融合提示学习的故事生成方法

倪宣凡　李丕绩*
计算机科学与技术学院/人工智能学院,
南京航空航天大学
江苏省，南京市，210016
xuanfanni@gmail.com, pjli@nuaa.edu.cn

摘要

开放式自动故事生成通过输入故事的开头、大纲、主线等，得到具有一致性、连贯性和逻辑性的故事。现有的方法想要提升生成故事的质量，往往需要大量训练数据和更多参数的模型。针对以上问题，该文利用提示学习在零样本与少样本场景下的优势，同时使用外部常识推理知识，提出了一种故事生成方法。该方法将故事生成分为三个阶段：输入故事的开头，常识推理模型生成可能的事件；根据类型不同，将事件填入问题模板中，构建引导模型生成合理回答的问题；问答模型产生对应问题的答案，并选择困惑度最小的作为故事下文。重复上述过程，最终生成完整的故事。自动评测与人工评测指标表明，与基线模型相比，该文提出的方法能够生成更连贯、具体和合乎逻辑的故事。

关键词： 故事生成 ； 预训练模型 ； 提示学习

A Story Generation Method Incorporating Prompt Learning

Xuanfan Ni, Piji Li
College of Computer Science and Technology/Artificial Intelligence,
Nanjing University of Aeronautics and Astronautics
Nanjing, Jiangsu 210016, China
xuanfanni@gmail.com, pjli@nuaa.edu.cn

Abstract

Open-ended automated story generation obtains a consistent, coherent and logical story by entering the beginning, out-line, or main line of story. To improve the quality of generated stories, existing methods often require a large amount of training data and models with more parameters. Aiming at the above problems, this paper proposes a novel story generation method. The method divides the story generation into three stages: input the beginning of story, and the common sense reasoning model generates possible events; according to different types, fill in the events into the question template to construct questions; the question answering model generates answers to corresponding questions, and selects the one with lowest PPL score as story below. Repeat above process to finally generate a complete story. Automatic evaluation metrics show that the proposed method is able to generate more coherent, specific, and logical stories than baseline models.

Keywords: story generation , pre-trained model , prompt learning

* 通讯作者

第二十一届中国计算语言学大会论文集，第166页-第177页，南昌，中国，2022年10月14日至16日。

1 引言

开放式自动故事生成是自然语言处理(Natural Language Processing, NLP) 领域中一个非常经典的任务(Alabdulkarim et al., 2021)。在深度神经网络出现之后，特别是随着更大参数、更好架构的模型被提出，故事生成也取得了长足的发展。故事生成与其它的自然语言生成任务不同。机器翻译需要源语言与目标语言之间的匹配；文本摘要需要抽取输入文本的重要信息并进行填充，句子的结构、逻辑和语义大部分来自输入；而故事生成考验的是模型从训练故事中学到的知识，同时要兼顾连贯性和一致性。现在主流的做法都是探究故事的结构，由点到面，逐步生成(Ansag and Gonzalez, 2021)。但是考虑的角度不同、结构不同，性能会有很大的差异。

现有的故事生成模型或系统能够在连贯性和一致性上取得不错的效果，如采用Vaswani et al. (2017)提出的Transformer架构的GPT(Radford et al., 2018)，GPT2(Radford et al., 2019)，GPT3(Brown et al., 2020)等自回归语言模型；Liu et al. (2020)提出以角色为中心的神经故事模型；Tambwekar et al. (2018)训练实现给定故事目标或结局的神经语言模型；Fan et al. (2018)将故事生成分层，并训练模型定期给出指导。Yao et al. (2019)则在此基础上，使用高级故事生成计划来引导模型进行生成。这些工作往往需要大量的训练数据和结构复杂、参数量多的模型，在很多场景下是难以满足的。针对这一问题，使用预训练模型和注入外部知识来辅助生成是很好的解决方法：Ammanabrolu et al. (2021)通过常识推理、因果关系和情节顺序来构建故事生成系统，其中常识推理由COMET模型(Bosselut et al., 2019)给出；Guan et al. (2020)提出了知识增强的预训练模型，利用来自外部知识库的常识知识来生成合理的故事。但是，以前的研究更多是将这些外部知识作为数据，参与模型的训练。那么，是否有更好的外部知识使用方法?

最近，提示学习(Prompt Learning)的相关研究与应用发展的如火如荼(Liu et al., 2021)。大量工作都表明，提示学习在少样本和零样本场景下有着一般微调(Finetune)所不及的优势。例如，有一个文本情感分类的任务，对于输入“我爱这里的食物。”，去判断这句话的情感是积极的还是消极的。提示学习将输入重构成“我爱这里的食物。我觉得...”，然后交由预训练模型，如GPT-2等去生成。正面内容代表积极，负面内容代表消极。在这个例子中，重构后的输入被用来引导模型进行生成，原先的文本分类任务则被转化为文本生成任务。

仅仅更改输入的形式，在没有微调的情况下，预训练模型就可以执行不同于训练阶段输入的数据形式的下游任务。受提示学习的启发，我们将其应用到故事生成中，通过提示模板来重构任务形式，在少样本和零样本场景下，保证生成故事的质量。重构成何种任务形式是我们非常关注的一点。直觉上来讲，越相近的任务，重构的难度就越低，效果也越好。因此我们考虑同为文本生成任务的问答。相比较原先的故事生成任务与其他文本生成任务（文本摘要、机器翻译等），问答的优势有：

- 问答任务的输入通常是文档与问题，模型从文档与训练时学到的知识来对问题进行回答。将故事前文作为文档，通过模板构造合理的问题。生成的答案可以较好地保持一致性与连贯性；
- 在外部常识知识的帮助下得到的问题，可以看作是对上文的总结与后续发展的推测。使用这种优质的问题去引导问答模型进行回答，能够在保证逻辑性的情况下，推动故事发展；
- 应用于故事生成任务的自回归预训练模型，如果没有充足的训练数据，产生的内容很容易出现冗余、逻辑错误等情况；机器翻译、文本摘要任务都只能对输入进行总结与分析，其输出无法推动故事发展；而应用于问答任务的预训练模型，通过优质的问题进行引导，生成的内容是简短且与上文相关的答案，将这些答案作为故事下文能很大程度上避免这些不足之处。

通过以上思考，本文提出了一种故事生成方法。该方法将故事生成分为三个阶段：(1) 输入故事的开头，常识推理模型生成多个可能的事件；(2) 根据类型，将事件填入问题模板中，构建合理的问题；(3) 问答模型产生对应问题的答案，并选择困惑度最小的作为故事下文。重复上述过程，最终生成完整的故事。

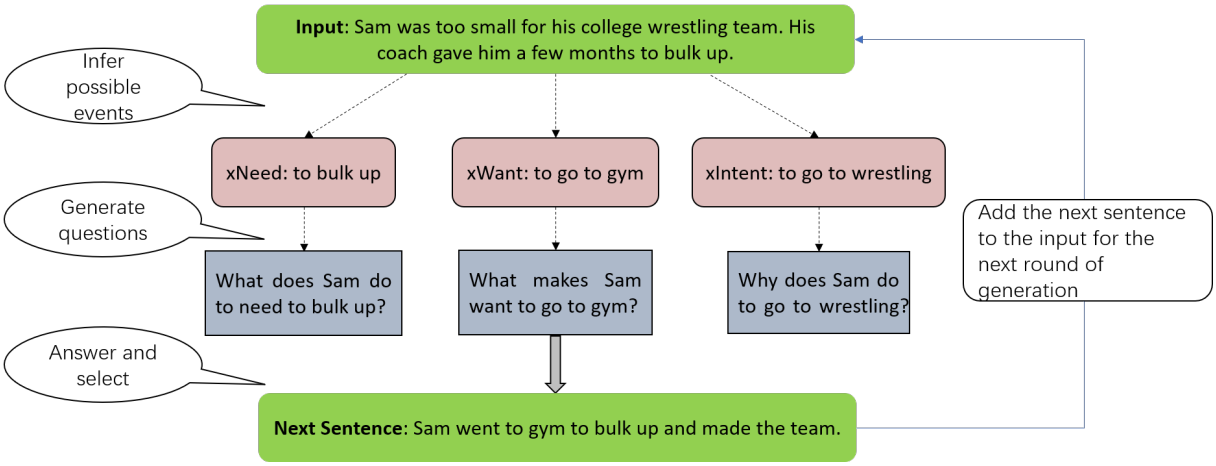


Figure 1: 融合提示学习的故事生成方法的示例

图1展示了本文提出的故事生成方法的一个样例。如图所示，对于输入的故事开头*Sam was too small for his college wrestling team. His coach gave him a few months to bulk up.* 常识推理模型生成多个可能的事件，如*to bulk up, to go to gym, to go to wrestling*等。这些事件有各自的类型，且与故事开头紧密相连。根据事件类型，构建一系列问题，对应上文三个事件，则为：*What does Sam do to need to bulk up? What makes Sam want to go to gym? Why does Sam do to go to wrestling?* 针对这些问题，模型生成答案，并选择句子*Sam went to gym to bulk up and made the team.*作为下文故事。通过优质的问题模板、合理的事件推理，这些问题能够很好地引导问答模型进行生成。最终，从生成答案中选择困惑度最低的作为故事下文。

我们选择Para-COMET(Gabriel et al., 2021)作为常识推理模型，为选定句子生成推理的同时结合故事中的其他句子；针对Para-COMET产生的事件，我们使用RoBERTa模型(Zhuang et al., 2021)找出与之对应的人物角色，并根据事件类型，构建提示问题模板，生成问题；我们使用ELI5QA模型(Fan et al., 2019a)和BART模型(Lewis et al., 2019)来对问题进行回答，并使用GPT2模型计算答案的困惑度。

本文的主要创新点有：

- 针对深度神经网络模型的训练缺少数据集的问题，本文提出的方法结合了新兴的提示学习思想，通过构建优质的提示模板，充分激发预训练模型的潜能，帮助模型回忆起训练时学到的知识，来较好地完成下游任务；
- 在故事生成过程中，通过常识推理构建的问题，可以看做是对前文内容的多角度、多方面的总结，并引导模型产生合理的下文句子；
- 本文在零样本与少样本场景下进行了实验，评测结果证明了该故事生成方法的有效性。进一步的消融实验突出了优质提示问题模板的重要性。

2 相关工作

2.1 故事生成方法

Liu et al. (2020)提出了以角色为中心的故事生成神经模型。其做法是，为一个故事分配一个角色，在上下文环境下生成该角色的一系列动作，最终生成完整的故事。由于在故事生成的每个阶段，给定的角色都参与选择动作，因此生成的故事具有很好的一致性。Fan et al. (2019b)为了解决长文本故事的一致性问题，提出了一种结构化故事生成模型。模型对动作序列、故事叙述以及命名实体进行建模。模型产生实体匿名故事，并用之前识别出来的实体去替换故事中的占位符。从不同角度来解构故事，在保证连贯性和一致性的前提下，将故事文本拆分细化，并对不同组成成分进行建模。本文提出的方法也将故事生成任务分解为多个阶段，每个阶段都使用不同的预训练模型，执行不同的任务。

Tambwekar et al. (2018)通过控制结局和事件顺序来控制故事情节。他们使用强化学习技术来优化预训练的序列到序列模型。他们的方法比单独的基础模型要好。但是，这种方法需要针对每个新的下游任务重新训练模型。Fan et al. (2018)将生成过程分为两个层次：前提和故事，来解决情节可控性问题。他们使用卷积网络首先生成一个写作提示，然后，该提示成为序列到序列模型的输入，并指导它生成以提示为条件的故事。这种方法通过直接编写故事提示，来适应不同的任务，但是编写的提示较为单调，这导致生成的故事缺乏趣味性。Yao et al. (2019)提出了计划写作(Plan-and-write)故事生成框架：该框架将故事的标题作为输入，然后生成故事情节。接着将故事情节和标题用作输入以控制在序列到序列模型中的故事生成。该模型存在几个主要问题：重复、偏离主题和逻辑不一致。这些模型采用分层故事生成，需要大量训练数据，且生成的内容往往会存在一定的缺陷。与之形成对比的，本文提出的方法在零样本场景下也能生成较好的故事。

Ammanabrolu et al. (2021)通过常识推理、因果关系和情节顺序来构建故事生成系统C2PO。他们将故事生成问题视为情节填充，从训练集中中提取情节点的轮廓，然后对其详细说明。在C2PO系统中使用软因果关系填充情节来生成叙事——创建一个可能的故事延续的分支空间，从COMET(Bosselut et al., 2019) 常识推理模型中迭代地提取常识因果推理。Guan et al. (2020)提出了知识增强预训练模型，利用来自外部知识库的常识知识来生成合理的故事。他们将常识知识编码，与大规模语料一起输入进Transformer模型中进行训练。还有很多类似的工作，都是将外部知识作为增强模型性能的手段。我们的工作则使用外部知识直接参与故事生成，同样为了提高生成故事的合理性与逻辑性。

2.2 提示学习

Liu et al. (2021)完成了一篇综述论文。论文总结了近几年提示学习的相关工作，并提出NLP中的新范式：预训练、提示、预测(Pre-train, Prompt, Predict)。他们从五个方面对提示学习方法做了一个介绍：预训练模型(Pre-trained Models)、提示工程(Prompt Engineering)、答案工程(Answering Engineering)、多提示学习(Multi-Prompt Learning)以及基于提示的训练策略(Prompt-based Training Strategies)。

在这篇综述之前，就有工作涉及到这种概念，也表现出提示学习的一些潜力。Li and Liang (2021)提出了前缀调优(Prefix-tuning, PT)。PT不改变模型参数，只是对不同的下游任务训练不同的连续向量，这个向量被称为前缀(Prefix)。在文本摘要任务中，前缀，输入，输出一起拼接，然后交由GPT2模型去训练。PT是连续提示的一种，这体现了提示模板形式的多样性：不一定是离散token，也可以是数字，符号，向量，词嵌入甚至是图片、音频、视频(已有工作将图片提示模板应用到计算机视觉中)。

在此之后，提示学习在文本生成领域的应用也被广泛探索。Castricato et al. (2021)通过模板构建提示，从后往前生成以目标事件为结尾的故事，这里的提示模板是对已生成内容的解释。而我们的工作构建的问题是对前文内容的总结，并进行推理生成。Lin et al. (2022)通过训练好的常识推理模型，生成对应输入事件的常识知识提示，来引导未来事件生成，提示形式是潜在的常识表征。我们的工作使用的提示是通过将推理事件填入问题模板中得到的，是具体的内容，相比之下具有更好的可解释性与可控性。

3 结合提示学习的故事生成

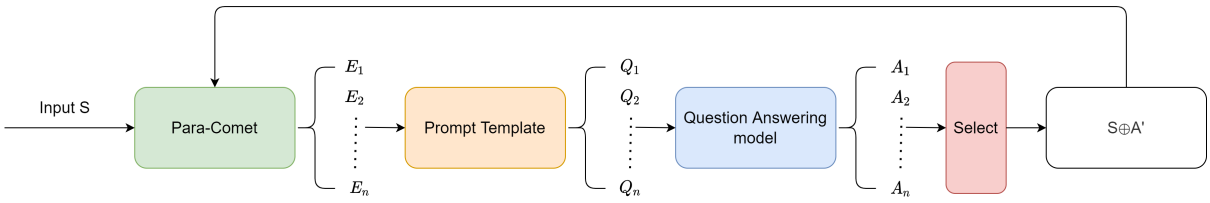


Figure 2: 故事生成方法的工作流程

本节介绍故事生成方法的工作流程。它由三个阶段组成，分别是事件推理、提示问题生成、答案生成与选择。其中每个阶段都使用了不同的预训练语言模型。整个流程通过优质的提

示问题模板，将不同模型结合起来，执行故事生成任务。如图2所示，对于输入S，常识推理模型生成n个可能的事件$E_1, E_2, \ldots, E_n$；接着用这些事件构造对应的问题$Q_1, Q_2, \ldots, Q_n$，并用问答模型进行回答，得到答案$A_1, A_2, \ldots, A_n$；选择困惑度最小的答案A'，作为故事下文。与开头合并为$S \oplus A'$，作为新一轮生成的开头。

3.1 事件推理

事件推理阶段使用Gabriel et al. (2021)提出的Para-COMET预训练模型，根据输入的故事开头，产生每一句对应的常识推理(Commonsense Inference)，这些常识推理表现为接下来可能发生的事件。常识推理一直被视为优质外部知识，来辅助文本生成。有了外部知识注入，模型便不会局限于输入的文本，而能从更符合人类社会常识的角度进行生成。

如图1 所示，对应输入的故事开头，模型生成如*to bulk up, to go to gym, to go to wrestling*等事件，这些事件有9种类型，如表1所示。本方法选择前六种用于后续生成。

Table 1: 事件的种类以及含义

Type	**Dimension**	**Template**
Causes	xIntent	PersonX wanted [ ]
	xNeed	PersonX needed [ ]
	xAttr	PersonX is seen as [ ]
Effects	xWant	PersonX wants [ ]
	xEffect	PersonX is likely [ ]
	xReact	PersonX then feels [ ]
	oWant	PersonY wants [ ]
	oEffect	PersonY is likely [ ]
	oReact	Others then feels [ ]

3.2 提示问题生成

由Para-COMET得到的事件不会显示与之对应的人物角色。*to bulk up, to go to gym, to go to wrestling*等事件的形式为to do结构。因此我们需要先通过预训练模型，得到对应事件的角色，然后将事件与角色填入提示问题模板中，生成最终问题。

3.2.1 链接角色与事件

我们初步的方案是使用BERT(Devlin et al., 2018)预训练模型。BERT是掩码语言模型，使用了Transformer模型的编码器层。BERT模型在完形填空任务上的表现非常优秀，因此我们将联系事件与角色的任务重构为完形填空的形式。以图1中的*to go to gym*为例，它的类型为**xWant**，那么我们将其重构为：

[MASK] wants to go to gym.

并将前文已生成的内容拼接在输入之前。模型生成的内容即为角色。但是，[MASK]的大小仅为一个单词。而很多故事中，角色的名字或代称会出现超过一个单词的情况，例如*Her mother, Daniel Wicky, My dog*等等。此时BERT会生成*She, He, It*等词，这些词虽然不会带来一致性与逻辑性的问题，但会大大降低故事的流畅性和多样性。

我们第二个考虑的模型是在Rajpurkar et al. (2016)提出的斯坦福大学问答数据集(The Stan-ford Question Answering Dataset, SQuAD)上训练的RoBERTa模型(Zhuang et al., 2021)。SQuAD是一个阅读理解数据集，给定一篇文章，准备相应问题，并给出问题的答案。相比填空，问答无疑更加自由，生成的内容也能应对大部分情况。依然以*to go to gym*为例，将其重构为：

Who wants to go to gym?

同样将前文以生成的内容拼接在问题之前。模型回答的内容经过筛选与清洗，得到与事件对应的角色。在本例中对应*to go to gym*的角色为*Sam*.

3.2.2 根据模板得到问题

有了事件与角色之后，根据不同的事件类型，我们将其填入问题模板之中以生成对应的问题。联系模板与生成模板如表2所示。

Table 2: 链接模板与问题模板

Event Type	Association Template	Question Template
xIntent	Who needs to [event]?	Why does [character] do [event]?
xNeed	Who needs [event]?	What does [character] do to need [event]?
xAttr	Who might be described [event]?	Why does [character] be [event]?
xEffect	Who [event]?	What makes [character] [event]?
xReact	Who feels [event]?	What makes [character] feel [event]?
xWant	Who may want [event]?	What makes [character] want [event]?

这样，由**xWant**类型的事件*to go to gym*，得到最终问题：

What makes Sam want to go to gym?

3.3 答案生成与故事选择

答案生成阶段，将问题生成模块中得到的问题输入进预训练的问答模型中，来让模型回答。本文选用了两种预训练模型：ELI5QA(Fan et al., 2019a)和BART(Lewis et al., 2019)。

3.3.1 使用ELI5QA模型进行回答

ELI5QA模型是一个在Fairseq-py框架下训练的、长文本形式的问答模型。它在ELI5数据集上进行训练。ELI5数据集全称是*Explain Like I'm Five*，从Reddit社区语料库收集。在这个数据集中，人们对开放式问题给出长而容易理解的答案，就像给五岁的孩子一样。

ELI5QA模型的输入是问题和文档，模型会从文档和训练得到的知识中生成对问题的回答。在本方法中，为了尽可能保证生成故事的一致性与连续性，我们将故事开头和已生成的内容作为文档，和问题一起输入进模型中。

问题模板的构建初衷是为了从ELI5QA模型中得到简短且相关的句子，作为故事的下文。为了保证这一点，ELI5QA模型采用Top-k采样算法进行生成。k设置过小则会容易生成更平淡或泛的句子，当k很大的时候，候选集合会包含一些不合适的token。我们取$k = 50$。

即使使用了Top-k 采样算法，生成的答案仍有一定概率出现无意义或重复的句子，因此我们需要对答案进行一定程度的清洗。采取的策略是：收集一些禁止短语，组成集合。我们舍弃掉那些包含禁止短语集合中的元素的句子以及它后面的所有句子。若首句长度不足6，我们也舍弃掉首句内容，并将剩余的句子添加到候选项中。这是因为首句有可能是对问题的*yes, no, Of course*这种回答，这些对故事生成没有帮助。这些禁止短语是生成样例中经常出现的无意义的词语、不友好的内容。通过这样的筛选，就能得到对应每个问题的答案集合。

3.3.2 使用微调的BART模型进行回答

除了ELI5QA模型外，我们还选用了BART模型作为生成答案的预训练模型。但是，原始的BART模型在执行QA任务时的效果不如人意，因为在训练时，数据的形式并非问答语料。因此我们将它在ROCStories数据集(Mostafazadeh et al., 2016)上进行微调，以提升生成效果。ROCStories数据集是常识性短篇小说的集合，包含100,000个五个句子的故事。每个故事都遵循一个日常主题。这些故事包含了日常事件之间的各种常识性因果关系和时间关系。我们将数据集按照70% : 15% : 15%的比例随机划分成训练集、验证集和测试集。对于训练集和验证集中的每一个故事，我们进行如下处理：

1. 对于每个故事中的一至四句，我们都遵循事件推理阶段和提示问题生成阶段，生成20个问题；

2. 当前句子以及它之前的所有句子作为Document，和问题以如下方式拼接起来：

   Question –T– Document

   拼接后的内容作为对应关键字Q的值；

第二十一届中国计算语言学大会论文集，第166页-第177页，南昌，中国，2022年10月14日至16日。

3. 当前句子后面的所有句子作为对应关键字A的值，和上文的关键字Q组成一个字典，存入jsonlines文件中。

这样就得到了训练集和验证集，设置学习率为$2e-5$，batch size为16，对BART进行训练。在解码时，将问题和前文内容拼接成训练集数据的形式，选用Top-k算法，设置$k=50$，进行生成。

3.3.3 选择答案作为故事下文

得到对于每个问题的答案集合之后，我们将这些集合合并，对其中的每个元素，我们都将它拼接在故事前文后，并使用在科幻摘要语料库上(Ammanabrolu et al., 2020)微调的GPT2模型来计算其困惑度（Perplexity, PPL），选择使拼接后困惑度最小的元素作为下文。该数据集由来自科幻电视和维基电影的2276个高质量情节摘要组成。这样做的目的是，选择那些更接近情节描述的答案作为故事下文。

4 实验设置

4.1 数据集

本文选取ROCStories数据集用于实验。训练集、验证集、测试集的划分与节3.3.2中一致。在训练时，使用ROCStories的训练集和验证集；在生成时，输入测试集中每个故事的前两句，模型或系统生成后三句。

4.2 指标

本文使用自动指标来评估实验结果，指标包括自动评测指标与人工评测指标。

4.2.1 自动评测指标

- 困惑度(**PPL**)：计算输入句子的指数平均负对数似然，评估文本流畅度和贴近人类语言的程度；
- 双语评估替补(Bilingual Evaluation Under-study，**BLEU** (Papineni et al., 2002))：计算生成句子和实际句子的N-grams，然后统计其匹配的个数；
- 基于召回率的主旨评估(Recall-Oriented Understudy for Gisting Evaluation, **ROUGE-L** (Lin, 2004))：计算最长公共子序列的重合率；
- 使用显式排序评估翻译的指标(Metric for Evaluation of Translation with Explicit Ordering, **METEOR** (Banerjee and Lavie, 2005))：基于单精度的加权调和平均数和单字召回率；
- 基于共识的图像描述评估(Consensus-based Image Description Evaluation, **CIDEr** (Vedantam et al., 2015))：利用TF-IDF来对不同N-gram赋予不同的权重；
- 使用BERT计算句子相似度得分的指标**BERTScore** (Zhang et al., 2019)
- 衡量文本多样性指标**Distinct-n**；(Li et al., 2015)：计算所有生成的文本中不同的n-gram的比率。

4.2.2 人工评测指标

- 连贯性得分(Coherence)：$0 \rightarrow 10$分，0分代表生成的文本完全无法阅读，10分代表生成的文本连贯清晰；
- 一致性得分(Consistency)：$0 \rightarrow 10$分，0分代表生成的文本和前文没有任何关联，10分代表生成的文本和前文叙事内容保持完全一致；
- 逻辑性得分(Logical)：$0 \rightarrow 10$分，0分代表生成的文本没有任何逻辑可言，10分代表生成的文本完全符合故事描述背景下的底层社会逻辑。

Table 3: 零样本场景实验结果

Models	PPL↓	BLEU-1↑	BLEU-4↑	DIST↑	METEOR↑	CIDEr↑	Rouge-L↑	BERTScore↑
GPT2	**10**	0.308	0.193	**0.950**	0.211	0.483	0.345	0.886
Ours wELI5	16	**0.386**	**0.282**	0.933	**0.230**	**1.151**	**0.384**	**0.891**

Table 4: 充足数据场景实验结果

Models	PPL↓	BLEU-1↑	BLEU-4↑	DIST↑	METEOR↑	CIDEr↑	Rouge-L↑	BERTScore↑
Finetune BART	11	0.474	0.358	0.918	0.251	2.459	0.469	0.912
Finetune GPT2	18	0.228	0.118	0.958	0.128	0.419	0.232	0.866
XLNET	14	0.266	0.144	**0.960**	0.190	0.056	0.299	0.878
HINT	11	0.440	0.283	0.916	0.232	1.566	0.425	0.919
Ours wBART	**10**	**0.493**	**0.376**	0.937	**0.262**	**3.058**	**0.475**	**0.923**

4.3 基线模型

本文实验选用的基线模型包括：

- GPT2：使用Transformer模型的解码器层
- BART：双向自回归的Transformer模型
- XLNET(Yang et al., 2019)：基于广义自回归预训练的双向模型
- HINT(Guan et al., 2021)：通过在解码过程中表示句子级别和语篇级别的前缀句子来生成连贯的文本

5 实验结果

5.1 零样本场景实验

本小节介绍在零样本场景下开展的实验，来说明本文提出的故事生成方法可以在没有训练数据的情况下生成不错的故事。实验选择预训练的GPT2作为基线模型，选择ELI5作为答案生成阶段的模型，比较二者生成的故事的各项指标得分。

如表3中的结果所示：

- 使用ELI5作为答案生成阶段的模型，在测试集上生成的故事的BLEU-1，BLEU-4，METEOR，CIDEr，Rouge-L，BERTScore指标得分要更高，说明其生成的故事比GPT2更贴近原故事。没有训练任何模型，仅仅通过构造合理提示问题模板，一个问答模型也能应用于故事生成任务中，并且有不错的表现；
- GPT2模型在DISTINCT指标上的得分保持领先，可能有两个原因：ELI5模型在处理冗余与重复时的能力不及GPT2模型；设计的问题模板没有达到最优；
- GPT2模型在PPL指标上的得分保持领先，这可能是由于预训练GPT2时，模型的参数数量与复杂度、训练数据等要超过ELI5QA模型，这使得GPT2模型生成的故事能够更为贴近人类语言。

5.2 充足数据场景实验

本小节介绍在充足数据场景下开展的实验，来说明本文提出的故事生成方法的性能也能随着数据量的增加而提升。实验选择在ROCStories的训练集和验证集上微调的BART、GPT2、XLNET作为基线模型，选择按照节3.3.2微调的BART作为答案生成阶段的模型，比较二者生成的故事的各项指标得分。

如表4中的结果所示，有了训练数据后，本文提出的故事生成方法的各项指标得分，相比零样本场景下的结果，都有了大幅提升，且能超过仅仅在数据集上微调的单个BART模型；相同训练数据下，本文提出的故事生成方法依然可以在多数指标上的得分保持对基线模型的优势。

第二十一届中国计算语言学大会论文集，第166页-第177页，南昌，中国，2022年10月14日至16日。

Table 5: 消融实验结果

Models	PPL↓	BLEU-1↑	BLEU-4↑	DIST↑	METEOR↑	CIDEr↑	Rouge-L↑	BERTScore↑
Ours w Prompt	**10**	**0.493**	**0.376**	**0.937**	**0.251**	**3.058**	**0.475**	**0.923**
Ours w/o Prompt	23	0.410	0.339	0.932	0.247	1.393	0.474	0.916

Table 6: 人工评测实验结果

Type	Models	Coherence↑	Consistency↑	Logical↑
Zero-shot	GPT2	5.43	5.20	5.33
	Ours wELI5	5.32	4.21	5.71
Enough Data	Finetune BART	5.76	5.45	5.80
	Ours wBART	**6.60**	**6.42**	**6.62**

5.3 消融实验

本小节介绍消融实验，来说明设置优质的提示模板能够为故事生成带来提升。实验分别选择按照节3.3.2微调的BART和训练过的不使用提示学习方法的BART模型作为答案生成阶段的模型。训练数据收集时，跳过提示问题生成阶段，直接将前文内容与事件拼接。

如表5中的结果所示，使用提示学习的故事生成方法的各项指标得分均超过不使用提示学习的故事生成方法。其中尤以PPL，BLEU-1，CIDEr指标提升幅度大，说明使用提示学习构建问题模板后，生成的句子比直接拼接要更贴近原文故事，且更为接近人类语言。具体原因有以下几点：

- 事件是短语结构，且没有对应的人物角色。直接使用时，对下文句子的影响很小，甚至会带来副作用。整个系统退化成了Encoder-Decoder框架，问题则变成输入故事前两句，BART模型输出后三句，且完全依赖模型本身的性能；
- 有了提示模板构建的问题，事件便不会独立于输入的故事前文，外部常识推理知识才能够帮助提升生成故事的逻辑性。

5.4 人工评估实验

本小节介绍人工评估实验，来更好地支撑通过自动评测指标得出的结论。实验从测试集中随机抽取100个故事开头，并由模型或系统生成后续故事。志愿者对这些故事从连贯性、一致性与逻辑性的角度进行打分，并计算平均得分。志愿者不会被告知故事的来源。

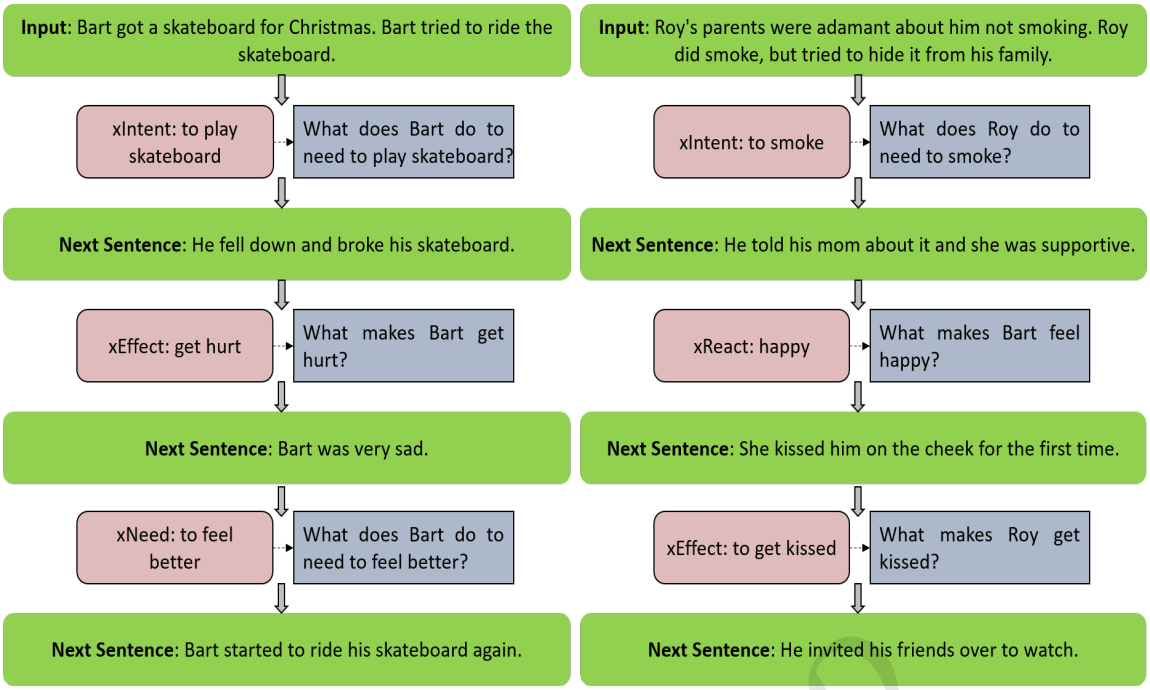
如表6中的结果所示，在零样本场景下，使用ELI5作为答案生成阶段模型的故事生成方法能在连贯性和逻辑性方面接近GPT2模型，但在一致性方面却有所不及。这可能是由于ELI5模型无法完全理解通过提示构建的问题，从而产生无意义、与前文无关或不连贯的回复。而在充足数据的场景下，使用微调BART作为答案生成阶段模型的故事生成方法则能在连贯性、一致性和逻辑性方面全面超越仅在数据集上训练的单个BART模型。

6 样例研究

图3展示了故事生成方法的两个样例的完整生成过程，包括：从给定故事开头，得到可能的事件；通过事件构建问题；根据问题，模型生成多个回答，并选择困惑度最小的那个作为下文句子。

6.1 验证生成流程

从图3(a)可以看出，在故事生成的第一轮，对于输入*Bart got a skateboard for Christmas. Bart tried to ride the skateboard.*，常识推理模型能够捕捉关键字，生成相关的事件*to play skateboard*。并且通过提示问题模板得到的问题*What does Bart do to need to play skateboard?*，也是对故事发展的一个很好的推测。有了优质的问题，预训练问答模型便能得到合乎逻辑的、保持一致性与连贯性的下文句子*He fell down and broke his skateboard.*。而随着生成故事的进行，这种逻辑性、一致性与连贯性并没有丢失。这说明我们的方法能够按照设计的思路进行故事生成。



(a) 样例1　　(b) 样例2

Figure 3: 故事生成样例

6.2 错误分析

但是，从图3(b)中，我们也能看到不符合逻辑的故事。如输入：*Roy's parents were adamant about him not smoking. Roy did smoke, but tried to hide it from his family.* 产生的下一句为：*He told his mom about it and she was supportive.* 故事开头已经说明Roy的家庭不允许Roy抽烟，且Roy也想方设法隐瞒，但第三句却变成了Roy告诉了他的妈妈（他抽烟的事实），结果他的妈妈还非常支持他，与前文逻辑不符合。导致这一原因是在生成第三句时，Para-COMET模型给出了事件推理*to smoke*，并且由该事件得到的问题*What does Roy do to need to smoke?*，其所生成的答案最终被选用。这也表明本文提出的故事生成方法主要受限于使用的预训练模型的性能。

7 总结与展望

本文针对故事生成任务需要大量数据的问题，提出了基于提示学习和预训练模型的故事生成方法，使用外部常识推理知识，充分发挥提示模板在少样本以及零样本场景下的优势。该方法将故事生成分为三个阶段：输入故事的开头，常识推理模型Para-COMET生成多个可能的事件，这些事件有六种类型，且是对故事发展方向的合乎社会逻辑的推测；根据类型，获取对应各个事件的人物角色，并将事件与角色填入问题模板中，构建总结上文并引导模型生成下文的问题；问答模型ELI5QA和BART产生对应的答案，并选择困惑度最小的作为故事下文。重复上述过程，最终生成完整的故事。实验表明，在提示学习与多个预训练模型的帮助下，无论是零样本场景还是充足训练数据场景，本文提出的故事生成方法都能在各项指标的得分上对基线模型保持优势。消融实验也突出了优质提示模板的重要性。

在未来，我们的工作主要可以分为三个方向：

- 选择更合适、性能更好的预训练模型，并设计与之匹配的提示问题模板，来提升生成故事的质量；
- 在选择答案作为下文句子时，尝试融合一些预训练模型，来避免选中逻辑上有问题、但困惑度得分低的句子，并且考虑在生成的各个阶段结束时，检查局部生成内容的正确性；
- 将提示学习的思想迁移到其他文本生成任务，如文本摘要，对话系统等。

8 致谢

我们感谢匿名审稿人，他们的建议有助于完善这项工作。本研究得到国家自然科学基金(No.62106105)、南京航空航天大学科研启动基金(No.YQR21022)和南京航空航天大学高性能计算平台的支持。

参考文献

Amal Alabdulkarim, Siyan Li, and Xiangyu Peng. 2021. Automatic story generation: challenges and attempts. *arXiv preprint arXiv:2102.12634*.

Prithviraj Ammanabrolu, Ethan Tien, Wesley Cheung, Zhaochen Luo, William Ma, Lara J Martin, and Mark O Riedl. 2020. Story realization: Expanding plot events into sentences. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7375–7382.

Prithviraj Ammanabrolu, Wesley Cheung, William Broniec, and Mark O Riedl. 2021. Automated storytelling via causal, commonsense plot ordering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5859–5867.

Rebeca Amaya Ansag and Avelino J Gonzalez. 2021. State-of-the-art in automated story generation systems research. *Journal of Experimental & Theoretical Artificial Intelligence*, pages 1–55.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. *arXiv preprint arXiv:1906.05317*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Louis Castricato, Spencer Frazier, Jonathan Balloch, Nitya Tarakad, and Mark Riedl. 2021. Automated story generation as question-answering. *arXiv preprint arXiv:2112.03808*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*.

Angela Fan, Yacine Jernite, Ethan Perez, David Grangier, Jason Weston, and Michael Auli. 2019a. Eli5: Long form question answering. *arXiv preprint arXiv:1907.09190*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2019b. Strategies for structuring story generation. *arXiv preprint arXiv:1902.01109*.

Saadia Gabriel, Chandra Bhagavatula, Vered Shwartz, Ronan Le Bras, Maxwell Forbes, and Yejin Choi. 2021. Paragraph-level commonsense transformers with recurrent memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12857–12865.

Jian Guan, Fei Huang, Zhihao Zhao, Xiaoyan Zhu, and Minlie Huang. 2020. A knowledge-enhanced pretraining model for commonsense story generation. *Transactions of the Association for Computational Linguistics*, 8:93–108.

Jian Guan, Xiaoxi Mao, Changjie Fan, Zitao Liu, Wenbiao Ding, and Minlie Huang. 2021. Long text generation by modeling sentence-level and discourse-level coherence. *arXiv preprint arXiv:2105.08963*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

Li Lin, Yixin Cao, Lifu Huang, Shuang Li, Xuming Hu, Lijie Wen, and Jianmin Wang. 2022. Inferring commonsense explanations as prompts for future event generation. *arXiv preprint arXiv:2201.07099*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Danyang Liu, Juntao Li, Meng-Hsuan Yu, Ziming Huang, Gongshen Liu, Dongyan Zhao, and Rui Yan. 2020. A character-centric neural model for automated story generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 1725–1732.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Pradyumna Tambwekar, Murtaza Dhuliawala, Lara J Martin, Animesh Mehta, Brent Harrison, and Mark O Riedl. 2018. Controllable neural story plot generation via reinforcement learning. *arXiv preprint arXiv:1809.10736*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7378–7385.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized bert pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227.