

文章编号: 1003-0077(2024)01-0074-12

InstructGPT 在命名实体识别任务中的表现和挑战

孙瑜¹, 颜航¹, 邱锡鹏¹, 王定², 牟小峰², 黄萱菁¹

(1. 复旦大学计算机科学技术学院, 上海 200433; 2. 美的集团 AI 创新中心, 广东 佛山 528311)

摘要: 当前, 关于大规模语言模型, 例如, InstructGPT 的研究主要聚焦在自由形式生成任务上, 而忽略了在结构化抽取任务上的探索。为了让未来的工作在结构化抽取任务上有一个全面的认知, 该文在零样本和少样本设定下, 全面分析了 InstructGPT 在基础的结构化抽取任务, 命名实体识别上的表现。为了让结论更加可靠, 该文的实验同时包含了生物医学领域和通用领域的常规和嵌套数据集。实验结果表明, InstructGPT 在零样本学习上的性能只能达到微调小规模语言模型的 11%~56%, 增加少量样本也最多只能提升至 72%。为了探究 InstructGPT 在命名实体识别上表现性能不佳的原因, 该文通过分析模型的输出, 发现接近 50% 的句子都存在无效生成的问题。另外, 由于无效生成会同时导致“虚假错误预测”和“虚假正确预测”, 解决生成无效问题并不能保证性能的提升。此外, InstructGPT 抽取嵌套实体的能力还有待提高, 抽取嵌套实体的比例也偏低。因此, 用 InstructGPT 解决命名实体识别任务, 除了要保证生成的有效性, 还需要更加深入地研究才能找到行之有效的方法。

关键词: 大规模语言模型; 命名实体识别; 上下文学习; 思维链

中图分类号: TP391

文献标识码: A

Performance and Challenges of InstructGPT in Named Entity Recognition

SUN Yu¹, YAN Hang¹, QIU Xipeng¹, WANG Ding², MU Xiaofeng², HUANG Xuanjing¹

(1. School of Computer Science, Fudan University, Shanghai 200433, China;

2. Midea AI Innovation Center, Foshan, Guangdong 528311, China)

Abstract: Currently, the research on Large Language Models (LLMs), such as InstructGPT, is primarily focused on free-form generation tasks, while the exploration in structured extraction tasks has been overlooked. In order to gain a deep understanding of LLMs on structured extraction tasks, this paper analyzes InstructGPT's performance on named entity recognition (NER), one of the fundamental structured extraction tasks, in both zero-shot and few-shot settings. To ensure the reliability of the findings, the experiments cover common and nested datasets from both biomedical domain and general domain. The results demonstrate that InstructGPT's performance on zero-shot NER achieves 11% to 56% of the performance by a finetuned small-scaled model. To explore why InstructGPT struggles with NER, this paper examines the outputs, finding invalid generation for 50% of them. Besides, the occurrence of both "false-negative" and "false-positive" predictions makes it difficult to improve performance by only addressing the invalid generation. Therefore, in addition to ensuring the validity of generated outputs, further research still should focus on finding effective ways of using InstructGPT in this area.

Keywords: large language model; named entity recognition; in-context learning; chain-of-thought

0 引言

大规模语言模型 (Large Language Models, LLMs), 如 GPT-3 系列^[1-3], 由于其不依赖下游数据

微调即可取得较好的性能表现^[4-5], 在自然语言处理 (Natural Language Processing, NLP) 社区引起了热烈讨论。具体而言, 大规模语言模型只需要使用适当的提示 (Prompt), 就可以在零样本 (Zero-shot) 设定下取得不错的性能^[4, 6]; 对于少样本 (few-

收稿日期: 2023-05-11 定稿日期: 2023-11-14

基金项目: 国家自然科学基金 (62022027); 国家重点研发计划 (2022CSJG0801)

shot)问题,大规模语言模型可以利用上下文学习(In-context Learning, ICL)^[1,7]或者思维链(Chain-of-Thought, COT)^[8]提示来提高性能。

尽管过去的文献[1,5]已经证明了大规模语言模型在自由生成任务和多项选择任务中表现十分出色,但他们在结构化抽取任务,如命名实体识别(Named Entity Recognition, NER)中仍然面临挑战^[9]。目前,没有工作对大规模语言模型在命名实体识别上的能力进行全面的,因此,本文设计了一系列实验,深入研究了 InstructGPT 在命名实体识别任务上的表现及面临的挑战。具体地,本文主要回答了下面两个问题:

(1) InstructGPT 在不同设定和不同领域上表现得怎么样? 为了回答这个问题,本文分别在生物医学领域和通用领域上,进行了零样本学习和少样本学习。其中,通用领域在中英文的常规数据集和英文的嵌套数据集上都进行了实验。为了增加可信

度,本文沿用了之前工作^[5]中采用的提示作为基准来解决零样本的命名实体识别任务。除此之外,由于抽取实体本质上是结构化抽取任务,模型的输出需要满足一定的格式才能从中成功地解析出实体。为了实现这个要求,本文设计了一个结构化提示,从而指导 InstructGPT 生成格式化的文本。零样本场景下两种提示的例子如图 1 所示。在少样本设定下,本文探索了上下文学习和思维链两种方法。上下文学习时,本文沿用了之前广泛使用的提示方法^[9]。目前的思维链方法主要在推理任务中使用,本文是第一个将该方法引入到命名实体识别任务中的。少样本场景下的两种提示的例子如图 2 所示。结果显示,InstructGPT 无论在生物医学领域还是通用领域的数据上的表现,都和微调模型的表现有巨大的差距。其中,在零样本命名实体识别任务上,InstructGPT 仅能达到微调模型性能的 11%到 56%,在少样本命名实体识别上,InstructGPT 的性能最多达到 72%。

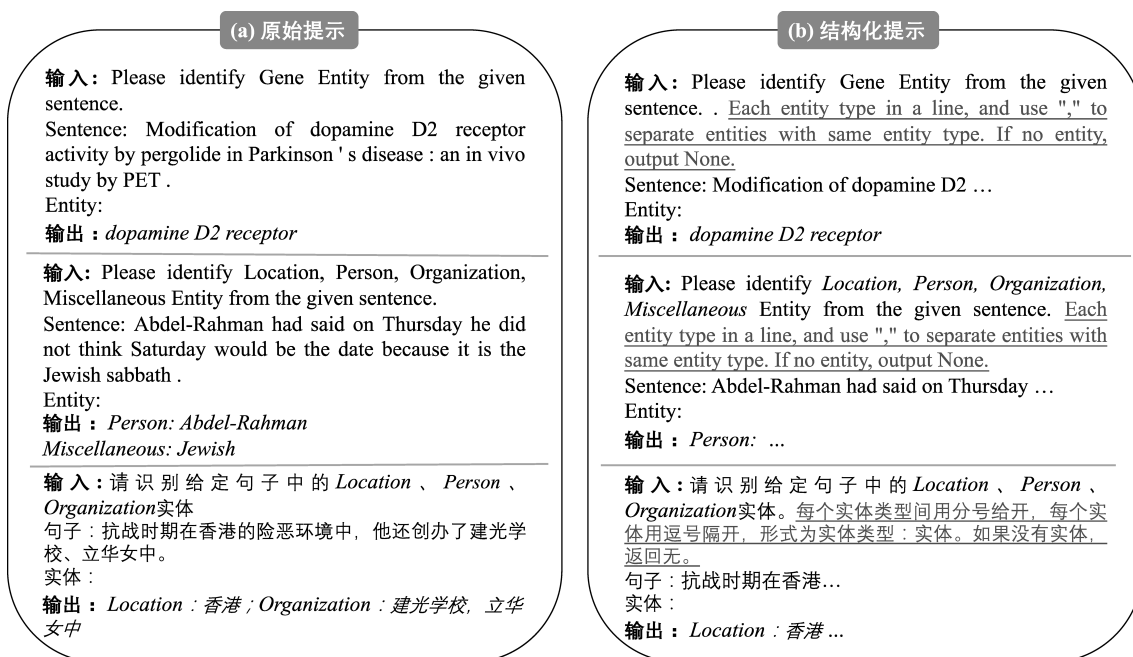


图 1 零样本命名实体识别提示的例子

(2) 为什么 InstructGPT 在命名实体识别任务上表现得不理想? InstructGPT 在很多生成式的任务上的表现与微调模型性能相近,甚至更好。然而, InstructGPT 却在命名实体识别上没有达到预期效果。为了探究阻碍 InstructGPT 成功抽取实体的原因,本文从两个方面来仔细评估了模型的输出: 无效生成和虚假预测。无效生成包括输出无法成功匹配解析格式的“格式无效”,和输出的实体片段不在

输入句子中的“片段无效”。结果表明,10%~70%的输出句子存在无效生成的问题,严重影响了解析过程。零样本场景给出更详细的指令或者少样本场景增加示例的个数可以有效减少“格式无效”。然而,“片段无效”仍是一个亟需解决的问题。此外,无效性可能会直接导致生成的句子解析失败,从而不能识别预测成功的实体。本文称这种情况为虚假预测。虚假预测除了存在导致性能偏低的“虚假错误

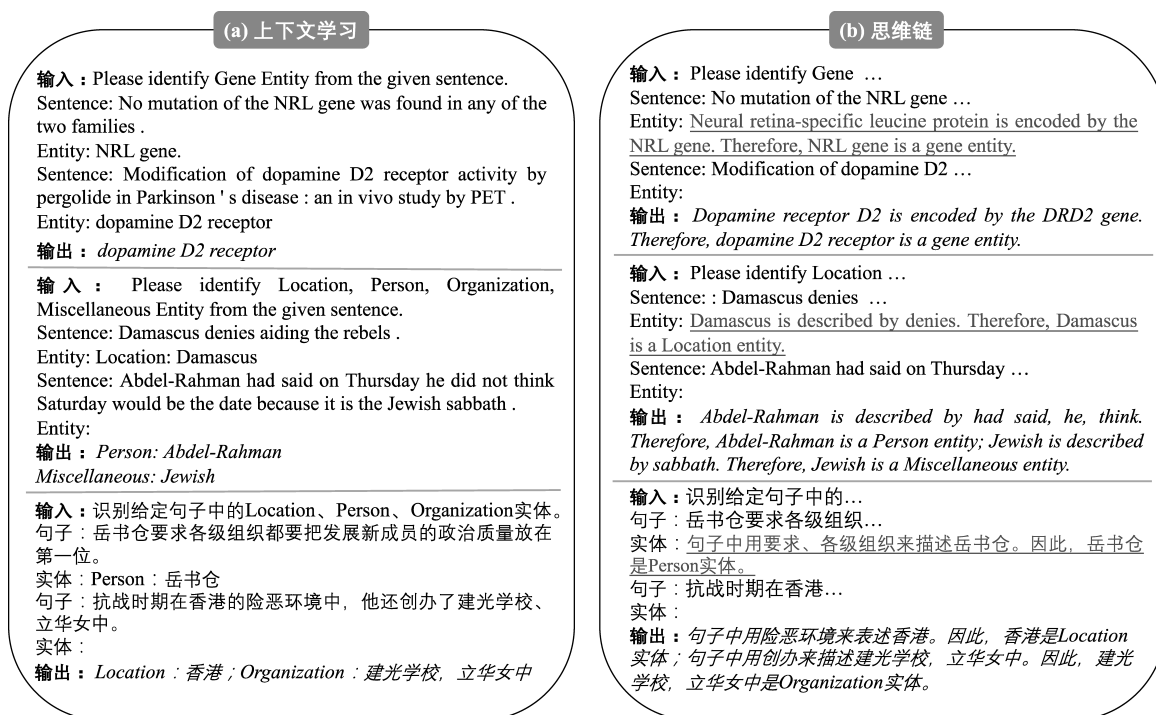


图2 少样本命名实体识别提示的例子

预测”，还存在导致性能偏高的“虚假正确预测”。当无效性问题严重时，“虚假正确预测”的数量甚至增加到40%。未来的工作在保证预测正确性的前提下，仍需从根本上提升 InstructGPT 等大模型抽取实体的能力。除此之外，针对嵌套命名实体识别任务，本文还统计了 InstructGPT 生成的实体中嵌套实体的比例，即实体嵌套率，发现其远低于数据集中的比例。并且，抽取嵌套实体的正确率也不足一半。这说明大模型在处理嵌套实体方面还是面临挑战的。

总的来说，本文的贡献主要体现在以下四个方面：

(1) 本文首次详细探究了 InstructGPT 在命名实体识别任务上的表现，覆盖了生物医学领域和通用领域、常规场景和嵌套场景的数据，并在零样本和少样本设定上都进行了实验。本文提供的实验结果可以作为之后研究的基准。

(2) 首次提出将思维链应用到命名实体识别任务上，并为生物医学领域和通用领域的数据设计了不同形式的思维链，并对其性能进行了测试。

(3) 本文从两个指标出发，对 InstructGPT 的输出进行了错误分析，发现了输出的无效性是阻碍

InstructGPT 性能提升的重要因素。通过分析无效生成问题导致的虚假预测问题，本文发现，解决无效生成问题并不一定会提升 InstructGPT 在命名实体识别上的性能。未来工作可以针对本文列举的无效生成类型，首先保证模型生成的正确性，其次设计更好的抽取实体的方法。

(4) 首次针对性地研究了 InstructGPT 在嵌套命名实体识别任务上的表现，发现 InstructGPT 的实体嵌套率和嵌套正确率较低。通过增加上下文示例的数量可以一定程度上增加 InstructGPT 生成嵌套实体的比例和识别的正确率。

1 任务定义

命名实体识别作为一个基本的结构化抽取任务，旨在从原始文本中抽取实体片段，并将它们分类到预定义的类型中。例如，“复旦大学是上海市的一所综合性大学。”中，“复旦大学”是组织类别的实体，“上海市”是地点类别的实体。本文为了提升效率，InstructGPT 需要一次性从句子中抽取出所有类型的实体片段并分类。即，上面句子的输出应该为“组织：复旦大学；地点：上海市”。最后，模型的原始输出被解析成结构化的实体，从而进行评测。

2 提示的设计

为了充分研究 InstructGPT 在命名实体识别任务中的表现,本节详细描述了在不同设定和领域中提示的设计。

2.1 零样本命名实体识别

零样本学习时,本文使用的提示由 3 个部分构成:任务指令、候选句子、指示词。如图 1 所示,本文为零样本命名实体识别设计了两种提示。图 1 中从上到下的示例分别来自生物医学领域、英文通用领域和中文通用领域。

2.1.1 原始提示

作为基准,本文沿用了之前工作的提示,其中任务指令是关于命名实体任务的描述,并给出需要抽取的实体类别。在抽取可能包含嵌套实体的句子的时候,本文在任务指令后面添加了“Notice that there may exist nested entities.”。候选句子由“Sentence:”/“句子:”拼接待抽取的句子。指示符是“Entity:”/“实体:”。原始提示的示例如图 1(a)所示。

2.1.2 结构化提示

正如第 1 章所讨论的,InstructGPT 在其生成格式方面展现出了相当大的灵活性。因此,设计提示来指导 InstructGPT 生成理想的格式化输出对于实现有效的解析至关重要。因此,本文修改了任务指令部分,添加了输出格式的详细指令。中英文关于输出格式的描述有所不同。其余两个部分与原始提示一致。如图 1(b)所示,结构化指令部分用下划线标出。

2.2 少样本命名实体识别

少样本学习时,本文使用的提示由四个部分构成:任务指令、示范(Demonstration)、候选句子、指示词。除了新增的示范部分,其他部分都与零样本学习的原始提示相同。如图 2 所示,本文使用了两种常见的大模型少样本学习方法,即上下文学习和思维链。不同方法的改变只存在于示范部分。图 2 中从上到下的示例分别来自生物医学领域、英文通用领域和中文通用领域,并只拼接了一个示范作为例子。

2.2.1 上下文学习

受到之前工作的启发^[5],本文设计的示范部分由几个完整训练示例组成,即训练示例构成的候选句子、指示词以及模型输出。其中训练示例的输出应与最终期望的模型输出具有相同的格式。图 2(a)给出了三种情况下的上下文学习输入输出的例子。

2.2.2 思维链

与上下文学习不同,思维链^[8]的示范部分由几个推理链组成。其中每个推理链都包括了一系列中间推理步骤,即原理(Rationale)和答案。本文是第一个将思维链提示应用到命名实体识别上的研究,并为不同领域的数据设计了不同的推理链。如图 2(b)所示,推理链部分用下划线标出。

对于生物医学领域,本文首先从维基百科检索出实体的条目页面,并选择定义句子(通常在文档开头)作为推理链。如果该实体没有条目,则随机选择一个包含该实体的句子。之后,在推理链后拼接“Therefore”开头的结论作为答案部分。对于通用领域,本文受到了 triggerNER^[10]的启发。triggerNER 是通过选择特定的单词和短语作为解释性标注来识别实体。对于 CoNLL03 数据集,本文直接使用 triggerNER 的标注;对于其他通用领域数据集,本文使用该技术原理,人工标了解释性单词和短语。之后,将这些解释性单词和短语用自然语言描述,作为推理链;用“Therefore/因此”开头的结论作为答案。

3 实验

本章节首先介绍了本文使用的 9 个数据集,然后展示并分析讨论了实验结果,来分别回答前面提出的两个问题。

3.1 数据集

本文在三个生物医学领域、四个常规通用领域和两个嵌套通用领域的命名实体识别数据集上进行了实验。沿用过去的工作,本文只在 1 000 条测试数据上评估 InstructGPT 的表现,如果测试集数据少于 1 000 条,则使用全部测试集。每个实验使用相同部分的子数据集。

3.1.1 生物医学领域数据集

本文选择了 BLURB 基准数据集^[11]中的三个数

数据集。其中生物创新 V 化学-疾病关系语料库 (Bio-Creative V Chemical-Disease Relation corpus)^[12] 包含了 PubMed 摘要中标注为疾病 (BC5CDR-disease) 和化学 (BC5CDR-chem) 实体的句子, 生物创造 II 基因提及 (Biocreative II Gene Mention, BC2GM)^[13] 数据集则包含了 PubMed 摘要中的基因标注。

3.1.2 通用领域数据集

本文选择了四个常用的通用领域的常规命名实体识别数据集, 其中, 两个是英文, 两个是中文。对于英文数据集, 本文使用了 CoNLL03^[14] 数据集和 OntoNotes5^[15] 数据集, 其中 OntoNotes5 数据集本文只保留了非数字的实体类别; 对于中文数据集, 本文使用了 MSRA^[16] 数据集和 OntoNotes4^[17] 数据集。

除此之外, 本文还选择了两个常用的英文嵌套命名实体识别数据集, 分别是 ACE04^[18] 和 ACE05^[19]。

3.2 结果: InstructGPT 在不同设定和不同领域上表现得怎么样?

本节的实验旨在探究 InstructGPT 在低资源场景下解决命名实体识别任务的能力。表 1 展示了生物医学领域数据集的实验结果, 表 2 和表 3 分别展示了通用领域常规实体和嵌套实体数据集的实验结果。少样本学习时, 本文分别在拼接 1 个、5 个和 10 个示例的情况下测试。每个结果都是 5 次实验结果的平均值, 下标表示标准差 (例如, 76.8_{11} 表示 76.8 ± 1.1)。

表 1 生物医学领域数据的主要结果

(单位: %)

	BC5CDR-disease			BC5CDR-chem			BC2GM		
	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1
BioBERT	微调小规模语言模型								
	86.5	87.8	87.2	93.7	93.3	93.5	84.3	85.1	84.7
原始提示 结构化提示 结构化提示	零样本学习								
	53.1 ₅	29.4 ₂	37.8 ₁	59.7 ₉	55.1 ₁₂	57.3 ₁₀	30.4 ₉	26.3 ₉	28.2 ₉
	37.5 ₆	35.4 ₄	36.4 ₅	42.0 ₄	62.5 ₁₀	50.3 ₅	29.8 ₄	41.2 ₃	34.6 ₄
k=1 k=5 k=10	上下文学习 (k 为示例数)								
	42.5 ₂₅	32.9 ₇	37.1 ₆	58.5 ₁₉	59.8 ₁₇	59.1 ₄	34.5 ₂₀	26.2 ₁₀	29.8 ₈
	56.4 ₂₂	41.0 ₅	47.4 ₉	72.3 ₃₅	65.7 ₇	68.8 ₁₈	37.1 ₂₁	31.1 ₁₁	33.8 ₁₂
	58.1 ₂₁	46.6₃	51.7 ₈	72.5 ₉	71.2₁₆	71.8 ₇	35.2 ₁₀	31.9 ₁₂	33.5 ₁₁
k=1 k=5 k=10	思维链 (k 为示例数)								
	50.8 ₃₂	7.0 ₁₃	12.3 ₁₉	83.9 ₂₃	21.5 ₂₈	34.1 ₃₆	38.4 ₆₅	2.2 ₇	4.1 ₁₂
	71.4 ₃₄	41.5 ₂₅	52.3 ₁₃	90.3 ₈	58.8 ₃₇	71.2 ₂₆	52.5 ₁₄	24.9 ₁₇	33.7 ₁₄
	76.8₁₁	43.9 ₁₀	55.9₈	93.0₄	62.1 ₁₀	74.5₇	56.1₁₅	37.5₁₄	44.9₁₄

表 2 通用领域数据的主要结果

(单位: %)

	CoNLL03			OntoNotes5			MSRA			OntoNotes4		
	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1
RoBERTa	微调小规模语言模型											
	92.3	94.0	93.5	90.3	92.1	91.2	95.8	95.8	95.8	82.8	81.3	82.0

续表												
	CoNLL03			OntoNotes5			MSRA			OntoNotes4		
	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1
原始结构化	零样本学习											
	33.4 ₂₃	29.5 ₁₅	31.2 ₇	13.4 ₈₀	8.4 ₅₈	10.3 ₆₇	37.8 ₂₀	53.3 ₁₁	44.2 ₁₇	31.2 ₂₄	41.2 ₄₅	35.5 ₃₂
	70.2₃₇	41.8 ₁₇	52.4 ₂₁	36.9 ₇₇	8.7 ₂₁	18.2 ₂₈	46.4 ₁₆	45.3 ₂₆	47.7 ₂₃	37.0 ₂₄	38.3 ₃₃	37.6 ₂₇
k=1 k=5 k=10	上下文学习(k 为示例数)											
	50.7 ₃₁	44.9 ₂₆	47.5 ₁₇	18.9 ₁₆	15.1 ₂₇	16.7 ₂₃	44.9 ₉	50.1 ₉	47.3 ₄	37.4 ₂₅	41.3 ₃₄	39.2 ₂₈
	62.2 ₄₀	50.2 ₁₃	55.5 ₂₁	32.9 ₂₅	26.3 ₂₃	29.2 ₂₃	59.3 ₁₅	51.9 ₁₄	55.3 ₃	48.9 ₂₅	46.1 ₃₅	47.4 ₂₆
	67.5 ₁₇	57.5 ₁₁	62.1 ₁₀	39.4₉	31.1₇	34.7₉	63.6₁₁	58.0₉	60.6₈	53.7₆	48.0 ₂₃	50.7 ₂₄
k=1 k=5 k=10	思维链(k 为示例数)											
	45.1 ₂₀	30.0 ₃₇	36.0 ₂₉	21.6 ₁₃	16.5 ₁₇	18.6 ₁₁	43.9 ₂₄	19.3 ₁₈	26.8 ₂₀	34.3 ₇	11.0 ₂₅	16.5 ₂₈
	55.0 ₁₀	70.4 ₁₄	61.8 ₁₀	30.1 ₂₉	31.0 ₂₂	30.5 ₂₅	55.2 ₁₉	39.0 ₈	45.7 ₈	45.8 ₃₄	39.1 ₂₉	42.1 ₃₁
	60.4 ₁₃	75.5₁₇	67.2₁₉	39.1 ₃₁	28.2 ₂₇	32.8 ₂₈	55.8 ₃₁	39.6 ₁₃	46.3 ₁₈	53.0 ₃	48.9₂₈	50.9₃₅

表 3 通用领域嵌套实体的主要结果
(单位：%)

	ACE04			ACE05		
	Pre.	Rec.	F_1	Pre.	Rec.	F_1
BERT	微调小规模语言模型					
	—	—	84.4	—	—	84.3
原始结构化	零样本学习					
	53.0 ₁₂	15.6 ₄	24.2 ₆	60.3 ₁₄	9.7 ₀	16.8 _{1.2}
	47.2 ₇	11.8 ₃	18.9 ₈	55.7 ₁₇	7.1 ₄	12.6 ₇
k=1 k=5 k=10	上下文学习(k 为示例数)					
	43.1 ₁₈	16.9 ₄	24.3 ₇	42.9 ₄	15.7 ₇	23.0 ₈
	49.1 ₁₁	23.8 ₇	32.0 ₇	47.4 ₁₆	23.8 ₁₈	31.7 ₁₉
	52.9₁₃	27.6 ₅	36.3 ₅	50.5 ₁₂	26.6 ₁₀	34.8 ₁₁
k=1 k=5 k=10	思维链(k 为示例数)					
	34.0 ₁₆	19.9 ₃₂	24.9 ₂₃	31.4 ₁₅	15.8 ₂₂	21.0 ₂₂
	42.8 ₇	30.0 ₁₉	35.2 ₁₅	44.0 ₈	27.5 ₁₅	33.8 ₁₂
	48.3 ₈	32.7₇	39.0₇	51.5₁₀	30.0₂	37.8₃

3.2.1 生物学领域

在生物学领域,本文对比了微调 BioBERT^[20] 预训练模型的结果。零样本学习时,InstructGPT 在生物学领域数据集上能达到 41%~61%微调模型的表现。本文发现结构化提示对零样本学习似乎没有影响,甚至会对疾病和化学类别实体的提取产生负面影响。

在少样本学习时,InstructGPT 在少样本的生物学领域的数据上的表现达到了 51%~80%微调模型的能力。增加示例数量(从 1 个到 10 个)在上下文学习和思维链方法中都有益于 F_1 的提高,分别平均有 +10.3 和 +41.6 的增长。但是,仅使用一个示例时,思维链方法的 F_1 相较于上下文学习平均下降了 25.2。这是由于思维链方法的生成中,原理部分的结构比较复杂,InstructGPT 难以保证有效输出,导致了严重的解析失败问题,从而表现不佳。一旦示例数量增加,InstructGPT 在生物学领域数据集上使用思维链的性能优于只使用上下文学习。

3.2.2 通用领域

在通用领域的常规命名实体识别数据集上,本文对比了微调 RoBERTa^[21] 预训练模型的结果。在通用领域的嵌套命名实体识别数据集上,本文对比了微调 BERT^[22] 预训练模型的结果。由于该篇工作没有汇报完整的实验结果,因此我们也只在表格中展示了 F_1 。在零样本常规实体的抽取上,InstructGPT 大概能达到 20%~56%微调模型的性能;在嵌套命名实体识别数据上大概能达到 20%~29%微调模型的性能。另外,本文发现,在常规实体的抽取上使用结构化提示可以显著提高性能, F_1 平均提高了 8.7。这意味着通过修改提示指导 InstructGPT 生成更加格式化的输出对于通用领域实体识别是有效的。嵌套实体抽取中使用结构化提示反而一定程度上损害了性能。

在少样本学习时,随着上下文示例数量的增加,

在常规命名实体识别任务上,上下文学习和思维链的性能分别平均提高了 14.4 和 24.8;在嵌套命名实体识别任务上,则分别提高了 11.9 和 15.5。对于通用领域常规实体的提取,实验发现思维链在常规实体抽取上的用处不大。其原因可能是实体触发词与实体本身之间没有强烈的因果关系,导致不能激发模型潜在的推理能力。并且由于思维链的输出结构更加复杂,反而会一定程度上损害模型的表现。值得注意的是,在每个设置中,OntoNotes5 的性能明显低于其他数据集。这是由于 OntoNotes5 有 11 个类别,远远多于其他数据集的 3、4 个类别。InstructGPT 在解决很多类别的分类时,也面临了巨大的挑战。思维链方法运用在嵌套实体中可以一定程度上提升性能,这可能是由于 ACE04 和 ACE05 数据集中有很多指代词被标注为实体。思维链的使用帮助 InstructGPT 利用上下文的逻辑来抽取实体。

总体而言,与微调模型相比,实验发现 InstructGPT 在每个命名实体识别数据集上的表现都不尽人意。具体来说,InstructGPT 只能通过零样本提示实现 11%~56% 的性能。即使添加了示例样本,InstructGPT 的 F_1 最多也只能达到 72%。通过之前的观察,本文建议在解决通用领域的命名实体识别问题时,可以使用结构化提示来提升最终的性能。面对少样本问题时,思维链提示在生物医学领域更有可能表现良好;而对于通用领域的少样本学习,上下文学习足以胜任。

3.3 结果:为什么 InstructGPT 在命名实体识别任务上表现得不理想?

正如前文所述,尽管在许多设定下进行了尝试,InstructGPT 的结果与微调模型之间仍存在显著差距。本节对输出文本进行了详细的分析,发现了导致结果不理想的一个重要原因是无效生成。这种无效性问题还会导致虚假预测问题,使得目前获得的结果不完全准确。在嵌套命名实体识别时,本文发现 InstructGPT 的嵌套实体识别率偏低,说明在识别嵌套的实体上还面临挑战。

3.3.1 无效生成

由于命名实体识别任务本质上是一个结构化抽取任务,因此最终目的是得到结构化的实体。要将 InstructGPT 输出的非结构化文本解析成结构化的实体,其生成的输出必须是格式化的,才能成功地匹配解析。基于这个特性,本文引入了“格式无效”这

一指标,表示模型输出的格式与期望格式不符。主要体现在解析阶段无法区分不同实体类型之间的实体,或者无法区分类别词和实体。例如,图 3 中的(1)就是“格式无效”的一个例子。此外,命名实体识别任务要求抽取的实体片段必须完全属于原始句子。由于实体有各种表达方式,InstructGPT 可能会生成该实体的其他形式,从而导致匹配解析失败。此外,单数和复数的使用不一致也会导致此问题。本文为此定义了“片段无效”指标来记录不符合该要求的实体片段的数量。例如,图 3 中(2)就是“片段无效”的一个例子。



图 3 无效生成的例子

除此之外,本文还引入了“无效句子”,来记录具有任何以上两种类型无效性的句子的数量。三种无效性的结果如图 4 所示。在每个子图中,“格式无效”和“片段无效”的单位都是片段数量,基于左边的 Y 轴,而“无效句子”的单位是句子数量,基于右边的 Y 轴。值得注意的是,生物医学领域数据集的“格式无效”在零样本学习和上下文学习时始终为 0。这是由于生物医学领域的数据集都只有一个实体类型,解析过程中不需要格式匹配来区分不同实体类型。

总的来说,无效生成问题有三个值得注意的结论:(1) 在零样本学习时添加结构化提示或者在少样本学习时增加示例数量都有助于提高输出格式的正确性。从图 3 中可以看出,增加结构化提示后,“格式无效”平均减少了 34 个;将示例数量从 1 个增加到 10 个,“格式无效”在上下文学习中平均减少了 42 个,在思维链上平均减少了 536 个。(2) 使用思维链方法时,如果只拼接一个示例,InstructGPT 很难模仿复杂的推理过程。如图 4 所示,每个数据集在只使用一个示例的思维链方法中,都有相当多的“格式无效”,达到了 600 到 900 个片段。增加示例的数量可以显著缓解这个问题,如可以将无效的片段数降低到个位数。(3) “片段无效”问题更具挑战性,并且使用现有的大模型学习方法难以解决。通

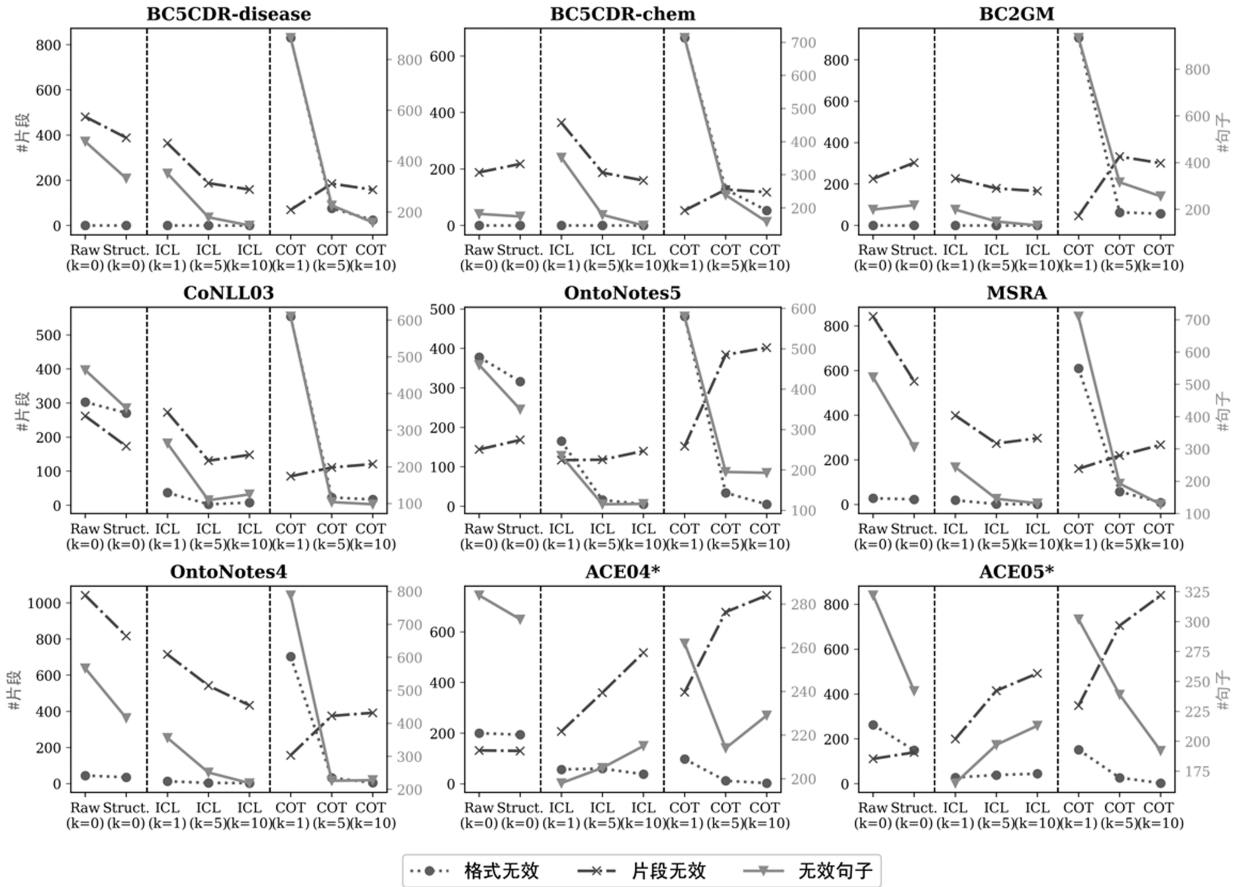


图4 无效生成的统计数据图

过提示或者示例引导 InstructGPT 生成格式化的输出,几乎可以将“格式无效”降至零;但几乎每个数据集的每个场景中,“片段无效”都超过了200个。InstructGPT 在中文数据集上的“片段无效”问题尤为严重。

3.3.2 虚假预测

上一节描述的无效生成问题可能会导致 InstructGPT 的输出面临解析失败的问题。解析失败会导致预测正确的实体没有被解析出来,即判别为预测错误的句子实际是正确的,使最终性能偏低,本文称之为“虚假错误预测”。直觉上,设计更好的提示或者更好的解析方法可以缓解无效生成的问题,进而减少“虚假错误预测”,从而得到更好的性能。然而,通过收集 InstructGPT 存在无效生成的句子,本文发现,解析失败还会使一些预测错误的实体躲过与目标实体匹配的过程,使得预测错误的句子被错误判别为正确预测,使最终性能偏高,本文称之为“虚假正确预测”。本文将这两种情况统称为虚假预测。

存在“虚假错误预测”的句子,一旦模型的输出

被更好地解析,就可能纠正这些虚假错误,性能就可以进一步提高。从表2、表3和图4中可以看出,零样本中增加结构化提示或者少样本中增加样本数量时,“格式无效”的数量减少, F_1 也相应地有所提升。

“虚假正确预测”的统计结果见图5,每个子图的单位都是句子数量,每个柱体的高度是正确预测的数量,包括“真实正确”(实心部分)和“虚假正确”(斜线部分)。如果 InstructGPT 预测正确并且输出的句子不存在无效生成问题,则表示该测试用例是真实正确的。“真实正确”部分表明,添加结构化提示比仅使用一个示例更有用,同时添加的示例越多,真实正确的数量也越多,表示模型能力越好。如果 InstructGPT 预测正确但是输出句子存在无效生成,则认为该测试用例是虚假正确的。实验表明,原始提示和使用一个样本的思维链方法均面临了严重的“虚假正确”问题。正如3.3.1节描述的,这两种情形下,InstructGPT 的输出存在很多“格式无效”问题。这个问题可以通过添加结构化提示和增加示例数量缓解。

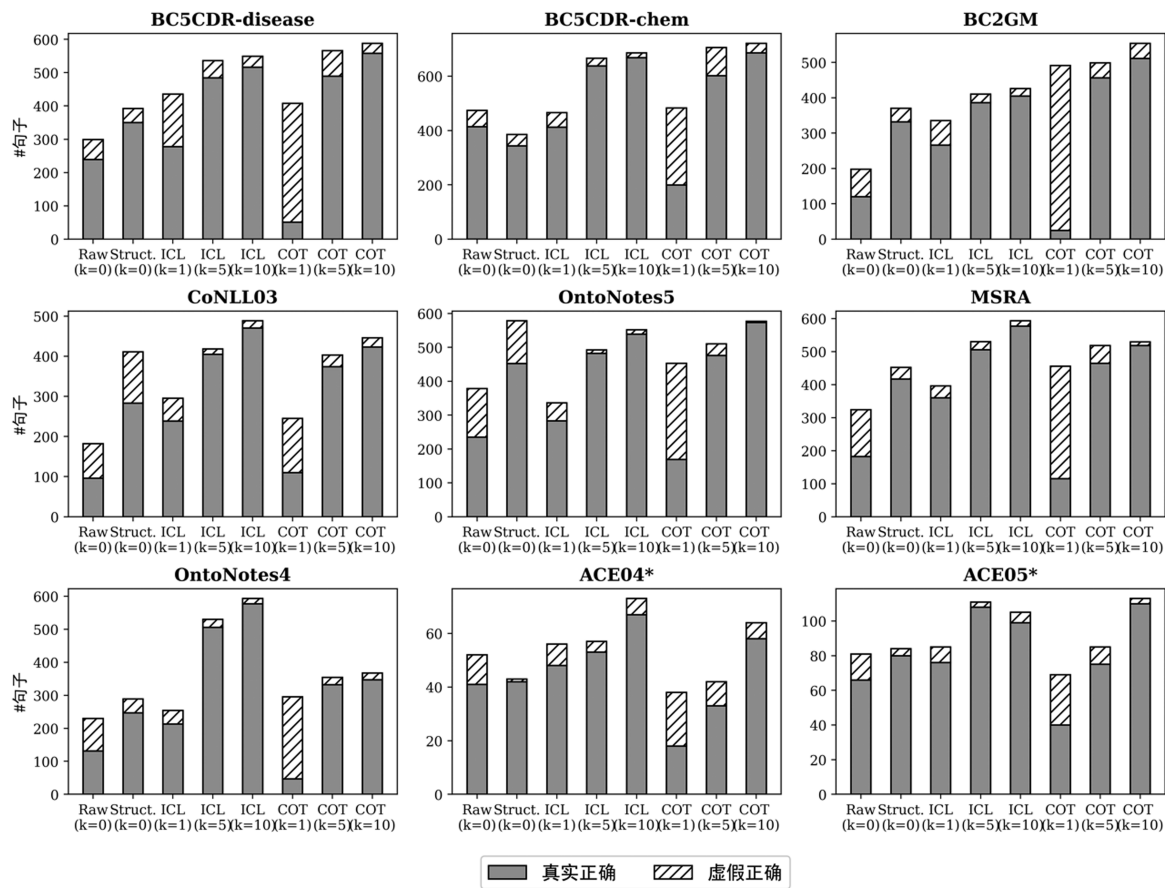


图 5 虚假正确预测的统计数据

总之解决 InstructGPT 生成的无效性并不能保证最终结果的提高。如果解决的“虚假错误预测”问题多于“虚假正确问题”，则最终性能会提升；反之，性能会下降。因此，为了从根本上提升 InstructGPT 在命名实体识别任务上的表现，未来工作首先应该解决虚假预测的问题，保证最终的性能是真实的。阻碍性能真实性的主要问题之一是无效生成问题。减少输出格式与期望格式不匹配的“格式无效”问题的一个主要方法是使用结构化提示或添加更多示例。如何确保模型输出的片段有效，仍是 InstructGPT 在这种结构化抽取任务上的瓶颈。未来的工作可以

更多地关注提升大规模语言模型在片段定位上的能力。

3.3.3 嵌套实体识别率

在嵌套命名实体识别场景下，实体之间可能存在嵌套关系，即一个实体包含了其他实体。本文统计了各个实验设定下，InstructGPT 抽取的嵌套实体数、抽取正确的实体数和总的实体数。为了公平对比，还计算了嵌套正确率=抽取正确的嵌套实体数量/嵌套实体数量，实体嵌套率=嵌套实体数量/总的实体数量。表 4 对比了上述统计数据与测试集中实际的数量。

表 4 嵌套实体识别率统计

	ACE04					ACE05				
	# 正确 嵌套	# 所有 嵌套	# 所有 实体	嵌套 正确率/%	实体 嵌套率/%	# 正确 嵌套	# 所有 嵌套	# 所有 实体	嵌套 正确率/%	实体 嵌套率/%
测试集	—	411	3 031	—	13.6	93.5	344	3 025	84.7	11.4
原始提示 结构化	零样本学习									
	8	13	890	61.5	1.5	9	15	752	60	2.0
	2	8	507	25	1.6	22	26	410	84.6	6.3

续表

	ACE04					ACE05				
	# 正确 嵌套	# 所有 嵌套	# 所有 实体	嵌套 正确率/%	实体 嵌套率/%	# 正确 嵌套	# 所有 嵌套	# 所有 实体	嵌套 正确率/%	实体 嵌套率/%
上下文学习(k 为示例数)										
$k=1$	4	22	1 088	18.2	2.0	2	17	1 085	11.8	1.6
$k=5$	21	47	1 425	44.7	3.3	22	54	1 467	40.7	3.7
$k=10$	40	73	1 532	54.8	4.8	37	75	1 495	49.3	5.1
思维链(k 为示例数)										
$k=1$	8	24	1 502	33.3	1.6	5	15	1 848	33.3	0.8
$k=5$	19	41	2 064	46.3	2.0	16	38	1 918	42.1	2.0
$k=10$	15	30	2 114	50.0	1.4	13	25	1 769	52.0	1.4

本文发现, InstructGPT 生成的实体嵌套率远远低于实际的比例, 最多只能达到 5%, 这说明 InstructGPT 还是更倾向于预测常规的实体。而预测出来的嵌套实体中, 模型也通常只能做对不足 50%, 这说明定位到正确的嵌套实体位置对于 InstructGPT 来说还是一项重大的挑战。除此之外, 上下文学习的场景下, InstructGPT 倾向于生成更多的嵌套实体, 本文推测通过添加存在嵌套实体的示例, 可以挖掘模型识别嵌套实体的能力, 而这比用指令指导 InstructGPT 生成嵌套实体更有用。而思维链提示中复杂的推理句子, 让 InstructGPT 模糊了嵌套实体的位置, 更注重抽取需要推理的代词实体。另外, 增加上下文示例的数量可以一定程度上增加嵌套正确率。

4 相关工作

近些年, 大规模语言模型^[1,4,6,23]如雨后春笋般涌现。他们庞大的模型规模带来了强大的能力, 尤其是在零样本学习和少样本学习上展现了惊人的效果。在解决零样本任务时, 大规模语言模型仅需要适当的提示就能在很多任务上达到和微调模型同样的效果^[5]。在解决少样本任务时, 大规模语言模型仅通过上下文学习便表现出令人惊叹的泛化能力^[1]。一些先前的工作^[8,24]探索了思维链提示, 即利用推理链进一步激发大规模语言模型的推理能力。在他们之中, GPT-3 系列^[1-3]是最受关注的模型。在这个系列中, InstructGPT^[3]引入了来自人类反馈的强化学习 (Reinforcement Learning from Human Feedback, RLHF), 是第一个与用户对齐的模型。因此, 本文选择探究 InstructGPT 的能力。

使用大规模语言模型解决自然语言处理任务中的零样本学习和少样本学习时, 主要有两种方法。其一, 一些工作^[25-27]使用大规模语言模型来标注没有标签的数据, 并使用这些伪标签对小规模预训练语言模型 (Pre-trained Language Models, PLMs) 进行微调。然而, 命名实体识别是一项词元级别的标注任务, InstructGPT 很难直接对句子中的实体片段进行标注。其二, 一些研究^[5,28]直接应用大规模语言模型进行推理。本文也采用这种方法直接评估模型的表现。之前也有一些工作^[9,28]评估了这种方式下命名实体识别任务的性能, 但他们只在特定领域进行了少样本的上下文学习。目前, 还没有工作全面地研究 InstructGPT 在各种领域的命名实体识别上各种设定下的性能和输出的分析。本文对命名实体识别的性能和面临的挑战进行了详细分析。

5 研究与展望

本文对 InstructGPT 在命名实体识别任务中的表现进行了全面研究, 实验的数据集覆盖了生物学领域和通用领域, 其中通用领域还考虑了中英文常规数据集和英文嵌套数据集。本文分别在零样本学习和少样本学习中各采用了两种提示, 来评估 InstructGPT 的表现。实验发现, InstructGPT 在命名实体识别中的表现并不理想, 离微调小模型还有很大差距。经过实验统计分析, 这主要是由于输出中存在“格式无效”和“片段无效”问题, 使得解析过程失败。为了减少无效性问题, 本文建议使用结构化提示或增加示例样例的数量来帮助 InstructGPT 克服“格式无效”的问题。除此之外, 解决“片段无

效”也是至关重要的,并且是未来使用大规模语言模型解决结构化抽取的重点研究方向。本文发现生成无效导致的虚假预测问题,不仅可能使结果偏低,也可能导致结果偏高。因此,未来工作仍需要从其他思路来进一步提升 InstructGPT 抽取实体的能力。对于更加复杂的嵌套实体来说,InstructGPT 也面临了很大的挑战。InstructGPT 的识别实体的嵌套率远低于数据集中本来的比例。因此,如何使得大模型识别更多的嵌套实体也是未来工作的一大难点。此外,本文是第一个在命名实体识别任务中实现思维链提示的研究。实验观察到,相比通用领域,思维链在生物医学领域中更有效。并且,使用思维链提示需要足够数量的示例以避免严重的“格式无效”问题。总之,本文探究了使用 InstructGPT 在命名实体识别任务中存在的限制和潜在的改进方法。希望本文的发现能为未来的大规模语言模型研究提供有价值的见解。

参考文献

- [1] BROWN T B, MANN B, RYDER N, et al. Language models are few-shot learners[J]. arXiv preprint arXiv:2005.14165, 2020.
- [2] CHEN M, TWOREK J, JUN H, et al. Evaluating large language models trained on code[J]. arXiv preprint arXiv:2107.03374, 2021.
- [3] OUYANG L, WU J, JIANG X, et al. Training language models to follow instructions with human feedback[J]. arXiv e-prints, 2022; arXiv: 2203.02155.
- [4] CHOWDHURY A, NARANG S, DEVLIN J, et al. Palm: Scaling language modeling with pathways[J]. arXiv preprint arXiv:2204.02311, 2022.
- [5] QIN C, ZHANG A, ZHANG Z, et al. IsChatGPT a general-purpose natural language processing task solver? [J]. arXiv preprint arXiv:2302.06476, 2023.
- [6] TOUVRON H, LAVRIL T, IZACARD G, et al. Llama: Open and efficient foundation language models [J]. arXiv preprint arXiv:2302.13971, 2023.
- [7] LIU J, SHEN D, ZHANG Y, et al. What makes good in-Context examples for GPT-3? [C]//Proceedings of Deep Learning Inside Out: The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures. 2022: 100-114.
- [8] WEI J, WANG X, SCHUURMANS D, et al. Chain of thought prompting elicits reasoning in large language models[J]. arXiv preprint arXiv:2201.11903, 2022.
- [9] GUTIÉRREZ B J, MCNEAL N, WASHINGTON C, et al. Thinking about gpt-3 in-context learning for biomedical ie? think again[J]. arXiv preprint arXiv:2203.08410, 2022.
- [10] LIN B Y, LEE D H, SHEN M, et al. TriggerNER: Learning with entity triggers as explanations for named entity recognition [C]//Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020: 8503-8511.
- [11] GU Y, TINN R, CHENG H, et al. Domain-specific language model pretraining for biomedical natural language processing[J]. ACM Transactions on Computing for Healthcare, 2021, 3(1): 1-23.
- [12] LI J, SUN Y, JOHNSON R J, et al. BioCreative V CDR task corpus: A resource for chemical disease relation extraction[J]. Database, 2016, 1: 10.
- [13] SMITH L, TANABE L K, KUO C J, et al. Overview of BioCreative II gene mention recognition[J]. Genome Biology, 2008, 9(2): 1-19.
- [14] SANG E T K, DEMEULDER F. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition[C]//Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL 2003: 142-147.
- [15] PRADHAN S, MOSCHITTI A, XUE N, et al. Towards robust linguistic analysis using ontonotes[C]//Proceedings of the 17th Conference on Computational Natural Language Learning. 2013: 143-152.
- [16] ZHANG S, QIN Y, HOU W J, et al. Word segmentation and named entity recognition for SIGHAN bake-off3[C]//Proceedings of the 5th SIGHAN Workshop on Chinese Language Processing. 2006: 158-161.
- [17] WEISCHEDEL R, PRADHAN S, RAMSHAW L, et al. Ontonotes release 4.0[C]//Proceedings of the LDC, Philadelphia, Penn.: Linguistic Data Consortium, 2011.
- [18] DODDINGTON G R, MITCHELL A, PRZYBOCKI M A, et al. The automatic content extraction (ace) program-tasks data and evaluation[C]//Proceedings of the Lrec. 2004, 2(1): 837-840.
- [19] WALKER C, STRASSEL S, MEDERO J, et al. ACE 2005 multilingual training corpus[J]. Linguistic Data Consortium, Philadelphia, 2006, 57: 45.
- [20] LEE J, YOON W, KIM S, et al. BioBERT: A pre-trained biomedical language representation model for biomedical text mining[J]. Bioinformatics, 2020, 36(4): 1234-1240.
- [21] ZHU E, LI J. Boundary smoothing for named entity recognition [C]//Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. 2022: 7096-7108.
- [22] STRAKOVÁ J, STRAKA M, HAJIÁ J. Neural architectures for nested NER through linearization[J].

- arXiv preprint arXiv:1908.06926, 2019.
- [23] HOFFMANN J, BORGEAUD S, MENSCH A, et al. Training compute-optimal large language models [J]. arXiv preprint arXiv:2203.15556, 2022.
- [24] ZHANG Z, ZHANG A, LI M, et al. Automatic chain of thought prompting in large language models [J]. arXiv preprint arXiv:2210.03493, 2022.
- [25] MEKALA D, WOLFE J, ROY S. ZEROTOP: Zero-shot task-oriented semantic parsing using large language models [J]. arXiv preprint arXiv:2212.10815, 2022.
- [26] SAHU G, RODRIGUEZ P, LARADJI I, et al. Data Augmentation for Intent Classification with Off-the-shelf Large Language Models [C] // Proceedings of the 4th Workshop on NLP for Conversational AI. 2022: 47-57.
- [27] YE J, GAO J, LI Q, et al. Zerogen: Efficient zero-shot learning via dataset generation [J]. arXiv preprint arXiv:2202.07922, 2022.
- [28] AGRAWAL M, HEGSELMANN S, LANG H, et al. Large language models are few-shot clinical information extractors [C] // Proceedings of the Conference on Empirical Methods in Natural Language Processing. 2022: 1998-2022.



孙瑜(1999—), 硕士, 主要研究领域为自然语言处理。

E-mail: yusun21@m.fudan.edu.cn



颜航(1994—), 博士, 主要研究领域为自然语言处理。

E-mail: hyan19@fudan.edu.cn



邱锡鹏(1983—), 通信作者, 博士, 教授, 主要研究领域为自然语言处理。

E-mail: xpqiu@fudan.edu.cn

(上接第 73 页)

- [30] THAKOOR S, TALLEC C, AZAR M G, et al. Bootstrapped representation learning on graphs [J]. arXiv preprint arXiv:2102.06514, 2021.
- [31] PEROZZI B, ALRFOU R, SKIENA S. DeepWalk: Online learning of social representations [C] // Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. USA: ACM, 2014: 701-710.
- [32] TANG J, QU M, WANG M, et al. LINE: Large-scale information network embedding [C] // Proceedings of the 24th International Conference on World Wide Web. Italy: ACM, 2015: 1067-1077.



岑科廷(1993—), 博士研究生, 主要研究领域为图表示学习和图神经网络。

E-mail: cenketing@ict.ac.cn



沈华伟(1982—), 通信作者, 博士, 研究员, 主要研究领域为社会计算和网络数据挖掘。

E-mail: shenhuawei@ict.ac.cn



曹婧(1992—), 博士, 助理研究员, 主要研究领域为社交媒体计算和可信图数据挖掘算法。

E-mail: caoqi@ict.ac.cn