上下文学习

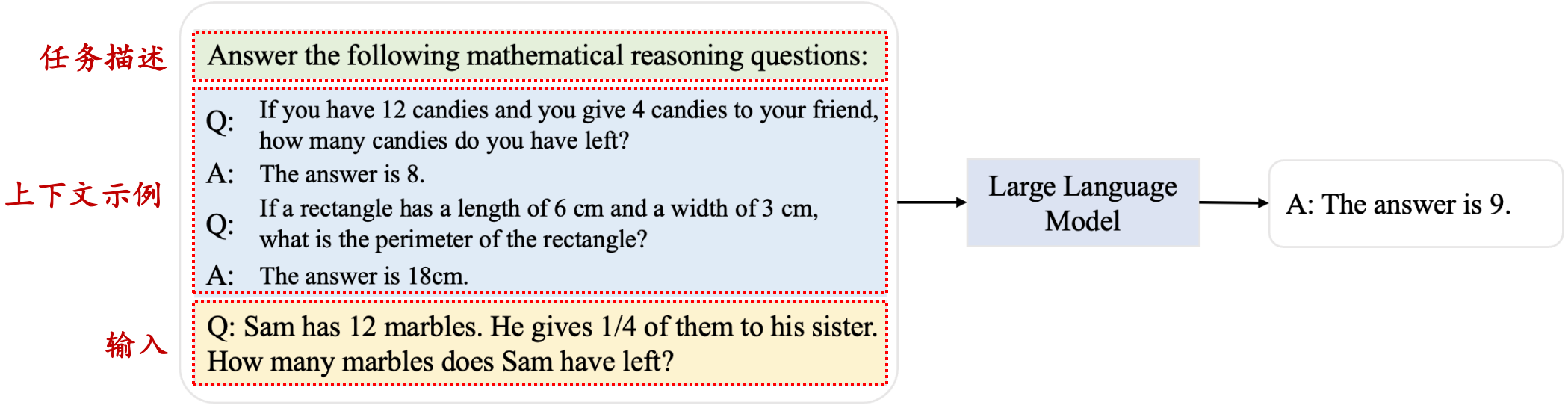
《大语言模型》编写团队：李军毅

上下文学习

AI Box

- 定义
  - 任务描述和(或)上下文示例所组成的自然语言文本作为提示

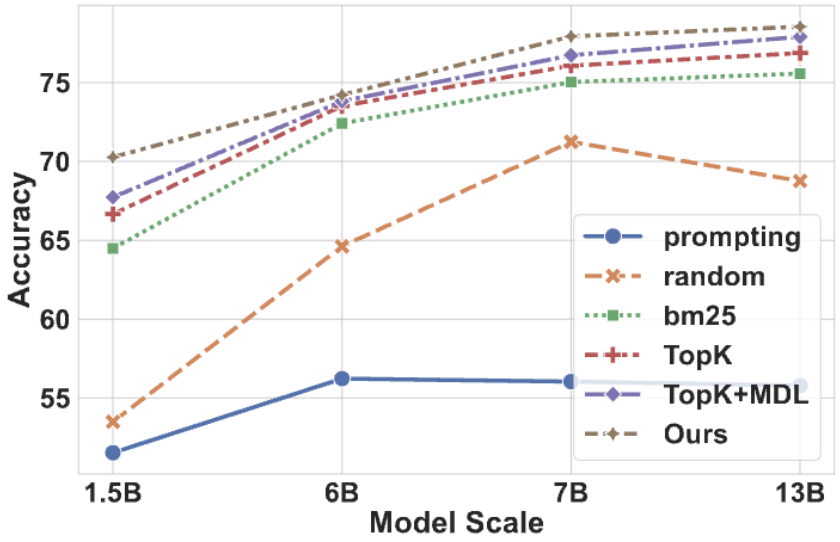
$$\underbrace{\text{LLM}}_{\text{大语言模型}}\Big(\underbrace{I}_{\text{任务描述}},\underbrace{f(x_1,y_1),\ldots,f(x_k,y_k)}_{\text{示例}},f(\underbrace{x_{k+1}}_{\text{输入}},\underbrace{\_\_\_}_{\text{答案}})\Big)\rightarrow\hat{y}_{k+1}.$$



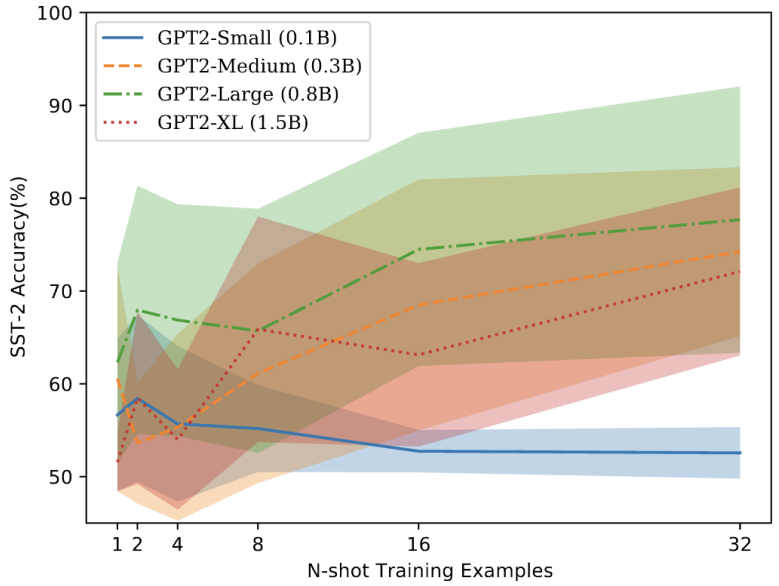
上下文学习

AI Box

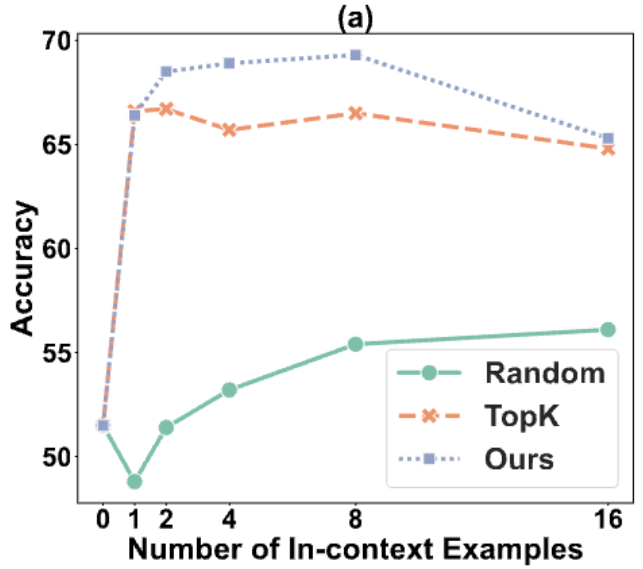
- 关键因素：上下文示例
  - 示例选择、示例顺序、示例数目



不同示例选择策略的性能差异



不同示例顺序导致性能方差大



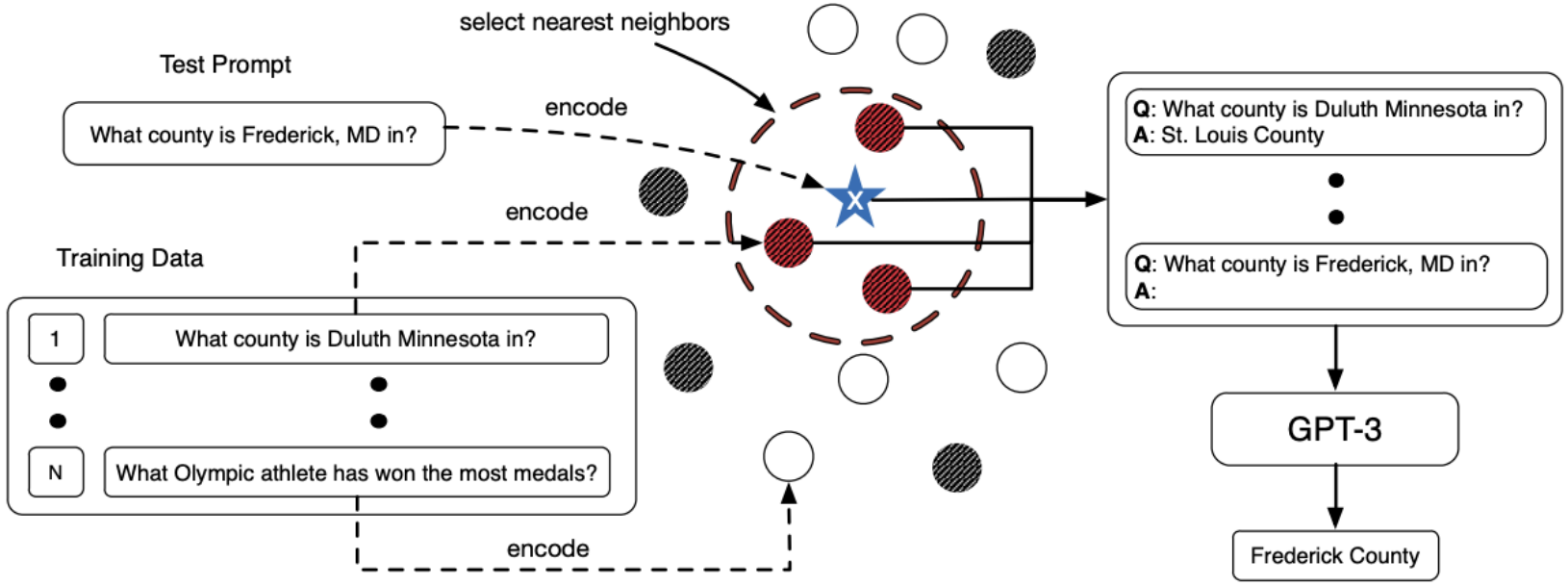
不同示例数目的性能差异

Revisiting Demonstration Selection Strategies in In-Context Learning. **ACL 2024.**

示例选择

AI Box

- 基于相关度排序的方法
  - 基于 k 近邻的相似度检索算法：根据候选样本与测试样本的嵌入语义相似度进行排序，并选择出最相关的 k 个示例



What Makes Good In-Context Examples for GPT-3? **ACL 2022.**

示例选择

AI Box

- 基于集合多样性的方法
  - 针对特定任务选择出具有代表性的、信息覆盖性高的示例集合
  - 如 MMR (Maximum Margin Relevance)、DPP (Determinantal Point Process) 算法

Algorithm 1 MMR-Based Exemplar Selection

procedure MMRSELECT($D, q, k, \mathcal{S}$)
input: exemplar pool $D = \{q_1 ... q_n\}$, test query q, number of shots m and similarity measurement $\mathcal{S}$
output: selected exemplars $T = \{q_1 ... q_m\}$

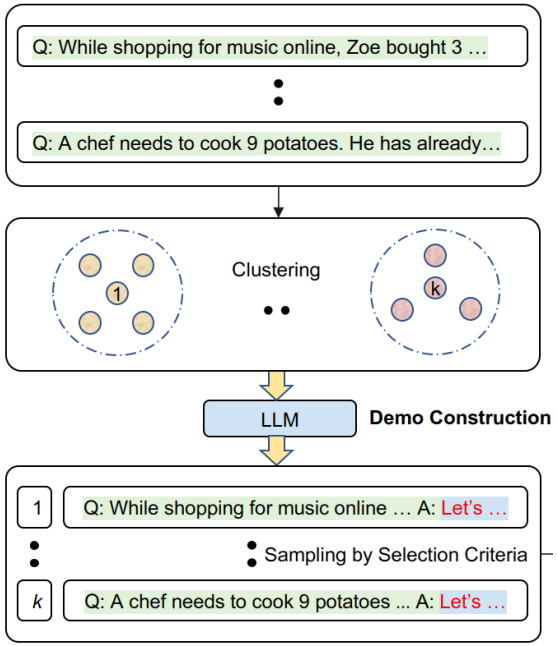
$\mathbb{S} := [[\mathcal{S}(q_i, q_j)]]_{q_i, q_j \in D}$; ▷ the pairwise similarity between exemplars in D
$\mathbb{Q} := [\mathcal{S}(q, q_i)]_{q_i \in D}$; ▷ the similarity between query and exemplars in T
$T := \{\}$;
while $|T| < k$ **do**
  $\hat{q} :=$ Equation(1); ▷ get the next exemplar based on Eq (1)
  T.add($\hat{q}$)
return T;

$$\arg\max_{q_j \in D/T} \lambda \mathcal{S}(q, q_j) - (1 - \lambda) max_{q_i \in T} \mathcal{S}(q_j, q_i) \quad (1)$$

将样例进行聚类

选取多样化的样例

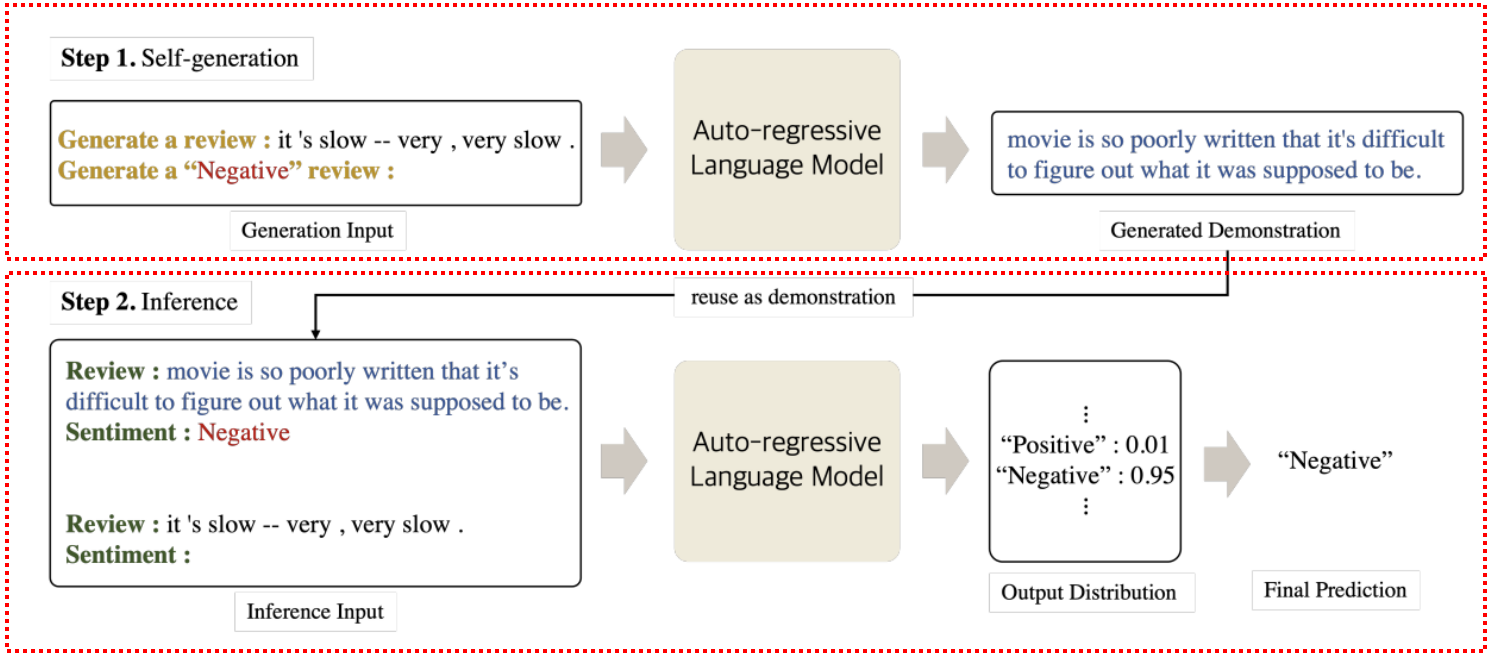
与问题相关但与集合示例相似度低



示例选择

AI Box

- 基于大语言模型的方法
  - 根据加入示例前后大模型性能的增益来评估示例选择的有效性
  - 直接利用大模型生成与测试样本相关的上下文示例

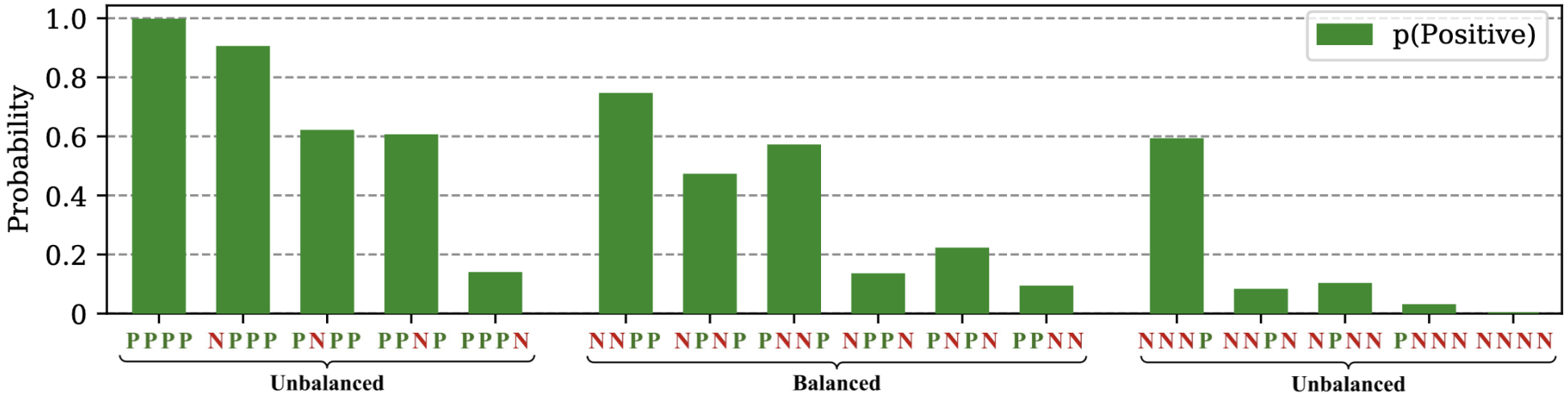


生成与输入评论相关且为“消极”的评论作为示例

基于大模型生成的评论示例预测输入评论的情感

示例顺序

- 基于相关度的示例顺序
  - 大模型在预测时更倾向于提示末端的示例 (更靠近测试输入)
  - 根据示例与测试样本的相似度排序，相似度越高则越靠近测试样本



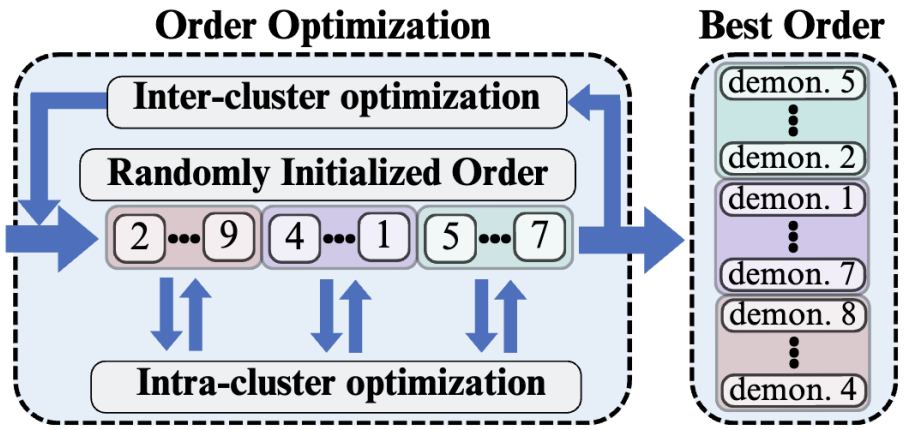
测试样本的预测结果与末端示例的标签高度正相关

Calibrate Before Use: Improving Few-Shot Performance of Language Models. **ICML 2021.**

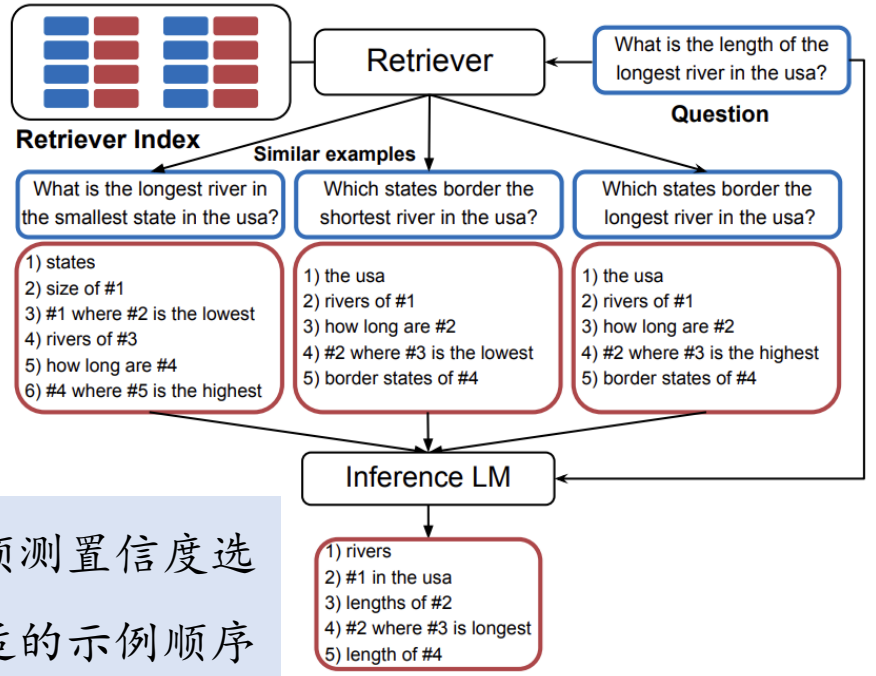
示例顺序

AI Box

- 基于任务性能的示例顺序
  - 选择少量数据测试大模型基于不同示例顺序的任务性能
  - 大模型预测置信度也可作为评估示例顺序的指标



根据任务性能优化示例顺序，选择最佳的示例顺序



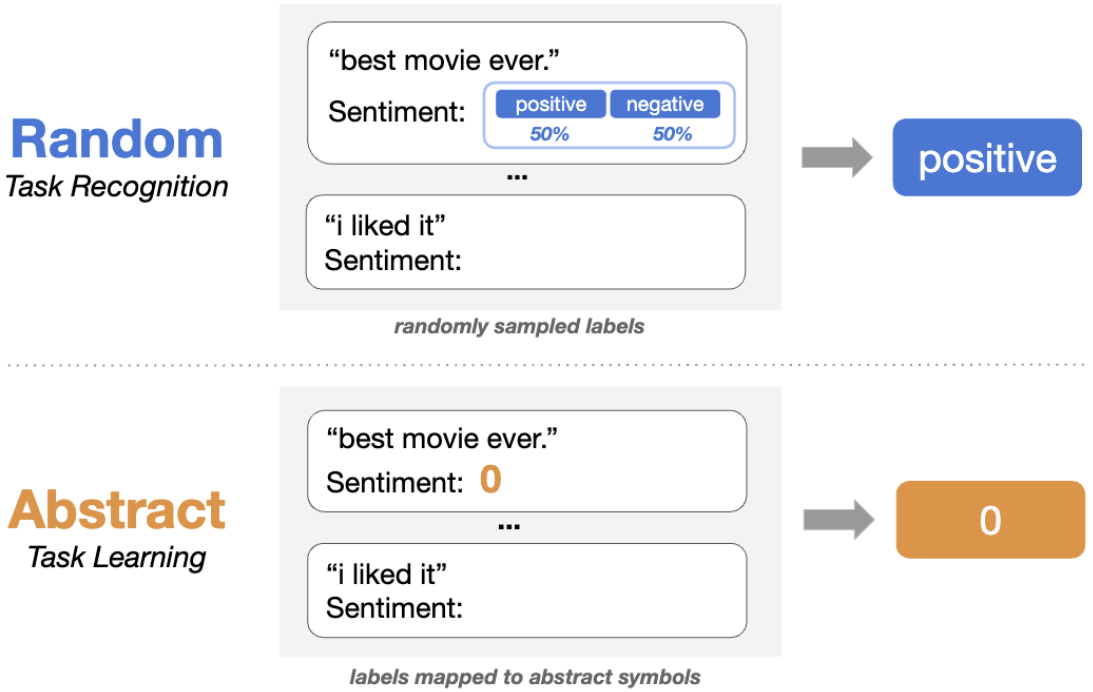
根据预测置信度选择合适的示例顺序

Learning To Retrieve Prompts for In-Context Learning. **NAACL 2022.**

上下文学习的工作机制

AI Box

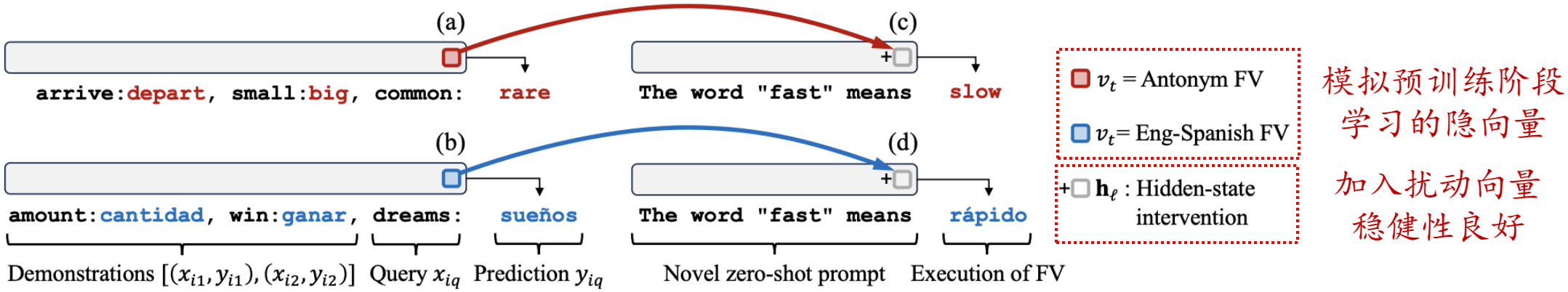
- 大模型执行上下文学习的两种范式
  - 任务识别 (Task Recognition)
    - 使用预训练阶段习得的知识解决任务
    - 例如，预测评论的情感
  - 任务学习 (Task Learning)
    - 从示例中学习如何解决新任务
    - 例如，预测评论的情感并转换为 0/1



上下文学习的工作机制

AI Box

- 任务识别
  - 模型在预训练阶段编码表征任务信息的隐向量 (学习的先验知识)，可用于在推理阶段解决任务
  - 以预测反义词和西班牙语为例



模拟预训练阶段学习的隐向量

加入扰动向量稳健性良好

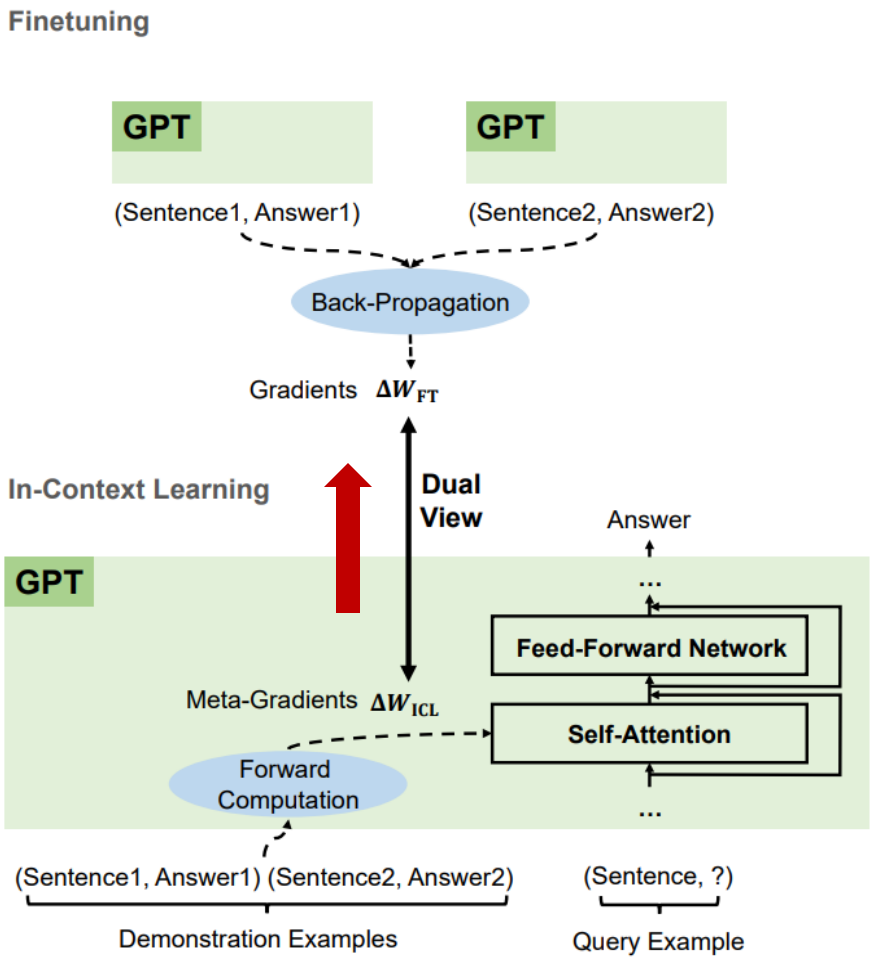
Function Vectors in Large Language Models. **ICLR 2024**

上下文学习的工作机制

AI Box

- 任务学习
  - 上下文学习是微调的对偶形式
    - 注意力机制 → 梯度下降
    - LinearAttn $(\boldsymbol{V}, \boldsymbol{K}, \boldsymbol{q})$

$$
\begin{aligned}
\mathcal{F}(\mathbf{x}) &= (W_0 + \Delta W)\,\mathbf{x} \\
&= W_0\mathbf{x} + \Delta W\mathbf{x} \\
&= W_0\mathbf{x} + \sum_i \left(\mathbf{e}_i \otimes \mathbf{x}_i'^T\right)\mathbf{x} \\
&= W_0\mathbf{x} + \sum_i \mathbf{e}_i \left(\mathbf{x}_i'^T\mathbf{x}\right) \\
&= W_0\mathbf{x} + \text{LinearAttn}\left(E, X', \mathbf{x}\right)
\end{aligned}
$$



上下文学习的工作机制

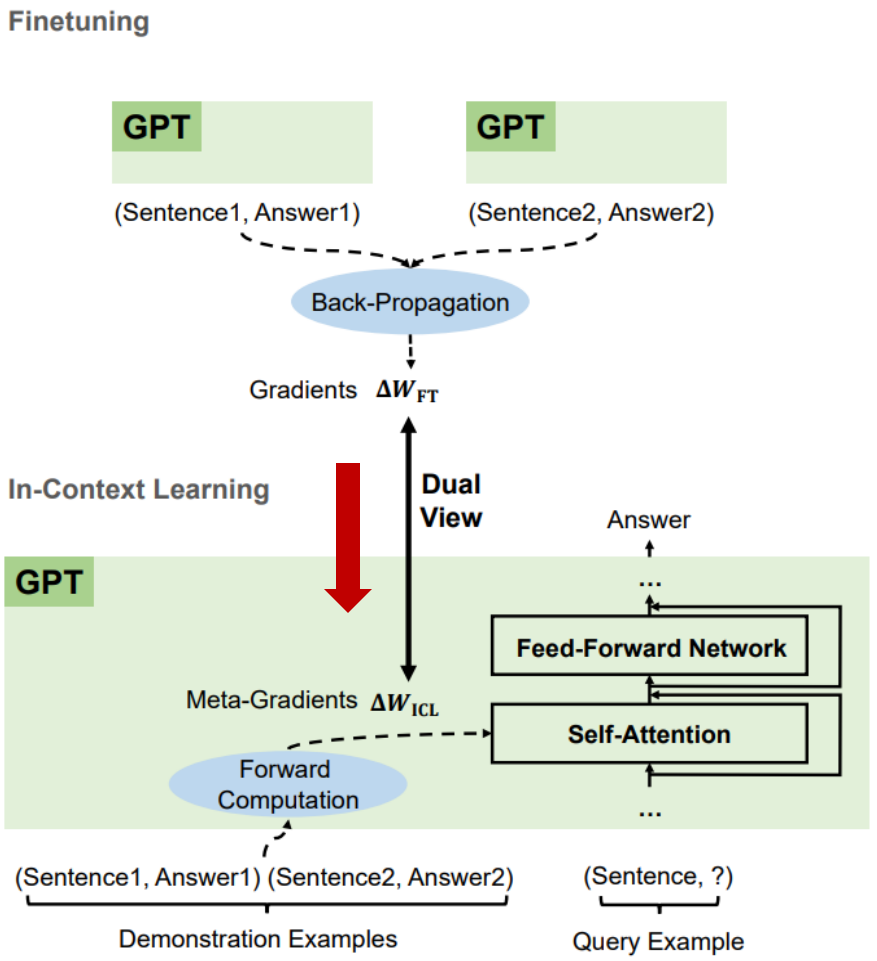
AI Box

- 任务学习
  - 上下文学习是微调的对偶形式
    - 梯度下降 → 注意力机制

$$
\begin{aligned}
\mathcal{F}_{\mathrm{ICL}}(\mathbf{q}) &= \mathrm{Attn}(V, K, \mathbf{q}) \\
&\approx W_V[X'; X] \left(W_K[X'; X]\right)^T \mathbf{q} \\
&= W_V X \left(W_K X\right)^T \mathbf{q} + W_V X' \left(W_K X'\right)^T \mathbf{q} \\
&= \widetilde{\mathcal{F}}_{\mathrm{ICL}}(\mathbf{q}).
\end{aligned}
$$

W_{ZSL}: no demonstration

$$
\begin{aligned}
\widetilde{\mathcal{F}}_{\mathrm{ICL}}(\mathbf{q}) &= W_{\mathrm{ZSL}}\mathbf{q} + W_V X' \left(W_K X'\right)^T \mathbf{q} \\
&= W_{\mathrm{ZSL}}\mathbf{q} + \mathrm{LinearAttn}\left(W_V X', W_K X', \mathbf{q}\right) \\
&= W_{\mathrm{ZSL}}\mathbf{q} + \sum_i W_V \mathbf{x}'_i \left(\left(W_K \mathbf{x}'_i\right)^T \mathbf{q}\right) \\
&= W_{\mathrm{ZSL}}\mathbf{q} + \sum_i \left(W_V \mathbf{x}'_i \otimes \left(W_K \mathbf{x}'_i\right)^T\right) \mathbf{q} \\
&= W_{\mathrm{ZSL}}\mathbf{q} + \Delta W_{\mathrm{ICL}}\mathbf{q} \\
&= \left(W_{\mathrm{ZSL}} + \Delta W_{\mathrm{ICL}}\right) \mathbf{q}.
\end{aligned}
$$

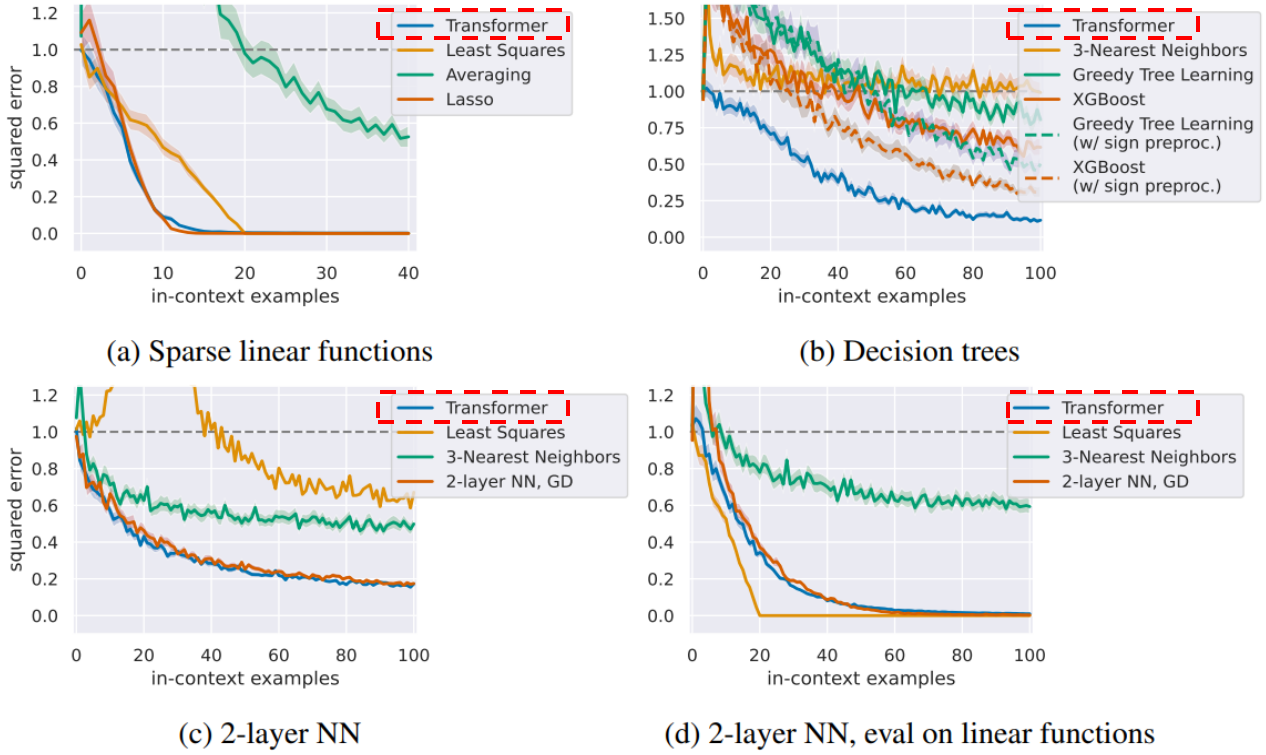


Why Can GPT Learn In-Context? Language Models Implicitly Perform Gradient Descent as Meta-Optimizers. **ACL 2023**

上下文学习的工作机制

AI Box

- 任务学习
  - 可以通过上下文学习学会复杂函数的输入输出映射关系



(a) Sparse linear functions　(b) Decision trees

(c) 2-layer NN　(d) 2-layer NN, eval on linear functions

- 在**预训练阶段**，大模型通过其内部参数能够学习隐式的函数关系
- 在**上下文学习阶段**，借助给定示例，大模型能够通过这个隐式模型模拟出诸如决策树等复杂学习算法

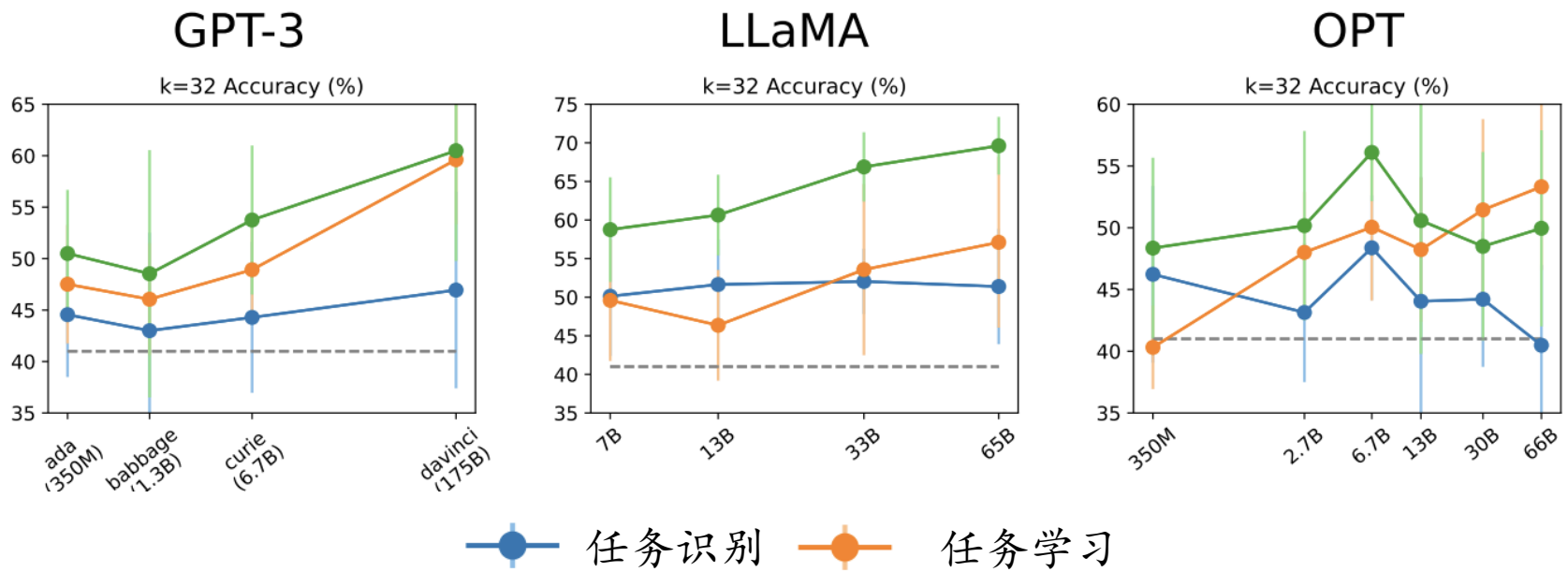
上下文学习的工作机制

AI Box

- 任务识别和任务学习
  - 大语言模型能够同时展现出任务识别和任务学习两种能力
  - 这两种能力的强弱与模型的规模紧密相关

规模较小的模型就能具备任务识别能力

较大规模的模型具备更强的任务学习能力



训练任务对上下文学习的影响

AI Box

- 训练任务
  - 通过设计专门的训练任务进行继续预训练或微调也能获得上下文学习能力
    - MetaICL: 元训练任务训练模型根据具体任务示例和输入预测相应的输出

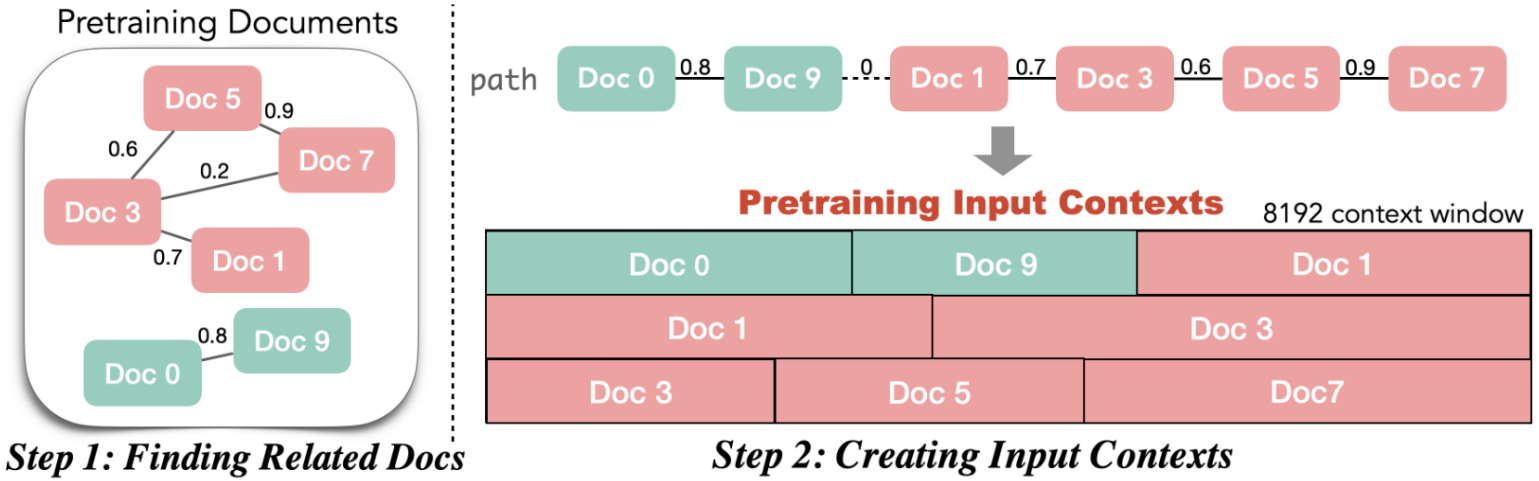
Method	HR→LR	Class →Class	non-Class →Class	QA →QA	non-QA →QA	non-NLI →NLI	non-Para →Para
			All target tasks				
0-shot	34.8	34.2	34.2	40.2	40.2	25.5	34.2
PMI 0-shot	35.1	33.8	33.8	40.2	40.2	27.9	39.2
Channel 0-shot	36.5	37.3	37.3	38.7	38.7	33.9	39.5
In-context	38.2/35.3	37.4/33.9	37.4/33.9	40.1/38.7	40.1/38.7	34.0/28.3	33.7/33.1
PMI In-context	39.2/33.7	38.8/30.0	38.8/30.0	40.3/38.8	40.3/38.8	33.0/28.0	38.6/33.4
Channel In-context	43.1/38.5	46.3/40.3	46.3/40.3	40.8/38.1	40.8/38.1	39.9/34.8	45.4/40.9
Multi-task 0-shot	35.6	37.3	36.8	45.7	36.0	40.7	30.6
Channel Multi-task 0-shot	38.8	40.9	42.2	42.1	36.4	36.8	35.1
MetaICL	43.3/41.7	43.4/39.9	38.1/31.8	**46.0**/44.8	38.5/36.8	49.0/44.8	33.1/33.1
Channel MetaICL	**49.1**/46.8	**50.7**/48.0	**50.6**/48.1	44.9/43.5	**41.9**/40.5	**54.6**/51.9	**52.2**/50.3
Fine-tune	46.4/40.0	50.7/44.0	50.7/44.0	41.8/39.1	41.8/39.1	44.3/32.8	54.7/48.9
Fine-tune w/ meta-train	52.0/47.9	53.5/48.5	51.2/44.9	46.7/44.5	41.8/39.5	57.0/44.6	53.7/46.9

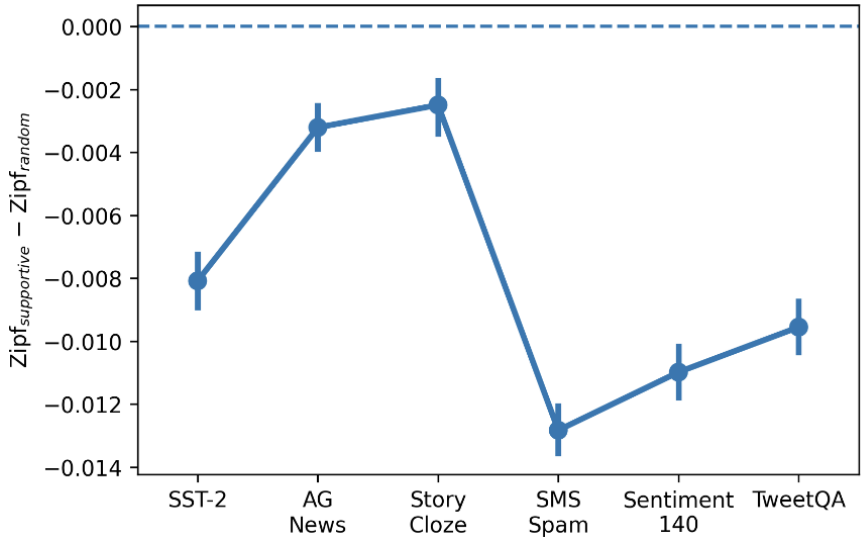
MetaICL: Learning to Learn In Context. **NACCL 2022.**

预训练对上下文学习的影响

AI Box

- 预训练数据
  - 预训练数据的多样性和长程依赖关系，以及高密度低频长尾词汇





根据训练数据相关性对文档排序依次训练

Zipfian系数显著低于随机训练数据

In-context Pretraining: Language Modeling Beyond Document Boundaries. **ICLR 2024.**
Understanding In-Context Learning via Supportive Pretraining Data. **ACL 2023.**

谢谢