【NLP】6种用于文本分类的开源预训练模型

机器学习初学者 7月4日

以下文章来源于磐创AI ，作者VK

磐创AI

AI行业最新动态，机器学习干货文章，深度学习原创博客，深度学习实战项目，Tenso...

来源 | PURVA HUILGOL

编译 | VK

作者 | Analytics Vidhya

【导读】我们正站在语言和机器的交汇处，这个话题我很感兴趣。机器能像莎士比亚一样写作吗？机器能提高我自己的写作能力吗？机器人能解释一句讽刺的话吗？我肯定你以前问过这些问题。自然语言处理（NLP）也致力于回答这些问题，我必须说，在这个领域已经进行了突破性的研究，促使弥合人类和机器之间的鸿沟。

介绍

文本分类是自然语言处理的核心思想之一。如果一台机器能够区分名词和动词，或者它能够在客户的评论中检测到客户对产品的满意程度，我们可以将这种理解用于其他高级NLP任务。

这就是我们在文本分类方面看到很多研究的本质。迁移学习的出现可能促进加速研究。我们现在可以使用构建在一个巨大的数据集上的预训练的模型，并进行优化，以在另一个数据集上实现其他任务。

迁移学习和预训练模型有两大优势：

1. 它降低了每次训练一个新的深度学习模型的成本
2. 这些数据集符合行业公认的标准，因此预训练模型已经在质量方面得到了审查

你可以理解为什么经过预训练的模特会大受欢迎。我们已经看到像谷歌的BERT和OpenAI的GPT-2这样的模型真的很厉害。在这里中，我将介绍6种最先进的文本分类预训练模型。

我们将介绍的预训练模型：

- XLNet

- ERNIE
- Text-to-Text Transfer Transformer（T5）
- BPT
- NABoE
- Rethinking Complex Neural Network Architectures for Document Classification

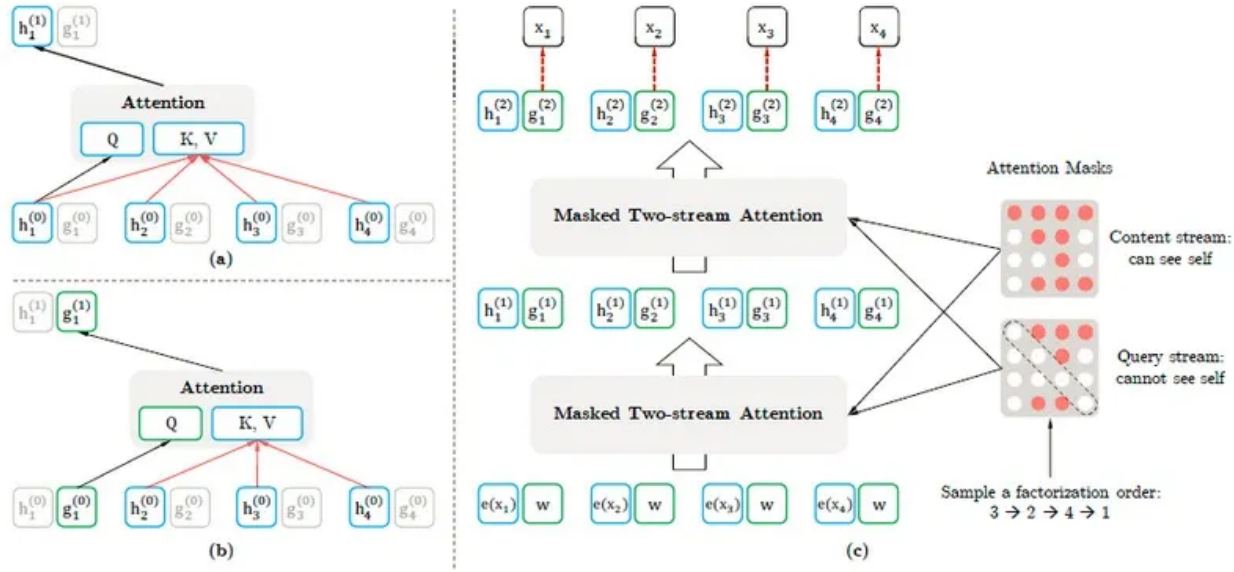
预训练模型1：XLNet

我们不能不提及XLNet!

谷歌的最新模型XLNet在文本分类，情感分析，问答，自然语言推理等主要NLP任务上取得了最先进的性能，同时还为英语提供了必要的GLUE基准。它的性能超过了BERT，现在已经巩固了自己作为模型的优势，既可以用于文本分类，又可以用作高级NLP任务。

XLNet背后的核心思想是：

1. 语言理解的广义自回归预训练
2. Transformer-XL



如果这听起来很复杂，别担心！我要把这些简化。

自回归模型用于预测下一个单词，使用的单词在已有的单词之前或之后出现。但是，不能同时处理前面和后面的单词，只能处理一个方向。

虽然BERT确实处理了这方面的问题，但它也有其他缺点，比如假设某些屏蔽词之间没有相关性。为了解决这个问题，XLNet在训练前阶段提出了一种称为排列语言模型(Permutation Language Modeling)的技术。这项技术使用排列同时从正向和反向生成信息。

Transformer已经不是什么秘密了。XLNet使用Transformer-XL。众所周知，在允许不相邻的标记也一起处理的意义上，Transformer是循环神经网络（RNN）的替代，因为它提高了对文本中远距离关系的理解。

Transformer-XL是BERT中使用的Transformer的增强版本，因为添加了这两个组件，：

1. 句段层级的循环
2. 相对位置编码方案

正如我前面提到的，XLNet在几乎所有任务上都超越BERT，包括文本分类，并且在其中18个任务上实现了SOTA性能！

以下是文本分类任务的摘要，以及XLNet如何在这些不同的数据集上执行，以及它在这些数据集上实现的高排名：

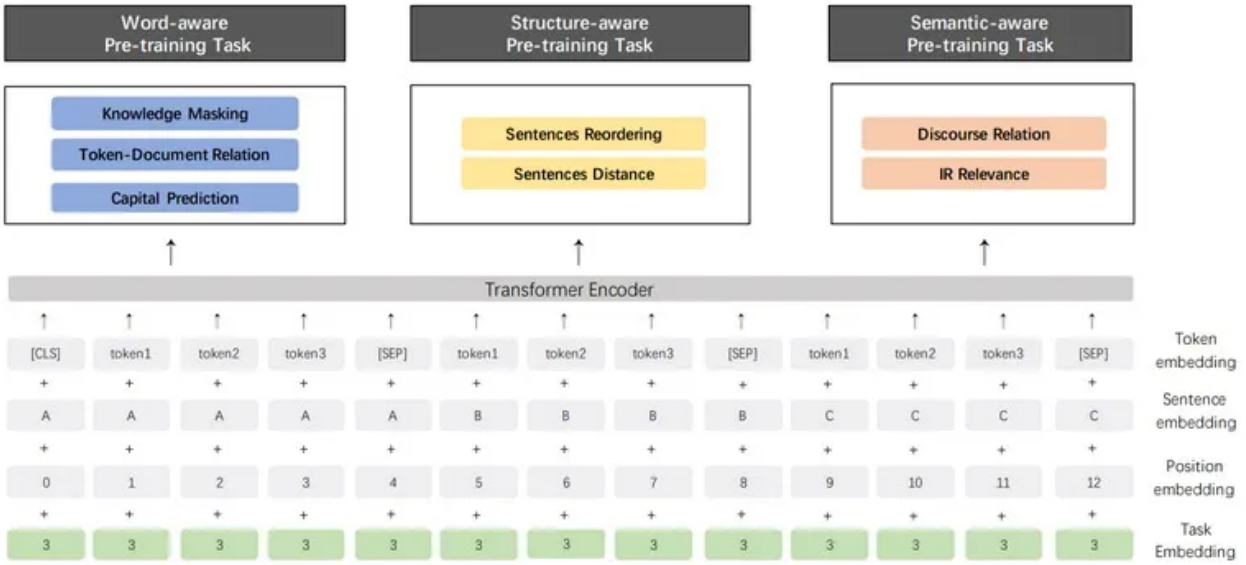
Task	Dataset	Metric	Metric Value	Global Rank
Text Classification	AG News	Error	4.49	1
Text Classification	Amazon-2	Error	2.40	1
Text Classification	Amazon-5	Error	32.26	1
Text Classification	DBPedia	Error	0.62	1
Text Classification	IMDb	Accuracy	96.21	1
Text Classification	Yelp-2	Accuracy	98.45	1
Text Classification	Yelp-5	Accuracy	73.20	1
Sentiment Analysis	SST-2 Binary Classification	Accuracy	96.8	2

预训练模型2：ERNIE

尽管ERNIE 1.0（于2019年3月发布）一直是文本分类的流行模式，但在2019年下半年，ERNIE 2.0成为热门话题。由科技巨头百度（Baidu）开发的ERNIE在英语GLUE基

准上的表现超过了Google XLNet和BERT。

ERNIE 1.0以自己的方式开辟了道路——它是最早利用知识图的模型之一。这一合并进一步加强了对高级任务（如关系分类和名称识别）模型的训练。



与它的前身一样，ERNIE 2.0以连续增量多任务学习的形式带来了另一项创新。基本上，这意味着模型定义了7个明确的任务，并且

- 可以同时生成多个任务的输出。例如，完成“I like going to New ...”->“I like going to New York”这句话，并将这句话归类为有积极的情绪。对于合并的任务，也相应地计算损失
- 将上一个任务的输出增量地用于下一个任务。例如，任务1的输出用作任务1、任务2的训练；任务1和任务2的输出用于训练任务1、2和3等等

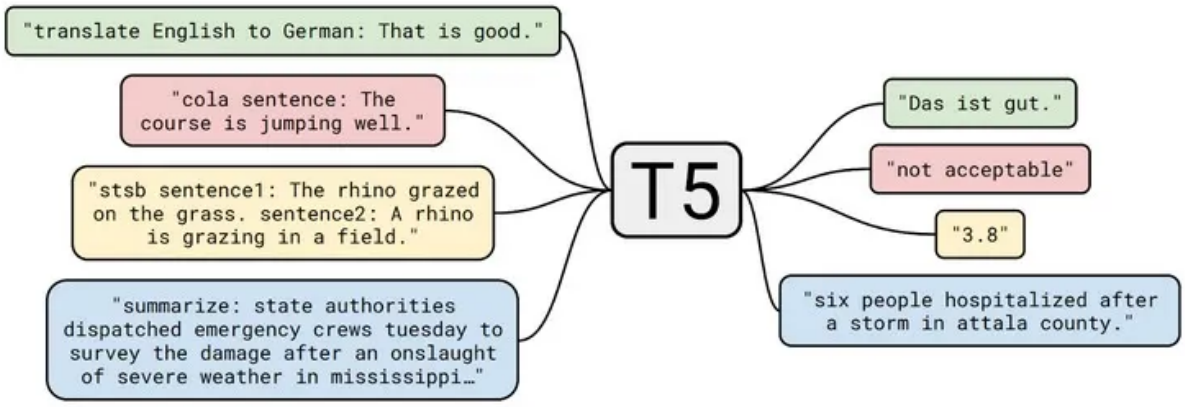
我真的很喜欢这个过程，他非常直观，因为它遵循人类理解文本的方式。我们的大脑不仅认为“I like going to New York"是一个积极的句子，它也同时理解名词“New York”和“I”，理解动词“like”，并推断纽约是一个地方。

ERNIE在关系抽取任务中的 F1度量为88.32。

预训练模型3：Text-to-Text Transfer Transformer（T5）

老实说，与其他模型相比，我在学习这个模型上获得了最大的乐趣。Google的Text-to-Text Transfer Transformer（T5）模型将迁移学习用于各种NLP任务。

最有趣的部分是它将每个问题转换为文本输入—文本输出模型。所以，即使对于分类任务，输入是文本，输出也将是文本而不是一个标签。这可以归结为所有任务的单一模型。不仅如此，一个任务的输出可以用作下一个任务的输入。



该语料库使用了Common Crawls的增强版本。这基本上是从网上刮来的文字。本文实际上强调了清理数据的重要性，并清楚地说明了这是如何做到的。虽然收集到的数据每月产生20TB的数据，但这些数据中的大多数并不适合NLP任务。

即使只保留文本内容（包含标记、代码内容等的页面已被删除），该语料库的大小仍高达750GB，远远大于大多数数据集。

注意：这已经在TensorFlow上发布了：https://www.tensorflow.org/datasets/catalog/c4。

将要执行的任务与输入一起编码为前缀。如上图所示，无论是分类任务还是回归任务，T5模型仍会生成新文本以获取输出。

T5在20多个已建立的NLP任务上实现了SOTA——这是很少见的，而且从度量标准来看，它尽可能接近人类的输出。

Task	Dataset	Metric	Metric Value	Global Rank
Question Answering	BoolQ	Accuracy	91.0	1
Document Summarization	CNN/Daily Mail	ROUGE-2	21.55	1
Linguistic Acceptability	CoLA	Accuracy	70.8	1
Semantic Textual Similarity	MRPC	F1	92.4	2
Sentiment Analysis	SST-2 Binary Classification	Accuracy	97.4	1

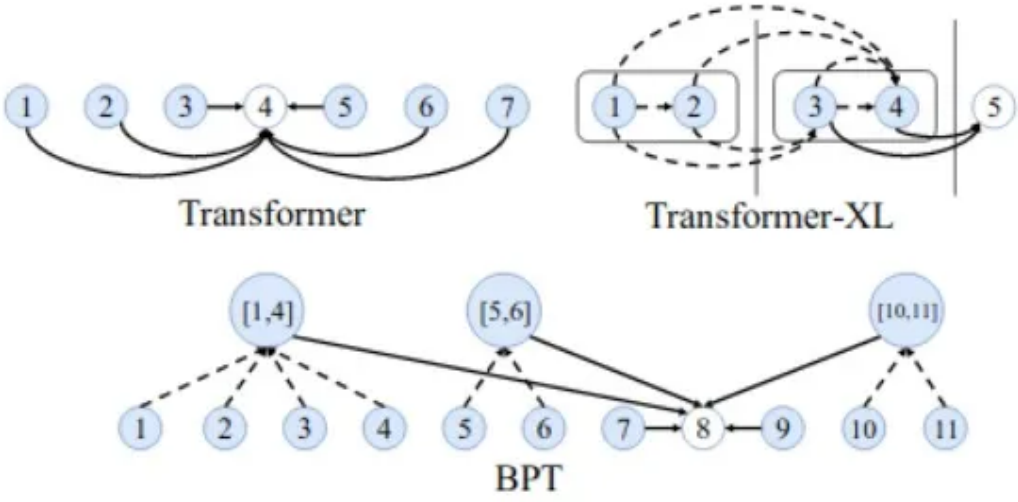
T5模型跟踪了最近关于未标记数据的训练趋势，然后在标记文本上微调该模型。可以理解的是，这个模型是巨大的，但是我们很有兴趣看到进一步研究如何缩小这种模型的规模，以获得更广泛的使

用和分布。

预训练模型4：BPT

正如我们目前所看到的，Transformer架构在NLP研究中非常流行。BP Transformer再次使用了Transformer，或者更确切地说是它的一个增强版本，用于文本分类、机器翻译等。

然而，使用Transformer仍然是一个昂贵的过程，因为它使用自我注意机制。自我注意只是指我们对句子本身进行注意操作，而不是两个不同的句子。自我注意有助于识别句子中单词之间的关系。正是这种自我关注机制导致了使用Transformer的成本。



Binary-Partitioning Transformer (BPT)将Transformer看作一个图形神经网络，旨在提高自注意力机制的效率。实际上，此图中的每个节点都表示一个输入标记。

BP Transformer的工作原理：

第一步：递归地把句子分成两部分，直到达到某个停止条件为止。这称为二元分区。因此，例如，“I like going to New York”这句话将有以下几个部分：

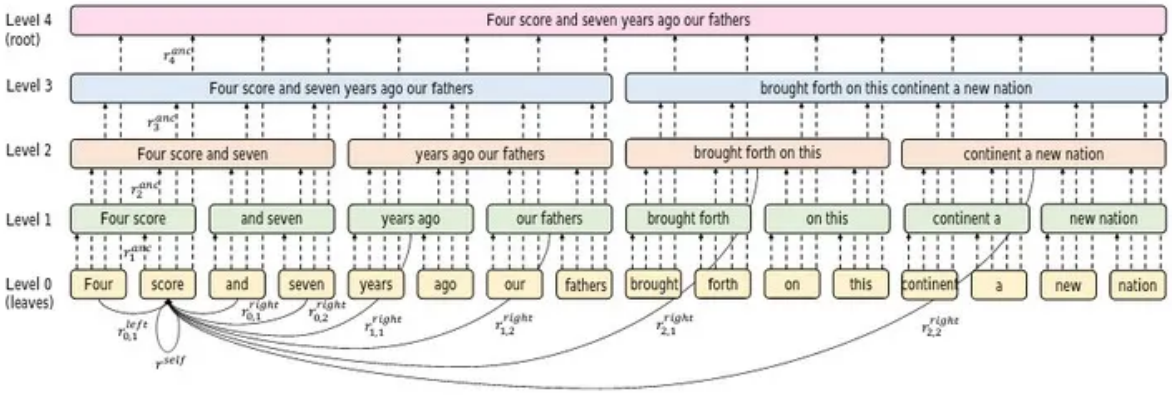
- I like going; to New York
- I like; going; to New; York
- I; like; going; to; New; York

注意：一个包含n个单词的句子将有2*n–1个分区，最后，你将得到一个完整的二叉树。

第二步：现在每个分区都是图神经网络中的一个节点。可以有两种类型的边：

- 连接父节点及其子节点的边
- 连接叶节点与其他节点的边

第三步：对图的每个节点及其相邻节点执行自注意：



BPT实现了：

- 在中英机器翻译上达到了SOTA的成绩(BLEU评分:19.84)
- IMDb数据集情绪分析的准确率为92.12(结合GloVE embedding)

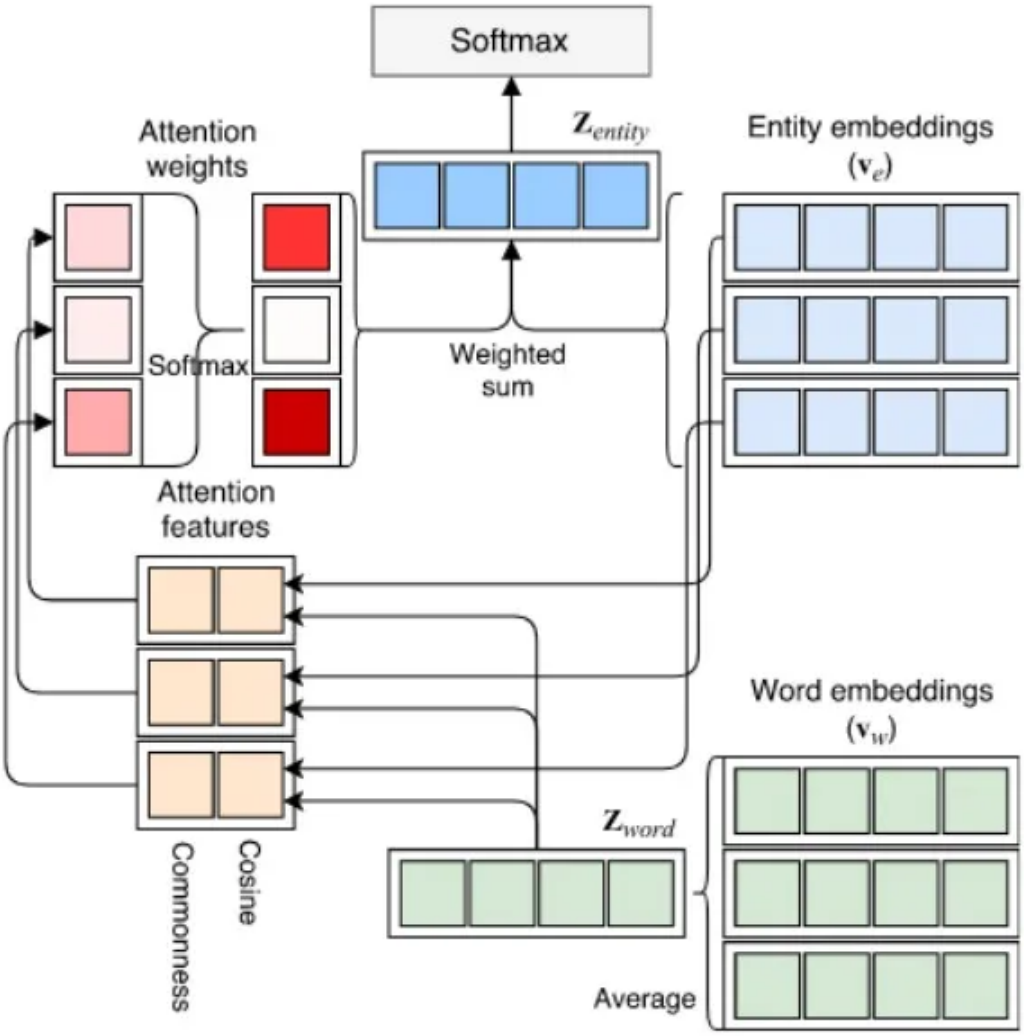
我很欣赏这个模型，因为它使我重新审视了图的概念，并使我敢于研究图神经网络。我肯定会在不久的将来探索更多的图形神经网络！

预训练模型 5：NABoE

神经网络一直是NLP任务最受欢迎的模型，并且其性能优于更传统的模型。此外，在从语料库建立知识库的同时用单词替换实体可以改善模型学习。

这意味着，我们不是使用语料库中的单词来构建词汇表，而是使用实体链接来构建大量实体。虽然已有研究将语料库表示为模型，但NABoE模型更进一步：

1. 使用神经网络检测实体
2. 使用注意力机制来计算被检测实体的权重(这决定了这些实体与文档的相关性)



实体模型的神经注意包使用Wikipedia语料库来检测与单词相关的实体。例如，单词“Apple”可以指水果、公司和其他可能的实体。检索所有这些实体后，使用基于softmax的注意力函数计算每个实体的权重。这提供了只与特定文档相关的实体的一个更小的子集。

最后，通过向量嵌入和与词相关的实体的向量嵌入，给出了词的最终表示。

NABoE模型在文本分类任务中表现得特别好:

Task	Dataset	Metric	Metric Value	Global Rank
Text Classification	20NEWS	Accuracy	88.1	2
Text Classification	R8(SOTA)	Accuracy	97.9	1

预训练模型6：Rethinking Complex Neural Network Architectures for Document Classification

现在，在研究了这么多的高级的预训练模型之后，我们要反其道而行之，我们要讨论一个使用老的双向LSTM的模型来实现SOTA性能。但这正是我最后决定介绍它的原因。

我们常常因为几棵树木而错过森林。我们往往忘记，一个简单的调优的模型可能会获得与这些复杂的深度学习模型一样好的结果。本文的目的就是要说明这一点。

双向LSTM和正则化的组合能够在IMDb文档分类任务上实现SOTA的性能。

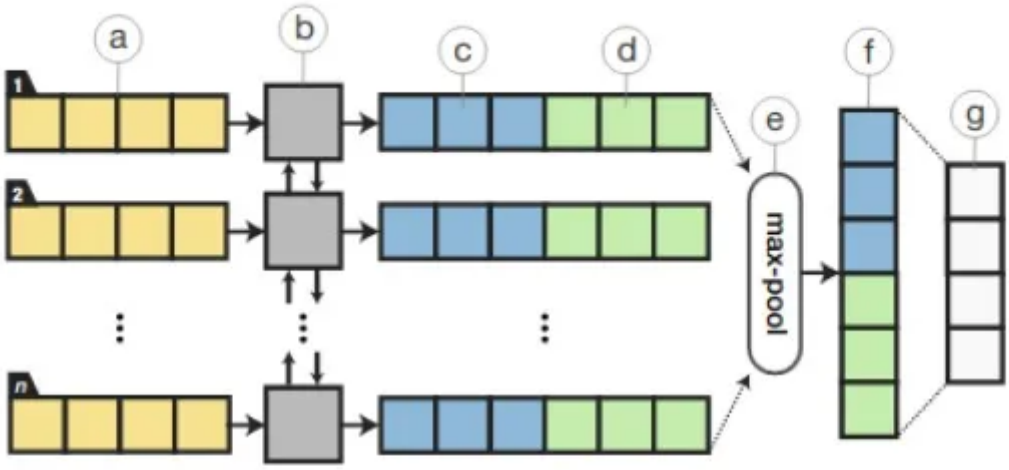


Figure 1: Illustration of the model architecture, where the labels are the following: (a) input word embeddings (b) BiLSTM (c, d) concatenated forward $h^f_{1:n}$ and backward $h^b_{1:n}$ hidden features (e) max-pooling over time (f) document feature vector (g) softmax or sigmoid output.

本文最有趣和值得注意的方面是：

- 它不使用注意力机制
- 这是第一篇使用LSTM +正则化技术进行文档分类的论文

这个简约的模型使用Adam优化器，temporal averaging和dropouts来达到这个高分。本文将这些结果与其他深度学习模型进行了实证比较，证明了该模型简单有效，并且结果说明了一切：

Task	Dataset	Metric	Metric Value	Global Rank
Document Classification	IMDb-M	Accuracy	52.8	1
Document Classification	Reuters-21578	F1	87.0	1
Document Classification	Yelp-5	Accuracy	68.7	1

对于行业而言，这种模型可以被认为是一种新颖的方法，在该行业中，构建可用于生产的模型并且在指标上取得高分非常重要。

结尾

在这里，我们讨论了最近在文本分类中达到最新基准的6种预训练模型。这些NLP模型表明还有更多的模型，我将期待今年学习它们。

所有这些研究中的一个令人敬畏的要素是这些预训练模型的可用性和开源性质。以上所有模型都有一个GitHub存储库，可以用于实现。另一个不可忽视的方面是它们在PyTorch上也可用。这强调了PyTorch正在快速取代TensorFlow作为构建深度学习模型的平台。

我鼓励你在各种数据集上尝试这些模型，并进行实验以了解它们的工作原理。

原文链接：https://www.analyticsvidhya.com/blog/2020/03/6-pretrained-models-text-classification/

- End -