面向自然语言处理的深度学习研究

奚雪峰 [1, 2, 3] 周国栋 [1]

摘 要 近年来, 深度学习在图像和语音处理领域已经取得显著进展, 但是在同属人类认知范畴的自然语言处理任务中, 研究还未取得重大突破. 本文首先从深度学习的应用动机、首要任务及基本框架等角度介绍了深度学习的基本概念; 其次, 围绕数据表示和学习模型两方面, 重点分析讨论了当前面向自然语言处理的深度学习研究进展及其应用策略; 并进一步介绍了已有的深度学习平台和工具; 最后, 对深度学习在自然语言处理领域的发展趋势和有待深入研究的难点进行了展望.

关键词 自然语言处理, 深度学习, 表示学习, 特征学习, 神经网络

引用格式 奚雪峰, 周国栋. 面向自然语言处理的深度学习研究. 自动化学报, 2016, **42**(10): 1445−1465

DOI 10.16383/j.aas.2016.c150682

A Survey on Deep Learning for Natural Language Processing

XI Xue-Feng[1, 2, 3] ZHOU Guo-Dong[1]

Abstract Recently, deep learning has made significant development in the fields of image and voice processing. However, there is no major breakthrough in natural language processing task which belongs to the same category of human cognition. In this paper, firstly the basic concepts of deep learning are introduced, such as application motivation, primary task and basic framework. Secondly, in terms of both data representation and learning model, this paper focuses on the current research progress and application strategies of deep learning for natural language processing, and further describes the current deep learning platforms and tools. Finally, the future development difficulties and suggestions for possible extensions are also discussed.

Key words Natural language processing, deep learning, representation learning, feature learning, neural network

Citation Xi Xue-Feng, Zhou Guo-Dong. A survey on deep learning for natural language processing. *Acta Automatica Sinica*, 2016, **42**(10): 1445−1465

深度学习 (Deep learning) 通过建立深层神经网络, 模拟人脑的机制进行解释并分析学习图像、语音及文本等数据, 是目前机器学习研究中的一个热点领域. 传统机器学习工作的有效性, 很大程度上依赖于人工设计的数据表示和输入特征的有效性; 机器学习方法在这个过程中的作用仅仅是优化学习权重以便最终输出最优的学习结果. 与传统机器学习方法不同的是, 深度学习试图自动完成数据表示和特征提取工作; 并且深度学习更强调, 通过学习过程提取出不同水平、不同维度的有效表示, 以便提高不同抽象层次上对数据的解释能力. 从认知科学角度来看, 这个思路与人类学习机理非常吻合.

在面对大量感知数据的处理过程中, 人脑对其中的重要信息有着特殊的敏感性. 例如即使是四岁孩童, 放学时间站在校门口观望大量的接送家长, 总是比较容易快速准确地发现家人熟悉的身影, 欣喜地扑进家人的怀抱. 因此, 在人工智能研究领域中, 对于如何模仿人脑开展高效的复杂数据处理, 引发了研究者的极大兴趣. 其中, 从仿生学角度开展的人脑生理结构研究, 以及从人脑应用角度开展的功能研究, 是两个典型的研究方向. 前者体现研究对象的结构特征, 后者体现研究对象的功能特征. 两类研究又是互相渗透, 相互支撑. 例如, 在对哺乳类动物开展的解剖研究中发现, 大脑皮质存在着层次化的系列区域; 在此基础上, 神经科学研究人员又通过测试视觉信号输入人脑视网膜后经大脑前额皮质层到达运动神经的时间, 推断发现大脑皮质层的主要功能在于将视觉信号通过复杂的多层网络模型后加以提取观测信息, 而并未直接对视觉信号进行特征处理. 这就说明, 人脑在识别物体过程中, 并未直接通过视网膜投影的外部世界进行感知, 而是需要依靠经过某种聚集和分解处理后的信息才能识别得到物体.

收稿日期 2015-11-02 录用日期 2016-06-12
Manuscript received November 2, 2015; accepted June 12, 2016
国家自然科学基金 (61331011, 61472264) 资助
Supported by National Natural Science Foundation of China (61331011, 61472264)
本文责任编委 柯登峰
Recommended by Associate Editor KE Deng-Feng
1. 苏州大学计算机科学与技术学院 苏州 215006 2. 苏州科技学院电子与信息工程学院 苏州 215009 3. 苏州市移动网络技术与应用重点实验室 苏州 215009
1. School of Computer Science and Technology, Soochow University, Suzhou 215006 2. School of Electronic and Information Engineering, Suzhou University of Science and Technology, Suzhou 215009 3. Suzhou Key Laboratory of Mobile Networking and Applied Technologies, Suzhou 215009

这一过程中, 视皮层的功能主要是开展对视觉信号的特征提取和计算, 而非简单重现视网膜图像. 这种具有明确层次结构的人类视觉感知系统在大大降低了视觉感知处理数据量的同时, 还能够保留被感知物体关键的结构信息. 大脑这种分层次结构启发了研究人员开展多层次神经网络的研究. 最早出现的多层网络训练算法是采用初始值随机选定及梯度下降优化策略的 BP (Back-propagation) 神经网络. 但是这种多层结构的主要缺陷在于输入与输出间存在的非线性映射导致能量函数或网络误差函数空间含有多个局部极小点, 同时采用的又是使能量或误差单一减小的搜索方向, 容易导致局部收敛最小而非全局最优. 相关实验及理论[1−2] 发现, 局部收敛最优的情况会随着网络层数的增加而变得越来越严重, 似乎表明 BP 算法在向多层深度结构方向发展上并无优势可言, 这在一定程度上影响了深度学习的发展.

浅层学习结构的共同特点是仅含一种将单个原始输入信号映射到特定问题空间的简单特征结构, 基本上可以认为这类模型带有一层或没有隐层节点. 常见的此类结构有条件随机场 (Conditional random field, CRF)、隐马尔科夫模型 (Hidden Markov model, HMM)、支持向量机 (Support vector machine, SVM)、多层感知器 (Multilayer perceptron, MLP) 及最大熵模型 (Maximum entropy, ME) 等. 这些模型大多应用在传统信号处理技术及机器学习研究中, 存在着对复杂函数表示能力有限、对复杂问题泛化处理能力不足的局限性[3].

这种情况直到 2006 年才出现转机. Hinton 等利用深度可信网络 (Deep belief network, DBN) 结构[4], 对组成 DBN 的每一层受限玻尔兹曼机 (Restricted Boltzmann machine, RBM) 结构进行无监督学习训练, 并将其用于 MNIST[1] 手写数字识别任务中, 取得了错误率仅为 1.2 % 的最好成绩[5]. 不久之后, Bengio 等也提出了一种基于自动编码器 (Auto-encoders) 的相关算法, 同样取得了较好结果[6−7]. 这些算法尽管形式不同, 但他们都遵循相同的原理: 即在每一层局部使用无监督的训练算法, 以引导完成特征中间表示层的训练目标. 此后, 其他一些非 RBM 或非 Auto-encoders 结构的深度学习算法也陆续提出[8−9]. 自 2006 年以来, 这些深度学习方法不仅在分类任务上取得显著结果[6,10−15], 而且在时序预测[16−17]、高维降秩[18−19]、纹理建模[20−21]、运动建模[22−23]、对象分割[24−25]、信息抽取[26−27] 及自然语言处理领域[28−30] 都有不俗表现. 此外, 尽管上述深度模型中, 普遍采用 Auto-encoders、RBM 和 DBN 结构, 能够以无监督的方式从未标注数据中学习到良好的结果, 但在面对特定任务领域时, 有监督反馈算法用来初始化深度结构的方式也有成功应用.

尽管当前深度学习还未有完备的理论体系支撑, 但并不妨碍在图像识别和语音识别等应用领域率先结出累累硕果. 2012 年, 一种称为 “深度神经网络 (Deep neural network, DNN)” 的机器学习模型在图像识别领域的 ImageNet 评测上被采用, 把识别错误率从 26 % 降到 15 %, 是图像识别领域近年来的最好结果. 而在此之前的 2011 年, 同样类似的 DNN 技术在语音识别领域也取得惊人效果, 降低语音识别错误率达 20 %～30 %, 从而大大推进了应用技术产品的开发. 比如基于 DNN 技术的微软全自动同声传译系统, 在 2012 年 11 月中国天津的一次公开活动中流畅地实现了自动语音识别、英文到中文的机器翻译以及合成中文语音输出的整个过程, 效果震惊全场.

尽管深度学习已经在上述图像和语音处理领域取得显著进展, 但是在同属人类认知范畴的自然语言处理任务中, 应用还未有重大突破. 本文重点分析了当前面向自然语言处理的深度学习研究进展, 并探讨了深度学习在自然语言处理领域的可能发展空间, 以图抛砖引玉. 下文第 1 节描述深度学习的基本概念; 第 2 节围绕数据表示和学习模型两方面, 重点分析讨论了当前深度学习在自然语言处理领域的研究现状、应用策略及其平台工具; 第 3 节对有待深入研究的难点和发展趋势进行展望, 最后是结束语.

1 深度学习概述

1.1 深度结构

与传统浅层学习的不同之处在于, 首先, 深度学习要求模型结构必须具有足够的深度 (Depth), 通常要求具有 3 层以上的隐层节点, 有的甚至可能达到 10 多层. 这种多层非线性映射结构, 有助于完成复杂函数逼近. 其次, 深度学习特别强调特征学习的重要性. 通过非监督预训练算法, 将输入原始样本在原空间的特征, 逐层变化, 映射到一个新的特征空间, 进而有可能使用新特征更加容易实现分类或预测. 此外, 生成性预训练方法也避免了因为网络函数表达能力过强而可能出现的过拟合 (Overfitting) 问题.

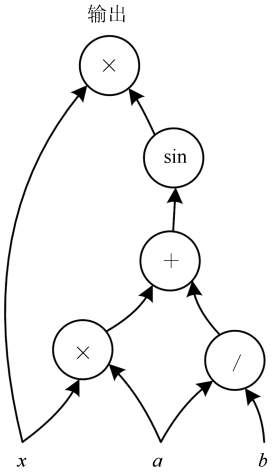
深度学习中深度的概念, 实际上来源于流图 (Flow graph) 的属性表示. 如图 1 (a) 所示, 流图可用于表示一个输入输出过程中所涉及的计算. 图中节点表示基本计算方法. 原始输入经过节点计算后生成的结果, 作为下一个节点的输入, 逐步计算传

[1] MNIST 是一个包含手写数字图片的数据集 http://yann.lecun.com/exdb/mnist/

递.

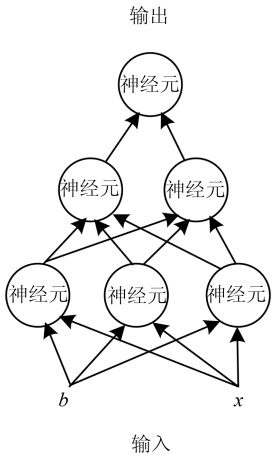
定义 1 (流图深度). 从一个输入到一个输出的最长路径长度, 即为流图的深度.

图 1 (a) 所示流图表示计算函数: $f(x) = x \times \sin(x \times a + a/b)$, 该结构具有深度 4. 图 1 (b) 所示多层人工神经网络 (Artificial neural network, ANN) 表示计算函数 $f(\boldsymbol{x}) = \tanh(b + w'\boldsymbol{x})$, 该结构具有深度 3. 对于输出层而言, 传统 BP 神经网络的深度一般定义为隐层数加 1, 如图 1 (c) 的结构具有深度 2. 深度神经网络则可能有更高深度 (大于或等于 3) 的结构.



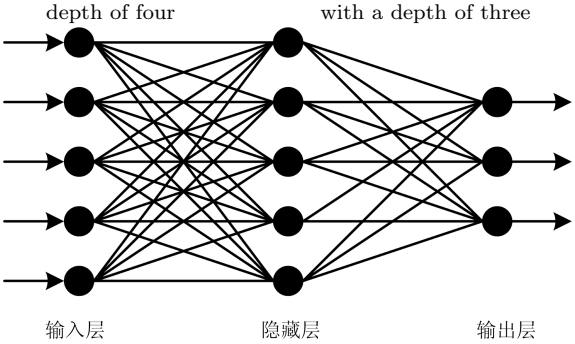
(a) 深度为 4 的流图

(a) A flow graph with depth of four



(b) 深度为 3 的多层神经网络

(b) A multilayer neural network with a depth of three



(c) 深度为 2 的 BP 神经网络

(c) A BP neural network with a depth of two

图 1　深度的概念示例图

Fig. 1　Concept example of depth

我们可以将深度结构看作一种因子分解. 大部分随机选择的函数, 通常都很难采用网络结构有效表示; 但是相对而言, 深度结构表示的有效性要高于浅层结构. 研究人员猜测, 这些可被深度结构但不能被浅层结构高效表示的函数中, 可能存在某种结构使得其能够被深层结构很好地泛化表示.

1.2　应用动机

采用特征来表示待处理问题中的对象, 是所有应用任务的首要工作. 比如在处理文本分类时, 经常用词集合特征来表示文档, 之后采用不同的分类算法来实现分类. 类似的, 在图像处理任务中, 最为普遍的就是把图像用像素集合特征加以表示. 选取不同的特征对任务的最终结果影响较大. 因此, 在解决实际问题时, 如何选取合适的特征非常重要.

对于很多训练任务来说, 特征具有天然的层次结构. 在语音、图像、文本处理任务中, 处理对象的层次结构如表 1 所示.

以图像识别为例. 最初的原始输入是图像的像素, 之后众多相邻像素可以组成线条, 多个线条组成纹理, 并进一步形成图案; 局部图案又构成了整个物体. 不难发现, 原始输入和浅层特征之间的联系较容易找到. 那么, 在此基础上, 能否通过中间层特征, 逐步获取原始输入与高层特征的联系呢? Olshausen 等的实验通过有效的特征提取, 将像素抽象成更高级的特征, 证实了这一设想的可能性[31]. 类似的结果也适用于语音特征.

传统机器学习方法过分依赖人工选取特征或表示, 不具备从数据中自动抽取和组织信息的能力. 尽管人工选择能够利用人类智慧和先验知识弥补这一缺陷, 但要达到能够深入理解问题的程度, 并挖掘合适的特征规则, 研究人员所需花费的时间代价也颇为昂贵. 这从某种程度上限制了机器学习向更聪明的人工智能方向迈进的步伐. 因此, 摆脱人工特征选择的局限性, 试图从大量可观测到的浅层感官数据中识别或解释关键特征, 便成为深度学习的主要思想, 这也是深度学习称为无监督特征学习的原因. 某种意义上, 凡是能够实现自动学习特征的方法, 都可以归为深度学习.

为什么深度学习方法可以实现自动学习特征呢? Hinton 等[3−4] 从不同角度探讨了可能的原因.

表 1　语音、图像、文本领域的特征层次结构[32]

Table 1　Feature hierarchy of speech, image and text[32]

任务领域	原始输入	浅层特征		中间特征		高层特征		训练目标
语音	样本	频段	声音	音调	音素	–	单词	语音识别
图像	像素	线条	纹理	图案	局部	–	物体	图像识别
文本	字母	单词	词组	短语	句子	段落	文章	语义理解

首先, 如果表示的深度不够, 就可能无法有效表示特征对象. 通常情况下, 一个给定目标精度的函数采用深度为 2 的网络结构就可以了, 如使用逻辑门. 但伴随而来的问题是需要大量计算节点. Hastad 从理论上证实了存在这样一类函数族[33], 即使用深度为 d 的结构和 $\mathrm{O}(n)$ 个节点可以有效表示的函数族, 当深度降低为 $d-1$ 时, 节点数呈现 $\mathrm{O}(2^n)$ 指数级增长, 这意味着增加表示深度的方式可以更加节约计算成本.

其次, 深度学习的分层概念符合人类认知学习过程. 从认知科学角度来看, 人类的认知学习过程是分层进行的, 分层结构是认知学习的基本要求. 例如工程师在解决复杂问题的过程中, 必定会将任务加以分解, 形成多个较小的子任务来处理, 子任务和总任务也处于不同的认知抽象层面.

最后, 神经生物学的研究表明, 人脑中也存在某种分层结构, 这进一步从仿生学角度为深度学习的有效性提供了佐证. 神经生物学家 Serre 等对人类大脑的研究表明[34]: 大脑皮质存在着层次化的系列区域; 每个区域都包含一个不同抽象层次的输入及到另一个区域的信号流向.

1.3 首要任务

深度学习的首要任务是尽可能采用一种简单的算法来实现所求解问题的分层特征表示. 经过特征的逐层变换, 使得原始样本特征可以映射变换到另一个新特征空间, 进而可以更加容易地利用特征完成分类或预测任务. 因此, 特别强调特征学习 (Feature learning) 或表示学习 (Representation learning) 的重要性, 这一点与传统机器学习方法是一致的, 所不同的是, 深度学习实现特征自动提取, 而传统机器学习更依赖于人工分析特征.

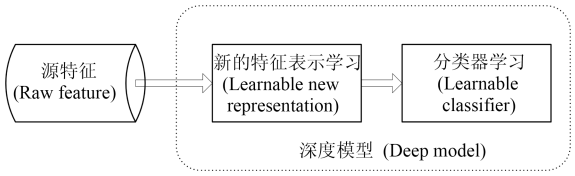


图 2 深度学习基本模型

Fig. 2 Basic model of deep learning

深度学习通过学习数据的某种变换形式, 当构建分类器或预测器时, 更容易抽取有效信息. 以概率模型为例, 能够抓取到所观察输入数据潜在解释因素后验分布的那个表示, 往往是一种好的表示形式. 在以深度学习方法为主的特征学习研究中, 还有许多问题有待进一步探索解决. 比如说, 一个特征表示优于另一个表示的主要因素是什么? 给定一个表示对象, 我们如何学习好的特征表示? $\cdots$ 诸如此类基本问题, 都有待研究解决.

1.4 基本框架

上节已经提到, 深度学习的首要任务其实是特征学习. 如图 2 所示, 深度学习模型本质上是一种基于原始特征 (或者说是未经过人类思维分析的数据) 输入, 通过多层非线性处理, 来学习复杂特征表示的方法. 如果结合特定的领域任务, 则深度学习可以通过自动学习的特征表示来构建新型分类器或生成工具, 以实现面向领域的分类或其他任务.

具体而言, 图 3 表示了深度学习的基本框架[35], 算法流程如下所示.

步骤 1. 随机初始化构建一个学习网络; 设置训练网络层数 n;

步骤 2. 初始化无标注数据作为网络训练输入集; 初始化训练网络层 $i=1$;

步骤 3. 基于输入集, 采用无监督学习算法预训练当前层的学习网络;

步骤 4. 每层的网络训练结果作为下一层的输入, 再次构建输入集;

步骤 5. 如果 i 小于网络层数 n, 则网络训练层 $i=i+1$, 算法跳转到步骤 3; 否则, 跳转到步骤 6;

步骤 6. 采用有监督学习方法来调整所有层的网络参数, 使误差达到要求;

步骤 7. 完成分类器 (如神经网络分类器) 构建; 或者完成深度生成模型 (如深度玻尔兹曼机) 构建.

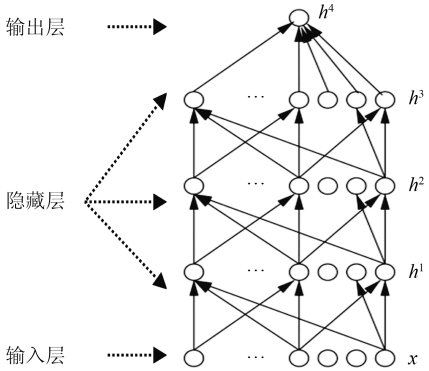


图 3 深度学习基本框架

Fig. 3 Basic framework of deep learning

上述基本框架中的步骤 2$\sim$4 是深度学习的关键, 也称为 "逐层预训练 (Layer-wise pre-training)" [5]. 如图 4 所示.

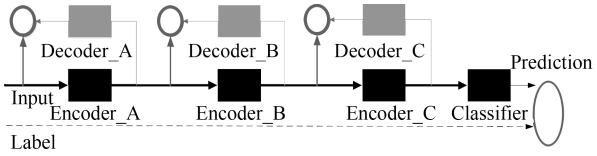


图 4 逐层预训练模型

Fig. 4 Layer-wise pre-training model

逐层训练中的关键部分是自动编码器 (Autoencoder) 的构建. 在深度学习模型中, 自动编码器可以是一种尽可能重现输入信号的神经网络.

1.4.1 无监督构建自动编码器

当原始输入确定后, 首先训练模型的第一层, 如图 4 中最左侧的黑色框图 Encoder_A, 表示编码器, 是整个模型的 "认知机构", 其将原始输入编码后形成第一层初级特征. 为了验证编码后的特征确实是原始输入的一种等价抽象表示, 没有丢失太多信息, 我们引入一个对应的解码器, 如图 4 中最左侧的灰色框图 Decoder_A, 它是这个模型的 "生成机构". 为了使 "认知" 和 "生成" 达成一致, 我们需要将编码后的特征经过解码器再生成, 目的是要与初始的原始输入做比较验证. 验证得到的结果误差定义为代价函数, 用于训练神经网络编码器和解码器. 当训练达到收敛目标后, 确定了具体各类参数的神经网络编码器就是我们需要的第一层模型 (而解码器可以不需要), 即可以得到原始数据的第一层抽象表示. 固定第一层神经网络编码器的参数, 并将第一层抽象输出作为输入, 再次重复操作, 陆续可以训练出第二层模型、第三层模型; 以此类推, 直至训练得到满足要求的最高层模型.

1.4.2 有监督训练分类器

通过上述训练后得到的自动编码器, 原始输入信号得到了不同的表达特征, 这些特征可以最大程度上代表原始输入信号. 但是, 这个自动编码器还不能用来实现分类功能. 为了实现分类, 我们需要在自动编码器最高层的编码层添加分类器 (Classifier), 结合标签 (Label) 样本, 基于标准神经网络的有监督训练方法调整参数.

参数调整方法分为两类: 一是仅仅调整最高层的分类器的参数; 二是通过标签样本, 调整所有自动编码器的参数, 也即实现对多层模型参数的精细调整.

深度学习所构建的深层模型具有较多局部最优解. 逐层初始化方法的目的就是最终将深层模型调整到较为接近全局最优解的位置, 从而获得最佳效果. 表 2 从不同角度比较了深层模型和浅层模型的特点. 浅层模型的一个主要局限性就是需要依赖人工经验来抽取作为模型输入的样本特征, 模型本身仅作为分类或预测工具. 因此在浅层模型实现的系统中, 起决定性作用的往往不是模型的优劣, 而是所选取的特征的优劣. 这也促使研究人员将研究精力重点投入到特征的开发和筛选中, 不仅对任务问题领域需要深刻的理解, 还需要花费大量时间反复实验摸索. 事实上, 逐层初始化深层模型也可以看作是特征学习的过程, 通过隐藏层对原始输入的一步一步抽象表示, 来学习原始输入的数据结构, 找到更有效的特征, 最终提高分类问题的准确性. 在获得有效特征之后, 模型整体训练也可以水到渠成.

2 面向自然语言处理的深度学习研究及应用

深度学习在图像和语音领域取得了突出成果, 但是在自然语言处理上还未取得重大突破. 与语音和图像不同, 语言是一种经过人类大脑产生并加工处理的符号系统, 似乎模仿人脑结构的人工神经网络应该在自然语言处理领域拥有更多优势, 但实际情况并非如此. 同时, 近几十年来, 在基于统计的模型成为自然语言处理主流方法之后, 属于统计方法典型代表的人工神经网络在自然语言处理领域依然没有得到足够重视. 当然, 这一切在 2006 年 Hinton 等提出深度学习[5] 以后, 情况发生了变化, 当前结合深度学习模型开展自然语言处理相关应用已经取得了一定成果, 并成为研究热点之一.

语言模型是最早采用神经网络开展研究的自然语言处理问题. 2003 年, Bengio 等提出词向量 (Word embedding 或 Word representation) 方法,

表 2 浅层和深层模型比对分析[32]

Table 2 Comparison and analysis of shallow model and deep model[32]

模型	浅层模型	深层模型
理论	有成熟的理论基础	理论分析困难
模型层数	1～2 层	5～10 层
训练难度	容易	复杂, 需要较多技巧
数据需求	仅需要简单特征的任务, 如发电机故障诊断、时间序列处理等	需要高度抽象特征的任务, 如语音识别、图像处理等
模型表达能力	有限	强大
特征提取方式	特征工程	特征自动抽取
代价函数凸性	凸代价函数; 没有局部最优点; 可以收敛到全局最优	高度非凸的代价函数; 存在大量的局部最优点; 容易收敛到局部最优
先验知识依赖度	依赖更多先验知识	依赖较少先验知识

可以将词映射转换到一个独立的向量空间; 进一步结合非线性神经网络提出了 n-gram 模型[36]; 受此启发, Collobert 等基于词向量方法及多层一维卷积神经网络 (Convolutional neural network, CNN), 实现了一个同时处理词性标注、语块切分、命名实体识别、语义角色标注四个典型自然语言处理任务的 SENNA (Semantic/syntactic extraction using a neural network architecture) 系统[28], 取得了与当时业界最好性能相当接近的效果. 尤其难能可贵的是, 相比传统算法, 仅用 3 500 多行 C 语言代码实现的 SENNA 系统, 运行速度更快, 所需内存空间更小.

对 Bengio 等提出的神经网络语言模型的进一步研究, Mikolov 等发现, 通过添加隐藏层的多次递归, 可以提高语言模型性能[37]; 将其应用于语音识别任务的结果令人吃惊, 在提高后续词预测的准确率及总体降低词的识别错误率方面都超越了当时最好的基准系统. 类似的模型也被 Schwenk 等用在统计机器翻译任务上[38], 其性能采用 BLEU (Bilingual evaluation understudy) 评分机制评判, 提高了将近 2 个百分点. 递归自动编码器 (Recursive auto-encoders) 模型[39] 在句段检测 (Sentence paraphrase detection) 任务中大大提高了 $F1$ 值. 此外, 基于深度模型的特征学习还在词义消歧[40]、情感分析[41−42] 等自然语言处理任务中均超越了当时最优系统, 取得不俗表现.

2.1 深度学习在自然语言处理领域应用的可行性分析

由上述应用可见, 自然语言处理领域中的深度学习技术已经表现出较强的生命力, 成为当前研究热点之一. 综合分析来看, 能够在自然语言处理领域中应用深度学习技术并取得良好效果, 我们认为主要有以下几点原因.

2.1.1 特征表示学习的需要

自然语言处理任务中首先要解决的问题是处理对象的表示形式. 为了表示对象, 通常必须抽取一些特征, 如文本的处理中, 常常用词集合来表示一个文档. 传统依赖手工的方式抽取特征, 费时费力; 不仅获取过程比较随意, 且完备性较差; 同时, 根据处理任务或领域的不同, 特征提取工作要重复进行, 无法实现表示共享. 能否使得机器也能像人类一样, 实现自动获取特征表示并进行推理学习? 深度学习就试图来解决这个问题. 深度学习中的特征提取, 即指可以自动从数据中学习获取特征.

2.1.2 无监督特征和权重学习的需要

目前大多数效果较好的自然语言处理任务和机器学习方法都依赖于标注数据. 在这种情况下, 基于标注语料库及有监督学习方式成为了主流手段. 但是, 就实际应用而言, 自然语言中大量存在的是未标注数据. 从这些未标注数据中挖掘信息, 就必须要考虑 (自动) 无监督方法. 深度神经网络采用无监督方式完成预训练过程, 恰恰提供了合适的训练模型.

2.1.3 学习多层分类表示的需求

仿生学的研究表明, 完成人类学习的大脑结构表现为一种多层 (深层) 不同的皮质层; 不同皮质层对应于不同的学习表示结构: 从抽象到具体, 逐层递减. 表示的抽象程度越高, 越能更多地交叉支持具体的处理任务. 因此, 我们需要利用好的学习模型, 更多地抽取出有用的中间表示形式 (Intermediate representations). 深度学习能够较好地抽取处理任务的多层分类表示.

此外, 人类自然语言具有递归特性 (Recursion). 比如, 自然语言中的句子, 事实上可以由词、短语递归组合而成. 深度学习提供了较为方便的递归操作, 可以支持这种自然语言递归组合特性的功能, 如递归神经网络 (Recursive neural network, RNN).

2.1.4 当前可用的技术及硬件平台支撑

深度学习结构一般由多层神经网络结点组成, 其预训练过程通常需要高性能计算的支持. 随着技术的发展, 能够提供高性能计算的硬件平台目前逐渐成熟, 如多核计算 (Multi-core computing)、图形处理单元 (Graphics processing unit, GPU) 等. 同时, 为深度网络结构中的组成单元提供算法支持的技术也有较好发展, 如 RBM、Auto-encoders 等; 并且各类结合自然语言处理的语言模型/算法[28, 37, 43−44] 等也逐渐得到优化, 性能得到提升. 这些硬件及软件技术的发展, 都为当前采用深度学习结构的自然语言处理提供了良好支撑环境.

2.2 面向自然语言处理的深度学习研究模型

面向领域任务的深度学习研究及应用, 需要解决两个普适问题: 1) 应用领域的原始特征表示; 2) 选择合适的深度学习算法. 前者实际是数据的表示问题, 后者代表了深度学习结构问题, 即深度学习模型. 例如在图像处理领域, 一般会选取图像像素矩阵作为原始特征表示[4, 6−7]; 而在语音处理任务中, 则会选取最基本的语音单位[43], 如音素 (Phonemes).

面向自然语言处理的深度学习研究, 同样需要考虑上述两个普适问题. 对于问题 1), 典型的有基于词向量空间[30, 45−46]、词袋模型 (Bag-of-words, BoW)、向量空间模型 (Vector space model, VSM) 等的表示方式; 对于问题 2), 目前普遍认可的是, 需要根据自然语言的特点, 来选择合适的深度学习模型. 人类自然语言具有递归特性. 比如, 自然语言中的句子, 事实上是由词、短语递归组合而成. 因此,

递归特性是自然语言的重要特征. 考虑自然语言递归特性的深度学习模型有循环神经网络 (Recurrent neural network, RNN)、递归神经网络、卷积神经网络及其系列改进模型[37, 47−50].

考虑上述两个问题之后, 在自然语言处理中应用深度学习的方式主要有两类: 1) 在深度学习模型中, 直接使用原始特征, 构建一类端到端 (End-to-end) 系统, 完成处理任务; 2) 在现有模型中, 将训练后的原始特征作为辅助特征扩充使用. 第 1) 种方式典型的工作如 SENNA 系统[30], 基于词向量方法及多层一维卷积神经网络完成了词性标注、语块切分、命名实体识别等系列任务; 类似的工作还有如 Socher 基于递归神经网络实现情感分析、句法分析等多项任务[51]. 第 2) 种方式典型的工作如 Turian 等将词向量作为额外的特征加入到现有最优系统中[52], 进一步提高了命名实体识别和短语识别的效果.

2.2.1 数据表示

2.2.1.1 One-hot representation

面向自然语言处理的深度学习, 首先要解决的是自然语言的表示问题. 在基于规则和统计的自然语言处理工作中, 最常见的是 One-hot representation 表示方法: 每个词表示为一个很长的向量; 其中只有一个维度的值为 1, 代表了当前的词; 其他绝大多数元素都为 0; 向量的维度是词表的大小. 如词 “话筒” 的向量可表示为 $[0001000000000000\cdots]$, 而词 “麦克” 的向量则可表示为 $[0000000010000000\cdots]$.

One-hot representation 如果采用稀疏方式存储, 形式上非常简洁. 结合传统机器学习算法, 如最大熵、支持向量机、条件随机场等, 该方法可以胜任大多数自然语言处理的主流任务; 但其纯粹的向量表示形式, 仅是孤立地表示单个词, 无法表达词与词之间的相关性. 如上述词 “话筒” 和 “麦克” 的表示向量, 单纯从这两个向量中, 无法看出两个词是否存在关系, 即使是麦克和话筒这样的同义词也不例外. Firth 提出一种利用相近邻词表示当前词的思想[53]: 通过计算不同范围的上下文相近邻词, 从而得到当前表示词的多种不同表达值. 比如当前中心词前后的词都可以用来计算得到当前中心词的表达值. 基于这种思想所产生的词表达方式, 被称为 Distributional similarity. 这也被誉为现代统计自然语言处理中最为成功的思想之一.

2.2.1.2 词向量

词向量表示方式延续并扩展了上述类似思想. 为了让相关或者相似的词, 在距离上更接近 (向量的距离可以用传统的欧氏距离来衡量), Hinton 提出了一种用 Distributed representation 表示词的方式[54], 通常被称为词向量. 词向量是一种低维实数向量, 如 $[0.792, -0.177, -0.107, 0.109, -0.542, \cdots]$. 用这种方式表示的向量, “麦克” 和 “话筒” 的距离会远远小于 “麦克” 和 “天气”. 词向量的方式是目前自然语言处理中应用深度学习的首选表示方式. 这种表示方法的好处在于: 首先, 如果采用传统 One-hot representation 的稀疏表示法, 在解决某些任务的时候, 比如构建语言模型, 可能会造成维数灾难[36], 而使用低维的词向量就可以避免类似问题; 其次, 从实践上看, 高维的特征如果要应用深度学习方法, 复杂度过高, 很难接受; 再有, 相似词的词向量距离相近, 这就让基于词向量设计的一些模型能够自带平滑功能.

词向量模型为文本中的每个单词构造一组特征, 较好地解决了自然语言中 “词” 一级的表示问题; 事实上, 也可以针对不同粒度进行推广, 如字向量、句子向量和文档向量[46], 从而实现字、短语、文本等表示. 而在文本级别, 另外一种常见的表示方法是词袋模型.

2.2.1.3 词袋模型

词袋模型是最早出现在自然语言处理领域中用来表示文档的方法. 词袋模型忽略文本的语法和语序, 用一组无序的单词来表达一个文档或一段文字, 文档中每个单词都是独立出现, 不依赖于其他单词是否出现. 文档或文字段仅仅看作是若干个词汇的集合.

例 1 a). Tom likes to play basketball. Mike likes too.

例 1 b). Mike also likes to play tennis.

根据上述两句话中出现的单词, 我们能构建出一个字典 (“Tom” : 1, “likes” : 2, “to” : 3, “play” : 4, “basketball” : 5, “Mike” : 6, “too” : 7, “also” : 8, “tennis” : 9).

该字典中包含 9 个单词, 每个单词有唯一索引, 注意它们的顺序和出现在句子中的顺序没有关联. 根据这个字典, 我们能将上述两句话重新表示为下述两个向量:

[1, 2, 1, 1, 1, 1, 1, 0, 0]

[0, 1, 1, 1, 0, 1, 0, 1, 1]

这两个向量共包含 9 个元素, 其中第 i 个元素表示字典中第 i 个单词在句子中出现的次数. 因此词袋模型可认为是一种统计直方图. 在文本检索和处理应用中, 可以通过该模型很方便地计算词频. 词袋模型典型的应用是文档分类. 定义文档集合 D,

共有 M 个文档; 将文档里面的所有单词提取出来后, 构成一个包含 N 个单词的词典. 基于词袋模型, 每个文档都可以被表示成为一个 N 维向量, 利用计算机就可以来完成海量文档的分类任务.

2.2.1.4 向量空间模型

向量空间模型 (Vector space model, VSM) 由 Salton[55] 于 20 世纪 70 年代提出, 并成功地应用于著名的 SMART (System for the mechanical analysis and retrieval of text) 文本检索系统. 向量空间模型概念简单, 把对文本内容的处理简化为向量空间中的向量运算, 并且它以空间上的相似度来表示语义的相似度, 直观易懂. 当文档被表示为文档空间的向量时, 就可以通过计算向量之间的余弦距离来度量文档间的相似性.

除了在信息检索领域的成功应用外, 向量空间模型也在自然语言处理的其他语义任务中有着令人印象深刻的结果. 如 Rapp 采用基于向量的词义表示方式来完成 TOEFL 考试的同义词多项选择问题[56], 取得了 92.5 % 的准确率, 相比之下, 当时的该项考试中考生的平均正确率也仅为 64.5 %. 类似的, Turney 使用语义关系的向量表示[57], 来完成 SAT 大学入学考试的推理多项选择问题, 取得了 56 % 的准确率, 和人类考试平均正确率 57 % 基本相当. 受向量空间模型思想启发, 在如何表示短语、句子、篇章等高一级的语言单元这一问题上, 我们认为, 可能的解决思路是: 以词向量为最小单位; 把同属一个短语、句子或篇章的词向量映射到同一向量空间中. 类似的工作在短语、篇章及文档的相似性判断中已经表现出较好的效果, 如 Manning 等使用向量空间模型作为搜索引擎[58], 来衡量一个查询与文档之间的相似度.

2.2.2 学习模型

词向量的获得一般都是依赖语言模型的训练. 常见的方式是在训练语言模型的过程中, 同时训练得到词向量.

定义 2. 定义语言单元集合 $E = \{$ 短语, 子句, 篇章 $\}$, 语言基础最小单元集合 $WordUnit = \{$ 词 | 字 $\}$. 其中, 英文中的语言基础最小单元是词, 而汉语的语言基础单位可以是字[26, 59].

定义 3. 语言模型可以形式化描述为: 给定一个字符串 $S = \{w_1 w_2 \cdots w_t\}$, 判断它属于自然语言的概率为 $P(S)$. 其中, $S \in E$, $w_i \in WordUnit$, $(i = 1, 2, \cdots, t)$. 简单的推论如下:

推论 1. $P(w_1, w_2, \cdots, w_t) = P(w_1) \times P(w_2|w_1) \times P(w_3|w_1, w_2) \times \cdots \times P(w_t|w_1, w_2, \cdots, w_{t-1})$.

在实际应用模型中, 一般都求近似解, 如 n 元语法 (n-gram) 模型就是如此.

2.2.2.1 神经网络与 n 元语法模型

神经网络与语言模型的结合工作, 最早源自 Xu 等[60] 提出一种使用神经网络构建二元语言模型的思想; 而 Bengio 等[36] 利用三层神经网络来构建 n 元语法模型的工作, 就把神经网络与语言模型训练的结合推上了一个新的台阶.

如图 5 所示最下方的 $w_{t-n+1}, \cdots, w_{t-2}, w_{t-1}$ 表示前 $n-1$ 个词. 根据前 $n-1$ 个词预测下一个词 w_t 是模型的终极目标. 其中, 模型使用了一个词向量库, 如定义 4 所示.

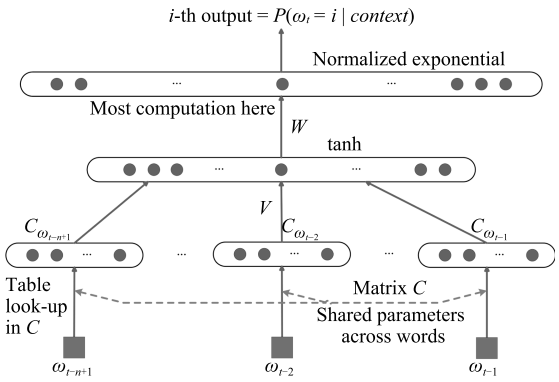


图 5 三层神经网络构建的 n-gram 模型[36]

Fig. 5 n-gram model constructed by three layer of neural networks[36]

定义 4. 词向量库定义为矩阵 $C = |V| \times m$, 其中 $|V|$ 表示语料中的总词数; m 表示词向量的维度; $\boldsymbol{c}(w)$ 表示从矩阵 C 中取出一行向量值, 用来代表词 w 所对应的词向量.

网络的输入层将 $C_{w_{t-n+1}}, \cdots, C_{w_{t-2}}, C_{w_{t-1}}$ 串连拼接起来, 构成一个 $m(n-1)$ 维的向量, 表示为 $\boldsymbol{x}$; 网络的第二层 (隐藏层) 基于 $d + H\boldsymbol{x}$ 计算方式直接得到结果 (其中 H 为隐藏层网络权重矩阵, d 为网络输入层到隐藏层的偏置项), 并使用 tanh 函数作为激活函数; 网络的第三层 (输出层) 共包含 $|V|$ 个节点, 使用 $softmax$ 激活函数将输出值 y 归一化, 如式 (1) 所示.

$$\hat{P}(w_t = i|w_1^{t-1}) = \frac{\mathrm{e}^{y_i}}{\sum\limits_j \mathrm{e}^{y_j}} \tag{1}$$

其中 y_i 表示下一个词为 i 的未归一化概率. 定义 y 的计算如式 (2):

$$y = b + W\boldsymbol{x} + U\tanh(d + H\boldsymbol{x}) \tag{2}$$

式中, b 为隐藏层到输出层的偏置项; 词特征输入层到输出层的权重矩阵 $W = |V| \times (n-1)m$; 隐藏层

到输出层的权重矩阵 $U = |V| \times h$, 其中 h 是隐藏层节点数量; 隐藏层权重矩阵 $H = h \times (n-1)m$; 矩阵 U 和网络隐藏层的矩阵乘法是模型的主要计算量. 为了提升模型的计算速度, 后期研究者的相关工作[29−30, 47], 都有对这一计算环节的简化. 式 (2) 中的矩阵 W 包含了从输入层到输出层的线性变换. 如果不需要线性变换的话, 可将 W 置为 0. 线性变换虽然不能提升模型效果, 但是可以减少一半的迭代次数[36].

最后, 采用随机梯度下降法实现模型优化工作, 在得到语言模型的同时, 也得到了词向量. 值得注意的是, 与一般神经网络输入层仅带一个输入值而无需优化不同, 为了使得到的模型自带平滑功能, 该模型的输入层参数是需要调整优化的. 相比于传统含有复杂平滑设计的 n 元语法模型而言, 该模型算法性能提升了约 10 % ∼ 20 %[36].

文献 [36] 最主要的思想, 随后在下面三个重要工作中体现出来: Log-bilinear 语言模型、Hierarchical log-bilinear 语言模型、循环神经网络语言模型.

2.2.2.2 Log-bilinear 语言模型

受文献 [36] 的影响, Mnih 等提出了一种 Log-bilinear 语言模型[61], 用于实现语言模型及词向量的训练. 这可以认为是自然语言处理中较早开始深度学习应用的尝试. 他们从最基本的受限玻尔兹曼机 (Restricted Boltzmann machines, RBM) 开始, 不断调整修改模型的能量函数, 最终获得了 Log-bilinear 模型. 采用神经网络的形式可以表示为:

$$\boldsymbol{h} = \sum_{i=1}^{t-1} H_i \boldsymbol{c}(w) \tag{3}$$

$$y_j = \boldsymbol{c}(w_j)^{\mathrm{T}} \boldsymbol{h} \tag{4}$$

式 (3) 和 (4) 可以合并表示为:

$$y_j = \sum_{i=1}^{t-1} \boldsymbol{c}(w_j)^{\mathrm{T}} H_i \boldsymbol{c}(w) \tag{5}$$

其中, $\boldsymbol{c}(w)$ 表示词 w 对应的词向量, Bilinear 模型形如 $\boldsymbol{x}^{\mathrm{T}} M \boldsymbol{y}$. 式 (3) 中, $\boldsymbol{h}$ 表示直接带有语义信息的隐藏层, 该隐藏层的维度为 m, 和词向量的维度保持一致. 矩阵 $H_i = m \times m$ 表示第 i 个词经过 H_i 变换之后, 对第 t 个词的贡献度. 式 (4) 中, y_j 等于 $\boldsymbol{c}(w_j)$ 和预测词向量 $\boldsymbol{h}$ 的内积, 可以反映两者的相似度, 直接表示下一个词为 w_j 的预测 log 概率. Log-bilinear 模型的理想实现是, 能够使用每个词的上文的所有词作为输入; 但是这样的实现成本极高. 一般采用类似 n 元语法模型的近似思想, 仅考虑上文 3 到 5 个词作为输入来预测下一个词.

2.2.2.3 Hierarchical log-bilinear 语言模型

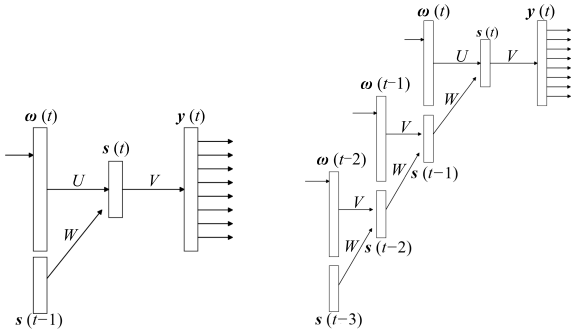
在 Log-bilinear 语言模型基础上, Mnih 等提出了一种带有层级思想的 HLB (Hierarchical log-bilinear) 语言模型替换了文献 [36] 提出的三层神经网络架构中计算成本最大的矩阵乘法, 在保证效果的基础上, 提升了速度[29].

这种层级的思想最初由 Morin 等提出[62], 他们采用 WordNet 中的 IS-A 关系, 将其转化为二叉树后再作分类预测. 实验结果表明尽管提高了速度, 但却降低了性能, 似乎有点得不偿失. Mnih 等借鉴了层级的思想, 但在实验中使用一种自举学习 (Bootstrapping) 的方法来自动构建平衡二叉树, 并将其用于替换网络最后一层[29]. 在预测向量分类时, 采用了二叉树中的非叶节点; 模型最后构建得到的叶子节点就用来确定具体的词. 计算复杂度也从原来的 $\mathrm{O}(|V|)$ 降低到 $\mathrm{O}(\log_2(|V|))$.

2.2.2.4 循环神经网络语言模型

文献 [36] 提出的模型中, 涉及大量训练参数. Mikolov 等提出了一种循环神经网络语言模型 (Recurrent neural network language model, RNNLM) 用于降低训练参数的数量[47]; 其采用 BPTT (Back-propagation through time) 优化算法, 取得了比 n 元语法模型中的最优方法更好的效果; 随后的研究中, Mikolov 等一直在 RNNLM 上作各种改进, 包括速度及正确率[37, 48−50].

循环神经网络与前面方法中使用的前馈网络训练的原理基本一致, 但是在结构上存在较大差别. 循环神经网络结构大致如图 6 所示.



(a) 神经网络结构抽象表示　(b) 神经网络流转过程

(a) Abstract representation of neural network architecture　(b) Transfer process of neural network

图 6　循环神经网络结构图

Fig. 6　Structure diagram of recurrent neural network

图 6 (a) 是网络的抽象表示结构, 由于循环神经

网络多用在时序序列上, 因此输入层、隐藏层和输出层都带有时序参数 t. 隐藏层计算公式表示为:

$$\boldsymbol{s}(t) = \text{sigmoid}(U\boldsymbol{w}(t) + W\boldsymbol{s}(t-1)) \quad (6)$$

式中, $\boldsymbol{w}(t)$ 是句子中用 One-hot representation 表示的第 t 个词的词向量; $\boldsymbol{s}(t)$ 向量表示隐藏层状态; $\boldsymbol{s}(t-1)$ 向量表示上一个隐藏层状态. 初始 $\boldsymbol{s}(0)$ 可以是含较小值 (如 0.1) 的一个向量, 随后的 $\boldsymbol{s}(1) = \boldsymbol{s}(0)$.

图 6 (b) 表示循环神经网络的流转过程. 每当一个新词输入, 循环神经网络联合输入新词的词向量与上一个隐藏层状态, 计算下一个隐藏层状态; 重复计算得到所有隐藏层状态; 各隐藏层最终通过传统的前馈网络得到输出结果.

不同于取 n 个词来近似预测下一个词的窗口模式, 循环神经网络可以真正充分地利用所有上文信息来预测下一个词. 这种方式实际上优劣并存, 如果一旦在实际使用中优化不足, 就可能丢失长距离信息, 导致预测词的性能甚至可能还比不上取 n 个词的窗口模式. 为了降低最后隐藏层到输出层的复杂计算量, Mikolov 等[47] 采用了一种分组的方法: 基于词频特点, 将 $|V|$ 个词分成 $\sqrt{|V|}$ 组, 先通过 $\sqrt{|V|}$ 次判断, 判断下一个词所属组别; 再通过若干次判断, 找出其属于组内的元素; 最后均摊复杂度约为 $\text{O}(\sqrt{|V|})$, 略差于 Mnih 和 Hinton 所提模型[29] 的复杂度 $\text{O}(\log(|V|))$. 但是这种方法最大的优点是结构比较简单, 可以减少误差传递.

2.2.2.5 基于词向量的改进模型

Collobert 和 Weston 在 2008 年首次提出了一种特殊的词向量计算方法[30], 文中系统地总结了他们基于词向量完成的多项自然语言处理任务, 如词性标注、命名实体识别、短语识别、语义角色标注等工作. 不同于求近似解的 n 元语法模型, 他们的词向量训练方法直接求解的近似解. 给出定义 5.

定义 5. 定义 $f(w_{t-n+1}, \cdots, w_{t-1}, w_t)$ 表示窗口连续 n 个词的分值. f 只有相对高低之分, 并不表示概率的特性. f 分值越高, 表明这句话越是正常; f 分值低, 表明这句话不合理. 极端情况, 如果随机把几个词堆积在一起, f 值将表示为负分.

基于此, Collobert 和 Weston 使用 Pair-wise 方法来训练词向量[30]. 其中, 需要最小化目标函数如下.

$$\sum_{x \in X} \sum_{w \in D} \max\{0, 1 - f(x) + f(x(w))\} \quad (7)$$

式中, X 为训练集中的所有连续的 n 元短语, D 是整个字典, x 表示正样本, $x(w)$ 表示负样本, 而函数 $f(x)$ 是正样本的分值转换, $f(x(w))$ 是负样本的分值转换. 式 (7) 中的第一个求和枚举计算将训练语料中的 n 元短语都作为正样本挑选出来了; 所有的负样本则通过第二个对字典的枚举构建得到. $x(w)$ 表示用 w 替换正常短语 x 的中间词, 这样处理后, 最终得到短语大多数情况下肯定不是正确的短语, 可以作为负样本使用. 由式 (7) 可见, 正样本最终的打分要比负样本至少高出 1 分.

f 函数的结构基本上和文献 [36] 中的网络结构一致. 它们的共同之处在于: 1) 窗口中的 n 个词所对应的词向量被串连形成一个长向量; 2) 隐藏层都经过一层网络计算后得到. 不同点在于: Collobert 和 Weston 模型[30] 的输出层只有一个节点表示得分, 而文献 [36] 模型则拥有 $|V|$ 个节点; 此外, 采用 $HardTanh$ 代替 tanh 激活函数以降低计算复杂度.

Collobert 和 Weston 模型中窗口 n 值设定为 11, 字典大小值 $|V|$ 设定为 130 000, 利用维基百科英文语料和路透社语料训练 7 周后得到了 $C\,\&\,W$ 词向量. 相比其他词向量, $C\,\&\,W$ 词向量主要特点有:

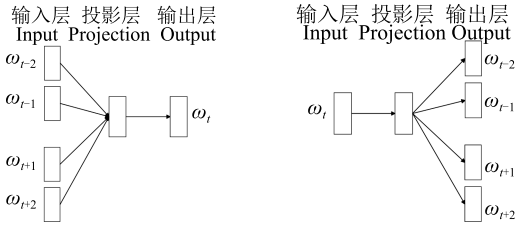
1) $C\,\&\,W$ 词向量仅包含小写单词. 也就是说, 不同于其他词向量对大小写词分开处理, 该词表不区分大小写, 它把单词都按照小写词加以处理.

2) $C\,\&\,W$ 词向量是通过半监督学习得到的. 因为 $C\,\&\,W$ 词向量是在通过词性标注、命名实体识别等多任务优化的半监督学习后得到的, 区别于其他方法中的无监督学习.

Turian 等在将 Collobert 和 Weston 所实现的 $C\,\&\,W$ 向量与 Mnih 和 Hinton 实现的向量[29] 做了对比实验[52], 并在其标注好的语料上运行了 HLB (Hierarchical log-bilinear) 模型, 得到了另一份词向量. Mikolov 等的系列论文[45−46] 介绍了将词表征为实数值向量的词向量工具包 word2vec (本文第 2.5.2 节讨论了该工具包), 其主要用到 CBOW 模型 (Continuous bag-of-words model) 和 Skip-gram 模型 (Skip-gram model), 分别采用 Hierarchical softmax 和 Negative sampling 框架进行设计. CBOW 模型和 Skip-gram 模型都包含三层架构, 即输入层、投影层和输出层, 所不同的是, 前者在已知当前词 w_t 的上下文 $w_{t-2}, w_{t-1}, w_{t+1}, w_{t+2}$ 的前提下预测当前词 w_t, 如图 7 (a) 所示; 而后者是在已知当前词 w_t 的前提下, 预测其上下文 $w_{t-2}, w_{t-1}, w_{t+1}, w_{t+2}$, 如图 7 (b) 所示.

经过 word2vec 工具包训练得到的词向量具备很好的类比 (Word analogy) 特性, 在一定程度上可以表示词语的语义和语法性质. 面向知识图谱的表示学习算法 TransE[63] 正是受此类比特性启发而提出的. 知识图谱包含大量实体、实体的语义类别和

实体间的关系, 可以用三元组 (主体、关系、客体) 来表示. TransE 算法将三元组中的关系看作主体到客体的翻译, 使得三元组满足线性转换. 利用特征表示向量描述实体和关系, 可以更加容易地计算实体之间的语义关系.



(a) CBOW 模型　　(b) Skip-gram 模型

(a) CBOW model　　(b) Skip-gram model

图 7　词向量 word2vec 的模型结构图

Fig. 7　Model structure diagram of word2vec

2.2.3　模型讨论

上述其他所有模型, 除了循环神经网络语言模型以外, 本质上模型的输入层到隐藏层 (第一层) 都是等价的. 即使形式比较特别的 HLB 语言模型, 如果把模型中的 H 看成 H_i 的拼接, 则也可以得到类似其他方法那样的等式:

$$[H_1 H_2 \cdots H_t]\boldsymbol{c}(w_1)\boldsymbol{c}(w_2)\cdots\boldsymbol{c}(w_t) = H_1\boldsymbol{c}(w_1) + H_2\boldsymbol{c}(w_2) + \cdots + H_t\boldsymbol{c}(w_t) \quad (8)$$

所以上述诸多模型, 本质上非常相似, 差别主要在于隐藏层到输出层的语义定义. Bengio 采用最朴素的线性变换[36], 从隐藏层直接映射到每个词; Collobert 和 Weston 将语言模型做了简化[30], 利用线性变换把隐藏层转换为 f 分值; Mnih 和 Hinton 复用了词向量[29], 进一步强化了语义, 并用层级结构加速; Mikolov 等则用了分组来实现加速[47].

此外, Collobert 和 Weston 的实验结果表明[30]: 相比于随机初始化, 将词向量作为初始值, 在不同任务上的效果都有显著提升; 同时发现训练语料越大, 实际效果越好. 在将词向量用作辅助特征时, Turian 等[52] 的实验表明 $C\,\&\,W$ 向量在命名实体识别和短语识别中的效果比 Mnih 和 Hinton[29] 实现的向量稍好些; 而两者联合使用, 效果更佳.

近期 Mikolov 等的研究发现了一个有意思的现象[45]: 两个词向量之间的关系, 可以用两个向量的差来体现. 例如已经知道 a 与 b 的关系, 类似等价于 c 与 d 的关系, 现在给定 a、b、c, 判断 $\boldsymbol{c}(d)$ 是否近似于词向量 $\boldsymbol{c}(a) - \boldsymbol{c}(b) + \boldsymbol{c}(c)$. 例如实验中发现有词向量 $\boldsymbol{c}(\text{king}) - \boldsymbol{c}(\text{queen}) \approx \boldsymbol{c}(\text{man}) - \boldsymbol{c}(\text{woman})$, 进一步发现, $\boldsymbol{c}(\text{queen})$ 居然就是最接近 $\boldsymbol{c}(\text{king}) - \boldsymbol{c}(\text{man}) + \boldsymbol{c}(\text{woman})$ 的词向量. 向量之间存在的这种线性平移关系, 极有可能成为词向量未来发展的关键. Mikolov 等的实验结果也同样表明, 语料越大, 词向量效果就越好, 这一点同 Collobert 和 Weston[30] 的实验结果是一致的.

2.3　面向自然语言处理的深度学习应用策略

Bengio 提出了采用梯度下降法 (Stochastic gradient descent, SGD) 训练深度结构的系列建议[64], 其中大致可将训练过程分为: 无监督预训练、模型参数初始化及后期优化、模型调试等. 参考这一过程, 我们定义如下在自然语言处理领域深度学习的应用策略, 应用架构如图 8 所示.

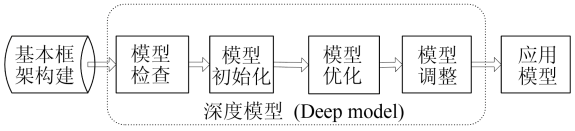


图 8　面向自然语言处理的深度学习应用架构图

Fig. 8　Deep learning application architecture for NLP

步骤 1. 构建基本模型框架. 针对处理任务, 选择合适的神经网络结构, 构建深度学习基本模型框架.

步骤 2. 模型检查. 采用梯度下降法检查模型实现是否存在错误. 这对于整个过程至关重要.

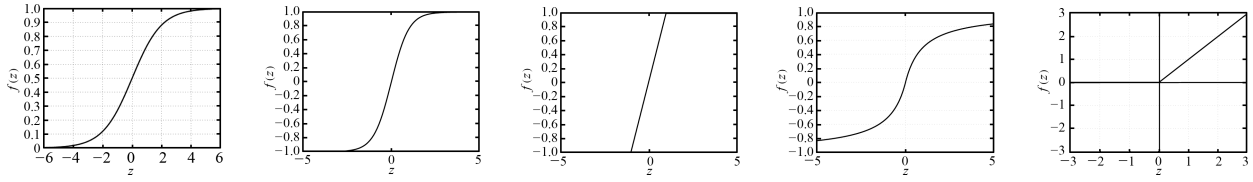
步骤 3. 模型初始化. 主要涉及神经网络隐藏层偏置量 b 和网络结点权重矩阵 W 的参数初始化.

步骤 4. 模型优化. 主要涉及模型参数调整优化.

步骤 5. 模型调整. 检查模型是否能够满足过拟合要求, 如果没有, 调整模型参数使其能够满足过拟合要求; 如果达到过拟合要求, 那就采用正则化 (Regularization) 方法调整模型.

2.3.1　构建基本模型框架

构建面向自然语言处理的深度学习模型, 首先要考虑基本表示结构, 可选的表示结构有 Single words、Fixed windows、Recursive sentence 或 Bag of words; 其次要考虑非线性化过程, 可选的非线性化函数有 logistic ("sigmoid")、tanh、hard tanh、soft sign、rectifier 等, 如图 9 所示. sigmoid 函数及其反函数都具有单调递增特点, 可实现变量在 $[0, 1]$ 区间的映射, 故经常作为神经网络阈值函数使用; 但是, sigmoid 函数初始化权重集后, 能够激活近半数的神经元, 这与模仿大脑神经元稀疏性工作的原理似乎相悖, 同时也不利于深度网络训练. 与此相比, rectifier 函数具有单侧抑制性, 可以相对有效降低深度网络训练复杂度. 此外, 统计表明, 对于深度网络而言, tanh 函数性能最佳, 使用频率也是最高; hard tanh 函数类似, 计算代价相对低廉. 上述几种常用的非线性函数如图 9 所示, 其公式如下:



(a) logistic 函数图 (b) tanh 函数图 (c) hard tanh 函数图 (d) soft sign 函数图 (e) rectifier 函数图

(a) Graphic of logistic function (b) Graphic of tanh function (c) Graphic of hard tanh function (d) Graphic of soft sign function (e) Graphic of rectifier function

图 9 几种常用的非线性化函数可视化表示

Fig. 9 Visual representation of several commonly used nonlinear functions

1) logistic ("sigmoid") 函数:

$$f(z)=\frac{1}{1+\mathrm{e}^{-z}} \tag{9}$$

2) tanh 函数:

$$f(z)=\frac{\mathrm{e}^{z}-\mathrm{e}^{-z}}{\mathrm{e}^{z}+\mathrm{e}^{-z}} \tag{10}$$

3) hard tanh 函数:

$$f(z)=\begin{cases}-1, & z<-1\\ z, & -1\le z\le 1\\ 1, & z>1\end{cases} \tag{11}$$

4) soft sign 函数:

$$f(z)=\frac{z}{1+|z|} \tag{12}$$

5) rectifier 函数:

$$f(z)=\max(z,0) \tag{13}$$

2.3.2 模型检查

梯度下降法是常用的模型检查方法. 通过模型检查, 能够验证所实现的模型是否存在明显缺陷. 首先, 在检查模型之前, 需要选择合适的梯度表示; 其次, 循环计算调整参数; 最后, 比较输出值和实际结果之间的偏差, 以确保其一致.

2.3.3 模型初始化

模型的初始化, 首先设置隐藏层的偏置量为 0, 并设置输出层的偏置量为假定权重值 w 都为 0 的情况下的最优值; 其次, 设置权重 $w\in(-r,r), r=\sqrt{6/(fan_{in}+fan_{out})}$, 其中 fan_{in} 为前一层网络的结点数, fan_{out} 为后一层网络的结点数; 最后, 完成预训练过程.

2.3.4 模型优化

模型优化主要涉及参数的训练. 设 θ 为参数 $\{W,b\}$, W 为网络权重矩阵, b 为网络单元的偏置 (Bias). 常规优化算法有随机梯度下降 (SGD)、LBFGS (Limited-memory Broyden-Fletcher-Goldfarb-Shanno)、共轭梯度下降 CG (Conjugate gradients).

SGD 形式化定义如下:

$$\theta^{(t)}\leftarrow\theta^{(t-1)}-\varepsilon_t\frac{\partial L(Z_t,\theta)}{\partial\theta} \tag{14}$$

式中, L 为损失函数, Z_t 为当前样本, θ 为参数向量, ε_t 为学习速率. SGD 算法中对于学习速率的选择, 简单的办法是选定一个固定值, 作为全局变量使用; 并且学习速率随着时间动态逐步递减, 以确保模型收敛. 典型的递减方式如取倒数形式 $\mathrm{O}(1/t)$, 形式化可表示为:

$$\varepsilon_t=\frac{\varepsilon_0\tau}{\max(t,\tau)} \tag{15}$$

在优化过程中, 不同的优化算法都有不同的优缺点, 需要区分不同应用场合, 加以选择使用. 比如在参数维度较低 (小于 1 万维) 的情况下, LBFGS 的效果最好; 而针对高维问题, CG 算法又要比其他两种算法更优. 此外, 如果是在小规模数据集上, 则 LBFGS 或 CG 算法较优; 如果是在大数据集合中, SGD 算法对模型参数的调整性能最佳[65]. 大数据集合经常伴随大规模训练集, 为降低训练集的计算复杂度, 在每次迭代时仅利用部分训练集样本加以训练. 这里的部分训练样本其实是训练集的一个子集, 一般称为 mini-batch. 在实际优化过程中, 目前常用的是带 mini-batch 的 SGD 优化算法.

在深度学习网络中, 梯度表示为雅可比行列矩阵的形式, 每一单元的结果都依赖于前一步计算. 这可能会使梯度结果变化速度过快, 从而导致梯度下降局部变化的假设不再成立.

2.3.5 模型调整

经过上述步骤得到的模型, 如果出现过拟合, 则需要在本阶段作正则化调整. 第一步最简单的方式是: 降低模型规模. 可以通过降低各种参数值达到这一目的, 如可以减少神经网络结点单元数、网络层数及其他可用参数等. 其次, 可以使用标准 $L1$ 或

$L2$ 的 Regularity 限制调整权重值, 或者采用稀疏化方式促使模型复杂度降低, 提升计算速度和模型的泛化能力.

2.4 面向自然语言处理的深度学习典型应用

相比于图像和语音领域所取得的成果, 深度学习在自然语言处理上尽管还未取得重大突破, 但也在以下相关诸多领域, 如词性标注、句法分析、词义学习、情感分析有着初步应用, 并取得较好效果.

2.4.1 分词和词性标注

分词是指按照一定的规范, 将连续的字序列重新组合成词序列的过程. 词性标注 (Part-of-speech tagging, POS) 则是指确定句子中每个词的词性, 如形容词、动词、名词等, 又称词类标注或者简称标注.

在英文分词和词性标注方面, 结合深度学习开展相关研究最有影响力的是 Collobert 等的研究工作[28], 他们基于词向量方法及多层一维卷积神经网络, 实现了一个同时处理词性标注、语块切分、命名实体识别、语义角色标注四个典型自然语言处理任务的 SENNA 系统, 取得了与当时业界最好性能相当接近的效果.

在中文分词和词性标注方面, Zheng 等分析了利用深度学习来进行上述两项工作的可行性[59], 主要集中在特征发现、数据表示和模型算法三方面工作. 在特征发现方面, 他们尝试采用深层神经网络来发现与任务相关的特征, 从而避免依赖于具体任务的特征工程 (Task-specific feature engineering); 在数据表示方面, 他们利用大规模非标注数据 (Unlabeled data) 来改善中文字的内在表示 (Internal representation), 然后使用改善后的表示来提高有监督的分词和词性标注模型的性能; 在模型算法方面, 他们提出 Perceptron-style 算法替代 Maximum-likelihood 方法, 在性能上接近当前最好的算法, 但计算开销更小. 特别有意思的是, 受英文的词向量[28, 36] 的概念启发, 他们提出以中文的字 (Character) 为基本单位的字向量概念, 由此提供了深度学习利用中文大规模非标注数据开展预训练的可能性.

2.4.2 句法分析

句法分析 (Syntactic analysis) 的主要任务是自动识别句子中包含的句法单位以及这些句法单位相互之间的关系, 即句子的结构. 通常的做法是: 给定一个句子作为输入, 利用语言的语法特征作为主要知识源构建一棵短语结构树.

Henderson 提出一种 Left-corner 句法分析器[66], 首次将神经网络成功应用于大规模句法分析中; 随后, Henderson 又基于同步网络训练句法分析器[67]; Titov 等使用 SVM 改进了一种生成型句法分析器用于不同领域的句法分析任务[68]; 他们还在特征学习基础上寻求进一步改进系统的方法[69]. Collobert 基于深度循环图转移网络提出了一种应用于自然语言句法分析的快速判别算法[70]. 该方法使用较少的文本特征, 所取得的性能指标与当时最好的判别式分析器和基准分析器相当, 而在计算速度上具有较大优势.

与此同时, Costa 等也尝试采用递归神经网络模型[71], 用于解决增量式句法分析器中侯选附加短语的排序问题. 他们的工作首次揭示了利用递归神经网络模型获取足够的信息, 从而修正句法分析结果的可能性; 但是他们只在大约 2 000 个句子的子集上做了测试, 相对来说测试集合显得有点少. Menchetti 等[72] 在使用 Collins 分析器[73] 生成侯选句法树的基础上, 利用递归神经网络模型实现再排序. 和他们的工作类似, Socher 等提出了一种 CVG (Compositional vector grammar) 模型用于句法结构预测[74], 该模型将 PCFG (Probabilistic context free grammars) 与递归神经网络模型相结合, 充分利用了短语的语法和语义信息. 与斯坦福分析器相比, 他们的系统不仅性能上提高了约 3.8 % (取得了 90.4 % 的 $F1$ 值), 而且在训练速度上提高约 20 %. Legrand 等基于简单神经网络模型, 提出了一种自底向上的句法分析方法[75]. 其主要优势在于结构简单, 计算开销少, 分析速度快, 且性能接近当前最好系统.

2.4.3 词义学习

基于无监督学习机制的词义表示在自然语言处理中有着非常广泛的用途, 例如可以作为某些学习算法的输入或者是特殊词的特征表示. 但是, 目前大多数词义表示模型都依赖本地上下文关系, 且只能一词一义. 这存在很大局限性, 因为通常可能一个词有着多个含义; 并且对于学习词义而言, 全局上下文关系能够提供更多有用的信息. Huang 等[76] 在 Collobert 和 Weston[30] 的基础上, 提出了一种新的深度神经网络模型用于词义学习. 该模型通过综合本地和全局文本上下文信息, 学习能够更好表达词义的隐藏词; 通过学习每个词的多义词表示, 来更好地解释同名歧义; 进一步, 在基于多个词向量表示词的多义性基础上, 通过对模型的改进, 使得词向量包含更丰富的语义信息. 实验表明, 相比于其他向量, Huang 等的方法与人工标注语义相似度最为接近.

Socher 等提到了对语言的深度理解概念[40]. 他们认为, 单个词的向量空间模型在词汇信息的学习中得到了充分成功的应用, 但是由于不能有效获取长短语的组合词义, 则在语言的进一步深度理解上产生了障碍. 他们提出了一种深度递归神经网络模

型, 该模型可通过学习短语和句子的组合向量来表示语义. 句子可以是任意句法类型和长度的句子. 该模型给句法树上的每个结点都分配一个向量和矩阵; 向量获取元素的本体语义; 矩阵捕获邻近单词和短语的变化信息. 该模型在三种不同的实验中取得了显著性能, 分别是副词–形容词组合对的情感分布预测、影评标记的情感分类、情感关系分类, 如因果或名词之间的主题信息等.

2.4.4 情感分析

情感分析 (Sentiment analysis) 又称为倾向性分析、意见抽取 (Opinion extraction)、意见挖掘 (Opinion mining)、情感挖掘 (Sentiment mining)、主观分析 (Subjectivity analysis) 等, 它是对带有情感色彩的主观性文本进行分析、处理、归纳和推理的过程, 如从评论文本中分析用户对 "手机" 的价格、大小、重量、易用性等属性的情感倾向.

Zhou 等提出一种称为主动深度网络 (Active deep network, ADN) 的半监督学习算法用于解决情感分类问题[77]. 首先, 在标注数据和无标注数据集上, 他们采用无监督学习算法来训练 RBM, 进而搭建 ADN, 并通过基于梯度下降算法的有监督学习方法进行结构微调; 之后, 结合主动学习 (Active learning) 方法, 利用标注好的评论数据来训练半监督学习框架, 将其与 ADN 结构融合, 实现了一个面向半监督分类任务的统一模型. 实验表明, 该模型在 5 种情感分类数据集上都有较为突出的性能. ADN 中 RBM 性能的提升, 部分得益于无标注训练数据的规模提高, 这就为大量丰富的无标注评论数据开辟了利用空间.

Glorot 等提出了一种采用无监督学习方式从网络评论数据中学习如何提取有意义信息表示的深度学习方法[78], 并将其用于情感分类器的构建中, 在 Amazon 产品的 4 类评论基准数据上的测试性能显著. Socher 等基于 RAE (Recursive auto-encoders) 提出一种深度学习模型[79], 应用于句子级的情感标注预测. 该模型采用词向量空间构建输入训练数据, 利用 RAE 实现半监督学习. 实验表明, 该模型准确性优于同类基准系统. 针对词向量空间在长短语表达上缺乏表现力这一缺点, Socher 等引入情感树库 (Sentiment treebank), 以增强情感训练和评价资源[51]; 在此基础上, 训练完成的 RNTN (Recursive neural tensor network) 模型, 性能表现突出: 简单句的正负情感分类准确率从 80 % 提高到 85.4 %; 短语情感预测从 71 % 提高到 80.7 %. 针对词袋模型的缺陷, Le 等提出了一种基于段落的向量模型 (Paragraph vector)[41], 该模型实现了一种从句子、段落和文档中自动学习固定长度特征表示的无监督算法, 在情感分析和文本分类任务中都有优异表现, 尤其是简单句的正负情感分类准确率相比 RNTN 模型[51] 提高了 2.4 %. Kim 在 Collobert 等构建的 CNN 模型基础上[28], 借助 Google 公司的词向量开源工具 word2vec 完成了 1 000 亿个单词的新闻语料训练, 并将其用于包括情感树库等试验语料上的简单句情感分类任务, 取得了 88.1 % 的当时最好性能[42]. 这似乎再次验证了 BigData 思想: 只要包含足够的训练数据, 深度学习模型总能够尽可能逼近真实结果.

2.4.5 机器翻译

机器翻译 (Machine translation) 是利用计算机把一种自然源语言转变为另一种自然目标语言的过程, 也称为自动翻译. 目前, 基于深度学习的统计机器翻译方法研究热点可以分为: 传统机器翻译模型上的神经网络改进、采用全新构建的端到端神经机器翻译 (Neural machine translation, NMT) 方法两种类型.

大多数统计机器翻译系统建模采用基于对数线性框架 (Log-linear framework), 尽管已经取得较为成功的应用, 但依然面临如下局限性: 1) 所选特征需要与模型本身成线性匹配; 2) 特征无法进一步解释说明以便反映潜在语义. 针对上述局限, Liu 等提出了一种附加神经网络模型 (Additive neural network)[80], 用于扩展传统对数线性翻译模型; 此外, 采用词向量将每个词编码转化为特征向量, 作为神经网络的输入值, 该模型在中英和日英两类翻译任务中均获得了较好性能. 词对齐 (Word alignment) 方法是机器翻译常用的基础技术. Yang 等基于深度神经网络 (DNN) 提出了一种新颖的词对齐方法[81]. 该方法将多层神经网络引入隐马尔科夫模型, 从而利用神经网络来计算上下文依赖的词义转换得分; 并采用大量语料来预先训练词向量. 在大规模中英词对齐任务的实验表明, 该方法取得较好的词对齐结果, 优于经典的隐马尔科夫模型和 IBM Model 4.

与上述传统机器模型中的神经网络针对翻译系统局部改进所不同的是, 近来出现的神经机器翻译构建了一种新颖的端到端翻译方法[82−85]: 其初始输入为整个句子, 并联合翻译输出的候选句子构成句子对; 通过构建神经网络, 并结合双语平行语料库来寻找条件概率最大时的候选句子对, 最终输出目标翻译句. 神经机器翻译试图构建并训练一个可以读取源句子, 直接翻译为目标句子的单一、大型的神经网络. 从统计角度来看, 机器翻译可以等价为在给定输入源句子 X 的情况下, 寻找条件概率最大时的翻译目标句子 Y 的值, 即求 $\arg\max_Y p(Y|X)$.

事实上, 目前提出的大多数神经机器翻译方

法都属于一类编码解码器 (Encoder-decoders) 模型[83−84], 其主要框架包含两部分: 首先编码器将输入的长度不固定的源句子编码转换为固定长度的向量, 之后解码器将向量解码输出为翻译的目标句. 这里的解码器, 就可以采用一类深度神经网络模型, 例如循环神经网络. 在使用循环神经网络作为编解码的框架中, 编码器读入输入句子, 经过编码输出为向量 $\boldsymbol{c}$. 表示如下:

$$h_t = f(x_t, h_{t-1}) \tag{16}$$

$$\boldsymbol{c} = q\left(\{h_1, \cdots, h_{T_x}\}\right) \tag{17}$$

其中, $h_t \in \mathbf{R}^n$ 表示时刻 t 时的隐藏状态, $\boldsymbol{c}$ 是由多个隐藏状态序列生成的向量, f 和 q 是非线性函数. 例如 Sutskever 等使用多层 LSTM (Long short-term memory) 表示 f 函数[83]. 在给定上下文向量 $\boldsymbol{c}$ 和前续已经预测得到的词序列 $\{y_1, \cdots, y_{t'-1}\}$ 的前提下, 循环神经网络训练的编码器用来预测下一个词 $y_{t'}$. 表示如下:

$$p(y) = \prod_{t=1}^{T} p(y_t | \{y_1, \cdots, y_{t-1}\}, \boldsymbol{c}) \tag{18}$$

其中, $y = \{y_1, \cdots, y_{T_y}\}$, 基于循环神经网络, 每个条件概率可以建模如下:

$$p(y_t | \{y_1, \cdots, y_{t-1}\}, \boldsymbol{c}) = g(y_{t-1}, s_t, \boldsymbol{c}) \tag{19}$$

其中, g 是非线性多层函数, 可以由循环神经网络建模表示, s_t 是循环神经网络的隐藏层. 类似的结构也可以采用循环神经网络和卷积神经网络混合表示[82].

编码解码器模型一个潜在的问题是所采用的神经网络需要能够把输入源句子的所有信息都压缩进入固定长度的向量中, 这在处理长句子时可能比较困难, 尤其是那些远比训练语料库中的长得多的句子. Cho 等实验表明随着输入句子长度的增加, 编码解码器模型性能快速降低[85]. 为了克服这个缺陷, Bahdanau 等引入了一个扩展的编码解码器模型[86]. 该模型在翻译过程中, 也是每次根据上下文相关信息, 以及已经找到的目标单词, 通过引入注意力机制来自动学习目标词在源语言上的对齐目标单词. 和基本编码解码器模型不同的是, 该模型并不是试图把整个输入句子编码转换放进单个固定长度的向量中, 而是编码转换放进一个向量序列中; 当解码时, 就可以在向量序列中选择一个合适的向量子集用于解码, 这种方式使得神经网络翻译模型不必过度纠结于输入句子的长度. 实验同时也表明这种改进的编码解码器模型在处理长句问题时性能表现更好. Dong 等基于多任务学习机制联合学习, 通过在一对多的序列到序列的机器翻译模型中共享源语言的表示, 构建了一种源语言到多个目标语言的翻译模型[87].

2.5 面向自然语言处理的深度学习平台工具

面向自然语言处理的深度学习平台或工具较多, 根据开发语言的不同, 可以分为基于 Python、C++、C 或 Java 等不同程序设计语言实现的算法库或框架; 根据实现的神经网络模型的不同, 可以分为面向 RBM/DBN (Deep belief network) 等组件、卷积神经网络 (CNN)、循环神经网络、递归神经网络实现的框架平台; 根据功能目标不同, 又可以分为提供深度学习基本功能实现的函数库/工具包、在函数库基础上面向领域任务构建的不同应用框架等. 下面从不同角度介绍几类典型的深度学习开源工具.

2.5.1 函数库/工具包

最早出现的, 较为完整实现深度学习框架的库函数包是由加拿大 Montreal 大学 LISA (Laboratoire d'Informatique des Systèmes Adaptatifs) 实验室 Bergstra 等开发的 Theano, 是一个基于 Python 语言的库, 实现了深度学习相关模型及算法, 如 RBM/DBN 等, 可有效支持涉及多维矩阵相关的定义、优化及评估等数学运算.

Theano 具有以下特点: 1) 有效集成 NumPy. NumPy 是一个用 Python 实现的科学计算包, 一般和稀疏矩阵运算包 Scipy 配合使用. Theano 使用 numpy.ndarray 集成编译函数, 全面兼容 Numpy 库函数. 2) 可方便应用于 GPU 平台. 在一类数据密集型的计算任务中, 与普通仅使用 32 位浮点数的 CPU 相比, 计算速度可提高 100 多倍. 3) 有效的符号区分能力. Theano 可有效支持带有 1 个或多个输入的扩展函数. 4) 速度及可靠性表现优异. 即便 x 取值很小, 也能计算得到 $\log(1+x)$ 的正确结果. 5) 支持动态 C 代码生成. 6) 具有众多测试和自检单元. 可方便地检测和诊断多种类型的错误.

在 Theano 基础上, 后续研究者陆续开发了众多深度学习框架, 如 Pylearn2、Blocks、Keras 等. 采用 Python 语言实现的 Keras 是一个追求简易、高度模块化的神经网络库, 开发的主要目的在于将研究创意能够快速转换为深度学习实验的原型框架, 避免因为实验困难而错过了创意的验证. Keras 的扩展性能非常好, 可以快速实现基于卷积神经网络、循环神经网络或者两者混合实现的经典模型, 同时能够运行于 CPU 和 GPU 平台. Keras 和前两个工具包都是在 Theano 库基础上构建的, 稍有不同的地方在于 Keras 还支持另一个函数库 TensorFlow.

TensorFlow 是一个开源软件库, 最早由 Google

公司机器智能研究部门的谷歌大脑团队 (Google Brain Team) 开发完成, 目的是为了搭建机器学习及深度神经网络研究平台. 该软件库采用数据流图模式实现数值计算, 数据流图中的结点表示数学运算, 图中的边表示多维数据阵列. 采用该软件库开发的平台, 架构灵活, 代码一次开发, 无需修改, 即可在单机、服务器或移动设备上流畅运行, 支持多 CPU/GPU 计算.

类似 TensorFlow 可以在各种设备上运行的轻量级函数库还有 MShadow, 这也是奉行简单实用、灵活方便主义的模板库, 基于 C++/CUDA 实现, 支持 CPU/GPU/多 GPU 以及分布式系统. 在该函数库上扩展开发了 CXXNet 和 MxNet 分布式深度学习框架, 也是一类高质量的软件工具包.

2.5.2 数据表征工具

第一个在自然语言任务中取得较好性能的深度学习应用软件是 SENNA, 由 Collobert[28] 团队开发, 具有架构简单、独立性强 (不依赖其他自然语言处理工具)、运行速度快等特点, 在 POS Tagging、Chunking、Named entity recognition、Semantic role labeling 等四个典型自然语言处理问题上取得的性能都与当时最好系统相当. SENNA 采用大约 3 500 行的标准 C 语言 (ANSI C) 代码实现, 可以运行在配备 150 MB 内存且支持浮点运算的计算机平台上. 目前最新的版本是 SENNA V 3.0, 更新于 2011 年 8 月. SENNA 特别强调它们在 Wikipedia 上花费 2 个月时间所训练的词向量, 将词表征为多维向量, 可以用于不同的自然语言处理任务.

与此相类似的, Google 公司在 2013 年开源软件 word2vec 也是将词表征为实数值向量的有效工具. word2vec 使用第 2.2.1 节中所提到 Distributed representation 词向量表示方式, 通过一个三层的神经网络模型训练, 可以将文本内容处理转化为 K 维向量空间中的运算; 进一步, 文本语义上的相似度, 就可以用向量空间中的距离 (如欧氏距离、cosine 相似度) 来表示. word2vec 在神经网络模型训练中, 根据词出现的频率采用 Huffman 编码设计隐藏层节点数目, 词频越高的词语, 所激活的隐藏层节点数目越少, 这就大大降低了计算复杂度. 实验表明, 优化的单机版本的 word2vec, 在一天内可以训练上亿个词. 这种训练的高效性, 也是 word2vec 在自然语言处理中大受欢迎的一个重要原因.

2.5.3 经典神经网络模型

能够将文本内容转换表示为向量形式, 开启了面向自然语言处理的深度学习应用热潮. 理论上, 基于向量表示, 所有的深度学习模型都用来处理不同的自然语言处理任务; 但在实践中, 使用频率最高、效果最为突出的还是卷积神经网络、循环神经网络和递归神经网络等.

2.5.4 深度神经网络组件

最早由 Ruslan Salakhutdinov 基于 Matlab 开发的一类小型函数库 (Matrbm、Estimating partition functions of RBM′s、Learning deep Boltzmann machines)[35], 主要用于训练构成深度学习网络的组件, 如 RBM, 规模不大. 随后出现的 Deeplearning4j 是一个规模较大, 完整实现深度学习框架的平台工具, 支持 GPU, 可以运行在 Hadoop 计算平台上, 这就为大规模数据处理提供了便利性. Deeplearning4j 采用 Java/Scala 语言实现了 RBM、深度可信网络 (DBN)、LSTM、递归自动解码器 (Recursive autoencoder) 等一类典型的深度神经网络组件, 为构建可靠的、分布式处理的深度神经网络框架提供了良好的基础.

2.5.5 卷积神经网络工具

卷积神经网络是一类典型经典的面向自然语言处理的深度学习模型. 上节提到的 SENNA 即是一种基于卷积神经网络原理的工具软件. 此外, 其他比较著名的卷积神经网络模型实现工具有 Cuda-convnet、ConvNet 以及第 2.5.1 节提到的 Keras 等. Cuda-convnet2 是当前 Cuda-convnet 的最新版本, 采用 C++/CUDA 实现, 训练过程基于 BP 算法; ConvNet 是一个采用 Matlab 实现的卷积神经网络工具包.

2.5.6 循环神经网络等工具

循环神经网络以及递归神经网络模型也是近年来在自然语言处理领域被认为是最有潜力的深度学习模型, 上文提及的很多函数库及工具包都提供了相应实现, 如采用 Python 语言实现、基于 Theano 的 Keras, 采用 Java 语言支持分布式大规模计算平台的 Deeplearning4j 等. 其他还有一些比较令人注目的开源工具如 Tomas Mikolov 开发的基于循环神经网络语言模型的工具包[2] (支持中文及 UTF-8 格式的语料)[47]、Richard Socher 开发的基于递归神经网络的工具包[3][39] 等, 当前在自然语言处理的各种任务中逐渐崭露头角.

3 存在的问题与未来的研究方向

3.1 数据表示问题及展望

“自然语言” 在深度学习中用于初始输入的 “数

[2]Mikolov 开发的循环神经网络模型 http://www.fit.vutbr.cz/ imikolov/rnnlm/
[3]Socher 的递归神经网络模型 http://www.socher.org

据源” 是字或词, 和图像、语音分别采用像素点及音素作为初始 “数据源” 相比较, 前者已经包含了人类的语义解释, 是经过人类主观思考处理后形成的, 而后者是原始的, 还没有经过人类加工处理. 这一点是自然语言处理和其他两种应用最大的不同. 由此, 我们联想到, 这是否也是深度学习在图像、语音方面能够取得巨大成功, 而在自然语言方面还没有成功的关键原因呢? 因为包含原始信号的情况下, 不容易丢失未知信息, 从而能够通过深度学习的不同分层特征表示, 更为全面地表征原始输入, 进一步为分类、聚类等具体应用提供充分的特征支撑.

目前来看, 面向自然语言处理的深度学习中的数据表征主要还是 Word embedding 概念, 只是可能在不同语言中, 具体 Word 的表示单位有所不同, 如英文中可以是单词或词缀, 中文中则换成了词组或字, 本质上还是通过某种映射规则, 将 Word 转换为向量表示.

在如何将深度学习与现有自然语言处理具体任务结合方面, 目前还没有比较明显有突破的方法或规律可以遵循. 现有工作中, 比较直接简单的做法是, 以词或短语作为原始输入, 构建向量类型的表达方式, 经过深度学习分层学习后得到的特征可以添加进现有基于特征的半监督学习系统中进行处理[49]. 此外, 还有将深度学习模型与当前经典问题结合后产生的应用模型, 如结合树形或链式结构的递归神经网络或循环神经网络模型等[39−40, 51, 88]. 因此, 考虑如何将深度学习与自然语言处理任务结合的具体落地应用也是值得研究的重点.

3.2 学习模型问题及展望

面向自然语言处理的深度学习研究工作, 目前尚处于起步阶段, 尽管已有的深度学习算法模型如循环神经网络、递归神经网络和卷积神经网络等已经有较为显著的应用, 但还没有重大突破. 围绕适合自然语言处理领域的深度学习模型构建等研究应该有着非常广阔的空间.

在当前已有的深度学习模型研究中, 难点是在模型构建过程中参数的优化调整方面. 主要如深度网络层数、正则化问题及网络学习速率等. 可能的解决方案比如有: 采用多核机提升网络训练速度; 针对不同应用场合, 选择合适的优化算法等.

深度学习模型的训练过程中, 最为突出的问题是训练速度. 普遍来看, 深度学习模型的训练速度远比线性模型来得慢. 此外, 模型性能的优劣, 一般与训练数据集的规模有关. 数据集越大, 训练结果越好[89]. 这一点, 非常符合目前主流的大数据应用趋势. 但是, 这也可能给学习模型的优化带来发展阻碍. 在极力追求产生大数据训练集的情况下, 是否会削弱对更优学习模型的研究热情呢?

3.3 其他问题及思考

3.3.1 自动学习和人工结合

围绕数据表示及特征提取问题, 已有大量文献分析了自然语言处理中的数据源特征和无监督自动学习方法. 深度学习一直强调学习特征采用自动的方法, 然而, 如果能够在训练过程中融合已有面向特定应用领域的显然的知识 (如人工选取的明显特征规律), 对于深度模型而言, 依然具有吸引力. 这就好比人类学习, 完全抛弃祖先的知识而白手起家开展工作, 是不可想象的. 但是, 要做到这点非常困难. 首先, 针对问题领域, 需要选择合适的模型架构, 比如针对自然语言的语义框架选择合适的深度结构; 其次, 人类知识的融合, 最佳的进入点应该是在模型的第一层, 类似线性模型一样, 总的目标是希望能够使模型具有自我学习的能力.

此外, 在自然语言处理领域, 已经有了大量的人工标注知识. 深度学习可以通过有监督学习得到相关的语义知识. 这种知识和人类总结的知识应该存在某种对应关系, 尤其是在一些浅层语义方面. 因为人工标注, 本质上已经给深度学习提供了学习的目标. 只是深度学习可以不眠不休地学习, 这种逐步靠拢学习目标的过程, 可能远比人类总结过程来得更快. 这一点, 从最近 Google 公司围棋人工智能软件 AlphaGo 短时间内连胜两位人类围棋高手的事实, 似乎能够得到验证[90].

3.3.2 自然语言的不确定性

由于一词多义的存在, 使得即使采用词向量技术作为深度学习的原始输入信号, 也还是不能如图像或语音一样将所有原始信息确定地输入到深度学习模型中. 在深度学习模型分层表示原始输入信号的不同特征时, 这种不确定性所带来的误差有可能在不同层间被传递并局部放大.

解决这种一词多义所带来的不确定性的方法, 似乎还是要结合上下文语言情境. 因此, 突破自然语言字、词、短语、小句等局部表示的局限性, 面向包含上下文全局信息的篇章、文本来开展深层语义理解, 如篇章分析、篇章理解等, 应该是重点发展的方向之一.

4 结束语

相比于图像处理, 自然语言的分层抽象其实并不明显. 自然语言处理在深度学习中所采用的特征表示, 目前主要是 Word embedding 机制. 尽管从语言表达的形式角度, 也可以构建字母、单词、词组、短语、句子等层次结构, 但从语义表达角度来看, 似乎没有如图像处理那样具有明显的抽象分层, 例如

单词和词组、词组和短语之间, 语义表达上面并没有非常明显的不同. 抽象层次不明显, 实质上就可能限制了特征表示的多样性, 从而无法最好地发挥深度学习多层特征表示的长处. 除了词向量之外, 是否还有更好的特征表示方式? 采用何种模型来构建明显分层机制? 等等此类问题, 也是面向自然语言处理的深度学习在未来发展中需要重点研究的内容. 当然, 尽管目前来看, 面向自然语言的深度学习还存在着各种各样的问题, 但是总体而言, 现有深度学习的特征自动表示及分层抽象思想, 为自然语言处理提供了一种将特征表示和应用实现独立分开的可行方法, 这将使得在领域任务和语言之间的泛化迁移变得较为容易.

致谢

本文作者衷心感谢苏州大学李正华博士、邹博伟博士及王中卿博士对本文写作的热情帮助.

References

1 Erhan D, Bengio Y, Couville A, Manzagol P A, Vincent P, Samy B. Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, 2010, **11**: 625−660

2 Sun Zhi-Jun, Xue Lei, Xu Yang-Ming, Wang Zheng. Overview of deep learning. *Application Research of Computers*, 2012, **29**(8): 2806−2810
(孙志军, 薛磊, 许阳明, 王正. 深度学习研究综述. 计算机应用研究, 2012, **29**(8): 2806−2810)

3 Bengio Y. Learning deep architectures for AI. *Foundations and Trends® in Machine Learning*, 2009, **2**(1): 1−127

4 Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets. *Neural Computation*, 2006, **18**(7): 1527−1554

5 Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks. *Science*, 2006, **313**(5786): 504−507

6 Bengio Y, Lamblin P, Popovici D, Larochelle H. Greedy layer-wise training of deep networks. In: Proceedings of the 2007 Advances in Neural Information Processing Systems 19 (NIPS′06). Vancouver, Canada: MIT Press, 2007. 153−160

7 Ranzato M A, Poultney C, Chopra S, LeCun Y. Efficient learning of sparse representations with an energy-based model. In: Proceedings of the 2007 Advances in Neural Information Processing Systems 19 (NIPS′06). Vancouver, Canada: MIT Press, 2007. 1137−1144

8 Weston J, Ratle F, Collobert R. Deep learning via semi-supervised embedding. In: Proceedings of the 25th International Conference on Machine Learning (ICML′08). New York, USA: ACM Press, 2008. 1168−1175

9 Srivastava N, Mansimov E, Salakhutdinov R. Unsupervised learning of video representations using LSTMs. In: Proceedings of the 32nd International Conference on Machine Learning (ICML′15). Lille, France: Omni Press, 2015. 843−852

10 Jia K, Sun L, Gao S H, Song Z, Shi B E. Laplacian auto-encoders: an explicit learning of nonlinear data manifold. *Neurocomputing*, 2015, **160**: 250−260

11 Chan T H, Jia K, Gao S H, Lu J W, Zeng Z N, Ma Y. PCANet: a simple deep learning baseline for image classification? *IEEE Transactions on Image Processing*, 2015, **24**(12): 5017−5032

12 Alain G, Bengio Y. What regularized auto-encoders learn from the data-generating distribution? *The Journal of Machine Learning Research*, 2014, **15**(1): 3563−3593

13 Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 2014, **15**(1): 1929−1958

14 Dosovitskiy A, Springenberg J T, Riedmiller M, Brox T. Discriminative unsupervised feature learning with convolutional neural networks. In: Proceedings of the 2014 Advances in Neural Information Processing Systems 27 (NIPS′14). Montréal, Quebec, Canada: MIT Press, 2014. 766−774

15 Sun Y, Wang X G, Tang X O. Deep learning face representation from predicting 10 000 classes. In: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, Ohio, USA: IEEE, 2014. 1891−1898

16 Qiao Jun-Fei, Pan Guang-Yuan, Han Hong-Gui. Design and application of continuous deep belief network. *Acta Automatica Sinica*, 2015, **41**(12): 2138−2146
(乔俊飞, 潘广源, 韩红桂. 一种连续型深度信念网的设计与应用. 自动化学报, 2015, **41**(12): 2138−2146)

17 Längkvist M, Karlsson L, Loutfi A. A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognition Letters*, 2014, **42**: 11−24

18 Han X F, Leung T, Jia Y Q, Sukthankar R, Berg A C. MatchNet: unifying feature and metric learning for patch-based matching. In: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR′15). Boston, Massachusetts, USA: IEEE Press, 2015. 3279−3286

19 Szegedy C, Liu W, Jia Y Q, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A. Going deeper with convolutions. In: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR′15). Boston, Massachusetts, USA: IEEE, 2015. 1−9

20 Denton E L, Chintala S, Szlam A, Fergus R. Deep generative image models using a Laplacian pyramid of adversarial networks. In: Proceedings of the 2015 Advances in Neural Information Processing Systems 28 (NIPS′15). Montreal, Canada: MIT Press, 2015. 1486−1494

21 Dong C, Loy C C, He K M, Tang X O. Learning a deep convolutional network for image super-resolution. In: Proceedings of the 13th European Conference on Computer Vision (ECCV′14). Zurich, Switzerland: Springer International Publishing, 2014. 184−199

22 Nie S Q, Wang Z H, Ji Q. A generative restricted Boltzmann machine based method for high-dimensional motion data modeling. *Computer Vision and Image Understanding*, 2015, **136**: 14−22

23 Jain A, Tompson J, LeCun Y, Bregler C. Modeep: a deep learning framework using motion features for human pose estimation. In: Proceedings of the 12th Asian Conference on Computer Vision (ACCV′2014). Singapore: Springer International Publishing, 2015. 302−315

24 Geng Jie, Fan Jian-Chao, Chu Jia-Lan, Wang Hong-Yu. Research on marine floating raft aquaculture SAR image target recognition based on deep collaborative sparse coding network. *Acta Automatica Sinica*, 2016, **42**(4): 593−604 (耿杰, 范剑超, 初佳兰, 王洪玉. 基于深度协同稀疏编码网络的海洋浮筏 SAR 图像目标识别. 自动化学报, 2016, **42**(4): 593−604)

25 Erhan D, Szegedy C, Toshev A, Anguelov D. Scalable object detection using deep neural networks. In: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR′14). Columbus, Ohio, USA: IEEE, 2014. 2155−2162

26 Qi Y J, Das S G, Collobert R, Weston J. Deep learning for character-based information extraction. In: Proceedings of the 36th European Conference on IR Research on Advances in Information Retrieval. Amsterdam, The Netherland: Springer International Publishing, 2014. 668−674

27 Nie L Q, Wang M, Zhang L M, Yan S C, Zhang B, Chua T S. Disease inference from health-related questions via sparse deep learning. *IEEE Transactions on Knowledge and Data Engineering*, 2015, **27**(8): 2107−2119

28 Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 2011, **12**: 2493−2537

29 Mnih A, Hinton G E. A scalable hierarchical distributed language model. In: Proceedings of the 2009 Advances in Neural Information Processing Systems 21 (NIPS′08). Vancouver, Canada: MIT Press, 2009. 1081−1088

30 Collobert R, Weston J. A unified architecture for natural language processing: deep neural networks with multitask learning. In: Proceedings of the 25th International Conference on Machine Learning (ICML′08). Helsinki, Finland: ACM Press, 2008. 160−167

31 Olshausen B A, Field D J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 1996, **381**(6583): 607−609

32 Overview of deep learning and parallel implementation [Online], available: http://djt.qq.com/article/view/1245, June 20, 2016

33 Hastad J. *Computational Limitations for Small Depth Circuits*. Cambridge, MA, USA: Massachusetts Institute of Technology, 1987

34 Serre C, Mellot-Draznieks C, Surblé S, Audebrand N, Filinchuk Y, Férey G. Role of solvent-host interactions that lead to very large swelling of hybrid frameworks. *Science*, 2007, **315**(5820): 1828−1831

35 Salakhutdinov R R, Hinton G. Deep Boltzmann machines. In: Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS′09). Florida, USA: Omni Press, 2009. 448−455

36 Bengio Y, Ducharme R, Vincent P, Jauvin C. A neural probabilistic language model. *The Journal of Machine Learning Research*, 2003, **3**: 1137−1155

37 Mikolov T, Deoras A, Kombrink S, Burget L, Černocký J H. Empirical evaluation and combination of advanced language modeling techniques. In: Proceedings of the 2011 Conference of the International Speech Communication Association (INTERSPEECH′2011). Florence, Italy: ISCA Press, 2011. 605−608

38 Schwenk H, Rousseau A, Attik M. Large, pruned or continuous space language models on a GPU for statistical machine translation. In: Proceedings of the NAACL-HLT 2012 Workshop: Will We ever Really Replace the *N*-gram Model? on the Future of Language Modeling for HLT. Montréal, Canada: ACL Press, 2012. 11−19

39 Socher R, Huang E H, Pennington J, Ng A Y, Manning C D. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In: Proceedings of the 2011 Advances in Neural Information Processing Systems 24 (NIPS′11). Granada, Spain: MIT Press, 2011. 801−809

40 Socher R, Huval B, Manning C D, Ng A Y. Semantic compositionality through recursive matrix-vector spaces. In: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Jeju Island, Korea: ACL Press, 2012. 1201−1211

41 Le Q, Mikolov T. Distributed representations of sentences and documents. In: Proceedings of the 31st International Conference on Machine Learning (ICML′14). Beijing, China: ACM Press, 2014. 1188−1196

42 Kim Y. Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP′2014). Doha, Qatar: ACL Press, 2014. 1746−1751

43 Dahl G E, Yu D, Deng L, Acero A. Context-dependent pre-trained deep neural networks for large vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 2012, **20**(1): 30−42

44 Mohamed A R, Dahl G E, Hinton G. Acoustic modeling using deep belief networks. *IEEE Transactions on Audio, Speech, and Language Processing*, 2012, **20**(1): 14−22

45 Mikolov T, Yih W T, Zweig G. Linguistic regularities in continuous space word representations. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT′2013). Atlanta, Georgia: ACL Press, 2013. 746−751

46 Mikolov T, Sutskever I, Chen K, Corrado G S, Dean J. Distributed representations of words and phrases and their compositionality. In: Proceedings of the 2013 Advances in Neural Information Processing Systems 26 (NIPS′13). Nevada, USA: MIT Press, 2013. 3111−3119

47 Mikolov T, Karafiát M, Burget L, Černocký, Khudanpur S. Recurrent neural network based language model. In: Proceedings of the 2010 International Conference on Spoken Language Processing (ICSLP′2010). Chiba, Japan: Speech Communication Press, 2010. 1045−1048

48 Mikolov T, Kombrink S, Burget L, Černocký J H, Khudanpur S. Extensions of recurrent neural network language model. In: Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Prague, Czech Republic: IEEE, 2011. 5528−5531

49 Mikolov T, Deoras A, Povey D, Burget L, Černocký J H. Strategies for training large scale neural network language models. In: Proceedings of the 2011 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). Waikoloa, Hawaii, USA: IEEE Press, 2011. 196−201

50 Mikolov T, Zweig G. Context dependent recurrent neural network language model. In: Proceedings of the 2012 IEEE Conference on Spoken Language Technology (SLT). Miami, Florida, USA: IEEE, 2012. 234−239

51 Socher R, Perelygin A, Wu J Y, Chuang J, Manning C D, Ng A Y, Potts C. Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP′2013). Seattle, USA: ACL Press, 2013. 1631−1642

52 Turian J, Ratinov L, Bengio Y. Word representations: a simple and general method for semi-supervised learning. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL′2010). Uppsala, Sweden: ACL Press, 2010. 384−394

53 Firth J R. A synopsis of linguistic theory 1930-55. *Studies in Linguistic Analysis*. Oxford: Philological Society, 1957. 1−32

54 Hinton G E. Learning distributed representations of concepts. In: Proceedings of the 8th Annual Conference of the Cognitive Science Society. Amherst, Massachusetts: Cognitive Science Society Press, 1986. 1−12

55 Salton G. Automatic processing of foreign language documents. *Journal of the American Society for Information Science*, 1970, **21**(3): 187−194

56 Rapp R. Word sense discovery based on sense descriptor dissimilarity. In: Proceedings of the 9th Conference on Machine Translation Summit. New Orleans, USA: IAMT Press, 2003. 315−322

57 Turney P D. Expressing implicit semantic relations without supervision. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics (COLING and ACL 2006). Sydney, Australia: ACL Press, 2006. 313−320

58 Manning C D, Raghavan P, Schütze H. *Introduction to Information Retrieval*. Cambridge: Cambridge University Press, 2008.

59 Zheng X Q, Chen H Y, Xu T Y. Deep learning for Chinese word segmentation and POS tagging. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP′2013). Seattle, Washington, USA: ACL Press, 2013. 647−657

60 Xu W, Rudnicky A I. Can artificial neural networks learn language models? In: Proceedings of 2000 International Conference on Spoken Language Processing (ICSLP′2000). Beijing, China: Speech Communication Press, 2000. 202−205

61 Mnih A, Hinton G. Three new graphical models for statistical language modelling. In: Proceedings of the 24th International Conference on Machine Learning (ICML′07). Corvallis, Oregon: ACM Press, 2007. 641−648

62 Morin F, Bengio Y. Hierarchical probabilistic neural network language model. In: Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics (AISTATS′2005). Barbados: Omni Press, 2005. 246−252

63 Bordes A, Usunier N, Garcia-Durán A, Weston J, Yakhnenko O. Translating embeddings for modeling multi-relational data. In: Proceedings of the 2013 Advances in Neural Information Processing Systems 26 (NIPS′13). Nevada, USA: MIT Press, 2013. 2787−2795

64 Bengio Y. Deep learning of representations for unsupervised and transfer learning. In: Proceedings of the ICML2011 Unsupervised and Transfer Learning Workshop. Bellevue, Washington, USA: ACM Press, 2012. 17−37

65 Le Q V, Ngiam J, Coates A, Lahiri A, Prochnow B, Ng A Y. On optimization methods for deep learning. In: Proceedings of the 28th International Conference on Machine Learning (ICML′11). Bellevue, Washington, USA: ACM Press, 2011. 67−105

66 Henderson J. Neural network probability estimation for broad coverage parsing. In: Proceedings of the 10th Conference on European Chapter of the Association for Computational Linguistics (EACL′03). Budapest, Hungary: ACL Press, 2003. 131−138

67 Henderson J. Discriminative training of a neural network statistical parser. In: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics (ACL′2004). Barcelona, Spain: ACL Press, 2004. 95−102

68 Titov I, Henderson J. Porting statistical parsers with data-defined kernels. In: Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL-2006). New York, USA: ACL Press, 2006. 6−13

69 Titov I, Henderson J. Constituent parsing with incremental sigmoid belief networks. In: Proceedings of the 45th Annual Meeting on Association for Computational Linguistics (ACL′2007). Prague, Czech Republic: ACL Press, 2007. 632−639

70 Collobert R. Deep learning for efficient discriminative parsing. In: Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS′2011). Fort Lauderdale, Florida, USA: Omni Press, 2011. 224−232

71 Costa F, Frasconi P, Lombardo V, Soda G. Towards incremental parsing of natural language using recursive neural networks. *Applied Intelligence*, 2003, **19**(1−2): 9−25

72 Menchetti S, Costa F, Frasconi P, Pontil M. Wide coverage natural language processing using kernel methods and neural networks for structured data. *Pattern Recognition Letters*, 2005, **26**(12): 1896−1906

73 Collins M. Head-driven statistical models for natural language parsing. *Computational linguistics*, 2003, **29**(4): 589−637

74 Socher R, Bauer J, Manning C D, Ng A Y. Parsing with compositional vector grammars. In: Proceedings of the 51st Annual Meeting on Association for Computational Linguistics (ACL′2013). Sofia, Bulgaria: ACL Press, 2013. 455−465

75 Legrand J, Collobert R. Recurrent greedy parsing with neural networks. In: Proceedings of the 2014 European Conference on Machine Learning and Knowledge Discovery in Databases. Nancy, France: Springer Press, 2014. 130−144

76 Huang E H, Socher R, Manning C D, Ng A Y. Improving word representations via global context and multiple word prototypes. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL′2012). Jeju Island, Korea: ACL Press, 2012. 873−882

77 Zhou S S, Chen Q C, Wang X L. Active deep networks for semi-supervised sentiment classification. In: Proceedings of the 23rd International Conference on Computational Linguistics (COLING′2010). Beijing, China: ACL Press, 2010. 1515−1523

78 Glorot X, Bordes A, Bengio Y. Domain adaptation for large-scale sentiment classification: a deep learning approach. In: Proceedings of the 28th International Conference on Machine Learning (ICML′11). Bellevue, Washington, USA: Omni Press, 2011. 513−520

79 Socher R, Pennington J, Huang E H, Ng A Y, Manning C D. Semi-supervised recursive autoencoders for predicting sentiment distributions. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP′2011). Edinburgh, UK: ACL Press, 2011. 151−161

80 Liu L M, Watanabe T, Sumita E, Zhao T J. Additive neural networks for statistical machine translation. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL′2013). Sofa, Bulgaria: ACL Press, 2013. 791−801

81 Yang N, Liu S J, Li M, Zhou M, Yu N H. Word alignment modeling with context dependent deep neural network. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL′2013). Sofa, Bulgaria: ACL Press, 2013. 166−175

82 Kalchbrenner N, Blunsom P. Recurrent continuous translation models. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP′2013). Seattle, Washington, USA: ACL Press, 2013. 1700−1709

83 Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks. In: Proceedings of the 2014 Advances in Neural Information Processing Systems 27 (NIPS′14). Montréal, Quebec, Canada: MIT Press, 2014. 3104−3112

84 Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP′2014). Doha, Qatar: ACL Press, 2014. 1724−1734

85 Cho K, van Merriënboer B, Bahdanau D, Bengio Y. On the properties of neural machine translation: encoder-decoder approaches. In: Proceedings of the 8th Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8). Doha, Qatar: ACL Press, 2014. 103−111

86 Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. In: Proceedings of the 3rd International Conference on Learning Representations (ICLR′2015). San Diego, California, USA: arXiv Press, 2015. 1409.0473V7

87 Dong D X, Wu H, He W, Yu D H, Wang H F. Multi-task learning for multiple language translation. In: Proceedings of the 53rd Annual Meeting on Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. Beijing, China: ACL Press, 2015. 1723−1732

88 Pinheiro P O, Collobert R. Recurrent convolutional neural networks for scene labeling. In: Proceedings of the 31st International Conference on Machine Learning (ICML′14). Beijing, China, 2014. 82−90

89 Le Q V. Building high-level features using large scale unsupervised learning. In: Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. Vancouver, BC: IEEE, 2013. 8595−8598

90 Tian Yuan-Dong. A simple analysis of AlphaGo. *Acta Automatica Sinica*, 2016, **42**(5): 671−675
(田渊栋. 阿法狗围棋系统的简要分析. 自动化学报, 2016, **42**(5): 671−675)

奚雪峰 苏州大学计算机科学与技术学院博士研究生. 主要研究方向为自然语言理解, 篇章分析, 自动问答.
E-mail: xfxi@mail.usts.edu.cn
(**XI Xue-Feng** Ph. D. candidate at the School of Computer Science and Technology, Soochow University. His research interest covers natural language understanding, discourse analysis and question-answering.)

周国栋 苏州大学特聘教授. 主要研究方向为自然语言理解, 中文信息处理, 信息抽取. 本文通信作者.
E-mail: gdzhou@suda.edu.cn
(**ZHOU Guo-Dong** Distinguished professor at the School of Computer Science and Technology, Soochow University. His research interest covers natural language understanding, Chinese computing, and information extraction. Corresponding author of this paper.)