[预训练语言模型的前世今生] 风起云涌

原创 管扬 朴素人工智能

来自专辑

预训练语言模型

前言

欢迎大家来到我们预训练语言模型的**专题系列分享**，本篇推送是该专题的第二篇！预训练语言模型已经成为了NLP研究中一个非常火热的话题，优秀的模型包括BERT，GPT2等都在学术研究、工业领域、算法比赛中大放光彩。

在专题上一期推送【萌芽时代】里，我们介绍了预训练语言模型思想的萌芽。今天我们推出的这篇推送， 将继续为大家介绍预训练语言模型是如何进一步发展和演进的。在此感谢清华大学自然语言处理实验室关于预训练语言模型的**必读论文**的整理（链接：https://github.com/thunlp/PLMpapers），我们将沿此脉络继续前行，分享在阅读中的理解和对某些常用模型实战中的一些心得。

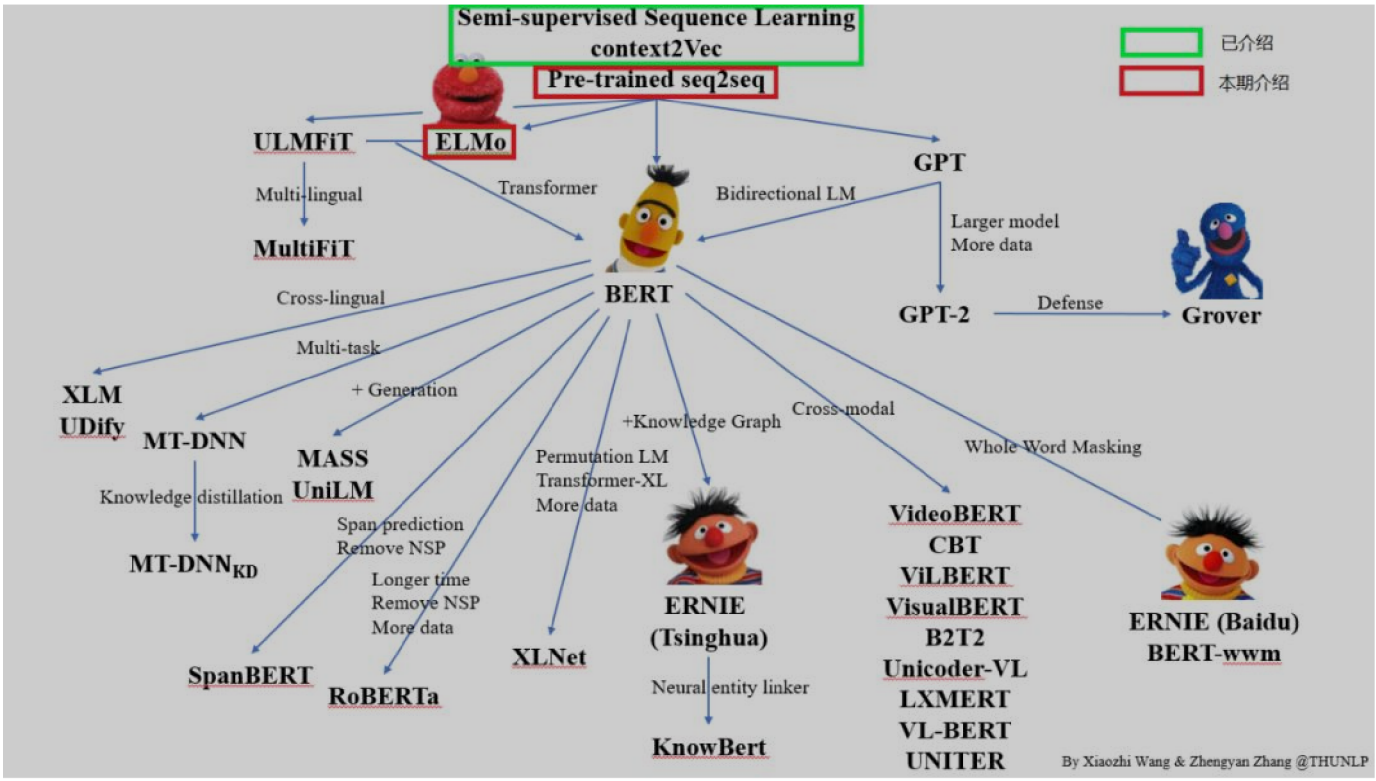

图1. PLMfamily （来源：https://github.com/thunlp/PLMpapers）

风起云涌（2017-2018）

随着预训练语言模型的能力被越来越多学者认可，该领域的论文呈现了**加速增长**的态势。上图收录了其中2015-2016年论文两篇、2017年-2018年论文四篇、以及2019年论文二十余篇。其

中2018年底到2019年初面世的BERT以其优异的性能，彻底改变了NLP发展的格局。在此之前，EMLo也以在不少自然语言任务上的效果的不俗表现而惊艳一时。从2016年后，大多数研究都开始重视长时的上下文语义在embeddings中所起作用和语言模型在大规模语料上提前预训练这两个核心观点。

上一期我们介绍了"上下文相关的embedding"，这里再来回顾一下。word2vec之类的词嵌入是和上下文无关的；当word2vec训练好词向量后，每一个独立的词在空间中就会有一个固定维度向量对应其语意，所有的词向量好像是被嵌入到了一个固定维度的空间中，所以叫做**word embedding**。朋友们可以查阅上期推荐阅读中的word2vec以及GloVe论文，或李宏毅老师的视频。

这样的话，同样是苹果这个词，在 “我今天买了一斤苹果和香蕉” 中和在 “我的苹果手机是去年刚买的” 这两句话中出现，它的词向量就还是一样的。那这就很不合理了，因为这两个“苹果”一个是手机一个是水果，表达的是完全不同的意思，属于**一词多义**，如果用同样的词向量表示，就违背了我们想要捕捉词与词之间关系的初衷了。手机中的“苹果”应该跟“华为”，“小米”，“三星”这些词在空间中接近。而水果中的“苹果”，应该跟“桃子”，“香蕉”，“橘子”更接近。为了弥补这个问题，许多学者就考虑，每个词在不同上下文中，它的向量表达都应该是不一样的，所以就有了上期专题中讲过的context2vec和我们这期后面会提到的ELMo这些模型的产生。

接下来我们将重点介绍本期两篇论文：

Unsupervised Pretraining for Sequence to Sequence Learning（2017）

这篇文章是在2017年ICLR会议上由Google Brain团队Prajit Ramachandran 、Peter J. Liu、Quoc V. Le共同发表的，与上期介绍的Google 2015年发表的 Semi-supervised Sequence Learning 可谓**一脉相承**。本文描述了一种通用的非监督预训练方法，提升了seq2seq模型的准确性。seq2seq模型是一种根据给定变长序列，通过特定方法生成另一个变长序列的方法，感兴趣的朋友可以查阅本文后的参考资料[1][2]。作者用两个语言模型的预训练权重分别初始化了seq2seq模型的encoder 与decoder，然后再用监督数据对模型进行finetune，这种做法在机器翻译和概要提取任务上大大超过了以前的纯监督模型，证明了预训练的过程直接提高了seq2seq模型的泛化能力，再次提出了预训练的重要性和通用性。

文章指出seq2seq模型以及所有深度神经网络的一个共同弱点就是当监督数据量很小的情况下，模型非常容易**过拟合**。他们提出了一种方法，使用两个预训练语言模型的参数来初始化seq2seq模型的encoder网络和decoder网络。并在finetune过程中，联合训练seq2seq的目标和语言模型的任务目标来避免过拟合的发生。他们的结果是该方法在CNN和Daily Mail文章数据上的英文转德文任务和概要提取任务的结果，超过了所有当时的最强算法。

同时文中通过做对比实验确认了，对机器翻译来说，模型对泛化能力的主要提升就来自于预训练的参数特征，而对摘要提取，encoder的预训练为模型效果的巨大提升和泛化能力的提高做出了贡献。图2为预训练seq2seq模型的结构，红色为encoder部分，蓝色为decoder部分，所

有方框内参数均为语言模型预训练的，而方框外的参数为随机初始化。在机器翻译任务中，作者对两种语言空间都收集了大量无标签数据，各自独立地训练了语言模型，作为encoder，decoder的初始化参数。为了使其后训练更加高效，decoder的softmax层也由目标侧的语言模型softmax参数初始化。

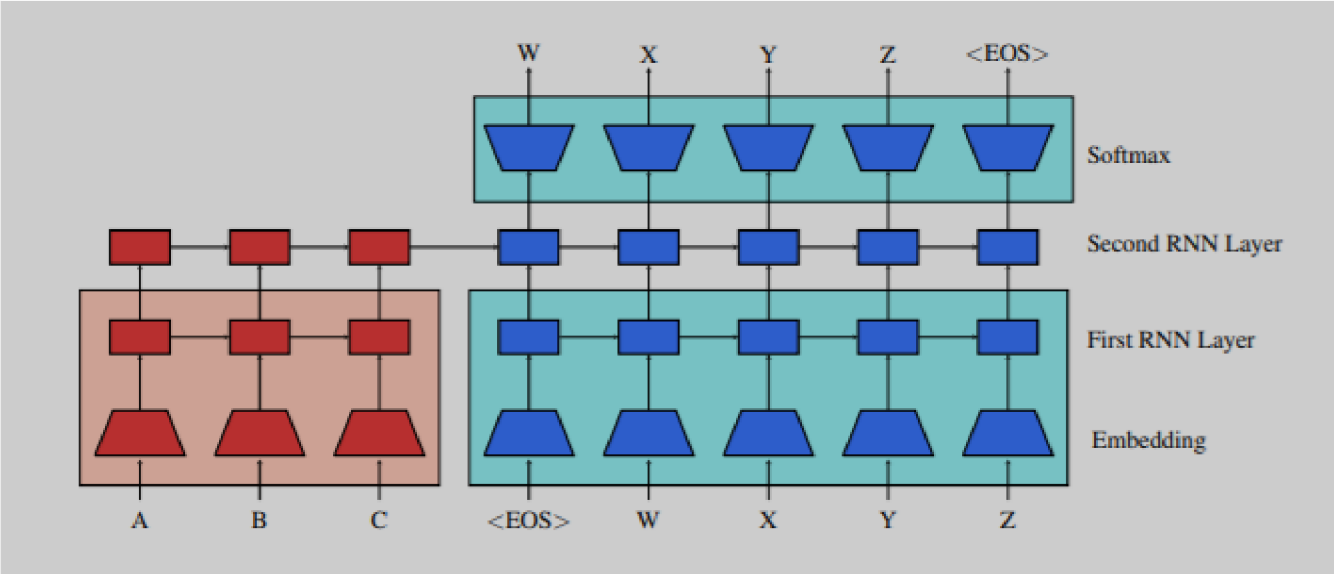

图2. seq2seq模型结构（来源https://arxiv.org/pdf/1802.05365.pdf）

当seq2seq模型按照上述方式被初始化以后，接着应该在监督数据上进行fine-tune。但作者引用了2013年Goodfellow的文章[3]，认为直接finetune会引起所谓的catastrophic forgetting（灾难性遗忘），即是模型在之前语言模型任务上的表现会急速下降，导致模型泛化能力降低，特别如果是在较小数据集上finetune的。所以他们通过将seq2seq模型loss和语言模型loss的平均，来达成**联合训练**，并显著提升了模型的效果。

作者提到除了为效果提供了最大贡献的预训练方法和seq2seq和语言模型的联合优化，另外还有两个贡献小但稳定的**优化点**，一个是residual connections（残差连接）， 另一个是Multi-layer attention（多层注意力机制）。使用残差连接原因是高层的LSTM参数是随机初始化的，可能会导致随机的梯度对预训练好的参数造成影响。所以，同时让第一层的LSTM也有梯度可以与decoder的Softmax之间流转，提高了模型稳定性和效果。多层注意力机制使模型对两层LSTM的节点都有关注，从而提升了模型效果。最后，文章在机器翻译和概要提取任务上进行了不少模型效果的实验比较，证明了方法可以打败当时最好的模型。

编者认为这篇文章最大的贡献是证明了在大量无监督数据上预训练语言模型，并在少量有监督数据上finetune这个思路对seq2seq模型**同样具有效性**，并提出了seq2seq目标和语言模型目标联合训练以提高泛化能力的想法。文章的思路借鉴了s上期介绍的Semi-supervised Sequence Learning对预训练语言模型的应用，并进一步将其发展，展现了该方法在自然语言处理中的**通用性**。

Deep contextualized word representations（ELMo)（2018）

接下来这篇就是在当时名噪一时的ELMo了，由Allen Institute for Artificial Intelligence 及University of Washington的多位作者联合发表，在2018年的NAACL 会议获得了最佳论文。

文章总结性地提出，模型在理想情况下应该做到两点，一个是对复杂的词语运用的建模，也就是理解语法与语意；另一个是建模该运用怎么在上下文语言内容变化时随之变化，也就是应对一词多义现象。本方法通过**深层双向语言模型**的内部状态来学习到词向量。所用的语言模型在一个很庞大的语料上预训练过。结果表明，学到的这些向量表达可以很容易地运用于现有的模型，并且大幅提升包含 question answering, textual entailment ， sentiment analysis 在内的六个NLP关键任务的state of the art结果。同时，论文揭示了，预训练语言模型能生成深层特征是关键，下游任务可以混合不同层的半监督语义信号，来提高自己的效果。

ELMo在大量不同领域的NLP任务中，都不同程度提升了原有state-of-art模型的效果，一时令大家侧目。之所以起名为EMLo（Embeddings from Language Models），是因为模型是从一个在大量语料上预训练的双向LSTM语言模型中提取embeddings。它最后得到的 embeddings 由内部不同层的**加权组合**所得，特别地，针对不同的任务，通过训练获得不同的加权方式，这也会提升模型效果，并明显超过只用顶层LSTM的结果。同时实验表明，更高层的LSTM层会捕捉关于词在上下文语言环境中的意义（不加修改就能在词义消歧任务上取得好的效果），而低层的LSTM会倾向于捕捉单个词的语意语法信息（在词性标注任务上表现良好）。总之所有的信号都很有用，下游任务可以通过训练选取最适合他们的信号或分配权重。

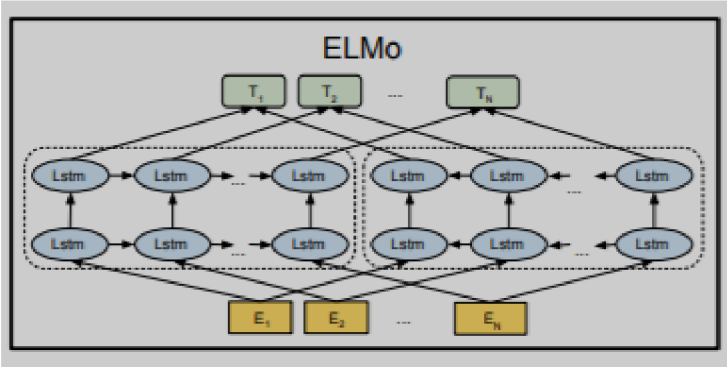

图3. ELMo模型结构图（来源：https://arxiv.org/pdf/1810.04805.pdf）

上图摘自BERT的论文，示意了ELMo的模型结构。虽然论文题目中有Deep，但实际最后ELMo的语言模型为两层。但文章指出对L层双向LSTM的情况下，应该会有L层前向的向量表征，L层后向的向量表征，加上原始的向量（图中的黄色的单元），一共**2 * L + 1**个向量表征。最简单的情况，是只提取最上层的LSTM的向量表征。更一般的情况是，所有层的向量表征都会被用上，并针对不同任务计算一个权重分配方式，进一步优化模型在新任务的效果。

ELMo语言模型跟之前的一些相似模型相比，首先它是**双向**的深层语言模型会优于单向， 这点在同期的许多论文中都有所提到。其次，在图3中可以看到，在上下层的LSTM之间有 **residual connection** ，加强了梯度的传播。另外，双向语言模型的训练目标是最大化前向和后向的联合对数似然概率，这点源于模型双向的特性。

$$\sum_{k=1}^{N} \Big(\log p(t_k \mid t_1, \ldots, t_{k-1}; \Theta_x, \overrightarrow{\Theta}_{LSTM}, \Theta_s) + \log p(t_k \mid t_{k+1}, \ldots, t_N; \Theta_x, \overleftarrow{\Theta}_{LSTM}, \Theta_s) \Big).$$

图4. ELMo语言模型训练目标(来源：https://arxiv.org/pdf/1802.05365.pdf)

最终，ELMo的语言模型经过**权衡**了模型性能、大小、以及所需算力定为两层双向LSTM，每层4096个units，512维projections。底层用的是字向量，进行字符卷积，再经过2个highway layers进入BiLSTM。最后在1B大小的Word Benchmark上 (Chelba et al., 2014) 训练了十轮。

经过预训练以后，实际下游模型使用起来就比较简单了。比如拿到一句句子，经过底层非上下文相关字嵌入层，生成字向量，就是图3中黄色的向量表征。经过语言模型的计算，内部各LSTM层输出加权后得到上方绿色的向量表征，和下方的黄色向量表征一起，作为下游模型的输入，进行接下来有监督的模型训练。

编者认为ELMo这篇文章的主要贡献是提出了**深层双向语言模型**的重要性，虽然ELMo只有两层，但作者在层数为L的假设下进行了讨论，并指出各层学习到的向量表征在各语言维度上有不同特性，加权后共同来使用会有更好的效果。同时ELMo拿出了让人信服的结果，在多个任务上表现出色，但可惜ELMo生不逢时，没过几个月，BERT就王者降临了，刷新了几乎能刷新的所有榜单。不是我军无能，而是敌人太狡猾呀~

未完待续

这期的论文就给大家分享到这里，感谢大家的阅读和支持，下期我们会给大家带来另外两篇2018年的重要论文的分享，敬请大家期待！

欢迎关注晴天1号，这里有很多最新最热的论文阅读分享，有问题或建议可以在公众号下留言。

参考资料

1. Sequence to sequence learning with neural networks

   http://de.arxiv.org/pdf/1409.3215

2. Learning phrase representations using RNN encoder-decoder for statistical machine translation

   https://www.aclweb.org/anthology/D14-1179.pdf

3. An empirical investigation of catastrophic forgetting in gradient-based neural networks

   https://arxiv.org/pdf/1312.6211.pdf

推荐阅读

- 能跟你聊DOTA的神经对话模型：Meena&DialoGPT
- LaserTagger: 文本生成任务的序列标注解决方案
- [CLS]预训练语言模型的前世今生[SEP]萌芽时代[SEP]
- 打造专属对话机器人，百度UNIT平台任务型对话体验
- REALM: Retrieval-Augmented Language Model Pre Training
- REALM后续：最近邻搜索，MIPS，LSH和ALSH