通过 ChatGPT 进行抽取式摘要以生成忠实摘要

HaopengZhang Xiao Liu Jiawei Zhang加州大学戴维斯分校计算机科学系，
IFM 实验室，美国加利福尼亚州戴维斯 haopeng,xiao,jiawei@ifmlab.org

arXiv:2304.04193v2 [cs.CL] 9 Oct 2023

摘要

抽取式摘要是自然语言处理中的一项关键任务，旨在通过直接提取句子，将长文档压缩成更短的版本。大语言模型近期的出现，凭借其在广泛下游任务中展现出的卓越性能，引起了 NLP 社区的极大兴趣。本文首先对 ChatGPT在抽取式摘要任务上的性能进行了全面评估，并在多个基准数据集上将其与传统微调方法进行了比较。我们的实验分析表明，与现有的监督系统相比，ChatGPT在ROUGE 分数方面的抽取式摘要性能较差，但基于大语言模型的评估指标，其性能更高。此外，我们还探究了上下文学习和思维链推理在提升其性能方面的有效性。进一步而言，我们发现，与抽象式基线方法相比，使用ChatGPT 应用先抽取后生成的流程，能够在摘要忠实性方面显著提升性能。这些观察结果凸显了利用两阶段方法增强 ChatGPT 忠实摘要生成能力的潜在方向。

1 引言

文档摘要生成旨在压缩文本材料，同时保留其中最重要的信息。随着公开可用的文本数据不断增加，自动摘要方法变得愈发重要。这些方法大致可分为两类：抽象式摘要生成和抽取式摘要生成。尽管抽象式方法（Nallapati 等人，2016年；Gupta 和 Gupta，2019年）具有生成灵活且重复度较低的摘要这一优势，但它们往往难以避免生成语法错误甚至不符合事实的内容（Kry´s-ci´nski 等人，2019年；Zhang 等人，2022b）。相比之下，抽取式摘要直接从源文档中选择句子来形成摘要，从而生成语法正确且忠实于原文的摘要。

人们对将 ChatGPT 等先进大语言模型（LLM）[1] 应用于文本摘要任务的兴趣日益浓厚，并引发了广泛关注。Goyal 等人（Goyal 等人，2022年）最近的一项研究将 GPT-3 与传统微调方法进行了比较，发现尽管其 ROUGE 分数较低，人类标注者却更偏好 GPT-3 生成的文本。另一项研究由（Zhang 等人，2023d）开展，对用于新闻摘要的大语言模型进行了全面分析，发现其生成的摘要与人工生成的摘要不相上下。然而，现有研究（Yang 等人，2023年；Luo 等人，2023年）仅关注抽象式摘要方法，而 ChatGPT 在抽取式摘要方面的表现仍是一个悬而未决的问题。此外，幻觉问题严重阻碍了抽象式摘要系统的实际应用，这凸显了探索利用大语言模型进行抽取式摘要以生成忠实摘要的必要性。

在本研究中，我们全面评估了 ChatGPT 在抽取式摘要方面的表现，并探讨了上下文内学习和思维链解释方法的有效性。实验分析表明，与现有监督系统相比，ChatGPT 在 ROUGE 分数方面的抽取式摘要表现较差，但基于 LLM 的评估指标则显示其表现更优。此外，我们观察到，使用 ChatGPT 的先抽取后生成流程，在摘要忠实度方面的表现远超抽象式基线方法。

本文的主要贡献是：**1)** 本研究首次尝试将 ChatGPT 应用于抽取式摘-

[1] https://chat.openai.com/chat

要并评估其性能。**2)** 我们研究了使用 ChatGPT 进行抽取式摘-要时上下文学习和思维链推理方法的有效性。**3)** 我们进一步将抽取步骤扩展到抽象式摘要，并发现，与仅使用抽象式摘要的基线方法相比，先抽取后生成框架能够大幅提高生成摘要的忠实度，同时不会损害摘要质量。

2 相关工作

大多数抽取式摘要工作将该任务表述为序列分类问题，并使用具有多种编码器的序列神经网络模型，例如循环神经网络（Cheng 和 La- pata，2016年；Nallapati 等人，2016年）和预训练语言模型（Liu和 Lapata，2019年；Zhang 等人，2023年b）。另一类工作将抽取式摘要表述为节点分类问题，并应用图神经网络来建模句子间依赖关系（Xu 等人，2019年；Wang 等人，2020年；Zhang 等人，2022年a，2023年a）。

一些研究还探索了大语言模型（Brown 等人，2020年）在摘要生成中的应用。Goyal 等人(2022年)发现，尽管前者获得的 ROUGE 分数略低，但人工评估者更青睐它们。同样，Zhang 等人(2023d)报告称，在新闻领域，大语言模型生成的摘要与人工撰写的摘要不相上下。此外，Yang 等人(2023年)探索了 ChatGPT 在基于查询的摘要生成方面（而非通用摘要生成）的能力边界。Luo 等人(2023年)探索了将 ChatGPT 用作抽象式文本摘要的事实不一致性评估器。Zhang 等人(2023c)提出了一个基于 ChatGPT 的自我评估与修订框架。尽管现有研究大多聚焦于抽象式摘要生成，本研究旨在探究 ChatGPT 在抽取式摘要生成中的适用性，并考察抽取式方法能否提升抽象式摘要生成的忠实度。

3 方法

3.1 任务形式化

抽取式摘要系统通过识别给定文档中最显著的句子并将其连接起来，形成摘要。这些方法凭借其能够.....的能力，在各种现实世界的应用中得到了广泛采用。以生成准确可靠且不存在语法不一致的摘要。

形式化地说，给定一个文档 d，由 n个句子组成，抽取式摘要生成系统的目标是生成一个摘要 s，包含m（$m \ll n$）个句子，直接从源文档中抽取相关句子。现有大多数工作将其表述为序列标注问题，其中句子由模型 M 根据其是否应被纳入摘要 s的概率进行选择：

$$\hat{s} = \arg\max_{s} p_M(s \mid d). \tag{1}$$

在监督式摘要生成模型的训练中，通常会采用一种贪心算法，如（Nallapati 等人，2017年）中所述，通过选择多个句子，使其相较于黄金摘要的 ROUGE 得分最大化，从而生成抽取式真实标签（ORACLE）。

3.2 上下文学习

近期研究表明，大型语言模型在各种下游任务上展现出强大的少样本性能，这被称为上下文学习（ICL）（Brown 等，2020年）。标准的 ICL 会向大型语言模型 M 提供一组k 示例文档-摘要对，并通过以下方式为文档预测摘要 $\hat{}s$：

$$\hat{s} = \arg\max_{s} p_M(s \mid d, \{(d^1, s^1)...(d^k, s^k)\}). \tag{2}$$

除了简单的输入-输出对之外，先前的研究还表明，在提示词中加入解释和思维链（COT）推理（Nye 等人，2021年；Wei 等人，2022年；Ye 等人，2022年）也有益于语言模型，其表示形式如下：

$$\hat{s} = \arg\max_{s} p_M(s \mid d, C), \tag{3}$$

其中 $C = \{(d^1, e^1, s^1)...(d^k, e^k, s^k)\}$ 是提示词中的输入-解释-输出三元组集合。除零样本设置外，本研究还探讨了上下文内学习对抽取式摘要的影响，包括有解释和无解释两种情况。

3.3 摘取式摘要生成

使用抽取式摘要来指导抽象式摘要生成并不新颖（Dou 等人，2020年；Wang 等人，2022年）。在此，我们也提出以两阶段方式使用大语言模型：首先，提取显著句子以形成抽取式摘要（s^E），然后

模型	CNN/DM				XSum			
	R1	R2	RL	G-EVAL	R1	R2	RL	G-EVAL
SOTA-Ext	**44.41**	**20.86**	**40.55**	3.28	**24.86**	4.66	**18.41**	2.60
ChatGPT-Ext	39.25	17.09	25.64	3.24	19.85	2.96	13.29	2.67
+ 上下文	42.38	17.27	28.41	**3.30**	17.49	3.86	12.94	2.69
+ 原因	42.26	17.02	27.42	3.10	20.37	**4.78**	14.21	**2.89**
SOTA-Abs	**47.78**	**23.55**	**44.63**	3.25	**49.07**	**25.13**	**40.40**	2.79
ChatGPT-Abs	38.48	14.46	28.39	**3.46**	26.30	7.53	20.21	**3.47**
模型	Reddit				PubMed			
	R1	R2	RL	G-EVAL	R1	R2	RL	G-EVAL
SOTA-Ext	**25.09**	**6.17**	**20.13**	1.82	**41.21**	**14.91**	**36.75**	2.03
ChatGPT-Ext	21.40	4.69	14.62	**1.87**	36.15	11.94	25.30	2.12
+ 上下文	22.32	4.86	14.63	1.83	36.78	11.86	25.19	2.14
+ 推理	21.87	4.52	14.65	1.83	37.52	12.78	26.36	**2.18**
SOTA-Abs	**32.03**	**11.13**	**25.51**	1.87	**45.09**	**16.72**	**41.32**	**2.78**
ChatGPT-Abs	24.64	5.86	18.54	**2.43**	36.05	12.11	28.46	2.70

表 1：四个基准数据集上的摘要生成结果。“+context”和“+reason”分别指使用三个上下文内示例和人工推理的 ChatGPT。抽取式和抽象式设置中的最佳结果均以粗体显示。

数据集	领域	Doc #字数	Sum #字数	#Ext
Reddit	媒体	482.2	28.0	2
XSum	News	430.2	23.3	2
CNN/DM	News	766.1	58.2	3
PubMed	论文	444	209.5	6

表2：数据集的详细统计信息。文档词数和摘要词数分别指源文档和摘要中的平均词数。# Ext 指要抽取的句子数。

然后要求大语言模型在抽取式摘要的引导下生成摘要，表示为：

$$p(s \mid d) = \prod_{t=1}^{T} p\left(s_t \mid s_{<t}, d, s^E\right), \qquad (4)$$

其中，$s_{<t}$ 表示步骤 t之前生成的标记。本研究探讨先抽取后生成流程，旨在缓解大型语言模型生成摘要时的幻觉问题。

4 实验与分析

4.1 实验设置

数据集： 我们选择了表 2 中列出的四个公开可用的基准数据集，以确保它们与以往的微调方法保持一致。CNN/DailyMail（Hermann 等，2015）是应用最广泛的摘要生成数据集，其中包含新闻文章及其作为摘要的相应要点。我们使用未经匿名化的版本，并遵循常见的训练/验证/测试划分（287,084/13,367/11,489）。XSum（Narayan 等人，2018年）是一个单句新闻摘要生成数据集，包含由专业人员撰写的摘要。我们遵循thecommon划分（204,045/11,332/11,334）。PubMed（Cohan等人，2018年）是一个科学论文摘要生成数据集，我们按照（Zhong等人，2020年）的做法，将引言部分作为文章，将摘要部分作为摘要，并采用常见划分（83,233/4,946/5,025）。Reddit（Kim等人，2018年）是一个从社交媒体平台收集的高度抽象式数据集，划分为（41,675/645/645）。

评估： 我们采用 ROUGE（Lin 和 Hovy，2003年）按照以往研究的方法评估 ChatGPT 的摘要生成性能。我们还采用基于 GPT 的评估指标 G-EVAL（Liu 等人，2023年）。为考察摘要的忠实度，我们采用了常用指标 FactCC（Kry´sci´nski 等人，2019年）和 QuestEval（Scialom 等人，2021年）。

我们在一个由50个示例组成的开发集上选出了最佳提示词，并从原始数据集的每个测试集里随机抽取了1000 个示例进行评估。实验中使用的详细提示词以及有关实验设置的更多细节可见表4 和附录 B。

4.2 实验结果

总体结果如表1所示。上方部分包含抽取式结果以及 MatchSum 的最先进评分（Zhong 等人，2020年）。下方部分包含 CNN/DM 和 XSum 的生成式结果以及 BRIO 的最先进评分（Liu 等人，2022年），以及 SummaReranker 的评分（Ravaut 等人，2022年）针对

数据集	设置	RL	G-EVAL	FactCC	QuestEval
Reddit	Abs	18.54	2.43	9.46	40.79
	Ext-Abs	18.26	2.60	**60.40**	**49.45**
	Oracle-Abs	**19.37**	**2.64**	59.75	48.93
XSum	Abs	20.21	2.67	5.42	46.14
	Ext-Abs	18.55	2.28	**55.73**	**53.25**
	Oracle-Abs	**21.10**	**2.72**	55.03	53.21
PubMed	Abs	28.46	2.70	8.37	42.83
	Ext-Abs	26.50	2.81	26.38	44.32
	Oracle-Abs	**26.51**	**2.83**	**27.35**	**44.50**
CNN/DM	Abs	28.39	3.24	6.35	45.32
	Ext-Abs	29.16	3.50	51.65	51.72
	Oracle-Abs	**33.32**	**3.51**	**53.67**	**52.46**

表3：先抽取后生成流程的摘要生成结果。Abs、Ext-Abs 和 Oracle-Abs 分别指仅生成基线、先抽取后生成流程，以及基于 ORACLE 的生成。

Reddit 和 GSum（Dou 等人，2020年）用于 PubMed。

我们观察到，与以往的微调方法相比，ChatGPT 在抽取式设置和抽象式设置下的所有数据集上通常取得更低的 ROUGE 分数，但在基于大语言模型的评估指标 G-EVAL 方面取得更高的 评分。这些发现与（Goyal 等人，2022年；Zhang 等人，2023d）中的先前结论一致。我们还观察到，在两个抽取式数据集 CNN/DM 和 PubMed 上，ChatGPT-Ext 的表现优于 ChatGPT-Abs，而在另外两个抽象式数据集上的表现则较差。我们认为，这些结果源于数据集参考摘要中的偏差以及 ROUGE 分数的局限性。尽管如此，我们注意到，ChatGPT 主要是为生成任务设计的，却在需要理解文档的抽取式摘要任务中取得了令人印象深刻的结果。与 BERT 等编码器模型相比，ChatGPT 的仅解码器结构并未削弱其理解能力。我们还发现，在抽取式设置下，ChatGPT 与最先进的微调基线之间的 ROUGE 分数差距小于抽象式设置下的差距。

结果还表明，对于涵盖不同领域的四个数据集， 上下文内学习和推理通常有助于抽取式摘要生成任务。我们只观察到，在 XSum 数据集上，上下文内学习会导致性能下降。我们认为，性能下降是由于 XSum 的 ORACLE 较短，少量 ORACLE 示例反而会造成更多混淆。然而，借助思维链推理解释，ChatGPT 能够更好地理解其中的模式，因此上下文推理带来了性能提升。更多上下文学习结果见表5 附录中。

4.3 提取后生成

我们进一步开展实验，以检验表中所呈现的先抽取后生成框架的有效性3。

结果显示，在全部四个数据集上，采用先抽取后生成框架都显著提升了摘要的事实一致性。值得注意的是，仅生成基线方法的 FactCC 评分极低（低于 10%），这突显了基于ChatGPT 的摘要生成所存在的幻觉问题：ChatGPT往往会在摘要中编造新的内容。尽管如此，先抽取后生成框架通过利用从文档中抽取的显著句子来引导摘要生成过程，有效缓解了抽象式摘要的幻觉问题。我们还发现，与使用 ORACLE 引导生成相比，使用 ChatGPT 自身抽取的摘要来引导 ChatGPT 生成摘要，也能带来相近的摘要忠实度提升。

就摘要质量而言，结果表明，当以 ORACLE 摘要为依据时，ChatGPT的 ROUGE分数表现大幅提升。然而，当以其自身的抽取式摘要为依据时，先抽取后生成框架的 ROUGE分数表现严重依赖于抽取性能。总之，先抽取后生成框架能够在保持相近甚至更优摘要质量的同时，有效提升摘要忠实度。

4.4 位置偏差

首段偏差是抽取式摘要中的常见现象，尤其是在新闻领域，因为文章的开头部分通常包含最重要的信息。如图 1所示，我们发现，ChatGPT 提取的摘要句子的位置分布相比 ORACLE 句子表现出更高的位置偏差。此外，上下文学习为摘要带来了更多的位置偏差。结果表明，大语言模型可能会依赖句子位置等表层特征来进行抽取式摘要。

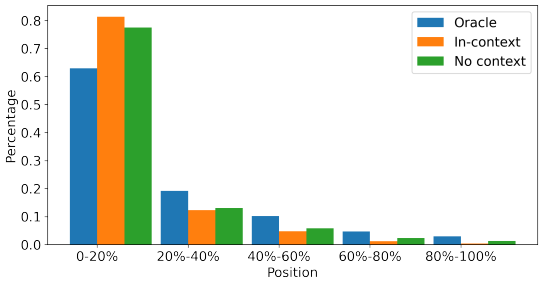

图1：从 CNN/DM 测试集中随机抽取的 1000 个样本中，抽取句子的位置分布。

5 结论

本文对 ChatGPT 在四个基准数据集上的抽取式摘要表现进行了全面评估。结果表明，ChatGPT 在这项任务中展现出巨大潜力，并且可以通过抽取-生成框架生成事实性摘要。总体而言，本研究表明 ChatGPT 是一种强大的文本摘要工具，我们希望本研究获得的见解能够为该领域的未来研究提供指导。

局限性

由于预算有限，我们没有在整个测试集上进行实验，而是从每个数据集的测试集中随机抽取 1000 个样本。此前的研究（ Goyal 等人，2022年；Zhang 等人，2023d）在使用少量实例测试 GPT-3 方面也存在局限。

我们的实验结果主要通过各种自动评估指标（摘要质量和忠实度）进行评估。未来，我们计划加入人工评估研究，以进一步验证这些结论。

我们仅使用 *gpt-3.5-turbo* 模型，来自 OpenAI API，作为大语言模型的一个实例。该本文重点是探索使用大语言模型进行抽取式摘要以及先抽取后生成流程，而不是比较不同的开源和闭源大语言模型。

参考资料

Tom Brown、Benjamin Mann、Nick Ryder、Melanie Subbiah、Jared D Kaplan、Prafulla Dhariwal、Arvind Neelakantan、Pranav Shyam、Girish Sastry、Amanda Askell 等。2020年。语言模型是少样本学习者。神经信息处理系统进展，33：1877–1901。

Jianpeng Cheng 和 Mirella Lapata。2016年。通过提取句子和词语进行神经摘要生成。*arXiv* 预印本 *arXiv:1603.07252*。

Arman Cohan、Franck Dernoncourt、Doo Soon Ki m、Trung Bui、Seokhwan Kim、Walter Chang 和 Nazli Goharian。2018年。用于长文档抽象式摘要的话语感知注意力模型。*arXiv* 预印本 *arXiv:1804.05685*。

Zi-YiDou、Pengfei Liu、Hiroaki Hayashi、Zhengbao Jiang和Graham Neubig。2020年。GSum：一种用于引导式神经抽象摘要生成的通用框架。*arXiv* 预印本 *arXiv:2010.08014*。

Tanya Goyal、Junyi Jessy Li 和 GregDurrett。2022年。GPT-3时代的新闻摘要与评估。*arXiv*预印本 *arXiv:2209.12356*。

Som Gupta 和 Sanjai Kumar Gupta。2019年。抽象式摘要：最先进技术综述。《*Expert Systems with Applications*》, 121:49–65.

Karl Moritz Hermann、Tomas Kocisky、Edward Grefen- stette、Lasse Espeholt、Will Kay、MustafaSuleyman和Phil Blunsom。2015 年。教会机器阅读和理解。载于神经信息处理系统进展，第 1693–1701 页。

ByeongchangKim、HyunwooKim 和GunheeKim。2018年。利用多层记忆网络对 Reddit 帖子进行抽象式摘要。*arXiv* 预印本 *arXiv:1811.00783*。

Wojciech Kry´sci´nski、Bryan McCann、CaimingXiong 和 Richard Socher。2019年。评估抽象式文本摘要的事实一致性。*arXiv* 预印本 *arXiv:1910.12840*。

Chin-YewLin 和 EduardHovy。2003 年。使用n元语法共现统计进行摘要的自动评估。载于 *2003* 年北美计算语言学协会分会人类语言技术会议论文集，第 150–157 页。

Yang Liu、Dan Iter、Yichong Xu、Shuohang Wang、Ruochen Xu 和 Chenguang Zhu。2023年。Gpteval：使用 GPT-4 进行自然语言生成评估，并提升与人类判断的一致性。*arXiv* 预印本 *arXiv:2303.16634*。

Yang Liu 和 Mirella Lapata。2019年。使用预训练编码器进行文本摘要。*arXiv* 预印本 *arXiv:1908.08345*。

Yixin Liu、Pengfei Liu、Dragomir Radev 和 Graham Neubig。2022年。BRIO：为抽象式摘要生成建立秩序。载于 第*60*届计算语言学协会年会论文集（第*1*卷：长篇论文），第2890–2903页，都柏林，爱尔兰。计算语言学协会。

罗哲恒、谢倩倩和 Sophia Ananiadou。2023年。ChatGPT 作为抽象式文本摘要的事实不一致性评估器。*arXiv* 预印本 *arXiv:2303.15621*.

Ramesh Nallapati、翟飞飞和周博文。2017年。SummaRunner：一种用于文档抽取式摘要的基于循环神经网络的序列模型。载于 第三十一届人工智能促进协会人工智能会议。

RameshNallapati、周博文、Caglar Gulcehre、向冰等。2016年。使用序列到序列循环神经网络及其他方法进行抽象式文本摘要。*arXiv*预印本 *arXiv:1602.06023*。

Shashi Narayan、Shay B Cohen 和 Mirella Lapata。2018年。别告诉我细节，只给我摘要！用于极限摘要的主题感知卷积神经网络。*arXiv*预印本 *arXiv:1808.08745*。

Maxwell Nye、Anders Johan Andreassen、Guy Gur-Ari、Henryk Michalewski、Jacob Austin、David Bieber、David Dohan、Aitor Lewkowycz、Maarten Bosma、David Luan 等。2021年。展示你的工作：用于语言模型中间计算的草稿本。*arXiv* 预印本 *arXiv:2112.00114*。

Mathieu Ravaut、Shafiq Joty 和 Nancy F Chen。2022年。Summareranker：一种用于抽象式摘要的多任务混合专家重排序框架。*arXiv* 预印本 *arXiv:2203.06569*。

Thomas Scialom、Paul-Alexis Dray、Gallinari Patrick、Lamprier Sylvain、Piwowarski Benjamin、Staiano Jacopo 和 Wang Alex。2021年。Questeval：摘要生成需要基于事实的评估。*arXiv* 预印本 *arXiv:2103.12693*。

王丹青, PengfeiLiu, 郑一宁, 邱锡鹏和黄萱菁。2020年。用于抽取式文档摘要生成的异构图神经网络。*arXiv* 预印本 *arXiv:2004.12393*。

Fei Wang、Kaiqiang Song、Hongming Zhang、Lifeng Jin、Sangwoo Cho、Wenlin Yao、Xiaoyang Wang、Muhao Chen 和 Dong Yu。2022年。以显著性分配指导抽象式摘要。*arXiv* 预印本 *arXiv:2210.12330*。

Jason Wei、Xuezhi Wang、Dale Schuurmans、Maarten Bosma、Ed Chi、Quoc Le 和 Denny Zhou。2022年。思维链提示能够激发大语言模型的推理能力。*arXiv* 预印本 *arXiv:2201.11903*。

Jiacheng Xu、Zhe Gan、Yu Cheng 和 Jingjing Liu。2019年。面向文本摘要生成的篇章感知神经抽取模型。*arXiv*预印本 *arXiv:1910.14142*。

Xianjun Yang、YanLi、Xinlu Zhang、HaifengChen 和 Wei Cheng。2023年。探索 ChatGPT 在查询或方面导向文本摘要生成方面的局限。*arXiv* 预印本 *arXiv:2302.08081*。

Xi Ye、Srinivasan Iyer、Asli Celikyilmaz、Ves Stoyanov、Greg Durrett 和 Ramakanth Pasunuru。2022年。用于有效上下文学习的互补解释。*arXiv* 预印本 *arXiv:2211.13892*。

Haopeng Zhang、Xiao Liu 和 Jiawei Zhang。2022年a。HEGEL：用于长文档摘要生成的超图 Transformer。载于 *2022*年自然语言处理经验方法会议论文集，第 10167–10176 页，阿联酋阿布扎比。计算语言学协会。

Haopeng Zhang、Xiao Liu 和 Jiawei Zhang。2023年a。用于科学文档摘要生成的对比式层级篇章图。载于第*4*届篇章计算方法研讨会论文集（*CODI 2023*），第37–47页，加拿大多伦多。计算语言学协会。

Haopeng Zhang、Xiao Liu 和 Jiawei Zhang。2023年b。DiffuSum：通过扩散增强生成的抽取式摘要。载于计算语言学协会论文集：*ACL 2023*年，第 13089–13100 页，加拿大多伦多。计算语言学协会。

Haopeng Zhang、Xiao Liu 和 Jiawei Zhang。2023c。Summit：通过 ChatGPT 进行迭代式文本摘要生成。

Haopeng Zhang、Semih Yavuz、Wojciech Kryscinski、Kazuma Hashimoto 和 Yingbo Zhou。2022b。通过实体覆盖控制提升抽象式摘要的忠实度。载于计算语言学协会：*NAACL 2022* 会议论文集，第 528–535 页，美国西雅图。计算语言学协会。

Tianyi Zhang、Faisal Ladhak、Esin Durmus、Percy Liang、Kathleen McKeown 和 Tatsunori B Hashimoto。2023d。面向新闻摘要的大语言模型基准测试。*arXiv* 预印本 *arXiv:2301.13848*。

钟明、PengfeiLiu、陈怡然、王丹青、邱锡鹏和黄萱菁。2020年。抽取式摘要生成作为文本匹配。*arXiv*预印本 *arXiv:2004.08795*。

附录

A 提示词

此处我们在表中列出了实验中用于抽取式摘要和生成式摘要的提示词 4。请注意，根据 OpenAI 的文档，模型可以接收两类提示词：系统提示词和用户提示词，其中系统提示词充当用于初始化模型的全局指令，而用户提示词则是用户提出的问题。

B 实验设置

我们采用gpt-3.5-turbo模型[2] 来生成和评估摘要，将温度设置为 0 以确保可复现性。

关于数据集，采用了随机抽样方法，为实验目的从每个数据集中选取了1000个样本。此外，还使用了一个包含50 个样本的较小子集，以寻找最优的提示词和超参数。随机种子设定为 101，以促进一致性。

根据既有研究，ROUGE[3] F-1 分数被用作自- 动评估指标（Lin 和 Hovy，2003）。具体而言，ROUGE-1/2 分数用于衡量摘要信息量，而 ROUGE-L 分数则用于评估摘- 要的流畅性。除这些指标外，还引入了GPT模型作为摘要评估器，以模拟人工评估过程。该评估器旨在基于对连贯性、一致性、流畅性和效率的综合分析来评估摘要。每项实验的结果均以单次运行结果的形式报告。

涉及每个数据集的实验包含1000个示例，将运行1.5小时，以执行推理和评估。

C 上下文学习结果

详细的上下文学习结果如表 5

D 文档长度分析

我们进一步研究了文档长度对摘要生成性能的影响，aspre-

[2]https://platform.openai.com/docs/guides/gpt/chat-completions-api [3]ROUGE: https://pypi.org/project/rouge-score/

设置	提示
抽取式	**系统**：你是一个遵循输出模式的抽取式摘要生成器，请遵循输出模式。 **用户**：请抽取句子作为摘要。摘要应包含 **m** 句子。 文档： [测试 文档-*ment*] [格式指令]。
抽象式	**系统**：你是一名抽象式摘要-遵循输出模式的摘要。 **用户**：请为以下内容撰写摘要。文档。文档： [测试文档] [格式指令]
上下文内	**系统**：你是一位抽取式摘要生成器，遵循输出模式。 **用户**： 以下示例是成功的抽取式摘要在-条目： [*n* 文档-摘要对]。请总结以下文档-部分。文档： [测试文档]。该摘要应包含 m 个句子。[格式指令]。
解释	**系统**：你是一名抽取式摘要师-遵循输出模式的摘要器。 **用户**： 以下示例是成功的抽取式摘要例： [*n* 文档-摘要-理由三元组]。请总结以下文档并说明理由。文档： [测试文档]。摘要应包含 m 个句子。 [格式结构]。
提取-摘要	**系统**：你是一个抽象式摘要器。遵循输出模式的摘要器。 **用户**：请修改提取的摘要-根据文档进行总结。重新-修订后的摘要应包含以下信息：提取式摘要中的信息。文档： [*TestDocument*]提取式摘要： [提取式摘要] [格式说明说明]。
评估器	**系统**：你是一个摘要评估器遵循输出模式。你需要给出摘要的评分，评分依据是对以下标准进行全面考量： 标准： （1）连贯性：“整体质量所有句子”； (2) 一致性：“事实对齐摘要与参考文献之间-ence”； (3) 流畅性：“单个内容的质量句子”； (4) 效率：“如果摘要是 con-简洁” **用户**： 请 计算总和-mary 基于参考文本的摘要-玛丽.参考文本：[参考摘要] 摘要：[预测摘要][格式指令]。

表4：用于抽取式和抽象式摘要生成的提示词。 m 是表中定义的抽取句子数量2。文档-摘要对和文档-摘要-理由三元组是输入上下文。n是上下文实例的数量。

# 上下文	CNN/DM			XSum		
	R1	R2	RL	R1	R2	RL
0	39.25 ± 0.23	15.36 ± 1.10	25.90 ± 0.97	19.85 ± 2.59	2.96 ± 2.59	13.29 ± 1.30
1	40.62 ± 0.70	17.00 ± 1.06	26.44 ± 0.84	15.33 ± 0.50	2.48 ± 0.19	11.48 ± 0.13
1w/R	38.83 ± 0.91	14.94 ± 2.53	25.36 ± 1.82	17.86 ± 1.73	3.29 ± 0.85	12.55 ± 1.29
2	40.91 ± 0.69	15.68 ± 0.61	26.13 ± 0.83	18.61 ± 0.39	4.42 ± 0.97	14.06 ± 2.01
2w/R	41.70 ± 0.70	15.95 ± 0.92	26.98 ± 1.33	17.95 ± 3.03	4.11 ± 1.01	13.46 ± 1.76
3	**42.38 ± 0.13**	17.27 ± 0.23	**28.41 ± 0.31**	17.49 ± 1.87	3.86 ± 1.55	12.94 ± 2.16
3w/R	42.26 ± 1.38	17.02 ± 1.60	27.42 ± 1.62	**20.37 ± 1.61**	**4.78 ± 0.44**	**14.21 ± 1.07**
4	42.26 ± 0.50	**17.41 ± 0.83**	27.96 ± 0.83	16.68 ± 1.56	3.72 ± 0.20	12.12 ± 1.19
4w/R	41.23 ± 0.93	17.08 ± 0.38	28.25 ± 0.93	18.17 ± 0.28	4.05 ± 0.38	12.74 ± 0.94
5	40.71 ± 1.92	16.96 ± 0.91	27.42 ± 1.26	17.43 ± 1.08	3.53 ± 0.96	12.33 ± 0.51
5w/R	40.18 ± 0.83	15.15 ± 1.44	25.98 ± 1.91	19.55 ± 0.64	4.29 ± 0.46	13.13 ± 0.68

表5：在 CNN/DM 和 XSum 数据集上的上下文学习实验结果。对于每个数据集，我们从测试集中随机抽取了50条数据。在每个部分中，w/R 表示我们为每篇上下文文档提供了人工撰写的理由。对于测试文档，我们还要求系统生成其选择这些句子的理由。

呈现在图2。我们的研究结果表明，Chat-GPT 在不同长度的文档上均保持稳定的性能，这表明该模型在抽取式摘要任务中具有较强的稳健性。

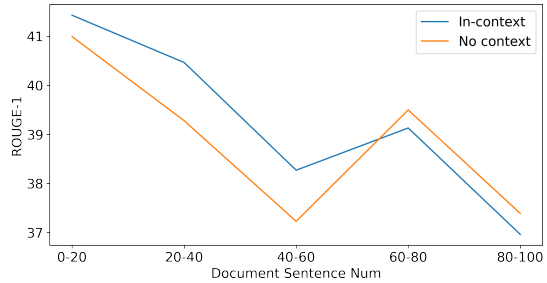

图 2：不同长度文档的 ROUGE-1 分数分布。

E 案例研究

这里我们在表格中展示使用不同提示设置生成的 ChatGPT 摘要6 ，以 CNNDM 数据集中的一个示例为例。

文档	特技表演者尼克·瓦伦达表示，他将在一座400英尺高的观景摩天轮顶部不系安全绳行走 本月将在佛罗里达州奥兰多进行这项挑战。沃伦达周一在纽约市的一场新闻发布会上表示， 他将于4月29日尝试这一壮举时，奥兰多之眼将会转动。奥兰多之眼是 一座新娱乐综合体的一部分；它开放供公众参观后，游客可从30个封闭式、 配有空调的玻璃座舱内欣赏佛罗里达州中部的景色。目标在望：走钢丝 表演者尼克·瓦伦达在一次新闻 会议于周一在纽约举行。艰巨挑战：这位36岁的特技表演者将行走于顶部 奥兰多之眼于4月29日开启时。奥兰多之眼团队发表声明称，其 很期待沃伦达尝试这项“惊人特技”。表演的距离尚未确定。 尚未确定，但36岁的沃伦达表示，他不太可能走完全程，因为那大约需要20分钟。 转完一整圈。沃伦达此前曾在圣克鲁斯的一座摩天轮上行走， 佛罗里达州，但他说，奥兰多那座大得多的摩天轮，以及他不会使用 平衡杆这两点让这次特技与众不同。“飞翔的沃伦达”杂技家族的第七代成员 曾走过大峡谷和尼亚加拉瀑布。11月，他两次 在芝加哥两座摩天大楼之间行走，全程没有安全网或安全带，第二次行走时还蒙着眼睛。 沃伦达是卡尔·沃伦达的曾孙；卡尔·沃伦达73岁时在波多黎各坠落身亡。
参考文本	这位36岁的男子将于4月29日上演他的下一场特技表演。11月，沃伦达走了回去，并且 在两座芝加哥摩天大楼之间来回行走，这是一场电视直播活动。他的曾祖父卡尔 沃伦达于1978年在波多黎各走钢丝时身亡。沃伦达也曾走过钢丝。 穿越尼亚加拉瀑布和大峡谷。
ORACLE	艰巨挑战：这名36岁的特技表演者将于4月在转动中的奥兰多之眼顶部行走 29. “飞翔的沃伦达”家族的第七代杂技演员已经走过 穿越大峡谷和尼亚加拉瀑布。11月，他在两个芝加哥之间步行了两次 没有安全网或安全带的摩天大楼，第二次行走时还蒙着眼睛。
生成式摘要	特技表演者尼克·瓦伦达计划不系安全绳，在400英尺高的奥兰多之眼观景 4月29日乘坐车轮。36岁的走钢丝表演者，是第七代成员。 “飞人沃伦达”杂技世家此前曾完成过横越大 峡谷、尼亚加拉瀑布，以及芝加哥摩天大楼之间的惊险行走。尽管这次 特技表演的确切距离尚未确定，沃伦达表示，他不太可能走完摩天轮转完一整圈所需的20分钟 以使车轮完成一整圈。
摘录式	特技表演者尼克·瓦伦达表示，他将不系安全绳，在一座400英尺高的观景摩天轮顶部行走 本月在佛罗里达州奥兰多。沃伦达周一在纽约市的一场新闻发布会上表示， 4月29日，他尝试这项壮举时，奥兰多之眼将会转动。这位36岁的特技表演者 4月29日，奥兰多之眼转动时，他将在其顶部行走。
上下文内	特技表演者尼克·瓦伦达表示，他将在一座400英尺高的观景摩天轮顶端不系安全绳行走，地点是 本月在佛罗里达州奥兰多。沃伦达周一在纽约市的一场新闻发布会上表示 届时奥兰多之眼将在他于4月29日挑战绝技时转动。沃伦达是 卡尔·沃伦达的曾孙。卡尔·沃伦达73岁时在波多黎各坠落身亡。
抽取式摘要	尼克·瓦伦达是一名特技表演者，也是“飞翔的沃伦达”家族的第七代成员，该家族世代 以杂技表演为业。他将于4月29日在佛罗里达州奥兰多，尝试不系安全绳，在一座400英尺高的观景摩天轮顶端行走。 4月29日，佛罗里达州奥兰多的奥兰多之眼将提供景观。这座观景摩天轮是一个新娱乐综合体的一部分。 从佛罗里达州中部的30个封闭式空调玻璃舱内观赏，待其于 5月4日向公众开放。沃伦达此前曾在加利福尼亚州圣克鲁斯市走过摩天轮顶部，但 他说，奥兰多摩天轮的规模大得多，而且他不会使用平衡杆，这些因素让这次特技与众不同。 沃伦达是卡尔·沃伦达的曾孙。卡尔·沃伦达在 73岁时在波多 黎各坠落身亡。

表6：不同设置的案例研究