Few-Shot NLG with Pre-Trained Language Model

会议：ACL2020

代码：<https://github.com/czyssrs/Few-Shot-NLG>

**概要**

基于神经的端到端的方法从结构化数据或知识生成自然语言（NLG）是一种数据饥渴的方法，这使得它们在有限的数据下难以应用于现实世界。本文提出了小样本自然语言生成的新任务。基于人们倾向于总结表格数据的动机，我们提出了一种简单而有效的方法，并表明它不仅表现出强大的性能，而且还提供了跨域的良好泛化。模型体系结构的设计基于两个方面：从输入数据中选择内容和从先验知识中获取连贯句子的语言建模。通过200个跨多个领域的训练实例，我们证明了我们的方法获得了非常合理的性能，并且平均比最强的基线提高了8.0个BLEU点。

**1引言**

基于结构化数据或知识的自然语言生成（Natural language generation，NLG）（Gatt and Krahmer，2018）是各种自然语言生成应用的重要研究课题。一些例子包括面向任务的对话、问答（He et al.，2017；Ghazvininejad et al.，2018；Su et al.，2016；Saha et al.，2018；Yin et al.，2016）和跨学科应用，如医学（Hasan and Farri，2019；Cawsey et al.，1997）和医疗保健（Hasan and Farri，2019；DiMarco et al.，2007）。

自动NLG系统在现实生活中有很大的应用潜力。最近，基于深度神经网络的NLG系统已经开发出来，如E2E挑战（Novikova等人，2017）、WEATHERGOV（Liang等人，2009）中所见，以及更复杂的系统，如WIKIBIO（Liu等人，2018）和ROTOWIRE（Wiseman等人，2017）。与传统的填槽流水线方法相比，这种基于神经网络的系统大大减少了特征工程的工作量，提高了文本的多样性和流畅性。

尽管他们在E2E challenge（Novikova et al.，2017）和WIKIBIO（Lebret et al.，2016）等基准上取得了良好的表现，但他们的表现取决于大的训练数据集，例如WIKIBIO的500k表文本训练对（Lebret et al.，2016）在单个域中。这种数据饥渴的特性使得基于神经的NLG系统很难在实际应用中被广泛采用，因为它们有大量的人工数据管理开销。这就引出了一个有趣的研究问题：

*1、我们是否可以显著减少人工注释的工作量，以使用神经NLG模型实现合理的性能？*

*2、我们能否充分利用生成性预训练作为先验知识，从结构化数据中生成文本？*

基于此，我们提出了few-shot自然语言生成的新任务：只要给定少量的标记实例（如50-200个训练实例），系统就需要产生令人满意的文本输出（如BLEU >20）。据我们所知，NLG社区的这一问题仍然没有得到充分的探讨。在此，我们提出一个简单但非常有效的方法，可以推广到不同的领域。

一般来说，要描述表中信息，我们需要两种技巧来组成连贯而可信的句子。一个技巧是从表格中选择和复制事实内容——这可以通过阅读一些表格快速学习。另一个是组成语法正确的句子，把这些事实结合起来-这不需要限制于任何领域。

人们可以想到一种潜在的“转换”，它帮助我们在这两种技能之间进行转换，从而产生事实上正确和连贯的句子。为此，我们使用预先训练的语言模型（Chelba等人，2013年；Radford等人，2019年）作为固有的语言技能，它提供了关于如何构成流畅连贯句子的强大先验知识。只需几个训练实例，就可以成功地学习从表中切换和选择/复制的能力，从而使神经网络学习模型从数据密集型训练中解放出来。以前基于大量训练数据的最佳性能方法，例如（Liu等人，2018），它不应用这种切换机制但训练强的领域特定语言模型，在很少样本设置下性能非常差。

由于我们是在一个高度数据限制的少样本体制下操作，我们努力简化模型架构。这种简单性还意味着现实世界应用程序具有更好的通用性和再现性。我们从Wikipedia抓取多域table-text数据作为我们的训练/测试实例。通过200个训练实例，我们的方法可以达到非常合理的效果。

简而言之，我们的贡献总结如下：

我们提出了few-shot-NLG的新的研究问题，这对实际应用有很大的潜力。

为了研究针对我们提出的问题的不同算法，我们创建了一个table-to­-text数据集。

我们提出的算法可以利用外部资源作为先验知识，显著减少人工标注的工作量，在不同领域的基线性能平均提高8.0bleu以上。

**2相关工作**

**2.1结构化数据的NLG**

基于结构化数据/知识的自然语言生成（NLG）是许多NLP应用的核心目标。早期的传统NLG系统遵循流水线模式，明确地将生成分为内容选择、宏观/微观规划和表面实现（Reiter和Dale，1997）。这种流水线模式主要依赖于模板和手工设计的特性。已经提出了许多解决单个模块的工作，例如（Liang等人，2009；Walker等人，2001；Lu等人，2009）。后来的工作（Konstas和Lapata，2012、2013）研究了统一框架中的建模上下文选择和表面实现。

最近，随着深度神经网络的成功，数据驱动的、基于神经的方法被使用，包括端到端的方法，它们联合建模上下文选择和表面实现（Liu et al.，2018；Wiseman et al.，2018；Puduppully et al.，2018）。此类数据驱动方法在E2E challenge（Novikova et al.，2017）、WebNLG challenge（Gardent et al.，2017）和WIKIBIO（Lebret et al.，2016）等多个基准上取得了良好的性能。然而，他们依赖大量的训练数据。ElSahar等人。（2018）提出从知识图生成问题的零样本学习，但他们的工作适用于基于已见知识及其文本上下文的未见知识库类型的转移学习设置，这仍然需要大量的领域内训练数据集。这不同于我们的少数射击学习设置。Ma等人。（2019）提出低资源表到文本生成需要1000个配对示例和大规模目标侧示例。相比之下，在我们的设置中，只需要几十到几百个成对的训练示例，而不需要任何目标示例。

这对于现实世界的用例尤其重要，在这些用例中，很难获得如此大的目标端gold引用。因此，我们的任务更具挑战性，更接近现实世界。

**2.2大型预训模型**

对于各种NLP任务，当前许多性能最佳的方法都采用预训练和监督微调相结合的方法，使用特定于任务的数据。不同级别的预训练包括单词嵌入（Mikolov等人，2013年；Pennington等人，2014年；Peters等人，2018年）、句子嵌入（Le和Mikolov，2014年；Kiros等人，2015年）以及最近的基于语言建模的预训练，如BERT（Devlin等人，2018年）和GPT-2（Radford等人，2019年）。该模型在大规模开放域语料库上进行预训练，在提高性能的同时，为下行任务提供丰富的先验知识。本文采用预先训练语言模型的思想，赋予领域内NLG模型语言建模的能力，这是从小样本训练实例中学不到的。

**3方法**

**3.1问题表述**

我们获得了半结构化数据：属性-值对表。和Vi都可以是字符串/数字、短语或句子。每个值都表示为一个单词序列。对于每个单词，我们都有其对应的属性名Ri和单词在值序列中的位置信息。目标是基于半结构化数据生成自然语言描述，只需提供少量训练实例。

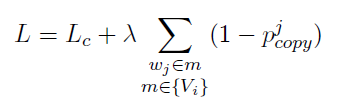
**3.2具有交换策略的基本框架**

我们从（Liu等人，2018）中提出的场选通双注意模型开始，该模型在WIKIBIO数据集上实现了最先进的性能（BLEU）。他们的方法使用一个具有双重注意权重的LSTM解码器。我们首先应用一种交换策略，将框架分离为表内容选择/复制和基于语言模型的生成。受到指针生成器的启发（见等人,2017）,在每个时间步，我们都维护一个软交换，以选择从softmax生成词汇表还是从带有注意权重作为概率分布的输入表值中复制，



其中，，是编码器隐藏状态，是分别在时间步骤t的解码器输入、状态和注意权重，。和b是可训练的参数。

指针生成器学习如何在复制和基于大量训练数据的生成之间进行交替，并显示了从输入中复制词汇表外单词的优势。在我们的任务中，训练数据非常有限，而且许多表值都不是OOV。我们需要明确地“教导”模型在哪里复制，在哪里生成。因此，为了提供模型对开关行为的精确指导，我们将目标文本与输入表值进行匹配，以获得复制位置。在这些位置，我们通过附加的损失项使复制概率最大化。我们的损失函数：



其中是模型输出和目标文本之间的原始损失。是位置j处的目标标记，是第3.1节中定义的输入表值列表，m表示匹配短语。是本次复制loss参数的超参数权重。我们还将解码器输入与其匹配的属性名和输入表中的位置信息连接为来计算。

**3.3 Pre-Trained LM as Generator**

我们使用预先训练的语言模型作为生成器，充当“先天语言技能”。由于训练实例的词汇量有限，我们在对预训练语言模型的其他参数进行微调的同时，对预训练的单词嵌入保持固定，这样就可以在训练过程中未见标记的情况下进行泛化。

图1显示了我们的模型架构。我们使用在（Radford等人，2019）中提出的预先训练语言模型GPT-2，它是一个12层的transformer。transformer的最终隐藏状态用于计算注意权重和复制转换pcopy，我们首先将嵌入的attribute-value列表作为生成的上下文。在这种结构中，生成器根据预先训练的参数进行微调，而编码器和注意部分则从头学习，两边的初始几何结构不同。因此，我们需要对复制损失pcopy应用更大的权重，以给模型一个更强的信号来“教导”它从输入表中复制事实。

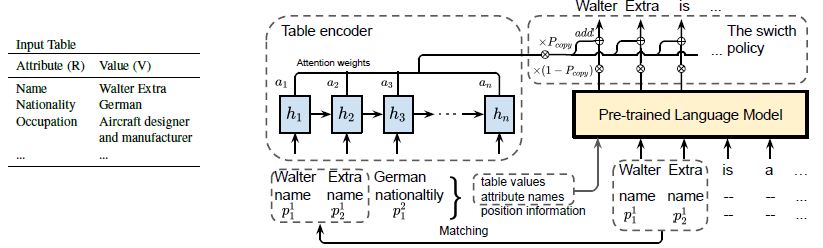


图1：我们方法的概述：在带有交换策略的基本框架下，预先训练的语言模型充当生成器。我们遵循与（Liu等人，2018）中相同的编码器。该架构在实现和参数空间方面都很简单，需要从头学习，考虑到小样本学习设置，参数空间不应该很大。

**4实验**

**4.1数据集和实验装置**

最初的WIKIBIO数据集（Lebret等人，2016年）包含70万篇知名人类的英文维基百科文章，其中Wiki信息框用作输入结构化数据，文章的第一句话用作目标文本。为了展示通用性，我们通过爬行维基百科页面从两个新领域收集数据集：书籍和歌曲。经过过滤和清理，我们最终得到了图书领域23651个实例和歌曲领域39450个实例。与原始WIKIBIO数据集的人类域一起，对于所有三个域，我们通过将训练数据集大小更改为50、100、200和500来进行实验。其余数据用于验证（1000）和测试。copy loss term的权重设为0.7。其他参数设置见附录A。为了解决小样本训练的词汇限制，对于所有模型，我们采用字节对编码（BPE）（Sennrich et al.，2016）和子单词词汇（Radford et al.，2019）。

我们将所提出的方法与第3节中研究的其他方法进行比较，这些方法作为基线-原始：原始模型in (Liu et al., 2018); Base：使用相同的体系结构，但在训练期间应用预先训练好的单词嵌入并修复；Base+switch：添加切换策略；Base+switch+LM scratch：使体系结构与我们的方法相同，只是从头训练模型而不是对生成器使用预先训练好的权重。模板：基于模板的非神经方法，为每个领域手工制作。

**4.2结果与分析**

继之前的工作（Liu等人，2018年）之后，我们首先使用BLEU4进行自动评估，如表1所示。ROUGE-4（F-measure）结果与BLEU-4结果的趋势相同，我们在附录B中给出了这一结果。

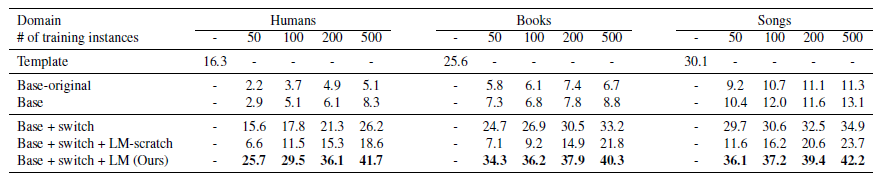


表1:BLEU-4在三个领域的结果。Base original:中的原始方法（Liu等人，2018）；Base:应用预先训练的字嵌入；Base+switch:添加开关策略；Base+switch+LM scratch:使架构与我们的方法相同，但从头开始训练模型，而没有为生成器预先训练权重。模板：手工创建的模板

如我们所见，原始模型Baseoriginal（Liu等人，2018）在WIKIBIO全套中获得了最新的结果，但在小样本设置下表现非常差。它在词汇表上从softmax生成所有标记，这会导致严重的训练数据过度拟合，并且结果远远落后于基于模板的基线。在交换策略中，Base+switch首先带来平均10.0个BLEU点以上的改进。这表明，通过少量的训练实例，内容选择能力更容易学习。然而，它形成的句子非常有限，不流利。随着预训练语言模型的增加，我们的模型库+switch+LM使平均BLEU点提高了8.0bleu点以上。我们使用表2中的200个训练实例提供这些方法的样本输出。

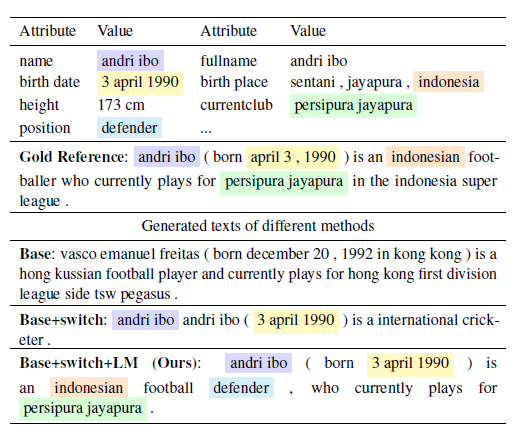


表2：一个示例输入表，使用200个训练实例从Humans域的测试集生成摘要

表3显示了第3.2节中介绍的copy switch loss pcopy的影响，使模型具有更强的信号，可以从输入表学习复制。

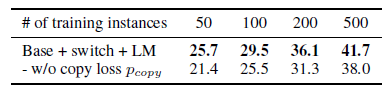


表3：消融研究：用BLEU-4测量复制丢失项对人类领域的影响。损失期平均提高了4.0个BLEU点。

Ma等人。（2019）提出了支点模型，用于1000个成对实例和大规模目标侧实例的低资源NLG。我们用表4中透视模型的方法比较。请注意，在这里，我们将对其工作中使用的原始WikiBio数据集上的模型进行培训和评估，以便为其设置保持目标端示例的大小。



表4：与Pivot模型的比较（Ma等人，2019年）。与使用附加的大规模目标侧示例的方法相比，我们的方法不需要附加的目标侧数据，同时获得更好的性能。

**Human Evaluation**

我们还使用Amazon Mechanical Turk进行了基于两个方面的人类评估研究：事实正确性和语言自然性。我们评估了500个样品。每个评估单元分配给3名工人，以消除人为差异。第一项研究试图评估生成的文本如何正确地在表中传递信息，方法是计算表支持的文本中的事实数，并与表相矛盾或从表中丢失。表5的第2列和第3列显示了与最强基线和黄金参考值相比，我们方法的支持和矛盾事实的平均数。

第二项研究评估生成的文本在语法上是否正确和流畅，而不考虑事实的正确性。我们对所有方法进行成对比较，并计算出每种方法的平均选择次数要优于另一种方法，如表5第4栏所示。我们的方法比最强的基线有显著的改善（在所有测量中，Tukey的HSD测试中p<0:01）。复制损失条款进一步缓解了产生不正确事实的情况。我们的方法在不丢失文本的情况下，语言自然度的结果稍好一些，因为这种评价没有考虑事实的正确性，因此生成的错误事实较多的文本仍然可以得到高分。有关我们的评估程序的更多详细信息，请参见附录C。

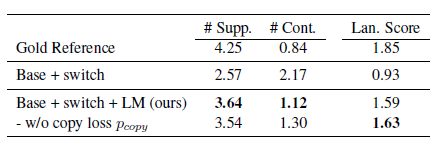


表5：人类评价结果：支持事实的平均数（第2栏，越大越好）、矛盾事实（第3栏，越小越好）、语言自然度得分（第4栏，越大越好）。

**5.结论**

本文提出了小样本自然语言生成的新研究课题。我们的方法简单，易于实现，同时在不同的领域获得强大的性能。我们获得语言建模先验的基本思想可以潜在地扩展到更广泛的生成任务范围，基于各种输入结构化数据，如知识图、SQL查询等。对于许多实际应用来说，为这些任务进行人工数据管理的工作是非常有潜力和重要的。

**附录A.实施细节**

我们使用Adam优化器（Kingma和Ba，2015），学习率设置为0.0003。最小批量设置为40，副本丢失项的权重设置为0.7。位置嵌入的维度设置为5。对于包含多个单词的属性名，我们将它们的单词嵌入平均为属性名嵌入。请参阅我们发布的代码和数据https://github.com/czyssrs/Few-Shot-NLG了解更多细节。

**附录B.** **ROUGE-4结果**

继之前的工作（Liu等人，2018年）之后，我们使用BLEU-4和ROUGE-4（F-measure）3进行自动评估。表6、7和8分别显示了人类、书籍和歌曲三个领域的ROUGE-4结果。

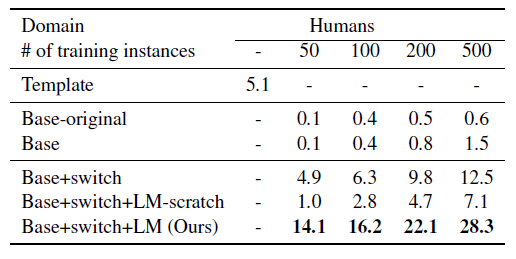


表6:ROUGE-4人类领域的结果

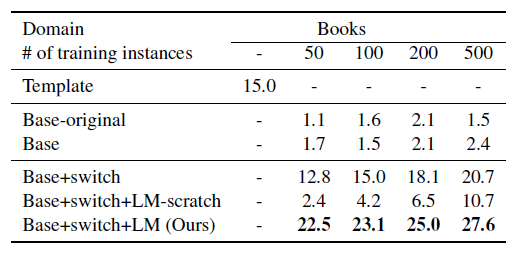


表7:Books域的ROUGE-4结果

**附录C.人工评估细节**

基于事实正确性和语言自然性两个方面，我们使用Amazon Mechanical Turk进行了人类评价研究。在这两项研究中，我们评估了200个人类领域训练实例的结果。我们从测试集中随机抽取500个实例，以及用不同方法生成的文本。每个评估单元分配给3名工人，以消除人为差异。

第一项研究试图评估生成的文本在表中正确传达信息的能力。每个工人都有输入表和生成的文本，并要求计算生成的文本中有多少事实得到了表的支持，以及有多少事实与表相矛盾或缺失，如中所述（Wiseman等人，2017年）。我们计算每种方法生成的文本的支持和反驳事实的平均数。

第二项研究旨在评估生成的文本在语法上是否正确，在语言上是否流利，而不管事实上是否正确。

每个工作人员都有一对由同一个输入表通过两种不同的方法生成的文本，然后要求他们只根据语言的自然程度选择一个更好的文本，如果两个文本质量相同，则要求他们“绑定”。输入表不会显示给工人。每次生成的文本被选为更好的文本时，我们都会分配1.0的分数。如果两个文本是并列的，我们为每个文本分配0.5。然后，我们计算每种方法生成的文本的平均得分，表明其在与所有其他方法的成对比较中的优越性。

分别对三项指标进行显著性检验：第一项研究的支持事实数和反驳事实数；第二项研究的指定分数。我们使用Tukey-HSD事后方差分析，其中工人的反应为因变量，方法和工人id为自变量。

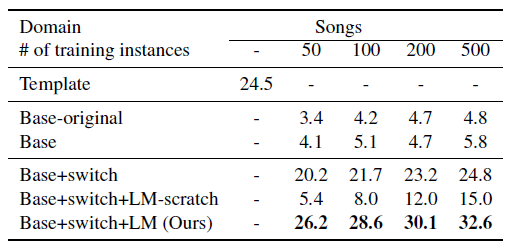


表8:Songs域的ROUGE-4结果