基于三智能体生成流水线的个性化抽象摘要

Wen Xiao[‡] **Yujia Xie**[‡] **Giuseppe Carenini**[†] **Pengcheng He**[‡]
[†]英属哥伦比亚大学，加拿大温哥华
[‡] Microsoft AzureAI{carenini}@cs.ubc.ca,{wxiao,yujiaxie,penhe}@microsoft.com

arXiv:2305.02483v2 [cs.CL] 1 Mar 2024

摘要

尽管大型语言模型（如 ChatGPT）具有出色的生成能力，但如何使其输出契合用户隐含的偏好仍是一项挑战。本文提出了一种由生成器、指导器和编辑器组成的三智能体生成流水线，以增强输出的个性化程度。生成器生成初始输出，指导器根据用户偏好自动生成编辑指令，编辑器则对输出进行润色，使其符合这些偏好。仅用于推理的大语言模型（ChatGPT）同时充当生成器和编辑器，而较小的模型则担任指导器，引导输出生成。我们采用编辑器引导的强化学习来训练指导器，利用大规模编辑模型的反馈来优化指令生成。在两个抽象摘要数据集上的实验结果表明，我们的方法能够有效生成更符合用户期望的输出。 [1]

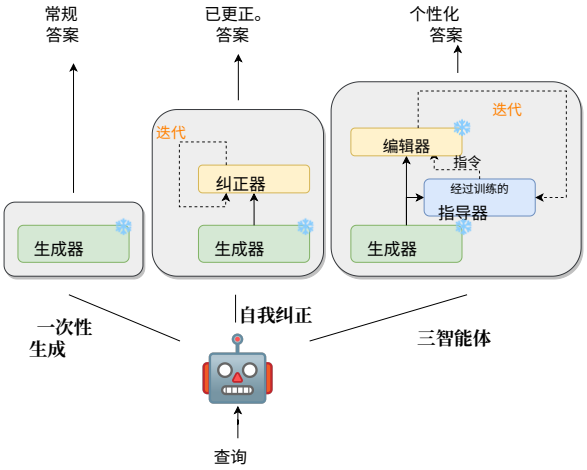

图 1：不同生成范式之间的比较。左侧是通用的一次性生-成过程，中间是Welleck 等人（2022）提出的方案，该方案使用经过训练的纠正器对生成的文本进行纠-正，通常用于处理特定问题，例如消除幻觉或毒性，而右侧是本文提出的三智能体流水线。

1 引言

以 InstructGPT（Ouyang 等人，2022）和 ChatGPT[2], 等知名模型为代表的大语言模型，已成为自然语言处理（NLP）领域的重要资源。这些模型在广泛的 NLP 任务中展现出卓越的能力，包括机器翻译、问答和文本摘要。鉴于它们有望推动语言技术领域的进一步创新，研究界对探索和推进大语言模型的热情日益高涨。然而，尽管这些模型取得了令人印象深刻的生成质量，但如何调整其输出以满足用户偏好（Liu 等人，2022b）仍是一项长期存在的挑战。在

在若干场景中，人们观察到，语言模型的输出并不能始终满足用户的偏好或预期（Bubeck 等人，2023年）。一种常见的解决方法是精心设计提示词，引导模型生成更符合用户目标的输出。然而，正如现有研究所指出的（Reid 和 Neubig，2022），语言模型传统的一次性从左到右生成过程，与人类通常采用的迭代式润色和编辑方式形成对比。此外，既往研究（Gu 等人，2019；Reid 和 Zhong，2021年）表明，与一次性生成相比，生成后编辑的流程效果更佳，即使只进行一轮编辑也是如此。受这些发现启发，本文探索将大型语言模型（ChatGPT）集成到自动迭代编辑流程中。

[1]代码可在 https://github.com/Wendy-Xiao/chatgpt_编辑_摘要[2]https://openai.com/blog/chatgpt

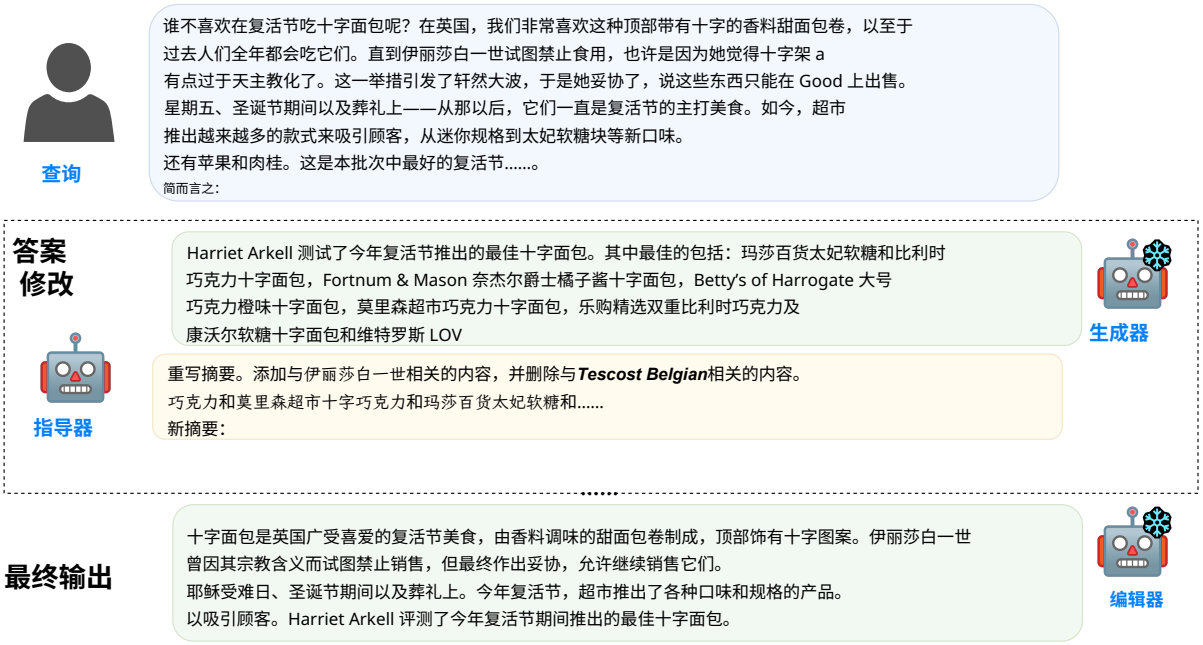

图 2：所提出的三智能体生成流水线示意图。给定查询后，生成器首先生成初始答案，指导器提供指令，说明如何使答案更贴合用户偏好，最后编辑器根据给定的指令生成个性化答案。

与Welleck 等人 (2022)采用的方法不同，他们将生成过程分解为生成器和纠正器；我们的方法则将其分解为三个组件：**生成器、指导器和编辑器**（参见图1）。这种结构使我们能够利用仅用于推理的大型模型执行内容生成和纠正等复杂任务，同时使用较小的模型生成个性化编辑指令这一较简单的任务。指导器旨在提供有针对性的指示，以编辑和润色生成器的初始输出。其初始训练使用人工撰写的指令，或预言机指令，这些指令可根据用户的行为历史获得。之后，再通过编辑器引导的强化学习对模型进行微调；在这一过程中，奖励函数直接量化**编辑器编辑后的输出**与用户偏好的契合程度，从而提升模型与编辑器的兼容性。

我们选择文本摘要作为评估这一新颖框架的核心任务，该框架旨在为给定的文档生成简洁且信息丰富的摘要。在本文中，我们对两个摘要数据集进行了实验评估（DeFacto (Liu et al., 2022b) 和 CNNDM (Nal-lapati etal., 2016 年）），重点关注用户偏好，尤其是事实一致性和覆盖度方面的偏好。我们使用 ChatGPT 作为生成器和编辑模型。实验结果表明，在小型指导器模型生成的指令下，编辑后的输出在两个数据集上都更符合用户偏好。关于迭代编辑的进一步实验表明，随着编辑迭代次数增加，输出能够更好地满足用户需求。

2 总体流程

为了增强生成流水线的灵活性，并优化其与强大大语言模型的兼容性，我们提出了一种新颖的方法，将生成过程分解为三个不同的组件，如图2所示。这些组件包括：(1) **生成器**，负责生成初始输出；(2) **指令生成器**，负责生成自然语言指令，引导对初始输出进行编辑，使其朝着符合用户偏好的方向调整；以及 (3) **编辑器**，根据提供的指令对初始输出进行优化。

鉴于已有研究证明，大语言模型既可以充当生成器，也可以充当编辑模型，因此我们选择使用一种仅用于推理的大语言模型，具体而言是 ChatGPT，作为我们的生成器和编辑器。尽管还可以进一步微调这些大语言模型，

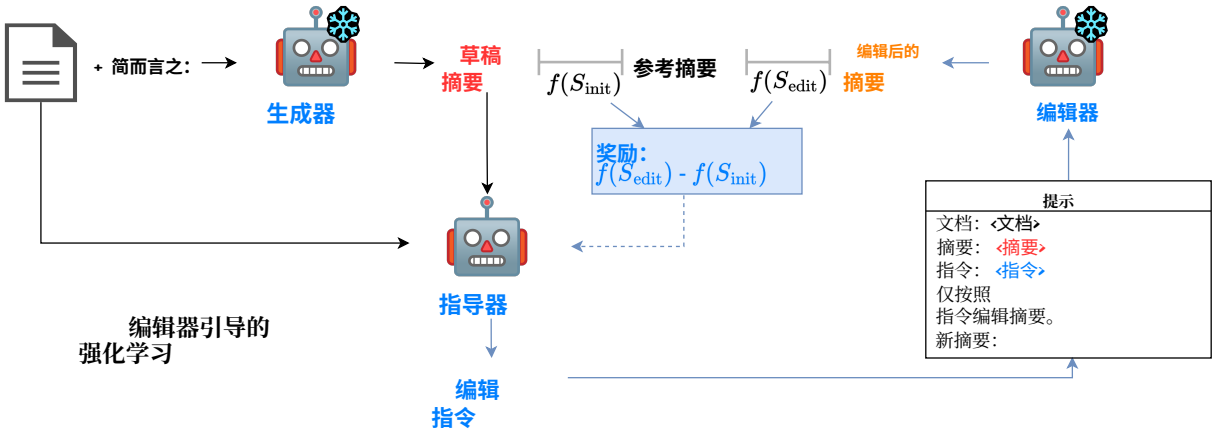

图 3：面向指导器的编辑器引导的强化学习。我们使用编辑器引导的强化学习对指导器进行微调，以最大化编辑器（例如 ChatGPT）的预期性能。

作为指导器，计算资源等实际限制（Touvron 等人，2023年）以及访问权限限制（Ouyang 等人，2022）可能导致无法直接进行微调，正如以往研究所采用的方法（Welleck 等人，2022；Liu 等人，2022a）。因此，我们提出采用编辑器引导的强化学习来训练一个较小的模型，使其充当用户专属指导器（如第 3 节所述），引导编辑器修改初始输出，从而更好地与人类期望保持一致。

3 编辑器引导的指导器

如上所述，所提出的指导器的核心目标是生成精确且可操作的指令，以指导大语言模型修正原始摘要，使其更贴合用户偏好。为实现这一目标，我们采用了一个两阶段训练过程，旨在使指导器能够与大语言模型协同工作。

具体而言，给定文档 D，生成器（摘要模型或大语言模型）会生成一份初始摘要，记为 S_{init}。指导器的目标是将 D 和 S_{init} 作为输入，生成一组指令 $I = \{i_1, i_2, ..., i_k\}$，旨在引导编辑模型生成一份更符合用户偏好的编辑后的摘要。最后，编辑器将 D、S_{init}和 I 作为输入，根据给定的指令生成修订后的摘要 S_{edit}。

3.1 步骤1：监督学习

在初始训练阶段，我们会根据用户的历史偏好生成一组用于修正摘要的预言机指令。[3] 这些预言机指令为指导器应当生成的指令提供了理想示例。随后，我们使用负对数似然损失，以监督方式训练指导器模型，即：

$$L = \sum_{k} P(i_1, i_2, ..., i_k | D, S_{\text{init}}).$$

该阶段的目标是为指导器生成符合用户预期的指令奠定坚实基础，使其能够学习输入（源文档和初始摘要）与期望输出（预言机指令）之间的关系。

3.2 第 2 步：编辑器引导的强化学习

在第二阶段，我们进一步使用编辑器引导的强化学习技术微调指令模型（见图3），具体采用NLPO算法（Ramamurthy等人，2023年）。

这一阶段的一个关键方面是奖励函数的设计，它为基于强化学习的微调过程提供指导信号。为确保生成的指令与

[3]这些预言机指令通过使用人工撰写的摘要作为参考摘要来模拟用户偏好而构建，这些参考摘要反映了每个来源独特的摘要偏好。例如，CNN和DailyMail在其为新闻文章生成的摘要中可能表现出特定的倾向。

编辑模型兼容，并能带来有意义的摘要修正，奖励函数依据编辑后的摘要制定。编辑后的摘要由编辑模型生成，其提示包含源文档、初始摘要，以及指导器模型提供的编辑指令（示例提示见图3右下角）。

为量化编辑后的摘要质量，我们采用评分函数 $f(\cdot)$，用于衡量摘要满足用户偏好的程度。由于我们将生成摘要的覆盖度和事实一致性作为用户的需求，因此评分函数 $f(\cdot)$ 设为ROUGE分数与知识覆盖度之和，用于衡量实体级覆盖度与参考摘要的相似性，

$$f(S) = \alpha \text{ROUGE}(S, S_{\text{ref}}) + \beta Cov(S, S_{\text{ref}}).$$

奖励信号本身定义为初始摘要与编辑后的摘要之间的评分差值，旨在捕捉摘要质量的提升，奖励值越高，表示质量提升越显著。

$$Reward = f(S_{\text{edit}}) - f(S_{\text{init}}).$$

此阶段旨在增强模型生成指令的能力，使其不仅能够遵循用户需求，还能有效地引导大语言模型生成质量更高的摘要。

4 实验

我们在两个不同的数据集上进行实验，每个数据集都体现了用户偏好的不同方面。

4.1 场景 1：DeFacto 上的事实一致性

在初始实验场景中，我们选择将事实一致性作为用户摘要偏好的首要标准。[4] 我们采用 DeFacto 数据集（Liu 等人，2022b），该资源经过专门整理，通过纳入人工标注的示范和反馈，旨在提升机器生成摘要的事实一致性。该数据集包含 701/341/779 训练集/验证集/测试集中的数据示例。[5] DeFacto 数据集中的每条数据都包含一篇源文档，以及由 PEGASUS 生成的初始摘要（Zhang 等人，2020）。标注人员需要提供一条指令，指导如何修改初始摘要以增强事实一致性。此外，标注人员还需生成一份修订后的摘要，使其符合所提供的指令，并具有更高的事实一致性。

为了评估系统生成的指令与人工编写的指令之间的一致性，我们采用 ROUGE 分数作为评估指标。此外，我们结合多种指标（包括 ROUGE 分数和事实准确性分数），评估生成摘要在人类期望和事实准确性方面的质量。具体而言，我们使用 DAE（依存弧蕴含）指标（Goyal andDurrett,2021年）和 QFE（用于事实评估的问答）指标（Fabbri et al.,2022），量化生成摘要的事实准确性。这些指标从与人类期望的一致性和对事实正确性的遵循程度两方面，对摘要质量进行全面评估。

设置 我们使用 FlanT5-large (700M)（Chung 等人，2022）作为指导器的骨干模型。指导器的训练过程分两个阶段执行，详见第 3 节。

结果 首先，我们评估 ChatGPT 作为编辑模型的潜力，即根据人工提供的指令修改摘要的能力。表 1所示的评估结果表明，当向 ChatGPT 提供源文档、初始摘要和人工撰写的编辑指令作为输入时，其表现与监督模型相当，ROUGE 分数和事实准确性分数也与监督模型不相上下。这些发现证实，只要提供适当的编辑指令，ChatGPT 就能有效地充当摘要编辑器。

随后，我们将系统生成的指令与人工编写的指令进行比较评估。我们的目标是确定 ChatGPT 和经过训练的指导器在准确辨别用户需求并据此生成相应指令方面的能力。结果

[4] 虽然事实一致性通常可作为摘要器的一项典型评判标准，但我们利用指导器获得编写特定指令的能力，以增强摘要的事实一致性。

[5] 遵循原论文的做法，所有实验均在标注有错误的示例上进行。

编辑器	DAE	QFE	R1	R2	RL
初始摘要	0.699	1.837	76.03	66.34	74.11
人工编辑者	0.906	2.717	100	100	100
T0PP-D+S+I（监督）	0.904	2.470	88.74	83.16	87.48
ChatGPT（10-shot）	0.884	2.568	88.48	81.41	86.17

表 1：DeFacto 上带有人类编写指令的编辑后摘要的 ROUGE分数和事实一致性分数，并与人工编辑的摘要进行比较。T0PP-D+S+I（Sup）是一种监督模型，其输入为源文档、初始摘要和指令（Liu 等人，2022b）。

模型	R1	R2	RL
ChatGPT（零样本）	36.05	22.98	30.66
ChatGPT（10-shot）	37.35	24.94	32.94
FlanT5（监督学习）	49.04	34.37	47.07
FlanT5（强化学习）	48.05	32.94	46.23

表 2：DeFacto 上生成的指令与人工编写的指令之间的 ROUGE分数。

如表2所示。值得注意的是，我们观察到 ChatGPT 生成的指令与人工编写的指令并不匹配，这一点从其在零样本和少样本设置下均表现欠佳可以得到印证。尽管我们使用的指导器模型比 ChatGPT 小得多（700M 对 175B），但它展现出了生成更符合用户需求的指令的能力。

在最后一组实验中，如表 3 所示，我们使用经过训练的和经过强化学习微调的指导器，以及 ChatGPT 在少样本设置下生成的指令，评估编辑模型（ChatGPT）的性能。结果表明，当使用 10-shot 提示和来自经过训练的指导器的指令时，ChatGPT 编辑的摘要在事实准确性方面（由 DAE/QFE 衡量）相比原始摘要有显著提升。采用融入 ChatGPT 生成奖励的强化学习后，摘要质量得到进一步提升。此外，我们还进行了使用 ChatGPT 生成的指令的实验。尽管这些指令与人工编写的指令表现出不够理想的对齐效果，但其事实准确性得分却出人意料地高，尤其是按照 QFE 指标衡量时。然而，与其他方法相比，ROUGE 分数明显下降。这些发现表明，ChatGPT 具备生成针对特定且定义明确的方面（例如，解决事实不一致问题）的指令的能力，但可能难以难以准确辨识并满足更广泛的人类期望。

4.2 场景 2：CNNDM 上的覆盖度

ChatGPT 已展现出生成流畅且信息丰富的新闻文章摘要的能力（Goyal 等人，2022）。尽管它擅长生成连贯的摘要，但未必总能像用户所期望的那样，充分覆盖关键主题。为应对这一挑战，我们开展了一项实验，训练并评估一个专门用于指导摘要编辑的指导器模型，以便根据用户历史记录提升知识覆盖度。指导器会预测当前摘要中应添加或删除的关键词，从而提供可执行的指令，使摘要更贴合用户偏好。在实践中，我们根据生成摘要与参考摘要在关键词内容上的匹配程度来评估知识覆盖度。

我们采用 CNNDM 数据集（Nallapati 等人，2016 年）作为本实验的基准。该数据集包含新闻文章与参考摘要配对，其中原始参考摘要作为用户对覆盖度偏好的目标表征。我们承认，根据近期研究（Goyal 等人，2022），CNNDM 数据集中的参考摘要可能存在一些质量局限，例如连贯性较差。不过，本实验主要关注知识覆盖度，而非摘要质量。我们希望评估生成摘要在多大程度上涵盖了参考摘要中的关键实体。

为衡量知识覆盖度，我们引入实体级匹配指标知识 F1。设 E_{gen}为生成的摘要中提及的实体， E_{ref} 为参考摘要中的实体。我们量化二者之间的重合程度

指导器	DAE	QFE	R1	R2	RL
初始摘要	0.699	1.837	76.03	66.34	74.11
FLAN T5（监督学习）	0.772	2.093	72.60	61.96	71.21
FLAN T5（强化学习）	0.803	2.198	74.77	64.73	73.44
ChatGPT（10-shot）	0.834	2.583	*56.54*	*41.29*	*53.06*

表 3：DeFacto 上使用不同指导器生成的指令编辑后的摘要的 ROUGE分数和事实一致性分数。表中所有结果均使用 ChatGPT（10-shot）作为编辑模型。

指导器	知识 F1	R1	R2	RL
初始摘要	44.15	40.28	16.65	33.23
FLAN T5（监督）	47.44	41.04	16.72	33.63
FLAN T5（强化学习）	47.99	41.21	16.80	33.90
ChatGPT（5-shot）*	43.43	39.46	15.43	32.40
预言机	60.80	43.08	18.37	35.24

表4：CNNDM 上由不同指导器生成指令后编辑的摘要的知识覆盖度和 ROUGE 分数。我们使用 ChatGPT（零样本）作为生成器模型（用于生成初始摘要）和编辑模型。* 如果提示中的示例数量超过长度限制（4k 个 token），我们会减少示例数量。

模型	知识 F1	R1	R2	RL
初始摘要	44.15	40.28	16.65	33.23
编辑迭代 1	47.99	41.21	16.80	33.90
编辑迭代 2	48.65	41.18	16.69	33.88
编辑迭代 3	48.99	41.14	16.63	33.83
编辑迭代 1（1&2）	48.08	41.25	16.91	33.94
编辑迭代 2（1&2）	48.87	40.62	16.60	33.45
编辑迭代 3（1&2）	49.20	41.15	16.87	33.86

表 5：在 CNNDM 上进行迭代编辑。第二个部分展示了仅使用第一轮迭代数据微调的模型结果，最下方的部分展示了使用第一轮和第二轮迭代数据微调的模型结果。

两者相乘

$$\text{Knlg F1} = \frac{2\text{Knlg}_\text{p} \times \text{Knlg}_\text{r}}{\text{Knlg}_\text{p} + \text{Knlg}_\text{r}}, \text{ where}$$

$$\text{Knlg}_\text{p} = \frac{|E_\text{ref} \cap E_\text{gen}|}{|E_\text{gen}|}, \quad \text{Knlg}_\text{r} = \frac{|E_\text{ref} \cap E_\text{gen}|}{|E_\text{ref}|}.$$

通过最大化这种重叠，指导器旨在生成能够有效涵盖参考摘要所示相关信息的摘要。

设置： 我们使用 ChatGPT 生成的摘要作为初始摘要。[6]此外，我们使用 FlanT5-large (700M) 作为指导器模型，输入原始文档和 ChatGPT 生成的初始摘要，以预测关键词。监督训练使用预言机关键词列表进行，这些列表指定了要添加和删除的关键词。随后，模型会按照第 3 节所述进行编辑器引导的强化学习微调；为提高效率，使用了数据集中的 10,000 个训练样本子集。

结果： 我们的实验结果见表 4，展示了其有效性我们的指导器模型在提升知识覆盖度方面的作用，体现在实体匹配和 ROUGE 分数上。在零样本设置下，ChatGPT 作为摘要器表现出色。重要的是，在获得 Oracle 指令后，ChatGPT 也展现出可靠的能力，能够根据指定指令纠正和完善初始摘要。

整合由我们训练好的指导模型生成的指令，显著提升了知识覆盖度，表明摘要与用户偏好更加吻合（比较 FLANT5（监督）与初始摘要）。此外，我们发现，强化学习微调过程（FLAN T5（强化学习））进一步提升了模型性能，使各项评估指标取得适度但有意义的提升。

相比之下，在少样本设置中，如果 ChatGPT 不使用经过训练的指导器而直接生成指令（ChatGPT（5-shot）），编辑后的摘要表现会有所下降。具体而言，知识 F1 和 ROUGE 分数均低于初始摘要，这表明 ChatGPT 在生成有效指令以编辑摘要、使其更符合用户偏好方面存在局限。

总的来说，这些发现凸显了.....的价值。

[6]该数据集已发布，可在 GitHub 代码仓库中找到。

初始摘要	一名前惩教官在飞机上被一名年轻男子打了一拳，此前他要求对方停止说脏话。随后，这名前惩教官制服了这名年轻男子，并按住他，直到警察赶到。来源：《每日邮报》
预言机指令 人工撰写的参考摘要	<添加> ChadHurst <删除>《每日邮报》 ChadHurst 来自犹他州盐湖城的 Chad Hurst 被一名飞机乘客迎面打了一拳，当时他们于周日抵达了这座城市。此前，赫斯特曾要求这名年轻人停止使用他们逃离后使用了粗俗语言。赫斯特曾是一名惩教官，随后将其制服。该男子，并将他的双臂反扣在背后，同时等待执法部门到来。The 这名年轻男子尚未被警察公布姓名，被控袭击罪及扰乱公共秩序罪，且醉酒。
预测指令 ChatGPT 编辑后的摘要	<添加> ChadHurst <移除>《每日邮报》 Chad Hurst，一名来自犹他州盐湖城的前惩教官，被一名在飞机上，一名年轻男子在被要求停止使用粗俗语言后，赫斯特平静地拿起了……将这名年轻男子制服并控制住，直到警察赶到。该年轻男子被指控犯有袭击罪和公共场合醉酒。赫斯特曾担任惩教官，这段职业经历使他学会了……绝不还击，而是控制局势并制服对方。

表 6：CNNDM 数据集中的一个示例。

我们的指导器可作为强大的中介，引导 ChatGPT 等大语言模型编辑摘要，使其更贴合用户偏好。

5 讨论

5.1 迭代编辑

除了进行单步编辑外，我们还在 CNNDM 数据集上开展实验，以探索迭代编辑的效果[7]。迭代编辑实验的结果见表5。我们仅使用第一轮迭代的数据进行强化学习（RL）训练，观察到在迭代编辑过程中，编辑后的摘要覆盖率有所提升。我们进一步使用来自第一轮和第二轮迭代的数据混合对模型进行微调，结果显示迭代编辑后的摘要的知识F1值有所提高，性能也因此得到改善。

5.2 定性示例

我们在表格中展示了 CNNDM 数据集的示例6。指导器模型能够正确识别用户的期望并生成编辑指令。ChatGPT 能够根据给定的指令编辑初始摘要，充当编辑器。[8]

[7]我们没有在 DeFacto 数据集上开展类似实验，因为对于大多数数据示例而言，只需一步编辑，就能将初始摘要改为人工编辑后的摘要。[8]DeFacto 的示例见附录。

6 相关工作

6.1 文本编辑

后编辑技术已在各种自然语言处理任务中得到广泛研究，包括句子融合（Malmi 等人，2019）、风格迁移（Reid 和 Zhong，2021年）以及维基编辑（Reid 和 Neubig，2022；Faltings 等人，2021年）。这些方法涉及插入、删除和替换等细粒度定义的操作。然而，它们通常需要大量人工标注数据或复杂的编辑链。相比之下，我们的工作侧重于利用 ChatGPT 等大语言模型的能力，通过自然语言指令进行抽象层面的文本编辑。同样，Liu 等人(2022b) 提出了一种方法，其中采用评论模型生成反馈，并利用编辑模型修订初始摘要。我们通过将这一方法形式化为迭代编辑流程，并结合仅用于推理的语言模型和编辑器引导的指导器，对其进行了扩展和增强。

近期，（Liu 等人，2022a）提出了一种新颖的训练范式，通过由动态规划推导出的编辑链，使生成文本与人类价值观保持一致。然而，该方法需要对语言模型进行额外微调，这对于资源和可访问性有限的模型而言可能并不切实际。

在另一项研究中，Welleck 等人 (2022) 提出了一种框架，将原始生成过程分解为生成器和纠正器两个组件，其中纠正器通过在线训练进行训练，以迭代地优化不完善的生成结果。我们的工作与其不同之处在于，我们将生成过程分解为三个组件：生成器、指导器和编辑器。这

分解使我们能够利用大型模型完成复杂的生成和校正任务，同时使用较小的模型来预测用户特定的编辑指令。

与我们的研究并行，Madaan 等人(2023年) 提出了一种类似的生成流水线，旨在迭代地优化生成的输出。然而，他们的方法有所不同：他们使用同一个大语言模型（配合不同的提示）来生成初始输出、提供反馈，并根据收到的反馈编辑输出，而不考虑任何针对特定用户的反馈。相比之下，本文重点关注在经过训练的指导器引导下，使生成的输出更贴合用户需求。

6.2 大语言模型

自然语言处理领域在大语言模型（大语言模型）方面取得了显著进展（Chowdhery 等人，2022；Zhang 等人，2022；Thoppilan 等人，2022），催生了具有卓越语言处理能力的模型。在这些模型中，GPT 系列（Brown 等人，2020）是一个杰出的代表，因其在不同语言相关任务中的多样化表现而广受认可。

指令微调的引入（Wei 等人，2021年）进一步推动了语言模型的发展，尤其是在使用人类指令进行训练时（Sanh 等人，2021年）。值得注意的是，这种方法带来了显著改进，在零样本和少样本学习场景中尤为突出。InstructGPT（Ouyang 等人，2022）采用了基于人类反馈的强化学习（RLHF）训练范式，体现了这一趋势，使模型能够有效遵循人类指令，并为我们的当前工作奠定了基础。

近期发布的 LLAMA（Touvron 等人，2023年）进一步拓展了这一领域的探索空间，因为研究人员已开始使用 GPT 模型生成的任务增强型数据集来训练或微调模型（Wang 等人，2022）。

不同于上述研究成果，我们的工作提出了三智能体流水线这一新颖范式，充分利用大语言模型处理下游任务的能力。与众不同的是，我们的方法旨在优化性能，同时最大限度地降低计算资源需求，并适应访问受限的情况（例如，仅限 API 的访问）。

6.3 使用大语言模型进行摘要生成

在大语言模型出现之前，文本摘要任务的一种常见方法是使用以任务为重点的目标，在大规模语料库上对模型进行预训练，然后再使用特定任务的数据集进行微调。这种范式在文本摘要方面表现出色，PEGASUS（Zhang 等人，2020）、Primera（Xiao 等人，2021年）和 Z-Code++（He 等人，2022）等模型均采用了这一方法。然而，近期研究（Goyal 等人，2022；Zhang 等人，2023年）表明，在零样本设置下将 GPT-3（Brown 等人，2020）和 InstructGPT（Ouyang 等人，2022）用于新闻摘要任务，所得结果不仅比监督模型的结果更受人工评估者青睐，甚至还优于参考摘要本身。

这些发现为文本摘要任务指明了一个方向。与其在可能并非最优的参考摘要上训练有监督的摘要器，不如利用大语言模型，并着重编辑其输出，使之符合用户需求；这也与本文提出的三智能体流水线相一致。

7 结论与未来工作

本文提出了一种新颖的生成范式，将生成过程分解为三个不同的组件：生成器、指导器和编辑器。我们的方法专门设计用于充分发挥大语言模型的能力，同时兼顾访问权限和计算资源有限等约束，并促进定制生成内容，使其符合用户偏好。通过在文本摘要任务上进行一系列初步实验，我们发现，以 ChatGPT 为代表的大语言模型能够有效地充当编辑器；在提供人工编写的指令时，其性能可媲美监督式编辑模型。然而，要让大语言模型生成与人工编写的指令高度一致的指令，仍然颇具挑战。

为应对这一挑战，我们采用一个更小的模型作为指导器，该模型通过编辑器引导的强化学习（RL）进行训练，其奖励基于编辑后的摘要质量。我们的实验结果证明了该方法的有效性。

这种方法可指导编辑器（ChatGPT）生成更符合用户预期的摘要。

展望未来，后续工作将包括把我们的实验扩展到其他任务，例如维基编辑（Reidand Neubig，2022）、新闻编辑（Spangher et al.，2022）以及数学问题合成（Wellecket al.，2022）。此外，我们还可以使用 Self-instruct 技术（Wangetal.，2022）生成更多指令数据，以训练更好的指导器。

局限性

尽管我们提出的生成流程旨在改善大语言模型输出与用户偏好的对齐，但我们承认，本研究受到资源限制。 因此，我们的实验仅聚焦于 ChatGPT；该模型已在一系列任务中展现出顶尖性能。然而，未来的工作也应探索该流程在其他大语言模型上的适用性和性能。此外，需要注意的是，与所有大语言模型一样，我们系统的输出仍可能存在幻觉和偏见等问题。尽管我们的流程在一定程度上解决了这些问题，但我们无法保证结果完全不含幻觉和偏见。

参考摘要

汤姆·B·布朗、本杰明·曼、尼克·莱德、梅兰妮·苏比亚、贾里德·卡普兰、普拉富拉·达里瓦尔、阿文德·尼拉坎坦、普拉纳夫·夏姆、吉里什·萨斯特里、阿曼达·阿斯克尔、桑迪尼·阿加瓦尔、阿里尔·赫伯特-沃斯、格雷琴·克鲁格、汤姆·亨尼根、雷文·柴尔德、阿迪蒂亚·拉梅什、丹尼尔·M·齐格勒、杰弗里·吴、克莱门斯·温特、克里斯托弗·黑塞、马克·陈、埃里克·西格勒、马泰乌什·利特温、斯科特·格雷、本杰明·切斯、杰克·克拉克、克里斯托弗·伯纳、萨姆·麦坎德利什、亚历克·拉德福、伊利亚·苏茨克维和达里奥·阿莫迪。2020年。语言模型是少样本学习者。*CoRR*，abs/2005.14165。

塞巴斯蒂安·布贝克、瓦伦·钱德拉塞卡兰、罗南·埃尔丹、约翰内斯·格克、埃里克·霍维茨、埃斯·卡马尔、彼得·李、尹泰特·李、袁志力、斯科特·伦德伯格、哈沙·诺里、哈米德·帕兰吉、马尔科·图利奥·里贝罗和张毅。2023年。通用人工智能的火花：GPT-4的早期实验。

AakankshaChowdhery、Sharan Narang、Jacob Devlin、Maarten Bosma、Gaurav Mishra、Adam Roberts、Paul Barham、Hyung Won Chung、Charles Sutton、Sebastian Gehrmann、Parker Schuh、Kensen Shi、Sasha Tsvyashchenko、Joshua Maynez、Abhishek Rao、Parker Barnes、Yi Tay、Noam Shazeer、Vinodkumar Prabhakaran、Emily Reif、Nan Du、Ben Hutchinson、Reiner Pope、James Bradbury、Jacob Austin、Michael Isard、Guy Gur-Ari、Pengcheng Yin、Toju Duke、Anselm Levskaya、Sanjay Ghemawat、Sunipa Dev、Henryk Michalewski、Xavier Garcia、Vedant Misra、Kevin Robinson、Liam Fedus、Denny Zhou、Daphne Ippolito、David Luan、Hyeontaek Lim、Barret Zoph、Alexander Spiridonov、Ryan Sepassi、David Dohan、Shivani Agrawal、Mark Omernick、Andrew M. Dai、Thanumalayan Sankaranarayana Pillai、Marie Pellat、Aitor Lewkowycz、Erica Moreira、Rewon Child、Oleksandr Polozov、Katherine Lee、Zongwei Zhou、Xuezhi Wang、Brennan Saeta、Mark Diaz、Orhan Firat、Michele Catasta、Jason Wei、Kathy Meier-Hellstern、Douglas Eck、Jeff Dean、Slav Petrov 和 Noah Fiedel。2022 年。PaLM：通过路径扩展语言建模。

Hyung Won Chung、Le Hou、Shayne Longpre、Barret Zoph、Yi Tay、William Fedus、Eric Li、Xuezhi Wang、MostafaDehghani、SiddharthaBrahma、Albert Webson、Shixiang Shane Gu、Zhuyun Dai、Mirac Suz- gun、Xinyun Chen、Aakanksha Chowdhery、Sharan Narang、Gaurav Mishra、Adams Yu、Vincent Zhao、Yanping Huang、Andrew Dai、Hongkun Yu、Slav Petrov、Ed. H. Chi、Jeff Dean、Jacob Devlin、Adam Roberts、Denny Zhou、Quoc V. Le 和 Jason Wei。2022 年。 扩展指令微调语言模型。

Alexander Fabbri、Chien-Sheng Wu、Wenhao Liu 和 Caiming Xiong。2022 年。QAFactEval：改进的问答式事实一致性摘要评估。载于 *2022* 年北美计算语言学协会分会会议论文集：人类语言技术，第 2587–2601 页，美国西雅图。计算语言学协会。

Felix Faltings、Michel Galley、Gerold Hintz、Chris Brockett、Chris Quirk、Jianfeng Gao 和 Bill Dolan。2021年。通过命令进行文本编辑。载于*2021*年北美计算语言学协会分会年会论文集：人类语言技术，第 5259–5274 页，线上举行。计算语言学协会。

Tanya Goyal 和 Greg Durrett。2021年。摘要中细粒度事实性的标注与建模。载于 北美计算语言学协会分会 *2021*年会议论文集：人类语言技术。

Tanya Goyal、Junyi Jessy Li 和 Greg Durrett。2022。GPT-3 时代的新闻摘要与评估。

Gu Jiatao、Wang Changhan 和 Zhao Junbo。2019。Levenshtein Transformer。收录于神经信息处理系统进展，第 32 卷。Curran Associates, Inc.。

Pengcheng He、Baolin Peng、Liyang Lu、SongWang、Jie Mei、Yang Liu、Ruochen Xu、Hany Hassan Awadalla、

Yu Shi、Chenguang Zhu、Wayne Xiong、Michael Zeng、Jianfeng Gao 和 Xuedong Huang。2022。Z-code++：一种针对抽象式摘要优化的预训练语言模型。

Ruibo Liu、Chenyan Jia、Ge Zhang、Ziyu Zhuang、Tony Liu 和 Soroush Vosoughi。2022a。第二次思考往往更好：从文本编辑中学习重新与人类价值观对齐。载于神经信息处理系统进展，第 35 卷，第 181–196 页。Curran Associates, Inc.

Yixin Liu、Budhaditya Deb、Milagro Teruel、Aaron Halfaker、Dragomir Radev 和 Ahmed H. Awadallah。2022b。基于自然语言反馈提升摘要的事实一致性。

Aman Madaan、Niket Tandon、Prakhar Gupta、Skyler Hallinan、Luyu Gao、Sarah Wiegreffe、Uri Alon、Nouha Dziri、Shrimai Prabhumoye、Yiming Yang、Sean Welleck、Bodhisattwa Prasad Majumder、Shashank Gupta、Amir Yazdanbakhsh 和 Peter Clark。2023年。自我精炼：通过自我反馈进行迭代优化。

Eric Malmi、Sebastian Krause、Sascha Rothe、Daniil Mirylenka 和 Aliaksei Severyn。2019 年。编码、标记、实现：高精度文本编辑。载于 *2019* 年自然语言处理经验方法会议暨第九届国际自然语言处理联合会议（*EMNLP-IJCNLP*）论文集，第 5054–5065 页，中国香港。计算语言学协会。

Ramesh Nallapati、Bowen Zhou、Cicero dos Santos、Ça˘glar Gulçehre 和 Bing Xiang。2016 年。使用序列到序列 RNN 及其扩展进行抽象式文本摘要。载于第 *20* 届 *SIGNLL* 计算自然语言学习会议论文集，第 280–290 页，德国柏林。计算语言学协会。

Long Ouyang、Jeff Wu、Xu Jiang、Diogo Almeida、Car- roll L. Wainwright、Pamela Mishkin、Chong Zhang、Sandhini Agarwal、Katarina Slama、Alex Ray、John Schulman、Jacob Hilton、Fraser Kelton、Luke Miller、Maddie Simens、Amanda Askell、Peter Welinder、Paul Christiano、Jan Leike 和 Ryan Lowe。2022。通过人类反馈训练语言模型以遵循指令。

Rajkumar Ramamurthy、Prithviraj Ammanabrolu、Kianté Brantley、Jack Hessel、Rafet Sifa、Christian Bauckhage、Hannaneh Hajishirzi 和 Yejin Choi。2023年。强化学习是否适用于自然语言处理：自然语言策略优化的基准、基线与构建模块。

Machel Reid 和 Graham Neubig。2022。学习对编辑过程建模。载于计算语言学协会：*EMNLP 2022* 论文集，第 3822–3832 页，阿联酋阿布扎比。计算语言学协会。

Machel Reid 和 Victor Zhong。2021年。LEWIS：用于无监督文本风格迁移的莱文斯坦编辑。载于计算语言学协会：*ACL-IJCNLP 2021* 论文集，第 3932–3944 页，线上举行。计算语言学协会。

Victor Sanh、Albert Webson、Colin Raffel、Stephen H. Bach、Lintang Sutawika、Zaid Alyafeai、Antoine Chaffin、Arnaud Stiegler、Teven Le Scao、Arun Raja、Manan Dey、M. Saiful Bari、Canwen Xu、Urmish Thakker、Shanya Sharma、Eliza Szczechla、Taewoon Kim、Gunjan Chhablani、Nihal V. Nayak、Debajyoti Datta、Jonathan Chang、Mike Tian-Jian Jiang、Han Wang、MatteoManica、ShengShen、ZhengXin Yong、Harshit Pandey、Rachel Bawden、Thomas Wang、Tr- ishala Neeraj、Jos Rozen、Abheesht Sharma、Andrea Santilli、Thibault Févry、Jason Alan Fries、Ryan Tee- han、Stella Biderman、Leo Gao、Tali Bers、Thomas Wolf 和 Alexander M. Rush。2021年。多任务提示训练实现零样本任务泛化。*CoRR*，abs/2110.08207。

Alexander Spangher、Xiang Ren、Jonathan May 和 Nanyun Peng。2022 年。Newsedits：新闻文章修订数据集与文档级推理挑战。

Romal Thoppilan、Daniel De Freitas、Jamie Hall、Noam Shazeer、Apoorv Kulshreshtha、Heng-Tze Cheng、Alicia Ji n、Taylor Bos、Leslie Baker、Yu Du、YaGuang Li、Hongrae Lee、Huaixiu Steven Zheng、Amin Ghafouri、Marcelo Menegali、Yanping Huang、Maxim Krikun、Dmitry Lepikhin、James Qin、Dehao Chen、Yuanzhong Xu、Zhifeng Chen、Adam Roberts、Maarten Bosma、Yanqi Zhou、Chung-Ching Chang、Igor Krivokon、Will Rusch、Marc Pickett、Kathleen S. Meier-Hellstern、Meredith Ringel Morris、Tulsee Doshi、Renelito Delos Santos、TojuDuke、Johnny Soraker、Ben Zevenbergen、Vinodkumar Prabhakaran、Mark Diaz、Ben Hutchinson、Kristen Olson、Alejandra Molina、Erin Hoffman-John、J osh Lee、Lora Aroyo、Ravi Rajakumar、Alena Butryna、Matthew Lamm、Viktoriya Kuzmina、Joe Fenton、Aaron Cohen、Rachel Bernstein、Ray Kurzweil、Blaise Aguera-Arcas、Claire Cui、Marian Croak、Ed H. Chi 和 Quoc Le。2022 年。Lamda：面向对话应用的语言模型。*CoRR*，abs/2201.08239。

Hugo Touvron、Thibaut Lavril、Gautier Izacard、Xavier Martinet、Marie-Anne Lachaux、Timothée Lacroix、BaptisteRozière、Naman Goyal、Eric Hambro、Faisal Azhar、Aurelien Rodriguez、Armand Joulin、Edouard Grave 和 Guillaume Lample。2023年。Llama：开放且高效的基础语言模型。

Yizhong Wang、Yeganeh Kordi、Swaroop Mishra、Alisa Liu、Noah A. Smith、Daniel Khashabi 和 Hannaneh Hajishirzi。2022。Self-instruct：使用自生成指令对齐语言模型。

Jason Wei、Maarten Bosma、Vincent Y.Zhao、Kelvin Guu、Adams Wei Yu、Brian Lester、Nan Du、An- drew M. Dai 和 Quoc V. Le。2021年。微调后的语言模型是零样本学习者。*CoRR*，abs/2109.01652。

Sean Welleck、Ximing Lu、Peter West、Faeze Brahman、Tianxiao Shen、Daniel Khashabi 和 Yejin Choi。2022 年。通过学习自我纠正来生成序列。

Wen Xiao、Iz Beltagy、Giuseppe Carenini 和 Arman Cohan。2021年。PRIMER：面向多文档摘要的金字塔式掩码句子预训练。*CoRR*，abs/2110.08499。

Jingqing Zhang、Yao Zhao、Mohammad Saleh 和 Peter J. Liu。2020。Pegasus：通过提取间隙句子进行抽象式摘要的预训练。载于第*37*届国际机器学习大会论文集，ICML'20。JMLR.org。

Susan Zhang、Stephen Roller、Naman Goyal、Mikel Artetxe、Moya Chen、Shuohui Chen、Christopher De-wan、Mona Diab、Xian Li、Xi VictoriaLin、Todor Mi-haylov、Myle Ott、Sam Shleifer、Kurt Shuster、Daniel Simig、P unit Singh Koura、Anjali Sridhar、Tianlu Wang 和 Luke Zettlemoyer。2022 年。Opt：开放式预训练 Transformer 语言模型。

Tianyi Zhang、FaisalLadhak、Esin Durmus、Percy Liang、Kathleen McKeown 和 Tatsunori B. Hashimoto。2023年。新闻摘要大语言模型基准测试。

A 提示

我们在表7和表8中分别展示了用于摘要编辑和指令生成的提示。

CNNDM
摘要： [初始摘要] 文档： [文章] 为文档 [指令]重写摘要 新摘要：
DeFacto
文档： [article] 摘要： [初始摘要] 指令： [指令] 仅按照指令编辑摘要，并且 仅输出修正后的摘要。 新摘要：

表 7：用于摘要编辑的提示。

CNNDM
少样本提示 $\times N$，直至达到长度上限 文档： [文章]$_i$ 摘要： [初始摘要]$_i$ 指令： [指令]$_i$ 文档： [文章] 摘要： [初始摘要] 摘要可能未涵盖显著内容，生成 指令，使摘要聚焦于显著 内容。指令应从以下格式中 选择： 删除与以下内容相关的内容 __. 添加与.....相关的内容 __. 无需进行任何操作。 仅输出指令，不要输出修正后的摘要， 并谨慎地制定指令。 指令：
DeFacto
少样本提示 $\times 10$文档： [文章]$_i$摘要： [初始摘要]$_i$ 该摘要可能包含一些事实性错误，请生成用于纠正 摘要的指令。指令：
文档： [文章] 摘要： [初始摘要] 摘要可能包含一些事实性错误，请生成 用于更正摘要的指令。 指令应从以下 格式中选择： 从摘要中删除有关__的信息。 将有关 __的信息添加到摘要中。 将有关 __ 的信息替换为该信息 关于 __. 修改摘要中有关 __的信息。 通过__彻底重写摘要。 仅输出指令，不要输出更正后的总和。 摘要，并使指令更加保守。 指令：

表 8：用于指令生成的提示

B 定性示例

我们在表 6中展示了来自 DeFacto 数据集的示例。指导器模型能够正确识别用户的期望并生成编辑指令。ChatGPT 能够根据给定指令编辑初始摘要，充当编辑器。

初始摘要	备受争议的库丹库拉姆核电站位于印度泰米尔纳德邦，已开始发电。
人工撰写的指令	从摘要中删除印度泰米尔纳德邦的地理位置信息。
人工编辑后的摘要	备受争议的库丹库拉姆核电站已开始发电。
预测的指令	从摘要中删除有关泰米尔纳德邦的信息。
ChatGPT 编辑后的摘要	备受争议的库丹库拉姆核电站已开始发电。
初始摘要	科特迪瓦已传出枪声第二大城市布瓦凯的士兵发生骚乱一天后因薪酬问题发动兵变
人工撰写的指令	从摘要中删除有关第二个的信息。
人工编辑的摘要	科特迪瓦城市布瓦凯传出枪声，此前一天，士兵因.....发生兵变。pay.
预测的指令	从摘要中删除关于第二个的信息。
ChatGPT 编辑后的摘要	科特迪瓦城市布瓦凯传出枪声，士兵因.....发生哗变的一天后pay.

表 9：DeFacto 数据集中的示例。

C 软件与许可证

我们的代码采用 Apache License 2.0 许可证。
我们的框架依赖项包括：

- HuggingFace 数据集[9]，Apache 2.0
- NLTK[10]，Apache 2.0许可证
- Numpy[11]，BSD 3-Clause“新修订版”
- Transformers[12]，Apache 2.0
- PyTorch[13]，杂项
- ROUGE[14]，Apache 2.0
- FLAN T5[15]，Apache 2.0
- ChatGPT[16]，专有

[9] https://github.com/huggingface/datasets/blob/master/LICENSE [10] https://github.com/nltk/nltk [11] https://github.com/numpy/numpy/blob/main/LICENSE.txt [12] https://github.com/huggingface/transformers/blob/master/LICENSE [13] https://github.com/pytorch/pytorch/blob/master/LICENSE [14] https://github.com/google-research/google-research/tree/master/rouge [15] https://huggingface.co/google/flan-t5-large [16] https://openai.com/chatgpt