计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.191000167

面向自然语言处理的预训练技术研究综述

李舟军　范　宇　吴贤杰
北京航空航天大学计算机学院 北京 100191

摘　要　近年来,随着深度学习的快速发展,面向自然语言处理领域的预训练技术获得了长足的进步。早期的自然语言处理领域长期使用 Word2Vec 等词向量方法对文本进行编码,这些词向量方法也可看作静态的预训练技术。然而,这种上下文无关的文本表示给其后的自然语言处理任务带来的提升非常有限,并且无法解决一词多义问题。ELMo 提出了一种上下文相关的文本表示方法,可有效处理多义词问题。其后,GPT 和 BERT 等预训练语言模型相继被提出,其中 BERT 模型在多个典型下游任务上有了显著的效果提升,极大地推动了自然语言处理领域的技术发展,自此便进入了动态预训练技术的时代。此后,基于 BERT 的改进模型、XLNet 等大量预训练语言模型不断涌现,预训练技术已成为自然语言处理领域不可或缺的主流技术。文中首先概述预训练技术及其发展历史,并详细介绍自然语言处理领域的经典预训练技术,包括早期的静态预训练技术和经典的动态预训练技术;然后简要梳理一系列新式的有启发意义的预训练技术,包括基于 BERT 的改进模型和 XLNet;在此基础上,分析目前预训练技术研究所面临的问题;最后对预训练技术的未来发展趋势进行展望。

关键词:自然语言处理;预训练;词向量;语言模型
中图法分类号　TP391

Survey of Natural Language Processing Pre-training Techniques

LI Zhou-jun, FAN Yu and WU Xian-jie
School of Computer Science and Engineering, Beihang University, Beijing 100191, China

Abstract　In recent years, with the rapid development of deep learning, the pre-training technology for the field of natural language processing has made great progress. In the early days of natural language processing, the word embedding methods such as Word2Vec were used to encode text. These word embedding methods can also be regarded as static pre-training techniques. However, the context-independent text representation has limitation and cannot solve the polysemy problem. The ELMo pre-training language model gives a context-dependent method that can effectively handle polysemy problems. Later, GPT, BERT and other pre-training language models have been proposed, especially the BERT model, which significantly improves the effect on many typical downstream tasks, greatly promotes the technical development in the field of natural language processing, and thus initiates the age of dynamic pre-training. Since then, a number of pre-training language models such as BERT-based improved models and XLNet have emerged, and pre-training techniques have become an indispensable mainstream technology in the field of natural language processing. This paper first briefly introduce the pre-training technology and its development history, and then comb the classic pre-training techniques in the field of natural language processing, including the early static pre-training techniques and the classic dynamic pre-training techniques. Then the paper briefly comb a series of inspiring pre-training techniques, including BERT-based models and XLNet. On this basis, the paper analyze the problems faced by the current pre-training technology. Finally, the future development trend of pre-training technologies is prospected.

Keywords　Natural language processing, Pre-training, Word embedding, Language model

1　引言

自然语言处理是人工智能和语言学领域的分支学科,主要探讨如何处理及运用自然语言。近年来,随着深度学习方法的快速发展,自然语言处理领域中的机器翻译、机器阅读理解、命名实体识别等技术都取得了重要突破。借助于深度学习技术,面向自然语言处理领域的预训练技术也获得了长足的进步。

到稿日期:2019-09-25　返修日期:2020-01-15　本文已加入开放科学计划(OSID),请扫描上方二维码获取补充信息。
基金项目:国家自然科学基金(U1636211,61672081);软件开发环境国家重点实验室课题(SKLSDE-2019ZX-17);北京成像理论与技术高精尖创新中心课题(BAICIT-2016001)
This work was supported by the National Natural Science Foundation of China(U1636211,61672081),Fund of the State Key Laboratory of Software Development Environment (SKLSDE-2019ZX-17) and Beijing Advanced Innovation Center for Imaging Technology(BAICIT-2016001).
通信作者:李舟军(lizj@buaa.edu.cn)

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

在自然语言处理领域的背景下，预训练技术通过使用大规模无标注的文本语料来训练深层网络结构，从而得到一组模型参数，这种深层网络结构通常被称为“预训练模型”；将预训练好的模型参数应用到后续的其他特定任务上，这些特定任务通常被称为“下游任务”。

通常来说，大多数基于深度学习的自然语言处理任务可以分为以下 3 个模块：数据处理、文本表征和特定任务模型。其中，数据处理模块和特定任务模型模块需要根据具体任务的不同做相应设计，而文本表征模块则可以作为一个相对通用的模块来使用。类似于计算机视觉领域中基于 ImageNet[1] 预训练模型的做法，自然语言处理领域也可以预训练一个通用的文本表征模块，这种通用的文本表征模块对于文本的迁移学习具有重要意义。

以 Word2Vec[2-3] 为代表的词向量技术是自然语言处理领域一直以来最常用的文本表征方法，但这种方法仅学习了文本的浅层表征，并且这种浅层表征是上下文无关的文本表示，对于后续任务的效果提升非常有限[4-6]。直到 ELMo[7] 提出了一种上下文相关的文本表示方法，并在多个典型下游任务上表现惊艳，才使得预训练一个通用的文本表征模块成为可能。随后，GPT[8] 和 BERT[9] 等预训练语言模型相继被提出，自此便进入了动态预训练技术的时代。其中，BERT 在击败 11 个典型下游任务的 State-of-the-art 结果之后，成为了自然语言处理领域预训练技术的重要里程碑，极大地推动了自然语言处理领域的发展。此后，基于 BERT 的改进模型、XLNet[10] 等大量预训练语言模型涌出，预训练技术逐渐发展成了自然语言处理领域不可或缺的主流技术。

预训练技术取得的巨大成功，很大程度上归功于其实现了迁移学习[11] 的概念。迁移学习本质上是在一个数据集上训练基础模型，通过微调等方式，使得模型可以在其他不同的数据集上处理不同的任务。预训练的过程如上文所述，是将预训练好的模型的相应结构和权重直接应用到下游任务上，从而实现“迁移学习”[12-15] 的概念，即将预训练模型“迁移”到下游任务。

本文主要概述面向自然语言处理领域的预训练技术。按照时间顺序，预训练技术大致可分为 3 个阶段：早期的静态预训练技术、经典的动态预训练技术和最新发布的新式预训练技术。第 2 节简要概述预训练技术的整个发展历史；第 3 节详细介绍自然语言处理领域早期的静态预训练技术和经典的动态预训练技术；第 4 节主要梳理近期发布的有启发意义的新式预训练技术；第 5 节分析目前预训练技术研究所面临的问题；第 6 节对自然语言处理领域的预训练技术的未来发展趋势进行展望。

2 发展历史

预训练技术最早被应用于计算机视觉领域，自 ResNet[6] 出现后，便开始在视觉领域广泛应用。大量实验[16-18] 证实，使用预训练技术可以大幅提升下游任务的效果。更重要的是，充分使用预训练模型极大地改善了下游任务模型对标注数据数量的要求，从而可以很好地处理一些难以获得大量标注数据的新场景。因此，在计算机视觉领域，使用训练好的预训练模型，再用特定下游任务数据微调模型，已经成为惯例[19-22]。

借鉴视觉领域的做法，自然语言处理领域开始尝试使用预训练技术实现迁移学习。一般来说，自然语言处理领域使用语言模型来做预训练[23]。语言模型能够量化一个句子近似人类自然表达的概率。大量研究表明：语言模型可以捕获与下游任务相关的许多知识，例如长期依赖、层次关系和情绪[24-25]。语言模型的最大优势之一是训练数据可以来自任意的无监督文本语料，这意味着可以获得无限量的训练数据。

早期的预训练技术是一种静态技术。2003 年 Bengio 提出的 NNLM[26] 是使用神经网络实现语言模型的经典范例。2013 年，Word2Vec 借鉴 NNLM 的思想，提出使用语言模型得到词向量。随后，GloVe[27] 和 FastText[28] 等相继被提出。这种词向量的方法作为早期的预训练技术，逐渐成为了最常用的文本表征技术[29-33]，对大多数任务是有帮助的，但其本质是一种静态的预训练技术，即不同上下文中的同一词语具有相同的词向量，因而无法解决自然语言中经常出现的多义词问题，且其给下游任务带来的提升也非常有限。

对此，预训练语言模型提供了一种动态的预训练技术方案。2018 年，ELMo 提出了一种上下文相关的文本表示方法，并在多个典型任务上表现惊艳，能有效处理一词多义问题。其后，GPT，BERT，XLNet 等预训练语言模型相继被提出，预训练技术开始在自然语言处理领域大放异彩。从实验效果来看，预训练语言模型在诸多下游任务上的表现较传统词向量方法取得了很大的提高，此类下游任务几乎涵盖了自然语言处理领域的典型任务，例如句子语义关系判断、命名实体识别、阅读理解等，这充分说明了预训练模型的普适性。此外，这些模型已经被证明具有极高的采样效率，只需数百个样本就可以取得很好的性能，甚至可以实现零样本学习。

预训练语言模型的核心在于关键范式的转变：从只初始化模型的第一层，转向了预训练一个多层网络结构。传统的词向量方法只使用预训练好的静态文本表示，初始化下游任务模型的第一层，而下游任务模型的其余网络结构仍然需要从头开始训练。这是一种以效率优先而牺牲表达力的浅层方法，无法捕捉到那些也许更有用的深层信息[34-36]；更重要的是，其本质上是一种静态的方式，无法消除词语歧义。而预训练语言模型是预训练一个多层网络结构，用以初始化下游任务模型的多层网络结构，可以同时学到浅层信息和深层信息。此外，预训练语言模型是一种动态的文本表示方法，会根据当前上下文对文本表征进行动态调整，经过调整后的文本表征更能表达词语在该上下文中的具体含义，能有效处理一词多义的问题。

3 经典预训练技术

本节主要梳理经典的预训练技术，包括早期的静态预训练技术和动态预训练技术。静态预训练技术主要概述 NNLM，Word2Vec，GloVe 和 FastText；动态预训练技术主要梳理经典的 ELMo，GPT 和 BERT。

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

3.1 静态预训练技术

2003 年 Bengio 提出的 NNLM 是早期使用神经网络实现语言模型的经典模型。2013 年，Word2Vec 借鉴 NNLM 的思想，提出用语言模型得到词向量。随后，GloVe 和 FastText 相继被提出，这种静态的预训练技术逐渐成为了最常用的文本表征技术。

3.1.1 NNLM 模型

NNLM 使用神经网络来搭建语言模型，并且优化后的模型的副产品就是词向量。语言模型能够量化一个句子近似人类自然表达的概率，如果文本序列 S 用 $(w_0, w_1, \cdots, w_{t-1})$ 来表示，那么语言模型即计算：

$$P(S)=P(w_0)P(w_1 \mid w_0)\cdots P(w_{t-1} \mid w_0, w_1, \cdots, w_{t-2}) \tag{1}$$

但式(1)的计算过于复杂。早期基于统计的语言模型一般会引入马尔可夫假设：假定一个句子中的词只与它前面的 n 个词相关，并且用词频来估计语言模型中的条件概率，这使得语言模型的计算变得可行。然而，这种基于统计的语言模型无法把 n 取得很大，否则会带来参数过多的问题，因而无法建模语言中上下文较长的依赖关系，具有很大的局限性。

2003 年，Bengio 将深度学习的思想融入语言模型中，并发现将训练得到的 NNLM 模型的第一层参数当作词语的文本表征时，能够很好地获取词语之间的相似度[26]。

NNLM 模型的结构如图 1 所示，分为 3 个部分：词到词向量的映射；词向量到隐藏层的映射；隐藏层到输出层的映射。其损失函数如下：

$$L=\frac{1}{T}\sum_t \log P(w_t \mid w_{t-1}, \cdots, w_{t-n+1}; \theta)+R(\theta) \tag{2}$$

其中，$R(\theta)$ 为正则化项。由损失函数可以看出，NNLM 本质上是一个 N-Gram 的语言模型。此外，NNLM 的参数个数是窗口大小为 n 的线性函数，此时的取值不再受模型参数量的限制，因此能对更长的依赖关系进行建模。

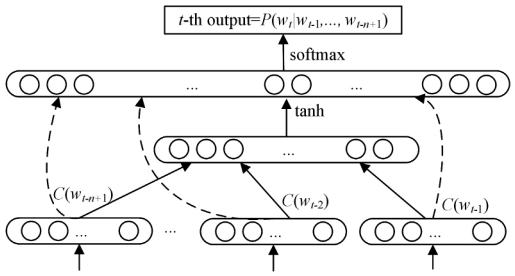

图 1 NNLM 模型的结构

Fig. 1 Structure of NNLM model

NNLM 的主要贡献是创见性地将模型的第一层特征映射矩阵当作词的文本表征，从而开创了将词语表示为向量形式的模式，这直接启发了后来的 Word2Vec 的工作。

3.1.2 Word2Vec 模型

自 NNLM 被提出后，出现了很多对其进行改进的工作，如 LBL[37-38]，C&W[39] 和 RNNLM[40] 模型等。这些方法主要针对以下两个问题进行优化：(1)NNLM 只用了上文信息，并没有利用更多的上下文信息；(2)NNLM 的输出层存在词表较大所带来的计算量太大的问题。

2013 年 Mikolov 发布的 Word2Vec 针对 NNLM 存在的问题，提出使用语言模型作词向量。Word2Vec 对 NNLM 的优化主要体现在模型结构和训练技巧两方面。

为了简化模型结构，Word2Vec 提出 CBOW 和 Skip-gram 两种模型结构。CBOW 使用语境的上下文来预测中心词，模型结构上取消了 NNLM 中的隐藏层，直接将输入层和输出层相连。此外，求语境向量时丢弃词序，直接由语境中的每一个词向量简单求和后得到。Skip-gram 的基本思想与 CBOW 非常类似，只是由中心词预测语境词。

无论 CBOW 还是 Skip-gram，都是对 NNLM 的网络结构进行化简，本质上是两个全连接层相连，大大减少了模型的参数数量，且与窗口大小无关。模型中的语境向量由上下文的词向量求和得到，更多地利用了上下文信息，弥补了 NNLM 只利用了上文信息的缺陷。

优化训练技巧主要针对输出层的普通 Softmax 计算量过大的问题，使用了 Hierarchical Softmax 和负采样技术。

Hierachical Softmax 的基本思想是对大小为 V 的词典按照词频构建哈夫曼树，每一个词都处于哈夫曼树的叶子节点上。普通 Softmax 要遍历词典中的每一个词才能计算目标词发生的概率；而在 Hierachical Softmax 中，只须找到目标词在哈夫曼树中的路径即可，每一个路径节点都对应一个二分类问题，这样就将原本的一个 V 分类问题变成了 $\log V$ 次的二分类问题。

负采样的思想启发于 C&W 模型[39] 中构造负样本的方法，同时参考了 NCE[41] 的思想，每次按照一定的概率随机采样 K 个词 $w_i(i=1,2,\cdots,K)$ 当作负例，将正例记作 w_0。此时，模型的训练目标是最大化似然函数：

$$\prod_{i=0}^{K} P(context(w_0), w_i) \tag{3}$$

这样就将原来的 V 分类问题变成了 K 分类问题。

Word2Vec 经过模型结构和训练技巧的双重优化，终于使得在大规模无监督的文本语料上快速训练得到词向量成为了现实，并且得到的词向量在语义上有非常好的表现[7-8]。

3.1.3 GloVe 模型

相比 Word2Vec，GloVe 主要利用词语的共现信息构建模型，它的训练主要分为统计共现矩阵和训练获取词向量两个步骤。损失函数如下：

$$J=\sum_{i=1}^{V}\sum_{j=1}^{V} f(X_{ij})(\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{w}_j+b_i+b_j-\log X_{ij})^2 \tag{4}$$

其中，V 是词典大小，$\boldsymbol{w}_i$ 和 $\boldsymbol{w}_j$ 分别是词典中第 i 个词和第 j 个词的词向量，X_{ij} 是这两个词在窗口中的共现频率，$f(X_{ij})$ 是一个权重系数。

从损失函数来看，GloVe 的核心思想是共现的两个词语的词向量的点乘要尽量拟合其共现频率的对数值。两个词的共现频率越高，其对数值也越高，因而要求两个词语的词向量的点乘也越大。这其实包含了两层含义：

(1)要求各自词向量的模尽量大，通常来说，词向量模长会随着词频的增大而增大；

(2)要求两个词向量的夹角尽量小，两个词出现在同一个语境下的频率越大，则其语义越接近，因而词向量的夹角也偏向于越小。

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

GloVe 其实是没有网络结构的，整个算法都是基于矩阵分解的做法来获取词向量，本质上与 LSA[42] 这种基于 SVD[43] 的矩阵分解方法类似。

3.1.4 FastText 模型

Word2Vec 和 GloVe 都是使用无监督数据得到词向量，而 FastText 则是利用带有监督标记的文本分类数据完成训练。

FastText 的网络结构与 CBOW 基本一致，均使用了 Hierachical Softmax 技巧来加速训练。其模型结构与普通的 CBOW 在输入层和输出层有所不同：在输入层，CBOW 输入的是窗口中除目标词外的所有其他词，而 FastText 为了利用更多的语序信息，增加了 N-Gram 的输入信息；在输出层，CBOW 的预测目标是语境中的中心词，而 FastText 的预测目标是输入文本的类别，因此也称 FastText 是一个监督模型。

FastText 最大的特点在于其预测速度有较大优势，文献[28]对这一点也做了详细的实验验证。在一些分类数据集上，FastText 通常可以把要耗时几小时甚至几天的模型训练时间大幅压缩到几秒钟。

3.1.5 小结

以 Word2Vec 为代表的静态预训练技术将每一个词表示成词向量，并将其语义通过上下文来表征，其理论基础来自 1954 年 Harris[44] 提出的分布假说：上下文相似的词，其语义也相似。

这些静态预训练技术的贡献远不只是给每一个词赋予一个分布式的表征，它开启了一种全新的模型训练方式——迁移学习。使用词向量方法学习到的词语表征，初始化下游任务网络结构的第一层，能够为下游任务带来显著的效果提升，以至于这种做法早已成为业内的标配，极大地促进了自然语言处理领域的发展。

3.2 动态预训练技术

静态的预训练技术推动了自然语言处理领域的快速发展，然而这种静态的词向量技术无法较好地处理一词多义问题。对此，预训练语言模型提供了一种动态的预训练技术方案。2018 年，ELMo 提出了一种上下文相关的文本表示方法，能够有效处理一词多义问题。其后，GPT 和 BERT 等预训练语言模型相继被提出，尤其是 BERT 模型横扫自然语言处理领域的诸多典型任务，成为了自然语言处理领域的一个重要里程碑。

3.2.1 ELMo 模型

静态的词向量方法存在一个重要缺陷，即无法较好地处理一词多义问题；而 ELMo 通过使用针对语言模型训练好的双向 LSTM 来构建文本表示，由此捕捉上下文相关的词义信息，因而可以更好地处理一词多义问题。

为了使用大规模无监督语料，ELMo 使用两层带残差的双向 LSTM 来训练语言模型，如图 2 所示。此外，ELMo 借鉴了 Jozefowicz 等[45] 的做法，针对英文形态学上的特点，在预训练模型的输入层和输出层使用了字符级的 CNN 结构。这种结构大幅减小了词表的规模，很好地解决了未登录词的问题；卷积操作也可以捕获一些英文中的形态学信息；同时，训练双向的 LSTM，不仅考虑了上文信息，也融合了下文信息。

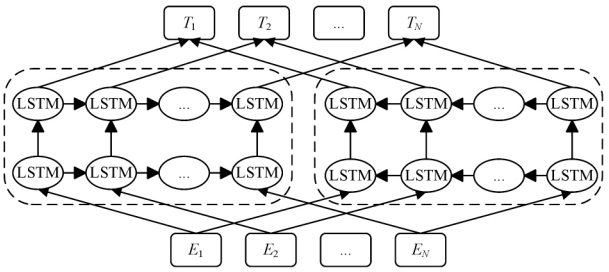

图 2 ELMo 模型的结构

Fig. 2 Structure of ELMo model

从预训练模型的迁移方式来看，ELMo 是一种特征抽取式的预训练模型。对于第 k 个词来说，ELMo 有 3 层的文本表示可以利用：输入层 CNN 的输出 $h_{k,0}$、第一层双向 LSTM 的输出 $h_{k,1}$ 和第二层双向 LSTM 的输出 $h_{k,2}$。设 3 层文本表示如下：

$$R_k=\{h_{k,j}\}(j=0,1,2) \tag{5}$$

则第 k 个词经过预训练模型得到的文本表示为：

$$ELMo_k^{\text{task}}=\gamma^{\text{task}}\sum s_j^{\text{task}}h_{k,j}(j=0,1,2) \tag{6}$$

其中，γ^{task} 是一个缩放因子，用以将 ELMo 输出的向量与下游任务的向量拉到同一分布；s_j^{task} 是针对每一层的输出向量设置的不同权值参数，用以组合不同层次的语义信息。

ELMo 模型不仅简单，而且表现出众，在自然语言处理领域的 6 个典型下游任务的数据集上全面刷新了最优成绩，尤其在阅读理解任务上提高了 4.7 个点[12]。其主要贡献是提供了一种新的文本表征的思路：在大规模无监督数据上训练预训练语言模型，并将其迁移到下游特定任务中使用。

3.2.2 GPT 模型

ELMo 使业界意识到了基于大规模语料集预训练的语言模型的威力；同期，ULMFiT[46] 提出的多阶段迁移方法和微调预训练模型的技巧为后来预训练技术的发展提供了重要指导意义；与此同时，Transformer[47] 在处理长期依赖性方面比 LSTM 有更好的表现，它在机器翻译等任务上取得的成果也使一些业内人士开始认为其是 LSTM 的替代品。在此背景下，OpenAI 的 GPT 预训练模型应运而生。

GPT 主要借鉴了谷歌 Liu 等的工作[48]，使用生成式方法来训练语言模型。该工作中的解码器在逐字生成翻译的过程中屏蔽了后续的词语序列，天然适合语言建模，因此 GPT 采用了 Transformer 中的解码器结构，并没有使用一个完整的 Transformer 来构建网络。GPT 模型堆叠了 12 个 Transformer 子层，并用语言建模的目标函数来进行优化和训练。GPT 模型的结构如图 3 所示。

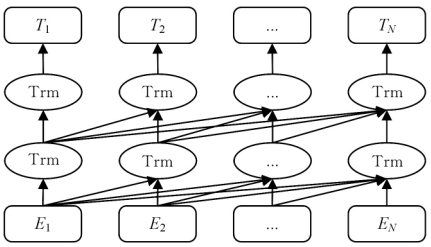

图 3 GPT 模型的结构

Fig. 3 Structure of GPT model

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

在迁移学习的模型设计方面，GPT 同样借鉴了 Liu 等的做法，巧妙地将整个迁移学习的框架做到非常精简和通用。在输入层，若输入只有一个序列，则直接在原序列的首尾添加表示开始和末尾的符号；若输入有两个序列，则通过一个中间分隔符“$”将其连接成一个序列，然后同样在开头和末尾添加标记符号。这套输入的表示方法，基本可以使用同一个输入框架来表征大多数文本问题。除此之外，在输出层，只需要接入一个全连接层或其他简单结构，一般不需要非常复杂的模型设计。

基于这种输入层和输出层的通用化设计，只要中间多层解码器层的表征能力足够强，迁移学习在下游任务中的威力就会变得非常强大。GPT 在公布的结果中，一举刷新了自然语言处理领域中的 9 项典型任务，效果不可谓不惊艳。GPT 模型使用的是 Tranformer 的解码器结构，正是 Transformer 强大的表征能力，为最终的模型表现奠定了坚实的基础。

3.2.3 BERT 模型

GPT 模型虽然达到了很好的效果，但本质上仍是一种单向语言模型，对语义信息的建模能力有限。因此，建立一个基于 Transformer 的双向预训练语言模型是一种重要的研究思路。

BERT 使用了一种特别的预训练任务来解决这个问题。与 GPT 相同，BERT 同样通过堆叠 Transformer 子结构来构建基础模型，模型结构如图 4 所示，但通过 Masked-LM 这个特别的预训练方式达到了真双向语言模型的效果。

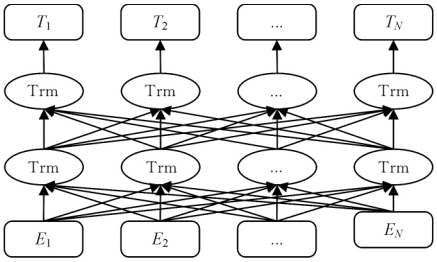

图 4 BERT 模型的结构

Fig. 4 Structure of BERT model

Masked-LM 预训练类似于一种完形填空的任务，即在预训练时，随机遮盖输入文本序列的部分词语，在输出层获得该位置的概率分布，进而极大化似然概率来调整模型参数。文献[9]实际随机选择文本序列中 15% 的词用于后续替换，但这些词也并非全部被替换为[MASK]，其中 10% 替换为随机词，10% 保持不变。这种操作可以理解为通过引入噪声来增强模型的鲁棒性。

与此同时，为了更好地处理多个句子之间的关系，BERT 还利用和借鉴了 Skip-thoughts[26] 中预测下一句的任务来学习句子级别的语义关系。具体做法是：按照 GPT 提出的组合方式将两个句子组合成一个序列，模型预测后面句子是否为前面句子的下文，也就是建模预测下一句的任务。因此，BERT 的预训练过程实质上是一个多任务学习的过程，同时完成训练 Masked-LM 和预测下一句这两个任务，损失函数也由这两个任务的损失组成。

在预训练细节上，BERT 借鉴了 ULMFiT 的一系列策略，使模型更易于训练。在如何迁移到下游任务方面，BERT 主要借鉴了 GPT 的迁移学习框架的思想，并设计了更通用的输入层和输出层。此外，在预训练数据、预训练模型参数量和计算资源上，BERT 也远多于早期的 ELMo 和 GPT。BERT 的表现是里程碑式的，在自然语言处理领域的 11 项基本任务中获得了显著的效果提升。而自然语言处理领域的许多后续研究一般也以 BERT 模型为基础进行改进，学界普遍认为，从 BERT 模型开始，自然语言处理领域终于找到了一种方法可以像计算机视觉那样进行迁移学习。

总而言之，BERT 的出现是建立在前期很多重要工作之上的，包括 ELMo，ULMFiT，GPT，Transformer 以及 Skip-thoughts 等，是一个集大成者。BERT 的出现极大地推动了自然语言处理领域的发展，凡需要构建自然语言处理模型者，均可将这个强大的预训练模型作为现成的组件使用，从而节省了从头开始训练模型所需的时间、精力、知识和资源。

3.2.4 小结

虽然静态的预训练技术带来了一定程度的性能提升，但是这种提升非常有限；更重要的是，这种静态的词向量技术无法解决一词多义问题。ELMo 的出现开创了一种上下文相关的文本表示方法，很好地处理了一词多义问题，并在多个典型任务上有了显著的效果提升。其后，GPT 和 BERT 等预训练语言模型相继被提出，自此便进入了动态预训练技术的时代。尤其是 BERT 的出现，横扫了自然语言处理领域的多个典型任务，极大地推动了自然语言处理领域的发展，成为预训练史上一个重要的里程碑模型。此后，基于 BERT 的改进模型、XLNet 等大量新式预训练语言模型涌出，预训练技术在自然语言处理领域蓬勃发展。在预训练模型的基础上，针对下游任务进行微调，已成为自然语言处理领域的一个新范式。

4 新式预训练技术

BERT 的出现开启了一个新时代，此后涌现出了大量的预训练语言模型。这些新式的预训练语言模型从模型结构上主要分为两大类：基于 BERT 的改进模型和 XLNet。基于 BERT 的改进模型主要是针对原生的 BERT 模型进行改进，主要改进方向包括：改进生成任务、引入知识、引入多任务、改进掩码方式，以及改进训练方法。基于 BERT 的改进模型都是自编码语言模型；而 XLNet 与 BERT 模型区别较大，是自回归语言模型的一个典型范例。

4.1 基于 BERT 的改进模型

本节主要介绍了在各个方向上基于 BERT 的改进工作。改进的方向主要包括：改进生成任务、引入知识、引入多任务、改进掩码方式，以及改进训练方法。

4.1.1 改进生成任务

由于 BERT 本身在预训练过程和使用过程中不一致，并且没有为生成任务设计相应的机制，导致其在生成任务上效果不佳。本节主要介绍 MASS[49] 和 UNILM[50] 两个模型，这两个模型基于 BERT 改进了其在生成任务上的表现。

MASS 的模型结构如图 5 所示。MASS 使用 4 层的 Transformer 结构，训练数据是单句话，编码器模块会随机掩码连续的个词，然后把这些词放入解码器模块的相应位置。

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

MASS 期望解码器模块利用编码器编码的信息和解码前面的词，来预测这些被掩码的词。

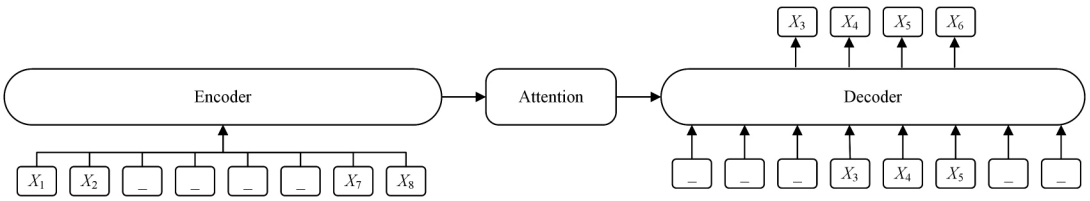

图 5 MASS 模型的结构

Fig. 5 Structure of MASS model

BERT 和 GPT 都是 MASS 的特例。当 $k=1$，即随机掩码单个词时，MASS 就退化成 BERT；当与句子长度相等，即掩码所有词时，MASS 就退化成 GPT，即标准的单向语言模型。

MASS 把 BERT 推广到生成任务，并设计统一了 BERT 和传统单向语言模型框架 BERT＋LM，使用 BERT 作为编码器，使用标准单向语言模型作为解码器。文献[49]在 WMT14 英语-法语、WMT16 英语-德语和 WMT16 英语-罗马尼亚语的机器翻译数据上进行了对比实验，实验结果均为 BLEU 评测值，如表 1 所列。从表中数据可以看出，MASS 优于对比框架 BERT＋LM。

表 1 BERT＋LM 与 MASS 的对比

Table 1 Comparison of BERT＋LM and MASS

Dataset	BERT＋LM	MASS
en-fr	33.4	37.5
fr-en	32.3	34.9
en-de	24.9	28.3
de-en	32.9	35.2
en-ro	31.7	35.2
ro-en	30.4	33.1

UNILM 模型的核心框架也是 Transformer，但同时以双向语言模型、单向语言模型和 Seq2Seq 语言模型为目标函数。这些目标函数共享一个网络结构，训练也都使用了类似 BERT 中的掩码机制。

与 BERT 的双向语言模型不同，单向语言模型在训练时不能使用下文的信息。Seq2Seq 语言模型在解码器预测时也有与单向语言模型类似的约束，UNILM 使用掩码机制来满足这些约束。

UNILM 模型训练时目标函数的设定也参照 BERT，但要兼顾双向语言模型、单向语言模型和 Seq2Seq 语言模型；使用的模型大小与 BERT-large 相同，即 24 层的 Transformer。在微调阶段，对于自然语言理解任务，UNILM 和 BERT 的处理相同；对于自然语言生成任务，UNILM 随机掩码解码器中的词，再进行预测。

UNILM 通过在预训练阶段同时训练双向语言模型、单向语言模型和 Seq2Seq 语言模型，使用掩码机制解决了语言模型中的约束问题，可以更好地处理自然语言理解和自然语言生成的各种任务。从实验结果来看，UNILM 在摘要生成任务、问题生成任务、生成式问答任务和对话回复生成任务上的效果提升明显；同时，在其他自然语言处理任务上也有较大的效果提升，如在 GLUE 上首次不加外部数据的情况下的实验效果优于 BERT[50]。

4.1.2 引入知识

BERT 模型通过训练 Masked-LM，利用多层双向 Transformer 的建模能力，取得了很好的效果。但是，BERT 模型主要建模原始的语言内部的信号，较少利用语义知识单元建模，导致模型很难学出语义知识单元的完整语义表示。这个问题在中文方面尤为明显，例如，对于“乒[X]球”“清明上[X]图”等词，BERT 模型可以通过字的搭配推测出掩码的字信息，但没有显式地对语义知识单元（如乒乓球、清明上河图）及其对应的语义关系进行建模。本节主要介绍 ERNIE1.0[51] 和 ERNIE(THU)[52] 两个模型，它们通过引入知识，使预训练模型学习到海量文本中蕴含的潜在知识，进一步提升了预训练语言模型在各个下游任务中的效果。

针对 BERT 模型的不足，百度提出基于知识增强的 ERNIE1.0 模型，通过建模海量数据中的实体概念等先验语义知识，学习语义知识单元的完整语义表示。ERNIE1.0 的模型结构与 BERT 基本一致，不同点在于 BERT 是对字进行随机掩码，而 ERNIE1.0 通过掩码词和实体概念等完整语义单元来训练 Masked-LM，从而使得模型对语义知识单元的表示更贴近真实世界。

相较于 BERT 基于局部词语共现学习的语义表示，ERNIE1.0 直接对语义知识单元进行建模，增强了模型的语义表示能力。表 2 对 BERT 与 ERNIE1.0 进行了对比说明。

表 2 BERT 与 ERNIE1.0 的数据对比

Table 2 Data comparison of BERT and ERNIE1.0

模型	哈	尔	滨	是	黑	龙	江	省	会
BERT	哈	X	滨	是	X	龙	江	省	会
ERNIE1.0	X	X	X	是	黑	龙	江	省	会

在 BERT 模型中，通过『哈』与『滨』的局部共现，可以判断出『尔』字，然而模型没有学习与『哈尔滨』相关的知识。而 ERNIE1.0 通过学习词与实体的表达，使模型能够建模出『哈尔滨』与『黑龙江』的关系。

此外，在训练数据方面，BERT 仅使用百科类语料训练模型；而 ERNIE1.0 则对此进行了改进，使用包括百科类、新闻资讯类、论坛对话类语料来训练模型，进一步提升了模型的语义表示能力。

相比 BERT 而言，ERNIE1.0 的优势在于：通过学习实体概念知识，可以获得知识单元的完整语义表示；对训练语料的扩展，尤其是论坛对话类语料的引入，增强了模型的语义表示能力。如表 3 所列，在自然语言推断、语义相似度、命名实体识别、情感分析、问答匹配任务的公开中文数据集上的实验结果表明：ERNIE1.0 模型较 BERT 取得了更好的效果[51]。

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

表 3 ERNIE1.0 与 BERT 的对比实验

Table 3 ERNIE1.0 and BERT compative experiments

(单位:%)

Task	Metrics	BERT		ERNIE1.0	
		dev	test	dev	test
XNLI	*acc*	78.1	77.2	79.9	78.4
LCQMC	*acc*	88.8	87.0	89.7	87.4
MSRA-NER	*f*1	94.0	92.6	95.0	93.8
ChnSentiCorp	*acc*	94.6	94.3	95.2	95.4
	mrr	94.7	94.6	95.0	95.1
Nlpcc-DBQA	*f*1	80.7	80.8	82.3	82.7

BERT 存在只学习语言相关的信息,而忽略了将知识信息整合到语言理解中的缺陷。清华大学与华为的研究者认为,知识图谱中的多信息实体可以作为外部知识改善语言表征,并提出 ERNIE(THU)模型,该模型通过使用知识图谱增强 BERT 的预训练效果。

ERNIE(THU)主要分为抽取知识信息与训练语言模型两大步骤。在抽取知识信息部分,研究者首先识别文本中的命名实体,将识别到的实体与知识图谱中的实体进行匹配。在训练语言模型部分,与 BERT 类似,ERNI(THU)采用 Masked-LM 任务以及预测下一句任务作为预训练的目标。为了更好地融合文本和知识特征,研究者设计了新的预训练目标,即随机掩码掉一些对齐了输入文本的命名实体,并要求模型从知识图谱中选择合适的实体以完成对齐。

ERNIE(THU)是结合大规模语料库和知识图谱训练出的增强版的语言表征模型,新的预训练目标要求模型同时聚合上下文和知识事实的信息,充分利用词汇、句法和知识信息,从而构建一种知识化的语言表征模型。

ERNIE(THU)针对知识驱动型任务进行了实验,实验结果表明,ERNIE(THU)在知识驱动型任务中效果显著,超过当前最佳的 BERT,如表 4 所列。此外,在其他类型的自然语言处理任务上,ERNIE(THU)也能获得与 BERT 相媲美的性能[52]。

表 4 ERNIE(THU)和 BERT 的对比实验

Table 4 ERNIE(THU) and BERT comparative experiments

(单位:%)

Dataset	Metrics	BERT	ERNIE(THU)
FIGER	*acc*	52.04	57.19
Open Entity	*f*1	76.37	78.42
FewRel	*f*1	84.89	88.32
TACRED	*f*1	66.00	67.97

4.1.3 引入多任务学习

在预训练模型背景下,引入多任务学习指在预训练模型过程中同时学习多个任务,这些任务在训练过程中共享预训练模型的结构和参数,利用多个任务之间的相关性来改进预训练模型的性能和泛化能力。BERT 的预训练过程实质上是一个多任务学习的过程,通过同时训练 Masked-LM 和预测下一句两个任务,提高了预训练模型的语义表达能力。本节主要介绍了 MT-DNN[53] 和 ERNIE2.0[54] 两个模型,它们通过引入多任务学习来提升预训练语言模型的表现。

MT-DNN 的模型架构主要包括输入层、文本编码层和任务特定层。文本编码层采用与 BERT 相同的机制,并在后续学习具体任务时共享参数;任务特定层是特定于具体任务的,例如单句分类、文本相似性、成对文本分类等任务。

MT-DNN 的训练过程分为两个阶段:预训练阶段和微调阶段。预训练阶段与 BERT 相同,通过训练 Masked-LM 和预测下一句两个无监督任务学习共享的文本编码层的参数;不同点在于,MT-DNN 的微调阶段引入了多任务学习机制,使用多个任务来微调共享的文本编码层和任务特定层的参数,这种微调的方法使得预训练模型能在更多的数据上进行训练,同时还能获得更好的泛化能力。

MT-DNN 具有良好的迁移能力,在训练数据很少的情况下,较 BERT 可以获得更好的性能。MT-DNN 可以较好地处理 BERT 在一些小数据集上微调可能存在无法收敛而表现很差的问题,同时节省新任务上标注数据和微调的成本。MT-DNN 可被看作一个集成学习的过程,因此可以用知识蒸馏[55]进行优化。采用知识蒸馏后,模型在 GLUE 中的表现有了明显提升,如表 4 所列。

BERT 主要通过词或句子的共现信号预训练语言模型。然而,除语言共现信息之外,语料中还包含词法、语法、语义等更多有价值的信息。基于此,百度团队提出可持续学习的语义理解框架 ERNIE2.0,在预训练阶段引入多任务学习机制,这也是对其早期发布的 ERNIE1.0 的改进。ERNIE2.0 框架支持增量地引入词汇、语法、语义多个层次的自定义预训练任务,能全面捕捉训练语料中的词法、语法、语义等潜在信息。在预训练阶段,通过交替学习这些不同种类的任务,对模型不断训练更新,这种连续交替的学习范式使模型不会忘记之前学到的语言知识,持续提升模型效果。

依托 ERNIE2.0 框架,百度团队充分借助飞桨多机分布式训练的优势,使用 79 亿词语的训练数据(约 1/4 的 XLNet 数据)和 64 张 V100(约 1/8 的 XLNet 算力)训练预训练模型。该团队还试验了此预训练语言模型在中英文领域的效果:在英文领域,ERNIE2.0 在 GLUE 的 7 个任务上的表现超越了 BERT 和 XLNet;在中文领域,其在包括阅读理解、情感分析、问答等 9 个不同类型的数据集上的表现超越了 BERT 并刷新了最佳成绩,如表 5 所列[54]。ERNIE2.0 的工作表明,在预训练阶段,通过构建多个训练任务可以显著提升模型效果。

表 5 MT-DNN 和 ERNIE2.0 在 GLUE 数据集上的实验结果

Table 5 MT-DNN and ERNIE2.0 results on GLUE

(单位:%)

Dataset	BERT	MT-DNN	ERNIE2.0
CoLA	60.5	62.5	63.5
SST-2	94.9	95.6	95.6
MRPC	89.3	91.1	90.2
STS-B	86.5	88.8	90.6
QQP	72.1	72.7	73.8
MNLI-m/mm	86.7/85.9	86.7/86.0	88.7/88.8
QNLI	91.1	93.1	94.6
RTE	70.1	81.4	80.2
WNLI	65.1	65.1	67.8

4.1.4 改进掩码方式

BERT 模型使用 Masked-LM 任务进行训练,按照字粒度进行掩码,这种掩码方式不利于学习到完整的词义表示。本

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

节介绍 BERT WWM(Whole Word Masking)系列模型[56]和 SpanBert[57]模型，它们改进了原生 BERT 模型的掩码方式，进一步提升了模型性能。

BERT WWM 是一种全词掩码方式，是谷歌发布的一项 BERT 的升级版本，主要更改了预训练阶段 Masked-LM 的掩码策略。BERT 采用 WordPiece 的分词方法，把一个完整的词切分成若干个子词。最初的 BERT 的掩码策略是随机掩码一个句子中的部分子词，而在全词掩码中，如果一个完整词的部分子词被掩码，则同属该词的其他部分也会被掩码。

这种全词掩码的策略使得预训练模型在训练 Masked-LM 的过程中将恢复整个词语作为训练目标，而不是仅恢复部分子词。全词掩码策略克服了原生 BERT 模型掩码部分子词的缺点，进一步提升了 BERT 模型的性能水平。谷歌现已发布了基于全词掩码方式训练好的预训练模型(BERT-large-wwm)。

此外，在中文领域，哈工大讯飞联合实验室发布了基于全词掩码的中文 BERT 预训练模型 BERT-wwm-ext[56]，其在多个中文数据集上取得了当前中文预训练模型的最佳水平，实验效果甚至超过了原生 BERT 和 ERINE 等中文预训练模型。

SpanBERT 是一个新的分词级别的预训练模型，能够对分词进行更好地表示和预测。该模型与 BERT 的差别主要体现在掩码机制和训练目标上。

在掩码机制方面，与 BERT 团队的全词掩码类似，SpanBERT 不是随机地对单个子词进行掩码，而是对随机的邻接分词添加掩码。每次掩码的过程是先从一个几何分布中采样得到需要掩码的分词的长度，并在此分词级别上进行掩码。

在训练目标方面，SpanBERT 提出了一个新的模型训练目标 Span Boundary Object(SBO)，通过使用分词边界的表示来预测被添加掩码的分词的内容，不再依赖分词内单个子词的表示。这种训练目标能使模型在边界词中存储其分词级别的信息，有助于模型的调优。

从实验效果来看，SpanBERT 在多个任务中的表现都超越了所有的 BERT 基线模型，且在问答任务、指代消解等分词选择类任务中均取得了重要的性能提升。特别地，在使用与 BERT 相同的训练数据和模型大小时，SpanBERT 在 SQuAD1.0 和 2.0 中的 $F1$ 值分别为 94.6%和 88.7%。此外，SpanBERT 在不涉及分词选择的任务中也取得了进展，在 GLUE 数据集上的表现亦有所提升[57]。

4.1.5 改进训练方法

本节主要介绍 RoBERTa[58]模型，该模型与 BERT 基本一致，改进之处在于设计了更加精细的训练方法，提高了模型性能。

RoBERTa 是 Facebook AI 联合 UW 发布的基于 BERT 改进的预训练模型，在模型结构层面上较 BERT 并没有较大的改变，其改进主要体现在以下 4 个预训练的方法：动态掩码机制、移除预测下一句的任务、更大的批大小、更多的数据和更长的训练时间。

BERT 的掩码机制是一种静态的掩码策略，对于每一个序列来说，掩码的词语一旦选定，在之后的整个训练过程中都不会再发生改变。而 RoBERTa 提出了一种动态的掩码机制，即一开始把预训练的数据复制 10 份，每一份都随机进行掩码，则同一个序列会有 10 种不同的掩码方式，因而在模型预训练的过程中，每个序列被掩码的词语是会变化的。RoBERTa 在只将静态掩码改成动态掩码而其他训练方法不变的情况下进行实验，结果表明动态掩码机制确实能提高性能[58]。

为了捕捉句子之间的关系，BERT 除了使用 Masked-LM 任务外，还使用了预测下一句的任务来预训练模型，在预训练阶段每次拼接两个句子作为输入数据。而 RoBERTa 在预训练阶段去除了预测下一句的任务，改为每次输入连续的多个句子，直到序列达到最大长度，这种训练方式也叫作全句模式。实验表明，在推断句子关系的任务上，RoBERTa 也能有更好的性能[58]。

RoBERTa 还在批大小上设计了实验探索：BERT 的批大小是 256，RoBERTa 探索了 2k 和 8k 的批大小。这一思想主要借鉴了在机器翻译中，使用更大的批大小并配合更大的学习率能加快模型优化速率并提升模型性能，对比实验也证明了更大的批大小可以给模型性能带来一定程度的提升。

借鉴 XLNet 用了比 Bert 多 10 倍的训练数据的思想，RoBERTa 也使用了更多的训练数据，同时需要训练更长的时间。从实验效果来看，更多的训练数据配合更长的训练时间，确实可以带来模型性能的提高。这种思路一定程度上与 GPT2.0[59]扩充数据的方法类似，需要消耗大量的计算资源。

RoBERTa 模型主要在以上 4 个方面对 BERT 进行精细调参，在 GLUE 上对比当时最先进的 XLNet 模型[61]，其在多个任务上获得了超越 XLNet 的表现，如表 6 所列。

表 6 RoBERTa 与 XLNet 在 GLUE 数据集上的实验结果

Table 6 RoBERTa and XLNet results on GLUE

(单位：%)

Dataset	XLNet	RoBERTa
CoLA	67.8	67.8
SST-2	96.8	96.7
MRPC	93.0	92.3
STS-B	91.6	92.2
QQP	90.3	90.2
MNLI-m/mm	90.2/89.8	90.8/90.2
QNLI	98.6	98.9
RTE	86.3	88.2
WNLI	90.4	89.0

4.2 XLNet 模型

BERT 是典型的自编码模型，旨在从引入噪声的数据中恢复出原数据。BERT 的预训练过程采用了降噪自编码思想，提出了 Masked-LM 预训练任务，该任务的最大贡献在于使模型获得了真正的双向上下文信息，但是也带来了一些问题：首先，预训练时使用的掩码机制在下游任务微调时并不会使用，导致训练和使用两个过程存在数据偏差，对实际效果有一定影响；其次，BERT 中每个单词的预测是相互独立的，而类似于“New York”这样的实体，“New”和“York”是存在关联的，这个假设忽略了这样的情况。

自回归模型一般不存在第二个问题，但传统的自回归模型本质上是单向的，无法建模双向信息。XLNet 的贡献在于

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

提出了一种可以获得真双向的上下文信息的自回归语言模型，进而避免了第一个问题。XLNet 主要使用 3 种机制来解决上述问题：排列语言模型、双流自注意力和循环机制。

排列语言模型是指预测某个单词时，XLNet 使用原始输入次序的随机排列来获取双向的上下文信息，同时维持自回归模型原有的单向形式。它采用了一种比较巧妙的实现方式：使用单词在排列中的位置计算上下文信息。如对于一个 2→4→3→1 的排列，单词 2 和单词 4 就可以作为上文的输入来预测单词 3。当原句的所有排列都取完时，就能获得所有的上下文信息。为了考虑位置因素对预测结果的影响，引入了要预测单词的位置信息。此外，为了降低模型的优化难度，XLNet 使用了部分预测的方式，最终优化目标如式(7)所示：

$$\max_{\theta} E_{z\sim Z_T}\left[\sum_{t=1}^{T}\log p_\theta(x_{z_t}\mid X_{z_{<t}})\right] \tag{7}$$

其中，Z_T 表示长度为 T 的序列的所有排列组成的集合，z 是一种排列方法，x_{z_t} 表示排列的第 t 个元素，$X_{z_{<t}}$ 表示排列的第 1 到第 $t-1$ 个元素。

双流自注意力机制要解决的问题是，当获得考虑了位置因素的向量表示后，只能获得该位置信息以及上文信息，不足以预测该位置后的单词；而原来的向量表示则因为获取不到位置信息，依然不足以预测该位置后的单词。因此，XLNet 引入了双流自注意力机制，将两者结合起来。

循环机制借鉴了 Transformer-XL[60] 的思想，即在处理下一个单词时结合上个单词的隐层表示，使得模型能够获得更长距离的上下文信息。XLNet 虽然在前端采用了相对位置编码，但在隐层表示时涉及到的处理与排列独立，因此还可以沿用这个循环机制。该机制使得 XLNet 在处理长文档时具有较好的优势。

相比 BERT，XLNet 采用自回归语言模型解决了单词之间预测不独立的问题，同时采用了排列语言模型等机制使自回归模型也可以获得真双向的上下文信息。XLNet 的最终结果与 BERT 进行了较为公平的比较，在模型的训练数据、超参数以及网格搜索空间等与 BERT 一致的情况下，使用单模型在 GLUE 的 dev 上进行对比实验。如表 7 所列，XLNet 的实验结果优于 BERT[10]。

表 7 BERT 与 XLNet 在 GLUE 数据集上的实验结果

Table 7 BERT and XLNet experimental results on GLUE

(单位：%)

Dataset	BERT	XLNet
CoLA	60.6	63.6
SST-2	93.2	95.6
MRPC	88.0	89.2
STS-B	90.0	91.8
QQP	91.3	91.8
MNLI-m/mm	86.6/—	89.8/—
QNLI	92.3	93.9
RTE	70.4	83.8
WNLI	—	—

4.3 小结

BERT 的出现开启了一个新时代，此后涌现出了大量的预训练语言模型。以上是依据模型结构，分为基于 BERT 的改进模型和 XLNet 进行的讨论。此外，预训练语言模型还可以从特征抽取、语言模型目标、特征表示 3 个方面进行划分。

特征抽取方面，主要分为 RNNs，Transformer 和 Transformer-XL 3 种。ELMo 和 ULMFiT 使用 RNNs 作为特征抽取器，自谷歌提出 Transformer 后，GPT 和 BERT 系列模型就使用 Transformer 的相关结构进行特征抽取，XLNet 则使用 Transformer-XL。

语言模型目标方面，主要分为自编码语言模型和自回归语言模型。BERT 系列模型均使用自编码语言模型和单向语言模型，包括 ELMo，ULMFiT，GPT 等；XLNet 则使用自回归语言模型。

特征表示方面，主要分为单向特征表示和双向特征表示。单向语言模型使用单向特征表示，BERT 系列模型和 XLNet 均使用真双向的特征表示。

5 面临的主要挑战

预训练模型自出现以来，就以绝对的优势取代了早期的传统词向量技术，并极大地推动了自然语言处理领域的研究进展。当前预训练模型虽然已经取得了很好的成果，但依然面临一些问题和挑战。

5.1 无法处理常识和推理问题

目前预训练模型在自然语言处理领域的大多数任务中都取得了耀眼的成绩，甚至有人认为预训练模型几乎解决了自然语言处理领域的问题。然而，Niven[61] 和 McCoy[62] 等的工作表明，到目前为止，预训练模型可能并没有学习到真正的语义信息，因而无法很好地处理常识和推理问题。而现实生活中，大多数的自然语言处理任务都需要具有一定的推理能力，或运用现实世界中的常识知识的能力。因此，如何改进预训练模型，使其能够较好地处理常识和推理任务是一个重要问题。

5.2 生成任务中表现逊色

预训练模型如 GPT 和 BERT 等即使刷新了多项自然语言处理领域的任务的最高纪录，但都没有处理文本生成问题。两者主要刷新的是 GLUE 榜单，其中 GPT 刷新了 9 项数据集纪录，BERT 更是刷新了 11 项数据集纪录，然而这些数据集全是自然语言理解领域的问题；模型在自然语言生成领域的效果没有通过实验得到验证，尤其是 BERT 在生成任务上的表现较为逊色，这可能是由于训练阶段采用的是真双向的语言模型，而在生成任务中无法提前看到下文。虽然模型 MASS 和 UNILM 在生成任务上的效果有所提升，但相比自然语言理解任务而言，预训练模型在生成任务上的表现依然逊色。因此，将预训练模型更好地应用于自然语言生成任务是一个需要解决的问题。

5.3 资源消耗过大

目前的自然语言处理方法大多缺乏坚实的理论基础。以 BERT 为代表的做法，实际上是用大数据、大模型和大计算量这种简单粗暴的方法来处理自然语言，而并非像人类一样能真正理解和灵活运用自然语言。主流的预训练模型的资源使用情况对比如表 8 所列，除早期的 ELMo 模型外，后续的预训练模型，包括 GPT，BERT，XLNet 等，在训练数据规模、模型参数量和计算资源使用量上都远远超于自然语言处理领域

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

的一般模型。这种靠大数据和大模型来比拼实验效果的做法,使得一般的科研人员由于缺乏计算力而无法训练一个完整的大规模预训练模型,这种过高的计算力门槛可能会丢失探索更多其他方法的可能。

表 8 预训练模型资源使用情况的对比

Table 8 Pre-training model resource usage comparison

模型	数据量/亿词	模型结构	参数量/M
ELMo	10	2 层 BiLSTM	90
GPT	8	12 层 Transformer	110
BERT	33	24 层 Transformer	340
XLNet	330	24 层 Transformer-XL	340

6 总结与展望

预训练模型在自然语言处理领域取得的优异成绩证明了其有效性,可以预见,未来的预训练模型将会进一步推动自然语言处理领域的快速发展。通过梳理近年来经典的预训练模型以及参考其他在经典模型上的改进工作,本文给出了预训练技术在未来的几个可能的发展方向。

6.1 如何处理常识和推理问题

如何有效地处理常识和推理问题是自然语言处理领域面临的一个关键问题。结合知识图谱的预训练模型已经被证明可以为知识驱动型任务带来性能提升。例如,百度的 ERNIE1.0 直接对语义知识单元进行建模,以增强模型的语义表示;清华大学的 ERNIE(THU)使用知识图谱中的多信息实体作为外部知识来改善语言表征。目前,预训练模型尚不能有效地解决常识和推理问题,其瓶颈主要体现在以下两个方面:(1)如何使预训练模型从大规模语料中学习到客观事实和常识信息;(2)如何使预训练模型能够利用知识库进行推理。ERNIE1.0 和 ERNIE(THU)的工作在解决以上问题方面具有启发意义,可以尝试结合知识图谱改进预训练模型,使模型对语义知识单元的表示更贴近真实世界,从而可以更好地处理常识和推理问题,这将是一个有价值的研究方向。

6.2 如何改善生成任务

针对如何改进预训练模型,使其能更好地处理生成任务的问题,现有模型 MASS 和 UNILM 进行了有益的探索,使其更适用于生成任务。从实验效果来看,基于 BERT 改进后的模型可以更好地处理生成任务,这也证明了将预训练模型迁移到生成任务的思路是可行的。此外,上述工作都是基于 BERT 进行的改进,预训练模型的框架本质上还是编码器-解码器框架,而未来这种框架是否要保留值得深入研究。如果可以将预训练模型的训练模式改造成与在生成任务中使用时一样的模式,统一生成任务框架,则可能会迎来一个阶段性的大一统时代:一个基本模型做到一个闭环,即从文本到语义表征,再从语义表征重新生成文本。这种尝试是值得期待的。

6.3 如何降低训练成本

如何以更少的代价,探索更小更快的预训练模型,是一个亟待解决的事情。目前经典的预训练模型都存在数据量过大、模型过大和计算力需求过大的问题,以至于大多数从业人员只能在公开发布的、已经训练好的预训练模型上进行微调,而没有足够的资源进行从头训练。这种过高的门槛使得很多科研人员无法对预训练技术提出新的方法,不利于整个领域的发展。此外,过大的资源消耗以及较低的效率问题,导致工业界无法更广泛地应用预训练技术,例如其无法应用于移动设备等。因此,如何降低预训练模型的训练成本,在损失尽可能少的精度的情况下探索更小、更快的预训练模型,可能是未来的一个发展方向。Hugging Face 发布的 DistilBERT[63] 进行了相关的探索,在仅使用一半参数的情况下,保留了 95% 的性能,证实了探索更小、更快的预训练模型是可行的。近期发布的超小型的 BERT——ALBERT_TINY[64-65] 为探索小型化的预训练模型提供了启示,其训练和推理预测速度相比 BERT 提升 10 倍,精度基本保留,模型大小仅为 BERT 的 1/25。能否在模型体量、性能和效率等方面进行更深入地探索,将是未来的一个发展方向。

结束语 本文主要概述了面向自然语言处理领域的预训练技术及其发展历史。以 BERT 为分界点,预训练技术的发展历史大致可以分为 3 个阶段:早期的静态预训练技术、经典的动态预训练技术以及新式的动态预训练技术。早期的静态预训练技术主要是以 Word2Vec 为代表的词向量技术;经典的动态预训练技术主要是 ELMo,GPT 和 BERT 等;新式的动态预训练技术主要包括基于 BERT 的改进模型和 XLNet。

目前,预训练技术已经在自然语言处理领域取得了很大进展,但同时也面临诸多挑战:无法处理常识和推理问题,在生成任务中表现逊色,以及资源消耗过大等。未来预训练技术可能会致力于解决上述问题,重点研究如何处理常识和推理问题,如何改善生成任务,以及如何降低训练成本等。

参 考 文 献

[1] HE K,ZHANG X,REN S,et al. Deep residual learning for image recognition[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016:770-778.

[2] MIKOLOV T,CHEN K,CORRADO G,et al. Efficient estimation of word representations in vector space[J]. arXiv:1301.3781.

[3] MIKOLOV T,SUTSKEVER I,CHEN K,et al. Distributed representations of words and phrases and their compositionality[C]// Advances in Neural Information Processing Systems. 2013:3111-3119.

[4] ABADI M,BARHAM P,CHEN J,et al. Tensorflow:a system for large-scale machine learning[J]. arXiv:1605.08695.

[5] LE Q,MIKOLOV T. Distributed representations of sentences and documents[C]// International Conference on Machine Learning. 2014:1188-1196.

[6] DENG L,YU D. Deep learning:methods and applications[J]. Foundations and Trends in Signal Processing,2014,7(3/4):197-387.

[7] PETERS M E,NEUMANN M,IYYER M,et al. Deep contextualized word representations[J]. arXiv:1802.05365.

[8] RADFORD A,NARASIMHAN K,SALIMANS T,et al. Improving language understanding by generative pre-training[J/OL]. https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/languageunsupervised/language understanding paper.pdf,2018.

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

[9] DEVLIN J, CHANG M W, LEE K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv: 1810.04805.

[10] YANG Z, DAI Z, YANG Y, et al. XLNet: Generalized Autoregressive Pretraining for Language Understanding[J]. arXiv: 1906.08237.

[11] YOSINSKI J, CLUNE J, BENGIO Y, et al. How transferable are features in deep neural networks?[C]// Advances in Neural Information Processing Systems. 2014: 3320-3328.

[12] OQUAB M, BOTTOU L, LAPTEV I, et al. Learning and transferring mid-level image representations using convolutional neural networks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 1717-1724.

[13] GLOROT X, BORDES A, BENGIO Y. Domain adaptation for large-scale sentiment classification: A deep learning approach [C]// Proceedings of the 28th International Conference on Machine Learning (ICML-11). 2011: 513-520.

[14] CHEN M, XU Z, WEINBERGER K, et al. Marginalized denoising autoencoders for domain adaptation[J]. arXiv: 1206.4683.

[15] GANIN Y, USTINOVA E, AJAKAN H, et al. Domain-adversarial training of neural networks[J]. The Journal of Machine Learning Research, 2016, 17(1): 2096-2030.

[16] SZEGEDY C, IOFFE S, VANHOUCKE V, et al. Inception-v4, inception-resnet and the impact of residual connections on learning[C]// AAAI. 2017: 12.

[17] WU Z, SHEN C, HENGEL A V D. Wider or Deeper: Revisiting the ResNet Model for Visual Recognition[J]. arXiv: 1611.10080.

[18] SINGH S, HOIEM D, FORSYTH D. Swapout: Learning an ensemble of deep architectures[C]// Advances in Neural Information Processing Systems. 2016: 28-36.

[19] REN S, HE K, GIRSHICK R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]// Advances in Neural Information Processing Systems. 2015: 91-99.

[20] HUANG G, LIU Z, VAN DER MAATEN L, et al. Densely connected convolutional networks[J]. arXiv: 1608.06993.

[21] HE K, ZHANG X, REN S, et al. Identity mappings in deep residual networks[C]// European Conference on Computer Vision. Cham: Springer, 2016: 630-645.

[22] LEDIG C, THEIS L, HUSZÁR F, et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network[J]. arXiv: 1609.04802.

[23] PETERS M, AMMAR W, BHAGAVATULA C, et al. Semi-supervised sequence tagging with bidirectional language models [C]// Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. 2017: 1756-1765.

[24] KIROS R, ZHU Y, SALAKHUTDINOV R R, et al. Skip-thought vectors[C]// Advances in Neural Information Processing Systems. 2015: 3294-3302.

[25] VINCENT P, LAROCHELLE H, BENGIO Y, et al. Extracting and composing robust features with denoising autoencoders [C]// Proceedings of the 25th International Conference on Machine Learning. ACM, 2008: 1096-1103.

[26] BENGIO Y, DUCHARME R, VINCENT P, et al. A neural probabilistic language model[J]. Journal of Machine Learning Research, 2003, 3(6): 1137-1155.

[27] PENNINGTON J, SOCHER R, MANNING C. Glove: Global vectors for word representation[C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014: 1532-1543.

[28] JOULIN A, GRAVE E, BOJANOWSKI P, et al. Bag of Tricks for Efficient Text Classification[J]. arXiv: 1607.01759.

[29] CHEN D, MANNING C. A fast and accurate dependency parser using neural networks[C]// Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014: 740-750.

[30] BORDES A, USUNIER N, GARCIA-DURAN A, et al. Translating embeddings for modeling multi-relational data[C]// Advances in Neural Information Processing Systems. 2013: 2787-2795.

[31] TAI K S, SOCHER R, MANNING C D. Improved semantic representations from tree-structured long short-term memory networks[J]. arXiv: 1503.00075.

[32] GROVER A, LESKOVEC J. node2vec: Scalable feature learning for networks[C]// Proceedings of the 22nd ACM SIGKDD international Conference on Knowledge Discovery and Data Mining. ACM, 2016: 855-864.

[33] TANG J, QU M, WANG M, et al. Line: Large-scale information network embedding[C]// Proceedings of the 24th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee. 2015: 1067-1077.

[34] NICKEL M, KIELA D. Poincaré embeddings for learning hierarchical representations[C]// Advances in Neural Information Processing Systems. 2017: 6338-6347.

[35] KAHNG M, ANDREWS P Y, KALRO A, et al. A cti v is: Visual exploration of industry-scale deep neural network models [J]. IEEE Transactions on Visualization and Computer Graphics, 2018, 24(1): 88-97.

[36] YANG X, MACDONALD C, OUNIS I. Using word embeddings in twitter election classification[J]. Information Retrieval Journal, 2018, 21(2/3): 183-207.

[37] MNIH A, HINTON G. Three new graphical models for statistical language modelling[C]// Proceedings of the 24th International Conference on Machine Learning. ACM, 2007: 641-648.

[38] MNIH A, HINTON G E. A scalable hierarchical distributed language model[C]// Advances in Neural Information Processing Systems. 2009: 1081-1088.

[39] COLLOBERT R, WESTON J, BOTTOU L, et al. Natural language processing (almost) from scratch[J]. Journal of Machine Learning Research, 2011, 12(1): 2493-2537.

[40] MIKOLOV T, KARAFIÁT M, BURGET L, et al. Recurrent neural network based language model[C]// Eleventh Annual Conference of the International Speech Communication Association. 2010.

[41] GUTMANN M U, HYVÄRINEN A. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics[J]. Journal of Machine Learning Research, 2012, 13: 307-361.

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

[42] DEERWESTER S,DUMAIS S T,FURNAS G W,et al. Indexing by latent semantic analysis[J]. Journal of the American Society for Information Science,1990,41(6):391-407.
[43] GOLUB G H,REINSCH C. Singular value decomposition and least squares solutions[M]// Linear Algebra. Berlin:Springer,1971:134-151.
[44] HARRIS Z S. Distributional structure[J]. Word,1954,10(2/3):146-162.
[45] JOZEFOWICZ R,VINYALS O,SCHUSTER M,et al. Exploring the limits of language modeling[J]. arXiv:1602.02410.
[46] HOWARD J,RUDER S. Universal language model fine-tuning for text classification[J]. arXiv:1801.06146.
[47] VASWANI A,SHAZEER N,PARMAR N,et al. Attention is all you need[C]// Advances in Neural Information Processing Systems. 2017:5998-6008.
[48] LIU P J,SALEH M,POT E,et al. Generating wikipedia by summarizing long sequences[J]. arXiv:1801.10198.
[49] SONG K,TAN X,QIN T,et al. Mass:Masked sequence to sequence pre-training for language generation[J]. arXiv:1905.02450.
[50] DONG L,YANG N,WANG W,et al. Unified Language Model Pre-training for Natural Language Understanding and Generation[J]. arXiv:1905.03197.
[51] SUN Y,WANG S,LI Y,et al. ERNIE:Enhanced Representation through Knowledge Integration[J]. arXiv:1904.09223.
[52] ZHANG Z,HAN X,LIU Z,et al. ERNIE:Enhanced Language Representation with Informative Entities [J]. arXiv:1905.07129.
[53] LIU X,HE P,CHEN W,et al. Multi-task deep neural networks for natural language understanding[J]. arXiv:1901.11504.
[54] SUN Y,WANG S,LI Y,et al. Ernie 2.0:A continual pre-training framework for language understanding[J]. arXiv:1907.12412.
[55] HINTON G,VINYALS O,DEAN J. Distilling the knowledge in a neural network[J]. arXiv:1503.02531.
[56] CUI Y,CHE W,LIU T,et al. Pre-Training with Whole Word Masking for Chinese BERT[J]. arXiv:1906.08101.
[57] JOSHI M,CHEN D,LIU Y,et al. SpanBERT:Improving pre-training by representing and predicting spans[J]. arXiv:1907.10529.
[58] LIU Y,OTT M,GOYAL N,et al. Roberta:A robustly optimized BERT pretraining approach[J]. arXiv:1907.11692.
[59] RADFORD A,WU J,CHILD R,et al. Language models are unsupervised multitask learners[J]. OpenAI Blog,2019,1(8).
[60] DAI Z,YANG Z,YANG Y,et al. Transformer-xl:Attentive language models beyond a fixed-length context[J]. arXiv:1901.02860.
[61] NIVEN T,KAO H Y. Probing neural network comprehension of natural language arguments[J]. arXiv:1907.07355.
[62] MCCOY R T,PAVLICK E,LINZEN T. Right for the Wrong Reasons:Diagnosing Syntactic Heuristics in Natural Language Inference[J]. arXiv:1902.01007.
[63] WOLF T,DEBUT L,SANH V,et al. Transformers:State-of-the-art Natural Language Processing[J]. arXiv:1910.03771.
[64] Bright. GitHub repository [OL]. https://github.com/bright-mart/albert_zh.
[65] LAN Z,CHEN M,GOODMAN S,et al. ALBERT:A Lite BERT for Self-supervised Learning of Language Representations[J]. arXiv:1909.11942.

LI Zhou-jun, born in 1963, is a professor and doctoral tutor of Beihang University of Computer. He is currently a member of the Network Space Security Discipline Review Group of the Academic Degrees Committee of the State Council, the executive director of the China Cyberspace Security Association, the deputy director of the Language Intelligence Committee of the China Artificial Intelligence Society, and a member of the ACM, IEEE, and AAAI. He is mainly engaged in the research of artificial intelligence and natural language processing such as intelligent question and answer, semantic analysis, information extraction and OCR. He has published more than 300 academic papers in SCI journals including TKDE, TIFS and other top international conferences such as AAAI, IJCAI, ACL, EMNLP, and won the ECIR 2010 Best Paper Award. The team he directed has won several championships in artificial intelligence and cybersecurity competitions at home and abroad.

(C)1994-2022 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net