P3

首先我们对这篇论文进行一个总体的概括。本篇论文基于三个事实对模型进行改进，最后通过实验得到了较好的结果。

首先，双向长短期记忆神经网络模型也就是BiLSTM已经在命名实体识别任务中得到了广泛的应用，上一位同学的网络模型就是基于BiLSTM的。

与此同时，Transformer具有更好的并行性，能更好地利用GPU，效率更高，它在其他自然语言处理任务中也应用广泛，比如机器翻译这些，但普通的transformer在命名实体识别的任务中表现又不太好。

这里我们就考虑，为什么普通的transformer在命名实体识别的任务里表现不好，BiLSTM相比较transformer的话，是什么特点让它效果更好？如何改进tranformer使得它更适用于命名实体识别的任务？

于是本篇论文对普通的transformer进行了改进，一方面，使用结合方向感知、距离感知并且不经缩放的注意力机制，另一方面，将tranformer作为字符编码器去模拟字符级别和词级别的特征，就是在词嵌入和字符嵌入的时候都使用我们改进的transformer。这里文章主要对前一方面进行讲解。

最终，通过在两个英文数据集和四个中文数据集上进行试验，实验结果证明，经过改良后的模型结构表现出了 比基于BiLSTM的模型 更好的效果。

P4

下面，我们对整体的网络结构进行一个概括的介绍，之后再详细介绍具体的内容。

我们首先整体看这个模型结构，自下而上，主要包括三个模块，嵌入模块，transformer模块以及条件随机场CRF模块。

嵌入模块这里，首先是字符嵌入，接着多头自注意力机制，然后向上是提取到的字符特征与预训练得到的词嵌入的串联。

之后是n个transformer编码器结构，这里的输入是embedding模块的输出，最上层为了利用不同标签之前的依赖关系，CRF处理得到输出。

我们主要的改进是针对transformer模块的注意力机制，下面将进行详细讲解。

P5

我们的第一个改进，也是要解决的问题就是距离感知和方向感知。

受 BiLSTM 在 NER 任务中取得成功的启发，我们考虑了 Transformer 与基于 BiLSTM 的模型相比缺少哪些属性。

一个观察结果是 BiLSTM 可以从其左侧和右侧有区别地收集token的上下文信息。但是 Transformer 很难区分上下文信息来自哪一边。

这是因为我们在transformer中加入了正弦位置嵌入，这个方法能够把字词的绝对位置信息输入到网络当中去。

在transformer的原理里面，我们会把两个正弦位置嵌入做一个点积，就是矩阵內积，这个点积能够反映它们的距离，但它缺乏方向性。

而对于NER任务而言，模型对距离和方向的感知同样重要。

我们举一个例子来说明这个问题。相对方向在 NER 任务中很重要，因为“Inc.”之前的单词很可能是一个组织，“in”后面的词更有可能是时间或地点。

此外，单词之间的距离也很重要，因为只有连续的单词才能形成一个实体，以前的“Louis Vuitton”不能与“Inc.”形成一个实体。

因此，距离感知能够帮助一个单词更好地识别出它的近邻。

为了赋予这个trransformer方向和距离敏感的能力，我们采取了一种相对位置编码来代替绝对位置编码。我们提出了一种改进的相对位置编码，它使用更少的参数并且表现更好。

P6

注意力机制不知道不同标记的位置，使其无法捕捉语言的顺序特征。 为了解决这个问题，使用由不同频率的正弦曲线生成的位置嵌入。

其中 i 在 [0; d /2 ]，d 是输入维度。这种基于正弦曲线的位置嵌入使 Transformer 能够对token的位置和每两个token的距离进行模拟。对于任何固定偏移量 k，PEt+k 可以用 PEt 的线性变换表示（Vaswani 等，2017）。

另外这个公式里i表示的是维度的索引，2i表示的是偶数位，2i+1是奇数位。

虽然两个正弦曲线之间的点积 位置嵌入能够反映他们的 距离，它缺乏方向性和这个属性 在普通transformer的注意力机制上失效。 为了说明这一点，我们首先证明正弦位置嵌入的两个性质 。

这两个性质的证明不再赘述。

第一个性质是说，对于偏移量 k 和位置 t， PE t+k PE t 只取决于 k，这意味着 两个正弦位置嵌入的点积 可以反映两个token之间的距离。 这个可以通过和差公式证明。

第二个性质是说，对于偏移量 k 和位置 t， PE t PE t-k = PE t PEt+k，这意味着正弦 位置嵌入不知道方向性。 这个特性可以通过换元法证明。

由此，我们可以绘制出d,k, 和这个內积之间的曲线

P7

PE t PE t+k 这里一个点表示距离为 k 的两个正弦位置位置嵌入的点积 。

D是维度，随着d的增大，可以看到距离感知是更加敏感的。

从这个图可以看到它是对称的，并且随着 k的绝对值的增大，它有减少的趋势，但这个减少并不单调。

这就说明了正弦位置嵌入 有距离意识但缺乏方向性。

然而，当 PE t 投射到 自注意力的K和Q空间时，距离感知的特性会消失 。 因为普通的Transformer 中 PE t PE t+k之间的计算 实际上是 PE t Wq Wk PE t+k，其中 Wq;Wk 是两个学习到的参数矩阵。

在数学上，它可以被视为 PEt W PEt+k，只有一个 参数 W。 PE t PE t+k 和 PE t W PE t+k 之间的关系如图 4 所示。

我们可以看到这种加了映射之后的点积既不能反映距离，也不能反映方向。

P8

为了解决这个问题，我们先看一下计算attention分数的公式。

1.

Transformer 编码器接收一个矩阵 H 大小是l\* d，其中 l 是序列长度，d 是输入维度。 然后使用三个可学习矩阵 Wq、Wk、Wv 将 H 投影到不同的空间中。 通常，三个矩阵的矩阵大小都是 Rd\* dk ，其中 dk 是一个超参数。 之后，缩放的点积注意力可以通过以下等式计算。

其中 Qt 是第 t 个标记的查询向量，j 是第 t 个token旁边的token。 Kj 是第 j 个token的Key向量表示。 softmax 沿着最后一个维度。

不使用一组Wq、Wk、Wv，使用几组会增强自我注意力的能力。 （多头注意力机制）

多头注意力的输出将由位置前馈网络进一步处理 。

2.

自从NER 数据集的大小通常很小，我们避免两个可学习参数的直接乘法，因为它们可以用一个可学习的来表示范围。 因此我们在方程（16）中不使用 Wk。

其中 t 是目标标记的索引，j 是上下文标记的索引。 Qt; Kj 是分别 token t，j的查询向量和关键向量；

Wq; Wv d \* dk 。 为了得到 Hdk 的大小是l \*dk ，我们首先在第二个维度中将 H 拆分为 d/dk 分区 ，然后对于每个head我们使用一个分区。

u, v 是可学习的参数， Rt-j 是相对位置编码， i 在[0; dk/ 2]。 Q t Kj 第一项是 两个token的attention分数； Q t Rt-j 第二项是第 t 个token在确定相对距离上的偏置； u Kj 第三项是第 j 个的token上的偏置； vTRt-j 第四项是确定距离和方向的偏置项 。

3.

我们把计算相对位置编码的这个式子单独拿出来，带入t和-t，就能够发现它们的不同。

这意味着对于偏移量 t，向前和向后 相对位置编码在cos(cit) 项上相同 ，但在sin(cit) 项上相反，因为sin是奇函数，cos是个偶函数。 所以， 通过使用 Rt-j ，attention score 可以区分 不同的方向和距离。

P9

第二处的改进时在注意力机制里去除缩放因子，回到这个注意力机制的公式，我们在计算attention分数的时候先计算Query矩阵和Key矩阵的內积，然后再将其除以一个平滑因子，再做softmax处理。

但经过实验发现，去除这个缩放因子，可能效果会更好，这是一个实验发现，没有经过证明的。

文章推测了一下，可能是因为没有比例因子，注意力会更敏锐。这种锐化后的注意力可能有利于 NER 任务，因为句子中只有少数单词是命名实体。

P10

对于字符编码器而言，提取字符级别的特征以及缓解未登录词oov的问题是很重要的，CNN就是一种常用的字符编码器，但它在表示字符级别信息的时候，效果并不完美。

一方面，它的感受野有限，经常是3\*3,5\*5的卷积核大小，这时候它对于2gram或者4gram模式的处理效果就没那么好。

当然也可以人为把卷积核设置为我们想要的大小，但也很难识别出不连续的模式。

关于这方面的内容，我们使用Transformer可能效果就会更好，因此在本篇论文里，我们使用改进的transformer作为字符编码器。

后面的实验中，论文也使用不同的encoder做出了对比。

P11

我们在两个英文 NER 数据集和四个中文 NER 数据集上评估我们的模型。

1. CoNLL2003 是评价最高的英语 NER 数据集之一，它包含四个不同的命名实体：人、位置、组织和其他（Sang 和 Meulder，2003 年）。
2. OntoNotes 5.0是英文NER数据集，其语料来自不同领域，如电话交谈、新闻专线等。我们排除了New Testaments部分，因为其中没有命名实体（Chen 等，2019；Chiu 和 Nichols，2016）。该数据集有 11 个实体名称和 7 个值类型，如 CARDINAL、MONEY、LOC。
3. Weischedel (2011) 发布 OntoNotes 4.0。在本文中，我们使用中文部分。我们采用了与 (Che et al., 2013) 相同的预处理。
4. 中文 NER 数据集 MSRA 的语料库来自新闻领域 (Levow, 2006)。
5. 微博NER是基于中国社交媒体新浪微博(Peng and Dredze, 2015)中的文本构建的，它包含4种实体。
6. Resume NER 由 (Zhang and Yang, 2018) 注释。

P12

对于所有数据集，我们将所有数字替换为“0”，并使用 BIOES 标记模式。

对于英语，我们使用 Glove 100d 预训练嵌入 (Pennington et al., 2014)。

对于字符编码器，我们使用 30d 随机初始化的字符嵌入。

对于中文，我们使用了 (Zhang and Yang, 2018) 发布的字符嵌入和二元嵌入。

所有预训练的嵌入都在训练期间进行了微调。

为了减少随机性的影响，我们将所有实验至少运行了 3 次，并报告了其平均 F1 分数和标准偏差。

我们使用随机搜索来找到最佳超参数，

我们使用 SGD 和 0.9 动量来优化模型。

我们运行了 100 个 epoch，每个批次有 16 个样本。

该模型实现了最高的开发性能，用于评估测试集。

P13

1.

中文 NER 数据集的 F1 分数。 |，是 (Zhang and Yang, 2018) 和 (Gui 等人，2019a)的结果，

“w/ scale”是指使用方程（19）中的缩放注意力的 TENER。 他们的结果不是 与我们的直接比较，因为他们使用了 100d 预训练字符和二元嵌入。 其他型号 使用相同的嵌入。

vanilla Transformer 表现不佳，并且比基于 BiLSTM 和 CNN 的模型差。

然而，当相对位置编码相结合时，性能得到了极大的提升，在所有数据集上都取得了比 BiLSTM 和 CNN 更好的结果。

微博数据集的训练样本数量很少，因此 Transformer 的性能非常糟糕，这符合预期，因为 Transformer 需要大量数据。

尽管如此，当使用相对位置编码和未缩放的注意力进行增强时，它可以获得比基于 BiLSTM 的模型更好的性能。

改良 Transformer 在从小数据集到大数据集的四个数据集中的卓越性能表明适配 Transformer 对训练示例的数量比 vanilla Transformer 更稳健。正如表 2 的最后一行所描述的，缩放注意力会降低性能。

2.

英语 NER 数据集的 F1 分数。

我们 仅列出基于非上下文化嵌入的结果， 和使用预训练语言模型的方法， 预训练的特征，或更高维的词向量 被排除在外。

TENER (Ours) 使用 Transformer 字符级和字级的编码器。 “w/ scale”表示使用缩放注意力的 TENER 在等式（19）中。 “w/ CNN-char” 表示 TENER 使用 CNN 作为字符编码器而不是 AdaTrans。

不同NER模型在英语NER数据集上的比较如表3所示。(Guo et al., 2019) 也报告了 Transformer 在 NER 数据集中的糟糕表现。

尽管 Transformer 的性能高于（Guo 等人，2019），但仍落后于基于 BiLSTM 的模型（Ma 和 Hovy，2016 年）。

尽管如此，通过将相对位置编码和未缩放的注意力整合到 Transformer 中，性能得到了极大的提高。改良不仅使 Transformer 实现了优于 BiLModels STM 模型的性能，

当仅使用 Glove 100d 嵌入和 CNN 字符嵌入时，还会在两个 NER 数据集中展示新的最先进性能。使用ElMo作为英文数据集嵌入层处理时效果更好。

使用缩放注意力时，也观察到了同样的性能下降。

P14

1.

在本节中，我们跨越不同的字符级编码器（BiLSTM、CNN、Transformer 编码器和我们适配的 Transformer 编码器（简称 AdaTrans））和不同的词级编码器（BiLSTM、ID-CNN 和 AdaTrans）来实现 NER 任务。

CoNLL2003 和 OntoNotes 5.0 的结果分别列于表 5a 和表 5b 中。

ID-CNN 编码器来自 (Strubell et al., 2017)，我们在 Py-Torch 中重新实现了他们的模型。

对于不同的组合，我们使用随机搜索来找到其最佳超参数。字符编码器的超参数是固定的。

对于表 5a 中描述的 CoNLL2003 数据集的结果，AdaTrans 在不同字符编码器场景中，从表 5b 中，我们可以发现当使用不同的词级编码器时，AdaTrans 字符编码器超过 BiLSTM 和 CNN 字符编码器的模式。

此外，无论使用何种字符编码器或不使用何种字符编码器，AdaTrans 字级编码器都能获得最佳性能。

这意味着当训的平均表现与 BiLSTM 一样好。此外练样例数量增加时，AdaTrans 字符级和词级编码器可以更好地发挥其能力。

2.

我们比较了 OntoNotes 5.0 的开发集中 BiLSTM、ID-CNN、Transformer 和 TENER 的收敛速度。曲线如图 5 所示。TENER 的收敛速度与 BiLSTM 模型一样快，并且优于 vanilla Transformer。

P15

P16

在本文中，我们提出了 TENER，这是一种采用 Transformer Encoder 的模型，具有针对 NER 任务的特定改进。

Transformer Encoder 具有强大的捕获长范围上下文的能力。

为了使 Transformer 更适合 NER 任务，我们引入了方向感知、距离感知和非缩放注意力。

在两个英文 NER 任务和四个中文 NER 任务中的实验表明，性能可以大幅提升。

在相同的预训练嵌入和外部知识下，我们提出的修改在六个数据集中优于以前的模型。

同时，我们还发现适配后的 Transformer 适合用作英文字符编码器，因为它具有从字符中提取复杂模式的潜力。

在两个英文 NER 数据集中的实验表明，改进后的 Transformer 字符编码器的性能优于 BiLSTM 和 CNN 字符编码器。