**Learning to Customize Model Structures for Few-shot Dialogue Generation Tasks**

会议：ACL2020

代码：https://github.com/zequnl/CMAML

**Abstract**

用最少的语料训练生成模型是构建开放域对话系统的关键挑战之一。现有的方法倾向于使用元学习框架，它预先训练所有非目标任务的参数，然后对目标任务进行微调。然而，微调将任务与参数视角区分开来，却忽略了模型结构视角，从而导致不同任务的相似对话模型。在本文中，我们提出了一个算法，可以为每一个任务定制一个独特的对话模型。在我们的方法中，每个对话模型由一个共享模块、一个选通模块和一个私有模块组成。前两个模块在所有任务之间共享，而第三个模块将区分为不同的网络结构，以更好地捕获相应任务的特征。对两个数据集的大量实验表明，该方法在任务一致性、响应质量和多样性方面都优于所有基线。

**1 Introduction**

生成性对话模型通常需要大量的对话来进行训练，而构建能够适应数据有限的新领域或新任务的模型是一项挑战。随着大规模预训练的最新进展【Peters等人，2018年；Howard and Ruder，2018年；Radford等人，2018年；Devlin等人，2018年】，我们可以先在非目标领域的大规模对话上预训练生成模型，然后在特定任务的数据语料库上进行微调【Wang等人，2019a；Alt等人，2019a；Klein，2019年】。虽然预先培训是有益的，但此类模型仍然需要足够的特定于任务的数据来进行微调。当很少的例子时，他们不能达到令人满意的性能

不幸的是，在许多对话生成场景中经常出现这种情况。例如，在个性化对话生成中，我们需要通过用户的几段对话快速适应用户角色的反应风格【Madoto等人，2019；Zhang等人，2018】；在情感对话生成中，我们需要使用很少包含此表情符号的话语生成迎合新表情符号的反应【Zhou等人。，18、 周和王，2018年）。因此，这就是我们论文的重点-few-shot对话生成，即训练一个生成模型，该模型可以推广到其对话的有k-shot的一个新任务（领域）。

有人提出了一些将few-shot对话生成视为元学习问题的研究（Madotto等人，2019；Qian和Yu，2019；Mi等人，2019）。它们都依赖于流行的模型不可知元学习（MAML）方法【Finn等人，2017年】。以构建个性化对话模型为例，以往的研究将不同角色的学习对话视为不同的任务【Madotto等人，2019年；Qian和Yu，2019年】。当应用于新任务时，他们使用MAML通过最大化损失函数的灵敏度来寻找模型参数的初始化。对于一个目标任务，它的对话模型是通过对MAML中的初始参数进行微调，并结合其特定任务的训练样本得到的。

尽管在few-shot对话中取得了明显的成功，MAML仍然有局限性[Zintgraf等人，2019]。生成对话模型的目标是构建一个将用户查询映射到其响应的函数，其中该函数由模型结构和参数共同决定[Brock等人，2018]。通过对固定模型结构的微调，MAML只从参数优化的角度搜索最优参数设置，而忽略了从结构优化的角度搜索最优网络结构。语言数据本质上是离散的，对话模型比图像相关模型更不容易受到输入变化的影响[Niu和Bansal，2018年]，这意味着从几句话计算出来的梯度可能不足以将输出单词从一个改变到另一个。因此，有必要开发一种有效的方法来调整MAML，以适应对话生成任务中的大模型多样性。

本文提出了一种定制模型不可知元学习算法（CMAML），该算法能够在MAML框架下从参数和模型结构两个角度定制对话模型。每个任务的对话模型由三部分组成：一个共享模块学习任务之间的通用语言生成能力和共同特征；一个私有模块学习任务的独特特征；一个从共享模块和私有模块中吸收信息并生成最终输出的门。共享模块和gating模块的网络结构和参数在所有任务之间共享，而私有模块从同一个网络开始，但区分为不同的结构，以捕获特定于任务的特征。

总之，我们的贡献如下：

提出了一种CMAML算法，该算法可以在few-shot设置下，针对不同任务定制不同网络结构的对话模型。该算法具有通用性和良好的统一性，能够适应few-shot生成场景。

我们提出了一种能够调整网络结构以更好地拟合训练数据的剪枝算法。我们使用这个策略为不同的任务定制独特的对话模型。

研究了元学习方法的两个关键影响因素，即训练数据量和任务相似度。然后我们描述了元学习优于其他微调方法的情况。

**2 Related Work**

**Few-shot Dialogue Generation.** 在过去几年中，人们越来越关注在few-shot设置中构建对话模型，例如能够快速适应每个用户的个人资料或知识背景的个性化聊天机器人（Zhang等人，2018年；Madotto等人，2019年），或以特定情感回应的聊天机器人（Zhou等人，18年；Zhou和Wang，2018年）。早期的解决方案是使用显式[田等人，2017；张等人，2018；周等人，18]或隐式[李等人，2016b；周和王，2018；周等人，18]任务描述，然后将此信息引入生成模型。但是，这些方法需要手动创建任务描述，这在许多实际情况下是不可用的。

建立few-shot对话模型的另一个有希望的解决方案是元学习方法，特别是MAML【Finn等人，2017年】。Madotto等人。（2019）提出将每个用户的对话语料库作为学习任务，通过对任务特定数据的初始化参数进行微调，赋予个性化对话模型。Qian and Yu（2019）和Mi等人。（2019）将多领域任务导向对话生成中各个领域的学习视为一项任务，并以类似的方式应用MAML。由于MAML的模型不可知特性，所有这些方法都不改变原始的MAML，而是直接将其应用到场景中。因此，任务区分总是依赖于微调，微调只在参数级别搜索每个任务的最佳模型，而不是模型结构级别。

Meta-learning.

元学习由于其对新任务的快速适应能力，近年来在许多NLP问题上取得了很好的效果，使用的训练数据很少[Yu等人，2019；Wang等人，2019b；Obamuyide和Vlachos，2019b；Alt等人，2019b]。一般来说，元学习方法分为三类：基于度量的方法【Vinyals等人，2016年；Snell等人，2017年；Sung等人，2018年；Ye和Ling，2019年】，它们将样本与学习距离度量一起编码到嵌入空间，然后应用匹配算法、基于模型的方法【Santoro等人，2016年；Obamuyide和Vlachos，2019a]，这取决于模型结构设计，例如外部存储器存储，以促进学习过程，以及基于优化的方法[Finn等人，2017；Andrychowicz等人，2016；Huang等人。，2018年]它学习了良好的网络初始化，从中微调可以收敛到新任务的最佳点（仅举几个例子）。

前两类方法被提出用于分类，而第三类方法是模型不可知的。因此，将最流行的基于优化的MAML方法应用于对话生成任务是直观的。

然而，一些研究人员发现，原始的MAML在图像或文本分类场景中对特定任务特征建模的能力有限[Jiang等人，2018；Sun等人。，

Jiang等人。（2018）在卷积层上建立注意层，其中卷积层用于一般特征，注意层用于特定任务特征。Sun等人。（2019）建议学习通用共享前馈层上的特定任务转移和缩放操作。然而，这两种方法中涉及的操作，如移位和缩放，都是为前馈网络设计的，不能应用于通常依赖Seq2seq[Sutskever等人，2014]模型和递归GRU[Cho等人，2014]或LSTM[Hochreiter和Schmidhuber，1997]细胞的生成模型。本文提出了一种新的基于MAML的元学习算法，可以增强生成模型的任务特性。

**3 Dialogue Model**

在这一部分中，我们首先描述了所提出的对话模型的网络结构，然后简要介绍了它的预训练。

**3.1 Model Architecture**

我们的目标是为不同的生成任务在少数镜头设置对话模型。现在，我们首先描述在我们的训练算法中使用的每个任务的对话模型。它包括三个网络模块，如图1所示：

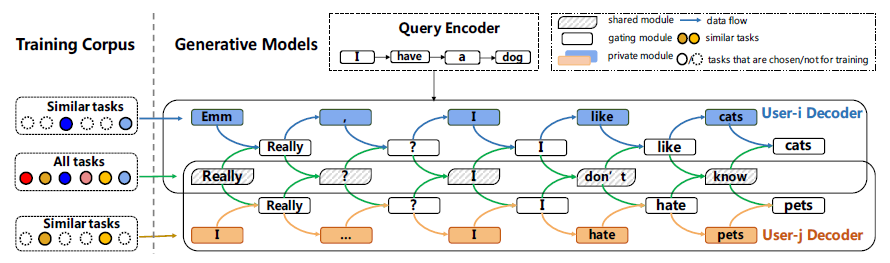
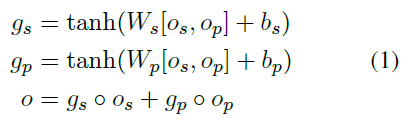


图1：提出的CMAML算法在个性化对话系统中的应用。每个定制的对话模型Seq2SPG由一个共享、一个私有和一个选通模块组成。共享模块和选通模块在用户之间是相同的，并接受了所有任务的培训。私有模块对于每个用户来说都是唯一的，用来描述这个用户的角色，并且在相应的和类似的任务上进行训练。彩色线表示数据流的方向。

**Shared Module.** 它获得生成句子的基本能力，因此它的参数在所有任务中共享。我们采用一种流行的Seq2seq对话模型【Bahdanau等人，2014年】。在每个解码步骤t，我们将单词和最后的隐藏状态馈送到解码单元，并获得词汇表上的输出分布。

**Private Module.** 它旨在为每个任务的独特特性建模。为了实现这一目标，我们在解码器中设计了一种多层感知（MLP）。每一个任务都有其独特的MLP网络，它从相同的初始化开始，在训练过程中演化为不同的结构。在每个解码步骤t中，MLP以单词xt和步骤t-1处共享模块的输出作为输入，然后输出词汇表上的分布。在我们的实验中，我们还探索了私有模块的不同输入。

Gating Module. 我们使用一个门来融合来自共享和私有模块的信息：



其中Ws、Wp、bs、bp是参数，是元素级积，o是单词分布。

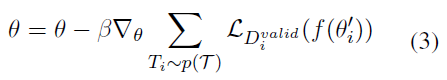
**3.2 Training Overview**

其余部分，表示任务分布，T i表示第i个待训练任务，和表示任务Ti的训练和验证语料库，表示对话模型的所有训练参数，分别包括共享/私有/选通模块中的参数。我们考虑一个由带参数的参数化函数f表示的模型。所有任务的模型训练包括两个步骤：预训练和定制模型训练。

在预训练中，CMAML使用香草MAML获得一个预训练的对话模型作为所有任务的初始模型。在MAML开始时，被随机初始化。然后，两个主要程序执行迭代：元训练和元测试。在元训练中，MAML首先抽取一组任务。然后，对于每一个任务i，MAML调整以获得任务特定数据的，也就是说，



在元测试中，MAML使用测试任务以获得损失，然后更新



这里，和是超参数。

在标准的MAML中，每个任务通过微调预先训练的参数来获得其参数。然而，回想一下，微调无法从网络结构的角度搜索最佳模型。此外，生成模型不易受输入变化的影响，因此一些话语可能不足以适应不同任务的多样性。为了解决这些问题，我们没有对每个任务执行直接的微调，而是设计了第二个训练步骤-定制模型训练，其中预训练的私有模块可以演变成不同的结构，以捕获每个任务的特征并鼓励模型的多样性。

**4 Customized Model Training**

在从MAML获得预训练模型后，我们采用定制模式训练，更新步骤如下：

专用网络修剪。此步骤仅适用于私有模块，即区分每个任务的MLP结构。每个任务都有一个不同的MLP结构，通过保留自己的活动MLP参数子集来刻画任务的唯一性。

联合元学习。在这一步中，我们再次使用MAML重新训练每个任务的所有三个模块的参数，但是每个私有模块现在都有其剪枝的MLP结构。同时，对具有相似剪枝MLP结构的相似任务进行联合训练，以丰富训练数据。

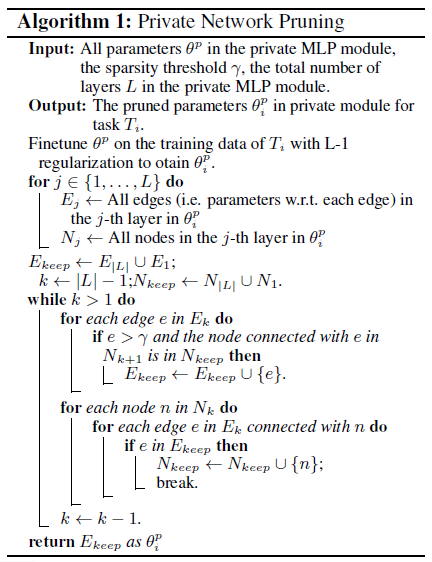
接下来，我们将分别描述这两个步骤以及整个对话模型的梯度更新。

**4.1 Private Network Pruning**

经过预训练后，不同任务的对话模型保持相同的参数，包括共享/私有/选通模块中的。在这一步中，带有参数的私有模块将演化为具有参数的不同结构，以捕获任务的独特特性。

首先，利用MAML初始化后的训练数据对每个任务的整个对话模型进行微调，并对私有模块的参数进行L-1正则化。这里L-1正则化的目的是使参数稀疏，使得只有有利于生成任务特定语句的参数才是活动的。

其次，我们应用自上而下的策略来修剪每个任务的私有MLP。这等于在MLP中的完全连接层中选择边。我们不修剪与MLP的输入和输出连接的层。对于其他层，我们首先从最接近输出的层开始修剪。对于第l层，我们认为上面的层（>l）更接近输出，下面的层（<l）更接近输入。当我们处理第l层时，它的上层应该已经修剪过了。我们只保留当前处理层的边，其权重超过某个阈值。如果l层中连接到某个节点的所有边都被修剪，则连接到l-1层中该节点的所有边也将被修剪。这样，私有模块中的参数分化为参数，其中每个是的子集。算法1说明了上述修剪算法。

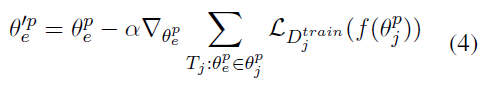


**4.2 Joint Meta-learning**

到目前为止，每个任务在其私有模块中都有一个独特的网络结构。现在我们共同训练所有任务的整个对话模式。

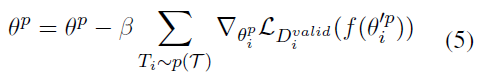
我们再次从预先训练的MAML初始化开始。对于共享模块和选通模块，所有任务共享相同的参数，并使用所有培训数据对它们进行培训。私有模块是用来捕获每个任务的唯一性的，它应该被训练成特定于任务的数据。但是，由于在few-shot设置下，我们没有足够的训练数据来完成每个任务，因此私有模块可能无法得到很好的训练。幸运的是，所有私有模块都是从相同的MLP结构演变而来的，类似的任务自然共享重叠的网络结构，即剪枝后的剩余边是重叠的。这激励我们通过所有任务的训练样本来训练私有MLP中的每一条边，在这些训练样本中，这个边没有被修剪。

具体地说，我们以这种方式培训私有MLP：对于MLP中的每个边e，如果它在多个任务中处于活动状态，则其相应的参数将根据所有任务j的数据进行更新，其中边处于活动状态，即



其中每个只包含i-th任务中所有激活边的。

在元测试过程中，损失由使用相应对话模型的任务累积，因此被更新为，

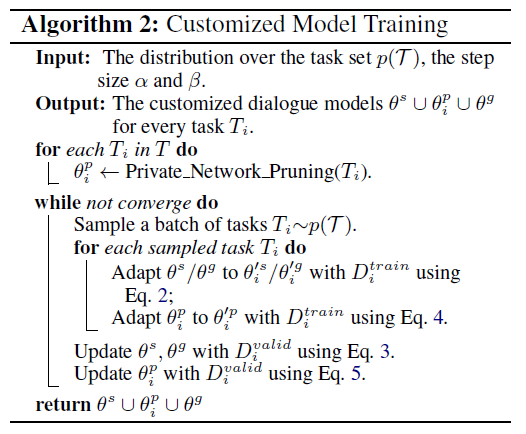


**4.3梯度更新**

在算法2的定制模型训练中，我们总结了我们提出的对话模型中三个模块的梯度更新。对于共享和选通模块，渐变的更新方式与MAML相同。私有模块的更新被上述在联合元学习中引入的等式4和等式5所取代。

在我们的模型中，用于计算梯度的损失函数是给定输入查询q，生成响应r的负对数似然，





**5 Experiments**

**5.1 Datasets**

我们在人物角色聊天[Madotto等人，2019]和MojiTalk[Zhou and Wang，2018]中进行了实验，这些实验在之前的工作中被视为很少的镜头对话生成任务[Zhang等人，2018；Madotto等人，2019；Zhou and Wang，2018；Zhou等人，18]。Persona chat有1137/99/100个用户进行培训/验证/评估，每个用户平均有121个话语。我们遵循前面的工作【Madotto等人，2019年】并连接所有上下文语句，包括查询作为输入序列。我们把为用户建立对话模型看作是在这个数据集上的一项任务。MojiTalk有50/6/8个emojis用于培训/验证/评估。每个训练/验证表情符号平均有1000个训练样本，每个评估表情符号平均有155个样本。我们把用指定的表情符号生成响应当作一项任务。在这两个数据集上，元训练和元测试的数据比率为10:1。

**5.2实施细则**

我们在Seq2seq模型的基础上实现了我们的共享模块，其中包含预先训练的手套嵌入[Pennington等人，2014]和LSTM单元，并使用4层MLP作为私有模块。字嵌入、隐藏状态和MLP输出的维度设置为300。在CMAML中，我们对模型进行了10个阶段的预训练，并对每个模型进行了5个步骤的重新训练，以修剪私有网络。再训练阶段L-1体重为0.001，阈值为0.05。我们遵循Madotto等人的其他超参数设置。[2019年]。

**5.3竞争方法**

仅限训练前：我们使用所有训练任务的数据预先训练一个统一的对话生成模型，然后直接在测试任务上进行测试。我们尝试了三种基本生成模型：Seq2seq[Bahdanau等人，2014]和Speaker模型[Li等人，2016b]以及第3.1节中提出的Seq2SPG。Speaker将任务（user/emoji）嵌入到LSTM单元中，测试任务的任务嵌入在此设置中是随机参数。

fine tune：我们对每个测试任务的预训练模型进行微调，表示为Seq2seq-F、Speaker-F和Seq2SPG-F。

MAML[Madotto等人，2019]：我们将MAML算法应用于基本模型Seq2seq和Seq2SPG，并将它们记为MAML-Seq2seq和MAML-Seq2SPG。MAML-Seq2SPG使用与所提出的CMAML相同的基本模型，但不使用剪枝算法，这有助于验证剪枝算法和联合元学习的有效性。注意，我们没有在扬声器模型上应用MAML，因为它与Seq2seq相比没有任何改进。

CMAML：我们尝试了我们提出的算法的两种变体。CMAML-Seq2SPG是我们的完整模型（在前面的章节中等同于CMAML），其中对话Seq2SPG是基本模型，修剪算法用于定制任务的唯一模型结构。CMAML-Seq2SP0G使用了另一个称为Seq2SP0G的基本模型，其中私有模块仅将共享模块的输出作为输入。在网络定制的私有模块中也采用了剪枝算法。

**5.4评价指标**

自动评估。我们从三个角度执行自动评估指标：

反应质量/多样性：我们使用BLEU【papinini等人，2002年】测量参考句和生成句之间的词重叠；PPL，生成句的负对数；Dist-1【Li等人，2016a；Song等人，2017年，2018年】评估反应多样性，它计算所有测试生成的响应中不同的1-gram的比率。

任务一致性：我们在人物角色聊天中使用C分数【Madotto等人，2019年】，使用预先训练的自然语言推理模型来测量与人物角色描述的反应一致性；在MojiTalk中使用E-acc【Zhou和Wang，2018年】，使用情绪分类器来预测反应与指定情绪之间的相关性。

模型差异：由于我们没有基本的真值模型，因此很难衡量定制的模型能力。因此，我们将成对任务的平均模型差定义为每种方法的差异得分，将方法微调前后的模型差定义为得分。Ti和Tj之间的模型差异是由其参数计数规范化的参数的欧氏距离：这里，包含此任务的所有模型参数，M是模型的总参数个数。一组能够捕捉到每个任务独特特征的模型应该是彼此不同的，并且具有更高的Diff分数，这表明较大的Diff分数是具有较强定制能力的充分条件。同样，在微调过程中，一个对任务特定适应变化较大的模型将获得更高的分数，这表明分数也是获得良好适应能力的充分条件。

**人的评价**。我们邀请了3名受过良好教育的毕业生为每种方法生成的100个回复添加注释。对于每个数据集，注释者被要求按照“质量”和“任务一致性”（即人物聊天中的个性一致性和MojiTalk中的表情一致性）对每个响应进行独立的三个等级：2（好）、1（好）和0（坏）。“质量”衡量回答的恰当性，我们引用2表示流畅、高度一致（查询和回答之间）和信息性，1表示语法错误少、适度一致和普遍回答，0表示不可理解或无关的主题。“任务一致性”衡量一个回复是否符合某个任务的特征，我们将2表示高度一致，1表示不冲突，0表示矛盾。注意，用户描述（Persona数据集）和带有特定emoji（Mojitalk数据集）的语句被提供为引用。志愿者，而不是作者，对洗牌样本进行双盲注释，以避免主观偏见。

**5.5总体性能**

质量/多样性。在Persona聊天数据集中，Pretrain Only方法提供了所有方法的边界。仅在Pretrain中，Seq2SPG在自动和人工测量方面都达到了最佳性能，表明模型的表现。在大多数情况下，微调方法比仅预应变方法要好。MAML方法在BLEU分数上的表现并不比Finetune方法好，但Dist-1分数相对较高。这表明MAML有助于提高响应多样性。通过改进剪枝算法，我们可以看到CMAML方法在质量和多样性度量方面都比所有竞争方法有了很大的改进。特别地，我们的全模型CMAML-Seq2SPG显示出明显的更好的性能，其原因可以归结为两个方面：第一，我们提出的Seq2SPG具有更好的模型结构，用于我们的任务；第二，修剪算法使得模型更容易产生用户一致的响应。

MojiTalk数据集中竞争方法的大部分性能与Persona聊天数据集相似，但不同之处在于说话人在所有方法中获得的Dist-1分数最高。通过仔细分析生成的案例，我们发现所有的非元学习方法（仅限Pretrain和Finetune）一致地生成随机的单词序列，这意味着它们在这项任务的少量镜头设置中完全失败。然而，基于元学习的方法仍然存在。

任务一致性。在这两个数据集上，Finetune方法与Pretrain-Only方法相比，在C得分、E-acc和任务一致性上没有显著差异，说明简单的Finetune方法对提高任务一致性没有作用。包括MAML和cmam在内的所有元学习方法都优于Finetune。与MAML-Seq2seq和MAML Seq2SPG相比，cmam-Seq2SPG的C评分和Eacc分别提高了22.2%/12.5%和11.8%/5.6%。这意味着CMAMLSeq2SPG中的私有模块被很好地修剪，以便更好地描述每个任务的独特特性。

我们还观察到，在MojiTalk中，CMAMLSeq2SPG在BLEU得分上与其他基线相比有很好的提高，但在E-acc和任务一致性得分上与Persona chat相比有有限的提高。这说明当训练数据有限时，生成模型倾向于关注响应的正确性，而不是任务的一致性。

通过对响应质量和任务一致性度量的联合分析，我们可以很容易地得出结论：在CMAML-Seq2SPG中，我们的算法产生的响应不仅在响应质量上优越，而且符合相应任务的特点。

模型差异。尽管任务之间的高分并不能说明每个模型都捕捉到了其独特的特征，但是一组能够捕捉到自身特征的模型会有更高的分。因此，我们提出了竞争方法的差异得分作为参考指标。在表1中，我们可以看到，对非元学习方法（仅限Pretrain和Finetune）的微调并不能增强任务之间的模型差异。MAML有助于增加模型差异，但不如所建议的cmaml方法。CMAMLSeq2SPG在两个数据集上的模型差异得分最高，因为它在参数和模型结构级别上区分了不同的任务。

一种方法的得分越高，其生成的对话模型就越容易微调。所有非元学习方法的得分都比MAML方法低得多。CMAMLSeq2SPG在这两个数据集上的得分最高，这表明私有模块中的活动边更有可能被微调以更好地适应相应任务的语料库。我们还观察到CMAML-Seq2SP0G的得分相对较低，这表明其基代模型Seq2S0G不如Seq2SPG。

**5.6影响因素**

我们进一步研究了可能对绩效有很大影响的两个因素：训练数据的数量和任务之间的相似性。

Few-shot设置。我们只使用Persona聊天数据集进行分析，因为MojiTalk的数据太少，无法进一步减少。在人物角色聊天中，每个用户平均有121个训练样本，我们在表2中的100和110个样本设置（训练和测试中）中评估所有方法，因为当每个任务包含少于100个样本时，所有方法都倾向于生成随机序列。

对于非元学习方法，包括只有预训练和细调，质量分数随着训练数据量的增加而提高，而C分数几乎保持不变，这些方法对任务间的差异不敏感。随着数据的增长，MAML方法在BLEU评分上没有太大的变化，但其C评分不断增加。CMAML-Seq2SPG的BLEU评分和C评分都随着数据的增长而不断提高，在所有任务中始终取得最佳性能。这证明了定制的生成模型适合于相应的任务，能够充分发挥训练数据的潜力。

任务相似性。同样，我们只使用Personachat数据集，因为我们无法定义表情符号之间的相似性。我们构建了两个数据集：一个包含100个相似的用户，另一个包含100个不同的用户（训练和测试中）。

在类似的用户设置中，所有方法的性能都很接近。这意味着基于元学习的方法在类似的任务中没有优势。在不同的用户设置中，cmam-Seq2SPG在C分数和BLEU上表现最好。我们的结论是，用户相似度会影响模型的性能。与不同用户设置相比，相似用户设置的BLEU值较高，而C值较低。可能的原因是生成模型没有区分相似的任务，在训练中把所有的任务看作一个任务。

**5.7 Case Study**

由于表3中的空间有限，我们只在Persona聊天数据集中呈现一个案例。仅预训练和精细调整方法产生的信息较少的一般反应。由于初始参数易于微调，MAML方法往往会产生不同的响应。即使用户配置文件不用于培训，CMAML-Seq2SPG可以从训练对话中快速学习“儿科医生”的角色信息，而其他基线则不能。从另一个角度看，CMAMLSeq2SPG中修剪的私有模块可以看作是一种特殊的存储器，它存储特定于任务的信息，而不需要明确定义存储单元。

**6 Conclusion**

本文讨论了few-shot对话生成问题。我们提出了CMAML，它能够为不同的任务定制独特的对话模型。CMAML为每个任务的对话模型引入了一个私有网络，其结构将在训练过程中不断演变，以更好地适应该任务的特点。私有模块将只接受相应任务及其类似任务的语料库培训。实验结果表明，CMAML在响应质量、多样性和任务一致性方面都取得了最好的性能。我们还测量了任务之间的模型差异，结果证明CMAML为不同的任务生成了不同的对话模型。