针对数学推理的样本难度自适应的有监督微调

童雨轩
2021012894
计13
tongyx21@mails.tsinghua.edu.cn

覃秋杰栎
2021010859
计17
tqjl21@mails.tsinghua.edu.cn

洪亦歆
2020011629
经01
hongyx20@mails.tsinghua.edu.cn

1 简介

本工作希望研究**数学推理**中，**训练数据样本的难度分布**对LLM（Large Language Model，大语言模型）的**SFT (Supervised Fine-Tuning，有监督微调)** 过程有什么影响，以及如何利用这一点改善LLM 的数学推理能力。

已有的数学推理LLM SFT 训练数据集与框架通常**没有考虑到训练样本的难度分布这一因素**，而仅仅进行了数据合成、增强或数据集汇编，**停留在数据集层面**。

先前的研究和与人类的类比都显示，**神经网络模型可能对于简单的问题需要学习的次数更少，反之困难的问题则更多**。同时，我们注意到，常规的训练方法中，通常**在同一批数据上反复训练n 个epochs，这近似于将同一批数据上采样为n 倍并遍历一次**（尽管shuffle 的范围略有不同），这很可能不是最优的策略。综合这两点，我们提出一种新的SFT 方法，**根据不同训练样本的难度决定最终训练过程中每个样本被学习的次数，两者成正相关关系。基于我们的方法训练得到的模型，在MATH 这一困难测试集上相对baseline (36.5%) 获得了$\sim 2\%$ 的提升，但在部分简单测试集上则performance 略有下降**。我们认为后者可能是因为我们对训练样本难度的定量刻画方式、样本学习次数关于难度的设置策略、相关超参数设置并非最优，还需要进一步的探索。

此外，由于通过自然语言或编程进行数学推理无论在原理上还是实际效果上都存在显著差别，我们对CoT(Chain-of-Thoughts)[1]，PoT(Program-of-Thoughts)[2] 及其组合[3]等decoding 方法进行了消融实验分析，并分析了这些推理形式与样本难度的关系。我们发现，**尽管对于目前的LLM，使用编程进行数学推理通常比自然语言效果更好，但两种推理形式具体擅长的任务以及对应训练样本难度的分布都很不同**。因此，我们认为，对于数学推理LLM，同时掌握自然语言与编程两种推理形式非常重要，并且对样本难度分布的精细调整可能会有利于实现这一点。

代码与数据可见GitHub 仓库。

2 相关工作

LLM 数学推理SFT：先前在LLM 数学推理任务中到达过SotA 的工作中，

- MAmmoTH 提出的MathInstruct 数据集[3]是将MATH[4], GSM8K[5] 等多个常用数据集汇编到一起，将MATH 中的训练样本上采样到1.5x，并合成了一些使用PoT 推理的训练样本。
- MetaMath 提出的MetaMathQA 数据集[6] 通过4 种方法对MATH, GSM8K 进行了数据增强，对于生成的数据，仅过滤掉了答案错误的样本。

训练样本难度对LLM 学习效果的影响：Bao 等人[7]指出，在token 级别，存在以下现象：在训练结束时，简单的token 已经被过拟合，而困难的token 还欠拟合。

3 方法

我们的方法在数学推理任务上，根据训练样本的难度自适应地调整其分布，并进行SFT。具体来说，我们的方法包含2 个初始要素：

1. **base model** m ；
2. N 个不同的训练样本组成的**初始训练数据集**$D = \{s_1, s_2, \ldots, s_N\}$，其中$s_i$ 是query q_i 与response r_i 组成的序偶，即$s_i =< q_i, r_i >$。

我们的方法具体包含如下步骤：

1. **对每个样本**s_i **计算其难度**d_i。此处，我们先计算**base model 对query** q_i **进行ICL（In-Context Learning, 上下文学习）[8]采样的通过率**$p_i \in [0, 1]$ （此时需要将使用到的ICL 示例从原始数据集中删去，得到D'），再线性映射到离散的难度等级$d_i = \lfloor (1 - p_i) * L \rfloor \in \{0, 1, \ldots, L\}$ （其中L 为最高的难度等级），作为难度的proxy，故$d_i = f_m(s_i)$，其中f_m 是与m 有关的从训练样本到难度等级的映射；
2. **调整最终的训练数据集**D^* **中不同难度**d_i **训练样本**s_i **的出现次数**$g(s_i, d_i)$ **，实现训练样本越困难，学习次数越多**。此处，我们指定**2 个超参数，基值**b **与权重**w，对训练样本进行**线性（上）采样**；具体来说，对于任意样本$s_i \in D'$，采样$\lfloor b + w * d_i \rfloor$ 次，加入D^*，即

$$D^* = \{s_1, \ldots, \underbrace{s_i, s_i, \ldots, s_i}_{\lfloor b + w * d_i \text{ 个} s_i \rfloor}, \ldots, s_N\}$$

3. **在**D^* **上对**m **SFT 1 个epoch**，即每个不同样本被学习的次数仅取决于（上）采样后在D^* 中的数量。

4 实验

4.1 实验设置

4.1.1 训练

Base Model：我们选择了**经过了代码与数学继续预训练的LLemma 系列模型[9]**作为base model，由于算力与时间限制，我们目前只基于LLemma-7B 进行了实验，在进行更充分的探索后，再考虑将方法拓展至LLemma-34B 与其他系列的base model。

原始训练数据集：我们选择了**MathInstruct 的子集**作为原始数据集D，其包含$N \approx 169$k 个训练样本。具体来说，我们去除了其中的CAMEL-Math 与TheoremQA 部分，原因如下：

- CAMEL-Math[10]
  - 数据不服从固定格式，难以提取最终答案并计算通过率；
  - 样本的正确性未经过验证与过滤。
- TheoremQA[2]
  - 经检查发现，数据质量较差，大部分样本并不包含推理过程；
  - 样本数量非常小（$600 * 2 \ll 169$k）。
- MAmmoTH[3] 论文中关于数据集构成的消融实验也说明这两个子集对模型performance 影响很小。

样本难度的定量估计：

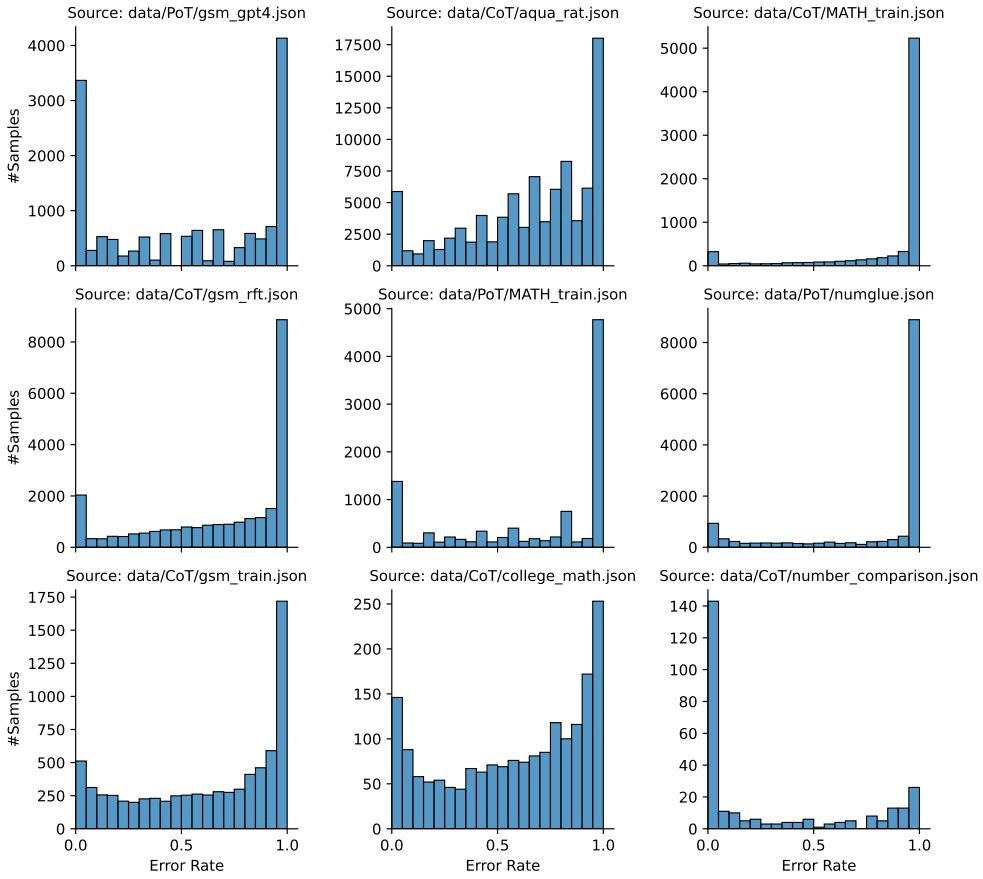

Figure 1: 本工作采用的原始训练数据集的不同来源的子集中，LLemma-7B 使用 2048 tokens 示例进行ICL 的不同错误率的样本数分布。横轴为错误率，越接近原点，表示样本query 越简单；纵轴为对应样本数。"source" 表示样本的初始来源。

- 计算base model 对query 进行ICL 采样的通过率时，我们需要**选择合适的ICL 示例，以保证通过率计算在不同样本之间的公平性**。由于D 通常由M 个不同来源的子数据集S_j $(j \in \{1, \ldots, M\})$ 汇编得到，即$D = \bigcup_{j=1}^{M} S_j$，对于每个$S_j$ ，我们随机选择l_j 个ICL 示例$e_{j,k} \in S_j$ $(k \in \{1, \ldots, l_j\})$，直到**这些示例与对应的指令的token 数之和首次超过base model 预训练上下文窗口长度的一半（此处为$2048 = 4096/2$）。
- 将通过率映射到线性离散的难度等级时，我们设置$L = 5$。

难度自适应采样训练样本的超参数：此处，对于$< b, w >$，我们尝试了多种取值，具体参见4.3 ，4.3 或1。

训练超参数：对于所有最终数据集$D^*_{b,w}$，如方法一节所述，我们**均只训练1 个epoch**；对于其他超参数，**仿照MAmmoTH[3]**，我们使用如下取值

- 优化器：AdamW[11]
- 学习率：$2 * 10^{-5}$
- Batch Size：128
- 最大序列长度：512（所有序列都pad 到该长度）
- Weight Decay：0
- 使用了DeepSpeed-ZeRO-1[12] 与Flash-Attention-v2[13] 来节约显存与提高训练速度。

4.1.2 评测

Decoding 方法:

- 对于所有模型，由于经过了指令微调，且数据集中包含选择题格式数据，故均采用0-shot 设定，无需提供上下文学习示例。
- 为了利用同时利用模型的自然语言推理与编程能力，仿照MAmmoTH[3]，我们采用了**结合PoT/PAL 与CoT 的decoding 方法，称为"PoT||CoT" 或简称为"fallback"：先提示模型生成Python 程序尝试解决问题，如果程序报错或超时，则提示模型使用自然语言推理来解决问题**。
- 此外，对于选择题，模型若未直接输出选项，则提示模型选择最接近的选项；若再次失败，则默认选择A。

评测数据集:

- 对于分布内performance 的评测，我们使用了MATH[4], GSM8K[5], AQuA[14], NumGLUE[15] 的测试集。其中MATH 采样自美国高中数学竞赛，涉及较为复杂的数学知识，而其他数据集主要由相对简单的数学应用题(Math Word Problem, MWP) 组成。
- 对于分布外performance，即泛化能力的评测，我们使用了MMLU-Math[16], DeepMind-Mathematics[17], SAT-Math[18], SVAMP[19] 这4 个benchmarks。其中，MMLU-Math 涉及至多大学级别的数学问题，DeepMind-Mathematics 构造自涉及8 个数学子领域的问题，SAT-Math 采样自SAT 数学考试，涉及至多（美国）高中级别的数学知识，前三者相对困难，而SVAMP 由数学应用题的简单变式组成，相对简单。

Baseline：我们选择了**在LLemma-7B 上，仿照MAmmoTH，使用完整的MathInstruct 数据集SFT 3 个epochs**，其余超参数设置也与MAmmoTH 相同，得到LLemma-7B-MathInstruct，作为baseline。注意，baseline 使用的训练数据集是我们使用的初始训练数据集的超集，尽管如"原始训练数据集"一段分析，差集对模型效果的影响并不显著。

4.2 实验结果

4.3 主要实验结果

从分布内、外数据集上的评测结果4.34.3可见：

- 对于**我们的方法的效果**，我们的方法在MATH, DeepMind-Mathematics 这两个较为困难的数据集上带来了一定提升，在其他较为简单的数据集上则通常导致performance 或多或少的下降。这说明，**为困难样本分配较大的权重，确实能提高在困难数据集上的performance，但这可能以在简单数据集上的performance 下降为代价；需要思考如何缓解这一代价**。我们猜测，**这可能是因为简单的样本仍然需要保持一定的学习次数，此处即b 仍然不能太小**，但由于算力与时间限制，我们未能探索更大的b 取值。
- 对于**训练样本难度分布的设置，此处即b, w 的取值**，可以看到，通过调整训练样本难度的分布，有可能超越baseline，但问题在于**分数的最大值通常出现在不同的b, w 取值中**，这可能表示**最佳的训练样本难度分布无法仅用通过率与线性映射描述**，需要进一步研究如何更好地对训练样本难度分布进行自适应地调整。
- 值得注意的是，**许多benchmarks 上的最优值都出现在小于baseline 的训练步数上**，这意味着简单重复若干个epoch 确实会导致部分数据被过拟合，模型对应能力下降。

4.4 消融实验

从对分布内、外测试集上不同Decoding 方法的评测结果4.4 4.4 4.4 可见：对于**Decoding 方法本身**，

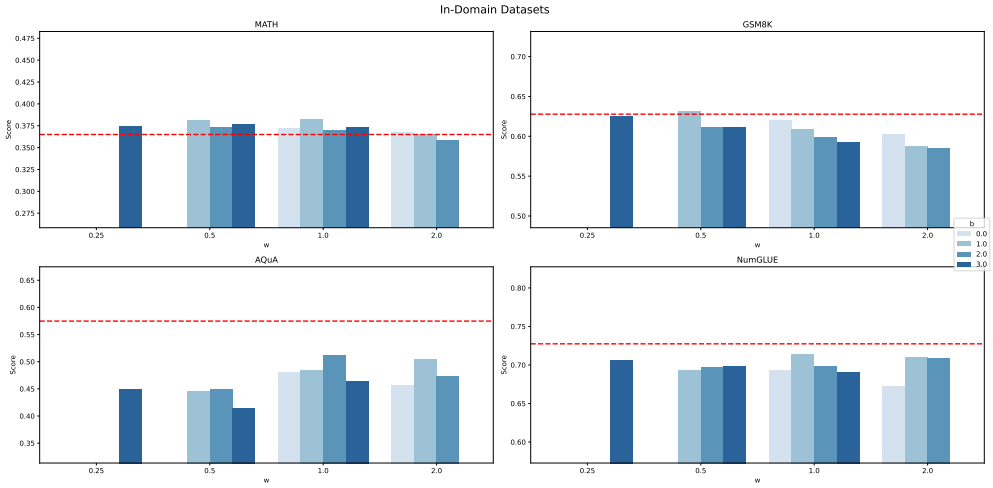

Figure 2: 分布内数据集上，不同b, w 取值下训练LLemma-7B 得到的模型的评测得分。红色虚线表示baseline （LLemma-7B-MathInstruct）的得分。b 的取值由颜色表示，颜色越深，b 越大。具体数值详见1

o

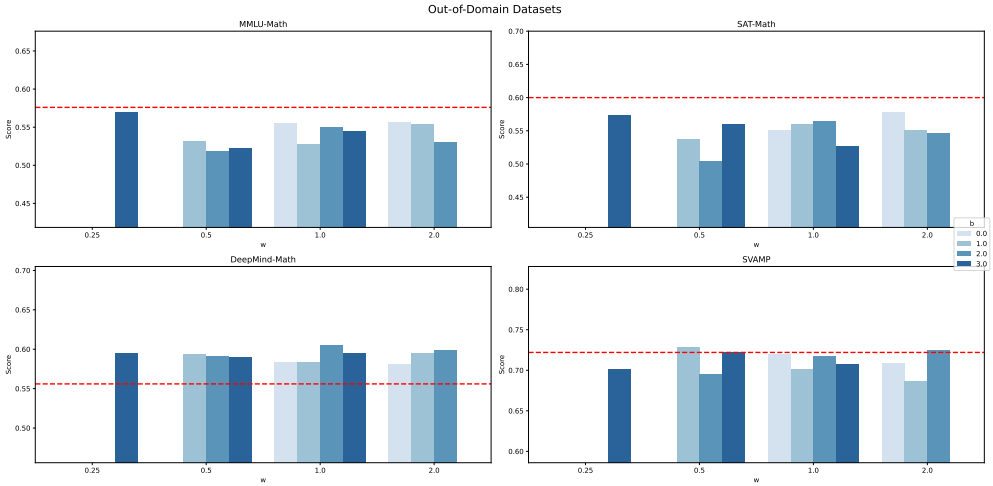

Figure 3: 分布外数据集上，不同b, w 取值下训练LLemma-7B 得到的模型的评测得分。红色虚线表示baseline （LLemma-7B-MathInstruct）的得分。b 的取值由颜色表示，颜色越深，b 越大。

- **CoT 与PoT 能解决的问题重合度较低，CoT 与PoT 相比All 能解决的问题都明显增多**，说明自然语言与程序对应推理能力差异较大。

- PoT 能解决的问题相比CoT 的多少与数据集难度很有关：**对于较困难的数据集MATH，PoT 显著强于CoT；反之，PoT 与CoT 能力相近**。这意味着目前阶段的LLM 使用自然语言推理解决数学问题的能力仍然弱于使用编程。

因此，**同时掌握自然语言推理与编程能力对于LLM 提升数学推理能力非常重要**。

对于**我们的方法与Decoding 方法及其相应能力的关系**，我们的方法**目前倾向于提升PoT 的performace，略微降低CoT 的performance**。结合4.1.1 分析，这可能是因为**PoT 相关训练样本的难度分布更倾向于困难一端，因此在我们的方法中受益更多**。

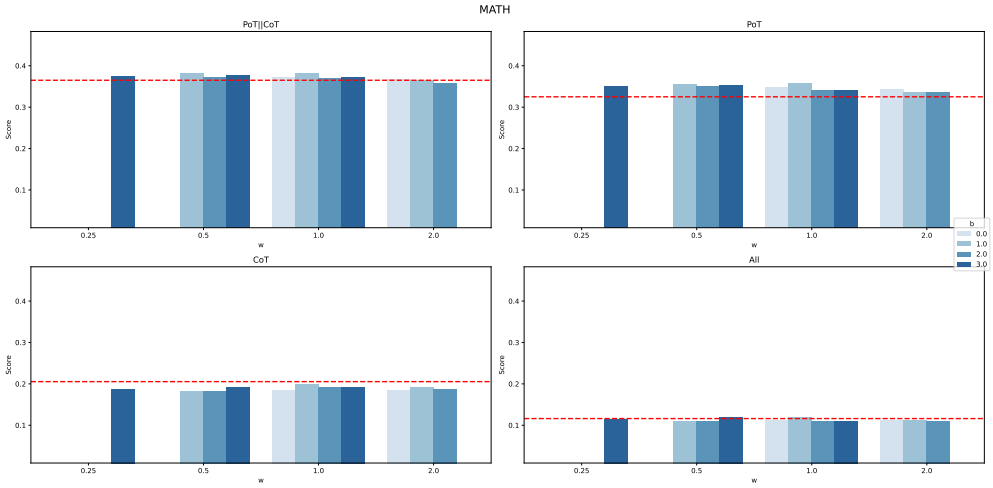

Figure 4: MATH 测试集（分布内数据集之一）上不同Decoding 方法的评测结果。其中"PoT||CoT" 表示**Decoding 方法** 部分介绍的方法，All 表示两者都可以正确解答。

5 总结

5.1 贡献

- 提出了一种对训练样本难度自适应的SFT 方法，验证了在困难任务上的效果提升，发现了在简单任务上效果下降的问题。
- 对LLM 使用自然语言或编程进行数学推理进行了一些分析，指出目前两者的能力范围差异较大，LLM 进行进行数学推理是应当同时掌握使用两者的能力。

5.2 未来工作

更好的训练样本难度的定量指标。base model 在训练样本query 上进行ICL 的采样通过率尽管具有面向模型的优势，但至少存在以下缺陷：

- 取值范围有限（$[0, 1]$），大量样本位于$0/1$ 这两端，没能被精确刻画难度；
- 计算成本高，依赖于LLM 的推理；

Model Info			In-Domain Datasets				Out-of-Domain Datasets			
b	w	Steps	MATH	GSM	AQuA	NumG.	MMLU	SAT	DM	SVA.
Baseline		6144	36.5	62.77	**57.48**	**72.74**	**57.60**	**60.00**	55.60	72.20
0	1	4651	37.18	62.09	48.03	69.29	55.54	55.00	58.30	72.00
0	2	9302	36.76	60.27	45.67	67.27	55.65	57.73	58.10	70.80
1	0.5	**3195**	38.18	**63.15**	44.49	69.39	53.18	53.64	59.40	**72.80**
1	1	5970	**38.28**	60.88	48.43	71.40	52.77	55.91	58.30	70.10
1	1	5970	35.68	59.29	48.03	66.41	52.77	51.82	55.90	72.00
1	2	10621	36.54	58.76	50.39	71.02	55.34	55.00	59.50	68.60
2	0.5	4514	37.32	61.18	44.88	69.77	51.85	50.45	59.10	69.50
2	1	7289	36.98	59.89	51.18	69.87	55.03	56.36	**60.50**	71.70
2	2	11940	35.84	58.53	47.24	70.83	53.08	54.55	59.90	72.50
3	0.25	4749	37.42	62.55	44.88	70.63	56.98	57.27	59.50	70.10
3	0.5	5833	37.70	61.18	41.34	69.87	52.26	55.91	59.00	72.20
3	1	8608	37.34	59.29	46.46	69.00	54.52	52.73	59.50	70.70

Table 1: baseline 与不同b, w 取值下训练得到的模型的评测得分。缩写说明：GSM(GSM8K)，NumG.(NumGLUE)，MMLU(MMLU-Math)，SAT(SAT-Math)，DM(DeepMind-Mathematics)，SVA.(SVAMP)。

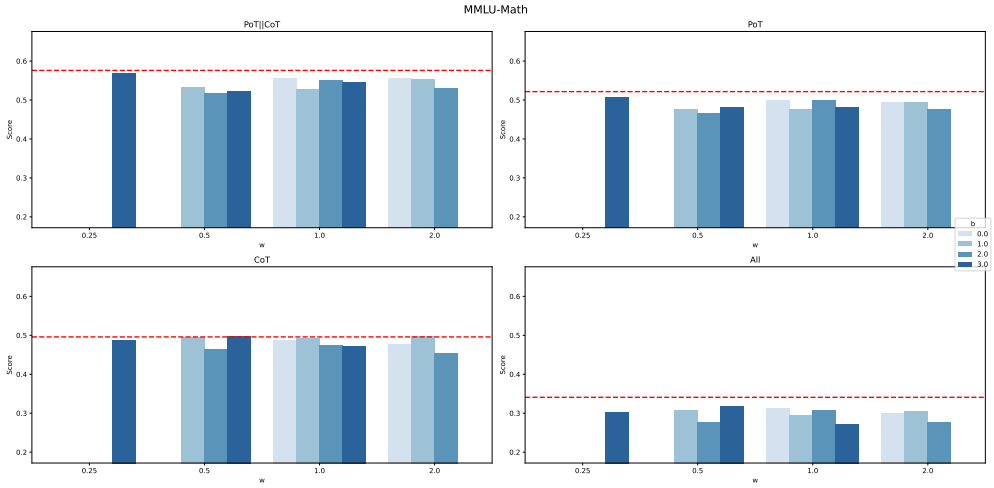

Figure 5: MMLU-Math 测试集（分布外数据集之一）上不同Decoding 方法的评测结果。其余同Figure 3。

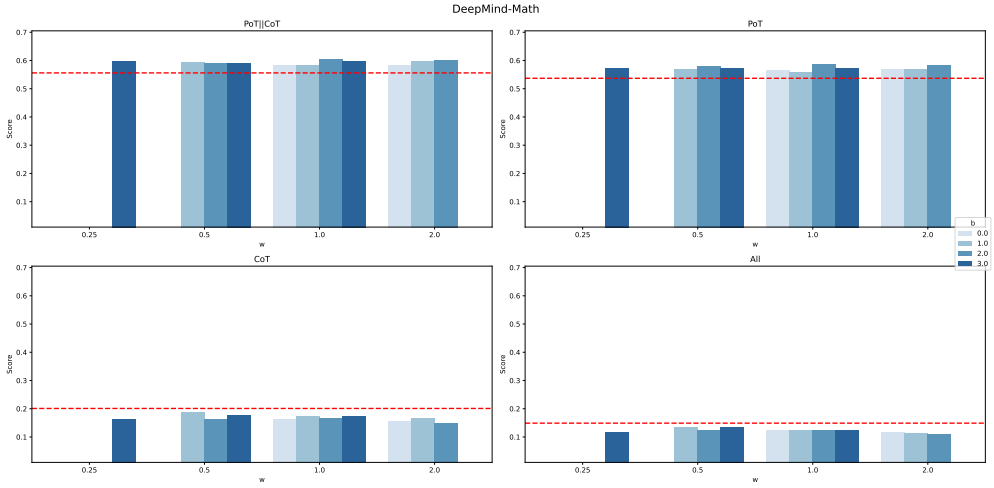

Figure 6: DeepMind-Mathematics 测试集（分布外数据集之一）上不同Decoding 方法的评测结果。其余同Figure 3。

- 无法反映response 的难度；
- 取值不连续，变化的单位是一个query 上采样结果的正误，当采样次数较少时误差较大。

除此之外，还可以尝试基于LLM 对训练样本进行对比、排序、打分等，并进一步计算难度评分或训练奖励模型等。

更好的训练样本难度的学习次数设置策略。本工作假设了合适的学习次数与训练样本难度成线性关系，这是一个简略的设置，很可能并非最优策略。除此之外，还可以

- 尝试更多分布模型，例如指数分布等；
- 更仔细地考虑边界情况，例如部分难度指标特别高的样本，可能是由于其噪声非常大而无法被正确回答，这些样本更应该被下采样或直接舍弃。

更多角度的训练过程设置。本工作主要关注不同难度训练样本的数量分布。除此之外，还可以尝试

- 引入课程学习视角，考虑不同难度训练样本的顺序安排；

- 修改模型的loss 函数，例如调整不同难度样本的loss 权重等。

更多维度、更细粒度的训练数据信息利用。本工作主要关注训练样本的难度，但

- 显然训练样本的其他属性（例如质量、多样性等）同样可能对训练过程有着显著影响；
- 而且许多属性可能适合更细粒度的刻画，例如在步骤粒度衡量数据的难度可能是更加合理的选择。

参考文献

References

[1] Wei, J., X. Wang, D. Schuurmans, et al. Chain-of-thought prompting elicits reasoning in large language models.

[2] Chen, W., X. Ma, X. Wang, et al. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks.

[3] Yue, X., X. Qu, G. Zhang, et al. Mammoth: Building math generalist models through hybrid instruction tuning.

[4] Hendrycks, D., C. Burns, S. Kadavath, et al. Measuring mathematical problem solving with the math dataset.

[5] Cobbe, K., V. Kosaraju, M. Bavarian, et al. Training verifiers to solve math word problems.

[6] Yu, L., W. Jiang, H. Shi, et al. Metamath: Bootstrap your own mathematical questions for large language models.

[7] Bao, G., Z. Teng, Y. Zhang. Token-level fitting issues of seq2seq models. In B. Can, M. Mozes, S. Cahyawijaya, N. Saphra, N. Kassner, S. Ravfogel, A. Ravichander, C. Zhao, I. Augenstein, A. Rogers, K. Cho, E. Grefenstette, L. Voita, eds., *Proceedings of the 8th Workshop on Representation Learning for NLP (RepL4NLP 2023)*, pages 276–288. Association for Computational Linguistics.

[8] Brown, T., B. Mann, N. Ryder, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, vol. 33, pages 1877–1901. Curran Associates, Inc.

[9] Azerbayev, Z., H. Schoelkopf, K. Paster, et al. Llemma: An open language model for mathematics.

[10] Li, G., H. A. A. K. Hammoud, H. Itani, et al. Camel: Communicative agents for "mind" exploration of large language model society.

[11] Kingma, D. P., J. Ba. Adam: A method for stochastic optimization.

[12] Rasley, J., S. Rajbhandari, O. Ruwase, et al. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, pages 3505–3506. Association for Computing Machinery.

[13] Dao, T. Flashattention-2: Faster attention with better parallelism and work partitioning.

[14] Ling, W., D. Yogatama, C. Dyer, et al. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In R. Barzilay, M.-Y. Kan, eds., *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167. Association for Computational Linguistics.

[15] Mishra, S., A. Mitra, N. Varshney, et al. Numglue: A suite of fundamental yet challenging mathematical reasoning tasks. In S. Muresan, P. Nakov, A. Villavicencio, eds., *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3505–3523. Association for Computational Linguistics.

[16] Hendrycks, D., C. Burns, S. Basart, et al. Measuring massive multitask language understanding.

[17] Davies, A., P. Veličković, L. Buesing, et al. Advancing mathematics by guiding human intuition with ai. 600(7887):70–74.

[18] Zhong, W., R. Cui, Y. Guo, et al. Agieval: A human-centric benchmark for evaluating foundation models.

[19] Patel, A., S. Bhattamishra, N. Goyal. Are nlp models really able to solve simple math word problems? In K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, Y. Zhou, eds., *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094. Association for Computational Linguistics.