**论文：*Multi-Level Matching and Aggregation Network for Few-Shot Relation Classification***

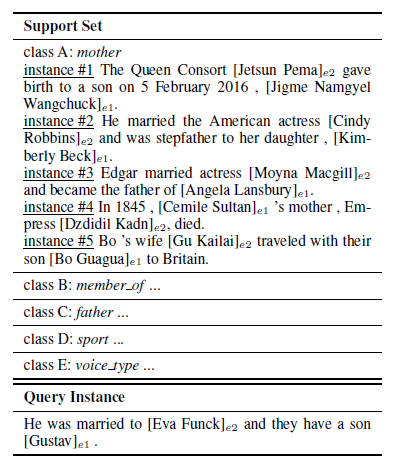
**会议**：ACL2019

**代码**：https://github.com/ZhixiuYe/MLMAN

本文提出了一种用于小样本关系分类的多层匹配和聚合网络（MLMAN）。 以前对此主题的研究都采用了原型网络，该网络分别计算查询实例的嵌入向量和每个支持集的原型向量。本文提出的MLMAN模型通过考虑局部（即单词级和短语级）和实例级别的匹配信息以交互方式对查询实例和每个支持集进行编码。 每个支持集的最终类原型是通过对其支持实例的表示进行注意汇总而获得的，其中权重是使用查询实例计算的。

**数据集**：

FewRel开发集中5-way-5-shot关系分类的一个数据实例。查询实例的正确关系类是类A:mother。为了节省空间，省略了其他关系类的实例。



问题定义：

在原型网络中，查询和支持实例独立地编码到嵌入空间中。然后，将每个候选类的一个原型向量作为其在嵌入空间中的支持实例的平均值。最后，通过计算查询的嵌入向量与所有类原型之间的距离来进行分类。

在小样本关系分类中，给出了两个数据集：和Dmeta-test。每个数据集由一组样本（x；p；r）组成，其中x是由T个词组成的句子，第t个词是，p=（p1；p2）表示两个实体的位置，r是实例（x；p）的关系标签。这两个数据集都有各自不相交的关系标签空间。在小样本配置下，将Dmeta-test分为Dtest-support和Dtest-query两部分。如果Dtest-support包含N个关系类中每一个的K个标记样本，则该目标小样本问题称为N-way-K-shot。Dtest-query包含测试样本，每个样本都标记有N个类中的一个。假设我们只需要Dtest-support和Dtest-query，我们就可以使用Dtest-support训练一个模型，并在Dtest-query上评估其性能。但由于支撑样本的数量有限（即N K），很难从零开始训练一个好的模型。

虽然Dmeta-train和Dmeta-test具有不相交的关系标号空间，但Dmeta-train也可用于帮助Dmeta-test中的小样本关系分类。一种方法是Vinyals等人提出的范式。（2016），它遵循一个重要的机器学习原则，即测试和训练条件必须匹配。也就是说，我们还将Dmeta-train分解为Dtrain-support和Dtrain-query两部分，并在训练阶段模拟了小样本习设置。在每一次训练迭代中，从Dtrain-support中随机选择N个类，从每个类中随机选择K个支持实例。这样，我们就构造了train-support集，其中是类i中的第k个实例。此外，我们从这些N个类的剩余样本中随机选择R个样本，并构造了train-query集，其中是实例的标签。

就像传统的原型网络一样，我们期望在训练时最小化以下目标函数，

函数用于计算查询实例q和支持实例集间的匹配率。如何设计该方程是本文的重点。

**原理：**

与原型网络不依赖于查询实例来表示支持集不同，我们提出的MLMAN模型通过在局部（即单词级和短语级）和实例两级考虑支持集的匹配信息，以交互的方式对每个查询实例和支持集进行编码。

在局部（即单词级和短语级）级别，查询实例和支持集的局部上下文表示按照句子匹配框架彼此软匹配。然后，使用max和average池将匹配的局部表示聚合为每个查询和每个支持实例的嵌入向量。

在实例级，通过多层感知器（MLP）计算查询实例与每个支持实例的匹配度。以匹配度为权值，对支持集中的实例进行聚合，形成最终分类的类原型。

MLMAN模型中的所有匹配层和聚集层都是利用训练数据联合估计的。由于每个类中的支持实例的表示预计彼此接近，因此进一步设计了一个辅助损失函数来度量每个类中所有支持表示之间的不一致性。我们的任务和上面提到的其他句子匹配任务的区别在于，我们的目标是将一个句子匹配到一组句子，而不是匹配到另一个句子（Bowman等人，2015）或一系列句子（Lowe等人，2015）。

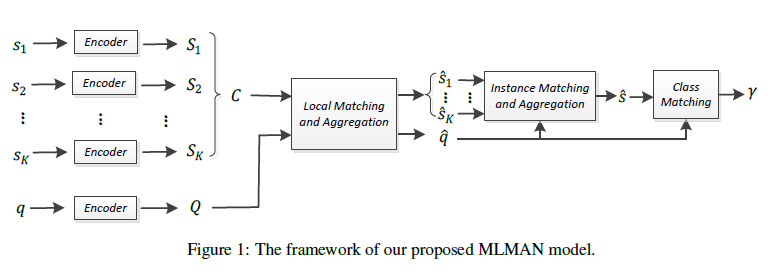
**贡献**

首先，提出了一个多级匹配和聚合网络，以交互的方式对查询实例和类原型进行编码。

其次，设计了一个辅助损失函数来度量支持实例之间的一致性。

第三，我们的方法在FewRel上实现了一个新的性能状态

**模型框架**



**上下文编码器。**

考虑到一个句子和句子中两个实体的位置，采用CNNs推导句子中每个单词的局部上下文表示。

对于查询或支持实例，首先将句子x中的每个单词映射到维单词嵌入。为了描述这种情况下两个实体的位置信息，提出了位置特征（PFs）。PFs描述当前单词和两个实体之间的相对距离，并进一步映射到dp维的两个向量和。最后，将这三个向量连接起来，得到dw+2dp维的单词表示，实例可以写成.

本文采用CNNs构建上下文编码器。对于一个输入实例，我们将其输入到带有滤波器的CNN中。CNN的输出是一个具有维数的矩阵。这样，就得到了查询实例的上下文表示和支持实例的上下文表示，其中和分别是查询语句和第k个支持语句的语句长度。

**局部匹配和聚合。**

考虑到查询实例的局部表示和K个支持实例的局部表示，使用注意方法收集它们之间的局部匹配信息。然后，对匹配的局部表示进行聚合，将每个实例表示为嵌入向量。

为了获得Q 和之间的匹配信息，我们首先将k个支持实例表示连接到一个矩阵中，如下所示其中,其中。

然后，我们收集Q和C之间的匹配信息，并计算它们的匹配表示如下

其中在等式（5），在等式（6）。和分别是Q和的第m行，和分别是 C和的第n行。

接下来，利用ReLU层将原始表示和匹配表示融合如下，

式中为各元素的乘积，为该层上用于降维的权重矩阵。 进一步分成K个表示，对应于K个支持实例，其中。所有的k和Q都被输入到一个具有dh隐藏单元的单层双向LSTM（BLSTM）中，沿着每个方向获得最终的局部匹配结果。并且

局部聚合的目的是将局部匹配的结果转换为每个查询和每个支持实例的单个向量。在本文中，我们将最大池和平均池结合起来，并将它们的结果连接到一个向量或中。

，

其中

**实例匹配和聚合。**

使用MLP计算查询实例和每个K个支持实例之间的匹配信息。然后，以匹配度作为权重，对支持实例的表示进行求和，得到类原型。

我们不使用简单的平均值操作，而是通过attention来聚合实例级表示，其中每个权重都来自实例和之间的匹配分数。匹配函数如下：

其中且,描述查询实例q与支持实例的实例级匹配度。

然后，所有聚合为一个向量，

是类原型

**类匹配**

MLP用于计算查询实例的表示与类原型之间的匹配分数。在确定类原型和查询实例的嵌入向量之后，等式（2）中的类级匹配函数定义为

式（11）和（13）的形式相同。在我们的实验中，共享这两个方程中的权重W2和v，即在每次训练迭代中对实例级和类级匹配使用完全相同的函数，可以获得更好的性能。

**不一致测量联合训练。**

如果类中所有支持实例的表示彼此相距甚远，则派生类原型将很难捕获所有支持实例的公共特征。

因此，设计了一个度量支持实例集之间不一致性的函数。为了避免直接比较类中每两个支持实例的复杂性，我们将支持实例与类原型之间的平均欧氏距离计算为不一致度量

其中i是类索引，计算向量的2-范数。

通过组合等式（1） 以及（14），训练整个模型的最终目标函数定义为，

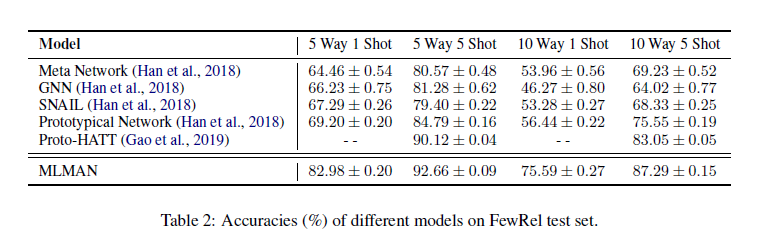
其中是一个超参数，在我们的实验中设置为1，没有任何调整。

**总结**

本文提出了一种基于多级匹配和聚合的神经网络用于小样本关系分类。首先，查询和支持实例通过本地匹配和聚合进行交互编码。然后，进一步聚合类中的支持实例，形成类原型，并通过基于注意的实例匹配计算权重。最后，利用一个可学习的MLP匹配函数计算查询实例与每个候选类之间的类匹配得分。此外，还设计了一个额外的目标函数，以提高类中所有支持实例的向量表示之间的一致性。

5.3与以往工作的比较

表2显示了在FewRel测试集上测试的不同模型的结果。前四个模型的结果，元网络（Munkhdalai和Yu，2017）、GNN（Garcia和Bruna，2017）、SNAIL（Mishra等人，2018）、Prorotypic网络（Snell等人，2017）由Han等人报告。（2018年）。



这些模型最初是为图像分类而提出的。Han等人。（2018）刚刚将他们的图像编码模块替换为一个实例编码模块，并保持其他模块不变。ProtoHATT（Gao等人，2019）在原型网络中加入了混合注意机制，主要集中在提高N>1的少镜头关系分类性能上。从表2可以看出，我们提出的MLMAN模型在很大程度上优于所有其他模型，这表明考虑查询实例和支持集在多个层次上的交互。

5.4消融研究

为了评估单个模型组分的贡献，进行了消融研究。表4显示了我们的模型的性能及其在FewRel开发集上的烧蚀。考虑到前6次烧蚀只影响N>1的少量射击学习任务，模型2至模型7在5向1射击和10向1射击配置下获得了与完整模型（即模型1）完全相同的性能。

5.4.1实例匹配和聚合

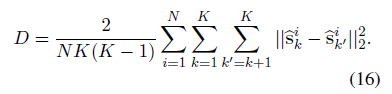
首先，第4.3节中介绍的基于注意力的实例聚合被替换为最大池（模型4）或平均池（模型5）。我们可以看到，具有实例级关注聚合（模型1）的模型在5次任务上的性能优于使用最大池（模型4）或平均池（模型5）的模型。t检验在1%显著性水平上差异显著。注意池的优点是可以根据查询动态地确定集成所有支持实例的权重。例如，当查询实例与表1中的支持集进行实例匹配和聚合时，类A中的5个实例的权重分别为0.03、0.46、0.25、0.08和0.18。实例#2的权重最高，因为它与查询实例的相似度最高，在将查询实例与类A匹配时被认为是最有帮助的实例。

然后，讨论了在等式中共享权重参数的有效性。（11） （13）通过解绑进行评估（模型3）。模型3的性能比表4所示的完整模型（模型1）差得多，这表明需要在实例和类级别上共享计算匹配分数的权重。

5.4.2不一致性度量

第4.5节介绍的，Jincon旨在度量类中所有支持实例的表示之间的不一致性。去除Jincon后，仅使用目标函数Jmatch对模型2进行优化。我们可以看到它的性能比完整的模型差得多。

此外，我们分别使用模型1和模型2计算了同一类中每个支持实例对（bs i k；bs i k0）之间的欧氏距离的平均值。对于每个支撑集，计算可以写成



我们在5路5炮配置下抽样了2万个保障组，并计算了它们的平均值。模型1和模型2的结果分别为0:0199和0:0346，这意味着Jincon可以有效地迫使同一类中支持实例的表示彼此接近。

从模型5中进一步去除Jincon，得到模型6。结果表明，从模型5到模型6的精度下降幅度大于从模型1到模型2的精度下降幅度。这意味着Jincon目标函数还受益于对支持实例的集中关注。

5.4.3局部匹配

首先，在本消融研究中，从模型6中移除了局部匹配中的级联操作。也就是说，不是将所有支持实例的表示形式fSkgKk=1作为等式（3）连接到一个矩阵中，而是在查询实例和每个支持实例之间分别进行本地匹配，以获得它们的向量表示形式f（bs k；bq k）；k=1；：：；Kg（模型7）。应该注意的是，这导致了根据每个支持类，查询实例的K种不同表示。然后，计算bs k和bq k的平均值overk，得到支持集和查询实例bq的表示。通过比较模型6和模型7，我们可以看出级联操作在我们的模型中起着重要的作用。一个可能的原因是，连接操作可以帮助本地匹配来抑制与查询相似性较低的支持实例。

其次，从模型6中删除了整个局部匹配模块以及连接和注意聚合操作，从而得到模型9。模型9与Snell等人提出的模型相似。（2017）独立编码支持和查询实例。不同之处在于，模型9配备了更多的组件，包括一个LSTM层、两个池操作和一个可学习的类匹配函数。通过比较表4中模型6和模型9的性能，我们可以看出，局部匹配操作显著提高了在少镜头关系分类中的性能。图。2显示了表1中在查询实例和类A的支持实例#2之间计算的注意权重矩阵。从这个图中我们可以看到，基于注意力的局部匹配能够捕捉到一些局部上下文的匹配关系，如头部实体Eva Funck和Cindy Robbins，尾部实体Gustav和Kimberly Beck，关键短语son和butter，相同关键字“married”等等。

在本实验中，我们比较了两类匹配函数：（1）欧氏距离（ED）（Snell等人，2017）和（2）可学习的MLP函数，如等式（13）所示。为了忽略实例级注意聚合的影响，在模型6和模型9的基础上对这两种匹配函数进行了比较。将模型6和模型9中的MLP函数转换为欧氏距离，得到模型8和模型10。比较表4中这些模型的性能，我们有两个发现。（1） 当采用局部匹配时，用于类匹配的可学习MLP（模型6）的性能大大优于ED度量（模型8）。（2） 移除本地匹配后，类匹配的可学习MLP（模型9）的性能不如ED度量（模型10）。一个可能的原因是，当计算bs和bq时，本地匹配过程增强了查询实例和支持集之间的交互。因此，它们之间的简单欧氏距离可能无法描述它们之间的复杂相关性和依赖性。另一方面，MLP映射比计算欧几里德距离更为强大，并且在采用局部匹配的情况下更适合于类匹配。