中国科学: 信息科学 2026 年 第 56 卷 第 4 期: 850–867
SCIENTIA SINICA Informationis
《中国科学》杂志社
SCIENCE CHINA PRESS
论文

大语言模型任务辨识能力的非平凡上界存在性证明及改进研究

倪宣明, 赵翘楚, 黄嵩*

北京大学软件与微电子学院, 北京 100871
* 通信作者. E-mail: huangsong@ss.pku.edu.cn

收稿日期: 2024–11–06; 修回日期: 2025–02–25; 接受日期: 2025–11–06; 网络出版日期: 2026–03–13

摘要 大规模语言模型 (LLMs) 在解决自然语言任务方面取得了显著进步, 缩小了人类与人工智能之间的差距. 然而, LLM 的性能仍然对提示的细微变化非常敏感. 其中引起本文注意的一个现象是, 为 LLM 提供几个问题 – 答案配对的示例, 即使示例的答案是随机选择的, 仍然可以显著提高其性能. 在上下文学习中, LLM 利用示例的机制能够分解为任务辨识 (task recognition) 和任务学习 (task learning) 两种, 而随机标注示例对性能的提升主要归功于任务辨识. 这一现象表明, LLM 在任务辨识上仍有不足. 本文主要研究 LLM 在任务辨识上不足的可能原因, 指出这一不足可能源于自然语言数据的长尾分布特征和任务之间固有的相似性. 本文进一步构建理论模型对这一猜想进行论证, 而后通过一系列实验验证了理论假设与理论结果, 分析了模型参数量等因素对模型任务辨识能力的影响. 最后, 本文进一步探究了针对 LLM 任务辨识不足的可能改进方向.

关键词 大规模语言模型 (LLM), 任务辨识, 少样本提示, 数据分布, 经验损失

1 引言

近些年, 大规模语言模型 (large language models, LLMs) 在许多自然语言任务上展现了优秀的性能, 极大地缩短了人工智能与人类的差距 [1,2]. 然而, 最近的一些实验证据显示, 大语言模型的性能在很大程度上受到提示中一些与任务本身关系不大的细节影响 [3,4]. 这使得对大语言模型的提示 (prompt) 的研究作为提升性能的关键之一广受关注 [5~7]. 以由若干个问题 – 答案示例组成的上下文提示 (in-context prompt) [8] 为例, 其性能关于提示中的样例并不稳定, 表现出较大的随机性 [9,10]. 最近的研究发现, 大语言模型的性能会受到样例的选择 [11,12]、样例的顺序 [13]、标签的选择 [14,15] 等多种因素 [16,17] 的影响.

其中引起本文注意的一个现象是: Min 等 [18] 通过实验发现, 使用随机标签构造的上下文提示和使用真实标签构造的上下文提示在提升模型在某些数据集上的评分方面几乎没有差别. 他们认为, 上下文提示对模型性能的提升主要归功于其为模型提供了输入的分布和输出的格式, 而非标签本身的正确性. 基于这一现象, Pan 等 [19] 进一步指出, 上下文提示对大语言模型性能的提升可以分解为两种不同的机制: 任务辨识 (task recognition) 和任务学习 (task learning). 其中, 任务辨识指的是大语言模型根据提示选择合适的预训练先验分布的能力, 任务学习指的是大语言模型从上下文样本中更新预训练先验分布的能力 [19]. 他们的实验结果表明, 少量样本的上

引用格式: 倪宣明, 赵翘楚, 黄嵩. 大语言模型任务辨识能力的非平凡上界存在性证明及改进研究. 中国科学: 信息科学, 2026, 56: 850–867, doi: 10.1360/SSI-2024-0329
Ni X M, Zhao Q C, Huang S. Existence proof and improvement study of nontrivial upper bounds on task recognition of large language models. Sci Sin Inform, 2026, 56: 850–867, doi: 10.1360/SSI-2024-0329

© 2026《中国科学》杂志社

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

Underspecified prompt	Specified prompt
Sentence: effective but too-tepid biopic Label: 1 Sentence: I am tired. Label: 0 Sentence: a masterpiece four years in the making . Label:	Judge whether the following sentence is about a movie. Sentence: effective but too-tepid biopic Answer: True Judge whether the following sentence is about a movie. Sentence: I am tired. Answer: False Judge whether the following sentence is about a movie. Sentence: a masterpiece four years in the making . Answer:

图 1　未明确说明提示与明确说明提示.

Figure 1　Underspecified versus specified prompt.

下文提示主要通过增强模型的任务辨识能力来提升性能, 这一结果为该现象提供了合理的支持. 而这一分解的理论基础在于, 大语言模型的输出是其根据提示预测的下一位字符块 (token) 的分布, 而这一分布可以按照任务做以下分解 [20].

$$\Pr(\text{output} \mid \text{prompt}) = \int_{\text{task}} \Pr(\text{output} \mid \text{prompt}, \text{task}) \Pr(\text{task} \mid \text{prompt})\, d(\text{task}). \tag{1}$$

在实际应用中最贴合这一分解的是, 最近的一些大语言模型成功地利用了混合专家模型 (mixture of experts, MoE) [21, 22]. 具体地, 混合专家 transformer 模型中的每一个混合专家层由若干个子网络和一个门控网络组成, 并在训练和推断中由门控网络动态地激活这些子网络 [23]. 本文认为, 混合专家模型将任务辨识与任务学习分离开, 由若干个子网络学习不同的子分布, 并由门控网络学习如何辨识不同的子分布. 混合专家模型的成功为上述分解提供了现实基础.

基于上述分解, 提升大语言模型的性能可以从任务辨识和任务学习两方面进行. 本文认为, 随机标签的上下文提示对模型性能的提升这一现象反映了大语言模型针对零样本提示在任务辨识上仍存在不足. 为了说明这一点, 首先本文需要区分未明确说明提示与明确说明提示 [24, 25]. 图 1 左边展示的提示属于未明确说明提示, 从这一少样本提示的单一样例中难以推断出提示要求执行的具体任务. 不论是大语言模型还是人类, 在面对这类提示时, 都要根据少样本提示中的所有样例来推断提示所属的任务. 因此, 少样本提示的示例对任务辨识起到的作用是直观的. 相对地, 图 1 右边展示的提示属于明确说明提示, 少样本提示的每一个样例中都有任务的明确描述. 对于这类提示, 人类即使没有多个样例, 通常也能几乎完美地判断提示所涉及的任务. 理论上, 一个最优的模型在面对明确说明的零样本提示时, 应该至少有人类水平的任务辨识能力. 但是, 按照 Min 等 [18] 以及 Pan 等 [19] 的实验方法, 本文发现随机标签的明确说明少样本提示依然对模型的性能有显著的提升. 这一结果表明, 大语言模型在这些任务上的任务辨识能力尚未达到最优, 仍存在进一步提升的空间.

具体如图 2 所示, 对于一个要求以 “是” 或 “否” 回答的任务, 模型的输出分布中可能包含若干种类的输出, 即若干个子分布的混合. 模型的任务辨识能力指的是模型输出分布中正确的子分布所占的权重, 模型的任务学习能力指的是正确的子分布下模型回答问题的正确率.

基于这一观察, 本文探索了大语言模型在任务辨识上不足的产生原因和改进方法. 通过对模型和数据分布做一定的假设, 本文证明了在某些情况下, 由于自然语言数据分布的特征以及不同任务之间的相似度, 模型在训练过程中倾向于混淆相似任务, 导致训练得到的模型任务辨识的正确率低于理论最优值. 此外, 任务与任务之间的出现频率差距越大, 相似度越高, 则模型在对应任务上的任务辨识能力就越弱. 这一结果在一定程度上给出了大语言模型任务辨识能力不足的一个理论解释. 基于这些理论结果, 本文进一步讨论了潜在的改进方向, 以提升大语言模型在任务辨识上的表现.

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

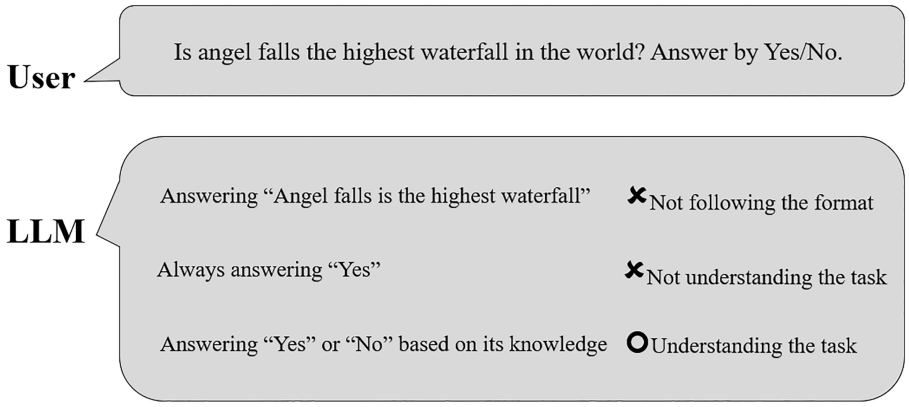

图 2　大语言模型根据提示输出的几种情况.

Figure 2　Some typical situations of LLM's response.

在实验部分, 本文首先参考了之前论文的实验设计, 基于理论讨论的假设, 提出了估计模型在某些测试数据集上任务辨识能力的方法. 随后, 本文在多个模型和数据集上应用这一方法, 比较了它们的任务辨识能力差异, 并讨论了模型参数规模、微调等对实验结果的影响. 此外, 本文还通过在合成数据集和真实数据集上的实验验证了理论假设的现实合理性和理论结论的正确性.

本文的剩余章节结构如下. 第 2 节定义了全文使用的记号, 给出了理论推导的预备知识. 第 3 节是理论部分, 先后给出了本文的两个关键定理及其证明, 讨论了任务辨识不足的改进方向. 第 4 节通过实验估计了模型在测试数据集上的任务辨识能力并验证了理论假设和结论. 第 5 节是本文的结论.

2　预备知识

首先规定本文中出现的一些记号, 用 $[N]$ 表示集合 $\{1,2,\ldots,N\}$, 用 (c,c_0) 表示将字符串 c 与字符块 (token) c_0 拼接成的字符串, 用 $c(1:l)$ 表示字符串 c 的前 l 位构成的子字符串. 这里的字符串指的是由字符块构成的序列.

在本文的设定中, 大语言模型即为拟合自然语言分布的一个分布. 参考之前的研究 [26,27], 本文假设自然语言数据的分布是多个子分布按一定比例的混合. 记总的分布为 M, 每个子分布为 M_i, 并用 $M(x)$ 记样本 x 在 M 中的概率, 则有如下假设.

假设1 (参见文献 [26,27])　自然语言数据的分布可以分解为 N 个子分布按比例 c_i, $\sum_{i=1}^{N} c_i = 1$ 的混合. 对于每个样本 x, 有 $M(x) = \sum_{i\in[N]} c_i M_i(x)$.

假设 1 提出的这种性质可以在很多种类的自然数据中观察到. 例如 Zhu 等 [28] 的研究发现, 一些图片数据集中的图片按照类别和更加细致的子类别 (图片中物体的角度、人的姿态等) 进行分类后, 每一个类别在数据集中的频率呈现出长尾分布的特征. van Horn 等 [29] 给出的更加具体的一个例子是, 鸟的图片这一大类可以按照鸟的品种、拍摄角度、鸟所处的环境等分为若干个小类, 即鸟的图片的分布可以分解为若干子分布的混合.

而对于自回归 (autoregressive) 语言模型而言, 其目的是针对输入的提示输出下一位字符块 (token) 的概率分布 [30,31], 假设 1 中每个样本 x 为一对提示字符串与下一位字符块的组合, 记为 (z,y). 从分布 M 中取出的每一个样本为 $(z,y)\sim M$.

不同子分布即为不同的 (z,y) 的对应关系, 例如, 提示 z 可能来源于不同的语境或领域, 对下一位字符块的预测也分为不同的类型. 更加直观地, Zoph 等 [32] 在对混合专家模型的研究中发现, 预训练混合专家模型中不同的专家网络擅长生成不同类型的内容, 如标点符号、连接词、数字等.

设模型的预训练数据集由 n 个从分布 M 中独立同分布取出的样本构成, 记为 $Z=\{(z_1,y_1),\ldots,(z_n,y_n)\}$. 并且设训练集 Z 中由子分布 M_i 中取出的样本构成 Z 的子集 Z_i. 也就有 $Z=\sqcup_{i\in[N]} Z_i$, 其中 $\sqcup$ 表示不交并. 并

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

且记 $n_i = |Z_i|$ 为子集 Z_i 中的训练数据个数, 有 $\sum_{i\in[N]} n_i = n$.

在实际情况下, 大语言模型的预训练数据集是由自然语言中的字符串构成的集合. 在预训练的过程中, 每个长度为 l 的字符串被分解为 $l-1$ 个提示与字符块的组合. 这里要特别说明的是, 自然语言中的同一个字符串分解出的提示与字符块的组合并不一定属于同一个子分布. 相反地, 一般来说它们更可能属于不同的子分布. 也就是说, 从真实数据出发确定 (z, y) 的分布, 首先应确定字符串的分布, 再从字符串出发得到 (z, y) 的分布. 本文认为, 在预训练数据集足够大的情况下, 从分布 M 中取样得到的数据集的分布可以近似于从真实数据得到的分布.

将预训练算法记为 A, 预训练得到的模型记为 $h = A(Z)$. 对于任意提示 z, 模型的输出 $h(z)$ 是全部字符块上的一个概率分布, $\Pr(h(z) = y)$ 表示这一概率分布中字符块 y 的占比. 预训练的优化目标一般为最小化模型在预训练数据集上的经验损失. 给定损失函数 L, 本文定义经验损失如下.

定义1　任一模型 h 在数据集 Z 上的经验损失为 $\overline{\mathbf{err}}(h) = \frac{1}{|Z|}\sum_{(z,y)\in Z} L(h(z), y)$. 其中, $L(h(z), y)$ 表示分布 $h(z)$ 与集中于 y 上的单点分布的损失. 本文也将模型 h 在数据集 Z_i 上的经验损失记为 $\overline{\mathbf{err}}_i(h)$.

本文需要指出, 最近的一些机器学习模型, 尤其是大语言模型, 训练集十分大, 往往不经过一个完整的周期 (epoch) 的训练 [8,33], 其训练误差通常也并不收敛. 但是, 这并不意味着经验损失的减少没有意义. 一般来说, 若两个模型 h_1 和 h_2 满足 $\overline{\mathbf{err}}(h_1) < \overline{\mathbf{err}}(h_2)$, 那么训练的结果将会更倾向于模型 h_1, 训练得到的模型 h 的输出与模型 h_1 也更为相似. 这也是接下来的理论讨论的基础.

3　理论结果

3.1　任务辨识不足的理论解释

本文试图通过自然数据分布的特征解释大语言模型分辨任务的能力不足这一现象. 本文认为这一现象有两个原因. 一是自然语言数据分布中不同的子分布之间的出现频率有很大的差距. 经验表明自然语言的分布呈现出一种长尾的特征 [34], 大量的子分布的数据会以极小的频率出现在训练集中 [35], 甚至可能不存在于训练集中 [36]. 二是子分布之间有一定的相似性. 由于这两个原因, 用某个常见的子分布的经验去补全来自某个少见的子分布的提示可能会得到比正确判断子分布更少的训练损失. 于是训练得到的模型会趋向于更加模糊地分辨各个子分布.

为了从理论上验证这一猜想, 首先需要考察模型 h. 对于给定输入 z, 模型的输出 $h(z)$ 是所有可能字符块 y 上的一个分布. 给定 z 的边缘分布 $M(z)$, 则模型 h 同样定义了一个 (z, y) 上的分布, 记为 M', $M'(z, y) = M(z)\Pr(h(z) = y)$. 若此分布也可以按照 N 个子分布分解, 即存在 c_i, M'_i, 使得 $M' = \sum_{i\in[N]} c'_i M'_i$, 则可以定义 $\Pr(h_i(z) = y) = M'_i(z, y)/\sum_{y'} M'_i(z, y')$ 为模型判断输入 z 属于子分布 i 后的下一位字符块分布. 由于分布 M' 与 M'_i 满足其在 z 上的边缘分布分别与 M, M_i 在 z 上的边缘分布相同, 因此有

$$\Pr(h(z) = y) = \frac{\sum_{i\in[N]} c'_i M'_i(z, y)}{\sum_{y'}\sum_{i\in[N]} c'_i M'_i(z, y')} = \frac{\sum_{i\in[N]} c'_i M'_i(z, y)}{M(z)}. \tag{2}$$

同理对 h_i 有 $\Pr(h_i(z) = y) = M'_i(z, y)/M_i(z)$. 因此可以得到

$$\Pr(h(z) = y) = \sum_{i\in[N]} h(i \mid z)\Pr(h_i(z) = y), \quad h(i \mid z) = \frac{M_i(z)}{M(z)} c'_i. \tag{3}$$

也就是模型 h 的预测可以分为两部分, 其中 $h(i \mid z)$ 为模型预测输入 z 属于子分布 i 的概率, $h_i(z)$ 为模型在认为 z 属于子分布 i 的情况下对下一位字符块做出的预测.

接下来需要解决的问题是如何确定 c'_i 和 M'_i. 首先, 由于可能存在的提示 z 的个数一定不小于全部子分布的个数, 因此一定至少存在一组 c'_i 和 M'_i 使得 $M' = \sum_{i\in[N]} c'_i M'_i$ 以及 M'_i 在 z 上的边缘分布与 M_i 相同. 但是这样的 c'_i 和 M'_i 并不一定唯一. 虽然也可以通过其他条件限制, 如要求 $\sum_{i\in[N]} \mathrm{Dist}(M_i, M'_i)$ 取极小值, 来获得

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

唯一的一组 c_i' 和 M_i', 但是由于本文的理论结论是构造性的, 只需要存在一个模型 h 满足本文定理所陈述的条件, 并不依赖于分解的唯一性, 因此本文并不做出这样的限制.

综合以上讨论, 本文提出如下假设, 这一假设等价于假设每个模型 h 定义的分布 M' 都可以做分解 $M' = \sum c_i' M_i'$.

假设2　本文假设模型 h 所作推断可以分解为两部分, 一部分是模型对提示所属子分布的推断, 一部分是在确定提示所属子分布的条件下对下一位字符块的推断. 记 $h(i \mid z)$ 为模型预测输入 z 属于子分布 i 的概率, $\Pr(h_i(z) = y)$ 为模型在认为 z 属于子分布 i 的情况下下一位字符块是 y 的概率, 则有

$$\Pr(h(z) = y) = \sum_{i \in [N]} h(i \mid z) \Pr(h_i(z) = y). \tag{4}$$

为了对训练误差进行比较, 需要定义一个比较的基准, 本文将任一模型与一个在训练和推断时能够完美分辨子分布的模型相比较.

定义2　设 $\mathbf{OPT}$ 为一个能够完美分辨子分布的模型, 即 $\mathbf{OPT}(i \mid z) = 1, z \sim M_i$ 以及 $\mathbf{OPT}(i \mid z) = 0, z \sim M_j, j \neq i$. 并且对于 $\forall i \in [N]$, $\mathbf{OPT}_i$ 总是以数据集 Z_i 训练出最优的模型.

直觉上能够完美分辨子分布并且在每个子分布上都是最优的模型 $\mathbf{OPT}$ 应该最小化训练损失. 但是在实际情况下则不一定这样. 若子分布 i 和 j 存在一定程度的相似性, 例如 M_i 与 M_j 的全变差距离足够小, 并且 Z_j 足够小, 则可能有

$$\mathbf{E}_{(z,y)\sim M_j}[L(\mathbf{OPT}_i(z), y)] < \mathbf{E}_{(z,y)\sim M_j}[L(\mathbf{OPT}_j(z), y)]. \tag{5}$$

为了理解式 (5) 在实际情况中能否成立, 本文以一个极端情况为例进行说明. 给定具有一定相似性的两个子分布 M_i 与 M_j, 若混合分布中 M_j 的占比为 0, 而 M_i 的占比显著大于 0, 则显然有式 (5) 中的不等号严格成立. 如果式 (5) 中的公式关于子分布占比具有一定的连续性, 那么一定存在一组大于 0 的占比, 使得在该占比的条件下式 (5) 成立. 考虑到自然语言分布的长尾特征, 有一定相似度且在混合分布中占比差距足够大的两个子分布存在的可能性很高, 因此假设条件 (5) 成立具有一定的现实合理性.

接下来本文将证明, 在这一条件下, 存在一个任务辨识能力弱于 $\mathbf{OPT}$ 的模型, 然而其经验误差好于 $\mathbf{OPT}$.

首先为了简化讨论, 本文假设模型的任务辨识能力对于同一个子分布中的提示是一致的.

假设3　对于模型 $h = A(Z)$, 假设对任意的 i 和任意的 $z \sim M_j$, $h(i \mid z)$ 为只与 i, j 有关的常数. 为了简化记号, 规定对于 $z \sim M_j$, $h(i \mid z) = \alpha_{ij}$. 并且假设对任意 i, j, 有

$$\frac{\alpha_{ji}}{n_j} = \frac{\alpha_{ij}}{n_i}. \tag{6}$$

假设 3 的第一部分认为 $h(i \mid z, z \sim M_j)$ 是只与 i 和 j 有关的常数. 实际上这一条件可以放宽至 $h(i \mid z, z \sim M_j)$ 至少以大概率落在某一非平凡区间内, 仍然能够使本文的理论结论成立, 并且不需要大幅度改变证明过程. 为了理解假设 3 的第二部分, 考虑后验概率 $\Pr(i \mid z) \propto \Pr(z \mid i) \Pr(i)$. 其中, $\Pr(i)$ 可以由 $n_i/|Z|$ 代替. 也就是说假设 $h(i \mid z)$ 为只与 i, j 有关的常数等价于假设 $\Pr(z \mid i)$ 为常数, 即从一个子分布里取出任一样本的概率都是相同的. 在此之上根据后验公式易得式 (6).

假设4　首先假设式 (5) 成立, 并假设 $\mathbf{OPT}_i$ 与 $\mathbf{OPT}_j$ 在 Z_j 上的经验误差之差为 k_{ij}, $k_{ij} > 0$.

$$\overline{\mathbf{err}}_j(\mathbf{OPT}_i) = \overline{\mathbf{err}}_j(\mathbf{OPT}_j) - k_{ij}. \tag{7}$$

接着考虑模型 h 在数据集 Z_i, Z_j 上的训练损失. 错误的判断训练数据所属的子分布会对 h_i, h_j 的性能造成影响. 不过在 $n_i \gg n_j$ 的情况下, 由式 (6) 可得 $\alpha_{ji} \ll 1$, 因此可以认为 $h_i = \mathbf{OPT}_i$. 至于 h_i 在 Z_i, Z_j 上的性能, 简化起见, 这里都假设为关于 α_{ij} 的线性关系.

$$\overline{\mathbf{err}}_j(h_j) \leqslant \overline{\mathbf{err}}_j(\mathbf{OPT}_j) + k_{jj}\alpha_{ij}, \quad \overline{\mathbf{err}}_i(h_j) \leqslant \overline{\mathbf{err}}_i(\mathbf{OPT}_i) + k_{ji}\alpha_{ij}. \tag{8}$$

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

基于以上假设和定义, 可以证明存在某个 $\alpha_{ij} > 0$ 的模型 h, 在训练集上的训练损失小于 **OPT**.

定理1　假设数据集 $Z = (Z_i, Z_j)$, 即只由两个子分布的数据构成. 若 $k_{jj} < k_{ij}$, $k_{jj} < k_{ji}$ 成立, 则存在 $\alpha_{ij} > 0$ 的 h, 使得 $\overline{\mathbf{err}}(h) < \overline{\mathbf{err}}(\mathbf{OPT})$.

证明　首先将 $\overline{\mathbf{err}}(h)$ 展开,

$$\frac{1}{|Z|}\sum_{(z,y)\in Z} L(h(z),y) = \frac{1}{n_i+n_j}\left(\sum_{(z,y)\in Z_i} L(h(z),y) + \sum_{(z,y)\in Z_j} L(h(z),y)\right). \tag{9}$$

对括号内左边这项, 利用假设 2 和 4, 有

$$\begin{aligned}\sum_{(z,y)\in Z_i} L(h(z),y) &= \sum_{(z,y)\in Z_i} (\alpha_{ii}L(h_i(z),y) + \alpha_{ji}L(h_j(z),y))\\ &= \sum_{(z,y)\in Z_i} (\alpha_{ii}L(\mathbf{OPT}_i(z),y) + \alpha_{ji}L(h_j(z),y))\\ &\leqslant (1-\alpha_{ji})\sum_{(z,y)\in Z_i} L(\mathbf{OPT}_i(z),y) + \alpha_{ji}\sum_{(z,y)\in Z_i} L(\mathbf{OPT}_i(z),y) + n_i\alpha_{ji}k_{ji}\alpha_{ij}\\ &= \sum_{(z,y)\in Z_i} L(\mathbf{OPT}_i(z),y) + n_jk_{ji}\alpha_{ij}^2.\end{aligned} \tag{10}$$

同样对式 (9) 括号内右边这项, 利用假设 2~4, 有

$$\begin{aligned}\sum_{(z,y)\in Z_j} L(h(z),y) &= \sum_{(z,y)\in Z_j} (\alpha_{ij}L(h_i(z),y) + \alpha_{jj}L(h_j(z),y))\\ &= \sum_{(z,y)\in Z_j} (\alpha_{ij}L(\mathbf{OPT}_i(z),y) + \alpha_{jj}L(h_j(z),y))\\ &\leqslant \alpha_{ij}\sum_{(z,y)\in Z_j} L(\mathbf{OPT}_j(z),y) - n_jk_{ij}\alpha_{ij}\\ &\quad + (1-\alpha_{ij})\sum_{(z,y)\in Z_j} L(\mathbf{OPT}_j(z),y) + n_jk_{jj}\alpha_{ij}(1-\alpha_{ij})\\ &= \sum_{(z,y)\in Z_j} L(\mathbf{OPT}_j(z),y) + n_j(-k_{ij}\alpha_{ij} + k_{jj}\alpha_{ij}(1-\alpha_{ij})).\end{aligned} \tag{11}$$

结合式 (10) 和 (11), 有

$$\frac{1}{|Z|}\sum_{(z,y)\in Z} L(h(z),y) \leqslant \overline{\mathbf{err}}(\mathbf{OPT}) + \frac{n_j}{n_i+n_j}((k_{ji}-k_{jj})\alpha_{ij}^2 - (k_{ij}-k_{jj})\alpha_{ij}). \tag{12}$$

在 $k_{jj} < k_{ij}$, $k_{jj} < k_{ji}$ 的情况下, 有 $\overline{\mathbf{err}}(h) < \overline{\mathbf{err}}(\mathbf{OPT})$.

关于定理 1 成立的条件, $k_{jj} < k_{ij}$, $k_{jj} < k_{ji}$ 为与假设 4 有关的条件. 具体要求模型的 h_j 部分在 Z_j 上的经验误差与 $\mathbf{OPT}_j$ 足够小. 由于本文已经假设 $n_i \gg n_j$ 以及 $|Z_j|$ 很小, 这一条件本身容易满足. 而实际上定理 1 成立的最主要的条件是 $k_{ij} > 0$, 也就是条件 (5) 的成立. 这一条件的成立要求子分布 i 与 j 之间出现频率有一定的差距, 以及两个子分布之间有一定的相似度. 为了保证这一条件在实际情况下有可能成立, 本文在下文第 4.2 节中通过实验讨论了该条件成立的条件.

3.2　任务辨识不足的进一步拓展分析

上一节的定理 1 限制在了 $N = 2$ 的情况, 这一节会将定理 1 推广到 $N > 2$ 的情况.

这一节沿用假设 3, 认为 $h(i \mid z)$ 是只与 i 以及 z 所属的子分布 j 有关的常数, 记作 α_{ij}. 为了讨论非 0 的 α_{ij} 对模型在训练集上的错误率 $\overline{\mathbf{err}}(h)$ 的影响, 作为假设 4 的推广, 有必要做出以下几个假设.

首先作为假设 4 中式 (7) 的推广, 本文做出下面这一假设.

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

假设5　存在一组 $k_{ij}, k_{ij} \in \mathbb{R}$, 使得 $\overline{\mathbf{err}}_j(\mathbf{OPT}_i) = \overline{\mathbf{err}}_j(\mathbf{OPT}_j) - k_{ij}$.

假设 5 与式 (7) 的差别除了允许 $N > 2$ 之外, 假设 5 中 k_{ij} 在 $\mathbb{R}$ 上取值, 也就是说并不是所有的 k_{ij} 都是大于 0 的. 不过为了结论的成立, 需要存在一些 $k_{ij} > 0$.

理论上, α_{ij} 越大, 意味着越多来自子分布 j 的数据被认为来自子分布 i, $\overline{\mathbf{err}}_j(h_j)$ 也会随之增加. 但是, 这一变化也与子分布 i 与 j 之间的相似度有关, 子分布 i 与 j 越相似, 这一增加幅度越小. 尽管可以用全变差距离定义两个子分布之间的相似度, 但是在这种定义下 $\overline{\mathbf{err}}_j(h_j)$ 关于此相似度的表达式十分复杂. 因此, 本文直接假设子分布之间存在相似度, 并基于相似度与 α_{ij} 对 $\overline{\mathbf{err}}_j(h_j)$ 与 $\overline{\mathbf{err}}_j(\mathbf{OPT}_j)$ 之间的差做出如下假设.

定义3　子分布 i 和 j 之间的相似度记为 s_{ij}. 并且有 $s_{ij} = s_{ji}$.

假设6　存在参数 $k_1 \geqslant 0$, 使得 $\overline{\mathbf{err}}_j(h_j) \leqslant \overline{\mathbf{err}}_j(\mathbf{OPT}_j) + k_1 \sum_{i \neq j}(1 - s_{ij})\alpha_{ij}$.

最后为了理论讨论的方便, 本文对 h_i 在 Z_j 上的经验误差受任务辨识能力的影响做出如下假设.

假设7　存在参数 $k_2 \geqslant 0$, 使得 $\frac{1}{n_j}\sum_{(z,y)\in Z_j} L(h_i(z), \mathbf{OPT}_i(z)) \leqslant k_2\alpha_{ii}$.

利用假设 3 和 5~7, 可以得到以下定理.

定理2　若假设 5~7 成立且对应的参数 n_j, k_{ij}, k_1, k_2 满足特定条件, 存在一组不全为 0 的 α_{ij} 使得对应的模型 h 满足 $\overline{\mathbf{err}}(h) < \overline{\mathbf{err}}(\mathbf{OPT})$.

证明　首先将 $\overline{\mathbf{err}}(h)$ 用 $\overline{\mathbf{err}}_j(h)$ 表示, 有 $\overline{\mathbf{err}}(h) = \sum_j \frac{n_j}{n}\overline{\mathbf{err}}_j(h)$. 利用假设 2, 3, 和 6, 有

$$\overline{\mathbf{err}}_j(h) = \alpha_{jj}\overline{\mathbf{err}}_j(h_j) + \sum_{i\neq j}\alpha_{ij}\overline{\mathbf{err}}_j(h_i) \leqslant \alpha_{jj}\left(\overline{\mathbf{err}}_j(\mathbf{OPT}_j) + k_1\sum_{i\neq j}(1-s_{ij})\alpha_{ij}\right) + \sum_{i\neq j}\alpha_{ij}\overline{\mathbf{err}}_j(h_i). \tag{13}$$

注意到 L 作为损失函数的性质, 利用假设 7, 有

$$\begin{aligned}\frac{1}{n_j}\sum_{(z,y)\in Z_j} L(h_i(z), y) &\leqslant \frac{1}{n_j}\sum_{(z,y)\in Z_j} L(h_i(z), \mathbf{OPT}_i(z)) + \frac{1}{n_j}\sum_{(z,y)\in Z_j} L(\mathbf{OPT}_i(z), y)\\ &\leqslant \overline{\mathbf{err}}_j(\mathbf{OPT}_i) + k_2\alpha_{ii}.\end{aligned} \tag{14}$$

再利用假设 5, 可以得到 $\overline{\mathbf{err}}_j(h)$ 的如下上界:

$$\begin{aligned}\overline{\mathbf{err}}_j(h) &\leqslant \alpha_{jj}\left(\overline{\mathbf{err}}_j(\mathbf{OPT}_j) + k_1\sum_{i\neq j}(1-s_{ij})\alpha_{ij}\right) + \sum_{i\neq j}\alpha_{ij}(\overline{\mathbf{err}}_j(\mathbf{OPT}_j) - k_{ij} + k_2\alpha_{ii})\\ &= \overline{\mathbf{err}}_j(\mathbf{OPT}_j) + k_1\alpha_{jj}\sum_{i\neq j}(1-s_{ij})\alpha_{ij} + k_2\sum_{i\neq j}\alpha_{ij}\alpha_{ii} - \sum_{i\neq j}k_{ij}\alpha_{ij}.\end{aligned} \tag{15}$$

于是有

$$\overline{\mathbf{err}}(h) \leqslant \overline{\mathbf{err}}(\mathbf{OPT}) + \sum_j \frac{n_j}{n}\left(k_1\alpha_{jj}\sum_{i\neq j}(1-s_{ij})\alpha_{ij} + k_2\sum_{i\neq j}\alpha_{ij}\alpha_{ii} - \sum_{i\neq j}k_{ij}\alpha_{ij}\right). \tag{16}$$

在参数 n_j, k_{ij}, k_1, k_2 满足一定条件时, 存在一组满足假设 3 的 α_{ij}, 使得不等式 (16) 右边这项取负值. 为了保证使该项能够取负值的参数范围不是空集, 这里举一个特例. 考虑一组特殊的 $\alpha_{ij}, \alpha_{ii} = 1, \forall i \geqslant 3$. 这样的 α_{ij} 可以被单一变量 α_{12} 完全确定. 将其代入不等式 (16) 右边这项, 可以得到关于 α_{12} 的二次函数 $k_1n_2(1-s_{21})\alpha_{12}(1-\frac{n_2}{n_1}\alpha_{12}) + k_2n_2\alpha_{12}(1-\alpha_{12}) - k_{21}n_2\alpha_{12} + k_1n_2(1-s_{12})\alpha_{12}(1-\alpha_{12}) + k_2n_2\alpha_{12}(1-\frac{n_2}{n_1}\alpha_{12}) - k_{12}n_2\alpha_{12}$. 显然若参数满足一定的条件, 此式有 $[0,1]$ 之间的负值解.

定理 2 指出, 在某种条件下, 任务辨识能力较差的模型反而能够取得更好的经验损失. 同定理 1 一样, 定理 2 的成立更多取决于条件 (5) 的成立. 而条件 (5) 的成立及显著性又取决于子分布之间出现频率以及相似度的大小. 这样就得出推论, 模型在出现频率差距越大, 越相似的任务之间的辨识能力越倾向于不足.

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

3.3　任务辨识不足的改进方向分析

本节中将会对定理 2 的结果进行一些拓展讨论并提出一些改进 LLM 任务辨识能力的可能方法.

首先, 只需要利用假设 3 及假设 5, 可以得到以下结果:

$$\begin{aligned}\overline{\mathbf{err}}(h) &= \sum_j \frac{n_j}{n}\overline{\mathbf{err}}_j(h) = \frac{1}{n}\sum_j n_j\left(\sum_{i\neq j}\alpha_{ij}(\overline{\mathbf{err}}_i(h_i)-k_{ji})+\alpha_{jj}\overline{\mathbf{err}}_j(h_j)\right)\\ &= \frac{1}{n}\left(\sum_j\sum_{i\neq j} n_i\alpha_{ji}(\overline{\mathbf{err}}_i(h_i)-k_{ji})+\sum_j n_j\alpha_{jj}\overline{\mathbf{err}}_j(h_j)\right)\\ &= \frac{1}{n}\left(\sum_j\sum_{i\neq j} n_j\alpha_{ij}(\overline{\mathbf{err}}_j(h_j)-k_{ij})+\sum_j n_j\alpha_{jj}\overline{\mathbf{err}}_j(h_j)\right)\\ &= \frac{1}{n}\left(\sum_j\left(\sum_{i\neq j}\alpha_{ij}+\alpha_{jj}\right)(\overline{\mathbf{err}}_j(h_j)-k_{ij})-\sum_{i,j} n_j\alpha_{ij}k_{ij}\right)\\ &= \sum_j \frac{n_j}{n}\overline{\mathbf{err}}_j(h_j)-\sum_{i,j}\frac{n_j}{n}\alpha_{ij}k_{ij},\end{aligned} \tag{17}$$

其中 $\sum_j \frac{n_j}{n}\overline{\mathbf{err}}_j(h_j)$ 可以理解为一个能够正确分辨子分布, 但是在第 i 个子分布上是 h_i 而不是 $\mathbf{OPT}_i$ 的模型的经验损失. 于是 $-\sum_{i,j}\frac{n_j}{n}\alpha_{ij}k_{ij}$ 可以理解为模型因为不能正确分辨子分布而出现的多余的经验损失. 由于 k_{ij} 的取值在 $\mathbb{R}$ 上, 在 k_{ij} 不一定满足定理 2 的条件的一般情况下, $-\sum_{i,j}\frac{n_j}{n}\alpha_{ij}k_{ij}$ 这一项并不一定小于 0.

注意到式 (17) 是等式, 与式 (16) 结合可以得到

$$\sum_j \frac{n_j}{n}\overline{\mathbf{err}}_j(h_j) \leqslant \overline{\mathbf{err}}(\mathbf{OPT})+\sum_j\frac{n_j}{n}\left(k_1\alpha_{jj}\sum_{i\neq j}(1-s_{ij})\alpha_{ij}+k_2\sum_{i\neq j}\alpha_{ij}\alpha_{ii}\right). \tag{18}$$

而式 (18) 右边这项一定是非负的, 可以理解为预训练时任务辨识能力的局限导致即使训练后的模型能够完美分辨子分布, 其经验损失仍然无法达到最优.

从式 (17) 和 (18) 可以得知, 要使定理 2 成立, 假设 5 是最关键的, 因为其提供了式 (17) 中的负值项. 而假设 5 依赖于条件 (5), 并且条件 (5) 中不等式两边之差的大小决定了假设 5 中的参数 k_{ij}, 从而决定了定理 2 中不等式的显著程度.

因此, 对条件 (5) 的改进是提升大语言模型任务辨识性能的重点. 本文认为, 增加预训练数据集中来自长尾分布尾端的子分布的占比, 是改进条件 (5) 的一个可能方向. 例如, 最近的一些研究尝试控制大语言模型预训练数据集中不同来源数据 (如社交网络、书籍、Wikipedia 等) 在每一个训练所用的小批量中的比例, 以此来提升预训练模型的性能 [37,38]. 更进一步地, Wettig 等 [39] 的研究将语料的分类进一步细化, 并控制细分后的每一类语料在小批量中的比例. 他们的研究发现, 通过提升不同领域语料数据所占的比例, 可以提升预训练模型在不同领域的性能.

另一方面, 对损失函数的改进是改进条件 (5) 的另一个可能方向. 依照本文提出假设 2 时的讨论, 给定模型 h 定义的 (z,y) 上的分布 M', 可以在一定条件下将其唯一分解为 $M'=\sum c_i'M_i'$. 如果有一种高效计算 c_i' 的方法, 就可以通过将 c_i' 与经验得到的 c_i 的差距加入损失函数的计算, 改进条件 (5), 进而提升模型的任务辨识能力.

此外, 对于混合专家模型 [21,22] 而言, 模型的任务辨识基本由门控网络决定, 因此增加门控网络的复杂度可以提升模型的任务辨识能力.

4　实验分析

4.1　估计模型任务辨识能力

本节通过实验来检验最近的大语言模型任务辨识能力. 然而, 大部分的大语言模型的输出只有对下一位字

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

符块的预测, 并不会显式地推断提示所属的子分布. 这使得大语言模型任务辨识能力的检验存在一定的困难.

利用前面的理论假设, 本文可以构造出一个粗略估计模型任务辨识能力的方法. 本文在这里进一步假设对于 $\forall(z,y) \in M_j$, $\{h(z) = y\} \supset \{h_j(z) = y\}$, 即模型只有正确分辨了子分布才能输出正确的结果, 则有对于 $\forall(z,y) \in M$, $\Pr(h(z) = y) = \sum_i h(i \mid z)\Pr(h_i(z) = y) = h(j \mid z)\Pr(h_j(z) = y)$, 也就是 $h(j \mid z) = \Pr(h(z) = y)/\Pr(h_j(z) = y)$, 模型 h 针对 (z,y) 的任务辨识能力可以由 $\Pr(h(z) = y)$ 与 $\Pr(h_j(z) = y)$ 的比值计算出. 其中, $\Pr(h(z) = y)$ 可以由模型的输出直接得到, 而 $\Pr(h_j(z) = y)$ 则需要估计. 本文受之前研究 [18,19] 的启发, 使用如下方法估计. 首先取 5 个与测试数据 (z,y) 同分布的样本 (z_i,y_i), $i = 1,\ldots,5$, 将这些样本的标签随机化为 y_i^{rnd}, $i = 1,\ldots,5$. 接着用处理过的样本 (z_i,y_i^{rnd}) 构造上下文提示 $\text{prompt}_{\text{icl}}$. 由于这样构造的上下文提示中的样本较少, 并且样本标签是随机化的, 理论上这些样例不会通过上下文学习的作用提升模型的性能, 因此这样构造出的上下文提示只会提升模型任务辨识的能力. 本文记使用上下文提示 $\text{prompt}_{\text{icl}}$ 的情况下的模型输出为 $h_j^{\text{icl}}(z)$, 则有 $\Pr(h_j(z) = y) \gtrapprox \Pr(h_z^{\text{icl}}(z))$. 于是有如下估计:

$$h(j \mid z) \lessapprox \frac{\Pr(h(z) = y)}{\Pr(h_z^{\text{icl}}(z))}. \tag{19}$$

在假设 3 的第一部分成立的条件下, $h(i \mid z), z \sim M_j$ 是只与 i,j 有关的常值. 于是对 $\Pr(h(z) = y) = h(j \mid z)\Pr(h_j(z) = y)$ 等式两边取期望得到 $h(j \mid z) = \mathbf{E}_{(z,y)\in M_j}[h(z) = y]/\mathbf{E}_{(z,y)\in M_j}[h_j(z) = y]$. 对于服从子分布 M_j 的测试数据集 Z, 可以用 Z 上的均值近似上式中的期望. 类似式 (19), 有

$$h(j \mid z) \lessapprox \frac{\frac{1}{|Z|}\sum_{(z,y)\in Z}\mathbf{1}(h(z) = y)}{\frac{1}{|Z|}\sum_{(z,y)\in Z}\mathbf{1}(h_j^{icl}(z) = y)}. \tag{20}$$

以下的实验将使用式 (20), 通过模型在零样本提示和少样本随机标签提示下的正确率的差距来估计任一模型 h 的任务辨识能力. 并且将使用式 (19), 通过两种提示下模型给出的正确概率的比值来估计 $h(j \mid z, z \sim M_j)$, 进一步验证假设 3 的正确性.

4.1.1　在分类及多项选择测试数据集上进行估计

首先要指出, 对于所有的测试数据集, 本文实验中使用的都是明确说明提示, 因为本文所关注的大语言模型任务辨识能力的不足在明确说明提示这种理论上能够完美分辨任务的情况下更容易讨论. 其次, 在随机化标签时, 本文首先选择一组尽可能不重复的随机标签, 这组标签的数量与少样本提示中样例数量一致, 再采用不放回取样的方法将它们赋予提示中的样例, 以此来减少样本中重复标签对模型性能造成的影响. 最后, 使用式 (20) 的方法有一定的局限性. 构造随机标签少样本提示的做法基本上只适用于分类和多项选择等少数类型的测试数据集. 对于更一般的数据集, 虽然也可以构造上下文提示, 但是没有很好的办法将样本随机化来避免上下文学习能力的影响.

本文在 2 个最近的大语言模型, Mistral Nemo[1)] 和 Llama 3.1 8B[2)], 和 12 个分类和多项选择测试数据集上分别使用零样本 (0-shot)、随机标签少样本 (few-shot random), 和真实标签少样本 (few-shot golden) 的提示进行评估. 实验结果在表 1 中展示, 每组实验中的最低值用粗体表示.

如表 1 所示, 实验所用的两个大语言模型在不同的测试数据集上的任务辨识能力各不相同. 从一个模型在某一测试数据集上的任务辨识能力更好并不能推断出该模型在其他测试数据集上的任务辨识能力更好. 而在大多数的情况下, 零样本提示的正确率都显著地小于随机标签少样本提示的正确率, 同时随机标签少样本提示的正确率接近真实标签少样本提示的正确率. 这一结果表明即使是最新结构的大语言模型依然存在着任务辨识能力不足的问题.

观察具体的模型输出, 可发现模型在零样本提示下的输出有很强的偏见. 例如, 本文发现该模型在 TweetEval sentiment 测试集上偏向于回答 True, 而在 glue-mrpc 测试集上偏向于回答 False. 两个具体的例子见表 2. 本

1) https://mistral.ai/news/mistral-nemo/.
2) https://ai.meta.com/blog/meta-llama-3-1/.

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

表 1　两个模型在不同测试数据集上使用 3 种不同提示风格的准确率 (%).

Table 1　Accuracies (%) of the models on various datasets using the three different prompting styles.

Dataset	Mistral Nemo			Llama 3.1 8B		
	0-shot	few-shot random	few-shot golden	0-shot	few-shot random	few-shot golden
MMLU	**50.85**	60.27	60.37	**54.03**	57.51	58.13
arc-challenge	**76.26**	84.61	85.04	**71.74**	77.30	77.39
openbookQA	**72.20**	79.60	79.00	**64.20**	74.80	75.00
TweetEval sentiment	67.15	**62.79**	66.03	**54.81**	60.71	65.15
TweetEval offensive	**31.67**	52.33	71.55	**50.09**	51.69	54.79
race	**65.91**	82.73	83.73	**74.61**	78.23	79.08
cmnli	**34.59**	46.77	56.85	**34.69**	48.07	51.17
glue-mrpc	**33.51**	68.81	70.09	**46.61**	71.19	73.74
glue-rte	**50.54**	79.42	79.42	**68.95**	77.26	79.42
glue-qnli	**53.63**	67.25	70.43	**57.79**	66.97	58.05
glue-qqp	**64.25**	81.66	81.62	**63.31**	75.51	73.85
glue-cola	**33.37**	46.41	72.87	**66.92**	78.81	78.90

表 2　模型在零样本提示下的输出样例. 模型的输出用粗体表示.

Table 2　Examples of model outputs employing 0-shot prompts. Bold texts denote the model's output.

Example output on TweetEval	Example output on glue-mrpc
Is the following Tweet offensive? Your answer should include only True or False. ### Tweet: #SilsilaBadallteRishtonKa tag is filled with appreciation of @user only.... Never watched this serial but I always wanted appreciation for Aditi... She is underrated since starting of the show... She is worth of all this appreciation.. [Clapping Hands][Clapping Hands][Clapping Hands] ### Answer: The correct answer is **True.** **### Explanation:** **The Tweet is offensive because it is a personal attack on Aditi.**	Below are two sentences. Judge whether they are equivalent. Your answer should only be True or False. ### Sentence1: According to the federal Centers for Disease Control and Prevention (news - web sites) , there were 19 reported cases of measles in the United States in 2002 . ### Sentence2: The Centers for Disease Control and Prevention said there were 19 reported cases of measles in the United States in 2002 . ### Answer: The correct answer is **False.** **### Explanation:** **The first sentence is a news report, while the second sentence is a quote. The first sentence is written in the past tense, while the second sentence is written in the present tense.**

文认为, 这种偏见体现了模型在这些数据集上认为应以大概率输出某个标签, 从而很难正确分辨提示所属的子分布.

此外, 由于这一实验中随机标签少样本提示的标签是完全随机的, 实验结果可能受随机性以及样例标签中标签真伪的比例的影响. 因此本文多次选取随机标签进行重复实验, 来探究随机性对表 1 中随机标签下模型表现的影响. 实验中, 本文分别采用了两种随机化标签的方法. 第一种方法如上文中所述, 首先选取尽可能不重复的一组标签, 这组标签的数量与少样本提示中样例数量一致, 然后从这组标签中不放回取样, 为每个少样本样例分配标签. 第二种方法为了确保少样本样例中真伪标签的比例平衡, 首先随机选取少样本样例中合适的一部分, 这部分样例保持真实标签. 例如在 10 个样例的少样本提示中, 若多项选择的选项有 4 个, 则选择 2 或 3 个样例保持真实标签. 而剩下的样例则随机分配一个非真实的标签. 例如若真实标签为 “A”, 选项为 “A”～“D”, 则随机

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

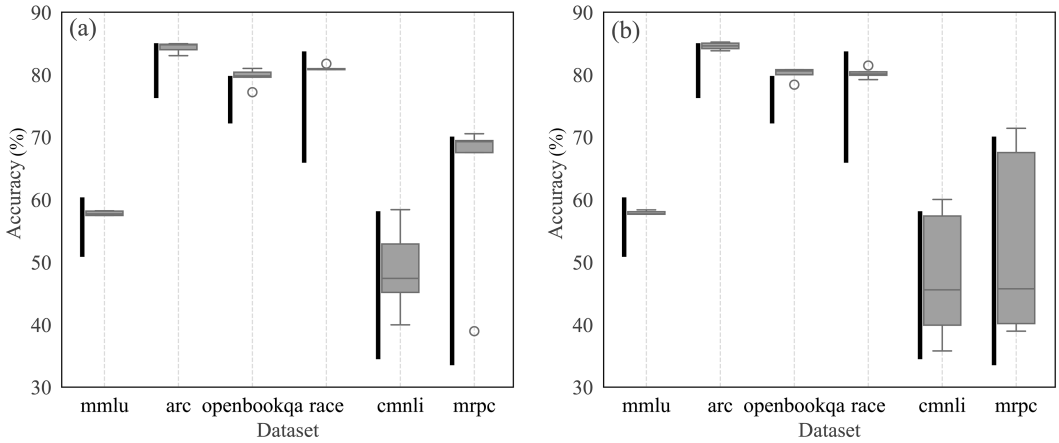

图 3　部分数据集上 Mistral Nemo 模型在随机标签少样本提示下准确率的箱型图, 其中的竖线表示从零样本提示下的准确率到少样本真实标签提示下的准确率的范围. (a) 用方法一随机化标签; (b) 用方法二随机化标签.

Figure 3　Boxplot of the scores of the Mistral Nemo model on some test datasets using few-shot random prompts, where the vertical lines range from the model's scores using 0-shot prompts to scores using few-shot golden prompts. (a) Using label randomization method 1; (b) using label randomization method 2.

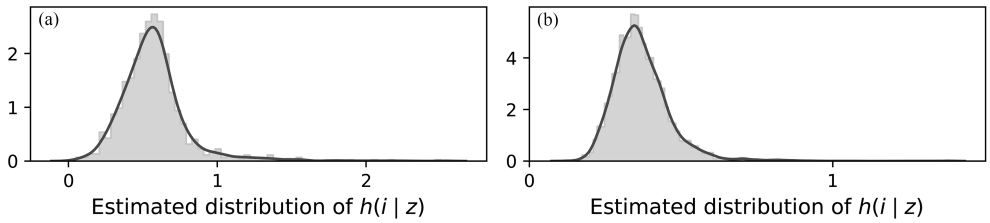

图 4　模型在某一测试数据集上 $h(i \mid z, z \sim M_j)$ 经验分布估计的直方图以及核密度估计得到的分布密度. (a) Llama 3.1 8B 模型在 arc 数据集上的结果; (b) Mistral Nemo 模型在 TweetEval-offensive 数据集上的结果.

Figure 4　The histogram and probability density function by kernel density estimation of $h(i \mid z, z \sim M_j)$. (a) The result of the Llama 3.1 8B model on the arc dataset; (b) the result of the Mistral Nemo model on the TweetEval-offensive dataset.

分配 "B", "C", "D" 中的任一个标签.

图 3 展示了 Mistral Nemo 模型在部分测试数据集上重复实验得到的少样本随机标签提示下的准确率的箱型图. 从图中可以看出, 虽然随机标签少样本提示下模型的性能受到标签的随机性的影响, 但是这种随机性并不影响上述的实验结论.

4.1.2　验证任务辨识能力的一致性假设

在假设 3 下面的讨论中, 本文提出 $h(i \mid z, z \sim M_j)$ 应该为只与 i 和 j 有关的常数, 或者其关于 z 的分布应该以大概率集中在 $(0,1)$ 间某个实数附近. 而式 (19) 给出了一个粗略估计 $h(i \mid z, z \sim M_j)$ 的方法. 本文对上面实验中用到的提示使用这一估计方法, 对任一测试数据集 S 中的提示 $z \in S$ 估计 $h(i \mid z, z \in S)$, 并记录其经验分布. 图 4 展示了部分实验结果的直方图以及用核密度估计平滑化得到的分布密度函数.

如图 4 所示, 估计得到的 $h(i \mid z, z \sim M_j)$ 的分布集中在 $(0,1)$ 之间的某个实数附近. 这一结果表明, 本文选择的模型在同一测试数据集中的提示上的任务辨识能力大致集中在某一常数附近, 这在一定程度上验证了本文的假设 3.

4.1.3　模型参数大小的影响

实验结果表明, 参数规模较小的语言模型与参数规模较大的语言模型在上下文学习中表现出不同的模式. 参数规模较小的语言模型对上下文提示的利用更加侧重任务辨识, 而参数规模较大的语言模型更加侧重任务学

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

表 3　Llama 3.1 70B 在不同测试数据集上使用 3 种不同提示风格的准确率 (%).

Table 3　Accuracies (%) of Llama 3.1 70B on various datasets using the three different prompting styles.

Dataset	0-shot	few-shot random	few-shot golden	Dataset	0-shot	few-shot random	few-shot golden
arc-challenge	**90.81**	91.91	92.21	MMLU	74.63	69.73	**69.13**
openbookQA	**87.60**	90.40	90.60	race	87.61	**82.32**	86.01
TweetEval sentiment	**48.55**	61.44	67.13	glue-rte	84.12	80.87	**75.45**
TweetEval offensive	**34.07**	78.62	79.02	glue-cola	82.52	**67.53**	79.32
cmnli	**51.45**	52.45	59.44	glue-qnli	75.23	**54.45**	85.51
glue-mrpc	**58.54**	74.53	75.82				
glue-qqp	**80.12**	85.71	86.51				

习 [19]. 因此使用参数规模更大的模型重复上述实验是有必要的. 本文选择了 Llama 3.1 70B 模型重复上述实验, 并将其结果与表 1 中的结果进行比较.

实验结果在表 3 中展示, 其中每组实验的最低值用粗体表示. 如表 3 左半部分所示, 在部分测试数据集上, Llama 3.1 70B 表现出与较小规模语言模型相似的结果, 零样本提示的正确率小于随机标签少样本提示的正确率. 与较小规模语言模型上的实验结果不同的是, Llama 3.1 70B 在零样本提示下和在随机标签少样本提示下的正确率的差距有所减小, 即 Llama 3.1 70B 的任务辨识能力相对更小规模语言模型更加优秀. 这可能源于更大参数规模带来的模型能力的提升, 以及更大的训练集带来的理论假设 3 的拓宽.

但是, 如表 3 的右半部分所示, 在另一部分测试数据集上, Llama 3.1 70B 表现出与较小规模语言模型不同的结果, 随机标签少样本提示下的正确率显著地低于零样本提示和真实标签少样本提示下的正确率. 本文认为这一部分表现出 Llama 3.1 70B 在上下文学习中更加侧重任务学习的特点. 由于模型在很少样本的上下文提示下仍然进行了任务学习, 不满足式 (20) 的应用条件, 因此这些实验结果无法衡量该模型的任务辨识能力. 具体地, 由于模型在随机标签样本或真实样本上的任务学习对模型在该数据集上的能力产生了负面作用, 因此产生了表中所示的实验结果.

4.1.4　微调的影响

在实际应用中, 上下文学习与微调一般被视为可以互相替代的两种实践. 然而在本文的理论框架中, 这两者有着不同的机制. 上下文学习通过提升模型的任务辨识能力来提升模型的表现, 而微调通过作用于模型的训练集来改变假设 4 及假设 5 中的参数, 进而改变模型的表现. 因此即使随机标签的上下文提示可以提升模型在测试数据集上的表现, 在随机标签的数据集上进行微调并不一定能够提升模型的表现. 具体地, 尽管进行微调能够提升模型的任务辨识能力 $h(i \mid z, z \sim M_i)$, 但是同时模型在该任务上的表现 $h(z \mid i)$ 也会受到影响, 无法做到像上面的实验一样通过尽可能减少上下文提示中的样本将任务学习的影响降到最低. 因此可以推断, 不同于使用随机标签样本的上下文提示, 在随机标签的数据集上进行微调并不会显著提升模型的表现.

严谨起见, 本文使用随机标签及真实标签的数据集微调模型, 并重复第 4.1.1 节的实验. 实验结果如表 4 所示, 其中每组实验的最低值用粗体表示. 实验结果表明, 模型在随机标签的数据集上微调后性能呈现出不规律的上升或下降, 在正确标签的数据集上微调后性能一致上升, 这一现象与随机标签上下文提示下的结果完全不同.

4.2　验证关键理论假设

本文的两个理论结论, 定理 1 与 2, 都十分依赖于条件 (5) 的成立. 本节通过实验来说明条件 (5) 的现实合理性, 并探索条件 (5) 成立的条件.

4.2.1　使用合成任务验证

本节的实验通过在人工合成的数据上预训练大语言模型对比了一般模型与 3.1 节中定义的 **OPT** 的性能差

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

表 4　两个模型在随机标签及真实标签数据集上微调前后的准确率 (%).

Table 4　Accuracies (%) of the models before and after finetuning on random label and golden label datasets.

Dataset	Mistral Nemo			Llama 3.1 8B		
	Base model	Finetune with random label	Finetune with golden label	Base model	Finetune with random label	Finetune with golden label
MMLU	50.85	**35.71**	56.11	54.03	**37.11**	55.47
arc-challenge	76.26	**46.17**	84.78	71.74	**39.91**	77.83
openbookQA	72.20	**25.20**	81.80	64.20	**33.00**	77.80
TweetEval sentiment	67.15	**51.45**	66.85	54.81	**20.48**	63.85
TweetEval offensive	**31.67**	54.67	68.49	**50.09**	53.67	70.19
race	65.91	**27.50**	82.49	74.61	**30.54**	79.23
cmnli	**34.59**	35.21	67.92	34.69	36.23	**33.51**
glue-mrpc	**33.51**	45.91	66.49	**46.61**	50.49	66.49
glue-rte	**50.54**	57.76	87.37	68.95	**55.96**	80.14
glue-qnli	53.63	**50.47**	87.88	57.79	**57.43**	86.72
glue-qqp	**64.25**	67.81	84.88	63.31	**62.45**	83.68
glue-cola	**33.37**	53.69	85.24	66.92	**47.27**	77.76

异, 进而验证了条件 (5). 这里使用的人工合成数据的生成过程参考了之前的工作 [27,40]. 首先用 Zipf 分布确定 N 个子分布所占的比例, $c_i \propto \frac{1}{i+2.7}$. 其次对每个子分布选择一个长为 d 的 0-1 字符串, 第 i 个子分布对应的字符串记为 s_i, 并定义任意两个子分布之间的相似度为 $s_{i,j} = \sum_{k=1}^{d} \mathbf{1}(s_i(k) = s_j(k))$. 第 i 个子分布中的样本为以 σ 的概率随机翻转 s_i 的每一位得到的. 这种合成数据分布能够模拟自然语言数据分布的基础在于, 给定输入的提示, 大语言模型可以视为全部可能输出上的一个分布. 若输入本身也服从某种先验分布, 则大语言模型完全等价于所有可能字符串上的一个分布. 又由于任意字符串都可以用 0-1 字符串表示, 即大语言模型等价于全部可能 0-1 字符串上的一个分布. 而上述合成分布的构造实际上假设了每一个子分布都是退化的.

本文在这个合成数据分布上取样得到一个数据集, 并在这一由 0-1 字符串构成的数据集上分别训练两个小型的 GPT-2 模型 [33]. 第一个模型, 记作模型 1, 在该数据集上以下一位预测为目标训练. 第二个模型, 记作模型 2, 在训练参数上与第一个模型一致. 两者的差别在于, 训练第二个模型的数据集中的每个样本都在字符串的开头加上了用来识别子分布的长度为 L 的 0-1 字符串, 其与样本所属的子分布一一对应且不添加噪声. 并且在训练以及评估的过程中, 每个样本的前 L 位并不参与损失函数的计算. 这样一来, 模型 2 在训练以及推断时可以完美地判断提示所属的子分布, 可以被认为是 **OPT**. 图 5 展示了合成分布的生成、训练数据的生成, 以及训练及评估目标.

直觉上, 模型 2 在所有子分布上的错误率都会小于模型 1, 因为模型 2 在训练和推断时可以完美地分辨子分布. 然而若条件 (5) 对子分布 i 和 j 成立, 则对子分布 j 有相反的事实. 在实验中本文关注第 1 个子分布与第 N 个子分布, 即占比最大和最小的子分布, 并控制参数观察两个模型在其上的测试错误率. 本文对每个参数设置都重复 7 次实验, 来避免字符串 s_i 的随机性对实验结果产生影响. 实验结果见图 6.

从图 6(b) 和 (c) 中可以看出, 如果噪声程度太小或者两个子分布之间出现频率的差距不够大, 模型 2 在第 N 个子分布上的错误率都会小于模型 1. 也就是说足够的噪声以及两个子分布出现频率之间足够的差距是条件 (5) 能够成立的必要条件. 从图 6(a)~(d) 中可以看出, 两个子分布之间的相似度增加、噪声程度在一定范围内的增加、两个子分布出现频率差距的增加, 以及上下文长度的增加都使得错误率的比值增加. 从定理 1 和 2 的证明中可知, 式 (5) 越显著, 两个定理的结论也会越显著. 因此在上述 4 种情况下, 训练得到的模型 h 的任务辨识能力会更差. 此外, 由于子分布 i 在合成分布中的占比正比于 $\frac{1}{i+2.7}$, 因此减少 N 意味着增加第 N 个子分布在预训练数据中的占比. 因此从图 6(c) 中能够得到的另一个推论是, 增加预训练数据中来自长尾分布尾端的子分

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

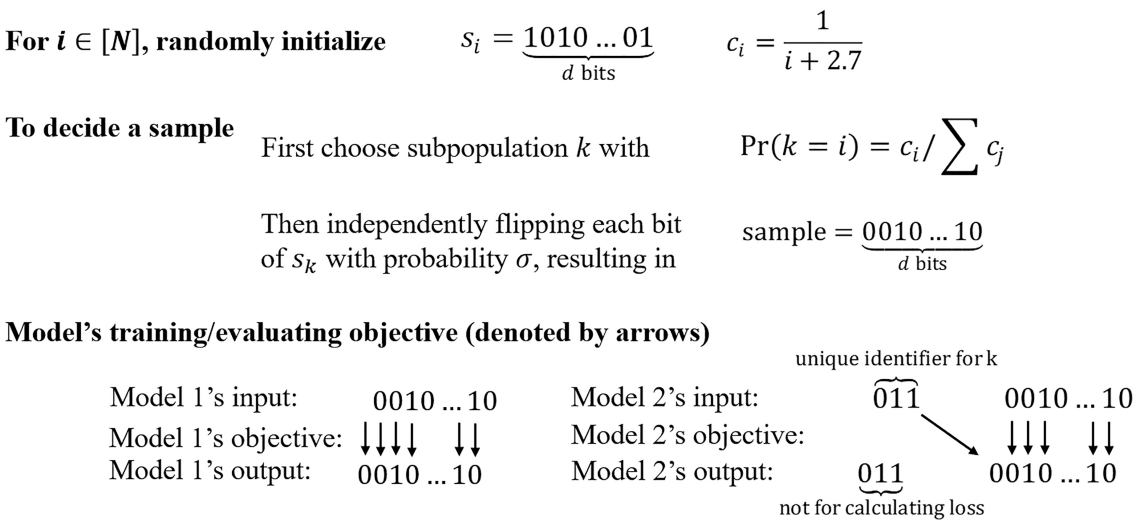

图 5　合成任务中的数据分布生成、取样、训练/评估目标的示例.

Figure 5　An example of synthetic data distribution generation, data sampling, and training/evaluating objective.

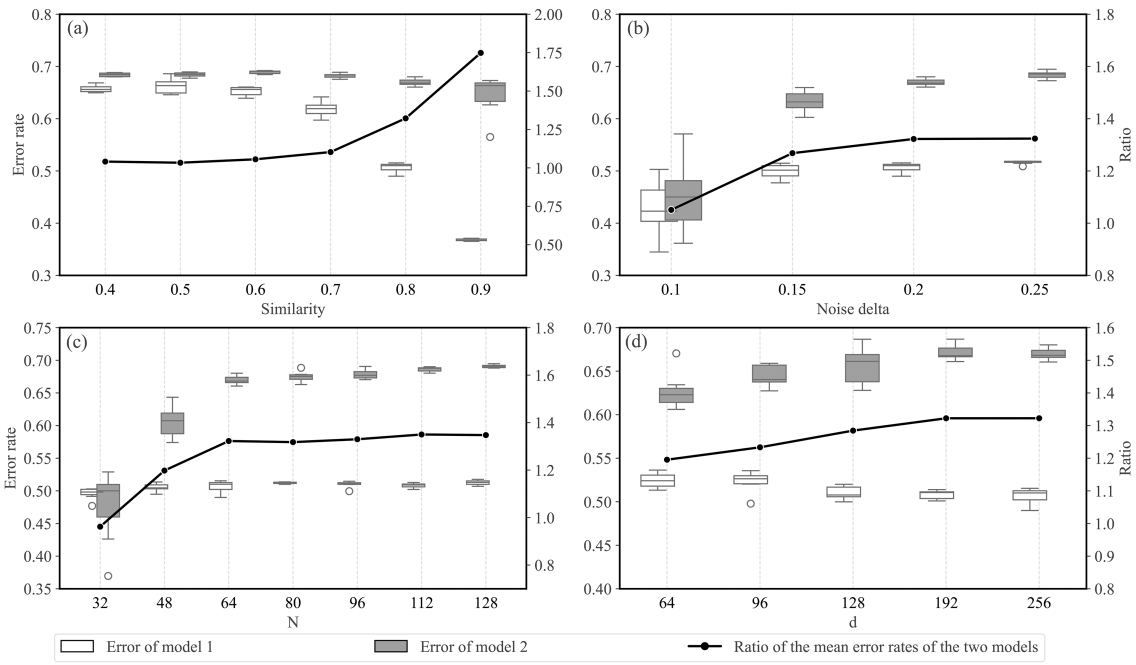

图 6　模型 1 与模型 2 在第 N 个子分布上的错误率的箱形图, 折线为模型 2 与模型 1 错误率的比值. (a) x- 轴为不同的 $s_{1,N}$, 其他参数分别为 $d=256$, $N=64$, $\delta=0.2$; (b) x- 轴为不同的噪声 δ, 其他参数分别为 $d=256$, $N=64$, $s_{1,N}=0.8$; (c) x- 轴为不同的子分布数量 N, 其他参数分别为 $d=256$, $\delta=0.2$, $s_{1,N}=0.8$; (d) x- 轴为不同的上下文长度 d, 其他参数分别为 $N=64$, $\delta=0.2$, $s_{1,N}=0.9$.

Figure 6　Boxplots of the two models' error rates on the Nth subpopulation. The line refers to their ratio. (a) Error rates against different $s_{1,N}$. The other parameters are $d=256$, $N=64$, $\delta=0.2$. (b) Error rates against noise level δ. The other parameters are $d=256$, $N=64$, $s_{1,N}=0.8$. (c) Error rates against number of subpopulations N. The other parameters are $d=256$, $\delta=0.2$, $s_{1,N}=0.8$. (d) Error rates against context length d. The other parameters are $N=64$, $\delta=0.2$, $s_{1,N}=0.9$.

布的占比可以改进条件 (5), 进而提升模型的任务辨识能力, 即本文在第 3.3 节中提出的任务辨识不足的改进方向之一在合成任务的实验上得到了验证.

最后本文需要说明的是, 在图 6(a) 中可以看出两个子分布之间的相似度 s_{ij} 至少为 0.7 或 0.8 时, 比值才显

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

表 5　Minerva 7B 模型在若干西班牙语测试数据集上的评估. 若无特殊说明, 默认的指标为准确度, 基准为随机猜测的指标.

Table 5　The evaluation results of Minerva 7B on various Spanish benchmarks. The default metric for evaluation is accuracy unless otherwise specified, and the baseline is given by a random guess.

	arc	hellaswag	belebele	COPA-es	Global-MMLU	xquad (exact match)
Score (%)	27.26	35.89	37.11	60.80	33.75	31.09
Baseline (%)	25	25	25	50	25	N/A

著地大于 1, 这一 s_{ij} 并不过高. 首先, 在 0-1 字符串的前提下, 由于概率分布的支撑集是一个只有两个元素的集合 $\{0,1\}$, 即使是随机选取两个字符串 s_i 与 s_j, 它们之间的相似度也会以极大概率接近 1/2. 其次, 实际情况中子分布之间出现频率的差距远大于实验中的 $N=64$ 个子分布的情况. 因此, 在实际情况下式 (5) 的成立条件并不如此实验中一样严格.

4.2.2　使用真实任务验证

上一节中本文在合成生成的任务上训练了一个语言模型, 验证了条件 (5). 然而, 合成任务上的实验结果不一定能够代表真实情况. 本节使用真实大语言模型和真实任务, 通过一种间接的方式验证条件 (5).

在真实数据集上用真实模型验证条件 (5) 的最大障碍是, 为了验证条件 (5) 需要控制模型的预训练数据集, 而预训练大语言模型会消耗大量的时间和资源. 为了以一个较低的消耗验证条件 (5), 本文采用一个间接的方式. 注意到语言本身也是一种任务, 而西班牙语与意大利语是两个非常相似的语言, 有着接近 82% 的词汇相似度. 如果设某个西班牙语的测试数据集所属的任务为 M_i, 设意大利语所属的任务为 M_j, 在这两个任务上验证条件 (5) 就可以使用已有的预训练模型, 极大地降低实验产生的浪费. 本文选择了只用意大利语和英语预训练出的模型 Minerva 7B[3)], 在若干个原生西班牙语任务及从英语翻译为西班牙语的任务上评估模型的表现, 实验的结果见表 5, 其中, 前 5 个测试数据集为分类或多项选择任务, xquad 为生成式测试数据集.

如表 5 所示, Minerva 7B 模型在若干西班牙语的测试数据集上有着非平凡的表现, 尽管这个模型的训练数据集中并没有西班牙语数据. 这一实验结果意味着在 $|Z_i|=0$ 的情况下, 条件 (5) 中的不等号严格成立. 显然这也意味着对于足够小的 $|Z_i|$, 条件 (5) 仍然成立.

4.3　验证理论结果

为了在合成数据上验证理论结论, 这里对第 4.2.1 节的模型 1 稍作修改, 使其在训练和推断时不仅预测下一位字符块的概率分布, 还预测提示所属子分布的概率分布. 这一修改是通过增加模型的词汇容量实现的. 修改后的模型容量为 $2^{1+\lceil\log_2 N\rceil}$. 在训练时, 首先将数据集中的每个长为 d 的字符串分解为 $d-1$ 个提示与下一位字符块的组合. 接着保持提示为 0-1 字符串不变, 但是改变下一位字符块, 使其在二进制形式下最末尾为原本的 0 或 1, 而前 $\lceil\log_2 N\rceil$ 位为能够唯一识别这一组合所属的子分布的一串 0 或 1. 这样一来, 训练出的模型的输出也将保持这种格式, 对于作为提示输入的 0-1 字符串, 既预测下一位字符块在 $\{0,1\}$ 上的分布, 也推断提示所属子分布在 $[N]$ 上的分布.

这样的输出格式使得对模型预测下一位字符块的能力和推断提示所属子分布的能力可以分开计算. 在评估模型时, 先只考虑模型输出在预测下一位字符块上的损失, 再只考虑模型输出在推断提示所属子分布上的损失, 得到两个测试集上损失的值. 用这种评估方法, 以及上一节中控制子分布之间相似度的方法, 就可以具体地观察模型的任务辨识能力与各参数之间的关系.

用如上方法训练出的模型在第 N 个子分布上的评估结果如表 6 所示. 可以看出, 随着分布之间相似度的增加, 模型在第 N 个子分布上预测下一位的错误率减少, 但是在分辨提示所属的子分布上的错误率增加. 这一结果符合本文中的定理.

3) https://nlp.uniroma1.it/posts/34.

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

表 6　训练出的模型在预测下一位和分辨子分布上各自的测试损失.

Table 6　The model's losses in inferring next bit and identifying subpopulation, respectively.

	Similarity = 0.4	Similarity = 0.5	Similarity = 0.6	Similarity = 0.7	Similarity = 0.8	Similarity = 0.9
Loss in inferring next bit	0.68465	0.68514	0.68426	0.65207	0.54745	0.45864
Loss in identifying subpopulation	0.16968	0.18699	0.2031	0.26822	0.43541	0.95021

5　结论与展望

本文从随机标签的少样本提示能够提升大语言模型性能这一现象, 以及少样本提示作用于模型性能的两种机制出发, 得出了大语言模型在任务辨识上的能力不足这一推论. 接着通过理论推导证明了, 大语言模型由于自然语言的长尾分布特征和任务之间的相似度两个原因, 任务辨识能力更差的模型能够在训练集上达到更好的经验误差. 通过一系列的实验, 本文展示了当前最新的大语言模型在任务辨识上的不足, 分析了模型参数大小、微调等因素对任务辨识能力不足的影响, 验证了理论假设与理论结论. 由此, 本文给出了大语言模型在任务辨识上能力不足这一现象的一种可能解释.

本文的理论结果并不意味着提升大语言模型任务辨识能力的尝试没有意义, 而是指出在数据集和模型满足某些条件的情况下, 模型的任务辨识能力存在一个并非完美的上限. 首先, 对本文中定理所需条件的改进可以作为提升大语言模型任务辨识能力的尝试. 通过增加预训练数据中自然语言分布的长尾分布尾端的数据量, 改进训练所用的损失函数, 运用更加复杂的任务辨识架构, 可以期望模型的任务辨识能力得到改善. 其次, 关于当前的大语言模型的任务辨识能力是否达到了本文定理所陈述的限制尚不明确. 对于大语言模型本身结构的改进仍有价值.

参考文献

1 Che W X, Dou Z C, Feng Y S, et al. Towards a comprehensive understanding of the impact of large language models on natural language processing: challenges, opportunities and future directions. Sci Sin Inform, 2023, 53: 1645–1687 [车万翔, 窦志成, 冯岩松, 等. 大模型时代的自然语言处理: 挑战, 机遇与发展. 中国科学: 信息科学, 2023, 53: 1645–1687]

2 Shao Y F, Geng Z C, Liu Y T, et al. CPT: a pre-trained unbalanced transformer for both Chinese language understanding and generation. Sci China Inf Sci, 2024, 67: 152102

3 Jiang Z, Xu F F, Araki J, et al. How can we know what language models know? Trans Assoc Comput Linguistics, 2020, 8: 423–438

4 Turpin M, Michael J, Perez E, et al. Language models don't always say what they think: unfaithful explanations in chain-of-thought prompting. In: Proceedings of Advances in Neural Information Processing Systems, 2023. 74952–74965

5 Kojima T, Gu S S, Reid M, et al. Large language models are zero-shot reasoners. In: Proceedings of Advances in Neural Information Processing Systems, 2022. 22199–22213

6 Wei J, Wang X, Schuurmans D, et al. Chain-of-thought prompting elicits reasoning in large language models. In: Proceedings of Advances in Neural Information Processing Systems, 2022. 24824–24837

7 Liu P, Yuan W, Fu J, et al. Pre-train, prompt, and predict: a systematic survey of prompting methods in natural language processing. ACM Comput Surv, 2023, 55: 1–35

8 Brown T, Mann B, Ryder N, et al. Language models are few-shot learners. In: Proceedings of Advances in Neural Information Processing Systems, 2020. 1877–1901

9 Mueller A, Webson A, Petty J, et al. In-context learning generalizes, but not always robustly: the case of syntax. In: Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2024. 4761–4779

10 Pecher B, Srba I, Bielikova M. A survey on stability of learning with limited labelled data and its sensitivity to the effects of randomness. ACM Comput Surv, 2025, 57: 1–40

11 Liu J, Shen D, Zhang Y, et al. What makes good in-context examples for GPT-3? In: Proceedings of Deep Learning Inside Out: The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures, 2022. 100–114

12 Rubin O, Herzig J, Berant J. Learning to retrieve prompts for in-context learning. In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2022. 2655–2671

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

13 Lu Y, Bartolo M, Moore A, et al. Fantastically ordered prompts and where to find them: overcoming few-shot prompt order sensitivity. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022. 8086–8098
14 Wei J, Wei J, Tay Y, et al. Larger language models do in-context learning differently. 2023. ArXiv:2303.03846
15 Reif Y, Schwartz R. Beyond performance: quantifying and mitigating label bias in LLMs. In: Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2024. 6784–6798
16 Webson A, Pavlick E. Do prompt-based models really understand the meaning of their prompts? In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2022. 2300–2344
17 Han Z, Hao Y, Dong L, et al. Prototypical calibration for few-shot learning of language models. In: Proceedings of the 11th International Conference on Learning Representations, 2023
18 Min S, Lyu X, Holtzman A, et al. Rethinking the role of demonstrations: what makes in-context learning work? In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, 2022. 11048–11064
19 Pan J, Gao T, Chen H, et al. What in-context learning "learns" in-context: disentangling task recognition and task learning. In: Proceedings of Findings of the Association for Computational Linguistics, 2023. 8298–8319
20 Xie S M, Raghunathan A, Liang P, et al. An explanation of in-context learning as implicit Bayesian inference. In: Proceedings of International Conference on Learning Representations, 2022
21 Jiang A Q, Sablayrolles A, Roux A, et al. Mixtral of experts. 2024. ArXiv:2401.04088
22 Liu A, Feng B, Wang B, et al. Deepseek-v2: a strong, economical, and efficient mixture-of-experts language model. 2024. ArXiv:2405.04434
23 Cai W, Jiang J, Wang F, et al. A survey on mixture of experts in large language models. IEEE Trans Knowl Data Eng, 2025, 37: 3896–3915
24 Si C, Friedman D, Joshi N, et al. Measuring inductive biases of in-context learning with underspecified demonstrations. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, 2023. 11289–11310
25 McMilin E. Underspecification in language modeling tasks: a causality-informed study of gendered pronoun resolution. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2024. 18778–18788
26 Feldman V. Does learning require memorization? A short tale about a long tail. In: Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, 2020. 954–959
27 Brown G, Bun M, Feldman V, et al. When is memorization of irrelevant training data necessary for high-accuracy learning? In: Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing, 2021. 123–132
28 Zhu X, Anguelov D, Ramanan D. Capturing long-tail distributions of object subcategories. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014. 915–922
29 Van Horn G, Perona P. The devil is in the tails: fine-grained classification in the wild. 2017. ArXiv:1709.01450
30 Dai A M, Le Q V. Semi-supervised sequence learning. In: Proceedings of Advances in Neural Information Processing Systems, 2015
31 Yang Z, Dai Z, Yang Y, et al. Xlnet: generalized autoregressive pretraining for language understanding. In: Proceedings of Advances in Neural Information Processing Systems, 2019
32 Zoph B, Bello I, Kumar S, et al. St-moe: designing stable and transferable sparse expert models. 2022. ArXiv:2202.08906
33 Radford A, Wu J, Child R, et al. Language models are unsupervised multitask learners. OpenAI blog, 2019, 1: 9
34 Razeghi Y, Logan IV R L, Gardner M, et al. Impact of pretraining term frequencies on few-shot numerical reasoning. In: Proceedings of Findings of the Association for Computational Linguistics, 2022. 840–854
35 Kandpal N, Deng H, Roberts A, et al. Large language models struggle to learn long-tail knowledge. In: Proceedings of the 40th International Conference on Machine Learning, 2023. 15696–15707
36 Wang J, HU X, Hou W, et al. On the robustness of ChatGPT: an adversarial and out-of-distribution perspective. In: Proceedings of ICLR 2023 Workshop on Trustworthy and Reliable Large-Scale Machine Learning Models, 2023
37 Xie S M, Pham H, Dong X, et al. Doremi: optimizing data mixtures speeds up language model pretraining. In: Proceedings of the 37th Conference on Neural Information Processing Systems, 2023
38 Liu Q, Zheng X, Muennighoff N, et al. Regmix: data mixture as regression for language model pre-training. In: Proceedings of the 13th International Conference on Learning Representations, 2025
39 Wettig A, Lo K, Min S, et al. Organize the web: constructing domains enhances pre-training data curation. 2025. ArXiv:2502.10341
40 Kang M, Lee S, Baek J, et al. Knowledge-augmented reasoning distillation for small language models in knowledge-intensive tasks. In: Proceedings of Advances in Neural Information Processing Systems, 2023. 48573–48602

https://www.sciengine.com/doi/10.1360/SSI-2024-0329

Existence proof and improvement study of nontrivial upper bounds on task recognition of large language models

Xuanming NI, Qiaochu ZHAO & Song HUANG*

School of Software and Microelectronics, Peking University, Beijing 100871, China
* Corresponding author. E-mail: huangsong@ss.pku.edu.cn

Abstract Recent advancements in large language models (LLMs) have significantly enhanced their ability to solve tasks traditionally performed by humans, thereby narrowing the gap between human and artificial intelligence. However, the LLM performance remains highly sensitive to minor variations in prompts. One of the phenomena highlighted in this paper is that providing multiple examples of question-answer pairs can substantially improve LLM performance, even when the answers are randomly assigned. In the domain of in-context learning, LLMs leverage these examples through two mechanisms: task recognition and task learning, with the performance boost from randomly labeled examples attributed primarily to task recognition. This paper posits that the continued reliance on such examples underscores the need for improvement in task recognition. Building on this insight, we propose that this deficiency arises from the long-tailed characteristics of natural language data distribution and the inherent similarities between tasks. A series of experiments were conducted to validate the theoretical assumptions and conclusions. In addition, the impact of several factors on LLMs' task recognition ability was empirically analyzed. Based on the theoretical discussion, we also explore possible directions for improving LLMs' task recognition.

Keywords large language model (LLM), task recognition, few-shot prompting, data distribution, empirical risk

https://www.sciengine.com/doi/10.1360/SSI-2024-0329