁰。
- 与多任务学习(Multi-Task Learning)的对比: 多任务学习的目标是训练一个单一模型, 使其能够同时在一组固定的、已知的任务上都表现良好。所有任务的数据通常被汇集在一起进行联合训练¹⁷。元学习则不同, 它的目标是训练一个模型, 使其具备快速适应未来可能遇到的全新、未曾见过的任务的能力。其评估标准不是在已知任务上的平均性能, 而是在新任务上的适应速度和性能¹⁷。
- 与超参数优化(Hyperparameter Optimization, HPO)的对比: 虽然HPO是元优化的一个经典实例, 但当代神经网络领域的元学习与其有显著区别。传统的HPO方法(如随机搜索或贝叶斯优化)通常将内部的学习过程视为一个黑箱。而现代元学习, 特别是基于梯度的方法, 通过一个明确定义的元级目标函数, 以端到端(end-to-end)的方式对内部学习算法进行优化。这种方法能够将元优化扩展到数百万甚至数亿个参数(例如, 整个神经网络的初始权重), 这

是经典HPO方法难以企及的¹⁸。

第二部分:架构与方法论

第3节 元学习策略分类法

元学习领域已经发展出多种技术路径,它们从不同角度实现了“学习如何学习”的核心思想。这些方法可以被系统地归纳为三个主要类别:基于优化的、基于度量的和基于模型的元学习⁹。这种分类法不仅有助于理解现有算法,也揭示了不同方法在编码“元知识”方面的根本性差异。

3.1 基于优化的元学习:学习一个更好的起点

这类方法的核心思想是,学习一组模型的初始参数 θ ,使得这组参数对于新任务高度敏感,只需通过少数几步梯度下降更新就能快速适应新任务并取得良好性能⁹。在这里,元知识被直接编码为模型的初始参数本身¹⁸。

- 关键算法:
 - **模型无关元学习 (Model-Agnostic Meta-Learning, MAML)**:作为该领域的开创性工作, MAML的框架极具影响力。它通过双层优化循环实现目标:在内循环中,模型从初始参数 θ 出发,针对特定任务的支持集数据进行几步梯度下降,得到适应后参数 θ' ;在外循环中,计算 θ' 在任务查询集上的损失,并将这个损失的梯度反向传播*穿过*整个内循环的优化过程,最终用于更新初始参数 θ 。由于其原理不依赖于具体的模型结构,只要模型能通过梯度下降进行训练,就可以应用MAML,因此得名“模型无关”⁹。
 - **Reptile**: Reptile是MAML的一种简化版一阶近似算法。它避免了MAML中计算元梯度所需的高阶导数,从而大幅降低了计算复杂度。其更新规则很简单:在每个任务上独立进行多步梯度下降后,将初始参数 θ 朝着每个任务最终学到的参数 θ' 的方向移动一小步⁹。
 - **LSTM元学习器 (LSTM Meta-Learner)**:这种方法将学习过程本身参数化。它使用一个循环神经网络(特别是LSTM)作为元学习器,来学习并输出基础学习器的参数更新规则。换言之,元学习器不再是学习一个好的起点,而是学习一个定制的优化算法,该算

法专门为小样本场景设计，能够为基础学习器生成高效的梯度更新量⁹。

3.2 基于度量的元学习：学习一个通用的比较函数

这类方法的目标是学习一个通用的嵌入空间(embedding space)或一个距离度量函数。在这个空间里，可以通过比较新样本与少量有标签的“支持集”样本之间的距离来进行分类或回归，通常在测试时无需对模型进行任何微调⁹。这里的元知识被编码在度量空间的结构中。

- 关键算法：

- 孪生网络(**Siamese Networks**)：该网络由两个共享权重的“孪生”子网络组成。输入一对样本，每个子网络分别提取其特征向量，然后顶层的一个函数会计算这两个向量的相似度得分。通过在大量样本对上训练，网络学会了一个可泛化的度量函数，用于判断新样本对的相似性⁹。
- 匹配网络(**Matching Networks**)：该网络将小样本学习任务视为在支持集上进行的一种“带权重的最近邻”分类。它使用注意力机制来计算查询样本与支持集中每个样本的匹配权重，然后根据这些权重加权支持集样本的标签来做出预测。整个过程在一个端到端学习到的嵌入空间中进行⁹。
- 原型网络(**Prototypical Networks**)：该网络基于一个核心假设：在学习到的嵌入空间中，每个类的所有样本都会聚集在该类的一个“原型”(prototype)周围。这个原型通常被计算为该支持集样本特征向量的均值。对于一个新的查询样本，分类任务就简化为计算其特征向量与哪个类的原型距离最近⁹。
- 关系网络(**Relation Networks, RN**)：关系网络通过一个专门的神经网络模块来端到端地学习一个非线性的深度距离度量。该模块接收查询样本和支持集样本的特征对，并输出一个0到1的“关系分数”，表示它们的相似程度。这种方式比简单的距离计算(如欧氏距离)更具表达力⁹。

3.3 基于模型的元学习：为快速学习而设计的架构

这类方法不依赖于通用的模型架构，而是专门设计一种能够快速吸收新信息并在模型内部更新其状态的特殊网络结构。学习过程被内嵌在模型的前向传播过程中，使其能够在仅有少量样本的情况下迅速调整其行为⁹。

- 关键架构：

- 记忆增强神经网络(**Memory-Augmented Neural Networks, MANN**)：这类模型，如神经图灵机(Neural Turing Machine)，配备了外部记忆模块。模型可以通过读写操作与这个记忆模块交互，从而将任务相关的信息快速存储起来，并在需要时检索。这种机制使得模型能够在不改变其网络权重的情况下，仅通过更新记忆内容来适应新任务⁹。

- 元网络(**Meta Networks, MetaNet**): 元网络通过学习一个跨任务的元级知识表征, 并利用一个独立的“快速参数化”机制, 为基础学习器动态生成权重。这意味着基础学习器的参数不是通过梯度下降慢慢调整, 而是在遇到新任务时被元网络“即时生成”, 从而实现极速的泛化⁹。

3.4 元学习策略的比较分析

这三种元学习策略并非简单的技术堆砌, 它们代表了“学习如何学习”这一理念在不同层面上的实现, 体现了对归纳偏置(inductive bias)的不同编码方式。基于优化的方法将偏置置于一个灵活模型的初始状态中; 基于度量的方法将偏置置于嵌入空间的结构中; 而基于模型的方法则将偏置硬编码在学习器本身的固定架构中。

这种差异揭示了元学习领域一个根本性的权衡: MAML等基于优化的方法具有极高的灵活性(“模型无关”), 但元梯度的计算成本高昂; 基于模型的方法适应速度极快, 但其架构受到特定设计的限制, 通用性较差; 基于度量的方法则在两者之间取得了平衡, 测试时效率高, 但可能对任务间的领域漂移较为敏感。下表对这三种策略进行了系统性比较。

维度	基于优化的方法 (例如, MAML)	基于度量的方法 (例如, 原型网络)	基于模型的方法 (例如, MANN)
核心思想	学习一个好的初始化参数, 以便快速微调。	学习一个通用的嵌入空间/距离函数。	设计一种能够快速吸收新知识的架构。
元知识编码于	模型的初始参数 θ 。	度量/嵌入空间的结构。	模型的内部架构和记忆模块。
适应机制	少步梯度下降(微调)。	非参数比较(例如, 最近邻)。	通过前向传播更新内部状态(无梯度步骤)。
主要优势	灵活性高, 模型无关。	测试时无需针对特定任务进行微调。	适应速度极快。
主要劣势	计算成本高(元梯度)。	可能难以处理任务间的领域漂移。	架构受限, 通用性较差。

主要应用场景	小样本回归/分类, 元强化学习。	小样本图像分类。	对适应速度要求极高的小样本学习。
--------	------------------	----------	------------------

第三部分:元优化的实践支柱:自动化机器学习生态系统

元优化的理论和方法论并非空中楼阁, 它们在现代人工智能领域中有着广泛而深刻的应用。实际上, 几个主流的AI研究方向——超参数优化(HPO)、神经架构搜索(NAS)以及算法选择与配置(ASC)——可以被视为元优化核心原则的大规模、系统化的实践。这些领域共同构成了自动化机器学习(Automated Machine Learning, AutoML)生态系统的基石, 其终极目标是自动化整个机器学习 workflow。

第4节 自动化超参数优化 (HPO)

4.1 HPO问题定义

HPO旨在自动寻找机器学习模型的最佳超参数组合(例如, 学习率、正则化强度、网络层数等), 在给定的数据集上最大化其性能¹⁴。这是一个经典的元优化问题: 对象级过程是模型的单次训练和评估, 而元级过程则是在超参数构成的搜索空间中寻找最优解⁷。手动调参的低效和“维度灾难”问题是驱动HPO发展的核心动力⁷。

4.2 HPO技术的发展

HPO技术经历了从简单暴力搜索到智能高效搜索的演进。

- 经典方法: 网格搜索(Grid Search)和随机搜索(Random Search)是最早期的基线方法。网格搜索穷举所有参数组合, 计算成本极高; 随机搜索则证明在许多情况下比网格搜索更高效, 因为它不会在不重要的维度上浪费过多算力²⁰。
- 基于模型的序列优化(SMBO): 这类更先进的技术通过构建目标函数(例如, 模型验证集上的准确率)的代理模型(surrogate model), 来智能地指导搜索过程, 以期用更少的评估次数找到最优解²³。

- 贝叶斯优化 (**Bayesian Optimization**): 这是当前HPO领域的主流方法。它使用概率模型(通常是高斯过程)来拟合已观测到的“超参数-性能”数据点, 并利用一个“采集函数”(acquisition function, 如预期提升EI)来平衡“探索”(在不确定性高的区域进行尝试)和“利用”(在当前最优解附近进行挖掘)²⁰。贝叶斯优化在处理计算成本高昂的黑盒函数优化问题上表现出色, 其应用已扩展到供应链库存管理等复杂工业场景²⁶。
- 演化算法 (**Evolutionary Algorithms**): 遗传算法(GA)、粒子群优化(PSO)等方法通过维护一个超参数配置的“种群”, 并通过模拟自然选择、交叉和变异等过程, 来“演化”出性能更优的配置²⁰。

4.3 与元学习的联系

元学习可以显著提升HPO的效率。通过在大量历史HPO任务上进行学习, 元学习模型可以掌握关于“哪些超参数组合在哪些类型的数据集上可能表现更好”的先验知识。这种元知识可以用来“热启动”(warm-start)一个新的HPO任务, 即从一个更有希望的区域开始搜索, 而不是从零开始, 从而大大缩短收敛时间²⁹。

第5节 神经架构搜索 (NAS)

5.1 NAS问题: 自动化架构工程

NAS旨在将设计神经网络架构这一过程自动化, 而这项工作传统上极度依赖人类专家的经验、直觉和大量的试错, 耗时且易出错³⁰。NAS被视为AutoML的一个核心子领域, 与HPO和元学习有显著的重叠³¹。其目标是自动发现一个在特定任务上不仅准确率高, 而且计算效率(如低延迟、少参数)也满足要求的网络结构³⁴。

5.2 NAS的三大维度

NAS的研究框架可以被清晰地分解为三个相互关联的核心组成部分³¹。

- 搜索空间 (**Search Space**): 定义了所有可能发现的候选网络架构的集合。一个设计良好的搜索空间可以在不牺牲最终性能的前提下, 显著降低搜索难度。
 - 全局搜索空间: 早期的NAS方法直接搜索整个网络的宏观结构, 例如每一层的类型和连

接方式。这种方式灵活性高，但搜索空间巨大，计算成本极高³⁶。

- 基于细胞的搜索空间(**Cell-Based Search Space**):受现代高性能网络(如ResNet, DenseNet)中重复模块结构的启发, 这种方法不再搜索整个网络, 而是搜索一个小的、可重复使用的计算单元(cell), 然后通过堆叠这些cell来构建最终的神经网络。这极大地缩小了搜索空间, 并使得搜索到的cell具有更好的可迁移性³⁰。
- 搜索策略(**Search Strategy**):指用于在搜索空间中探索和发现优秀架构的算法。它需要在“探索”(尝试新颖的架构)和“利用”(在已知高性能架构的基础上进行改进)之间做出权衡³¹。
 - 强化学习(**Reinforcement Learning**):将NAS视为一个RL问题, 其中一个“控制器”(通常是RNN或LSTM)作为智能体, 其“行动”是生成一个网络架构的描述, 而“奖励”则是该架构在验证集上训练后的性能³⁵。
 - 演化计算(**Evolutionary Computation**):将网络架构视为“个体”, 性能作为“适应度”。通过维护一个架构种群, 并应用选择、交叉、变异等操作, 来演化出更优的架构³⁸。
 - 基于梯度的搜索(**Gradient-Based Methods**):这类方法, 如DARTS, 通过一种巧妙的连续松弛技术, 将离散的架构选择问题转化为连续可微的优化问题, 从而可以直接使用梯度下降来同时优化网络权重和架构参数, 效率极高³⁹。
- 性能评估策略(**Performance Estimation Strategy**):这是NAS流程中的主要瓶颈, 即如何高效地评估一个候选架构的最终性能。从头完整地训练每一个候选架构(vanilla NAS)虽然准确, 但计算成本令人望而却步³¹。

第6节 算法选择与配置 (ASC)

6.1 算法选择问题

算法选择问题最早由Rice在1976年形式化提出, 其核心是在给定一个问题实例时, 从一个算法组合(portfolio)中, 自动选择最适合解决该实例的算法⁴²。这一问题在组合优化领域尤为重要, 例如, 对于NP难的布尔可满足性问题(SAT), 不同的求解器在不同类型的问题实例上表现差异巨大。一个好的算法选择系统可以获得远超任何单个求解器的性能⁴⁴。

6.2 从选择到配置

算法配置(AC)是算法选择的一个自然推广。它不再是从一组离散的算法中选择一个, 而是为一个高度参数化的算法寻找最佳的参数配置⁴³。例如, 一个现代的SAT求解器可能有数十个参数, 正

确的配置能使其求解速度提升数个数量级⁴⁴。AC的目标就是自动化这一繁琐的调优过程。

6.3 ASC中的元学习

现代ASC系统本质上是元学习系统。它们通过学习一个从“问题实例特征”(称为元特征, meta-features)到“算法(或配置)性能”的映射关系来实现自动化决策⁴²。

- **工作流程:**在离线训练阶段,系统会收集大量问题实例,并记录不同算法或配置在这些实例上的性能(如运行时间、解的质量)。然后,它提取每个实例的元特征(如变量数、约束密度等),并训练一个机器学习模型(元学习器)来预测在给定新实例的元特征时,哪个算法或配置会表现最好。
- **成功案例:Meta-QSAR:**在药物发现领域,Meta-QSAR项目是一个极具说服力的案例。定量构效关系(QSAR)预测是新药研发的关键步骤,但不存在任何一种机器学习算法能在所有QSAR问题上都表现最佳。研究人员首先在一个包含超过2700个QSAR问题的数据集上,系统性地评估了18种回归方法和3种分子表示方法的性能。然后,他们训练了一个元学习模型,该模型能够根据新QSAR问题的特征,自动选择最合适的学习算法。结果显示,这种元学习方法比表现最好的单个算法(使用分子指纹的随机森林)的平均性能高出13%⁴⁷。

AutoML:元优化的统一工业框架

HPO、NAS和ASC并非孤立的领域,它们在AutoML的旗帜下实现了融合。AutoML代表了将元优化原则应用于整个机器学习流程的工业级实践。一个成熟的AutoML平台能够自动化从数据预处理、特征工程、模型选择(ASC)、神经架构搜索(NAS)到超参数优化(HPO)的全过程⁵¹。

从本质上看,AutoML将构建一个高性能机器学习模型的完整过程,视为一个单一的、巨大的元优化问题。在这个问题中,“对象级”操作是一次完整的模型训练和评估流程,而“元级”优化器就是AutoML系统本身,它在由算法、架构、特征工程流水线和超参数构成的庞大组合空间中进行搜索。大量的工业案例研究雄辩地证明了这种统一方法的巨大商业价值:它不仅能将模型部署时间从数周缩短至数小时,还能显著提升模型的准确率,同时降低了对顶尖数据科学家的依赖,实现了AI技术的普惠化⁵¹。

第四部分:关键挑战与前沿课题

尽管元级优化系统展现出巨大的潜力，但其发展和应用仍面临着一系列严峻的挑战。其中，模型的泛化能力和高昂的计算成本是两个最核心的难题。本部分将深入剖析这些挑战，并介绍领域内为应对这些挑战而发展出的前沿技术。

第7节 元学习中的泛化问题

泛化是所有机器学习模型追求的终极目标，但在元学习中，这个问题变得更加复杂和微妙。确保元学习模型能够真正地泛化，是其能否在实际应用中取得成功的关键。

7.1 泛化的双重性

元学习的泛化挑战具有双重性，必须对其进行明确区分，因为针对两者的解决方案可能不同，甚至相互冲突⁵⁷。

- **元泛化 (Meta-Generalization)** : 指元模型 (meta-model) 泛化到全新、未曾见过的任务上的能力。这些新任务虽然未在元训练阶段出现，但被假设与元训练任务来自同一个任务分布。这是元学习最首要的目标，即学会一种能够普适于未来任务的学习策略。
- **适应泛化 (Adaptation-Generalization)** : 指经过内循环适应 (adaptation) 后的基础学习器，在单个特定任务的领域内泛化的能力。也就是说，一个模型可能很好地适应了某个任务的少量支持集样本，但在该任务的测试集 (来自同一数据分布的未见样本) 上表现不佳。这本质上是标准机器学习中的过拟合问题，但在小样本场景下尤为突出。

7.2 过拟合的根源

元学习引入了一个新的过拟合层面。元模型可能会过度拟合元训练阶段所使用的特定任务集，从而学到一种“伪”学习策略，这种策略只对这些见过的任务有效，而无法泛化到新任务上⁵⁷。此外，元训练任务分布与元测试任务分布之间的领域漂移 (domain shift) 是导致元泛化失败的主要原因之一。如果测试任务的性质与训练任务截然不同，元学习模型学到的“先验知识”可能不仅无用，甚至会有害¹⁵。

7.3 提升泛化能力的技术

为了应对泛化挑战，研究人员提出了一系列旨在增强模型鲁棒性的技术。

- **元正则化 (Meta-Regularization)**: 在元学习的外循环中引入正则化项，以提升元泛化能力。常见的方法包括：通过任务插值 (MLTI) 来数据增强，即在两个现有任务的特征和标签之间进行线性插值，从而生成新的虚拟任务来扩充元训练集⁵⁹；以及对元模型的参数空间施加约束，防止其过拟合到训练任务上⁵⁷。
- **极小化极大方法 (Minimax Approaches)**: 这是一种更深刻的正则化思想。例如，“极小化极大元正则化” (Minimax-Meta Regularization) 提出了一种反直觉的策略：在内循环中故意使用“反向正则化” (inverted regularization)，例如，使用一个负的L2正则化系数。这会使得基础学习器在适应支持集时更难泛化，即故意削弱其适应泛化能力。其背后的逻辑是，通过增加内循环学习的难度，迫使外循环中的元学习器去寻找一个更加鲁棒、不依赖于任务数据中偶然噪声的元初始化参数。这种“先抑后扬”的策略最终能同时提升适应泛化（在任务测试集上）和元泛化能力⁵⁷。
- **模拟领域漂移**: 在元训练阶段，通过在每个小批量 (mini-batch) 中人工合成虚拟的训练域和测试域，来显式地模拟领域漂移。元优化目标被设定为：在训练域上提升性能的梯度更新，也必须同时能在虚拟测试域上提升性能。这种方法迫使模型学习到领域不变的 (domain-invariant) 特征表示，从而对真实的领域漂移具有更强的鲁棒性⁵⁸。

元泛化与适应泛化之间存在一种内在的张力。一个能够极快地拟合支持集数据的元初始化参数，可能也更容易在该任务上过拟合。最优的元模型必须在快速适应能力和强大的泛化鲁棒性之间找到精妙的平衡。上述“反向正则化”的例子深刻地揭示了这一点：有时，通过让学习变得更“困难”，反而能学到更本质、更通用的知识。

第8节 应对计算复杂性与可扩展性

元级优化系统，特别是早期的NAS方法，因其惊人的计算需求而闻名，完成一次搜索可能需要数千个GPU小时，这极大地限制了其研究和应用³³。因此，提升计算效率和可扩展性成为该领域的一个核心研究方向。

8.1 元优化的 prohibitive 成本

元优化的计算成本主要来源于其双层结构或庞大的搜索空间。在MAML中，计算元梯度需要通过内循环的整个优化路径进行反向传播，成本高昂¹⁸。在NAS中，对搜索空间中的每一个候选架构进行完整的训练和评估，其计算开销是巨大的，这构成了研究和应用的主要障碍³⁷。

8.2 提升NAS计算效率的策略

为了解决NAS的计算瓶颈，研究界开发了多种创新的效率提升策略。

- **参数共享 / 一次性模型 (One-Shot Models)**: 这是降低NAS成本的最重要突破之一。其核心思想是，不再独立地从零开始训练成千上万个模型，而是构建并训练一个包含搜索空间中所有可能架构作为其子图的“超网”(supernet)。评估一个特定架构时，只需从已训练好的超网中继承相应的权重，而无需重新训练，从而将评估成本降低了几个数量级³⁷。
- **可微NAS (Differentiable NAS)**: 以DARTS为代表，这类方法通过连续松弛(continuous relaxation)将离散的架构选择(例如，选择卷积还是池化)转化为对不同操作进行加权的连续优化问题。这使得架构本身可以通过梯度下降进行优化，其效率远高于基于强化学习或演化算法的黑盒搜索方法³⁹。
- **性能预测器与代理基准 (Surrogate Benchmarks)**: 训练一个代理模型来预测一个架构的最终性能。这个代理模型的输入可以是一些计算成本较低的指标，输出则是对该架构完整训练后性能的预测。这避免了对每个架构都进行昂贵的完整训练⁶¹。
- **零成本代理 (Zero-Cost Proxies)**: 这是近年来一个令人兴奋的研究方向。这类方法试图在完全不进行任何训练的情况下，仅通过分析一个神经网络在初始化时的某些属性，来预测其训练完成后的性能。这些属性可以基于梯度流、激活值的重叠度等指标计算得出，能够在数秒内完成对一个架构的评估，极大地加速了搜索过程⁶²。

8.3 元学习的可扩展性挑战

除了计算成本，元学习自身也面临着可扩展性方面的挑战。

- **双层优化的扩展性**: 如前所述，MAML等需要计算高阶导数的方法在应用于大型模型和长内循环步骤时，会面临内存和计算时间的巨大压力。虽然Reptile等一阶近似方法缓解了这个问题，但可能会牺牲一定的性能。
- **对任务数量和多样性的依赖**: 元学习的成功在很大程度上依赖于元训练阶段有大量且多样化的任务。在许多现实世界的应用场景中，获取成百上千个相关的、带有标注数据的任务本身就是一个巨大的瓶颈⁵⁹。任务插值等技术虽然能部分缓解这个问题，但生成高质量的虚拟任务仍然是一个挑战。

第五部分：应用与未来展望

在深入探讨了元级优化系统的理论、方法和挑战之后，本部分将通过具体的案例研究，展示其在科学研究和工业界产生的实际影响，并展望该领域最前沿的发展方向。

第9节 现实世界的影响：科学与工业案例研究

元级优化系统已经从理论研究走向实际应用，在多个领域创造了可量化的价值。

9.1 科学发现

- **药物发现 (Meta-QSAR)**: 在新药研发中，预测化合物对特定靶点的活性(即定量构效关系，QSAR)至关重要。Meta-QSAR项目是元学习在科学发现中应用的典范。研究人员面对超过2700个不同的QSAR预测任务，利用元学习来自动为每个任务选择最佳的机器学习模型(从18种回归方法和3种分子表示中选择)。结果表明，这种元学习驱动的选择策略，其平均性能比全局最优的单一方法(使用分子指纹的随机森林)高出13%。这证明了元学习在处理大量异构科学问题上的巨大优势⁴⁷。
- **个性化医疗**: 元学习特别适用于数据稀疏的场景。在罕见病诊断或治疗方案制定中，每个病人的数据都极为有限。元学习模型能够利用从其他常见疾病中学到的“学习经验”，快速适应新病人的个体数据，从而做出更快速、更准确的预测和诊断，为实现真正的个性化医疗提供了可能¹¹。

9.2 工业自动化 (AutoML)

AutoML作为元优化的工业级应用，其商业成功案例层出不穷。

- **金融服务**: 一家全球性银行机构采用AutoML解决方案对其信贷风险评估流程进行革新。该系统能够自动分析复杂的金融数据，识别细微的风险模式，并实时生成高精度的预测模型。最终，该行在贷款违约风险的预测准确率上提升了40%，人工处理时间减少了25%⁵¹。另一家科技公司Consensus Corporation使用DataRobot平台进行欺诈检测，不仅将欺诈检测率提升了24%，将误报率降低了55%，更将模型部署时间从原先的3-4周戏剧性地缩短到了8小时⁵³。
- **零售与电子商务**: 通过部署一个由AutoML驱动的动态推荐引擎，一家零售公司能够实时分析海量的用户交互数据，并持续优化推荐模型。这带来了显著的业务增长：产品推荐的点击率增加了35%，平均订单价值提升了22%⁵¹。
- **物流与供应链**: 在复杂的多级库存管理问题中，使用贝叶斯优化(一种核心的HPO技术)来调

整库存策略,其收敛到最优解的速度在某些情况下比传统的遗传算法快22倍,极大地提升了决策效率和供应链的稳健性²⁶。

9.3 工程与机器人学

- 机器人学:元学习赋予了机器人快速学习新技能的能力。一个在多种抓取任务上进行过元训练的机器人,在遇到一个从未见过的物体时,只需几次尝试就能学会如何成功抓取它。这种能力是通过将在先前任务中学到的通用抓取策略迁移到新任务上实现的,这对于需要在动态、非结构化环境中工作的机器人至关重要¹³。
- 纳米光子逆向设计:设计高性能的纳米光子器件是一个高度非凸的复杂优化问题。研究人员利用HPO和元学习技术来自动优化器件的几何结构和材料参数。这些元优化方法能够高效地探索高维设计空间,找到比人类专家手动设计性能更优的解决方案⁶⁵。

第10节 元优化的前沿

元优化的边界正在不断被拓展。当前的研究正致力于将元优化的原则推向新的高度,以创造出更强大、更鲁棒、更自主的智能系统。

10.1 因果元学习:学习如何干预

标准元学习在模式识别任务上取得了巨大成功,但它主要学习的是变量之间的相关性。当数据分布发生改变,或者需要进行因果推理时(即回答“如果我做了X,会发生什么?”),仅靠相关性模型往往会失效。

- 零样本因果学习(**Zero-Shot Causal Learning, CaML**):这是元学习领域一个革命性的新方向。它将“预测一个干预措施的个性化效果”这一因果推断问题,构建为一个元学习任务。例如,通过在大量历史医疗数据上进行元训练,其中每个“任务”对应于一种特定的药物(干预措施)及其对不同患者(个体)产生的影响,CaML框架能够学习到一个通用的因果预测模型。这个模型的核心能力是,能够预测一种在训练时从未出现过的全新药物,对一个新病人可能产生的个性化治疗效果(Conditional Average Treatment Effect, CATE)⁶⁶。这标志着元学习从“学习如何更快地学习相关性”到“学习如何更快地推理干预效果”的重大飞跃,对于个性化医疗、政策制定和科学实验等领域具有深远意义。这一转变,如同从对象级到元级的跃升一样,代表了抽象层次的又一次提升,从关联阶梯迈向了干预阶梯。

10.2 自指系统:通往递归式自我改进之路

这是元优化的终极目标:创造能够动态修改自身学习算法,而无需一个固定的、外部的元优化器(外循环)的系统⁶⁹。

- 这种思想可以追溯到早期关于“自指”(self-referential)循环神经网络的研究,即RNN通过反向传播学习如何执行自己的权重更新算法,而这个算法本身可能与反向传播完全不同⁹。
- 现代研究则致力于在更复杂的系统中消除显式的元优化步骤,让系统实现真正的递归式自我改进⁶⁹。这正是第一部分所讨论的“打破受保护的元级”理念在工程上的具体实现,是通往更高级通用人工智能的潜在路径。

10.3 大型语言模型(LLM)的角色

大型语言模型(LLM)正在迅速成为元优化系统中的一个强大且灵活的核心组件,其作用远不止于一个简单的工具。

- **LLM用于HPO**:LLM不仅可以基于对算法和问题的理解,智能地提出超参数配置建议,甚至可以超越传统的、预先定义的搜索空间。例如,LLM可以直接输出修改模型训练逻辑的源代码,从而探索传统HPO方法无法触及的优化维度⁷¹。
- **LLM用于NAS**:在代码上进行微调的LLM,可以像人类工程师一样“编写”出新颖的神经网络架构。RZ-NAS等框架利用LLM生成架构的变体,并通过一个“类人反思”(humanoid reflections)模块,结合零成本代理的快速评估结果,来指导后续的搜索方向。这结合了LLM的生成创造力与传统NAS的高效评估策略⁶³。
- **LLM作为元学习器**:基于LLM的智能体(Agent)本身就可以通过强化学习等方法进行优化,以提升其在决策、规划和推理任务上的性能。这本质上是在对其自身的“思考过程”进行元优化,使其成为一个更高效的学习者和问题解决者⁷³。

LLM的出现正在重塑元优化的范式。传统的HPO和NAS方法在高度结构化的搜索空间中运行(例如,浮点数范围、离散操作列表),而LLM将这个搜索空间扩展到了自然语言描述、Python代码片段乃至逻辑推理链的广阔领域。LLM充当了一个通用接口,将人类可理解的高层概念(例如,“设计一个对旋转不变的卷积网络”)与机器可执行的底层配置连接起来。这极大地扩展了自动化的范围和能力,预示着一个未来:用户或许只需用自然语言描述一个问题,一个由LLM驱动的元优化系统就能自动完成方案设计、代码编写、模型调优和部署的全过程。

结论

元级优化系统代表了人工智能领域从关注“结果”到关注“过程”的深刻范式转移。本报告的分析始于对对象级与元级这一基本哲学区分的探讨，揭示了这一抽象层级的跃升如何从一个思维模型演变为一个可计算、可优化的工程框架。我们看到，无论是生物演化还是人类科学进步，历史上都存在一个相对稳定的“受保护的元级”，而现代AI研究的核心突破之一，正是要打破这种保护，实现对优化过程本身的递归式、加速改进。

元学习作为这一范式在机器学习中的核心体现，通过其标志性的双层优化结构，系统性地实现了“学习如何学习”的目标。无论是通过学习一个更优的“起点”（基于优化的方法），一个更通用的“尺子”（基于度量的方法），还是一个更高效的“学习机器”（基于模型的方法），元学习都在致力于提升AI系统的数据效率和适应性。

在实践层面，元优化的原则已经渗透到自动化机器学习（AutoML）的各个角落。超参数优化（HPO）、神经架构搜索（NAS）和算法选择与配置（ASC）共同构成了AutoML生态系统的三大支柱，它们将构建高性能模型的整个流程——从算法选择到架构设计再到参数调优——统一在一个宏大的元优化框架之下。来自金融、医疗、零售和科研等领域的众多成功案例，雄辩地证明了这一自动化范式的巨大价值。

然而，前路依然充满挑战。模型的泛化能力，特别是在元泛化与适应泛化之间的微妙平衡，以及巨大的计算成本，仍然是限制元优化系统广泛应用的主要障碍。但同时，前沿研究也为我们指明了未来的方向。因果元学习的兴起，预示着AI正从学习“相关性”迈向推理“因果性”的更高认知层次；自指系统的探索，则直接指向了递归式自我改进这一人工智能的终极理想；而大型语言模型的融入，正以前所未有的灵活性和通用性，重塑着我们与复杂优化空间交互的方式。

综上所述，元级优化系统不仅是一系列先进技术的集合，更是一种构建未来智能系统的核心思想。它驱动我们从设计单一的解决方案，转向设计能够自我完善的、不断进化的“问题解决引擎”。随着理论的不断深化和技术的持续突破，这一领域必将在推动通用人工智能发展的道路上扮演愈发关键的角色。

引用的著作

1. Object level and Meta level — LessWrong, 访问时间为 九月 24, 2025, <https://www.lesswrong.com/w/object-level-and-meta-level>
2. Object level vs Meta level : r/slatestarcodex - Reddit, 访问时间为 九月 24, 2025, https://www.reddit.com/r/slatestarcodex/comments/48d1vm/object_level_vs_meta_level/
3. What are meta-level and object-level proofs? - Mathematics Stack Exchange, 访问时间为 九月 24, 2025, <https://math.stackexchange.com/questions/3273812/what-are-meta-level-and-object-level-proofs>
4. Learning to select computations, 访问时间为 九月 24, 2025, <https://cocosci.princeton.edu/papers/callawayLearningToSelect.pdf>
5. Optimization and the Intelligence Explosion - Rationality: From AI to Zombies, 访问

时间为 九月 24, 2025,

<https://www.readthesequences.com/Optimization-And-The-Intelligence-Explosion>

6. Optimization and the Singularity - LessWrong, 访问时间为 九月 24, 2025,
<https://www.lesswrong.com/posts/HFTn3bAT6uXSNwv4m/optimization-and-the-singularity>
7. Meta-optimization - Wikipedia, 访问时间为 九月 24, 2025,
<https://en.wikipedia.org/wiki/Meta-optimization>
8. www.ibm.com, 访问时间为 九月 24, 2025,
<https://www.ibm.com/think/topics/meta-learning#:~:text=In%20Optimization%2Dbased%20meta%20learning,optimizing%20the%20optimization%20algorithm%20itself.>
9. Meta-learning (computer science) - Wikipedia, 访问时间为 九月 24, 2025,
[https://en.wikipedia.org/wiki/Meta-learning_\(computer_science\)](https://en.wikipedia.org/wiki/Meta-learning_(computer_science))
10. Meta Learning - Ultralytics, 访问时间为 九月 24, 2025,
<https://www.ultralytics.com/glossary/meta-learning>
11. Meta Learning: 7 Techniques & Use Cases - Research AIMultiple, 访问时间为 九月 24, 2025, <https://research.aimultiple.com/meta-learning/>
12. Meta-Learning - ALIN Lab, 访问时间为 九月 24, 2025,
https://alinlab.kaist.ac.kr/resource/Lec17_Meta_learning.pdf
13. Meta-Learning Optimization Explained, AI Consultants UK, 访问时间为 九月 24, 2025,
https://www.efficiencyai.co.uk/knowledge_card/meta-learning-optimization/
14. What Is Meta Learning? - IBM, 访问时间为 九月 24, 2025,
<https://www.ibm.com/think/topics/meta-learning>
15. Meta Learning: How Machines Learn to Learn | DataCamp, 访问时间为 九月 24, 2025, <https://www.datacamp.com/blog/meta-learning>
16. Exploring Meta-learning in Machine Learning | by Subash Palvel - Medium, 访问时间为 九月 24, 2025,
<https://subashpalvel.medium.com/exploring-meta-learning-in-machine-learning-10c889acef47>
17. Advances and Challenges in Meta-Learning: A Technical Review - arXiv, 访问时间为 九月 24, 2025, <https://arxiv.org/html/2307.04722>
18. Meta-Learning in Neural Networks: A Survey - arXiv, 访问时间为 九月 24, 2025,
<https://arxiv.org/abs/2004.05439>
19. Meta-Optimization Beyond MAML: Exploring NeuralGrok and Learnable Gradient Accumulation (LGA) | by James @ ForGen AI, 访问时间为 九月 24, 2025,
<https://blog.forgen.ai/meta-optimization-beyond-maml-exploring-neuralgrok-and-learnable-gradient-accumulation-lga-939e810889f2>
20. (PDF) Artificial Neural Network Hyperparameters Optimization: A Survey - ResearchGate, 访问时间为 九月 24, 2025,
https://www.researchgate.net/publication/366080627_Artificial_Neural_Network_Hyperparameters_Optimization_A_Survey
21. Hyperparameter optimization: Classics, acceleration, online, multi-objective, and tools, 访问时间为 九月 24, 2025,

- <https://www.aimspress.com/article/doi/10.3934/mbe.2024275?viewType=HTML>
22. Optimization Problem Solving Can Transition to Evolutionary Agentic Workflows - arXiv, 访问时间为 九月 24, 2025, <https://arxiv.org/html/2505.04354v1>
 23. Algorithms for Hyper-Parameter Optimization - NIPS, 访问时间为 九月 24, 2025, <http://papers.neurips.cc/paper/4443-algorithms-for-hyper-parameter-optimization.pdf>
 24. Algorithms for Hyper-Parameter Optimization - NIPS, 访问时间为 九月 24, 2025, <https://papers.nips.cc/paper/4443-algorithms-for-hyper-parameter-optimization>
 25. Hyperparameter Optimization For LLMs: Advanced Strategies - neptune.ai, 访问时间为 九月 24, 2025, <https://neptune.ai/blog/hyperparameter-optimization-for-llms>
 26. Bayesian Optimization Methods for Inventory Control with Agent-Based Supply-Chain Simulator - ResearchGate, 访问时间为 九月 24, 2025, https://www.researchgate.net/publication/358842409_Bayesian_Optimization_Methods_for_Inventory_Control_with_Agent-based_Supply-chain_Simulator
 27. Adaptive Bayesian Optimization Algorithm for Unpredictable Business Environments. - arXiv, 访问时间为 九月 24, 2025, <https://arxiv.org/pdf/2401.11264>
 28. Impact of Bayesian Approach to Demand Management in Supply Chains for the Consumption of Dynamic Products - SciELO México, 访问时间为 九月 24, 2025, https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1405-55462023000200545
 29. Meta-Learning - AutoML.org, 访问时间为 九月 24, 2025, <https://www.automl.org/meta-learning/>
 30. Advances in neural architecture search | National Science Review - Oxford Academic, 访问时间为 九月 24, 2025, <https://academic.oup.com/nsr/article/11/8/nwae282/7740455>
 31. Neural Architecture Search: A Survey - Journal of Machine Learning ..., 访问时间为 九月 24, 2025, <https://www.jmlr.org/papers/volume20/18-598/18-598.pdf>
 32. Neural Architecture Search Survey: A Computer Vision Perspective - PubMed, 访问时间为 九月 24, 2025, <https://pubmed.ncbi.nlm.nih.gov/36772749/>
 33. Neural Architecture Search - AutoML.org, 访问时间为 九月 24, 2025, <https://www.automl.org/wp-content/uploads/2018/12/nas-1.pdf>
 34. Neural Architecture Search (NAS) for Computer Vision Models - XenonStack, 访问时间为 九月 24, 2025, <https://www.xenonstack.com/blog/neural-architecture-search>
 35. Neural architecture search - Wikipedia, 访问时间为 九月 24, 2025, https://en.wikipedia.org/wiki/Neural_architecture_search
 36. Neural Architecture Search Survey: A Computer Vision Perspective - MDPI, 访问时间为 九月 24, 2025, <https://www.mdpi.com/1424-8220/23/3/1713>
 37. Advances in neural architecture search - PMC, 访问时间为 九月 24, 2025, <https://pmc.ncbi.nlm.nih.gov/articles/PMC11389615/>
 38. LLMatic: Neural Architecture Search via Large Language Models and Quality Diversity Optimization - arXiv, 访问时间为 九月 24, 2025, <https://arxiv.org/html/2306.01102v8>
 39. (PDF) Neural Architecture Search Benchmarks: Insights and Survey -

- ResearchGate, 访问时间为 九月 24, 2025,
https://www.researchgate.net/publication/369105356_Neural_Architecture_Search_Benchmarks_Insights_and_Survey
40. Neural Architecture Search: Optimal Performance Automation - KnowledgeNile, 访问时间为 九月 24, 2025,
<https://www.knowledgenile.com/blogs/neural-architecture-search-automatically-discovering-the-optimal-model-design>
 41. ICML Poster Few-Shot Neural Architecture Search, 访问时间为 九月 24, 2025,
<https://icml.cc/virtual/2021/poster/8891>
 42. Algorithm Selection for Combinatorial Search Problems: A Survey, 访问时间为 九月 24, 2025,
https://www.cs.uwyo.edu/~larsko/papers/kotthoff_algorithm_2014.pdf
 43. A Survey of Methods for Automated Algorithm Configuration - ResearchGate, 访问时间为 九月 24, 2025,
https://www.researchgate.net/publication/358344820_A_Survey_of_Methods_for_Automated_Algorithm_Configuration
 44. A Survey of Methods for Automated Algorithm Configuration (Extended Abstract) - IJCAI, 访问时间为 九月 24, 2025,
<https://www.ijcai.org/proceedings/2023/0791.pdf>
 45. Dagstuhl Seminar 16412: Automated Algorithm Selection and Configuration, 访问时间为 九月 24, 2025, <https://www.dagstuhl.de/16412>
 46. A Literature Survey on Offline Automatic Algorithm Configuration - MDPI, 访问时间为 九月 24, 2025, <https://www.mdpi.com/2076-3417/12/13/6316>
 47. Meta-QSAR: a large-scale application of meta-learning to drug design and discovery - PMC, 访问时间为 九月 24, 2025,
<https://pmc.ncbi.nlm.nih.gov/articles/PMC6956898/>
 48. Meta-QSAR: a large-scale application of meta-learning to drug design and discovery, 访问时间为 九月 24, 2025,
https://www.researchgate.net/publication/319662485_Meta-QSAR_a_large-scale_application_of_meta-learning_to_drug_design_and_discovery
 49. Meta-QSAR: a large-scale application of meta-learning to drug design and discovery - PubMed, 访问时间为 九月 24, 2025,
<https://pubmed.ncbi.nlm.nih.gov/31997851/>
 50. Meta-QSAR: a large-scale application of meta-learning to drug design and discovery, 访问时间为 九月 24, 2025,
<https://kclpure.kcl.ac.uk/portal/en/publications/meta-qsar-a-large-scale-application-of-meta-learning-to-drug-design>
 51. Automl case studies: Real-world applications of automated machine learning - BytePlus, 访问时间为 九月 24, 2025, <https://www.byteplus.com/en/topic/400275>
 52. What is automated ML? AutoML - Azure Machine Learning | Microsoft Learn, 访问时间为 九月 24, 2025,
<https://learn.microsoft.com/en-us/azure/machine-learning/concept-automated-ml?view=azureml-api-2>
 53. 22 AutoML Case Studies: Applications and Results, 访问时间为 九月 24, 2025,
<https://research.aimultiple.com/automl-case-studies/>

54. 22 AutoML Case Studies: Applications and Results in 2025 | ELECTRIX, 访问时间为九月 24, 2025, <https://www.electrixdata.com/automl-case-studies-2025.html>
55. AutoML Solutions - Train models without ML expertise | Google Cloud, 访问时间为九月 24, 2025, <https://cloud.google.com/automl>
56. Case Studies | H2O.ai, 访问时间为九月 24, 2025, <https://h2o.ai/case-studies/>
57. Improving Generalization of Meta-Learning With ... - CVF Open Access, 访问时间为九月 24, 2025, https://openaccess.thecvf.com/content/CVPR2023/papers/Wang_Improving_Generalization_of_Meta-Learning_With_Inverted_Regularization_at_Inner-Level_CVPR_2023_paper.pdf
58. Learning to Generalize: Meta-Learning for Domain Generalization - AAAI, 访问时间为九月 24, 2025, <https://aaai.org/papers/11596-learning-to-generalize-meta-learning-for-domain-generalization/>
59. Oral 3: Meta-learning and adaptation - ICLR 2026, 访问时间为九月 24, 2025, <https://iclr.cc/virtual/2022/session/8346>
60. Daily Papers - Hugging Face, 访问时间为九月 24, 2025, <https://huggingface.co/papers?q=meta%20relational%20learning>
61. Meta-Surrogate Benchmarking for Hyperparameter Optimization - NIPS, 访问时间为九月 24, 2025, <http://papers.neurips.cc/paper/8857-meta-surrogate-benchmarking-for-hyperparameter-optimization.pdf>
62. ICML Poster Neural Architecture Search without Training, 访问时间为九月 24, 2025, <https://icml.cc/virtual/2021/poster/9263>
63. RZ-NAS: Enhancing LLM-guided Neural Architecture Search via Reflective Zero-Cost Strategy | OpenReview, 访问时间为九月 24, 2025, <https://openreview.net/forum?id=9UEXqPH078>
64. A Comprehensive Survey on Meta-Learning: Applications, Advances, and Challenges, 访问时间为九月 24, 2025, https://www.researchgate.net/publication/384746068_A_Comprehensive_Survey_on_Meta-Learning_Applications_Advances_and_Challenges
65. Efficiency of machine learning optimizers and meta-optimization for nanophotonic inverse design tasks - AIP Publishing, 访问时间为九月 24, 2025, <https://pubs.aip.org/aip/aml/article/3/1/016101/3329131/Efficiency-of-machine-learning-optimizers-and-meta>
66. (PDF) Zero-shot causal learning - ResearchGate, 访问时间为九月 24, 2025, https://www.researchgate.net/publication/367557592_Zero-shot_causal_learning
67. Zero-shot causal learning, 访问时间为九月 24, 2025, https://proceedings.neurips.cc/paper_files/paper/2023/file/15ddb1773510075ef44981cdb204330b-Paper-Conference.pdf
68. Zero-shot causal learning - Stanford Computer Science, 访问时间为九月 24, 2025, <https://cs.stanford.edu/people/jure/pubs/zero-neurips23.pdf>
69. [PDF] Eliminating Meta Optimization Through Self-Referential Meta Learning | Semantic Scholar, 访问时间为九月 24, 2025, <https://www.semanticscholar.org/paper/84fbccdc59bf656478db9386918bd47429>

[d5614c](#)

70. Self-Referential Self-Improvement - Emergent Mind, 访问时间为 九月 24, 2025, <https://www.emergentmind.com/topics/self-referential-self-improvement>
71. Using Large Language Models for Hyperparameter Optimization - NeurIPS 2025, 访问时间为 九月 24, 2025, <https://neurips.cc/virtual/2023/82902>
72. RZ-NAS: Enhancing LLM-guided Neural Architecture Search via Reflective Zero-Cost Strategy - ICML 2025, 访问时间为 九月 24, 2025, <https://icml.cc/virtual/2025/poster/46224>
73. A Survey on the Optimization of Large Language Model-based Agents - arXiv, 访问时间为 九月 24, 2025, <https://arxiv.org/pdf/2503.12434>