CCF 人工智能与模式识别专业委员会
刘勇[1] 胡啸林[1] 唐鹏威[1] 龚子瑄[1]
1 中国人民大学，北京

摘 要

以 ChatGPT 为代表的大语言模型（Large Language Models， LLMs）在人机交互和多任务处理上带来了全新的突破，展现出了与传统操作系统融合的趋势，正通过与手机、个人笔记本等硬件的结合成为更懂用户的个人助理，同时在教育、法律、医疗和军事领域也有着巨大的应用前景。自从 2022 年 11 月 30 日 OpenAI 公司推出聊天机器人 ChatGPT 之后，相关的前沿技术突破层出不穷，逐渐深入到了千行百业，大算力和大数据驱动的大语言模型在助力人们工作生活的同时也成了国家间科技竞争的一个重要方面。然而当前大语言模型技术的发展还是以工程实践推动为主导，理论层面的指导和讨论比较有限，考虑到大语言模型的开发对算力等资源的消耗极大，如何在实践中融入相关理论指导原则以提高模型开发和部署的效率成为亟待解决的问题。基于以上分析，本文以助力大模型的开发、部署和使用为指引，对大语言模型相关理论工作进行综述。具体来讲，本文将从统计学习视角下的大模型理论概述入手，首先分析大模型理论与传统深度学习理论的异同，然后以大模型的工程实践为依托，总结大模型相关理论对实践的指导原则与潜在启发，最后给出对大语言模型涌现能力的机理分析。

关键词 大语言模型；统计学习理论；涌现机理；泛化分析；优化算法

Abstract

Large language models (LLMs) have transformed how we interact with computers, helping us accomplish a variety of tasks efficiently. As they integrate more with traditional operating systems and work on devices like smartphones and laptops, LLMs are becoming personal assistants. These models show huge promise in fields like education, law, healthcare, and defense. Since OpenAI

launched ChatGPT, progress in this area has been fast. Driven by powerful computing and large datasets, LLMs are enhancing both work and everyday life, while also playing a role in global technology competition. However, LLM development has largely been guided by engineering practice rather than theory, with few theoretical insights driving progress. Given the resources needed to create LLMs, especially in terms of computational cost, it is increasingly important to integrate theoretical principles to make model development and deployment more efficient. This paper reviews the theoretical work related to LLMs to support their development, use, and application. We begin with an overview of LLM theory from a statistical learning perspective, examining both the similarities and differences between theories for LLMs and traditional deep learning. Then, we summarize how theory can guide practical applications and suggest possible ways to improve LLMs. Finally, we analyze the mechanisms behind the emergent abilities of LLMs, identifying key factors that contribute to their effectiveness.

Keywords: large language models; statistic learning theory; emergence ability; generalization analysis; optimization theory

1 大语言模型理论概述：统计学习视角

大语言模型以端到端的方式实现了对自然语言处理任务的变革，同时带来了新的人机交互可能。从工程上讲，大语言模型的成功依赖于大模型、大算力与大数据。相应的理论分析也从这三个方面展开，模型结构的表达能力限制着大语言模型性能的上限，算力的充分利用依赖算法与硬件系统的紧密结合，而训练数据的分布则影响着大语言模型在不同任务上的性能差异。虽然大模型场景下模型结构、优化算法与训练数据之间的耦合关系更为复杂，相关理论分析仍旧可以在统计学习框架下展开。统计学习理论涉及表达能力、优化误差和泛化误差三个方面，表达能力限定了模型所能拟合函数的范围，优化误差刻画了训练数据上算法的收敛性，而泛化误差表征了训练得到的模型在测试数据上的最终性能[1-2]。机器学习的根本目标是得到泛化能力强的模型，而最终得到的模型性能同时受到优化算法、训练数据和模型结构的影响。本节从统计学习视角入手，首先概述经典的统计学习框架（图 1），进而综述大语言模型对应的统计学习理论结果，并结合实践展开讨论。

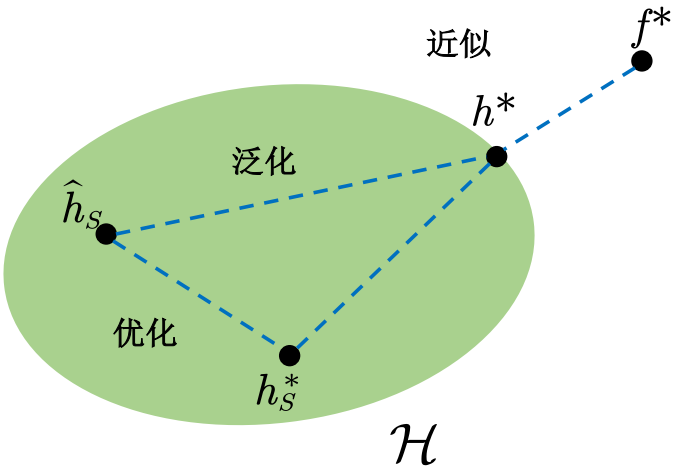


图 1 统计学习理论框架

1.1 大语言模型理论框架

与传统机器学习相比，大语言模型在优化、泛化与模型架构上都体现了新的特点，同时给实践和理论带来了新的挑战[1]。传统的统计学习框架是针对单一任务而言的[2]，其假设数据 x 和标签 y 采样自联合分布$(X,Y) \sim \mu$, 该联合分布是未知的。学习的目标可以描述为寻找函数 $h \in \mathcal{H}$ 使得对于任意采样自未知分布的 x，h 都能够输出 y 的预测值，预测性能用损失函数 ℓ 进行衡量：

$$L(w,\mu) = \int \ell(h_w(X),Y) d\,\mu(X,Y),$$

这里 $w \in \mathcal{W}$ 表示函数 h 的参数，ℓ表示损失函数。由于联合分布未知，即真实场景中只能够观测到有限的数据规模, 由此优化目标可以写为：

$$L(w,S) = \frac{1}{|S|} \sum_{(x,y)\in S} \ell(h_w(x),y),$$

这里S表示有限训练数据的集合，$|S|$表示训练样本数目。常用的损失函数包括平方损失、交叉摘损失和自回归损失等，其中交叉摘损失主要被用于分类任务，自回归损失被广泛用于大语言模型预训练对应的生成任务, 具体形式为：

$$L_{\mathrm{PT}}(w,x) = -\sum_{t=1}^{T} log\ P(x_t \mid x_{t-1},\cdots,x_0;w),$$

这里$P(x_t \mid x_{t-1},\cdots,x_0;w)$表示模型 w 在给定输入序列前 $t-1$ 个词元时对第 t 个词元概率的预测。

记给定给 x 生成 y 的真实函数为 $f^*(\cdot):\mathcal{X}\to\mathcal{Y}$, 考虑到假设空间 $\mathcal{H}$ 未必包含真实的数据生成函数 $f^*(\cdot)$，由此带来了近似误差（Approximation error）：

$$L(h^*,\mu)-L(f^*,\mu),$$

这里$h^*=\arg\min_{h\in\mathcal{H}} L(h,\mu)$表示假设空间内使得期望风险$L(h,\mu)$最小的函数。由定义可知，近似误差本质上是由假设空间表达能力决定的, 其只与模型架构有关, 而与算法和数据无关。

记训练数据 S 和算法导出的函数及其参数分别为 $\hat{h}_S$ 和 $\hat{w}_S$, 其对应的期望风险为：

$$L(\hat{h}_S,\mu)=\int \ell\big(h_{\hat{w}_S}(X),Y\big)d\,\mu(X,Y),$$

这里 $\hat{h}_S$与真实的数据生成函数$f^*(\cdot)$之间的距离可以进行如下分解：

$$L(\hat{h}_S,\mu)-L(f^*,\mu)=L(\hat{h}_S,\mu)-L(h^*,\mu)+L(h^*,\mu)-L(f^*,\mu),$$

最后两项构成近似误差 $L(h^*,\mu)-L(f^*,\mu)$, 前面两项可以进一步作如下分解：

$$L(\hat{h}_S,\mu)-L(h^*,\mu)=L(\hat{h}_S,\mu)-L(h_S^*,S)+L(h_S^*,S)-L(h^*,\mu),$$

这里 $h_S^*=\arg\min_{h\in\mathcal{H}} L(h,S)$ 代表假设空间内使得训练误差 $L(h,S)$最小的函数。由于期望意义下 $L(h^*,S)$与$L(h^*,\mu)$相同且$L(h_S^*,S)\le L(h^*,S)$, 可以导出上式最后两项满足:

$$E[L(h_S^*,S)-L(h^*,\mu)]=E[L(h_S^*,S)-L(h^*,S)]\le 0.$$

由此可得$L(\hat{h}_S,\mu)-L(h^*,\mu)\le L(\hat{h}_S,\mu)-L(h_S^*,S)$. 进一步分解可得：

$$L(\hat{h}_S,\mu)-L(h^*,\mu)\le L(\hat{h}_S,\mu)-L(\hat{h}_S,S)+L(\hat{h}_S,S)-L(h_S^*,S),$$

上式右侧前两项表示优化算法基于训练数据导出的模型 $\hat{h}_S$在分布 μ 和训练数据 S 上的性能差距, 这一项通常被称为泛化误差, 后面两项表示 $\hat{h}_S$和 h_S^* 在训练数据 S 上的性能差异, 这一项取决于优化算法的优劣。

综上可得，$L(\hat{h}_S,\mu)-L(f^*,\mu)$ 表征了数据 S 上训练得到的模型 $\hat{h}_S$ 的性能，其上界可以被分解为近似误差、泛化误差和优化误差三项，即：

$$L(\hat{h}_S, \mu) - L(f^*, \mu) \leq L(\hat{h}_S, \mu) - L(\hat{h}_S, S) + L(\hat{h}_S, S) - L(h_S^*, S) + L(h^*, \mu) - L(f^*, \mu),$$

其中近似误差 $L(h^*, \mu) - L(f^*, \mu)$ 取决于模型的表达能力，泛化误差$L(\hat{h}_S, \mu) - L(\hat{h}_S, S)$刻画了测试与训练性能之间的差距，优化误差$L(\hat{h}_S, S) - L(h_S^*, S)$取决于优化算法能力。接下来按照以上框架总结大语言模型场景下的相应理论并进行讨论。

1.2 大语言模型表达能力

经典的神经网络表达能力分析可以追溯到 1989 年，相关结论表明无限宽的单隐藏层神经网络能够以任意精度逼近紧支撑集（compact support）上的任意函数[3-4]。近年来相关研究集中在有限宽的神经网络上，例如 Lu 等[5]和 Hanin 等[6]表明当输入数据维度为 d 时，宽度为 $d+1$ 的无限深全连接 ReLu (Rectified Linear Unit)网络是连续标量函数的通用逼近器。与连续标量函数的逼近不同，大语言模型场景下表达能力关注的是模型在序列到序列函数上的逼近能力。以下从大语言模型架构与序列到序列函数两个角度进行展开论述，Yun 等[7]首次证明了 Transformer 模型能够逼近任意的序列到序列函数，之后 Kratsios 等[8]证明了 Transformer 对于受限的输出空间也有通用逼近能力，这里受限的输出空间包括了分类问题中的输出为概率单纯形（probability simplex)的情况，即要求输出向量中的元素始终构成一概率分布。Kim 等[9]分析了 Transformer 架构的记忆容量，这是理解模型表达力和泛化能力的关键。作者通过理论证明当输入维度为d时，Transformer 仅需$O\left(d + n + \sqrt{nN}\right)$个参数便能够记忆$N$个长度为$n$的序列到序列映射。

大语言模型自回归的推理方式成本较高，对更高效模型的探索长期受到关注[10]，尽管目前已经提出了许多高效的 Transformer 模型，但没有理论保证它们是标准 Transformer 的合适替代品，这其中不同模型架构在表达能力上有无差距成为研究人员关心的问题[11-12]。 Yang 等[13]研究了稀疏 Transformer[11]和线性 Transformer [12]等高效模型在思维链（Chain-of-Thought, CoT）场景下的推理能力。通过将推理任务建模为动态规划问题，作者证明了这些高效模型在理论上能够解决一般的动态规划任务，但需要随着问题规模的增长而增加模型大小，从计算效率上无法直接体现其相对于原生 Transformer 模型的优越性。Wen 等[14]最近探索了 RNN 表达能力与 Transformer 的差距，特别关注了 RNNs 在处理长序列时的内存效率优势，以及 Transformers 通过自注意力机制实现的密集信息路由能力。相关理论分析揭示了 RNNs 在上下文检索（In-context Retrieval）方面的局限性，即使在采用 Chain-of-Thought (CoT) 提示的情况下，RNNs 仍然无法解决某些需要从上下文中检索信息的算法问题，对于工程实践的启

发是在 RNN 中适当加入 Transformer 模块对其表达能力能起到较大帮助。

综上可得，Transformer 模型通用逼近能力保证了其能够实现对序列到序列函数的拟合，此时近似误差可能被忽略，实践中应当重点关注优化误差和泛化误差。依靠当前理论框架对思维链等任务表达能力等分析则揭示了稀疏 Transformer[11]、线性 Transformer [12]和 RNNs 等模型的局限性，相关结果可以用于对模型结构改进的理论依据，但是随着理论框架的发展相关结论是否依旧成立尚无定论，有待进一步发展。

1.3 大语言模型泛化分析

传统机器学习泛化分析针对的是单一数据分布场景[15]，大语言模型泛化分析涉及到多任务对应的多个数据分布场景，其预训练阶段的核心指标是多任务上的平均性能[16]，微调阶段的核心指标是下游目标任务上的性能[17]。传统泛化分析的理论工具包括复杂度、稳定性和信息论等[18]，近年来涌现出了一些新的理论工具。相比于传统机器学习，大语言模型的泛化分析有其自身特点，本节主要讨论大模型场景下相关理论结果的指导意义。

大语言模型的预训练用到了互联网上的海量数据，这些数据涉及多个主题/任务，其不再满足数据独立同分布的假设[16, 19]，已有工作中与大模型预训练最相关的框架为多任务或者元学习设定[20]。除此之外，GPT-2 论文中提出了大语言模型是多任务学习器的观点[21]，基于以上分析，可以在多任务学习理论框架下对预训练阶段的优化误差和泛化误差进行建模。多任务设定假设参与训练的数据是独立非同分布的，对于 m 个任务，每个任务采样 n 个样本的场景，其期望误差定义为：

$$L_{\mathrm{PT}}(w)=\frac{1}{m}\sum_{i=1}^{m}\mathbb{E}\left[\ell\left(w,X_i^j\right)\right],$$

训练（优化）误差定义为：

$$\hat{L}_{\mathrm{PT}}(w)=\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\ell\left(w,X_i^j\right),$$

泛化误差定义为：

$$L_{\mathrm{PT}}(w)-\hat{L}_{\mathrm{PT}}(w).$$

以复杂度为代表的方法用假设空间大小对泛化误差进行界定，其理论结果是算法无关的。

以稳定性和信息论为基础的泛化分析在泛化界限中引入了优化算法的影响[22-23]。其中稳定性将泛化分析转化为数据扰动对优化轨迹影响的分析，其往往需要梯度的模或者海森矩阵的谱范数有界，这两个条件刻画了优化目标的光滑性，但是其对优化过程中的特征进行了粗糙化的统一处理，无法细粒度地体现数据与算法的影响，在大模型泛化分析中局限性较强[24]。

大语言模型第一个非空的泛化界限拓展了假设空间有限设定下的结论, 这里先给出经典结论[25]。假设损失函数非负且存在上界 b, 对于包含有限个函数的假设空间 $w \in \mathcal{W}$, 记 w对应的先验分布概率为 $\mathrm{P}(w)$, 则以下结论以 $1-\delta$ 的概率成立：

$$L_{\mathrm{PT}}(w) \leq \hat{L}_{\mathrm{PT}}(w) + b\sqrt{\frac{\log\frac{1}{P(w)} + \log\frac{1}{\delta}}{2mn}},$$

当学习到的 w 使得 $P(w)$增加时, 泛化误差会得到改善。由于先验分布 P 未知, 文献中通过 Solomonoff 先验 $P(w) \leq 2^{-K(w|A)}$给出[26], 其中 K 是 w的前缀 Kolmogorov 复杂度[27], A为模型架构。虽然 K 是不可计算的，但可以计算其上界：

$$\log\frac{1}{P(w)} \leq K(w \mid A)\log 2 \leq C(w)\log 2 + 2\log C(w),$$

其中$C(w)$是 w在给定任何特定压缩策略下的压缩大小。基于以上分析可得, 如果能够找到使得训练误差较小的w同时其压缩之后也较小, 则可以得到较紧的泛化误差界。文章中提出了 SubLoRA 方法[25], 可以在保持表达能力的同时找到足够压缩的解决方案来拟合训练数据, 并且针对预训练函数可能无界的情况导出了新的泛化误差界。需要注意的是, 以上结论成立的条件是所有句子之间是相互独立的, 这些假设在大模型预训练中并不严格成立。类似地，以复杂度为代表的泛化理论分析结果同样刻画了模型规模与样本数量对泛化性能的影响, 大模型训练过程中发现的扩展法则（Scaling law）也体现了以上对应关系[28], 关于扩展法则的详细讨论在第三章展开。

以信息论为基础的泛化分析同样将优化迭代T步之后的参数 W_T视作随机变量[29], 其随机性的来源有两个, 分别是算法的随机性和数据的随机性, 这也决定了相关理论结果能够同时体现算法与数据的影响。需要注意的是，以 PAC-Bayes 为代表的传统信息论泛化分析均假设损失函数为有界的，然而大语言模型使用的自回归损失为无界的，无界损失是大语言模型泛化分析带来的挑战。最近研究人员将 PAC-Bayes 扩展到了损失函数无界的设定下[30], 以下大模型的泛化误差界可以在同样的框架下直接得到

$$L_{\mathrm{PT}}(w) \leq \hat{L}_{\mathrm{PT}}(w) + \mathcal{O}\left(\sqrt{\frac{D_{\mathrm{KL}}(Q|P)}{mn}}\right),$$

其中 mn 代表预训练阶段样本数量，$D_{\mathrm{KL}}(Q|P)$ 代表预训练后的模型的后验分布 Q 与任意先验分布。

以上结论显示后验分布与先验分布之间的距离决定了泛化性能的好坏，其中先验分布 P 理论上其可以取不依赖训练数据的任意分布。当 P 取预训练任务对应的潜在模型分布 P^* 时，上式中的 KL 散度刻画了模型经过预训练之后的分布 Q 与该潜在未知分布 P^* 之间的距离。注意这里的 P^* 表示未知的模型分布，其与训练数据对应的数据分布有关，但是与采样得到的训练数据本身没有统计上的依赖关系。由此出发，预训练过后的模型参数分布与潜在分布之间的距离越近则泛化误差越小。以上分析将预训练理解为模型参数分布逐渐逼近潜在真实分布的过程，类似地，已有研究将大语言模型预训练过程理解为知识压缩的过程，并且观察到预训练过后的大模型可以作为压缩器[31]。除此之外，Allen-Zhu 等[32]使用合成数据集来评估语言模型存储知识的能力，并且得出了语言模型每个参数恰好能够存储 2 比特的知识的结论，照此估算，一个 7B 的模型能够储存 14B 比特的知识，这个数量超过英文维基百科和教科书所储存知识的总量。

1.4 大语言模型优化算法

大语言模型的优化相比传统机器学习和深度学习发生了深刻变化，从系统层面讲，模型规模的变大使得训练过程必须引入数据并行、模型并行和流水线并行等工程上的解决方案，从算法角度讲，大模型训练带来的巨大资源消耗使得开发更高效的优化器越来越受到关注。以 GPT3 的训练为例，其模型规模为 175B [33]，按照 FP32 格式加载模型参数需要占据 700GB 内存，梯度规模与模型相同，Adam 优化器则需要内存空间为梯度的 2 倍，则模型、梯度和优化器状态总共需要的内存为模型所需内存的 4 倍，实际优化过程中还存在激活值带来的内存消耗，如此大规模大内存消耗使得必须通过分布式方式才能满足。

本节重点介绍数据并行相关的理论结果，模型并行与流水线并行等方法目前理论研究还十分有限，相关理论有待进一步发展[34]。由于大语言模型预训练使用的数据来源广泛，使用数据并行方法时并不能简单地假设所有单机上的数据分布一致，而是需要考虑不同单机之间的数据异质性。不失一般性地考虑 m 个单机组成的分布式系统[35]，其中每个单机上的样本

数量为 n, 假设每个单机上的数据采样自不同的数据分布 $\{\mu_i\}_{i=1}^m$, 第 i 个单机上的优化目标记为：

$$\widehat{L_\iota}(w) = \frac{1}{n}\sum_{j=1}^{n} \ell\left(w, X_i^j\right),$$

其中 $\left\{X_i^j\right\}_{j=1}^n$ 为从分布 μ_i 上独立采样得到 n 个样本，整个分布式系统的优化目标为

$$\hat{L}_{\mathrm{PT}}(w) = \frac{1}{m}\sum_{i=1}^{m} \widehat{L_\iota}(w) = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n} \ell\left(w, X_i^j\right),$$

首先给出随机梯度下降算法（SGD）在异质数据并行设定下的收敛性结论, 与深度学习模型类似, 大模型的优化是典型的非凸优化问题, 其收敛性分析通常需要用到函数光滑条件, 另外在数据异质设定下还需要梯度方差小于σ^2的假设。基于以上假设, 其收敛性如下[35]

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}\left[\left\|\nabla \hat{L}(w^t)\right\|^2\right] = \mathcal{O}\left(\frac{\sigma}{\sqrt{mT}}\right),$$

根据以上理论结果可得, 单机设定下需要T^*步才能达到的优化误差在数据并行设定下只需要T^*/m步即可达到, 从理论上确保了分布式扩展的有效性。

虽然以上理论保证了分布式扩展的有效性，但实际运行中的系统依然面临着众多挑战，其中最核心的挑战来源于机器之间的通信瓶颈。缓解通信瓶颈的方法主要从通信方式和梯度压缩两个角度展开，其中对通信方式的改进包括去中心化、本地更新和异步等，梯度压缩方法包括量化和低秩近似等。理论工作方面， Markov 等[36]提出了量化分布式训练（Quantized Distributed Training，简称 QSDP）方法并且在大语言模型上进行了实验，其是对数据并行（Fully-Sharded Data-Parallel，简称 FSDP）训练的一个变体，它支持梯度和权重的量化，并提供了理论上的收敛保证。Wang 等[37]提出了通信高效训练框架 CocktailSGD，以克服在带宽约为 500M 的慢速网络上分布式训练大模型时遇到的通信挑战。CocktailSGD 结合了随机稀疏化、Top-K 稀疏化和量化等技术，实现了比单独使用任何一种技术都更大的压缩比，同时保持模型训练的收敛性，还通过理论分析证明了其在平滑非凸目标函数上的收敛性。基于梯度低秩假设， Vogels 等[38]提出了梯度压缩算法 PowerSGD 来降低通信量，该方法结合了动量方法和误差反馈技术（Error feedback），并且被用到了 DALL-E 的训练中[39]。Liu 等[40]

通过系统实验对异步本地随机梯度下降（Asynchronous Local-SGD）大语言模型训练中的性能进行了详细测试，但该场景下的理论分析还比较有限，有待进一步探索。

除了分布式扩展方式之外，高效优化器的开发也越来越受到研究人员的关注。针对大批量数据训练提出的LAMB使用了层自适应策略来加速深度神经网络的训练[41]，成功将BERT的训练时间从3天减少到76分钟，并且提供了该算法在一般非凸设置中的收敛性分析。最近 LAMB 已经被用于大模型分布式训练中[42]。为加速模型训练提出的 Adan（ADAptive Nesterov momentum algorithm）优化器重新开发了一种新的Nesterov动量估计方法，避免了计算梯度的额外开销，并采用该方法估计梯度的一阶和二阶矩[43]。大量实验结果表明，Adan在视觉、语言和强化学习任务上表现良好，相关理论分析也证明了Adan在非凸随机问题上的复杂度与最佳已知下界相匹配。针对大模型优化提出的 Sophia 是一个二阶优化器[44]，它使用轻量级的对角化海森矩阵估计作为预处理器，通过元素级裁剪来控制最差情况下的更新大小，从而减轻非凸性和不稳定的更新带来的负面影响。此外，论文还提供了理论分析，证明了 Sophia 的运行时间界限不依赖于损失的条件数。针对大模型设计的内存高效优化器 Galore 将梯度投影到低秩子空间中进行优化[45]，其理论分析了在训练过程中梯度矩阵会变得低秩，这一点与低秩适应方法（LoRA）假设参数空间低秩存在根本不同。Galore 与PowerSGD都存在梯度投影步骤，两者区别在于PowerSGD依旧是在原始参数空间执行优化步骤，而Galore则在投影子空间执行优化步骤，所以更省内存。

随着分布式训练规模的增大，去中心化和异步的更新方式将变得越来越重要，虽然传统非凸优化已有相关结论，但大模型场景下如何体现异质数据分布对收敛的影响是重要但是还未得到充分探索的方向。除数据并行相关的理论结果之外，模型并行与流水线并行等方法目前理论研究还有待进一步发展。

本章在统计学习理论框架下对大语言模型的表达能力、泛化分析和优化算法等方面进行了概述。表达能力方面，Transformer 架构已被证实能够逼近任意序列到序列的函数，这从理论上保证了大模型在序列建模方面的能力上限。泛化能力方面，模型需要应对来自不同数据分布的挑战，理论工具如信息论等被用来深化我们对模型泛化行为的理解。在优化算法方面，随着模型规模的增大，传统的优化方法面临内存和计算的挑战，因此，优化策略如数据并行和模型并行成为必需，近期工作开发了多种高效的优化器，并探索了分布

式训练中的通信瓶颈解决方案。本章的讨论不仅强调了理论在指导实践中的应用价值，也揭示了当前理论分析的局限性，并对未来的研究方向提出了期待。

2 大语言模型构建原则：理论指导实践

大语言模型涉及海量数据与超大模型的相互作用，从工程开发角度来讲，预训练和微调阶段均需要协调数据、模型和算法以提升模型的性能，从部署应用的角度来讲，以提示工程为依托的大模型推理方法关注如何更好地激发模型内在能力。本节综述以上问题对应的理论进展，同时讨论相关理论在实践指导中的潜在应用。

2.1 预训练理论及启发

2.1.1 超参迁移理论及其启发

扩展法则（Scaling Laws)描述的是模型性能（如泛化误差）与模型架构（如模型参数量）或者优化过程（如数据规模、训练计算量）随规模变化的关系。以往的研究表明，大模型的扩展法则在数学上呈现幂律关系。假设使用表示模型的泛化误差，使用表示影响模型性能的因素（如数据规模、训练计算量、模型参数量等），那么扩展法则可以表示为

$$L = k \cdot B^{\alpha} + c,$$

其中k, α, c是该幂律公式的参数，L表示模型的泛化误差。近年来，由于模型规模不断扩大，大模型的预训练对计算资源要求非常高，例如 Llama2-70B 模型的预训练时间高达 1720320 个 GPU 小时[16]。因此，直接根据大模型训练的结果调试超参数变得极为困难。如果能够获得模型的泛化误差的扩展法则，就可以根据扩展法则预测出模型的泛化能力。通常可以在小数据集和小模型上进行快速训练以获得扩展法则，然后使用该法则预测大模型的泛化能力。

Kaplan 等[46]发现了神经语言模型的扩展法则。令N表示非嵌入参数量，D表示数据集数量，C_{min}表示最大分配计算预算。若固定其中一个，则扩展法则可以表示为

$$L(N) = (N_c/N)^{\alpha_N}; \alpha_N \sim 0.076, \quad N_c \sim 8.8 \times 10^{13}$$

$$L(D) = (D_c/D)^{\alpha_D}; \alpha_D \sim 0.095, \quad D_c \sim 5.4 \times 10^{13}$$

$$L(C_{min}) = \left(C_{\mathrm{c}}^{min}/C_{min}\right)^{\alpha_C^{min}};\alpha_C^{min} \sim 0.050, \quad C_{\mathrm{c}}^{min} \sim 3.1 \times 10^8(\text{PF-days})$$

Kaplan 等发现，这些关系在N跨越 6 个数量级、D跨越 2 个数量级和C_{min}跨越 8 个数量级上都是成立的。有了扩展法则，大模型开发者可以提取预测大量级的模型的泛化误差，从而有效降低训练成本。

2.1.2 数据配比理论及其启发

预训练大语言模型的训练数据是来自多个领域训练数据的混合，而这些不同领域数据的配比是影响预训练大语言模型的最终性能的关键因素。但是，数据配比的比例目前依赖启发式方法或者定性策略。Ye 等[47]给出了数据配比的扩展法则，从而使得在数据配比上定量预测模型性能成为了可能。Ye 等发现，泛化误差在多个混合数据上呈现线性组合关系，即：

$$L(r_1 \ldots M) = \sum_{i=1}^{K} s_i L_i(r_1 \ldots M),$$

其中，M是训练数据中子领域的个数，$L(r_1 \ldots M)$表示总的泛化误差，$L_i(r_1 \ldots M)$表示第i个领域的泛化误差，s_i表示第i个领域泛化误差对总泛化误差的贡献度。Ye 等发现，对于单独某个领域的数据，其泛化误差满足幂律：

$$L_i(r_i) = c_i + k_i \exp(t_{ii} r_i),$$

其中L_i和i是第个领域的泛化误差和数据数量占比，c_i, k_i, t_{ij}为特定系数。假设测试数据中有K个领域的数据，那么总的泛化误差可以写为：

$$L(r_{1\ldots M}) = \sum_{i=1}^{K} s_i\, L_i(r_{1\ldots M}) = \sum_{i=1}^{K} s_i \left[c_i + k_i \exp\left(\sum_{j=1}^{M} t_{ij}\, r_j \right) \right],$$

和 Ye 等同期，Ge 等[48]也提出了数据配比的扩展法则。假设训练步数为s，某领域数据的占比为r。针对训练步数和数据占比，Ge 等提出了一种双变量的扩展法则：

$$L(s, r) = \left(\frac{A}{s^{\alpha}} + C \right) \frac{B}{r^{\beta}}$$

其中A，B，C是常数，α，β是需要拟合的指数。

2.2 有监督微调理论及启发

将大模型部署到专有的下游任务上是一种常见的大模型应用方式，其做法通常是将大模型在专有下游任务的数据上进行进一步微调。但是，这存在两个问题：一是在专有的下游任务上微调可能会造成大模型原有的通用能力减弱；二是大模型的超大参数量使得普通开发者无法有效对大模型进行微调。下面从理论角度讨论大模型微调过程中如何选择数据，并分析参数高效微调模块的表达能力。

2.2.1 数据选择理论及其启发

大模型的微调常常忽视了训练与微调之间的联系，导致大模型的微调常常无法充分利用预训练中获取的知识。因此，本小节总结了由泛化理论增强微调效果的工作。这些工作建立了预训练数据对微调效果的泛化误差的影响，利用预训练知识增强微调的效果。

将大语言模型直接在目标数据上进行微调可能会导致大模型在微调后只对齐了目标数据，遗忘了原来的知识。Kang 等[49]提出了要将大语言模型的微调分为两个阶段:前微调阶段（Pre-Finetuning）和目标微调阶段(Targeted Finetuning)。前微调阶段使用候选集（大量开源的未标注数据）对预训练模型进行微调，目标微调阶段选择专门的目标数据对模型进一步精确微调。两阶段微调的目标是让大语言模型的分布是从原始的分布慢慢迁移，最终使得大语言模型的分布包含目标数据的分布，而不是仅仅只能够对齐目标数据的分布。为了实现这个目标，需要从无标签的大规模开源数据中选择合适的数据对大语言模型进行微调。令D_P表示大规模开源数据集的分布，D_U表示前微调阶段的数据集分布，D_T表示目标分布，预训练大语言模型的初始参数为M^0，经过前微调阶段后（微调数据集为D_U)的模型参数为$M^*(D_U)$。在以上的数据分布中，D_P和D_T都是已知的，D_U而需要通过求解获得。在前微调阶段，使用D_U进行微调；在目标微调阶段，使用D_T进行微调。Kang 等提出的前微调阶段的最优数据选择定理为：

$$E_{x\sim D_T}[L(M^*(D_U),x)] \le E_{y\sim D_M^*}[L(M^*(D_U),y)] + k\cdot \mathrm{OT}(D_M^*, D_T) + O(\varepsilon),$$

其中$D_{M^*} = \lambda\cdot D_{U^*} + (1-\lambda)\cdot D_P$，标量λ的取值为$(0,1)$，表示最优传输距离$\mathrm{OT}(\cdot)$，$\mathrm{D}_{U^*}$表示

前微调阶段的最优数据选择。根据该定理，可以得到最优数据选择的具体表达为：

$$D_{U^*} = \underset{D_U \subset D_P}{\operatorname{argmin}} D_U \cdot \frac{\partial \mathrm{OT}(D_P, D_T)}{\partial D_P},$$

Liu 等[50]从学习理论（Learning Theory）的角度分析了在微调数据中加入预训练数据能够缓解目标任务的超额风险界限（Excess Risk Bound）。由此，Liu 等从理论出发，提出了一种能够增强目标任务泛化的策略，这个策略是从预训练数据中选择一小部分数据加入微调数据。假设原本的微调损失函数为：

$$F_n(\theta) \triangleq \frac{1}{\mathrm{n}} \sum_{i=1}^{n} f(\theta; x_i, y_i),$$

其中(x_i, y_i)表示原本的微调数据。令$\xi_i = (x_i', y_i')$表示从训练数据中选出的数据，这部分数据在微调时候的损失函数为：

$$H_m(\theta) \triangleq \frac{1}{m} \sum_{j=1}^{m} h\left(\theta; x_j', y_j'\right),$$

其中h函数是与目标任务相关的函数。Liu 等给出了这种方法的超额风险界限：

$$F\left(\theta_{f^*}\right) - F(\theta^*) \leq O\left(\frac{\alpha \log\left(n \triangle^2 / \alpha\right)}{n} + (1-\alpha)\delta^2\right),$$

其中$\delta^2 \triangleq \max_{\theta_t, \xi_{i_t}} E\left[\left|\nabla F(\theta_t) - \nabla h\left(\theta_t; \xi_{i_t}\right)\right|^2\right]$, $\xi_{i_t} = \left(x_{i_t}', y_{i_t}'\right)$表示第 t 步从训练集中选择的数据。因此，根据该超额风险界限，应该选择让δ^2变小的数据，从而降低超额风险。

2.2.2 参数高效微调理论及其启发

由于大语言模型的巨大的参数量，普通研究人员微调预训练大模型变得十分困难，Llama2 70B[16]的全量微调需要超过 780GB 的显存开销。参数高效微调能够以极低的参数变动对大语言模型进行微调，且能够达到与全量微调相当甚至更好的微调效果。主流的参数高效微调算法有 LoRA[51]、Prompt Tuning[52]、Adapter[53]等。在过去的几年，虽然参数高效微

调在实验效果上取得了逼近全量微调的效果，但是如何深入理解参数高效微调模块的表达能力仍然是一个难题。理解参数高效微调的表达能力有助于更好地使用参数高效微调算法，也可以帮助研究人员设计出更好的参数高效微调算法。本小节总结了近年来理解参数高效微调表达能力的工作。

LoRA 适配器方法使用两个低秩矩阵的乘积近似需要更新的参数的增量矩阵，即$\Delta W = BA$[4]，从而极大降低需要微调的参数量，大大降低对显存开销的要求。

Zeng 等[54]从理论上分析了 LoRA 适配器的表达能力，从理论上回答 LoRA 适配器需要的最小秩、最小秩与网络结构的关系（宽度、深度、结构）。对于全连接神经网络（Fully Connected Neural Networks），Zeng 等指出：假设存在一个网络$\overline{f}$，一个目标全连接神经网络f，存在 LoRA 适配器能够使得网络$\overline{f}$精准表示目标网络f，LoRA 适配器的秩须满足：

$$r \geq \left(f\text{网络的宽度}\right) \times \frac{f\text{网络的深度}}{\bar{f}\text{网络的深度}}.$$

对于 Transformer 网络，Zeng 等指出：假设存在一个网络$\overline{f}$，一个目标 Transformer 网络f，存在 LoRA 适配器能够使得网络$\overline{f}$精准表示目标网络f，LoRA 适配器的秩须满足：

$$r \geq \left(\text{嵌入维度}/2\right).$$

如果 LoRA 适配器的秩达不到上述最小要求，将会产生近似误差。

Zhu 等[55]指出了 LoRA 适配器中矩阵的不对称性。假设x为输入，那么有：

$$\left(W + \triangle W\right)x = Wx + \triangle Wx = Wx + BAx.$$

Zhu 等认为，矩阵B比矩阵A更为关键，因为需要将映射到最终需要的空间。假设$W \in R^{d_{out} \times d_{in}}$，$A \in R^{r \times d_{in}}, B \in R^{d_{out} \times r}$。Zhu 等从泛化误差的角度证明了只微调矩阵$B$比微调矩阵$BA$要更好，有：

$$|\text{gen}(\mu, \mathcal{A}_{BA})| \leq \sqrt{\frac{2rq\sigma^2 \ln 2}{n} \sum_{i \in J} \left(d_{in}^{(i)} + d_{out}^{(i)}\right)},$$

$$|\mathrm{gen}(\mu, \mathcal{A}_B)| \leq \sqrt{\frac{2rq\sigma^2 \ln 2}{n} \sum_{i \in \mathcal{J}} d_{out}^{(i)}},$$

其中$|\mathrm{gen}(\mu, \mathcal{A}_{BA})|$表示微调矩阵$B, A$的泛化误差界限，$|\mathrm{gen}(\mu, \mathcal{A}_B)|$表示微调矩阵$B$的泛化误差界限。由此，可以看到仅微调矩阵$B$比同时微调矩阵$B, A$要具有更好的泛化误差界限。Hayou等[56]也发现 LoRA 适配器的不对称性，矩阵B的学习率应该要比矩阵A的学习率要更大。这也反映了矩阵B承担着映射Ax到合适空间的重任，需要快速寻找到合适的矩阵B。

Prompt Tuning 通过在模型输入前面增加一些可学习的嵌入，使得模型可以以极低可学习参数适配到下游任务。Petrov 等[57]指出了 Prompt Tuning 存在的缺陷，其仅能对原本的注意力分数矩阵进行缩放而无法改变原本注意力分数的相对值，从而导致其无法学习预训练知识之外的新知识。其经过公式推导，有：

$$t_i = \sum_{j=1}^{p} A_{ij} W_V x_j,$$

其中W_V表示 Query 矩阵，A_{ij}表示x_i对x_j的注意力分数，x_j表示输入的 token，t_i表示输入经过一层预训练模型中的自注意力层的中间表达。全量微调的公式可以表达为：

$$t_i^{\mathrm{ft}} = \sum_{j=1}^{p} A_{ij}^{\mathrm{ft}} \left(W_V + \Delta W_V \right) x_j,$$

其中A^{ft}表示全量微调（Full Finetuning）后的注意力分数矩阵，ΔW_V表示全量微调后的 Query 增量矩阵，t_i^{ft}为经过一层预训练模型中的自注意力层的中间表达的全量微调版本。Prompt Tuning 的公式可以表达为：

$$t_i^{\mathrm{pt}} = A_{i0}^{\mathrm{pt}} W_V s_1 + \sum_{j=1}^{p} A_{ij}^{\mathrm{pt}} W_V x_j = A_{i0}^{\mathrm{pt}} W_V s_1 + \sum_{j=1}^{p} A_{ij} \left(1 - A_{i0}^{\mathrm{pt}}\right) W_V x_j$$
$$= A_{i0}^{\mathrm{pt}} W_V s_1 + \left(1 - A_{i0}^{\mathrm{pt}}\right) t_i,$$

其中A^{pt}表示经过 Prompt Tuning 后的注意力分数矩阵，t_i^{pt}为经过一层预训练模型中的自注意力层的中间表达的 Prompt Tuning 版本。注意到，相比于全量微调可以随意改变注意力分

数矩阵，Prompt Tuning 仅能对该矩阵进行缩放，全量微调可以随意改变 Query 矩阵，但是 Prompt Tuning 仅能够对原本预训练模型的中间表达增加一个偏置。因此，Prompt Tuning 无法通过微调学习到新知识，仅可以激发组合大模型在预训练阶段学习到的知识。

2.3 提示工程理论及其启发

提示工程（Prompt Engineering）已成为引导大语言模型解决复杂任务的重要技术。其重要性在于其能够显著提升人机交互的效率和效果。随着任务日益复杂，近期的先进提示工程方法已超越单轮交互的限制，发展到多轮交互，使得人类与大语言模型的互动更深入、更细致[33,58]。本节总结了利用控制论和信息论角度对提示工程建模的论文，这些方法可以加强人与大模型之间的交互。

2.3.1 信息论视角下的提示工程理论

从概念上讲，通信过程通常被建模为一个信息处理步骤的链条，涉及发射机（Sender）和接收机（Receiver）之间信息的编码、传输和解码。发射机首先将信息编码（Encode），然后通过信道传输由接收机接收，然后接收机解码(Decode)。接收机解码后会通过信道对发射机做出反馈。由于编码、传输和解码过程中均可能存在噪声，上述流程通常是多轮的。

Song 等[59]提出了可以从通信理论的视角看待提示工程建模，将提示（Prompt)与大语言模型的交互当作一个通信系统。由此，Song 等从通信理论的视角给出提示工程的数学建模：

$$X \xrightarrow{g_{\omega_T}} P_T \xrightarrow{f_\theta} P_A \xrightarrow{h_{\omega_A}} Y,$$

其中X表示模型的输入，g_{ω_T}表示由输入到 Prompt 的映射，$P_T \xrightarrow{f_\theta} P_A$表示大模型依照 Prompt 做出的回答的映射，$P_A \xrightarrow{h_{\omega_A}} Y$表示依照大模型回答得出最终输出$Y$的映射。提示工程的建模可以表达为最大化输入和输出之间的互信息，即：

$$\max_{\omega_T,\omega_A} I(X,Y) = \max_{\omega_T,\omega_A} I\left(X, h_{\omega_A} \circ f_\theta \circ g_{\omega_T}(X)\right),$$

其中$f_\theta \circ g_{\omega_T}(X) = f_\theta(g_{\omega_T}(X))$，最大化输入和输出的互信息可以理解为最小化用户的

Prompt 和大模型之间的误解。

$X \xrightarrow{g_{\omega_T}} P_T$为提示模板建模（Prompt Template Engineering），即为输入选择合适的提示，让大模型更容易理解用户的输入，可以理解为通信系统中的发射机编码过程，其可以表达为：

$$\max_{\omega_T} I(X, P_A) = \max_{\omega_T} I\left(X, f_\theta \circ g_{\omega_T}(X)\right),$$

其充当了“编码器”的角色，通过以模型能够理解的方式对信息进行编码，从而弥合用户与大模型之间的差距，然后从大模型中引出知识。

$P_A \xrightarrow{h_{\omega_A}} Y$为提示应答工程（Prompt Answer Engineering），其旨在寻找易于人类理解的提示，可以理解为通信系统中的解码过程，其可以表达为：

$$\max_{\omega_A} I(P_T, Y) = \max_{\omega_A} I\left(P_T, h_{\omega_A} \circ f_\theta(P_T)\right),$$

其充当了“解码器”的作用，使得大模型输出的答案能够和人类期望高度对齐。在解码过程中，由于输出空间无限，大模型生成的输出往往除了预期答案外还携带冗余信息。答案工程旨在限制输出空间并提取目标答案，使最终输出（即Y）与终端用户的期望高度一致。

为了尽量减少用户与大模型之间的误解，可以学习通信系统的多轮交互，用户与大模型之间的提示工程建模也使用多轮交互的方法，即：

$$max_{\omega_{T_i}} \sum_{i=1}^{M} I\left(X, f_\theta \circ g_{\omega_{T_i}}(X)\right).$$

2.3.2 控制论视角下的提示工程理论

控制理论源自对自动控制系统的研究，它旨在探讨系统如何通过反馈和自我调节来实现目标。可以利用 Prompt 与大模型的反馈和自我调节为大模型寻找最优 Prompt。Bhargava 等[60]利用控制论的反馈机制，提出了一种逐步增加 Prompt 长度的方法。Luo 等[61]提出了一个针对与大语言模型多轮交互的最优控制框架。根据最优控制理论，提示工程可以建模为以下：确定合适的评估函数；为 Prompt 候选集建立更新规则；解决所得到的最优控制问题。

本章综述了大语言模型在预训练、微调及部署应用方面的理论进展和实际应用挑战。在预训练阶段，通过理论洞察如扩展法则和数据配比理论，开发者可以更有效地预测和优化模型的泛化能力，从而在有限的计算资源下实现最优的性能。微调阶段，理论工作帮助理解如何选择微调数据和如何设计高效的微调策略，特别是在大模型参数庞大且训练成本高昂的情况下。在模型部署和应用阶段，提示工程提供了一种新的方法论，通过精细控制信息的编码和解码过程，增强了人机交互的效果和效率。整体来看，这些理论和方法为大语言模型的开发和应用提供了重要的指导和支持，使其在处理复杂任务时更为高效和精确。

3 大语言模型涌现能力：数学机理分析

随着 GPT 系列模型的发展，大语言模型受到了业界和学界的高度关注，引发了重要的技术变革[62-63]。随着扩展法则（Scaling Law）[46, 64]的广泛探索，研究人员发现当扩大模型的参数规模、数据规模以及计算资源，语言模型的能力会出现显著的提升，甚至会出现某些特定能力的涌现[65]。从探索大模型的基础问题出发，提供扩展法则（Scaling Law）的理解，研究大模型的涌现能力如情境学习（In-Context Learning, ICL）与思维链（Chain-of-Thought, CoT）的数学机理，将为更好地理解和设计大模型提供重要的理论支持。

3.1 扩展法则(Scaling Law)机理分析

为了有效刻画预训练的规模效应，研究人员提出使用“扩展法则（Scaling Law）”进行大语言模型能力的定量建模[46, 64]。本节将分别从优化视角分析损失函数对扩展法则的影响；从泛化视角分析随数据规模变化的泛化与扩展法则的联系；从表达能力视角分析随模型规模变化的表达能力与扩展法则的联系。

3.1.1 优化视角：扩展法则受损失函数影响

现有的扩展法则主要是针对语言建模的“下一个词元预测损失”构建[46, 64]，本节将首先从优化视角，分析损失函数对扩展法则的影响。在大模型预训练过程中，可以使用语言建模损失(即词元预测的平均交叉熵损失)建立起可预测的模型能力演变趋势曲线，即扩展法则。基本思路是利用较小算力训练语言模型（对应小的数据规模或模型尺寸），根据这些较低成本获取到的模型效果表现去拟合模型性能函数，进而利用函数趋势去预测大尺寸模型（对应充分训练模型）的效果。扩展法则给出了一种模型性能可预测性的数学分析工具，可以用于

监测模型效果的变化，及时发现与修正预训练过程中所出现的错误。

在许多情况下，规模对性能的影响通常可以通过扩展法则进行方法论预测。例如，常用的损失度量如“下一个词元预测”的交叉熵损失构建较为平衡，容易建立起函数拟合模式，其规模曲线已被证明根据经验跨度超过七个数量级。但现有的基于语言建模损失（即建立在词元级别的交叉熵损失[46]）的扩展法则仍存在一定的问题。在训练中，直接对建立在词元级别的交叉熵损失进行优化使之尽可能地减少，在实践中我们更关心大模型在实际任务上的表现。该有益的扩展法则仅产生对损失的预测，而没有完全扩展到实践中遇到的实际任务性能[65-68]。具体地，Henighan 等[66]将语言建模损失的减少与下游任务性能的提高建立一定的对应关系，证明在具备自回归交叉熵损失的 Transformer 单一架构下，扩展法则适用于各种数据模态的生成建模，例如生成语言、图像和视频建模等。然而，值得注意的是，语言建模损失的减少并不总是表明下游任务的模型性能有所提高。Mckenzie 等[67]发现在某些任务中会出现逆缩放现象，随着语言建模损失的减少，任务性能会意外地变得更差。此外，Wei 等[65]和 Ganguli 等[68]的工作发现某些特定下游任务的性能似乎并没有随着规模的变化而不断提高，因此此类任务无法提前预测，如扩展法则无法刻画“涌现能力”所带来的性能跃升。这些工作表明词元级别的损失与任务性能之间存在不完全匹配的现象，即基于词元交叉熵损失的扩展法则在部分任务的性能预测中失效。

基于现有扩展法则在损失函数上的局限性，可进一步构建下游任务性能和各项能力涌现情况与多种指示性度量指标之间的关系，如探索训练损失的变种以刻画模型的多维度能力水平等。Srivastava 等[69]和 Schaeffer 等[70]提出在模型规模增大时所产生的性能跃升可能源于评测指标（损失函数）不够平滑、测试模型尺寸过于离散。Hu 等[71]针对该问题提出对任务进行分类的启发式指标、新的评估策略和实例级扩展曲线拟合，并根据损失扩展法则推导了任务扩展法则，从而可以量化模型在任务上的表现以提高模型能力的可预测性。

3.1.2 泛化视角：泛化误差随数据规模变化

扩展法则刻画了大语言模型在下游任务的预测能力随着训练数据和模型参数规模的变化规律[46, 64]。本节将从泛化视角，分析随数据规模变化的泛化与扩展法则的联系。

GPT-2 论文中明确提出了大语言模型是多任务学习器的观点[72]，基于此可以在多任务学习理论框架下对大语言模型的预训练优化过程以及泛化误差进行建模。自然地，大语言模型

训练过程中所展现出的扩展法则（能力预测）可以与其跨任务的泛化性能建立联系，甚至对于仅基于交叉熵损失的扩展法则无法解释的涌现能力（如情境学习、思维链等能力），均与其跨任务的泛化性能有关[65]。因此，从受数据规模影响的泛化视角理解扩展法则，可以表述为：扩展法则刻画了多任务泛化误差随着训练数据的变化规律。

Michaud 等[73]基于语言模型量子知识点的假设提出了针对扩展法则的理论解释，使用语言模型梯度自动将模型行为分解为一组不同的技能（量子），并假设对于每个量子，训练数据中需要使用量子的 τ 个示例阈值才可学习该量子，以此完成对数据扩展的研究。此外，Arora 等[74]将大语言模型的交叉熵损失与语言任务的基本技能联系起来，结合了扩展法则和统计学习工具，建立了对于涌现能力理论分析框架，提出扩展法则量化了预训练模型中强大的归纳偏差现象，以及数据扩展将影响大语言模型的泛化性能和涌现能力。进一步，Deletang 等[75]将预训练看作知识压缩，从压缩视角建立了信息论与大语言模型预训练之间的关系，基于信息论的泛化研究将进一步建立与预训练相关的扩展法则（包括数据扩展）与模型泛化的联系，从而提供扩展法则更深的见解。

3.1.3 表达能力：表达能力随模型规模变化

扩展法则刻画了大语言模型在下游任务的预测能力随着训练数据和模型参数规模的变化规律[46, 64]。本节将从表达能力视角，分析随模型规模变化的表达能力与扩展法则的联系。

追溯 GPT 系列模型的发展：遵循 GPT-1 [75]的生成式、仅解码器的 Transformer 架构，GPT-2 [72]将参数规模增加到 1.5B，GPT-3 [62]将模型参数缩放到更大的尺寸 175B。尽管 GPT-3 的论文[62]没有明确讨论 LLM 的涌现能力，但我们可以观察到可能超越基本扩展法则的巨大性能飞跃[46]，例如较大的模型具有明显更强的情境学习能力）。总体而言，GPT-3 可以被视为“大力出奇迹”的成功典范。GPT-4 [63] 进一步具备更强的解决复杂任务的能力，其中引入了可预测扩展法则，可以在模型训练期间用少量计算准确预测最终性能。Sharma 等[76]说明当数据充足时，训练有素的神经网络实现的测试损失按照网络参数数量的幂律进行扩展，并从理论上将与模型大小相关的扩展指数与数据流形的内在维度建立联系，展示了模型大小对模型扩展能力的影响。此外，Zeng 等[54]和 Merrill 等[77]分别从理论上提供了利用权重矩阵低秩自适应（LoRA）对预训练模型进行参数高效微调、以及使用思维链进行预训练模型能力激发的表达能力分析。然而，尽管经验证明，将模型扩展到相当大的参数规模可以导

致模型表达能力的巨大增加[78]，但其理论基础在很大程度上仍未得到探索。在未来研究中，可以进一步从理论上探索模型的表达能力随模型参数量的变化，从表达能力角度为扩展法则提供新的理论支撑和理解。

3.2 情境学习(In-context Learning)能力机理分析

情境学习（In-context Learning，ICL) 是大语言模型预训练后涌现出的重要能力，对于训练过程中没有见过的新任务，只需要给定几个输入输出样例对和测试输入，大语言模型就能实现在新任务上的泛化。与传统的学习范式不同，ICL 并没有显示更新参数就能实现新任务的泛化，这使得其成为激活大模型能力的高效方式。本节将分别从优化视角，分析 ICL 的隐式更新机理；从泛化视角，提供 ICL 的泛化误差分析；从表达能力视角，分析 ICL 的函数逼近性质。

3.2.1 优化视角：情境学习隐式更新机理

大语言模型完成 ICL 推理的过程中不需要进行显式梯度更新，却能在新任务上取得良好的效果。大语言模型是否存在隐式的更新过程成为解释 ICL 推理的一个直观的想法，并引起了大家关注。Transformer 架构是当前大模型使用的主流的架构，理解 Transformer 中注意力机制在大模型 ICL 推理过程中发挥的作用，是理解大模型隐式梯度更新的核心。本节将首先从优化视角，提供 ICL 的隐式更新机理分析。

Aizerman 等[79]和 Irie 等[80]首先提出神经网络的梯度下降过程可以被看作为线性注意力的对偶形式。在此基础上，Dai 等[81]在线性注意力的设定下，将 ICL 推理过程解释为对大模型的隐式微调。Akyurek 等[82] 提出，通过构建特定的模型权重，Transformer 可以执行平移、乘法、除法、仿射等基本操作，这些操作可以进一步组合起来执行梯度下降算法。Von Oswald 等[83]采用了另一种构造方法，使得在单个或多个线性注意力层上的推理过程可以等效地看作在进行一步或多步梯度下降算法以完成 ICL 线性回归任务。在该权重构造方法的基础上，随后的工作对 Transformer 在自回归设定下的 ICL 能力进行了更深入的探索，并指出该设定下的 ICL 推理类似于执行在线梯度下降[84-85]。Zhang 等[86]从优化角度进行了收敛性分析，具体提供了通过梯度流学习线性模型与 Transformer 上下文学习的优化收敛分析相一致的证据。Tarzanagh 等[87]提出了一种新颖的观点，将 Transformer 视为支持向量机（SVMs），

建立了自注意机制的优化几何与硬间隔 SVM 问题之间的关联。

这些研究从隐式梯度更新的角度，对大模型的 ICL 推理能力做出了一定的探索。但已有从隐式梯度更新角度对大模型 ICL 能力的研究，一般都基于线性注意力的设定，这与实际中大模型采用的 softmax 非线性注意力机制仍存在明显的差距；此外，现有研究给出的大模型的隐式梯度更新机理，更多为形式上的类比，该隐式更新过程的具体细节，包括损失函数和训练数据的形式，还有待进一步的完善；最后，现有部分研究在研究隐式更新机理时采用了特定的权重构造方法，即当模型权重满足理论上的某些形式时，大语言模型的 ICL 过程可以被看作隐式梯度更新，然而实际完成训练后的模型权重并不一定满足这些特定的形式，这就会导致理论分析的失效，因此，如何不依赖权重构造方法对隐式更新机理进行探究，同样成为当前研究大模型 ICL 能力的一个重要问题。

3.2.2 泛化视角：情境学习泛化误差分析

如 3.2.1 节介绍，已有一些研究工作对 “将 ICL 视为梯度下降算法的隐式执行” 进行了初步探讨，这些研究试图回答 ICL 是什么，但并未解释大语言模型是如何涌现出 ICL 能力。本节将从泛化视角，提供 ICL 的泛化误差分析。

Xie 等[88]从贝叶斯视角对 ICL 泛化性提供了初步洞见，将 ICL 视为隐式的贝叶斯推理，而预训练的大型语言模型在预测推理过程中被视为直观地推断概念。Wang 等[89]和 Jiang 等[90]同样持有类似的贝叶斯观点。然而这些工作[88-90]都假定大语言模型是预先固定的，并没有考虑优化过程对 ICL 能力的影响。针对该问题，可考虑优化对 ICL 泛化能力的影响[91-93]，建立预训练与 ICL 阶段之间的联系。进一步，Zhang 等[93]研究了自回归预训练对 ICL 能力的影响，但值得强调的是，已有工作对先验和后验分布的假设是比较苛刻和受限的，对 ICL 的理解仍然有限，亟待发展。

为了进一步探索 ICL 表达能力的起源，部分学者试图从泛化性的角度提出新的见解。Wei 等[65]发现大语言模型训练过程中所展现出的 ICL 能力与其跨任务的泛化性能有关。具体来讲，ICL 能力表明大语言模型在新任务上具备少样本泛化能力[62]，GPT-2 文中[72]明确提出了大语言模型是多任务学习器的观点，可以在多任务学习理论框架下对大语言模型的优化过程及泛化误差进行建模。基于上述观点，Arora 等[74]将大语言模型的交叉熵损失与语言

任务的基本能力联系起来，并结合规模效应和统计学习工具，建立了对于涌现能力的理论分析框架。进一步，Deletang 等[75]将预训练看作知识压缩过程，建立了信息论与大语言模型预训练之间的关系，相关研究表明即便是语言模型也可以被用作图像的压缩器，提供了对预训练过程进行建模的全新视角。

虽然已有工作从贝叶斯、泛化性等角度分析了大模型 ICL 能力产生的可能原因，但现有工作仍然聚焦在简单的线性回归的函数学习任务上，如何在更符合真实场景的自回归语言建模下，在多任务学习理论下、使用信息论或 PAC-Bayesian 等工具对大语言模型的预训练优化过程及 ICL 推理阶段的泛化误差进行理论分析，仍然是个亟待解决的重点和难点问题。

3.2.3 表达能力：情境学习函数逼近性质

本节将从表达能力视角，分析 ICL 的函数逼近性质。Zhang 等[86]研究了具有单个线性自注意力层的 Transformer 中 ICL 的动态，该线性自注意力层通过线性回归任务的梯度流进行训练，具有适当随机初始化的梯度流可以找到目标函数的全局最小值从而具备拟合任何线性函数的 ICL 能力。Huang 等[94]在单个 softmax 自注意力层的设定下分析了 Transformer 的阶段式训练动态，同样表明具有通过梯度下降训练的 softmax 自注意力，以便在上下文中逼近线性函数类。Chen 等[95]研究梯度流的动力学，以训练多头 softmax 自注意力模型，用于多任务线性回归的上下文学习。进一步，Cheng 等[96]同时去除了关于线性的两个严格假设，在更符合实际的非线性条件（包括基于 softmax 自注意力的 Transformer，学习非线性函数任务）提供了理论和经验证据，证明非线性 Transformer 可以且确实通过训练中学会了执行梯度下降算法以在上下文中学习非线性函数。该工作回答了使用 ICL 对 Transformer 进行能力激发，能够具备逼近非线性函数的能力。

此外，Bai 等[97]基于 ReLu 和的逼近定理完成 ReLu 自注意力的权重构造，使得在单层的 ReLu 自注意力上的推理过程可以等效地看作在预训练模型基础执行一步隐式的梯度下降算法。Akyurek 等[82]和 Bai 等[97]的工作均表明 Transformer 可以在上下文中实现广泛的标准机器学习算法，例如最小二乘法、岭回归、广义线性模型的凸风险最小化（例如逻辑回归）以及两层梯度下降神经网络，对各种上下文数据分布具有近乎最佳的预测能力。

已有工作在不同自注意力模型（线性自注意力，softmax 自注意力，ReLu 自注意力）设

置下，分析 ICL 在线性或非线性的函数拟合任务上的表现。在考虑 Transformer 架构的其他模块（例如位置编码）的设定下分析模型在更通用任务上的能力，仍是重要的研究问题。

3.3 思维链(Chain-of-thought)能力机理分析

思维链（Chain-of-Thought，CoT) 是大语言模型预训练后涌现出的重要能力。之前的研究强调，精心设计的提示对大语言模型的表现非常重要[98-99]。特别地，Wei 等[100]发现思想链提示对于涉及算术或推理的任务至关重要，生成答案的正确性可以通过大模型输出中间结果的修改来显著提高。

Li 等[101]的研究表明，思维链的成功可以归因于将组合函数的情境学习分解为两个不同的阶段：专注于组合每一步相关的数据，以及在上下文中学习单步组合函数。这种分解揭示了思维链在处理复杂推理任务时的机制。但该工作主要关注于基于 MLP 的任务上，子问题本质上是简单线性回归的实例。进一步地，Feng 等[102]分析了自回归 Transformer 结合思维链提示方法的表达能力，证明了深度受限的 Transformer 模型在不增加模型大小的情况下，无法直接为基本算术/方程任务生成正确答案，除非模型大小相对于输入长度呈超多项式增长。Merrill 等[77]也从研究表达能力出发，揭示了中间生成的 Transformer 可以学习内容的局限性，进一步可能沿着 Malach [103]的路线从学习理论的角度对带有 CoT 的 Transformer、使用不同类型的微调提供理论分析，为如何更好地允许模型使用思维链提供新的见解。

目前思维链背后的基本机制在很大程度上仍然有待进一步探索，如思维链成功提升大模型性能表现的根本原因，以及大模型在直接回答数学/推理问题方面是否有局限性等。在自回归语言建模任务上，探索思维链如何影响具有更复杂结构、更长组成链和更广泛子问题的任务的训练，从优化理论、泛化理论、表达能力理论分析大模型在生成 CoT 解的能力将是未来研究的重要方向。

本章深入探讨了大语言模型的核心理论与实践挑战，特别是扩展法则、情境学习（ICL）与思维链（CoT）的机理分析。我们从优化、泛化和表达能力三个维度解析了大模型能力的涌现，揭示了模型规模、数据规模和计算资源增加时模型性能的显著提升。扩展法则为理解和预测大模型性能提供了重要的定量工具，而 ICL 和 CoT 的分析则展示了模型在处理复杂任务时的适应性和高效性。通过这些理论洞察，我们能更好地设计和优化大模型，以应对更

广泛的实际应用挑战，同时这也为未来大模型的研究方向提供了理论基础和实践指南。

4 结论

本文从统计学习视角对大语言模型理论研究进展与趋势进行了综述，首先在统计学习框架下阐述了大语言模型的理论分析框架，主要涉及模型表达能力、算法收敛快慢与泛化误差分析。相比于传统机器学习来讲，大语言模型的理论分析需要考虑数据分布的异质性、模型结构的特殊性与优化算法的系统性，本文在综合考虑以上大模型特性基础上总结了相关理论结果，并且进行了讨论展望。除此之外，本文从理论指导实践的原则出发总结了预训练和微调阶段的相关理论，分别涉及预训练阶段的超参数迁移和数据配比，微调阶段的数据选择和低秩近似以及信息论和控制论视角下的提示工程。最后，本文对大语言模型涌现出的扩展法则、情境学习能力和思维链能力相关的机理分析进行了分析汇总，以期能够对工程实践带来相应的启发作用。大语言模型的工程和理论均处在快速发展阶段，除本文之外前期已有部分涉及大模型理论的综述工作，包括大语言模型综述[78]、大语言模型微调综述[104]、大语言模型对齐综述[105]与大语言模型高效训练综述[106]等。

参 考 文 献

[1] BISHOP C M, BISHOP H. Deep learning - foundations and concepts [M]. Berlin: Springer, 2023: 357-403.

[2] MOHRI M, ROSTAMIZADEH A, TALWALKAR A. Foundations of machine learning [M]. Cambridge, Massachusetts:MIT press, 2018: 1-29.

[3] CYBENKO G V. Approximation by superpositions of a sigmoidal function [J]. Mathematics of Control, Signals and Systems, 1989, 2: 303-314.

[4] HORNIK K. Approximation capabilities of multilayer feedforward networks [J]. Neural Networks, 1991, 4: 251-257.

[5] LU Z, PU H, WANG F, et al. The expressive power of neural networks: A view from the width [C/OL]//Neural Information Processing Systems,2017[2024-06-20].https://api.semanticscholar.org/CorpusID:3235741.

[6] HANIN B, SELLKE M. Approximating continuous functions by relu nets of minimal width [J]. arXiv preprint arXiv: 1710.11278, 2023.

[7] YUN C, BHOJANAPALLI S, RAWAT A S, et al. Are transformers universal approximators of sequence-to-sequence functions? [J]. arXiv preprint arXiv: 1912.10077, 2019.

[8] KRATSIOS A, ZAMANLOOY B, LIU T, et al. Universal approximation under constraints is possible with transformers [J]. arXiv preprint arXiv: 2110.03303. 2021.

[9] KIM J, KIM MY, MOZAFARI B. Provable memorization capacity of transformers [C/OL]// International Conference on Learning Representations. 2023[2024]. https://api.semanticscholar.org/CorpusID:259298704.

[10] TAY Y, DEHGHANI M, BAHRI D, et al. Efficient transformers: A survey [J]. ACM Computing Surveys, 2020, 55: 1 - 28.

[11] CHILD R, GRAY S, RADFORD A, et al. Generating long sequences with sparse transformers [J]. arXiv preprint arXiv: 1904.10509, 2019.

[12] KATHAROPOULOS A, VYAS A, PAPPAS N, et al. Transformers are RNNs: Fast autoregressive t ransformers with linear attention [C/OL]// International Conference on Machine Learning. 2020[2024]. https://api.semanticscholar.org/CorpusID:220250819.

[13] YANG K, ACKERMANN J, HE Z, et al. Do efficient transformers really save computation? [J]. arXiv preprint arXiv: 2402.13934, 2024.

[14] WEN K, DANG X, LYU K. Rnns are not transformers (yet): The key bottleneck on in-context retrieval [J]. arXiv preprint arXiv: 2402.18510, 2024.

[15] ZHANG C, BENGIO S, HARDT M, et al. Understanding deep learning (still) requires rethinking generalization [J]. Communications of the ACM, 2021, 64: 107 - 115.

[16] TOUVRON H, MARTIN L, STONE KR, et al. Llama 2: Open foundation and fine-tuned chat models [J]. arXiv preprint arXiv: 2307.09288, 2023.

[17] BOMMASANI R, HUDSON D A, ADELI E, et al. On the opportunities and risks of foundation models [J]. arXiv preprint arXiv: 2108.07258, 2021.

[18] SHALEV-SHWARTZ S, BEN-DAVID S. Understanding machine learning: From theory to algorithms [M]. Cambridge:Cambridge University press, 2014: 19-50.

[19] TOUVRON H, LAVRIL T, IZACARD G, et al. Llama: Open and efficient foundation language models [J]. arXiv preprint arXiv: 2302.13971, 2023.

[20] WANG H, ZHAO H, LI B. Bridging multi-task learning and meta-learning: Towards efficient training and effective adaptation [J]. arXiv preprint arXiv: 2106.09017, 2021.

[21] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners [J]. OpenAI blog, 2019, 1(8): 1-9.

[22] HELLSTRÖM F, DURISI G, GUEDJ B, et al. Generalization bounds: Perspectives from information theory and pac-bayes [J]. arXiv preprint arXiv: 2309.04381, 2023.

[23] LEI Y, YING Y. Fine-grained analysis of stability and generalization for stochastic gradient descent [J]. arXiv preprint arXiv: 2006.08157, 2020.

[24] LI Y, ILDIZ ME, PAPAILIOPOULOS D, et al. Transformers as algorithms: Generalization and stability in context learning [C]// International Conference on Machine Learning. [S.l.]: PMLR, 2023: 19565-19594.

[25] LOTFI S, FINZI M, KUANG Y, et al. Non-vacuous generalization bounds for large language models [J]. arXiv preprint arXiv: 2312.17173, 2023.

[26] SOLOMONOFF R J. A formal theory of inductive inference. part I[J]. Information and Control, 1964, 7(1): 1-22[2024-06-20].https://www.sciencedirect.com/science/article/pii/S0019995864902232.DOI: https://doi.org/10.1016/S0019-9958(64)90223-2.

[27] LI M, VITÁNYI P, et al. An introduction to kolmogorov complexity and its applications: volume 3 [M]. Berlin:Springer, 2008: 47-86.

[28] KAPLAN J, MCCANDLISH S, HENIGHAN T, et al. Scaling laws for neural language models [J]. arXiv preprint arXiv: 2001.08361, 2020.

[29] ALQUIER P. User-friendly introduction to pac-bayes bounds [J]. Found. Trends Mach. Learn., 2021, 17:

174-303.

[30] CASADO I, ORTEGA L A, MASEGOSA A R, et al. Pac-bayes-chernoff bounds for unbounded losses [J]. arXiv preprint arXiv: 2401.01148, 2024.

[31] DEL'ETANG G, RUOSS A, DUQUENNE PA, et al. Language modeling is compression [J]. arXiv preprint arXiv: 2309.10668, 2023.

[32] ALLEN-ZHU Z, LI Y. Physics of language models: Part 3.3, knowledge capacity scaling laws [J]. arXiv preprint arXiv: 2404.05405, 2024.

[33] BROWN TB, MANN B, RYDER N, et al. Language models are few-shot learners [J]. arXiv preprint arXiv: 2005.14165, 2020.

[34] ZHUANG B, LIU J, PAN Z, et al. A survey on efficient training of transformers [J]. arXiv preprint arXiv: 2302.01107, 2023.

[35] YUAN K. Lecture 6: Stochastic gradient descent [EB/OL]. (2023101-11)[2024-06-20]. https://kunyuan827.github.io/dlopt2023/.

[36] MARKOV I, VLADU A, GUO Q, et al. Quantized distributed training of large models with convergence guarantees [J]. arXiv preprint arXiv: 2302.02390, 2023.

[37] WANG J, LU Y, YUAN B, et al. Cocktailsgd: fine-tuning foundation models over 500mbps networks [C]//ICML'23: Proceedings of the 40th International Conference on Machine Learning. Honolulu, Hawaii, USA: JMLR.org, 2023: 19.

[38] VOGELS T, KARIMIREDDY S P, JAGGI M. Powersgd: Practical low-rank gradient compression for distributed optimization [J]. arXiv preprint arXiv: 1905.13727, 2019.

[39] SANGHI A, CHU H, LAMBOURNE J G, et al. Clip-forge: Towards zero-shot text-to-shape generation [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2022: 18603-18613.

[40] LIU B, CHHAPARIA R, DOUILLARD A, et al. Asynchronous local-sgd training for language modeling [J]. arXiv preprint arXiv: 2401.09135, 2024.

[41] YOU Y, LI J, REDDI SJ, et al. Large batch optimization for deep learning: Training bert in 76 minutes [J]. arXiv preprint arXiv: 1904.00962, 2019.

[42] JIANG Z, LIN H, ZHONG Y, et al. Megascale: Scaling large language model training to more than 10,000 gpus [J]. arXiv preprint arXiv: 2402.15627, 2024.

[43] XIE X, ZHOU P, LI H, et al. Adan: Adaptive nesterov momentum algorithm for faster optimizing deep

models [J]. arXiv preprint arXiv: 2208.06677, 2022.

[44] LIU H, LI Z, HALL DLW, et al. Sophia: A scalable stochastic second-order optimizer for language model pre-training [J]. arXiv preprint arXiv: 2305.14342. 2023.

[45] ZHAO J, ZHANG Z A, CHEN B, et al. Galore: Memory-efficient llm training by gradient low-rank projection [J]. arXiv preprint arXiv: 2403.03507, 2024.

[46] KAPLAN J, MCCANDLISH S, HENIGHAN T, et al. Scaling laws for neural language models [J]. arXiv preprint arXiv: 2001.08361, 2020.

[47] YE J, LIU P, SUN T, et al. Data mixing laws: Optimizing data mixtures by predicting language modeling performance [J]. arXiv preprint arXiv: 2403.16952, 2024.

[48] GE C, MA Z, CHEN D, et al. Data mixing made efficient: A bivariate scaling law for language model pretraining [J]. arXiv preprint arXiv: 2405.14908, 2024.

[49] KANG F, JUST HA, SUN Y, et al. Get more for less: Principled data selection for warming up fine-tuning in llms [J]. arXiv preprint arXiv: 2405.02774, 2024.

[50] LIU Z, XU Y, XU Y, et al. Improved fine-tuning by better leveraging pre-training data [J]. Advances in Neural Information Processing Systems, 2022, 35: 32568-32581.

[51] HU EJ, SHEN Y, WALLIS P, et al. Lora: Low-rank adaptation of large language models [J]. arXiv preprint arXiv: 2106.09685, 2021.

[52] LI XL, LIANG P. Prefix-tuning: Optimizing continuous prompts for generation [J]. arXiv preprint arXiv: 2101.00190, 2021.

[53] HOULSBY N, GIURGIU A, JASTRZEBSKI S, et al. Parameter-efficient transfer learning for NLP [C]// International conference on machine learning. [S.l.]: PMLR, 2019: 2790-2799.

[54] ZENG Y, LEE K. The expressive power of low-rank adaptation [J]. arXiv preprint arXiv: 2310.17513, 2023.

[55] ZHU J, GREENEWALD K, NADJAHI K, et al. Asymmetry in low-rank adapters of foundation models [J]. arXiv preprint arXiv: 2402.16842, 2024.

[56] HAYOU S, GHOSH N, YU B. Lora+: Efficient low-rank adaptation of large models [J]. arXiv preprint arXiv: 2402.12354, 2024.

[57] PETROV A, TORR PH, BIBI A. When do prompting and prefix-tuning work? A theory of capabilities and limitations [J]. arXiv preprint arXiv: 2310.19698, 2023.

[58] Muktadir GM. A brief history of prompt: Leveraging language models [J]. arXiv preprint arXiv: 2310.04438, 2023.

[59] SONG Y, HE Y, ZHAO X, et al. A communication theory perspective on prompting engineering methods for large language models [J]. arXiv preprint arXiv: 2310.18358, 2023.

[60] BHARGAVA A, WITKOWSKI C, SHAH M, et al. What's the magic word? A control theory of llm prompting [J]. arXiv preprint arXiv: 2310.04444, 2023.

[61] LUO Y, TANG Y, SHEN C, et al. Prompt engineering through the lens of optimal control [J]. arXiv preprint arXiv: 2310.14201, 2023.

[62] BROWN T, MANN B, RYDER N, et al. Language models are few-shot learners [J]. Advances in Neural Information Processing Systems, 2020, 33: 1877-1901.

[63] ACHIAM J, ADLER S, AGARWAL S, et al. Gpt-4 technical report [J]. arXiv preprint arXiv: 2303.08774, 2023.

[64] HOFFMANN J, BORGEAUD S, MENSCH A, et al. Training compute-optimal large language models [J]. arXiv preprint arXiv: 2203.15556, 2022.

[65] WEI J, TAY Y, BOMMASANI R, et al. Emergent abilities of large language models [J]. arXiv preprint arXiv: 2206.07682, 2022.

[66] HENIGHAN T, KAPLAN J, KATZ M, et al. Scaling laws for autoregressive generative modeling [J]. arXiv preprint arXiv: 2010.14701, 2020.

[67] MCKENZIE I, LYZHOV A, PARRISH A, et al. The inverse scaling prize[EB/OL]. [2024-06-20]. https://github.com/inverse-scaling/prize.

[68] GANGULI D, HERNANDEZ D, LOVITT L, et al. Predictability and surprise in large generative models [C]//Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency. New York: ACM, 2022: 1747-1764.

[69] SRIVASTAVA A, RASTOGI A, RAO A, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models [J]. arXiv preprint arXiv: 2206.04615, 2022.

[70] SCHAEFFER R, MIRANDA B, KOYEJO S. Are emergent abilities of large language models a mirage? [J]. Advances in Neural Information Processing Systems, 2024, 36: 1-12.

[71] HU S, LIU X, HAN X, et al. Unlock predictable scaling from emergent abilities [J]. arXiv preprint arXiv: 2310.03262, 2023.

[72] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners [J]. OpenAI blog, 2019, 1(8): 1-9.

[73] MICHAUD E, LIU Z, GIRIT U, et al. The quantization model of neural scaling [J]. Advances in Neural Information Processing Systems, 2024, 36: 1-12.

[74] ARORA S, GOYAL A. A theory for emergence of complex skills in language models [J]. arXiv preprint arXiv: 2307.15936, 2023.

[75] DELÉTANG G, RUOSS A, DUQUENNE P A, et al. Language modeling is compression [J]. arXiv preprint arXiv: 2309.10668, 2023.

[76] SHARMA U, KAPLAN J. Scaling laws from the data manifold dimension [J]. Journal of Machine Learning Research, 2022, 23(9): 1-34.

[77] MERRILL W, SABHARWAL A. The expresssive power of transformers with chain of thought [J]. arXiv preprint arXiv: 2310.07923, 2023.

[78] ZHAO WX, ZHOU K, LI J, et al. A survey of large language models [J]. arXiv preprint arXiv:2303.18223,2023 .

[79] AIZERMAN MA, BRAVERMAN EM, ROZONOER LI. Theoretical foundation of potential functions method in pattern recognition [J]. Avtomatika i Telemekhanika, 1964, 25(6): 917-936.

[80] IRIE K, CSORDÁS R, SCHMIDHUBER J. The dual form of neural networks revisited: Connecting test time predictions to training patterns via spotlights of attention [C]//International Conference on Machine Learning.Baltimore, Maryland: PMLR, 2022: 9639-9659.

[81] DAI D, SUN Y, DONG L, et al. Why can gpt learn in-context? Language models implicitly perform gradient descent as meta-optimizers [J]. arXiv preprint arXiv: 2212.10559, 2022.

[82] AKYÜREK E, SCHUURMANS D, ANDREAS J, et al. What learning algorithm is in-context learning? Investigations with linear models [J]. arXiv preprint arXiv: 2211.15661, 2022.

[83] VON OSWALD J, NIKLASSON E, RANDAZZO E, et al. Transformers learn in-context by gradient descent [C]//International Conference on Machine Learning. Seattle, Washington:PMLR, 2023: 35151-35174.

[84] VON OSWALD J, NIKLASSON E, SCHLEGEL M, et al. Uncovering mesa-optimization algorithms in transformers [J]. arXiv preprint arXiv: 2309.05858, 2023.

[85] DING N, LEVINBOIM T, WU J, et al. Causallm is not optimal for in-context learning [J]. arXiv preprint arXiv: 2308.06912, 2023.

[86] ZHANG R, FREI S, BARTLETT PL. Trained transformers learn linear models in-context [J]. arXiv preprint arXiv: 2306.09927, 2023.

[87] TARZANAGH DA, LI Y, THRAMPOULIDIS C, et al. Transformers as support vector machines [J]. arXiv preprint arXiv: 2308.16898, 2023.

[88] XIE SM, RAGHUNATHAN A, LIANG P, et al. An explanation of in-context learning as implicit Bayesian inference [J]. arXiv preprint arXiv: 2111.02080, 2021.

[89] WANG X, ZHU W, SAXON M, et al. Large language models are latent variable models: Explaining and finding good demonstrations for in-context learning [J]. Advances in Neural Information Processing Systems, 2024, 36: 1-9.

[90] JIANG H. A latent space theory for emergent abilities in large language models [J]. arXiv preprint arXiv: 2304.09960, 2023.

[91] LI Y, ILDIZ ME, PAPAILIOPOULOS D, et al. Transformers as algorithms: Generalization and stability in in-context learning [C]//International Conference on Machine Learning. Seattle, Washington: PMLR, 2023: 19565-19594.

[92] WIES N, LEVINE Y, SHASHUA A. The learnability of in-context learning [J]. Advances in Neural Information Processing Systems, 2024, 36: 1-9.

[93] ZHANG Y, ZHANG F, YANG Z, et al. What and how does in-context learning learn? Bayesian model averaging, parameterization, and generalization [J]. arXiv preprint arXiv:2305.19420, 2023.

[94] HUANG Y, CHENG Y, LIANG Y. In-context convergence of transformers [J]. arXiv preprint arXiv: 2310.05249, 2023.

[95] CHEN S, SHEEN H, WANG T, et al. Training dynamics of multi-head softmax attention for in-context learning: Emergence, convergence, and optimality [J]. arXiv preprint arXiv: 2402.19442, 2024.

[96] CHENG X, CHEN Y, SRA S. Transformers implement functional gradient descent to learn nonlinear functions in context [J]. arXiv preprint arXiv: 2312.06528, 2023.

[97] BAI Y, CHEN F, WANG H, et al. Transformers as statisticians: Provable in-context learning with in-context algorithm selection [J]. Advances in Neural Information Processing Systems, 2024(36): 1-9.

[98] JIANG Z, XU FF, ARAKI J, et al. How can we know what language models know? [J]. Transactions of the Association for Computational Linguistics, 2020(8): 423-438.

[99] GU J, HAN Z, CHEN S, et al. A systematic survey of prompt engineering on vision-language foundation models [J]. arXiv preprint arXiv: 2307.12980, 2023.

[100] WEI J, WANG X, SCHUURMANS D, et al. Chain-of-thought prompting elicits reasoning in large language models [J]. Advances in Neural Information Processing Systems, 2022(35): 24824-24837.

[101] LI Y, SREENIVASAN K, GIANNOU A, et al. Dissecting chain-of-thought: Compositionality through in-context filtering and learning [J]. Advances in Neural Information Processing Systems, 2024(36): 1-9.

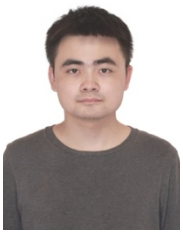
[102] FENG G, ZHANG B, GU Y, et al. Towards revealing the mystery behind chain of thought: A theoretical perspective [J]. Advances in Neural Information Processing Systems, 2024(36): 1-9.

[103] MALACH E. Auto-regressive next-token predictors are universal learners [J]. arXiv preprint arXiv: 2309.06979, 2023.
[104] DING N, QIN Y, YANG G, et al. Delta tuning: A comprehensive study of parameter efficient methods for pre-trained language models [J]. arXiv preprint arXiv: 2203.06904, 2022.
[105] CASPER S, DAVIES X, SHI C, et al. Open problems and fundamental limitations of reinforcement learning from human feedback [J]. arXiv preprint arXiv: 2307.15217, 2023.
[106] DUAN J, ZHANG S, WANG Z, et al. Efficient training of large language models on distributed infrastructures: A survey [J]. arXiv preprint arXiv: 2307.15217, 2023.

作者简介

刘勇 中国人民大学，长聘副教授，博士生导师，国家级高层次青年人才。长期从事机器学习基础理论研究，共发表论文 100 余篇，其中以第一作者/通讯作者发表顶级期刊和会议论文近 50 篇，涵盖机器学习领域顶级期刊 JMLR、IEEE TPAMI、Artificial Intelligence 和顶级会议 ICML、NeurIPS 等。主持国家自然科学面上/基金青年、北京市面上项目、中科院基础前沿科学研究计划、腾讯犀牛鸟基金、CCF-华为胡杨林基金等项目。

胡啸林 中国人民大学，高瓴人工智能学院在读博士生，研究方向为联邦学习、大语言模型高效微调，CCF 学生会员。在 ICLR、NeurIPS 与 KDD 等顶级会议发表多篇论文，参与小米揭榜挂帅——端侧大语言模型个性化高效微调项目。



唐鹏威 中国人民大学，高瓴人工智能学院在读博士生，研究方向为模型压缩、不平衡学习、大语言模型高效微调，CCF 学生会员。在 CVPR、ACML 等会议发表多篇论文。

龚子瑄 中国人民大学，高瓴人工智能学院在读直博生，研究方向为大模型优化和泛化基础理论，CCF 学生会员。参与国家自然科学基金面上项目——大语言模型上下文学习的数学机理分析和设计。