**FEW-SHOT TEXT CLASSIFICATION WITH DISTRIBUTIONAL SIGNATURES**

**会议**：ICLR2020

**代码**：https://github.com/YujiaBao/Distributional-Signatures

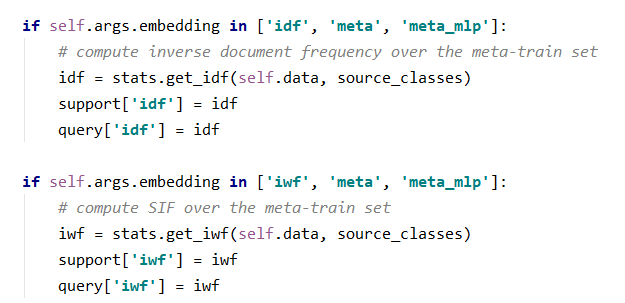
**概述**

我们的模型不仅从单词中学习，还利用了它们的分布特征，这些特征编码了相关的单词出现模式。我们的模型是在元学习框架内训练的，将这些特征映射到注意力得分中，然后用注意力得分来衡量单词的词汇表示。

在计算机视觉中，底层模式（如边缘）及其对应的表示可以在任务之间共享。然而，对于大多数任务在词汇层操作的语言数据，情况是不同的。对于一项任务来说信息量很大的单词可能与其他任务无关。比如在医药领域，可能sleep是一个很重要的词，在政治领域Trump是一个很重要的的词，但是当在不同领域的时候他们的重要性会下降

我们证明，尽管存在这些变化，我们仍然可以有效地跨类传递表示，从而使学习能够在低资源状态下进行。我们的方法没有直接考虑单词，而是利用它们的分布特征，即潜在单词分布的特征，这些特征在分类任务中表现出一致的行为。在元学习框架中，这些特征使我们能够跨任务转移注意力，从而可以用来衡量单词的词汇表示。

在此基础上，我们希望学习在跨类传输的上下文中使用分布签名（distributional signatures）。除了词频之外，我们还根据一个特定的类来评估单词的重要性。由于标记数据的稀缺性，后一种关系无法可靠地估计目标类。然而，我们可以利用为目标类提供的少数训练实例来获得该指标的噪声估计，然后在元学习框架内进一步细化该近似。我们注意到，虽然分布签名的表现力弱于其对应的词汇，但基于分布签名的元知识能够更好地泛化。



我们的模型由两部分组成。第一个是注意生成器（attention generator），它将分布特征转换为反映单词分类重要性的注意力分数。根据注意力生成器的输出，我们的第二个组件，岭回归器（ridge regressor），在只看到几个训练示例后，就可以快速地学习做出预测。注意生成器在所有的事件（Episode）共享，而岭回归器则在每个单独的事件中从头开始训练。后者的预测损失为注意生成器提供了监督。理论上，我们证明了注意生成器对词替换扰动是鲁棒的。

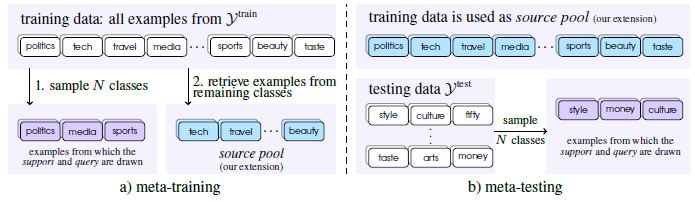
**相关工作**

**元学习**。正如我们的实验所表明的，图像数据和语言数据在可转移知识方面存在着内在的差异，而词汇感知的元学习器无法在标准的多类分类数据集上进行归纳。在这项工作中，我们观察到，尽管文本中的显著特征可能是不可转移的，但它们的分布行为是相似的。因此，我们专注于学习单词重要性和分布签名之间的联系。因此，我们的模型能够可靠地从新类中识别出重要的特征。

**迁移学习**。我们的工作还与迁移学习密切相关：我们假设可以从源类访问大量标记的示例，并且我们希望确定目标分类任务的单词重要性。当前方法通过微调预先训练的编码器或使用共享编码器进行多任务学习将知识从源传递到目标。最近，Bao等人。（2018）还成功地通过人的理性转移了特定任务的注意力。与这些方法相比，在预先设计传输机制的情况下，我们学习根据下游任务的性能进行传输。具体来说，我们利用分布统计来转移任务间的注意力。

**任务背景**

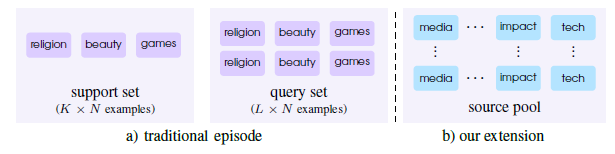
**问题陈述**。假设我们从一组类中得到了标记示例。我们的目标是开发一个从这些训练数据中获取知识的模型，以便我们可以对新的（但相关的）类进行预测，对于这些类，我们只有一些注释。这些新类属于一组类，与不相交。

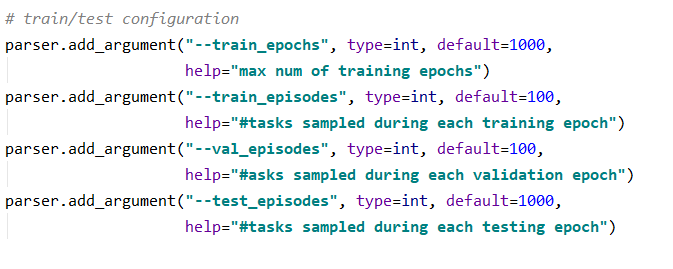


*图三。Episode generation。a） 元训练：首先，从中抽取N个类。然后，从N个类中抽样支持集和查询集。我们使用剩余类中的示例来形成source pool。b） 元测试：从Ytest中选择N个新类，并从这N个类中抽取支持集和查询集。我们使用来自的所有示例来形成source pool。*

**元训练**。在元学习中，我们在元训练期间模拟上述测试场景，因此我们的模型学习从一些注释中快速学习。为了创建一个单独的训练episodes，我们首先从Ytrain中抽取N个类。对于这N个类中的每一个类，我们分别以K个示例作为训练数据和L个示例作为测试数据。我们根据这些测试数据的损失来更新我们的模型。图4a显示了一个episode的示例。我们重复此过程以增加训练episodes的数量，每个训练集都是在它自己的N个类集合上构造的。在文献中，一个事件的训练数据通常表示为支持集，而相应的测试数据称为查询集。在给定支持集的情况下，我们将通过查询集进行预测的任务称为N-way K-shot分类。

**元测试**。在完成元训练之后，我们使用相同的基于episodes的机制来测试我们的模型是否真的能够快速适应新的类。为了创建一个测试episodes，我们首先采样来自Ytest的N个新类。然后我们从N个类中抽样支持集和查询集。我们评估所有测试episodes中查询集的平均性能。

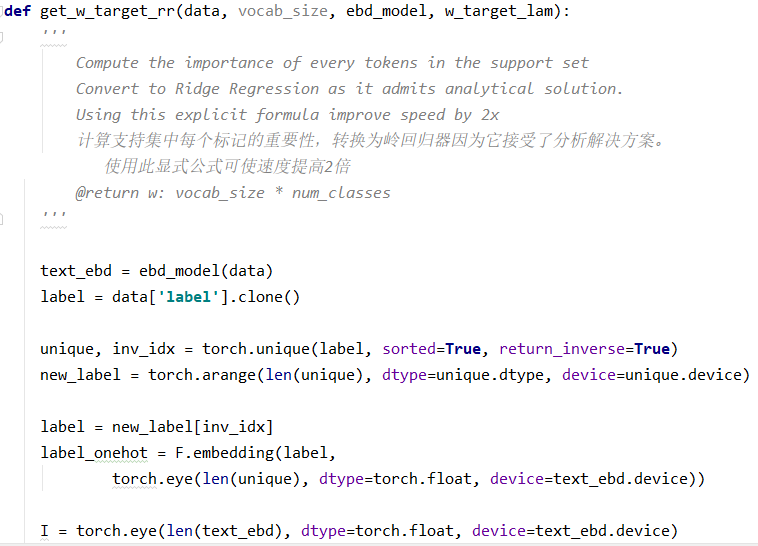
*图四。单episodes N=3，K=1，L=2。矩形表示输入示例。里面的文本对应于它们的标签。一个事件包含一个支持集、一个查询集和一个源池。*

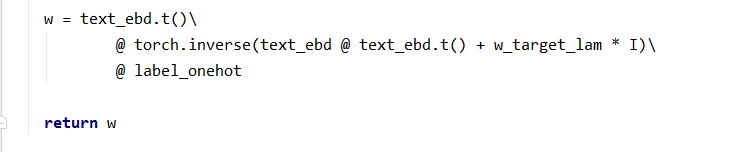


**我们的扩展**。我们观察到，尽管Ytrain中的所有示例都可以通过元训练访问，但标准元学习框架仅在每个训练episodes从这些数据的小子集中学习。相比之下，我们的模型利用分布统计对所有训练示例进行更稳健的推断。为了适应这种调整，我们用一个源池来扩充每一episodes（图4b）。在元训练（图3a）期间，这个源代码池包含了没有为特定episodes选择的训练类中的所有示例。在元测试期间（图3b），这个源池包括所有的训练示例。

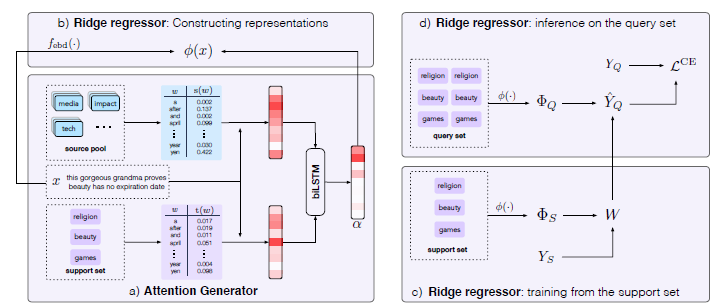
**方法**

**总体结构**。我们的目标是通过从输入的分布特征中学习高质量的注意力来提高小样本分类性能。给定一个特定的事件，我们从源池和支持集中提取相关的统计信息。由于这些统计数据只是粗略估计单词分类的重要性，因此我们使用注意力生成器（attention generator）将它们转换为对单词进行操作的高质量注意力。由此产生的注意力为下游的预测者岭回归器（ridge regressor）提供了指导，以便从一些标记的例子中快速学习。





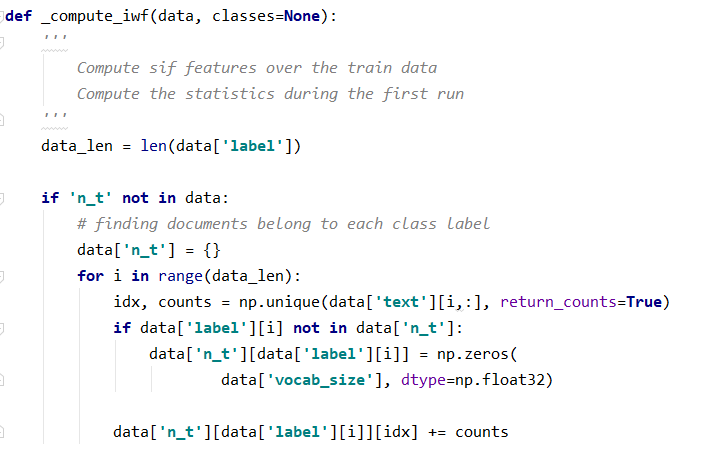
我们注意到注意力生成器在所有训练episodes中都是优化的，而岭回归器则在每个片段中从头开始训练。图5展示了我们模型的两个组件。

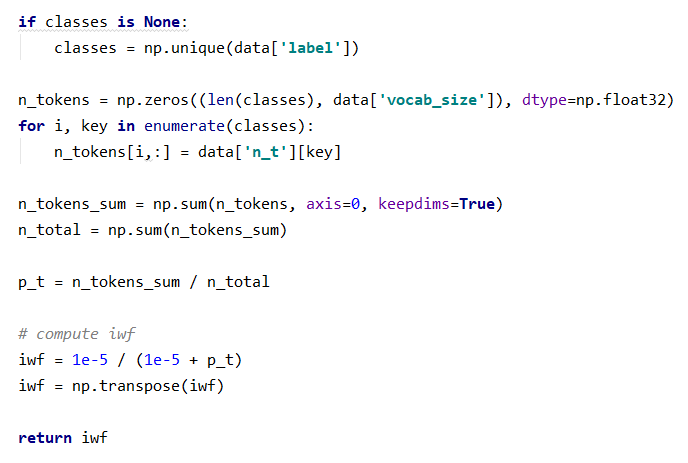


*图5:N=3，K=1，L=2的一episode的模型说明。注意生成器将来自源池和支持集的分布签名转换为每个输入示例x（5a）的注意。岭回归器利用产生的注意来加权词汇表征（5b）。然后它从支持集（5c）学习并通过查询集（5d）进行预测。*

**注意生成器**：这个模块通过合并源池和支持集的分布统计信息来生成类特定的注意（图5a）。产生的注意为岭回归器提供了对单词重要性的归纳偏向。我们根据岭回归器的反馈来训练这个模块。注意生成器的目标是根据每个输入示例的分布特征来评估单词的重要性。分配签名有多种选择。其中，我们重点讨论了对单词替换扰动具有可证明鲁棒性的单元统计函数。我们利用大的源池来通知一般单词重要性模型，并利用小的支持集来估计类特定单词的重要性。生成的注意力稍后将用于构造下游分类的输入表示。经常出现的词语不太可能提供信息，因此，我们想降低常用词权重和提升稀有词权重。为了衡量一般单词的重要性，我们选择了

其中，是输入实例X的第i个字，是上的源池的单义似然。



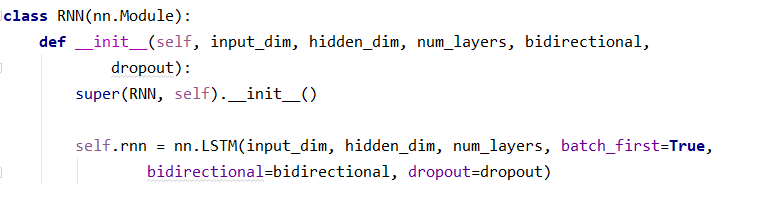


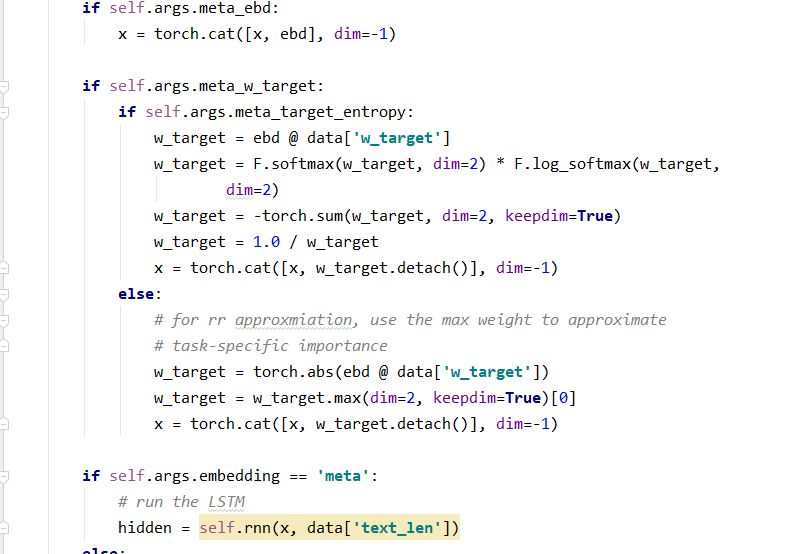
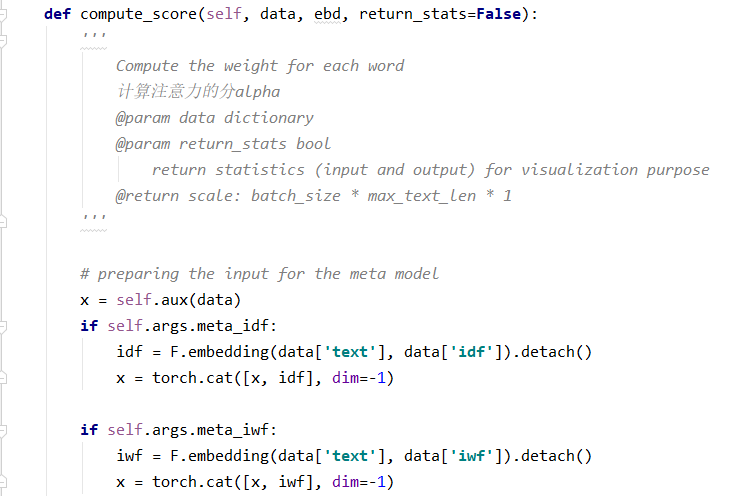
另一方面，支持集中有识别力的词在查询集中可能也有识别力。因此，我们定义以下统计数据来反映特定类别的单词重要性：

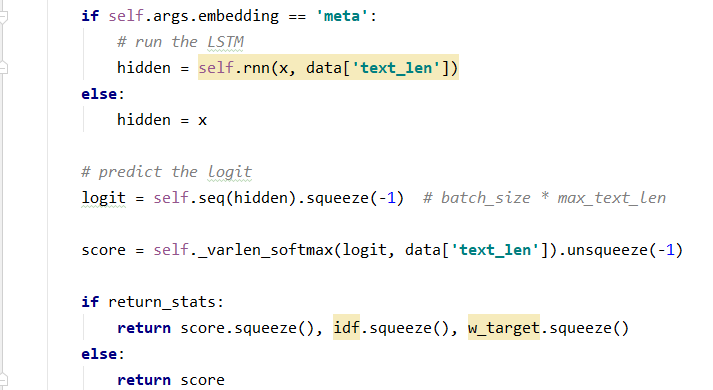
在支持集上使用正则线性分类和估计条件似然是熵算子。我们注意到度量给定单词的类标签y的不确定性。因此，呈现倾斜分布的单词将具有很高的权重。

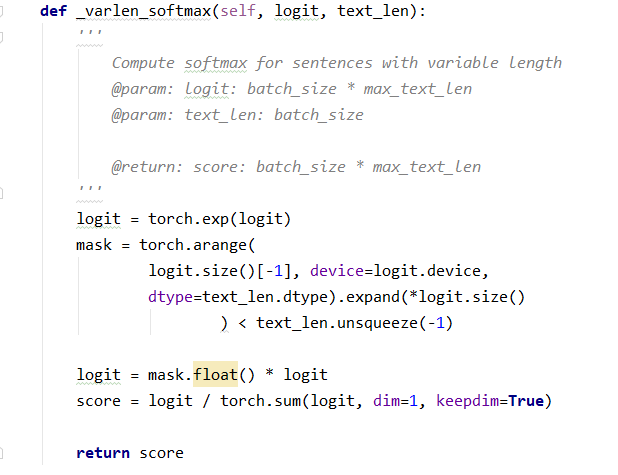
直接应用这些统计数据可能不会产生良好的性能，原因有二：1）这两个统计数据包含互补信息，并且不清楚如何将它们组合在一起；2）这些统计数据是用于分类的单词重要性的噪声近似值。为了弥补这一差距，我们将这些签名串联起来，并采用双向LSTM来融合输入端的信息：，最后，我们使用点积注意预测词的注意分数：

其中是位置i处的biLSTM的输出，v是可学习向量。





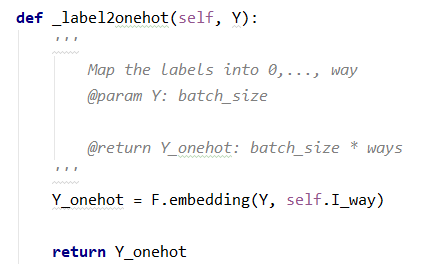




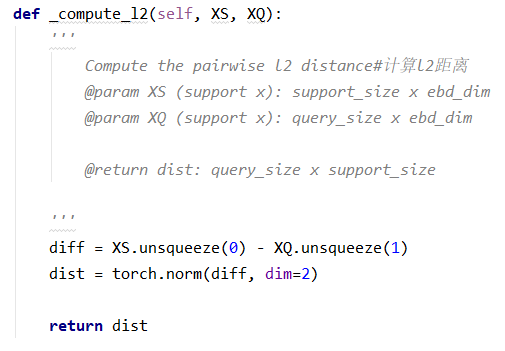
**岭回归器**。对于每一episode，这个模块使用从分布特征中获得的注意力构建词汇表示（图5b）。这个模块的目标是在从支持集学习之后，通过查询集进行预测（图5c和5d）。它的预测损失相对于注意生成器是端到端可微的，这导致了有效的训练。在注意生成器的提示下，岭回归器在看到一些例子后，很快学会了做出预测。首先，对于给定事件中的每一个例子，我们构建一个关注重要单词的词汇表征，如注意力得分所示。接下来，给出这些词汇表示，我们从头开始在支持集上训练岭回归器。最后，我们通过查询集进行预测，并使用损失来教导注意生成器产生更好的注意。构造表征，鉴于不同的词对分类显示出不同的重要性，我们构造有利于相关词的词汇表征。具体来说，我们将示例x的表示定义为

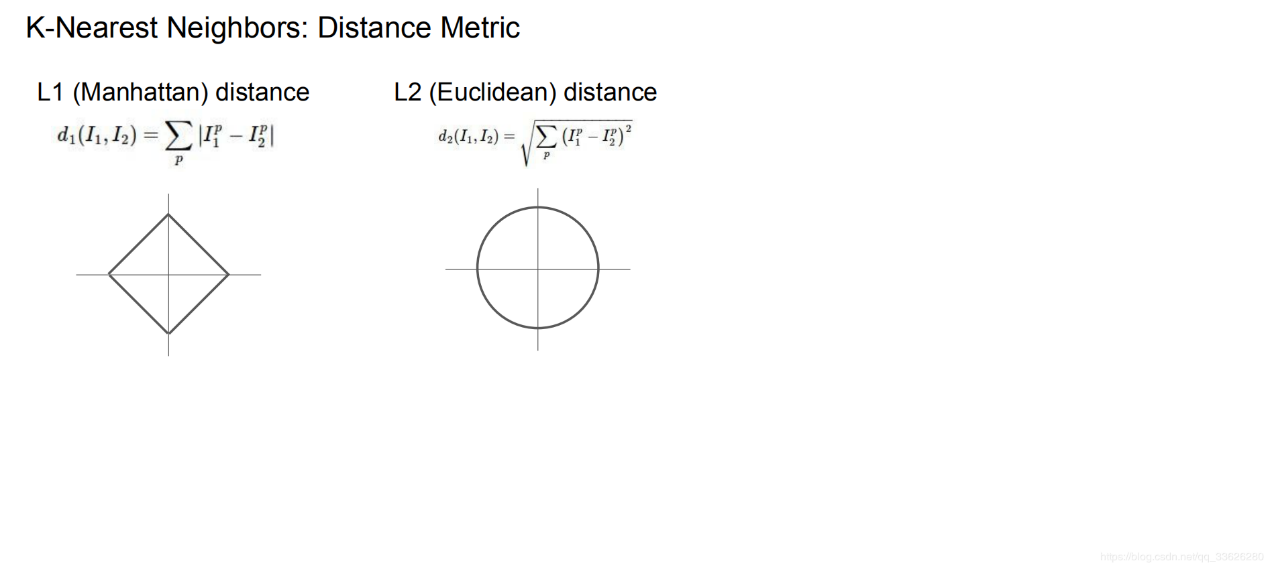
其中是一个预先训练的嵌入函数，它将一个单词映射到。

支持组的培训。给定一个N-way-K-Shot分类任务，让 表示从获得的支持集，而 表示one-hot标签。



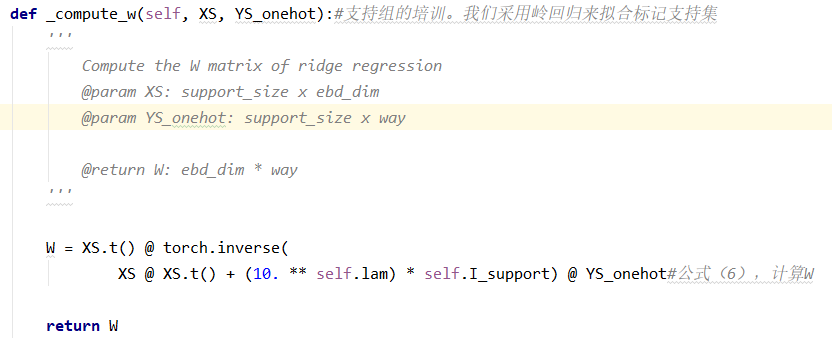
我们采用岭回归来拟合标记支持集，原因如下：1）岭回归允许通过模型实现端到端微分的封闭解；2）通过适当的正则化，岭回归减少了对小支持集的过度拟合。具体来说，我们最小化正则化平方损失

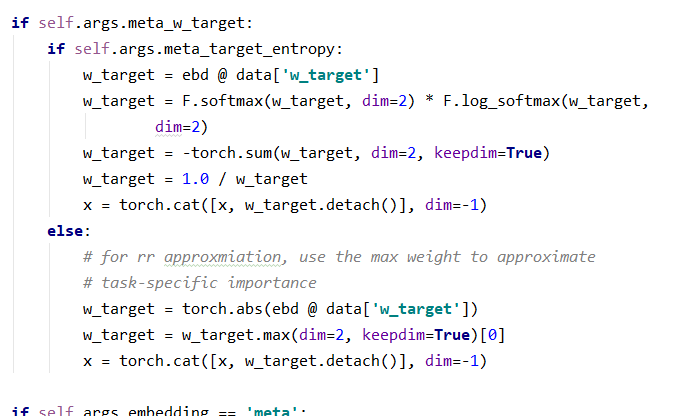




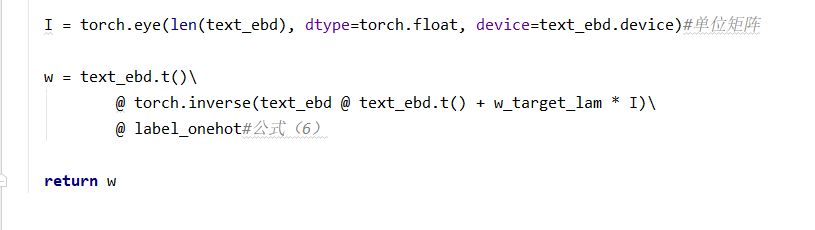
在权重矩阵。这里表示Frobenius范数，而控制学习变换W的条件。闭式解可获得为

其中I是单位矩阵。

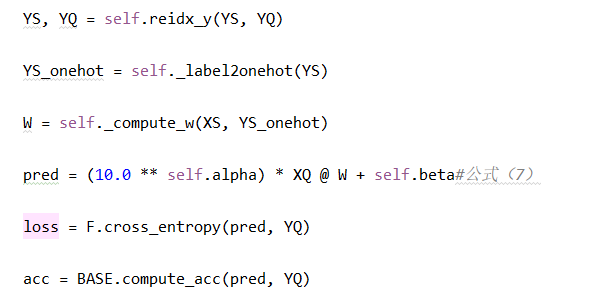


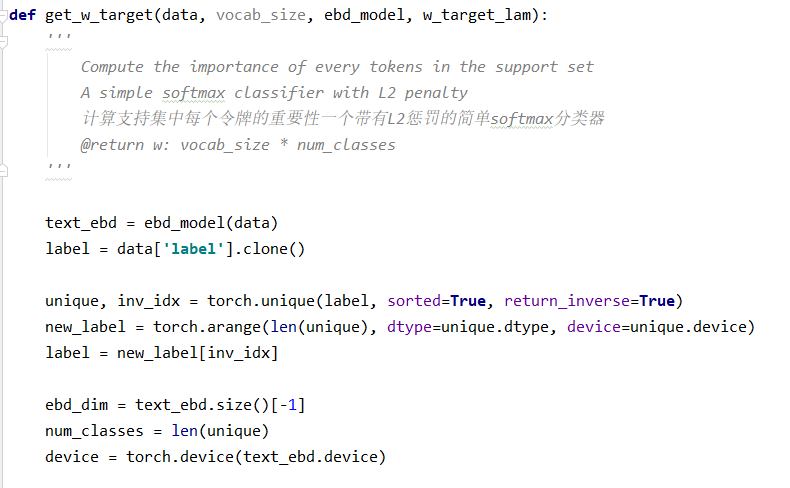


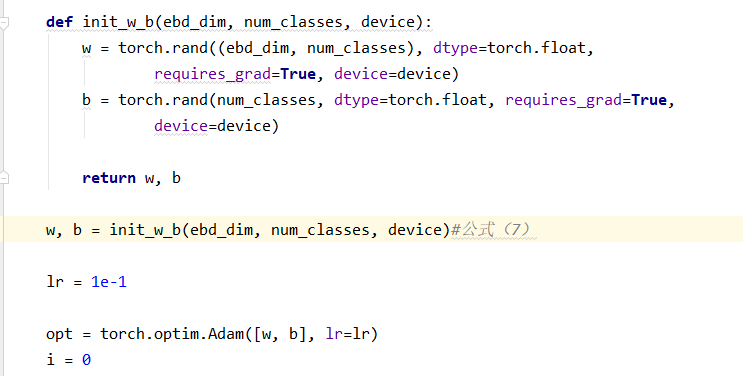


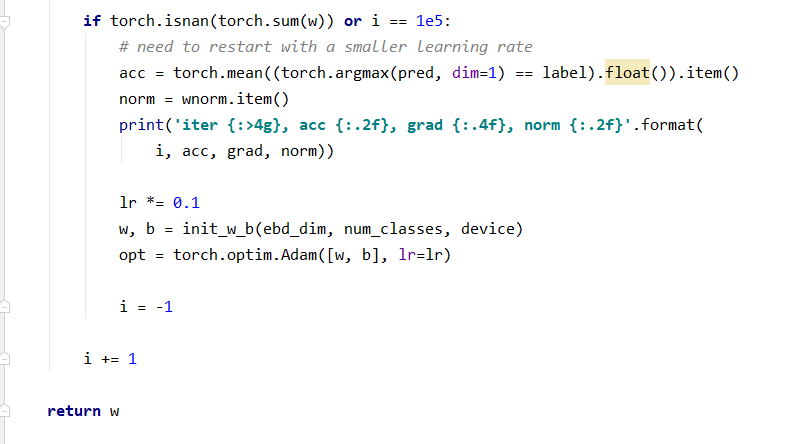
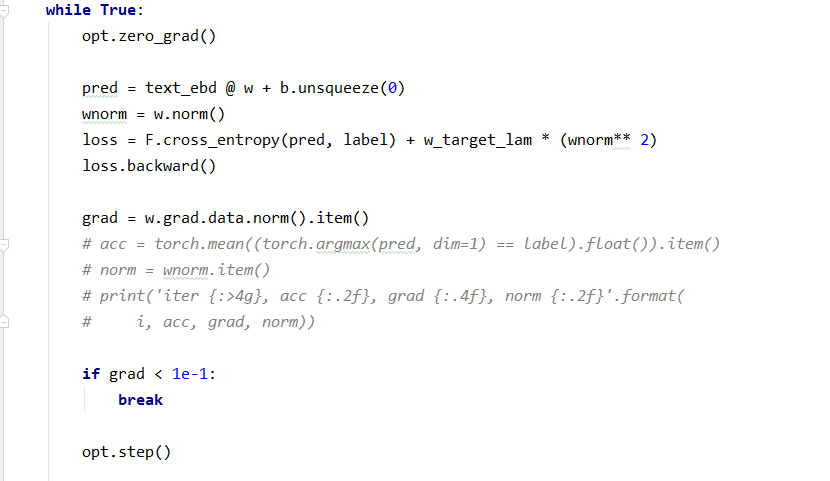


对查询集的推断。设表示查询集的表示形式。尽管我们针对等式5中的回归目标进行了优化，但经过校准步骤，学习的变换在小样本分类中表现良好，如









其中是通过元训练学习的元参数。最后，我们在上应用softmax来获得预测概率。请注意，此校准仅调整softmax的temperature和scale；其mode保持不变。在元训练过程中，我们计算查询集上和标签之间的交叉熵损失。由于和都取决于，为注意力生成器提供监督。

**理论分析**。在我们的理论分析中，我们证明了注意生成器的输出对字替换扰动是不变的，使用分布式签名带来了经验证的对输入扰动的鲁棒性。形式上，设为单个事件，其中是源池，是支持集，是查询集。对于任何输入文本，注意生成器根据支持集和源池生成单词级的重要性：现在考虑一个单词替换扰动，它用替换一个单词w。如果我们任意地交换单词，我们可能会遇到无意义的输出，因为重要的单词可能会被常见的单词替换，如“a”或“the”。因此，我们考虑保留单词的单程序概率的扰动，在源池中估计：表示所有。这样，重要的词（对于一个类）被映射到类似的重要词（对于另一个类）。在逐字应用后，我们使用和来表示支持集和查询集。

***Theorem 1.*** 假设：满足所有w的。如果是双射，那么对于任何输入文本及其扰动，注意生成器的输出是相同的：这个证明背后的思想是要证明一般单词重要性s和类特定单词重要性t对于这种扰动是不变的。或者，可以将这个单词替换扰动视为对输入分布的更改。设为给定类标签y的输入x的条件概率，假设是一个双射，扰动后的条件可表示为，其中是对x中的每个单词应用的结果。直观地说，定理1告诉我们，如果单词w是原始分类任务的判别特征，那么在输入分布改变后，单词应该是新任务的区别特征。这个属性很有意义，因为用于分类的单词重要性应该只取决于类之间的相对差异。事实上，当注意力生成器的输入是unigram计数的任何函数时，这个定理成立。我们还注意到，由于预先训练的嵌入函数在这种扰动下通常不是不变的，因此查询集上的分类性能可能不同。

**实验**

**数据集：**

***20 Newsgroups***由来自新闻讨论论坛的非正式话语组成（Lang，1995）。文件按20个主题组织。

***RCV1***是路透社1996年至1997年的新闻通讯文章集（Lewis等人，2004）。这些文章都是用正式的语言写的，并用一组主题代码标记。我们将71个二级主题视为总类集，并丢弃属于多个类的项目。

***Reuters-21578***是1987年路透社短篇文章的集合（Lewis，1997）。我们使用数据集的标准ApteMod版本。我们丢弃带有多个标签的文章，并考虑至少有20篇文章的31个类。

***Amazon product data***包含24个产品类别的客户评论（He&McAuley，2016）。我们的目标是将评审分为各自的产品类别。由于原始数据集非常大（14.28亿条评论），我们从每个类别中抽取1000条评论来选择一个更容易处理的子集。

***HuffPost headlines***由2012年至2018年在赫芬顿邮报上发布的新闻头条组成（Misra，2018）。这些标题分成41个类。它们比正式的句子短，语法性差。

***FewRel***是一个关系分类数据集，用于小样本学习（Han等人，2018）。每个例子都是一个单独的句子，用一个头实体、一个尾实体和它们之间的关系进行注释。目标是预测头部和尾部之间的正确关系。公共数据集包含80个关系类型。

**5.2基线**

我们将我们的模型（表示为我们的）与表示和学习算法的不同组合进行比较。基线详情见附录A.11

表示 我们评估三个表示。AVG将每个示例表示为其嵌入的平均值。IDF将每个示例表示为其单词嵌入的加权平均值，所有训练示例的权重由逆文档频率给出。CNN对输入词应用1D卷积，并通过max随时间池获得表示（Kim，2014）。

学习算法除了岭回归（RR）（bertineto等人，2019）外，我们还评估了两种标准监督学习算法和两种元学习算法。nn是欧氏距离下的1近邻分类器。FT在所有训练示例中预先训练分类器，然后使用支持集优化网络（Chen等人，2019）。MAML meta学习一个优先于模型参数的参数，以便模型能够快速适应新的类（Finn等人，2017）。原型网络（PROTO）通过元学习一最小化个度量空间来进行小样本分类每类质心与其组成示例之间的欧几里德距离（Snell等人，2017）。

**5.3实施细则**

我们使用预先培训的fastText嵌入（Joulin等人，2016）作为模型和所有基线。对于句子级数据集（HuffPost、FewRel），我们还使用HuggingFaces代码库（Wolf等人，2019）对预先训练的BERT嵌入（Devlin等人，2018）进行了实验。对于关系分类（FewRel），我们使用位置嵌入来增加注意力生成器的输入（Zhang等人，2017）。在注意力生成器中，我们使用具有50个隐藏单元的biLSTM，并对输出应用0.1的dropout（Srivastava等人，2014）。在岭回归器中，我们在对数空间中优化元参数和a以保持正性约束。所有参数均使用Adam优化，学习率为0.001（Kingma&Ba，2014）。

在元训练中，我们每一个episodes抽取100个训练片段。当20个阶段的验证损失没有改善时，我们应用提前停止。我们根据1000个测试集评估测试性能，并报告5个不同随机种子的平均准确性。

**6.结果**

我们评估了我们的模型在5-way-1-shot和5-way-5-shot设置。这些结果见表1。我们的模型在所有数据集上始终实现最佳性能。与每个数据集的最佳基线相比，我们的模型平均提高了7.5%的5-way-1-shot准确率和3.9%的5-way 5-shot准确率。与CNN+PROTO相比，我们的模型在单样本分类中平均提高了20.0%。实证结果清楚地表明，元学习者对词汇信息的依赖始终是失败的，而我们的模型能够概括过去特定类别的词汇。此外，图6说明了词汇感知元学习者（CNN+PROTO）能够比我们的模型更快地过拟合训练数据，但是我们的模型更容易推广到未见的类。

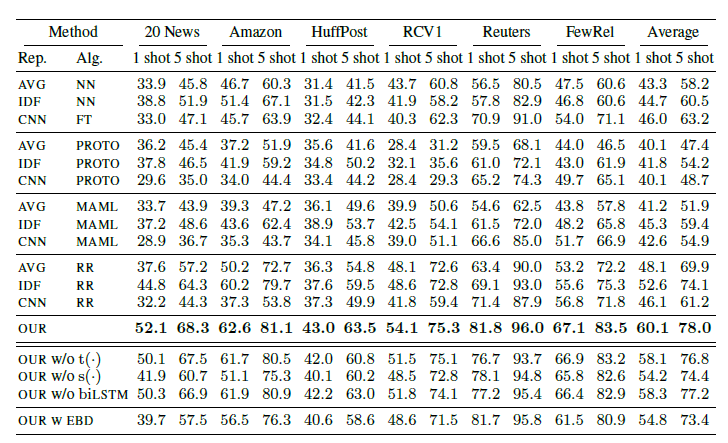


表1：6个数据集的5-way-1-shot和5-way-5-shot分类结果。下面四行是我们的消融研究。关于标准偏差的完整结果，见附录A.12中的表8和表9。

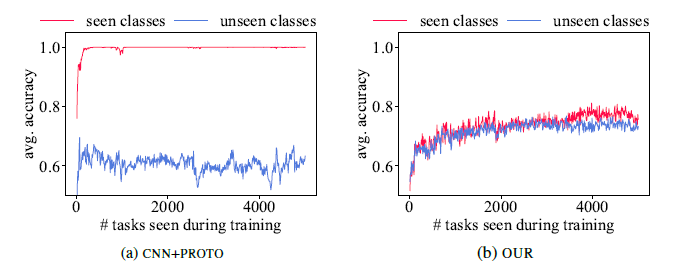


图6：路透社数据集上CNN+PROTO（左）v.s.OUR（右）的学习曲线。我们绘制了50集以上的平均5-way-1-shot准确率，这些片段分别来自已见类（蓝色）和未见类（红色）。虽然我们的表现力较弱，但它能更好地推广到未见的类。

**消融研究** 我们对注意力发生器进行消融研究。这些结果显示在表1的底部。我们观察到，统计数据s（）和t（）对性能都有贡献，尽管前者的影响更大。我们还注意到，与使用多层感知器（our w/o biLSTM）独立计算每个单词的单词重要性不同，使用biLSTM融合输入中的信息稍微提高了性能。

最后，我们观察到将元知识限制在分布式签名上对性能至关重要：当词汇词嵌入和分布式签名都被输入到注意生成器（our w EBD）时，性能会持续下降。

**上下文表示** 对于句子级数据集（FewRel，HuffPost），我们也用BERT给出的上下文表示进行了实验（Devlin等人，2018）。这些结果如表2所示。虽然BERT显著地提高了FewRel的分类性能，但是我们在HuffPost上没有发现性能提升。我们假设这种差异是因为关系分类是高度上下文性的，而新闻分类主要是基于关键字的。

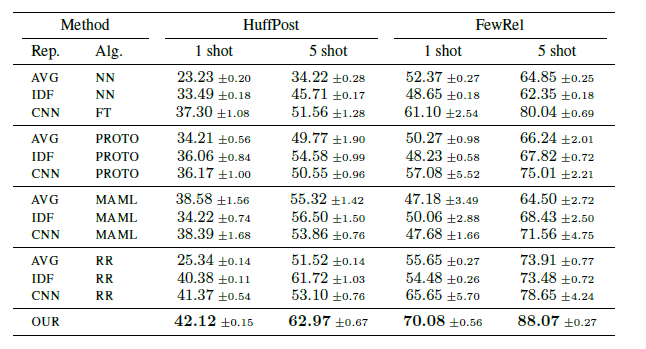


Table 2: 5-way 1-shot and 5-way 5-shot classification on HuffPost and FewRel using BERT.

**分析** 我们在图7中看到了我们的注意力加权表征。与直接使用一般单词重要性s（x）或类特定单词重要性t（x）相比，我们的方法产生了更好的分离，这使得能够从几个例子中有效地学习。这强调了我们方法的力量不在于分布签名本身，而在于在它们上面学习的表示。

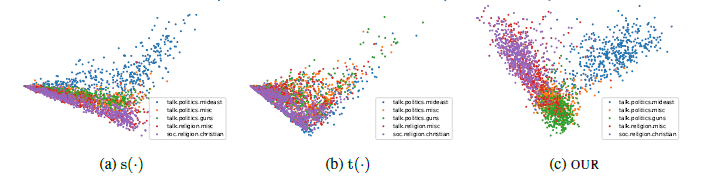


图7：从20个新闻组采样的测试集（N=5，K=5，L=500）的查询集的输入表示的PCA可视化。（a）s（），（b）t（）和（c）在一组不相交的训练类上训练的注意生成器元给出的单词嵌入的加权平均值。

图8在两个测试场景中可视化了同一个查询示例上的模型输入和输出。这个例子属于路透社数据集中的类作业。首先，我们观察到我们的模型从噪声分布签名中产生有意义的注意。此外，生成的注意是特定于任务的：在所描述的示例中，如果该事件包含其他与经济学相关的类，“统计”一词被我们的模型压制。相反，当我们将工作与其他遥远的阶级进行比较时，“统计”的权重会上升。

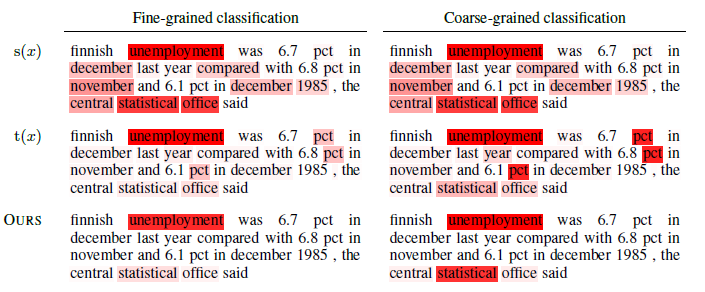


图8：我们的模型生成的注意权重是特定于任务的。我们将模型的输入s（x）（顶部）、t（x）（中间）和输出（底部）可视化为一个来自Reuters数据集中的类作业的查询示例。与其他经济类（左）相比，“统计”一词在就业方面的分量较轻，但在考虑不同的经济类（右）时，它变得重要。细粒度分类：就业、零售、工业生产指数、批发生产指数、消费生产指数。粗粒级：乔布斯、可可、铝、铜、储量。

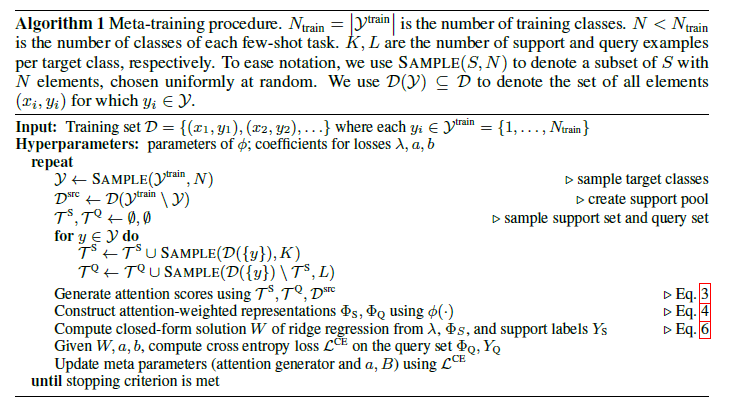
**总结**

本文提出了一种新的元学习方法，该方法利用词的重要性和分布特征之间的联系来改进小样本分类。具体来说，我们学习了一个注意力生成器，它将分布统计转化为高质量的注意力。由此产生的注意力为快速适应新的分类任务提供了指导。文本分类和关系分类的实验结果都验证了我们的模型能够识别新类的重要词。我们的方法的有效性证明了使用分布式签名进行元学习的前景。

**A 补充材料**

A．3学习过程

算法1包含我们学习过程的伪代码。当20个阶段的验证损失没有改善时，我们应用提前停止。



A．4数据集

为了可靠地测试模型的跨类泛化能力，我们在实验中考虑了两种数据分割机制：1）简单分割：随机排列所有类并将它们分割成train/val/test；2）硬分割：根据类层次结构选择train/val/test，train类离val和test很远。我们对一个句子级数据集（HuffPost）和一个文档级数据集（Reuters-21578）应用了easy split。硬拆分用于其他四个数据集（详细信息如下）。该设置测试算法的泛化能力，遵循Xian等人的说法。（2017年）。

**20 Newsgroups**每类20 Newsgroups属于6个顶级类别之一，大致相当于计算机、娱乐、科学、政治、宗教和待售。我们对标签集进行分区，以便没有顶级类别跨越两个拆分。列车包含“sci”和“rec”，val包含“comp”，test包含所有其他标签。

**Amazon**数据集没有预定义的顶级类别。为了产生一个硬分割，我们首先根据类的词分布对类应用谱聚类。然后从不同的集群中选择train/val/test。

**RCV1**我们采用与上述相同的方法。

**FewRel** 虽然FewRel没有提供更高级别的类别，但是我们观察到大多数关系发生在相似类型的命名实体之间。因此，我们使用预先训练的spaCy解析器为每个示例提取头和尾的命名实体类型。8对于每个类，我们确定最常见的头和尾实体类型。Test包含最常见的head实体类型为艺术品的所有类。训练和验证被任意分割以包含剩余的关系。

A． 5其他基线

我们还考虑了另外两个基线：induction network（Geng et al.，2019）和P-MAML（Zhang et al.，2019）。实施细节见附录A.11。

归纳网络使用具有自我关注池的biLSTM对输入示例进行编码（Lin等人，2017年）。基于编码表示，然后通过支持集上的动态路由计算每个类的原型（Sabour等人，2017）。最后，它使用神经张量层（Socher等人，2013）来预测每个查询示例和类原型之间的关系。

P-MAML结合了预训和MAML。它首先使用蒙面语言建模细化元训练数据上的预训练BERT表示（Devlin等人，2018）。基于这种精细的表示，它训练一阶MAML Finn等人。（2017）实现快速适应。

表4显示使用分布签名（OUR）的学习在所有数据集上都明显优于两个基线。这并不奇怪：这两个基线都是在词法表示的基础上构建元知识的，当看到的类和未看到的类之间的词法分布有很大不同时，这一点可能不会通用。图9显示了归纳网络的学习曲线。

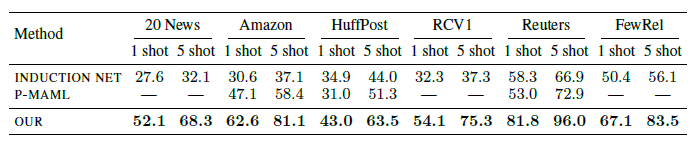


表4：与归纳网（Geng等人，2019）和P-MAML（Zhang等人，2019）的比较。我们对文档较短的文本分类数据集运行P-MAML，对其进行微调是可行的（在字块标记化之后，来自“较长”数据集的文档超过了BERT的512个标记的最大长度）。

类似于图6，我们看到归纳网络过度适合元训练类，而对元测试类的泛化较差。图10描述了在P-MAML的微调过程中，BERT的蒙面语言模型目标的困惑。再次，我们看到元测试类和元测试类表现出巨大的词法不匹配，因为这种困惑只在元测试类上略有改善。

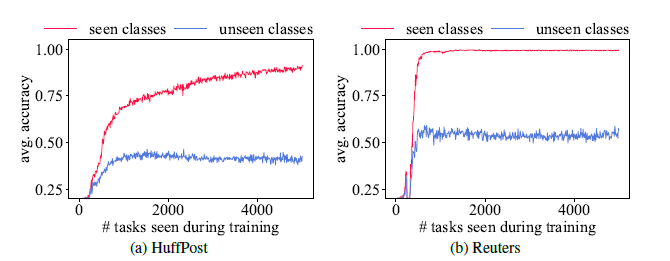


图9:Huffpost和Reuters上归纳网的学习曲线。我们绘制了50集以上的5-way-5-shot平均准确度，样本来自已见类（蓝色）和未见类（红色）。

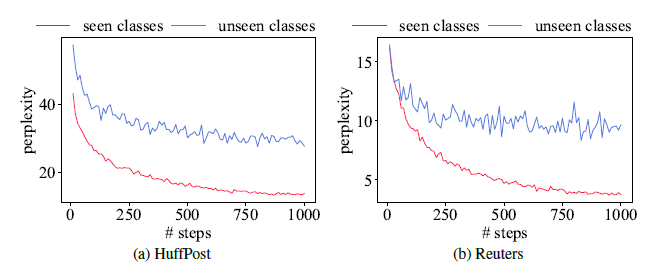


图10:P-MAML中BERT语言模型微调的困惑。元测试类与元训练类的词汇分布不匹配。

A．6学习表征分析

为了进一步了解绩效提升背后的基本原理，我们对模型的实证行为进行了更详细的分析。

通过可视化嵌入空间，我们将注意力集中在20个新闻组（图11）和HuffPost（图12）中。我们观察到我们的模型比未加权平均平均平均数产生更好的分离，并且直接使用分布统计，s（x）或t（x）。例如，在20个新闻组中，我们的模型识别出三个集群：{ftalk.religation.misc}、{soc.religation.christiang}、{ftalk.politics.mideastg}和{ ftalk.politics.misc }、{ talk.politics.gunsg }，

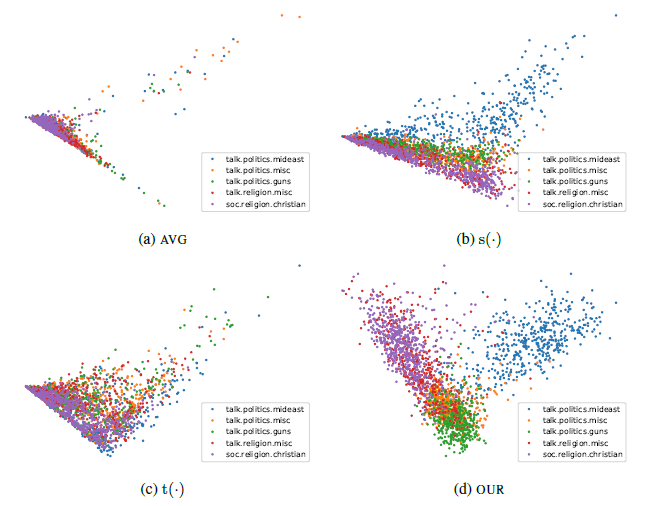


图11:20个新闻组（N=5，K=5，L=500）中测试集的输入表示的PCA可视化（查询集每个类有500个示例）。平均字嵌入。s（）：单词嵌入的加权平均值，权重由s（）给定。t（）：单词嵌入的加权平均值，权重由t（）给定。我们的：在一组不相交的训练班上训练的注意生成器元给出权重的单词嵌入的加权平均值。

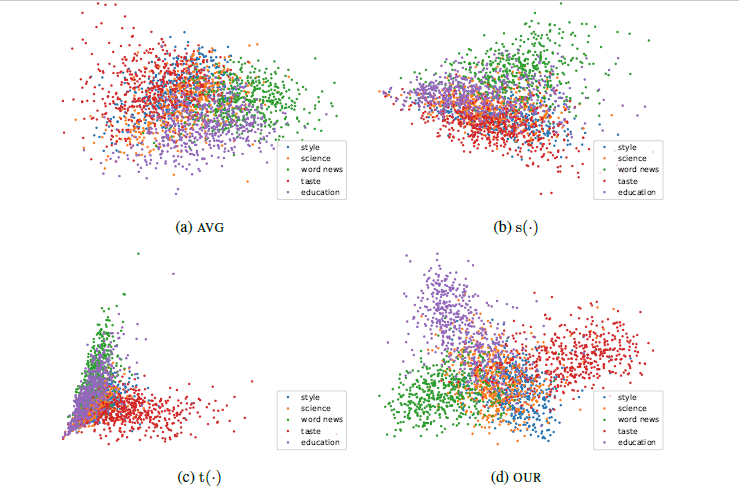


图12:N=5，K=5，L=500的《赫芬顿邮报》标题中测试事件的输入表示的PCA可视化。平均字嵌入。s（）：单词嵌入的加权平均值，权重由s（）给定。t（）：单词嵌入的加权平均值，权重由t（）给定。我们的：在一组不相交的训练班上训练的注意生成器元给出权重的单词嵌入的加权平均值。

余弦与oracle单词重要性的相似性我们还定量分析了20个新闻组和HuffPost中产生的注意（图13）。

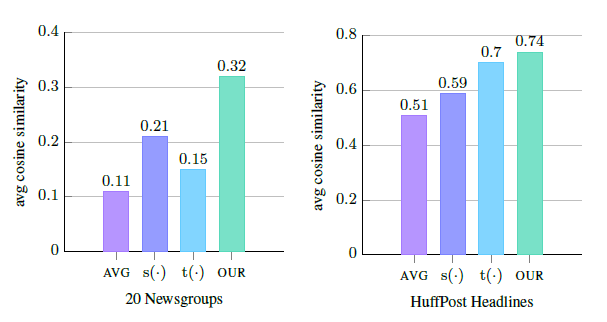


图13：在N=5，K=5，L=500的测试集的查询集中，与oracle单词重要性的平均余弦相似性。这个oracle是使用N个类中所有标记的示例来估计的。由于《赫芬顿邮报》标题中的例子短了30倍，因此该语料库中的余弦相似度更高。平均数：词的均匀分布。s（）：由s（）直接估计的单词重要性。t（）：由t（）直接估计的单词重要性。我们的：元学习注意生成器估计的单词重要性。

为了获得更可靠的单词重要性估计，我们对N个目标类中的所有示例训练oracle模型。这个oracle模型使用一个biLSTM来编码来自其单词嵌入的每个示例。然后，它基于这种编码生成一个注意力得分。为了估计单个单字的重要性，我们用这个注意分数来衡量“原始”字嵌入（不是biLSTM的输出）。利用一个隐藏层的MLP对注意加权表示进行N路分类。

与s（）和t（）相比，我们的模型产生的关注度更接近于oracle，这解释了我们的模型的巨大性能提升。注意，注意生成器在元训练期间看不到来自目标类的任何示例。

可视化生成的注意力 图14可视化了HuffPost中测试事件的生成注意力。我们观察到我们的模型从句子中识别出有意义的关键词。

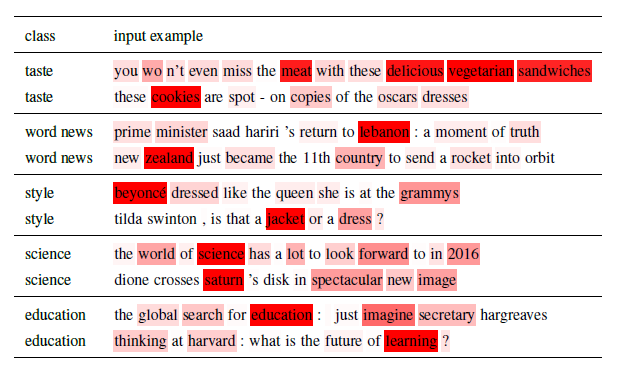


图14:Huffpost标题中5路5拍测试的10个查询示例中，我们的模型产生的注意力的可视化。

A．7 SOFTMAX校准的效果

在第4.2节中，由于脊回归的效率和有效性，我们使用softmax校准作为下游预测。为了研究softmax校准的效果，我们遵循Bertineto等人的观点。（2019）并将岭回归与二元logistic分类器（LR）的直接优化（五次迭代）进行比较。由于牛顿法在每次迭代时都允许有一个封闭形式的解，所以最终的解也是端到端可微的。对于每个N路K-shot事件，为了使二值logistic分类器适应多类预测，我们使用支持集训练N个二值（一个vsall）分类器。将N个二进制分类器的输出组合起来，对查询集进行最终预测（Bishop，2006，第4.1.2章）。

结果见表5。总的来说，RR和LR产生了相似的结果（分别是78.0对77.1对5投，60.1对58.5对5投），尽管RR表现稍好。当使用LR作为下游预测因子时，使用分布特征的学习比最佳基线分别提高了6.1%和3.2%。注意，LR需要为每一集求解N个二进制分类，这比RR效率低。

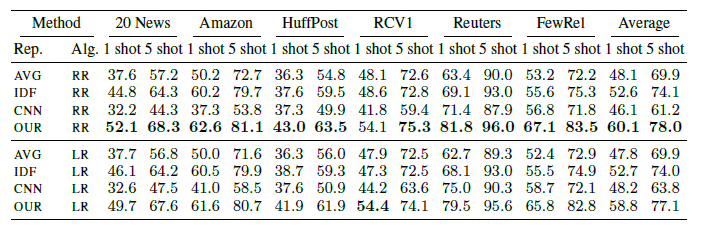


表5：用岭回归和用牛顿法进行logistic回归对6个数据集进行5-way 1-shot和5-way 5-shot分类的结果。

A．8元训练中的精细字嵌入

在表1中，我们修复了所有实验的元训练期间的单词嵌入。表6研究了在元训练中精调单词嵌入的效果。我们观察到，当单词embeddings被微调时，几乎所有模型的性能都会下降。

为了直观地理解这种行为，我们比较了元训练词汇和元测试词汇。在Amazon上，元测试词汇表中28591个令牌中有15095个（52.8%）不存在，元测试词汇表中34545个令牌中有21049个（60.9%）不存在。路透社报道，6372个元测试令牌中有3604个（56.6%）不在元测试中，5249个元测试令牌中有2481个（47.2%）不在元测试中。由于这种词法不匹配，精细调整会破坏预训练单词嵌入的原始几何结构，导致对未观察任务的泛化能力差。

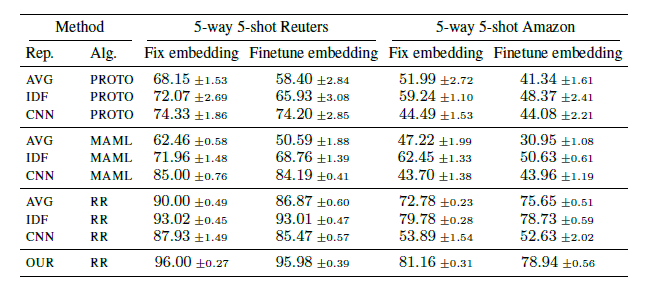


表6：元训练期间固定和精细的单词嵌入。由于元训练和元测试类表现出不同的单词分布，固定单词嵌入可以获得更好的泛化效果。也就是说，我们避免了对元串词汇的过度拟合和破坏预训练嵌入的几何结构。

A．9与其他分类器的分配签名

本文证明了在分布式签名的基础上学习元知识的优势。由于脊回归器的简单性和有效性，我们将其作为下游分类器。然而，使用分布特征进行学习的想法并不局限于岭回归；相反，我们可以将分布特征与其他学习方法相结合，例如原型网络（Snell等人，2017）和归纳网络（Geng等人，2019）。

**原型网络** 为了利用从分布式签名中学习到的特征来扩充原型网络，我们可以基于注意加权表示（x）构造每类原型（第4.2节）。从表7中，我们可以看到，与每个数据集的最佳原型网络基线相比，使用分布特征的学习提高了5路1-shot准确率9:9%，5路5-shot准确率16:7%。

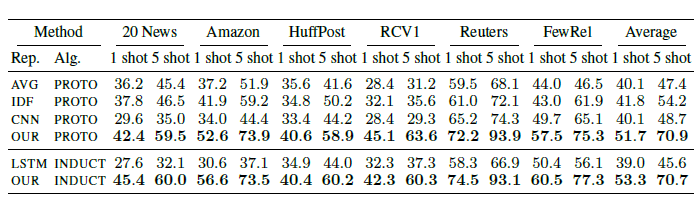


表7：典型网络和归纳网络在词汇信息和分布特征（OUR）上的表现。带有电感的LSTM是最初的电感网络结构。

**归纳网络** 我们也可以用从分布特征中学习的元知识来扩充归纳网络。具体来说，我们直接将注意力加权表示法（x）（第4.2节）提供给导入模块。为了避免元序列特征的过度拟合，我们替换了元学习关系模块（参见Geng等人。（2019）关于细节）通过无参数最近邻预测，类似于典型网络。从表7中，我们再次看到，使用分布特征的学习平均将5路单发准确率提高了14:3%，5路单发准确率提高了25:1%。

这些实证结果清楚地证明了我们的方法的好处。然而，无论是典型网络还是感应网络在所有实验方面的表现都始终不如岭回归器。我们的假设是岭回归器可以基于支持集进行特定任务的精细调整，而原型网络和归纳网络直接从元学习度量空间计算最近邻。

A．10小样本学习V.S.监督学习

为了了解少量镜头学习的效用，我们将我们的结果与完全监督的CNN分类器的结果进行比较，因为我们增加了训练点数。从图15中，我们可以看到我们的模型的5次射击精度大约相当于在50个例子上训练的完全监督分类器。

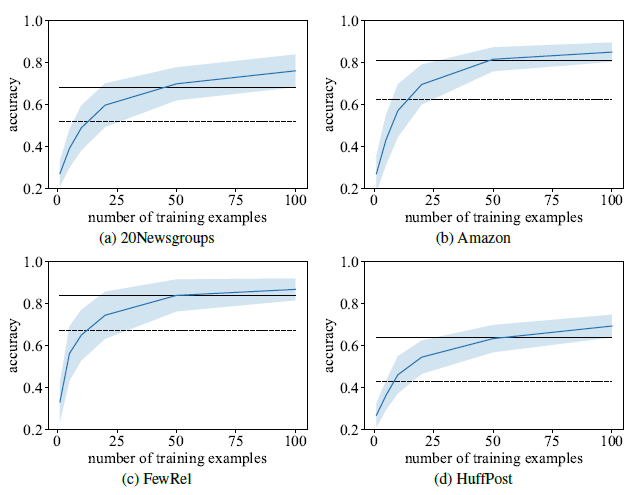


图15：完全监督CNN分类器与我们的分类器的学习曲线。蓝色表示CNN准确度，标准差阴影。实心水平线是我们的五杆精度；虚线是一杆精度。我们的5杆表现很有竞争力，特别是当标记的例子总数很少时。

A.11实施细节

我们在这里详细说明了基线的实现。所有基线代码都在我们的存储库中可用。

CNN 对于一维卷积，我们使用3，4，5的滤波窗口，每个窗口有50个特征图。随着时间的推移，我们在max之后应用了ReLU。

**原型网络** 网络元学习一个多层感知器，将输入表示转化为一个适合于少量镜头分类的嵌入空间。如果输入表示是可学习的（例如CNN），则输入表示的参数也使用元训练进行更新。在实验中，我们使用一个具有一个隐藏层和ReLU激活的MLP。隐藏层和输出层的尺寸都是300。我们将0.1的辍学率应用于隐藏层。

**MAML** MAML-meta学习初始化，以便模型在经过几个渐变步骤后能够快速适应新任务。对于输入表示的预测，我们使用一个隐藏层为300 ReLU单位的MLP。我们将0.1的辍学率应用于隐藏层。在MAML内循环（自适应阶段）中，我们执行了10次步长为0.1的更新（我们根据经验发现，这比一步MAML的性能更好）。我们在元训练的基础上反向传播高阶梯度。在MAML outerloop过程中，我们平均了10个采样任务的梯度，并使用Adam和学习率103来更新参数。

**Finetune** Chen等人。（2019）最近的研究表明，与最新的元学习相比，微调一个经过适当预训练的分类器可以获得具有竞争力的性能。在他们的工作之后，我们明确地减少了训练前阶段的班内变化。与MAML类似，我们使用一个具有一个隐藏层（300 ReLU单位）的MLP从输入表示（例如CNN）进行预测。在微调阶段，我们从头开始重新训练MLP，并微调输入表示的可学习参数。当梯度范数小于时，停止微调。

**归纳网络** 归纳网络由三个模块组成：编码器模块、感应模块和关系模块。编码器模块使用具有自关注的biLSTM来获得每个输入示例的固定长度表示。归纳模块执行动态路由以计算特定于类的原型。关系模块使用神经张量层来预测每个查询示例和原型之间的关系关联。遵循耿等人的观点。（2019），我们将biLSTM的隐藏状态大小设置为256（每个方向128个），注意维度设置为64。动态路由中的迭代次数设置为3。神经张量层的维数设置为100。

**P-MAML** P-MAML（Zhang et al.，2019）分为两个阶段：蒙面语言模型预训练（Devlin et al.，2018）和MAML（Finn et al.，2017）。对于预训练，我们使用了带有默认超参数的Hugging Face语言模型微调代码和BERT的预训练基无基础模型。当验证困惑度在两个时期内没有降低时，我们应用了提前停止。在预训练之后，我们在[CLS]标记的表示（Devlin et al.，2018）之上添加一个softmax层来进行预测。在MAML内环（自适应阶段）期间，我们执行了10次步长为10∏3的更新。跟随Zhang等人。（2019），我们不考虑高阶梯度。在主外循环过程中，我们对10个采样任务的梯度进行平均，并使用学习率为10°-5的Adam更新参数。

A、12结果

本节包含标准偏差的实验结果。

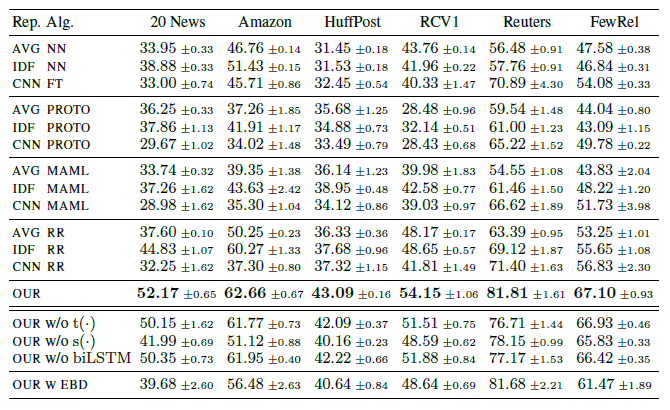


表8：5-way-1-shot分类。下面四行是我们的消融研究。

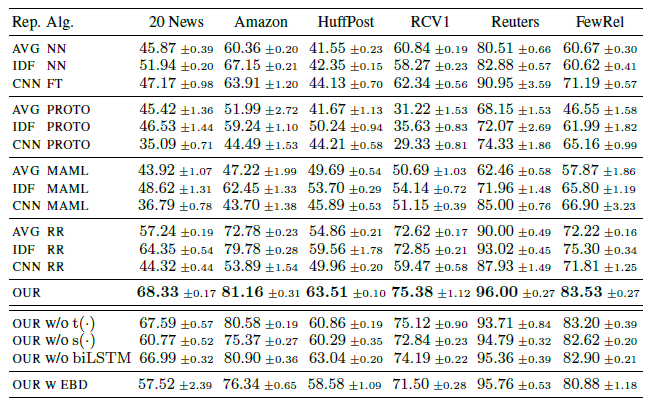
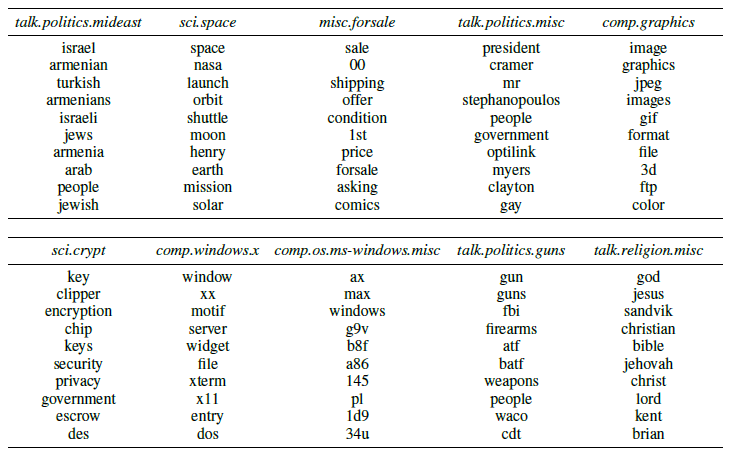


表9：5-way-5-shot分类。下面四行是我们的消融研究。



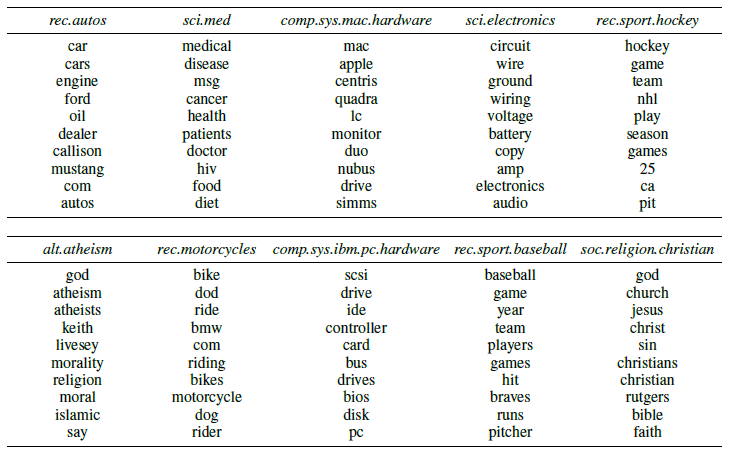


表10:20 Newsgroup.中，LMI排名前10位的单词。类名以斜体显示。不同的阶级表现出不同的显著特征。