单词嵌入表示学习综述

刘建伟†, 高 悦

(中国石油大学(北京) 自动化系, 北京 102249)

摘要: 单词嵌入表示学习是自然语言处理(NLP)中最基本但又很重要的研究内容, 是所有后续高级语言处理任务的基础. 早期的单词独热表示忽略了单词的语义信息, 在应用中常常会遇到数据稀疏的问题, 后来随着神经语言模型(NLM)的提出, 单词被表示为低维实向量, 有效地解决了数据稀疏的问题. 单词级的嵌入表示是最初的基于神经网络语言模型的输入表示形式, 后来人们又从不同角度出发, 提出了诸多变种. 本文从模型涉及到的语种数的角度出发, 将单词嵌入表示模型分为单语言单词嵌入表示模型和跨语言单词嵌入表示模型两大类. 在单语言中, 根据模型输入的颗粒度又将模型分为字符级、单词级、短语级及以上的单词嵌入表示模型, 不同颗粒度级别的模型的应用场景不同, 各有千秋. 再将这些模型按照是否考虑上下文信息再次分类, 单词嵌入表示还经常与其它场景的模型结合, 引入其他模态或关联信息帮助学习单词嵌入表示, 提高模型的表现性能, 故本文也列举了一些单词嵌入表示模型和其它领域模型的联合应用. 通过对上述模型进行研究, 将每个模型的特点进行总结和比较, 在文章最后给出了未来单词嵌入表示的研究方向和展望.

关键词: 单词嵌入表示学习; 神经网络; 语言模型; 跨语言; 双向编码器表示; 信息瓶颈

引用格式: 刘建伟, 高悦. 单词嵌入表示学习综述. 控制理论与应用, 2022, 39(7): 1171 – 1193

DOI: 10.7641/CTA.2022.10678

Survey of word embedding

LIU Jian-wei†, GAO Yue

(Department of Automation, China University of Petroleum, Beijing 102249, China)

Abstract: Word embedding is the most basic research content in natural language processing (NLP), and it is a very important research direction. It is the basis of all advanced language processing tasks, such as using word vectors to complete various tasks in NLP. At the beginning, the one-hot ignored the semantic information of words and often led to data sparsity in application. Later, with the development of the neural language model (NLM), words were represented as dense and low-dimensional vectors, which effectively solved the problem of data sparsity and their high dimensionality. The input of the models based on neural network language models are word-level word embedding, but a variety of models have been proposed from different directions. In this survey, from the point of view of the number of languages utilizing in the model, we divide word embedding models into single-language word embedding and cross-language word embedding. In single-language, according to the granularity of model input, the model is divided into character-level, word-level, phrase-level and above word embedding model. The application scenarios of models with different granularity level are different and each has its own strengths. These models are further classified according to whether context information is considered. At the same time, word embedding is often combined with other models, which can help to learn word embedding by introducing other models or correlation information to improve the performance of the model. Therefore, in this survey, we also list some joint applications of word embedding models and other domain models. Through the study and introduction of the above models, the characteristics of each model are summarized and compared. Finally, the future research direction and prospect of word embedding are given.

Key words: word embedding; neural network; language model: cross-lingual; BERT; information bottleneck

Citation: LIU Jianwei, GAO Yue. Survey of word embedding. *Control Theory & Applications*, 2022, 39(7): 1171 – 1193

收稿日期: 2021−07−27; 录用日期: 2022−01−25.

†刘建伟. E-mail: liujw@cup.edu.cn; Tel.: +86 13699114280.

本文责任编委: 苏剑波.

中国石油大学(北京)科研基金(2462020YXZZ023)资助.

Supported by the Science Foundation of China University of Petroleum, Beijing (2462020YXZZ023).

1 引言

在自然语言处理的历史上, 单词嵌入表示学习(word embedding)一直是人们研究的一个热点, 它是自然语言处理(natural language processing, NLP)中语言模型与表示学习技术的统称, 是将自然语言表示的单词转换为计算机能够处理的向量或矩阵形式的技术. 与先前的独热表示(one-hot)、n-元文法(n-gram)[1]、共现矩阵(co-occurrence matrix)不同, 单词嵌入表示保留了丰富的语义信息, 能更好地完成NLP任务. NLP任务包括词类语法解析[2–3]、命名实体识别[4]和语义角色标注[5]、机器翻译[6], 单词嵌入表示已被证明在各种任务中表现良好[7–10]. Harris[11]在1954年提出的分布假设(distributional hypothesis)为单词向量表示提供了理论基础: 上下文相似的单词, 其语义也应该相似. Firth[12]在1957年对分布假设进行了进一步明确的阐述: 单词的语义由其上下文决定. 基于分布假设得到的表示均可称为分布表示(distributional representation), 也就是单词嵌入表示(word embedding)[7]. Bengio等人[13]在2003年提出了神经语言模型(neural language model, NLM), NLM是单词嵌入表示的基本模型, 与n-gram通过联合概率分布考虑单词之间的位置关系不同的是, NLM利用单词向量表示进一步表示词语之间的相似性, 比如, 近义词在相似的上下文里的可替代性, 或者同类事物的词可以在语料中出现频数不同的情况下获得相似的概率, 从而克服维数灾难问题. Mikolov[14–15]在2013年提出了Word2Vec模型, 包含了两个单词向量表示学习模型: Continuous Bag–of–Words(CBOW)和Skip–Gram. 为了更深入的理解单词嵌入表示, 本文针对单词嵌入表示从不同的角度进行分类, 如图1所示. 本文结构安排如下: 第1节为引言, 简单介绍了单词嵌入表示的发展和现状; 接下来两节都是单语言单词嵌入表示模型, 第2节按照模型的输入的颗粒度分为字符级单词嵌入表示模型、单词级单词嵌入表示模型和短语级及以上的单词嵌入表示模型; 第3节根据模型是否结合上下文, 可以解决一词多义的问题, 分为上下文无关的模型和考虑上下文信息的模型, 由于第2节介绍的模型多为上下文无关的模型, 这里就不多加赘述了, 在第3 节主要介绍考虑上下文信息的模型; 第4节介绍跨语言单词嵌入表示模型; 第5节介绍的是单词嵌入表示模型与其它类型的模型的联合应用; 第6节对单词嵌入表示模型未来趋势及发展方向进行了讨论; 第7节是本文的总结.

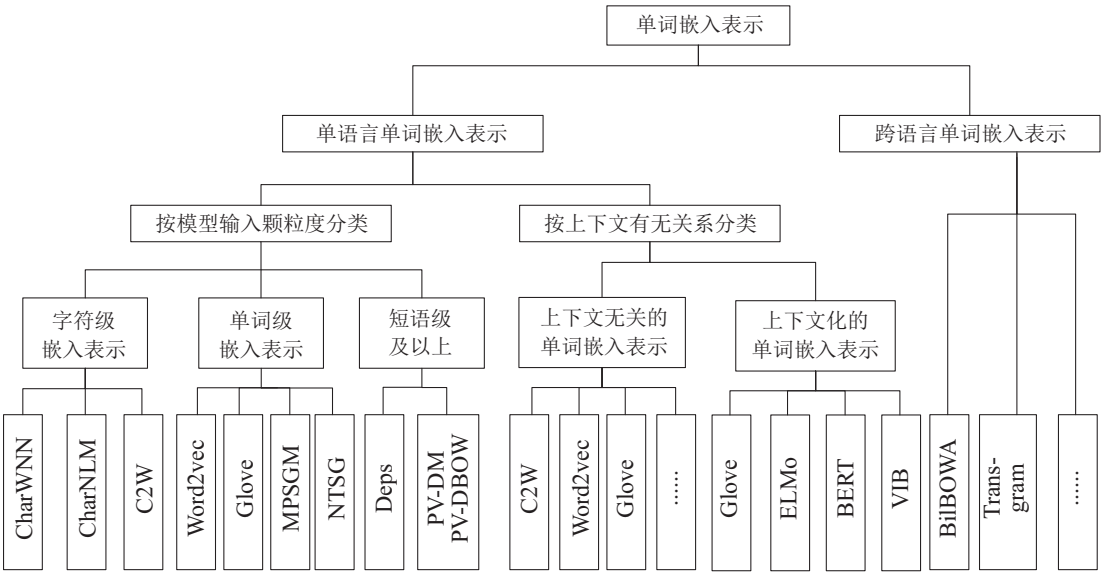


图 1 单词嵌入表示分类

Fig. 1 Classification of word embeddings

2 按模型输入颗粒度分类

单词嵌入表示是自然语言处理的重要方式, 但单词嵌入表示通常使用神经网络学习并捕获关于单词的句法和语义信息. 用单词嵌入表示学习单词表示时, 通常会忽略有关单词形态和形状的信息. 但对于词性标注这样的任务, 特别是在处理形态丰富的语言时, 单词内信息非常有用. 字符级单词嵌入表示考虑到了上述问题, 现对其介绍如下.

2.1 字符级单词嵌入表示模型

2.1.1 CharWNN

Dos Santos等人[16]在深度神经网络结构(deep neural network, DNN)基础上提出了字符级单词神经网络(character-level word neural network, CharWNN), CharWNN能够学习单词的字符级嵌入表示, 并将它们与常用的单词级表示相关联, 进行词性标注(part–of–speech, POS)等任务. CharWNN新颖之处在于使

用卷积层捕捉字符级嵌入表示, 能够从任意长度的单词中提取有效的特征表示. 比如, 在进行词性标注时, 卷积层为每个单词生成字符级嵌入表示, 包括超出词汇表范围的词(out of vocabulary, OOV), 实现过程如下.

网络的第1层是表示初始化, 将单词转换为特征向量(嵌入表示形式), 用于捕获关于单词的形态、句法和语义信息. 假定W是与训练文本有关且单词数量固定的单词词汇表, 单词由固定大小的字符词汇表C^{cclr}里的字符组成, 给定一个由N个单词$\{w_1, w_2, \cdots, w_N\}$组成的句子, 每个单词$w_n \in W$都被转换为一个向量$\boldsymbol{u}_n$, $\boldsymbol{n}_\pi = \{\boldsymbol{r}^w, \boldsymbol{r}^{\mathrm{otr}}\}$, $\boldsymbol{r}^w \in \mathbb{R}^{d^w}$为$w_n$的单词级嵌入表示, $\boldsymbol{r}^{\mathrm{chr}} \in \mathbb{R}^{\mathrm{clu}}$为$w_n$的字符级嵌入表示. 单词级嵌入表示旨在捕获语法和语义信息, 字符级嵌入表示可捕获形态和形状信息, 公式如下:

$$\boldsymbol{r}^n = Q^n \boldsymbol{v}^n, \tag{1}$$

$$\boldsymbol{r}^{\mathrm{chr}} = Q^{\mathrm{ckr}} \boldsymbol{v}^c, \tag{2}$$

其中: $Q^w \in \mathbb{R}^{d^w \times |W|}$和$Q^{\mathrm{chr}} \in \mathbb{R}^{d^{\mathrm{chr}} \times |C^{\mathrm{chr}}|}$是嵌入矩阵, $\boldsymbol{v}^w$和$\boldsymbol{v}^c$分别是单词和字符的独热表示, 如图2所示, 利用卷积层在单词的每个字符周围产生局部特征, 然后使用最大池化操作, 将局部特征组合在一起得到该单词的字符级嵌入表示.

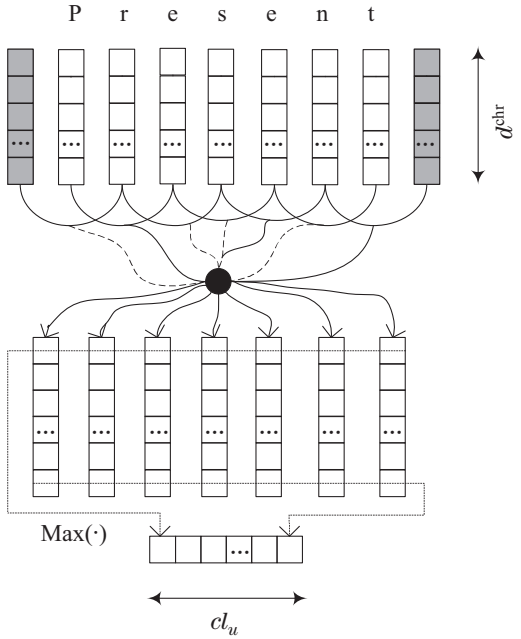


图 2　字符级特征提取的卷积方法

Fig. 2　Convolution method for CharWNN

字符级嵌入表示序列作为卷积层的输入. 卷积层使用大小为k^{chr}的连续窗口, 对在每个窗口内的字符序列应用矩阵向量运算得到$\{\boldsymbol{r}_1^{\mathrm{chr}}, \boldsymbol{r}_2^{\mathrm{chr}}, \cdots, \boldsymbol{r}_M^{\mathrm{chr}}\}$. 定义向量$\boldsymbol{z}_m \in \mathbb{R}^{d^{\mathrm{chr}} k^{\mathrm{chr}}}$为$m$个字符的嵌入表示串联形成的增广向量, $\boldsymbol{z}_m$有$(k^{\mathrm{chr}}-1)/2$个左近邻, $(k^{\mathrm{chr}}-1)/2$个右近邻:

$$\boldsymbol{z}_m = (\boldsymbol{r}_{m-(k^{\mathrm{chr}}-1)/2}^{\mathrm{chr}} \ \cdots \ \boldsymbol{r}_{m+(k^{\mathrm{chr}}-1)/2}^{\mathrm{chr}})^{\mathrm{T}}, \tag{3}$$

用卷积层计算单词w_n的字符级嵌入表示$\boldsymbol{r}^{\mathrm{chr}}$的第$j$个元素, 即

$$[\boldsymbol{r}^{\mathrm{chr}}]_j = \max_{1<m<M} [W^0 \boldsymbol{z}_m + \boldsymbol{b}^0]_j, \tag{4}$$

其中$W^0 \in \mathbb{R}^{cl_u \times d^{\mathrm{chr}} k^{\mathrm{chr}}}$是卷积层的权重矩阵. 用相同的矩阵提取给定单词的每个字符的局部特征, 为单词提取固定大小的“全局”特征向量, 其维数为该单词所有字符窗口中的最大值作为该向量的维数. 矩阵W^{chr}和W^0以及向量d^{chr}是要学习的参数. 字符向量的维数$\boldsymbol{b}^0$, 卷积层单元的个数cl_u (即单词的字符级嵌入表示的维数)和字符上下文窗口大小k^{chr}是由用户选择的超参数. 之后再进行评分、结构化推理以及网络训练.

Dos Santos等人[16]的实验结果显示该方法在英语和葡萄牙语语料库中的词性标注上有明显改善. 字符级表示的词性标注的主要创新之处有: 1) 提出了利用卷积神经网络提取字符级特征, 并将其与单词级特征联合, 用于词性标注的思想; 2) 证明使用相同的模型训练不同语言的POS标签是可行的, 可自动学习特征, 无需人工干预. 这种策略也可应用到其它自然语言处理任务中.

2.1.2　CharNLM

Kim等人[17]提出了字符感知神经语言模型(character-aware neural language models, CharNLM), 是一个将只依赖于字符级的向量表示作为模型的输入, 预测输出仍为单词级的神经语言模型, 该模型包含卷积神经网络(convolutional neural network, CNN)[18]、高速公路网 (highway network)[19]、长短时记忆 (long short-term memory, LSTM)[20] 和循环神经网络语言模型 (recurrent neural network language model, RNN–LM)[21]. 采用卷积神经网络(CNN)和高速公路网对输入的字符进行优化处理, 处理后输出到长短时记忆(LSTM)递归神经网络语言模型(RNN–LM)中.

1)　字符级卷积神经网络CharCNN.

设C^{chr}为字符词汇表集合, 字符词汇表中包含的字符的个数记为$|C^{\mathrm{chr}}|$, d为字符嵌入表示向量的维数, $Q^{\mathrm{chr}} \in \mathbb{R}^{d \times |C^{\mathrm{chr}}|}$为字符嵌入表示矩阵, W为与训练文本有关且单词个数固定的词汇表. 假设单词$w \in W$由字符序列$\{\boldsymbol{c}_1, \cdots, \boldsymbol{c}_l\}$组成, 其中, l为单词w的长度, 单词w的字符级表示矩阵为$M_w^{\mathrm{chr}} \in \mathbb{R}^{d \times l}$, 矩阵的第$j$列对应的是字符嵌入表示$\boldsymbol{c}_j$ (即Q^{chr}的第j列). 在M_w^{chr}和宽度为w^w的滤波器$H \in \mathbb{R}^{d \times w^w}$之间应用一个窄卷积, 之后加入一个偏置$\boldsymbol{b}$, 应用双曲正切激活函数得到特征映射$f^w \in \mathbb{R}^{l-m+l}$, f^w中第i个元素定义如下:

$$f^w[i] = \tanh(\langle M_w^{\mathrm{chr}}[*, i:i+k-1], H\rangle + \boldsymbol{b}), \tag{5}$$

这里$M_w^{\mathrm{chr}}[*, i:i+k-1]$是$M_w^{\mathrm{chr}}$中任意行的第$i$到$i+$

$k-1$列, $\langle A,B\rangle=\mathrm{Tr}(AB^{\mathrm{T}})$是弗罗贝尼乌斯内积. 学习的目标函数$y^w$作为单词$w$对应于滤波器$H$的特征, 即

$$y^w=\max_i f^w[i]. \tag{6}$$

2) 高速公路网.

CharCNN的输出y^w作为高速公路网[7]的输入, 利用Srivastava等人[22]提出的方法, 在多层感知器(multilayer perceptron, MLP)中的一层使用仿射变换, 再利用非线性激活函数构造一组新的特征, 即

$$z=g(Wy^w+b), \tag{7}$$

用高速公路网的其中一层实现以下变换:

$$z=t\odot g(W_Hy^w+b_H)+(1-t)\odot y^w, \tag{8}$$

其中: g是非线性激活函数, $t=\sigma(W_Ty^w+b_T)$为变换门, $(1-t)$是移位门, 构造的y^w, z维数要一样, W_H和W_T是方阵, σ是作用在预激活函数的分量上的Sigmoid函数, $\odot$是向量的分量上的乘积运算. 模型示意图见图3.

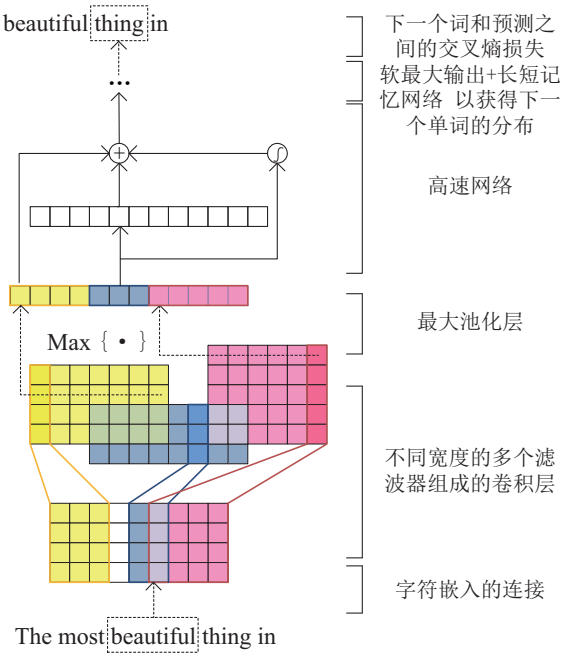


图 3 字符识别的神经语言模型

Fig. 3 CharCNN

Kim等人[17]的实验结果表明, 该模型在减少了很多参数的情况下, 困惑度PPL (perplexity)性能仍优于之前大多数NLMs, 这也证明了该模型能够仅从字符编码, 得到丰富的语义和拼写特征(orthographic features), 使用CharCNN和高速公路网络层进行表示学习仍然是未来研究单词嵌入表示的一个方向.

2.1.3 C2W

Ling等人[23]提出了基于双向长短时记忆网络LSTM (Bi–LSTMs)[24]的字符到单词的组合模型(character to word, C2W), Ling等人假设单词拼写间具有相似性, 则其语义、句法功能也具有相似性, 每个字符类型与向量表示相关联, 利用长短时记忆(LSTM)非线性地学习序列的隐表示. 这个模型学习的向量模型维数低, 仅使用了双向LSTM来读取构成每个单词的字符序列, 并将它们组合成单词的向量表示.

C2W能够学习序列模型中复杂的非局部依赖关系, 图4是一个示例, C2W模型的输入(见下部分图)是一个单词w, 希望获得用于表示单词w的d维向量. 该模型共享同一个单词查找表的输入和输出(见上半部分图), 定义字符表C^{chr}. 对于英语, 字符表包含每个字母的大写和小写, 以及数字和标点符号.

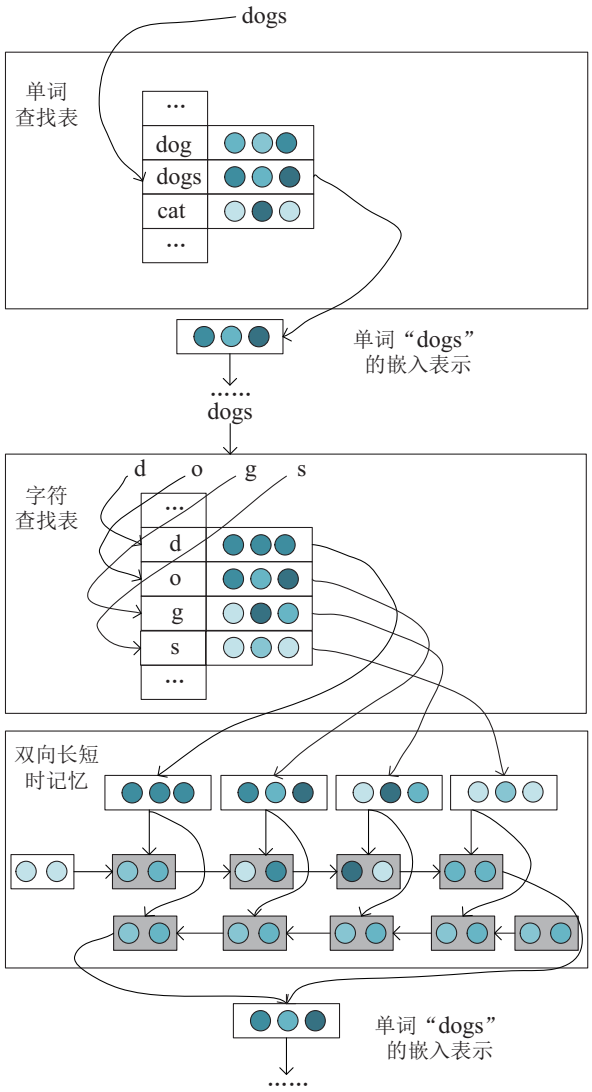


图 4 字符到单词的组合模型

Fig. 4 C2W

注: 单词查找表(顶部)和词汇组合模型(底部), 方框表示神经元激活的向量, 阴影框表示非线性

该输入单词w被分解为字符序列$\{c_1,c_2,\cdots,c_l\}$, 其中l是输入单词w的长度. 将每个c_i定义为一个独热表示1_{c_i}, 表示字符c_i出现在词汇表W位置为1, 其余为

0. 定义投影层$P_C \in \mathbb{R}^{d_c \times |C^{\mathrm{chr}}|}$, 其中$d_c$是字符表$C^{\mathrm{chr}}$中每个字符的隐表示的参数个数. 则每个输入字符$c_i$的投影可以写为

$$\boldsymbol{e}_{c_i} = P_C \cdot 1_{c_i}. \tag{9}$$

给定单词输入向量$\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_l\}$, 用LSTM计算状态序列$\{\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_{l+1}\}$, 即

$$\begin{aligned}
&\boldsymbol{i}_t = \sigma(M_{ix}\boldsymbol{x}_t + M_{ih}\boldsymbol{h}_{t-1} + M_{ic}\boldsymbol{c}_{t-1} + \boldsymbol{b}_i),\\
&\boldsymbol{f}_t = \sigma(M_{fx}\boldsymbol{x}_t + M_{fh}\boldsymbol{h}_{t-1} + M_{fc}\boldsymbol{c}_{t-1} + \boldsymbol{b}_f),\\
&\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} +\\
&\qquad \boldsymbol{i}_t \odot \tanh(M_{cx}\boldsymbol{x}_t + M_{ch}\boldsymbol{h}_{t-1} + \boldsymbol{b}_c),\\
&\boldsymbol{o}_t = \sigma(M_{ox}\boldsymbol{x}_t + M_{oh}\boldsymbol{h}_{t-1} + M_{oc}\boldsymbol{c}_{t-1} + \boldsymbol{b}_o),\\
&\boldsymbol{h}_t = \boldsymbol{o}_t \odot \tanh \boldsymbol{c}_t,
\end{aligned} \tag{10}$$

这里σ是作用在预激活函数分量上的Sigmoid函数, $\odot$是向量分量的乘积运算. LSTM定义一个额外的记忆存储器单元$\boldsymbol{c}_t$. 从$\boldsymbol{c}_{t-1}$传播到$\boldsymbol{c}_t$的信息由$\boldsymbol{i}_t$, $\boldsymbol{f}_t$, $\boldsymbol{o}_t$三个门控制, 决定了包含输入$\boldsymbol{x}_t$哪些内容, 包含从$\boldsymbol{c}_{t-1}$中要遗忘的内容, 以及与当前状态$\boldsymbol{h}_t$有关的内容. 用M表示LSTM中的全部参数(M_{ix}, M_{fx}, $\boldsymbol{b}_f \cdots$). 给定一个字符表示序列$\{\boldsymbol{e}^c_{c_1}, \cdots, \boldsymbol{e}^c_{c_l}\}$作为输入序列, 正向LSTM接收正向序列, 产生状态序列$\{\boldsymbol{s}^f_0, \cdots, \boldsymbol{s}^f_l\}$的同时, 反向LSTM接收反向序列, 产生状态序列$\{\boldsymbol{s}^b_0, \cdots, \boldsymbol{s}^b_l\}$, 两个LSTM使用一组不同的参数$M^f$和$M^b$. 单词$w$的表示是通过组合前向和后向的状态获得的

$$\boldsymbol{e}^C_w = D^f \boldsymbol{s}^f_l + D^b \boldsymbol{s}^b_o + \boldsymbol{b}_d, \tag{11}$$

这里D^f, D^b, $\boldsymbol{b}_d$是决定状态如何组合的参数.

Ling等人[23]的实验表明: C2W模型将字符作为基本单元来生成单词的嵌入向量表示, 它可感知单词内的字符变化, 在POS标记方面, 使用单独字符的模型仍然可以获得与较先进系统相当或更好的结果, 且无需人为进行特征选择.

2.1.4　小结与纵向分析

本节将输入为字符级的单词嵌入表示模型总结在一起, 上述各模型的优缺点、评价方法、数据集和应用领域的总结如表1所示.

表 1　字符级单词嵌入表示模型总结

Table 1　Summary of character-level word embedding model

模型	优缺点	数据集	评价方法	性能	应用领域
CharNLM	应用了字符集卷积神经网络, 仅对字符编码, 参数少, 语义丰富	English PTB-s English PTB-L	PPL PPL	92.3 78.9	词性标注等
CharWNN	字符与单词相结合, 准确率更高	English PTB Mac-Morpho Corpus	ACC ACC	97.32 97.47	词性标注等
C2W	在缺乏拼写线索的语言中也能够学习得很好	English PTB	ACC	97.78	词性标注等

注: CharNLM在相同数据集上性能优于只考虑单词级嵌入表示和在其之前的模型(如: Mikolov等人[26]的RNN–LDA)

在NLP中, 字符级单词嵌入表示对于改进表示学习的泛化能力非常重要, 最重要的是向量空间模型, 能够捕获语义和句法功能在几何特性方面的局部相似性. 字符级嵌入表示是在NLM或LSTM模型基础上进行改进的, 以组成单词的字符作为模型的输入, 减少了模型的参数, 同时在处理形态丰富的语言时, 能够更好地利用单词内信息, 能够更好地完成词性标注等自然语言处理任务.

这种利用字符级嵌入表示提高模型效率的方法还有很多, 比如, Chen等人[25]提出的字符增强的单词嵌入模型(character-enhanced word embedding model, CWE), CWE为汉字歧义提供了一种有效的解决方案, 同时也可将CWE扩展到英语等语言中.

2.2　单词级单词嵌入表示模型

2.2.1　Word2Vec模型

Mikolov等人在NLM, RNN–LM等模型的基础上, 于2013年在文献[14]中提出了Word2Vec模型, 包括Skip–Gram和CBOW两个模型, 旨在得到更好的单词嵌入表示向量, 使它们与之前的NLM模型相比, 减少非线性隐层, 大大降低模型的计算复杂度.

1) Skip–Gram模型.

Skip–Gram模型结构如图5右图所示, 该模型旨在利用中心单词预测上下文单词. 给定一个要训练的单词序列$w_1, w_2, \cdots, w_T$, Skip–Gram模型的目标是使以下平均对数概率最大化:

$$\begin{aligned}
&\max L(D) =\\
&\max \frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leqslant j\leqslant c, j\neq 0} \log p(w_{t+j}|w_t),
\end{aligned} \tag{12}$$

其中c是训练上下文的大小.

基本Skip–Gram模型中, 假设中心单词是w_I, 它的一个相邻单词是w_O, 使用软最大函数定义$P(w_O|w_I)$:

$$P(w_O|w_I) = \frac{\exp(\boldsymbol{u}^{\mathrm{T}}_{w_o}\boldsymbol{v}_{w_I})}{\sum\limits_{w\in W}\exp(\boldsymbol{u}^{\mathrm{T}}_{w}\boldsymbol{v}_{w_I})}, \tag{13}$$

其中: W表示词汇表, $\boldsymbol{u}_w \in \mathbb{R}^d$和$\boldsymbol{v}_w \in \mathbb{R}^d$表示单词$w$的$d$维输出、输入嵌入向量表示, 所有单词的输入输出向量都是要学习的参数, 即$U = \{\boldsymbol{u}_w | w \in W\}$, $V = \{\boldsymbol{v}_w | w \in W\}$, Skip–Gram模型是3层神经网络, 其中$U$和$V$表示神经网络的两个连接边参数矩阵.

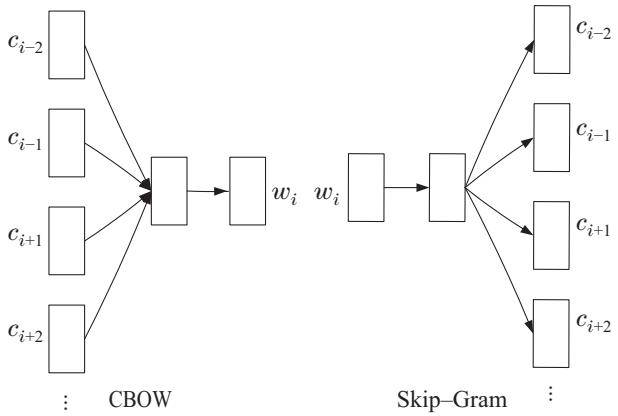


图 5 CBOW和Skip–Gram模型结构

Fig. 5 CBOW&Skip–Gram

2) CBOW模型.

CBOW的模型结构如图5左图所示, 其目标是在滑动窗口中给定上下文单词, 预测目标单词. 形式上, 给定一个单词序列$D_w = \{w_1, \cdots, w_M\}$, M为单词序列长度, CBOW的目标是使以下平均对数条件后验概率最大化:

$$\max L(D) = \max \frac{1}{M} \sum_{i=K}^{M-K} \log P(w_i | w_{i-K}, \cdots, w_{i+K}), \quad (14)$$

这里K是目标单词的上下文窗口的大小. CBOW用软最大函数表示条件后验概率$P(w_i|c)$, c为目标查询单词w_i前后各K个单词组成的上下文单词序列, 即$P(w_i|w_{i-K}, \cdots, w_{i+K})$,

$$P(w_i|c) = P(w_i|w_{i-K}, \cdots, w_{i+K}) = \frac{\exp(\boldsymbol{w}_o^{\mathrm{T}} \cdot \boldsymbol{w}_i)}{\sum\limits_{w' \in W} \exp(\boldsymbol{w}_o^{\mathrm{T}} \cdot \boldsymbol{w}_i')}, \quad (15)$$

其中: W是字典中所有单词集合, $\boldsymbol{w}_i$是目标单词的向量表示, $\boldsymbol{w}_o$是所有上下文单词向量的平均值:

$$\boldsymbol{w}_o = \frac{1}{2K} \sum_{j=i-K, \cdots, i+K_{j \neq i}} \boldsymbol{w}_j. \quad (16)$$

2.2.2 Glove模型

CBOW模型和Skip–Gram模型是单词嵌入表示发展中的一个里程碑. 后续有大量试图学习更好的单词嵌入表示的研究成果, 比如, 全局向量(global vector, Glove)[27].

为了充分利用语料库中的单词共现统计信息, Glove模型直接获取这些全局信息. 用X表示单词共现计数矩阵, 用X_{ij}表示w_j在w_i上下文中出现的次数, 则损失函数定义如下:

$$L = \sum_{i,j=1}^{|V|} g(\boldsymbol{X}_{ij})(\boldsymbol{C}(w_i)^{\mathrm{T}} \boldsymbol{C}(w_j) + \boldsymbol{b}_i + \boldsymbol{b}_j - \log \boldsymbol{X}_{ij})^2, \quad (17)$$

其中: $|V|$为词汇量的大小, $g(x)$为权重函数, 用于缓解罕见词和常见词间的数据不平衡问题, 矩阵$\boldsymbol{C} \in \mathbb{R}^{|V| \times d}$作为映射, 其中每一行表示单词$w$在词汇表$V$中的分布向量表示, 记为$\boldsymbol{C}(w)$. 但是这个模型仍然没有解决单词的一词多义的问题.

2.2.3 多原型单词嵌入表示学习的概率模型

Tian等人[28]提出了一种对一个多义词学习多个嵌入向量表示的有效方法, 他们首先从概率的角度对单词的多义性进行建模, 并将其与连续Skip–Gram模型相结合, 称为多原型Skip–Gram模型(multi-prototype skip-gram model, MPSGM), 类似于Skip–Gram模型[14], 用条件后验概率$P(w_O|w_I)$表示多个原型的Skip–Gram模型, 并使用输入输出矩阵对条件后验概率$P(w_O|w_I)$进行参数化, 不同之处在于, 给定输入单词w_I, 输出单词w_O的条件后验概率为一个有限混合模型, 其中每个混合分量对应于输入单词w_I的原型. 即, 假设单词w具有N_w个原型, $h_w \in \{1, \cdots, N_w\}$是单词$w$原型的索引. 条件后验概率$P(w_O|w_I)$可展开为

$$P(w_O|w_I) = \sum_{i=1}^{N_{v_I}} P(w_O|h_{w_I} = i, w_I) P(h_{w_I} = i|w_I) = \quad (18)$$

$$\sum_{i=1}^{N_{v_I}} \frac{\exp(\boldsymbol{u}_{w_o}^{\mathrm{T}} \boldsymbol{v}_{w_I,i})}{\sum\limits_{w \in W} \exp(\boldsymbol{u}_w^{\mathrm{T}} \boldsymbol{v}_{w_I,i})} P(h_{w_I} = i|w_I), \quad (19)$$

其中: $\boldsymbol{v}_{w_I,i} \in \mathbb{R}^d$是输入单词$w_I$的第$i$个原型的嵌入向量表示. $P(w_O|w_I)$是输入单词w_I的某个原型出现时, 观测到输出单词w_O的概率的加权平均值. $P(w_O|h_{w_I} = i, w_I)$采用与式(13)相似的软最大函数, 是输入单词w_I在每个原型中的先验概率.

式(19)中的分母$\sum\limits_{w \in W} \exp(\boldsymbol{u}_w^{\mathrm{T}} \boldsymbol{v}_{w_I,i})$与$|W|$有线性依赖关系, 导致计算量较大. 为此, 引用软最大分层树[29–30]的方法, 假设每个单词是二叉树的叶子节点, 输出单词与二进制向量相关联, 指定二叉树的根到叶子节点的路径, 其中是向量的维数, 则条件概率可表示为

$$P(w_O|h_{w_I} = i, w_I) = \prod_{t=1}^{L_{w_0}} P(b_t^{(w_o)}|w_I, h_{w_I} = i) = \prod_{t=1}^{L_{r_0}} \zeta(b_t^{(w_o)} \boldsymbol{u}_{w_o,t}^{\mathrm{T}} \boldsymbol{v}_{w_I,i}), \quad (20)$$

其中: $\zeta(x)=1/(1+\exp(-x))$, $\boldsymbol{u}_{w_o,t}$特指从二叉树根到叶子节点w_O的第t个节点相关的d维参数向量. 用式(20)代入式(18), 得到更实用的概率形式, 可以避免式(19)中复杂的软最大操作. 之后再用EM算法训练这个多原型的Skip–Gram模型. MPSGM弥补了原有Word2Vec模型的不足, 考虑了单词的多义性, 同时与以前的基于聚类的多原型算法相比, 该模型提出的概率框架除了能学习单词嵌入表示外, 还避免了额外的聚类工作, 算法复杂性大大降低.

MPSGM弥补了原有Word2Vec模型的不足, 考虑了单词的多义性, 同时与以前的基于聚类的多原型算法相比, 该模型提出的概率框架除了能学习单词嵌入表示外, 还避免了额外的聚类工作, 算法复杂性大大降低.

2.2.4 基于神经张量Skip–Gram模型的主题词嵌入学习

Liu等人[31]提出了神经张量Skip–Gram模型(neural tensor skip-gram model, NTSG), 该模型是Skip–Gram模型的扩展, 可以同时对单词和主题之间的交互作用关系进行建模, 在单词相似性和文本分类任务的实验中表现良好. 为了提高单词嵌入表示学习的表示能力, 引入了单词的潜在主题, 并假设每个单词在不同的主题下具有不同的嵌入表示. 具体实现过程如下.

NTSG用张量层替换了双线性层, 对Skip–Gram模型进行扩展, 以捕捉不同语境下单词与话题之间的相互作用关系. 为了计算单词w和它的主题t在特定上下文c中的相似度, 使用以下基于能量的函数:

$$g(\boldsymbol{w},c,\boldsymbol{t})=\boldsymbol{u}^{\mathrm{T}}f(\boldsymbol{w}^{\mathrm{T}}M^{[1:k]}\boldsymbol{t}+C^{\mathrm{T}}(\boldsymbol{w}\oplus\boldsymbol{t})+\boldsymbol{b}_c), \tag{21}$$

其中: $\boldsymbol{w}\in\mathbb{R}^d$, $\boldsymbol{t}\in\mathbb{R}^d$是单词$w$和主题$t$的向量表示, $\oplus$是串联叠加运算, $w\oplus t=\begin{bmatrix}w\\t\end{bmatrix}$, $M^{[1:k]}\in\mathbb{R}^{d\times d\times k}$是张量, 所有上下文使用相同的张量, 双线性张量积以两个向量, $\boldsymbol{w}\in\mathbb{R}^d$, $\boldsymbol{t}\in\mathbb{R}^d$作为输入, 生成一个$k$维的短语向量($\boldsymbol{z}$)作为输出

$$\boldsymbol{z}=\boldsymbol{w}^{\mathrm{T}}M_c^{[1\cdot k]}\boldsymbol{t}, \tag{22}$$

其中($\boldsymbol{z}$)的每一项是由张量的每一个切片$i=1,\cdots,k$得到的

$$\boldsymbol{z}_i=\boldsymbol{w}^{\mathrm{T}}M_c^{[i]}\boldsymbol{t}. \tag{23}$$

式(21)中的其它参数是神经网络的标准形式: $\boldsymbol{u}\in\mathbb{R}^k$, $C\in\mathbb{R}^{k\times(2d)}$, $\boldsymbol{b}_c\in\mathbb{R}^k$, $f(t)=\dfrac{1}{1+\exp(-t)}$是作用在元素上的标准的Sigmod非线性激活函数, 与Skip–Gram模型中一样.

图6展示了NTSG模型结构. 它的主要优点是能够同时考虑词语、话题和语境之间的潜在关系, 即, 引入的张量可以表示单词和主题之间的相互作用关系.

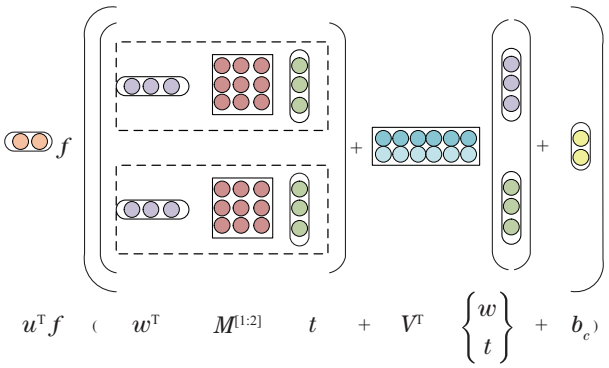


图 6 NTSG模型说明图

Fig. 6 NTSG

为了避免过拟合, 进行张量分解, 将每个张量切片分解为两个低秩矩阵的乘积. 每个张量切片$M^{[i]}\in\mathbb{R}^{d\times d}$被分解成两个低秩矩阵$P^{[i]}\in\mathbb{R}^{d\times r}$和$Q^{[i]}\in\mathbb{R}^{d\times r}$, 记为

$$M^{[i]}=P^{[i]}Q^{[i]},\ 1\leqslant i\leqslant k, \tag{24}$$

其中$r\ll d$是因子个数. 基于能量的函数变为

$$g(\boldsymbol{w},c,\boldsymbol{t})=\\ \boldsymbol{u}^{\mathrm{T}}f(\boldsymbol{w}^{\mathrm{T}}P^{[1:k]}Q^{[1:k]}\boldsymbol{t}+C^{\mathrm{T}}(\boldsymbol{w}\oplus\boldsymbol{t})+\boldsymbol{b}_c). \tag{25}$$

此时张量运算的复杂度是$\mathrm{O}(rdk)$. 只要r足够小, 分解后的张量运算就会比未分解的张量运算快得多, 自由参数的数目也会小得多, 这就防止了模型的过拟合. Liu 等人[31] 的实验表明, NTSG 比单一的 Skip–Gram模型在性能上提升很多, 很好地将单词与在不同上下文主题中的嵌入表示结合起来, 从而获得每个单词类型的上下文主题词嵌入表示, 使用张量分解的方法也提高了模型的效率, 在上下文相似度和文本分类两个任务中优于先前的多原型模型.

2.2.5 小结与纵向分析

与至少使用4个隐层(包括查找表层)的传统神经网络相比, Word2Vec模型大大减少了参数的数量, 在训练效率方面带来了显著的改进, Glove模型也在NLP领域取得了巨大成功, 但是因为在这些模型中, 每个词都是用一个不随上下文变化的原型向量来表示的, 所以多义词问题没有得到解决. 在这种情况下, 对于一个多义词来说, 在给定上下文的情况下无法区分其确切的含义. 何为多义词呢? 多义词在表示学习中又是怎样体现的呢? 比如, 当单词bank指的是“河岸”时, 观测到上下文单词可能是river, water 和slope等; 然而, 当bank指的是“银行”时, 观测到的上下文单词有可能是money, account和investment等.

一些学者也试图通过聚类词的上下文窗口特征表示来训练单词的多个原型的嵌入表示, 但训练参数庞大, 数据集大时, 算法的可扩展性有限, 学习效率也不

高, 如文献[32]指出的那样. 随后多原型Skip–Gram模型引入概率框架, 摒弃了聚类思想, 提高了模型的效率. 而NTSG的目标就是判断出一个单词w和它的主题t在上下文c中能否很好的匹配. 例如, (w,t) = (apple, company)能在上下文c = iphone下很好的匹配(w,t) = (apple, fruit)能在上下文c = banana下很好的匹配. 单词级单词嵌入表示的模型的介绍与总结如表2.

2.3 短语级及短语以上级的单词嵌入表示模型

2.3.1 基于句法依赖关系的单词嵌入表示

Skip–Gram模型具有很多优点, 但是在Skip–Gram算法中, 上下文词汇表C与词汇表W是相同的, 在此基础上, Levy等人[33]提出的基于句法依赖关系的单词嵌入表示方法 (dependency-based word embeddings, Deps), 将线性BOW模型上下文单词转换为任意词语的上下文单词, 使得上下文单词不只使用所学习的单词前后出现的单词, 上下文类型的数量可以远远超过所要学习单词前后出现的单词的个数.

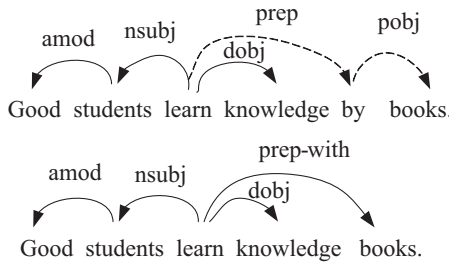
该基于句法依赖关系的模型利用单词袋子模型根据单词的句法关系得出语境相关的句子解析[16,24], 对句子解析之后, 得到单词的语境上下文单词. 对于一个目标单词w, 它修饰的单词记为$m_1,\cdots,m_k$, 修饰它的单词, 记为头部h, 则目标单词的上下文记为: $(m_1,lbl_1),\cdots,(m_k,lbl_k),(h,lbl_h^{-1})$, lbl表示目标单词与被修饰词的之间的句法依赖关系类型, 如, nsubj, dobj, prep with, amod, lbl^{-1}是lbl的倒数, 表示修饰目标单词的修饰语与目标单词的依赖关系, 即反向依赖关系. 在这里, 介词在提取上下文时就直接被包含在修饰词与被修饰词的依赖关系中. 图7给出了考虑句法依赖关系后, 上下文提取的一个例子.

表 2 单词级单词嵌入表示模型总结

Table 2 Summary of word-level word embedding model

模型	优缺点	数据集	评价方法	性能	应用领域
Word2Vec	减少一层网络, 参数减少, 训练效率高, 性能好	SCWS	$\rho\times100$	61.7	NLP任务
Glove	充分利用语料库中的单词共现统计	SCWS	$\rho\times100$	53.9	NLP任务
MPSGM	是对Word2Vec模型的优化, 改进多义词问题	SCWS	$\rho\times100$	65.3	NLP任务
NTSG	同时对单词和主题之间的交互进行建模, 在NLP任务的实验中表现良好	SCWS	$\rho\times100$	69.5	NLP任务

注: ρ表示斯皮尔曼相关系数



单词	上下文
good	students / amod^{-1}
students	good / amod, learn / nsubj^{-1}
learn	students / nsubj, knowledge / dobj, books / prep_with
knowledge	learn / dobj^{-1}
books	learn / prep_with^{-1}

图 7 基于句法依赖关系的上下文提取示例

Fig. 7 Dependency-based context extraction example

注: 上图: 介词关系被添加到修饰语与被修饰语集合中, books直接被learn修饰的单词; 下图: 为句子中每个单词提取上下文单词及句法依赖关系, 记录格式如图所示

基于句法依赖关系的模型比BOW模型包含更大范围的词语之间的依赖关系, 并且还能过滤掉窗口内与目标词没有直接关系的单词. 该模型能够利用句法上下文依赖关系产生更丰富语义信息, 更全面的单词嵌入表示, 捕获更多语义功能相似性, 学习的语言模型包含更少的主题相似性.

2.3.2 段落向量模型PV–DM&PV–DBOW

Le, Mikolov等人在文献[34]中, 提出了一种无监督的段落向量表示学习算法, 它可以从句子、段落等长度可变的文本片段中学习固定长度的特征表示, 该算法是用一个实数向量表示每个文档, 经过训练的段落向量可以预测文档中的单词, 这个算法可以克服BOW模型丢失单词的顺序和忽略单词的语义这两个缺点. 初始模型为分布式记忆段落向量模型(distributed memory model of paragraph vectors, PV–DM), 是将段落向量作为另一个向量插入到标准语言模型中, 旨在捕获文档的主题, 在这个模型中, 单词列向量组成的矩阵W在各个段落中是共享的, 利用随机梯度下降方法训练段落向量和单词向量, 并通过反向传播得到梯度. 在随机梯度下降计算的每一步中, 都可以从随机段落中抽取一个固定长度的上下文样本, 计算网络误差梯度, 并用这个梯度来更新模型中的参数.

在预测时, 通过梯度下降过程学习计算新段落的段落向量. 在此时, 剩余模型的参数, 神经网络连接边矩阵W和软最大权重是固定的, 为之前训练过程学习

得到的值, 虽然模型参数的个数可能很大, 但是训练期间的更新通常是稀疏的, 因此模型训练效率很高.

经过训练后, 段落向量可以用作新输入文本的段落特征向量, 将这些特征直接作为传统机器学习算法的输入, 如逻辑斯蒂回归、支持向量机或K–均值聚类(K-means)算法, 即让段落向量与局部上下文单词向量相关联, 或用K–均值聚类方法, 预测下一个单词. 如图8所示.

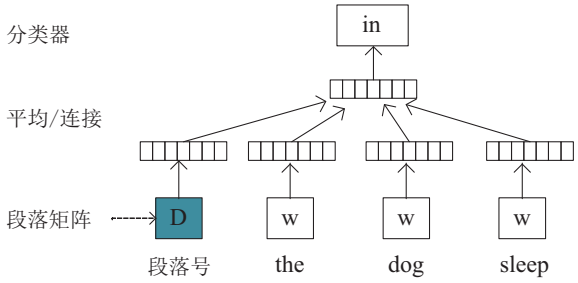


图 8　分布式存储段落向量模型

Fig. 8　A framework for learning paragraph vector

在预测任务中如果不使用局部上下文信息时, 可以进一步简化段落向量, 即分布式单词袋子版的段落向量模型(distributed bag of words version of paragraph vector, PV–DBOW), 如图9所示.

这个模型具有概念简单、参数少的优点, 只需要存储软最大权值, 比之前的模型中减少了单词向量的存储空间, 利用反向传播对段落向量进行调优.

2.3.3　小结与纵向分析

Deps模型充分利用句法依赖关系, 使单词嵌入表示更加全面. 段落向量将单句扩展到多句子, 采用无监督的框架, 可以减少数据标记的工作. PV–DM和PV–DBOW算法用实数向量表示每个文档, 该向量被训练用来预测文档中的单词, 假定文章由段落组成, 每一个段落又由向量表示, 使得整个文章的各段落列向量组成矩阵D, 段落中的单词列向量组成矩阵W, 文章中的多个段落向量的平均或者叠加增广向量, 以及段落中的多个单词向量的平均或者叠加增广向量, 可以用来预测下一个上下文中的单词的向量表示, 亦可作为后续分类或聚类算法的输入. 这个方法可以在一定程度上克服了BOW模型的缺点, 并且在实验中, 段落向量在文本表示方面优于BOW模型和其它单词模型. 像这种短语级及以上的单词嵌入表示还有很多, 比如, 短语嵌入的合成模型学习[35], 在这里只是列举了两个, 对比如表3.

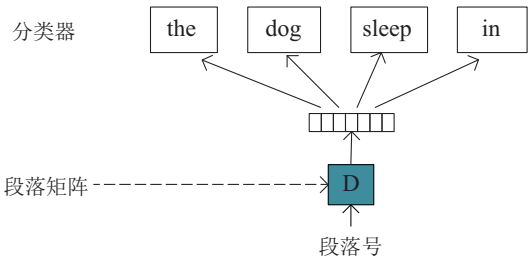


图 9　分布式无序袋子段落模型

Fig. 9　Distributed bag-of-words version of paragraph vectors

表 3　短语级及以上的嵌入表示模型总结

Table 3　Summary of embedding models at phrase level and above

模型	优缺点	数据集	评价方法	性能		应用领域
Deps	将Skip–Gram模型线性单词袋子模型上下文替换为任意上下文	WordSim353等	余弦相似度	batman	superman superboy supergirl catwoman aquaman	主题相似性 文本分类等
PV–DM PV–DBOW	段落向量在捕获段落语义方面有优势	Wikipedia	错误率	7.42%		捕捉段落语义等

3　按上下文关系分类

3.1　上下文无关的单词嵌入表示模型

第2.1节和第2.2节中介绍的模型很多都没有引入上下文信息, 如, Word2Vec, Glove, CharWNN等模型产生的单词嵌入表示都未考虑上下文信息, 虽然它们在单词嵌入表示学习中都是里程碑的存在, 解决了单词表示的语义相似性的问题, 但是没能很好地解决多义词问题, 这种单词表示学习模型, 本文称为与上下文无关的单词嵌入表示模型.

3.2　考虑上下文信息的单词嵌入表示模型

一种有效解决多义词问题的方法是考虑上下文信息的嵌入表示, 即表示随着上下文的变化而变化. 下面本文介绍几个典型的上下文有关的单词嵌入表示模型.

3.2.1　CoVe

McCann 等人[36]提出了上下文向量 (context vector, CoVe)模型, 使用深度LSTM编码器为机器翻译任务训练一个序列到序列的模型, 用于产生考虑上下文

信息的单词嵌入表示, 并应用到NLP任务中. CoVe模型的公式表示为

$$\mathrm{CoVe}(w) = \mathrm{MT} - \mathrm{LSTM}(\mathrm{Glove}(w)), \tag{26}$$

$$\tilde{w} = [\mathrm{Glove}(w); \mathrm{CoVe}(w)], \tag{27}$$

其中$\mathrm{Glove}(w)$是单词w的Glove向量.

如图10和式(26)–(27)所示, 该模型直接采用MT–LSTM模型的两层单向LSTM编码器对预先训练的Glove嵌入表示进行编码, 并将输出作为上下文向量与Glove向量拼接形成增广向量, 并作为下游自然语言处理任务的输入. CoVe改进了许多自然语言处理任务的性能, 包括情感分析、问题分类、问题回答等. 更重要的是, 它考虑了单词的上下文信息.

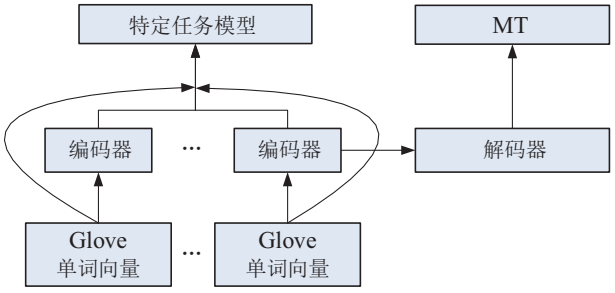


图 10 CoVe模型示意图

Fig. 10 CoVe

注: 右) 训练一个两层、双向的LSTM作为机器翻译序列到序列模型的编码器; 左) 使用它为其他自然语言处理模型提供上下文信息

3.2.2 ELMo

Peters等人[37]提出的语言模型的嵌入表示(embeddings from language model, ELMo) 学习方法, 是一个考虑上下文信息的单词嵌入表示深度模型, 其单词嵌入表示是从深度双向语言模型(bidirectional language models, biLM)[20]的隐层提取的, 需要在大规模未标记语料库中预先训练得到. 介绍如下.

给定长度为N的单词序列$(m_1, m_2, \cdots, m_N)$, 其中历史序列为$(m_1, \cdots, m_{k-1})$, biLM的前向语言模型计算目标单词m_k的条件概率; 给定未来上下文序列$m_{k+1}, m_{k+2}, \cdots, m_N$, biLM的后向语言模型计算目标单词$m_k$的条件概率. biLM联合最大化前向和后向语言模型的对数似然函数

$$\sum_{k=1}^{N} (\log p(m_k|m_1, \cdots, m_{k-1}; \Theta_x, \vec{\Theta}_{\mathrm{LSTM}}, \Theta_s) + \log p(m_k|m_{k+1}, \cdots, m_N; \Theta_x, \bar{\Theta}_{\mathrm{LSTM}}, \Theta_s)). \tag{28}$$

从图11所示的体系结构中可以清楚地看出, biLM首先用charCNN构造基于字符级的单词嵌入表示, biLM进一步应用两层双向LSTM来学习单词的隐表示, 该隐表示考虑了单词的上下文信息, 然后利用学习的隐表示, 使用软最大层计算前向和后向语言模型的条件概率.

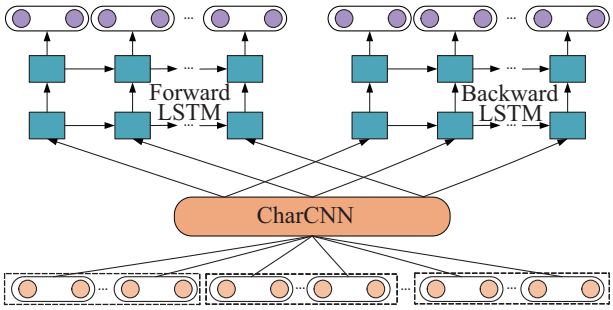


图 11 ELMo的网络结构

Fig. 11 ELMo

注: CharCNN代表字符级的CNN

ELMo是biLM层的线性组合. 假设biLM具有L层LSTM, 那么ELMo神经网络为目标单词m_k产生$2L+1$个表示, 即

$$R_k = \{\boldsymbol{x}_k^{\mathrm{LM}}, \overrightarrow{\boldsymbol{h}}_{kj}^{\mathrm{LM}}, \overleftarrow{\boldsymbol{h}}_{kj}^{\mathrm{LM}} | j = 1, \cdots, L\} = \{\boldsymbol{h}_{kj}^{\mathrm{LM}} | j = 0, \cdots, L\}, \tag{29}$$

其中: $h_{t,0}^{\mathrm{LM}} = x_k^{\mathrm{LM}}$是字符级CNN的输出, $\boldsymbol{h}_{kj}^{\mathrm{LM}} = [\overrightarrow{\boldsymbol{h}}_{kj}^{\mathrm{LM}}, \overleftarrow{\boldsymbol{h}}_{kj}^{\mathrm{LM}}]$是第$j$层前向和后向LSTM得到的隐表示叠加在一起形成的增广向量.

当将ELMo的输出应用于下游任务时, 最简单的情况是, 仅选择ELMo的顶层, 即$E(R_K) = \boldsymbol{h}_{kj}^{\mathrm{LM}}$作为下游任务的输入. 一般情形, 计算所有biLM层的特定任务权重

$$\mathrm{ELMo}_k^{\mathrm{task}} = E(R_k; \Theta^{\mathrm{task}}) = \gamma^{\mathrm{task}} \sum_{j=0}^{L} s_j^{\mathrm{task}} \boldsymbol{h}_{kj}^{\mathrm{LM}}, \tag{30}$$

其中s^{task}是软最大归一化的权重, 标量参数γ^{task}可以控制任务模型缩放整个ELMo向量.

ELMo利用的是几乎无限的无标记文本数据来学习单词的动态表示, 并显著提高了自然语言处理问题的预测性能.

3.2.3 BERT

Devlin等人[38]引入了带有变送器(transformer)的双向编码器表示(bidirectional encoder representations from transformers, BERT)语言模型. BERT框架分为两步: 预训练和微调. 在预训练期间, 通过不同的预训练任务对未标记的数据进行模型训练. 为了进行微调, 首先使用预训练的参数初始化BERT模型, 然后使用来自下游任务的标记数据对所有参数进行微调, 每个下游任务都有单独的微调模型(如图12所示).

BERT是使用多层双向变送器的编码器, 而变送器使用双向自注意力机制. BERT的输入表示如图13所示(每个序列的第1个标记总是一个特殊的分类标记[CLS], 用[SEP]分隔两个句子).

BERT的两步介绍如下:

1) 预训练的BERT.

与Chelba等人提出的方法不同[39], Devlin等人没有使用传统的从左到右或从右到左的语言模型来预训练BERT, 而是使用两个无监督任务对BERT进行预训练: 掩码语言模型学习(masked LM)和下一个句子预测任务(next sentence prediction, NSP). 如图12的左侧部分.

a) 掩码语言模型学习.

为了训练深度的双向嵌入表示, Devlin等人简单地随机遮蔽一定百分比的标记, 称为“有掩码的语言模型”(masked LM, MLM), 亦称为“完形填空任务”(cloze task). 然后, 使用特征嵌入表示的隐表示代入交叉熵损失, 求解使得交叉熵损失最小的原始特征标记的预测值.

b) 下一个句子预测任务.

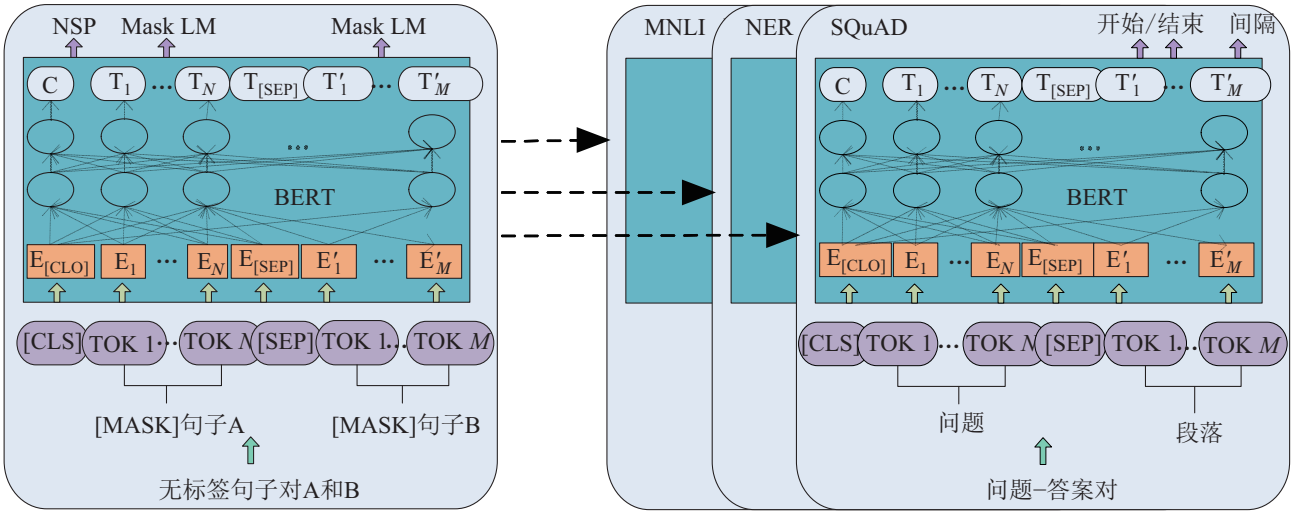
NSP任务的目标是预测句子之间的关系, 如预测句子b是否真的在句子a之后, 该任务的训练数据可以轻松地从任何单语语料库中得到. 将NSP与MLM联合优化便能在机器问答(question answering, QA)和自然语言推理(natural language inference, NLI)等下游任务中取得很好的预测性能. c) 预训练数据.

BERT的预训练过程和通用的预训练过程一样.

2) 微调BERT.

BERT使用自注意力机制来统一输入和输出这两个阶段, 对于每个任务, 只需将特定于任务的输入和输出代入BERT, 并端对端微调所有参数. 经过预训练的两个句子作为BERT微调网络的输入是, 在BERT预训练网络的输出处, 将学习得到特征表示, 输入到输出微调层中, 如, 在序列标记或机器问题回答任务中, 将[CLS]表示输入到微调输出层中, 执行特定的任务.

实验表明, BERT在11种自然语言处理任务上获得了很好的结果, Devlin等人主要贡献是将发现进一步推广到深层次的双向体系结构, 从而使相同的经过预训练的模型能够成功解决各种NLP任务.



(a) 预训练　　(b) 微调

图 12　BERT的总体预训练和微调过程

Fig. 12　Overall pre-training and fine-tuning procedures for BERT

注: 除了输出层, 预训练和微调中都使用相同的神经网络结构. 相同的预训练模型参数用于初始化不同的下游任务的模型并对所有参数都进行微调. 图中, [CLS]是在每个输入示例前添加的特殊符号, [SEP]是特殊的分隔符(例如, 分隔问题和答案)

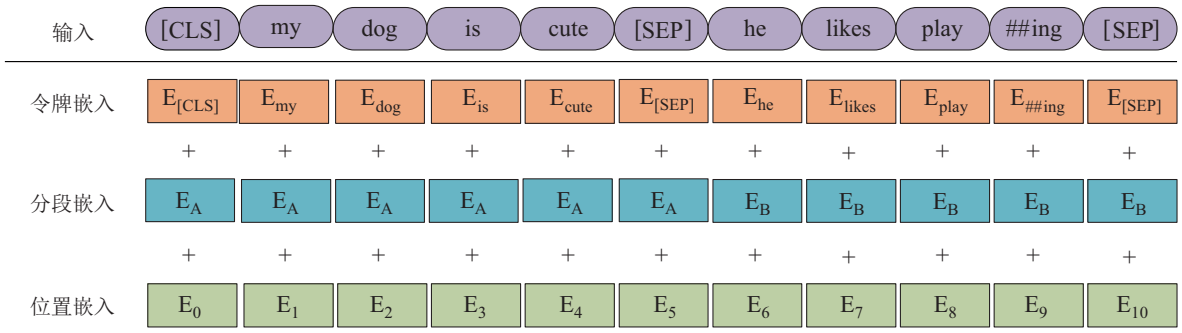


图 13　BERT输入表示

Fig. 13　BERT input representation

注: 输入的嵌入表示是特征嵌入表示, 分段嵌入表示和位置嵌入表示的总和

3.2.4 VIB

信息瓶颈(information bottleneck, IB)方法起源于信息论, 令X表示所要学习映射关系的“输入”变量, 例如句子, 而Y表示所要学习映射关系的“输出”变量, 例如语法分析. 假设联合分布$P(X,Y)$已知, IB目标是在X已知情况下, 学习某个压缩表示T的条件后验概率$p_\theta(t|x)$最大的参数θ和T, 类似于学习统计学中充分统计量. 定义目标函数(31), IB的目标就是学习使得式(31)最小化的压缩表示T

$$\mathcal{L}_{\rm IB} = -I(Y;T) + \beta \cdot I(X;T), \tag{31}$$

其中: $I(\cdot;\cdot)$是互信息. 优化目标函数(31), 即令T保留尽量少的与x有关的信息(第2项), 但这些信息足以预测Y. 两项之间的平衡关系由权衡参数β控制. 通过增加β就可以使$I(X;T)$尽可能小, “压缩瓶颈”可以理解成通过压缩预测精度$I(Y;T)$换取更大程度的压缩隐表示. IB的目标是通过对T携带的有关X的信息施加一些约束来最大化T的预测能力[40].

像前面提到的ELMo和BERT模型, IB可以作为预训练的手段, 以便得到单词的嵌入表示, 该表示包含了丰富的语法和语义信息. Li等人[41]提出了一种快速的变分信息瓶颈方法 (variational information bottleneck, VIB), 对这些嵌入表示的信息进行非线性压缩, 只保留有用的信息. VIB不是单纯地降维, 它有效地避免了对压缩表示中的可用维数的不必要使用, 利用随机性去除了不需要的部分信息. 下面对这个模型进行介绍.

Li等人扩展了原来的IB目标(31), 添加了

$$\gamma \sum_{i=1}^{n} I(T_i; X|\hat{X}_i)$$

项, 起到控制所提取的标签的上下文敏感性的作用. 这里T_i表示与第i个单词相关联的标签, X_i是第i个单词的ELMo特征嵌入表示, 而$\hat{X}_i$ 是同一单词的ELMo类型嵌入表示. 扩展后的目标函数表示为

$$\begin{aligned}\mathcal{L}_{IB} = &- I(Y;T) + \beta \cdot I(X;T) + \\ &\gamma \sum_{i=1}^{n} I(T_i; X|\hat{X}_i).\end{aligned} \tag{32}$$

如图14所示, 在依赖解析任务上应用VIB估计方法, 将句子的单词嵌入表示X_i压缩为连续的向量值标签或离散标签T_i (“编码”), 以使标签序列T保留预测依赖解析Y (“解码”)的最大信息. VIB使用相同的随机映射函数和信息损失变换结构, 独立地压缩每个X_i.

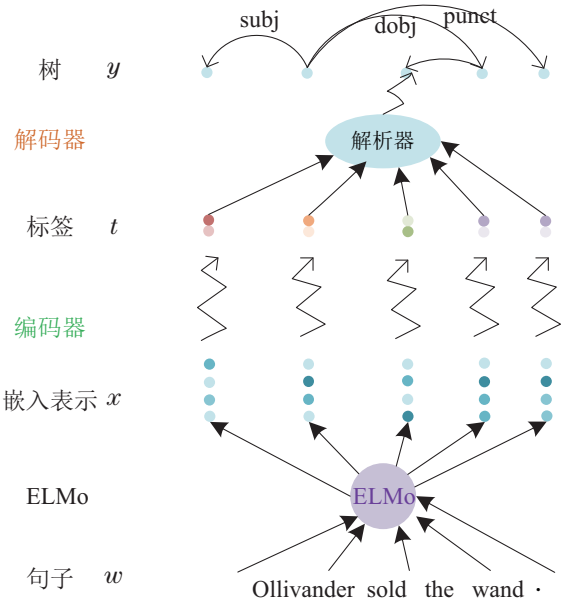


图 14 信息瓶颈示例

Fig. 14 VIB

注: $T = \{t_1, t_2, \cdots\}$表示瓶颈变量, 锯齿状箭头表示随机映射, 锯齿状箭头从分布的参数指向从该分布抽取的样本

IB方法通过定义条件分布$p_\theta(t|x)$引入了新的随机变量T, 即X的压缩标签序列. 选择p_θ的参数θ使IB目标(32)最小.

Li等人巧妙地利用变换和重参数化技巧, 提出了两种使用变分信息瓶颈的方法来压缩ELMo单词特征嵌入表示. 可以将相同的训练方法压缩ELMo或BERT特征序列, 执行其它学习任务, 所需要的只是一个特定于模型的解码器. 例如, 在情感分析的情况下, 该方法应仅保留情感信息, 不需要保留无用的语法知识. 序列为离散值时, 自动压缩标签, 形成一个侯选标签集, 实验证明, 这些标签可以捕获传统POS标签注释中的大多数信息, 可以在相同标签粒度级别上更准确地解析出标签序列; 序列为连续值时, 实验证明, VIB方法适度地压缩单词嵌入表示, 在9种语言中的8种中产生更准确的依赖解析, 如表4所示.

表 4 解析9种语言的准确性

Table 4 Parse the accuracy of nine languages

模型	阿拉伯语	北印度语	英语	法语	西班牙语	葡萄牙语	俄语	中文	意大利语
Iden	0.751	**0.870**	0.824	0.784	0.808	0.813	0.783	0.709	0.863
PCA	0.743	**0.866**	0.823	0.749	0.802	0.80	0.777	0.697	0.857
MLP	0.759	**0.871**	0.839	0.816	0.835	0.821	0.800	0.734	**0.867**
VIB	**0.779**	**0.866**	**0.851**	**0.828**	**0.837**	**0.836**	**0.814**	**0.754**	**0.867**

3.2.5 小结与纵向分析

上述的模型总结如表5–6所示. 上文提到的下游任务包括: 语言建模(language modeling, LM)[7, 13]、分块(chunking)[7–8]、词性标注(part-of-speech tagging)[8]、命名实体识别(named entity recognition, NER)[8, 27]、情感分析(sentiment analysis, SA)[42–43]、语义角色标注(semantic role labeling, SRL)[8]、依赖句法分析(ependency parsing)[3]、机器翻译(machine translation, MT)[44]、自然语言推理(nature language inference, NLI)[45–46]和机器阅读理解(machine reading comprehension, MRC)[47–48]. 本章介绍的模型都是近几年经典的模型, 它们巧妙地应用了预训练等技术.

表 5 解析9种语言的准确性

Table 5 Parse the accuracy of nine languages

模型	SST-2	SST-5	SNLI
CoVe	90.4	53.7	88.1
ELMo	78.6	54.7	88.7
BERT	94.9	—	—

表 6 加入预训练的单词嵌入表示

Table 6 Add pre-trained word embedding

模型	优缺点	评价方法	数据集	应用领域
CoVe	改进了很多NLP任务性能, 揭示了词语的上下文关系	ACC, F1等	SST-2等	下游任务
ELMo	经过预训练后, biLM可以计算任何任务的表示形式	ACC, F1等	SST-2等	下游任务
BERT	BERT在11种自然语言处理任务上获得了很好的结果	ACC, F1等	SST-2等	下游任务
VIB	只保留有助于区分解析器的信息	ACC等	UD等	POS等

CoVe推进了NLP模型的发展, 但是CoVe的一个不足是它依赖于有限标记的语言数据. 此后的ELMo模型针对这一问题有所改进, 利用的是几乎无限大的未标记数据, 这不仅减轻了未登录词(OOV)问题, 而且有效地减少了模型参数的个数. 但是上述模型都仅限于单向训练, 只利用训练目标左侧或者右侧的语境信息. BERT具有更强的双向捕获信息的能力, 在很多个NLP任务上都取得了巨大的进步, 也吸引了后续很多科研工作者, 试图以不同的方式改进它. 研究最多的问题是如何引入不同的预训练目标. BERT-wwm[49]和ERNIE (百度)[50]为了提高模型的泛化能力, 预测被遮盖的整个词/实体, 而不是词块. SpanBert[51]通过将多个单词“遮盖”为一个“区块”并预测一个“区块”内的整个内容, 改进“区块”预测结果. ELECTRA[52]没有预测被“遮盖”的嵌入表示, 而是用替换的嵌入表示检测任务预处理模型. Albert[53]用一个新颖的句子顺序预测任务取代了预测下一个句子的任务. 这些更难的预训练任务更好地发掘了变送器(transformer)模型的潜力.

4 跨语言单词嵌入表示模型

世界上大约有几千种不同的语言, 但只有少数语言有人为注释, 而这些语言之间的对应关系在单词表示学习任务中并没有很好的得到应用, 上文中提到的单词表示模型都是单语言模型, 这就需要单词嵌入表示的跨语言迁移学习, 在资源丰富的语言上训练的模型被应用到低资源语言中, 帮助低资源语言单词的表示学习, 单个语言的输入嵌入表示被投影到多个语言共享的语义空间中. 这种嵌入表示学习方法被称为跨语言单词嵌入表示[54]. 根据使用的单语嵌入表示类型, 跨语言嵌入表示学习方法可以分为静态表示学习和动态表示学习两种; 根据训练目标的不同, 可以分为在线表示学习和离线表示学习两种方式[55–56]. 而为什么又需要在平行语料上学习呢[57]? 因为一种语言中的多义词在另一种语言中可能有多个不同翻译, 这样有利于解决前文中提到的一词多义的问题.

在本节中, 笔者从众多跨语言模型中选择了以下两个有代表性的跨语言模型进行介绍.

4.1 BilBOWA

Guo等人[57]提出的无需词对齐的双语单词袋子(bilingual bag-of-words without word alignments, BilBOWA)模型, BilBOWA单词嵌入表示是一种无需考虑单词对齐关系的快速双语分布式嵌入表示学习方法, 模型简单且计算高效, 适用于大型单语数据集, 直接对单语数据进行训练, 并从一组较小的原始文本句子对齐数据中提取双语信息.

该模型的示意图如图15所示.

BilBOWA模型的损失函数分为两个部分, 目标是使总的联合损失函数最小:

$$\mathcal{L} = \min_{\theta^e,\theta^f} \sum_{l\in\{e,f\}_{w_t},h\in\mathcal{D}^l} \underbrace{\mathcal{L}^l(w_t,h;\theta^l)}_{\text{特征学习}} + \lambda \underbrace{\Omega(\theta^e,\theta^f)}_{\text{对齐}}, \quad (33)$$

其中: h是上下文, w_t是目标单词, $\mathcal{D}$是数据集, e和f表示两种语言, 此处以英语和法语为例.

1) 学习单语言的特征: 单语言隐表示的目标函数$\mathcal{L}^l$.

BilBOWA采用负采样Skip–Gram模型, 用于学习

得到高质量的单语言隐表示, 同时使损失最小.

2) 学习跨语言的特征: 无需单词对齐的双语单词袋子模型的损失函数Ω.

一般的, 在双语环境中, 单词相似性可以表示为矩阵A, 其中a_ij编码一种语言中单词i的翻译到另一种语言中单词j的得分. 将K维单词嵌入表示作为行向量$\boldsymbol{r}_i$堆叠起来以形成(W, K)维矩阵R, 则跨语言对齐部分的目标函数可以表示为

$$\Omega_A(R^e, R^f) = \sum_i \sum_j \boldsymbol{a}_{ij} \|\boldsymbol{r}_i^e - \boldsymbol{r}_j^f\|^2 = (R^e - R^f)^{\mathrm{T}} A (R^e - R^f). \quad (34)$$

矩阵A捕获了英语字典W^e中所有的单词与法语字典W^f中所有的单词之间的关系, 也是该模型学习的难点. 但是词对齐的训练计算成本高、噪音大, 所以利用平行训练数据获得局部单词共现统计近似式(34)中利用全局单词对齐统计得到的$\Omega(\cdot)$项, 得到式(35), 表示在第t步时, 优化该近似的跨语言目标, 过程如图16所示.

$$\Omega_A^{(t)}(R^e, R^f) \triangleq \|\frac{1}{m} \sum_{w_i \in s^e}^{m} \boldsymbol{r}_i^e - \frac{1}{n} \sum_{w_j \in s^f}^{n} \boldsymbol{r}_j^f\|^2. \quad (35)$$

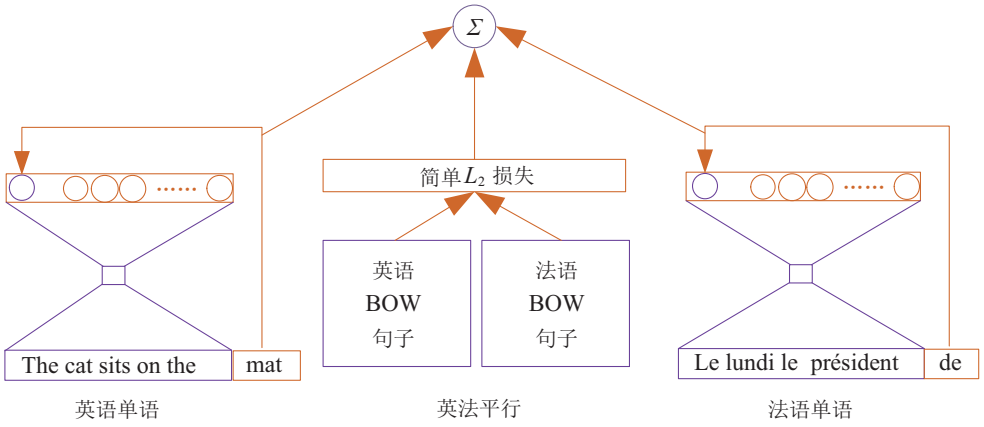


图 15 BilBOWA模型体系结构示意图

Fig. 15 BilBOWA

注: 联合训练两个单语Skip–Gram模型, 同时约束使样本的嵌入表示的L_2对齐损失最小, 以便为翻译对分配两种语言的相似嵌入

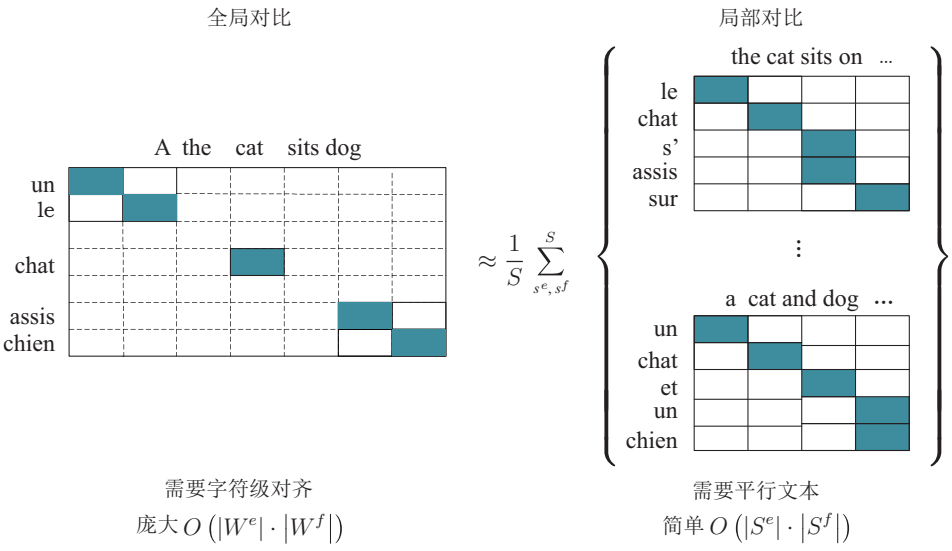


图 16 全局单词对齐统计与局部单词共现统计关系图

Fig. 16 Diagram of global word alignment statistics and local word co-occurrence statistics

使用全局单词对齐来对齐跨语言嵌入表示式(34)过程麻烦且复杂, 并且模型参数会随着两个语言词汇量的乘积而快速增长. 相比之下, BilBOWA损失式(35)是通过对平行句子对的有限样本中的隐式的局部共现统计值求平均来近似全局损失函数.

综上所述, BilBOWA是一种计算效率高的模型, 可直接从单语言原始文本和有限数量的平行数据中导出双语分布式单词表示形式, 不需单词对齐, 将训练单语单词嵌入表示的先进技术与采样式跨语言目标相结合. 使得每个训练步骤所需的计算仅与句子中

单词的个数成比例, 从而可以进行高效的大规模跨语言训练. Gouws等人[58]的实验也证明考虑跨语言对齐关系的方法优于针对跨语言文档分类任务以及针对WMT11数据的词汇翻译任务的技术.

4.2 Trans–Gram

Coulmance等人[59]提出了Trans–Gram模型, 简单且高效, 可仅使用单语言数据和少量句子对齐数据, 同时学习和对齐多种语言的单词嵌入表示, 用这个方法计算以英语为中心语言的21种语言的对齐单词嵌入表示.

Trans–Gram是在Skip–Gram的基础上提出的, 利用句子对齐关系而不是单词对齐关系构造隐表示, 是一种学习跨多种语言对齐关系的单词嵌入表示学习方法, 可以将单语言损失和跨语言损失的总和最小化. 假设学习语言e(例如英语)和f(例如法语)的对齐的单词向量, 用Skip–Gram表示单语言损失, 记为J_e和J_f. 在平行对齐语料库$A_{e,f}$中, 每个英语句子s_e与法语句子s_f是对齐的. 在Skip–Gram中, 在句子s_e中为目标单词w_e选择的上下文是出现在以w_e为中心的固定大小的窗口中的单词的集合, 记为$s_e[w_e-l, w_e+l]$. 在Trans–Gram中, 在句子s_e中为目标单词w_e选择的上下文是句子s_f中出现的全部单词c_f. 损失函数记为

$$\Omega_{e,f}=\sum_{(s_e,s_f)\in A_{e,f}}\sum_{w_e\in s_e}\sum_{c_f\in s_f}-\log\sigma(\overrightarrow{w_e}\cdot\overleftarrow{c_f}). \tag{36}$$

显然, 这种损失相对于这两个语言是不对称的. 因此, 需要有两个跨语言的损失函数: $\Omega_{e,f}$表示语言e的目标向量和语言f的上下文向量之间对齐的损失, $\Omega_{f,e}$表示语言f的目标向量和语言e的上下文向量之间对齐的损失. 图17表示了4种损失目标函数.

假设单词的含义在整个句子中均匀分布, 这个方法仅使用句子对齐的语料库, 而不使用单词对齐的语料库. 当要添加第3种语言i (例如意大利语)时, 只需为全局损失添加3个新目标函数(J_i, $\Omega_{e,i}$和$\Omega_{i,e}$), 简单高效. Upadhyay等人[56]提出的Trans–Gram在标准的跨语言文本分类和单词翻译任务上也取得了较好的结果, 为跨语言任务提供了新的方法, 推进了跨语言任务的发展.

4.3 小结与纵向分析

在本章中, 将从众多跨语言模型中选择了上述两个跨语言模型进行介绍, 在表7中, 给出了这两个模型在英语/德语分类任务上的准确度, 在表8中, 给出了这两个模型在翻译任务中的一个表现, 两个模型的优缺点总结如表9所示.

对于跨语言嵌入表示, 在线方法通常是学习源语言和目标语言的语言模型, 并通过跨语言目标共同优化其目标. 离线方法主要是学习投影矩阵(主要是线性变换矩阵), 将源语言的向量空间转换为目标语言的向量空间.

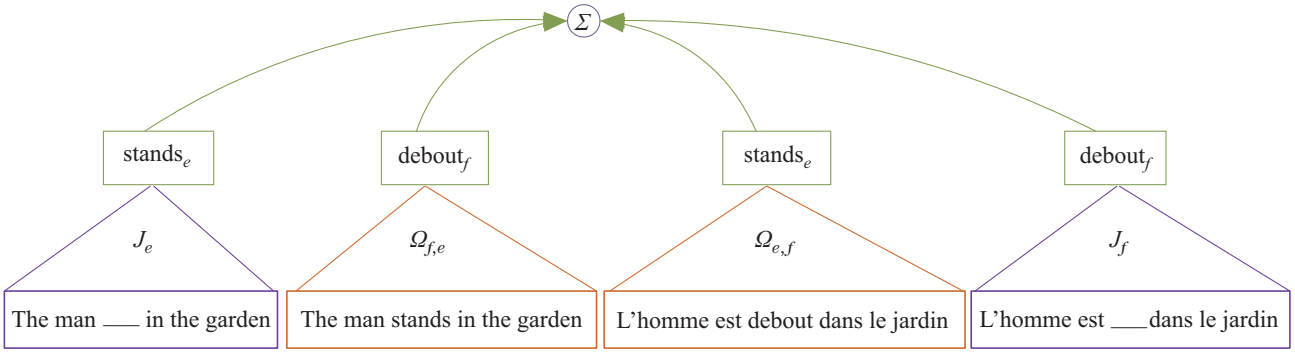


图 17　有助于英语和法语对齐的4个部分目标

Fig. 17　The four partial objectives contributing to the alignment of English and French

注: 有助于英语和法语对齐的4个部分目标: 在目标词(紫色)周围的窗口上, 每种语言的Skip–Gram目标(和)和两个Trans–Gram目标(和)在与从中提取目标词的句子对齐的整个句子(橙色)

表 7　模型在路透社英语/德语分类任务上的ACC

Table 7　ACC of model on reuters English/German classification

模型	En → Ge	Ge → En
BilBOWA	90.4%	53.7%
Trans–Gram	78.6%	54.7%

表 8　模型在翻译(英语–西班牙语)任务上的ACC

Table 8　ACC of model on translation (English - Spanish) task

模型	En → Sp	Sp → En
BilBOWA	44%	55%
Trans–Gram	61%	62%

表 9 跨语言单词嵌入表示模型总结

Table 9 Summary of the cross-language word embedding model

模型	优缺点	评价方法	应用领域
BilBOWA	计算效率高, 无需单词对齐或词典	ACC等	跨语言文本分类任务和单词翻译
Trans–Gram	只使用单语数据和较小的句子对齐数据集来同时学习和对齐各种语言的单词嵌入表示	ACC等	跨语言文本分类任务和单词翻译

在在线学习方面, Mulcaire等人[60]基于ELMo[37]模型, 在多语言数据中捕获字符级的语义信息, 提出了一种多语言考虑上下文信息的单词表示模型. Lample和Conneau[61]在BERT[38]基础上, 改变预训练的目标, 利用并行数据上的跨语言监督信息, 学习跨语言语言模型, 这些模型在几个跨语言任务上取得了不错的成果. 他们进一步证明, 大规模的预训练的多语言模型能够显著提高大量跨语言迁移任务的性能[62]. 在离线学习方面, Schuster等人[63]通过离线方法用线性投影对齐预训练过程, 并考虑上下文信息, 学习单词嵌入表示, 使用考虑上下文信息嵌入表示的均值, 作为每个单词的锚点, 并学习锚点集合中的转换矩阵. Wang等人[64]提出在不同语境中直接学习变换关系, 从而得到保留语义的跨语言动态嵌入表示. Mulcaire等人[65]评估了考虑上下文信息的跨语言嵌入表示方法, 这些模型极大地促进了跨语言依赖分析的学习效果, 与离线学习方法相比, 在线学习方法能更好地编码词汇对应的跨语言嵌入表示.

5 引入其它模态或关联信息帮助学习单词嵌入表示

根据模态之间的对应关系, 或引入句子中单词所起的语法作用, 或引入句子中单词的上下文信息, 或在视觉和文字模态之间, 引入语义信息, 都能够帮助更好地学习单词的向量表示.

5.1 联合单词–图像的嵌入表示模型

该小节介绍联合单词–图像双模态的嵌入表示学习方法. 现在图像标注数据集变得越来越大, 有数千万张图像和数万个标注. Weston等人[66]提出了一种高性能的方法, 通过同时学习优化给定的图像和标注文字, 并学习图像和文字标注的低维联合嵌入表示空间, 即联合单词–图像的嵌入表示.

表示图像和文字标注的映射函数是不同的, 但这两个映射函数需要被同时学习, 所以可以利用有监督的学习, 令损失函数最小化. 将图像表示$x \in \mathbb{R}^d$和文字标注$i \in \{1, \cdots, Y\}$, 对应到图像和文字标注共有的词汇表中.

学习图像特征空间到联合空间$\mathbb{R}^D$的映射: $\boldsymbol{\Phi}_{\rm I}(x): \mathbb{R}^d \to \mathbb{R}^D$, 同时学习文字标注映射: $\boldsymbol{\Phi}_M(i): \{1, \cdots, Y\} \to \mathbb{R}^D$, 采用线性映射, 即$\boldsymbol{\Phi}_{\rm I}(\boldsymbol{x}) = V\boldsymbol{x}$和$\boldsymbol{\Phi}_M(i) = M_i$, M_i是大小为$D \times Y$矩阵的第i个索引列, 可以使用任何非线性映射函数. 对图像向量$\boldsymbol{x}$使用视觉袋子模型的稀疏高维特征向量表示, 每个标注都有自己的学习表示.

对于给定的图像, 对可能的文字标注进行排序, 选择排名最高的标注来描述图像的语义内容, 表示为

$$f_i(\boldsymbol{x}) = \boldsymbol{\Phi}_M(i)^{\rm T}\boldsymbol{\Phi}_{\rm I}(\boldsymbol{x}) = M_i^{\rm T}V\boldsymbol{x}, \tag{37}$$

其中根据$f_i(\boldsymbol{x})$的大小对可能的文字标注i进行排序, 从大到小, 则模型族的范数约束为

$$\|V_i\|_2 \leqslant C,\ i = 1, \cdots, d, \tag{38}$$

$$\|M_i\|_2 \leqslant C,\ i = 1, \cdots, Y. \tag{39}$$

将范数约束引入到模型的损失函数中进行训练, 范数约束可以起到正则化项的作用.

Weston等人为了缩短训练时间, 使用加权近似秩两两损失(weighted approximate-rank pairwise loss, WARP loss), 类似于有序加权两两分类(ordered weighted pairwise classification, OWPC)[67], WARP使用随机梯度下降和一种新的抽样技巧来近似秩, 从而得到一种有效的在线优化策略, 作者通过实验证明该策略优于应用于在相同损失标准下的随机梯度下降算法, 证明该方法不仅性能优于几种基准方法, 而且计算速度更快, 内存消耗更少.

5.2 引入文本语义信息学习图像对应的语义嵌入模型

Frome等人[68]提出了一种新的深度视觉语义嵌入模型(a deep visual-semantic embedding model, DeViSE), 使用有文本标记的图像数据以及从未标注的文本中收集的语义信息训练该模型, 从而识别视觉对象. 模型实现过程如下.

这个模型的目标是利用在文本域中学习语义知识, 将其迁移到视觉对象识别的模型的训练中. 首先, 预训练一个简单的神经语言模型[14]; 同时, 预训练一个用于视觉对象识别的深度神经网络[69], 该网络与传统的软最大输出层组合在一起完成训练; 然后, 对预训练的视觉对象识别网络的较低层重新训练, 预测由语言模型学习的图像标签文本的向量表示, 从而构造深度视觉语义模型. 由两个预先训练的神经网络模型初始化深度视觉语义嵌入模型(DeViSE)如图18所示.

Frome等人的实验表明: 该模型能较好地实现有1,000个图像对象类的ImageNet目标识别任务, 同时

在零样本学习(zero-shot learning, ZSL)上也有不错的表现, 利用语义知识改进了对这种未出现的图像的预测, 在成千上万的视觉模型中对从未见过的新标签, 实现18%的预测准确率.

5.3　引入单词结构内容学习单词隐表示

Kiros等人[70]提出了在多神经语言模型[71]基础上引入结构内容的神经语言模型(structure-content neural language models, SC–NLM), 假定存在单词序列$S=\{w_1,\cdots,w_n\}$, 结构变量$T^{\rm str}=\{t_1,\cdots,t_N\}$, 这里, t_i对应单词w_i的词性, 给定目标单词w_i的上下文$w_{1:n-1}$对应的隐表示$\boldsymbol{u}$, 目标是通过单词上下文$w_{1:n-1}$和结构上下文$t_{n:n+k}$ (k是$\boldsymbol{w}_i$前面上下文单词的个数)来计算w_i的分布式表示的条件后验概率$P(w_i|w_{1:n-1},t_{n:n+k},\boldsymbol{u})$. 图19对多神经语言模型和SC–NLM模型的预测问题进行了展示. 可以发现引入单词在句子中起的结构作用变量有助于在生成短语模型时避免模型生成一些不必要的内容.

SC–NLM可以理解为一个多神经语言模型, 但是属性向量表示部分换成上下文向量$\boldsymbol{u}$和结构变量$T^{\rm str}$的加性函数. 假定$\{\boldsymbol{t}_n^{\rm str},\cdots,\boldsymbol{t}_{n+k}^{\rm str}\}$, $\boldsymbol{t}_i^{\rm str}\in\mathbb{R}^K$, $i=n,\cdots,n+k$是结构变量$T^{\rm str}$的嵌入向量表示, 这些向量是学习获得的. 引入$G\times G$的结构上下文矩阵. 令T^u为$G\times K$的加性模型向量$\boldsymbol{u}$的上下文矩阵. 由结构变量$T^{\rm str}$和上下文信息组成的属性向量$\hat{\boldsymbol{u}}$定义为

$$\hat{\boldsymbol{u}}=[(\sum_{i=n}^{n+k}T^{(i)}t_i)+T^{(u)}\boldsymbol{u}+\boldsymbol{b}]_+, \tag{40}$$

这里$[\cdot]_+=\max\{\cdot,0\}$是ReLU非线性激活函数, $\boldsymbol{b}$是偏置向量. 向量$\hat{\boldsymbol{u}}$替换多神经语言模型的向量$\boldsymbol{u}$, 并且模型的其余部分保持不变. Kiros等人[70]的实验表明, 用LSTM编码器与SC–NLM解码器将Flickr30K数据集和Microsoft COCO数据集联合训练, 获得了很好的结果, 生成的语言描述很准确, 也在DeViSE模型上提升了很多, 实验结果对比见表10.

5.4　引入句法依赖解析信息学习特征嵌入表示

Chen等人[72]提出了将特征嵌入表示用于依赖解析的方法 (feature embedding for dependency parsing, FEDP), 能自动学习特征嵌入表示, 解决句法依赖解析中特征稀疏的问题, 模型介绍如下.

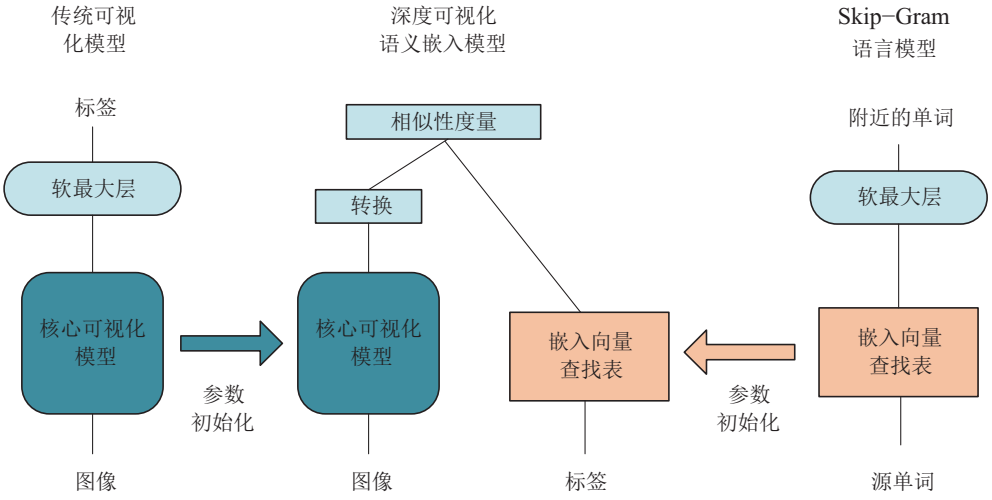
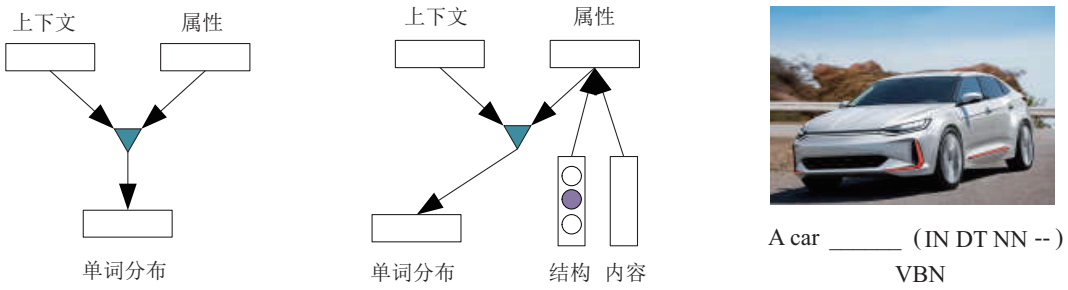


图 18　DeViSE联合模型

Fig. 18　DeViSE model

注: 左: 具有软最大输出层的视觉对象分类网络; 右: 一个Skip–Gram语言模型; 中心: DeViSE联合模型, 使用在其它两个模型的较低层预先训练的参数进行初始化



(a) 多神经语言模型　　(b) 结构−内容神经语言模型(SC−NLM)　　(c) SC−NLM预测问题

图 19　3种神经语言模型

Fig. 19　Three neural network language models

表 10 在Flickr30K数据集上的实验

Table 10 Experiments on Flickr30K data-sets

模型	图像注释				图像查找			
	R@1	R@5	R@10	Med r	R@1	R@5	R@10	Med r
DeViSE	4.5	18.1	29.2	26	6.7	21.9	32.7	25
SC–NLM	23.0	50.7	62.9	5	16.8	42.0	56.5	8

注: R@K表示Recall@K (越高越好), Med r表示median rank, 中位秩(越低越好)

如图20所示, 假定n个句子记为$x_1, \cdots, x_n$, $y_1, \cdots, y_n \in Y$是与n个句子对应的语法(或者句法)解析树, 假定第i个句子$x_i = w_{i,1}, \cdots, w_{i,P}$包含$P$个单词, 句子$x_i$的任意某个单词$w_{i,p}$的隐表示为$f_{i,p}$, 以下讨论中为了简化符号, 去掉下标索引, 简记为f, 其对应的语法(或者句法)解析树中有依赖关系的节点的隐表示记为cf, 在嵌入表示模型中使用当前基于依赖的特征表示来预测周围的特征表示, 如图21所示. 给定句子及其对应的依赖树Y, 学习的目标是使上下文特征的对数条件似然概率最大化:

$$\sum_{y_i \in Y} \sum_{f \in F_y} \sum_{cf \in CF_f} \log(P(cf|f)), \tag{41}$$

其中: F_y是从树y生成的一组特征表示, CF_f是特征表示f的M步(M-step)上下文中的周围特征隐表示的集合, $cf \in CF_f$. 把条件后验概率$P(cf|f)$参数化为软最大函数[15]

$$P(cf|f) = \frac{\exp(\boldsymbol{u}_{cf}^{\mathrm{T}} \boldsymbol{v}_f)}{\sum_{i=1}^{F} \exp(\boldsymbol{u}_{cfi}^{\mathrm{T}} \boldsymbol{v}_f)}, \tag{42}$$

其中: $\boldsymbol{v}_f$和$\boldsymbol{u}_f$分别是f的输入和输出单词向量表示, F是特征表中特征隐表示的个数. 在处理大规模数据时, 为了更好地计算概率, 引入负采样方法, 公式为

$$\begin{aligned} &\log(P(cf|f)) = \\ &\log \sigma(\boldsymbol{u}_{cfi}^{\mathrm{T}} \boldsymbol{v}_f) + \\ &\sum_{k=1}^{K} E_{cfk \sim P(cf)} [\log \sigma(-\boldsymbol{u}_{cfk}^{\mathrm{T}} \boldsymbol{v}_f)]. \end{aligned} \tag{43}$$

$\sigma(z) = 1/(1+\exp(-z))$, $P(f)$是数据上的噪声分布, K取经验值5[15]. 逐一预测这些特征表示. 在预测第i个特征表示后, 用随机梯度上升方法进行迭代更新的公式为

$$\theta \leftarrow \theta + \alpha \left(\frac{\partial \sum_{cf} \log(P(cf_i|f))}{\partial \theta} \right), \tag{44}$$

其中: α是学习率, θ包括模型的参数和特征的向量表示. α的初始值是0.025. 如果在一次更新后对数似然函数值没有显着改善, 则将学习率α减半[73].

将句法依赖解析信息引入特征表示学习中, 学习得到特征表示, 可用于大规模依赖解析学习任务中. 基于学习到的特征嵌入表示, 作为一组新的特征, 将它们与基于图模型的全局特征表示方法一起使用, 在英语和中文测试(如表11)时, 与基准方法相比, 该方法显著地提高了性能表现.

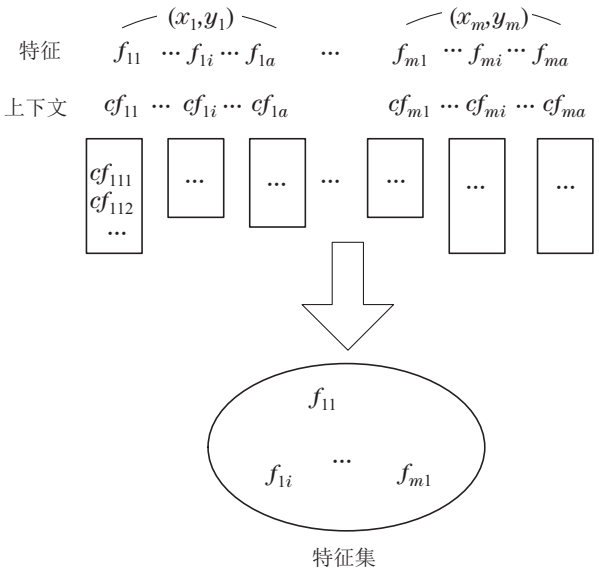


图 20 输入特征集

Fig. 20 Input feature set

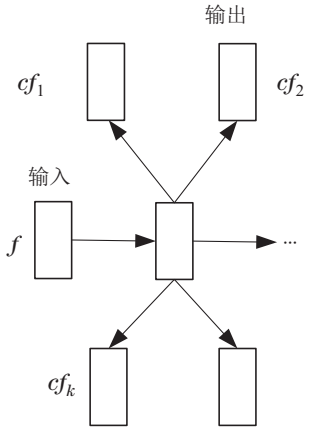


图 21 特征嵌入表示模型

Fig. 21 Feature embedding model

5.5 小结与纵向分析

上述模型总结如表12所示, 将图像与单词嵌入表示相结合, 对于给定的图像, 对可能的文字标注进行排序, 可以更好地描述图像的语义内容. 比如, DeViSE模型可以用于标签数目大得多的系统中, 包括

有类重叠或从未观察到的类别. 更直接地利用所学的语言嵌入表示中已有的结构信息, 大大降低联合模型的训练成本, 并进一步扩展单词嵌入表示的应用场景. 还有基于变送器的预训练语言模型, 特别是BERT, 以增强多模态表示的学习能力. 例如, Lu等人[74]和Su等人[75] 提出通过改进概念说明数据集 (conceptual captions dataset)[76]上BERT的预训练模型, 通过3个任务学习图像和自然语言的联合表示, 即“遮盖”语言建模、“遮盖”视觉特征分类和句子–图像对齐建模. 而Sun等人[77] 将BERT 应用于视频领域, 提出联合学习视频和语言表示, 在各种有趣的文本视频相关任务中取得了很好的结果. 鉴于这些工作已经取得的进展, 期待共同学习多种模态表示, 得到更多的研究.

表 11 FEDP在中文数据集上的表现

Table 11 FEDP performance on Chinese data-sets

模型	POS	UAS	COMP
Baseline	93.61	81.04	29.73
FEDP	93.61	82.94	31.72

注: 1. 数据集为: Chinese Treebank version 5.1 (CTB5)
2. POS: part-of-speech, UAS: unlabeled attachment, COMP: complete dependency tree matches

表 12 混合单词嵌入表示模型总结

Table 12 Summary of mixed word embedding model

模型	优缺点	应用领域
联合单词–图像的嵌入表示	方法不仅性能优于几种基准方法, 而且比它们更快, 消耗的内存更少	图像
DeViSE	可以准确地用于标签数目大得多的系统中, 包括重叠或从未观察到的类别	图像
SC–NLM	根据编码器产生的分布式表示, 将句子结构与其内容分离开来	图像
FEDP	从大量原始数据中学习特征嵌入以进行依赖解析	依赖解析

6 未来趋势与发展方向

综上所述, 单词嵌入表示是自然语言处理任务的基础工作. 嵌入表示是基于语义的单词向量表示, 而这个向量表示的每一点改进都可能为后续工作提供很大的帮助. 同时, 单词嵌入表示与其它模型的融合可以使自然语言处理任务更加多元化, 拓展自然语言处理应用范围, 比如, 可以有如下发展方向或应用方向.

1) 机器自动问答.

在当今的数字网络世界里, 人们热衷于通过上网来搜索自己需要的知识, 找到问题的答案, 所以有越来越多的机器自动问答系统出现. 尤其是随着预训练模型在一般自然语言处理任务中的成功应用, 预训练模型也被引入到了生物医学领域, 许多基于预训练模型的方法在生物医学机器自动问答任务中被证明是成功的. Peng等人[78]受迁移学习的影响, 利用合适的单词嵌入表示, 将命名实体识别应用到生物医学问答中, 性能得到很大的提高. Hajiaminshirazi等人[79]针对社区问答(community question answering, CQA)存在信息不全面的问题, 提出将跨语言嵌入表示用于资源较少的社区信息, 方便进行跨语言问题检索, 有效缩短翻译时间. 所以针对不同应用场景的机器自动问答任务提升单词嵌入表示的性能仍是一个值得研究的方向, 期待有更多领域的机器问答成果的出现.

2) 情感分析.

社交网络的发展离不开情感分析, 近年人们也是对情感分析有很大的兴趣. Alharbi等人[80]利用单词嵌入表示和不同RNN变体对在线评论情感分析进行评价. Dovdon等人[81]提出Text2Plot方法, 通过创建文本的二维情节表示来进行情感分析. Dong等人[82]提出了一种基于注意力的多任务学习网络, 该网络利用共享单词嵌入表示向下游任务传递信息, 允许提取和分类任务同时处理, 不同于以前特定情境中的情感分析, 避免了顺序模式导致的变换问题. 在社交网络上, 情感分析一直扮演着十分重要的角色, 所以想要发展社交网络固然离不开情感分析的发展, 故单词嵌入表示在情感分析领域的发展还是十分可观的.

3) 语义相似度.

微博、推特等在如今社会扮演着很重要的角色, 需要在事件发生的同时实时更新重要时刻记录、个人评论等, 而且大多数用户都更有兴趣阅读关于该事件最新进展的博文. 然而, 提取某事件的相关博文是一项具有挑战性的任务, 大量噪音博文和社交媒体内容的词汇变异问题都是干扰推送性能的因素. 为了应对这些挑战, Singh等人[83]提出了一种基于网页排名算法的计算博文语义相似度的方法. 该方法利用Word2Vec模型, 在博文图表示的邻接矩阵中包含基于词移动距离度量的语义相似度矩阵. 其实验结果也表明, 与基准方法相比, 该方法生成的排名靠前的博文更简洁、更有新闻价值. 单词嵌入表示在语义相似度上的应用

还有Zhao等人提出的一种新的基于单词网络和词嵌入的语义相似度度量模型[84].

4) 词义消歧、多义词.

单词嵌入表示能够很好地表示词的语义信息, 近年来也被广泛应用于词义消歧(word sense disambiguation, WSD)等任务中. 但是, 很多单词嵌入方法没有充分考虑词的同义性和多义性, 致使它们的性能表现有限. Jia等人[85]提出了一种有效的基于主题词嵌入(topical word embedding, TWE)的WSD方法: TWE–WSD, 该方法综合了隐狄利克雷分配(latent dirichlet allocation, LDA)模型和单词嵌入表示. TWE–WSD不是单纯地为每个单词生成一个单词向量表示, 而是为每个主题下的每个单词生成一个主题词向量表示, 同时设计了有效的集成策略来获得高质量的上下文向量, 并获得了不错的性能表现. Li等人[86]也针对多义词进行了研究, 提出了一种自适应跨语境词嵌入算法, 是针对多义词问题的无监督主题建模. 该方法在很多基准数据集上都取得了好的性能. 所以, 将单词嵌入表示应用于词义消歧也是一个很重要的研究方向, 能够更好地完成搜素引擎、意见挖掘、文本理解与产生、推理等任务.

5) 语义信息丰富的语言.

汉语的单词嵌入表示最近引起了相当大的关注, 汉字及其部首偏旁成分包含丰富的语义信息, 都能用于学习汉语的单词嵌入表示. 汉字包涵意思、结构和读音, 现有的嵌入学习方法主要关注汉字的结构和意思. Yang等人[87]就提出了一个方法: 语音增强的汉语单词嵌入表示, 一种能够完全利用汉字所代表的信息, 包括读音、形态、语义的嵌入表示学习方法, 即将上下文字符和目标汉字的发音同时编码到嵌入表示中. 在词语相似度评价、词语类比推理、文本分类和情感分析等任务上都验证了该方法的有效性. 鉴于这些工作已经取得的进展, 可以期待更多单词嵌入表示应用到其他语义信息丰富的语言中, 已完成翻译等任务.

6) 多模态学习.

单词嵌入表示早已不是独立存在的个体, 它已经在很多领域与其它模型方法相结合了, 上文也有提及, 比如, 与图像相结合, 提高图像识别的性能及应用领域等. Wang等人[88]将单词嵌入表示应用到遥感领域, 提出基于单词嵌入表示和端到端深度学习的遥感图像描述相结合的方法, 通过精确、简洁的自然语句描述, 实现了复杂遥感图像中胡杨和柽柳的分类识别. 笔者也期待单词嵌入表示应用于更多的行业中, 如医学影像、气象遥感等.

7) 简化训练过程.

BERT模型的提出掀起了注意力机制的热潮, 但是语言模型预训练方法往往需要大量的计算机资源, 很多研究者和机构无法轻易拥有相应的资源, 从而, 阻碍了单词嵌入表示研究的发展. 因此, 简化模型、减少计算量、缩短训练模型的时间也是一件很有挑战很有意义的事情.

7 结论与展望

本文对一些基本单词嵌入表示的模型及其变体进行了分析和研究, 对单词嵌入表示的方法和理论做了综述, 将单词嵌入表示按照语言数量分为单语言单词嵌入表示和跨语言单词嵌入表示, 在单语言中, 针对是否考虑上下文信息, 将单词嵌入表示模型再次分类, 字符级和单词级的单词嵌入表示多数都没有考虑上下文信息, 而后续的模型都注意到了多义词问题, 考虑了上下文信息.

第2节主要阐述了字符、单词、短语、句子嵌入表示之间的关系, 它们各有优缺点. 单词嵌入表示是目前自然语言处理任务中常用的将单词向量化的方法, 但是通常会忽略有关单词形态和形状的信息, 对于像词性标注这样的任务, 特别是在处理形态丰富的语言时, 单词内信息不可忽视, 所以有人研究提出了字符级模型, 也有人将字符表示与单词相结合, 如C2W模型. 也有人认为单词组成的短语是不应该随便拆分的, 学习短语更有助于语义的学习. Mikolov也发现当前的单词表示, 会丢失单词的顺序, 并且忽略了单词的语义, 所以提出了一种无监督的段落向量算法, 它可以从句子、段落和文档等长度可变的文本片段中学习固定长度的特征表示, 在情感分析等任务中表现出色. Levy等人对Mikolov等人提出的负抽样的Skip–Gram模型进行扩展, 引入基于依赖的上下文信息, 进行实验, 得到的模型主题性更少, 功能相似性更好.

第3节主要介绍了考虑上下文信息的单词嵌入表示, CoVe模型为NLP任务带来了很大的改进, 也为之后的预训练技术奠定了基础, 而ELMo更是刺激了NLP的发展, 后续有大量的考虑上下文信息单词表示的研究, 也使人们的注意力从大的无标记文本中捕获语言信息转移到了下游任务, 也是BERT模型和VIB模型的基础. VIB对嵌入表示的信息进行非线性压缩, 针对依赖解析任务, 只保留有助于鉴别依赖解析的信息, 把每一个单词嵌入到一个离散的标签或者连续的向量中, 避免了对压缩表示中的可用的维数的不必要使用, 利用随机映射去除了不需要的部分信息.

第4节是对跨语言单词嵌入表示模型的介绍, 由于世界上语言很多, 但只有少数语言有丰富的注释资源, 这就需要单词嵌入表示的跨语言迁移学习, 在丰富资源语言上训练的模型应用到低资源语言上, 输入嵌入表示被投影到共享的语义空间中, 跨语言单词嵌入表示不仅仅能用于机器翻译, 还能用于其它NLP任务, 缩减翻译时间.

第5节是将多种方法或者多个模态进行组合, 得到

效果更好的模型, 是对前面模型更深入的应用, 不仅仅局限于单词短语的嵌入表示, 还与多视图等其它研究融合, 这也体现了当今机器学习, 计算机视觉, 语音信号处理和自然语言理解之间, 共同发展相互融合的思想.

最后, 本文的第6节详细阐述了单词嵌入表示的未来趋势及发展方向, 进一步说明单词嵌入表示的研究价值和应用潜力.

总之, 从本文总结的工作中可以看出, 单词嵌入表示是目前研究的一个热点, 目前已经取得了大量的成果, 拥有重要的理论价值和应用价值, 有力地推动着自然语言处理的研究及应用. 随着理论研究的进一步深入和应用领域的进一步扩展, 单词嵌入表示必将会发挥越来越重要的作用.

参考文献:

[1] JONES M N, MEWHORT D J. Representing word meaning and order information in a composite holographic lexicon. *Psychological Review*, 2007, 114(1): 1.

[2] WANG P, QIAN Y, SOONG F K, et al. Part-of-speech tagging with bidirectional long short-term memory recurrent neural network. *arXiv Preprin*. ArXiv: 1510.06168, 2015.

[3] CHEN D, Manning C D. A fast and accurate dependency parser using neural networks. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: ACL, 2014, 740 – 750.

[4] LAMPLE G, BALLESTEROS M, SUBRAMANIAN S, et al. Neural architectures for named entity recognition. *arXiv Preprint*. ArXiv: 1603.01360, 2016.

[5] ZHOU J, XU W. End-to-end learning of semantic role labeling using recurrent neural networks. *Processing of the Asian Federation of Natural Language Processing*. Stroudsburg: ACL, 2015: 1127 – 1137.

[6] ZOU W Y, SOCHER R, CER D, et al. Bilingual word embeddings for phrase-based machine translation. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: ACL, 2013: 1393 – 1398.

[7] TURIAN J, RATINOV L, BENGIO Y. Word representations: A simple and general method for semi-supervised learning. *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. Stroudsburg: ACL, 2010: 384 – 394.

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

[9] AL-RFOU R, PEROZZI B, SKIENA S. Polyglot: Distributed word representations for multilingual NLP. *arXiv Preprint*. ArXiv: 1307.1662, 2013.

[10] SOCHER R, PENNINGTON J, HUANG E H, et al. Semi-supervised recursive autoencoders for predicting sentiment distributions. *Proceedings of Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: ACL, 2011: 151 – 161.

[11] HARRIS Z S. Distributional structure. *Word*, 1954, 10(2/3): 146 – 162.

[12] FIRTH J R. A synopsis of linguistic theory, 1930 – 1955. *Studies in Linguistic Analysis*. 1957, DOI: 10.1007/s10884-006-9039-9.

[13] BENGIO Y, DUCHARME R, VINCENT P, et al. A neural probabilistic language model. *The journal of machine learning research*, 2003, 3: 1137 – 1155.

[14] MIKOLOV T, CHEN K, CORRADO G, et al. Efficient estimation of word representations in vector space. *arXiv preprint*. ArXiv: 1301.3781, 2013.

[15] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality. *arXiv Preprint*. ArXiv: 1310.4546, 2013.

[16] DOS SANTOS C, ZADROZNY B. Learning character-level representations for part-of-speech tagging. *Proceedings of the 31st International Conference on Machine Learning*. Beijing: PMLR, 2014: 1818 – 1826.

[17] KIM Y, JERNITE Y, SONTAG D, et al. Character-aware neural language models. *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. Menlo Park, CA: AAAI, 2016: 2741 – 2749.

[18] LECUN Y, BOSER B E, DENKER J S, et al. Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 1990, 2(2): 396 – 404.

[19] SRIVASTAVA R K, GREFF K, SCHMIDHUBER J. Training very deep networks. *arXiv Preprint*. ArXiv: 1507.06228, 2015.

[20] HOCHREITER S, SCHMIDHUBER J. Long short-term memory. *Neural Computation*, 1997, 9(8): 1735 – 1780.

[21] MIKOLOV T, KARAFIÁT M, BURGET L, et al. Recurrent neural network based language model. *Proceedings of the 11th Annual Conference of the International-Speech-Communication-Association 2010*. Makuhari, JAPAN: International Speech Communication Association, 2010: 1045 – 1048.

[22] SRIVASTAVA R K, GREFF K, SCHMIDHUBER J. Training very deep networks. *arXiv Preprint*. ArXiv: 1507.06228, 2015.

[23] LING W, LUÍS T, MARUJO L, et al. Finding function in form: Compositional character models for open vocabulary word representation. *arXiv Preprint*. ArXiv: 1508.02096, 2015.

[24] GRAVES A, SCHMIDHUBER J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 2005, 18(5/6): 602 – 610.

[25] CHEN X, XU L, LIU Z, et al. Joint learning of character and word embeddings. *Proceedings of the 24th International Joint Conference on Artificial Intelligence*. Buenos Aires, ARGENTINA: International Joint Conferences on Artificial Intelligence, 2015: 1236 – 1242.

[26] MIKOLOV T, SUTSKEVER I, DEORAS A, et al. Subword language modeling with neural networks. *Preprint*(http://www.Fit.Vutbr.Cz/imikolov/rnnlm/char.Pdf), 2012, 8: 67.

[27] PENNINGTON J, SOCHER R, MANNING C D. Glove: Global vectors for word representation. *Proceedings of 2014 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: ACL, 2014: 1532 – 1543.

[28] TIAN F, DAI H, BIAN J, et al. A probabilistic model for learning multi-prototype word embeddings. *Proceedings of the 25th International Conference on Computational Linguistics*. New York: ACM, 2014: 151 – 160.

[29] MORIN F, BENGIO Y. Hierarchical probabilistic neural network language model. *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics*. Bridgetown, Barbados: The Society for Artificial Intelligence and Statistics, 2005: 246 – 252.

[30] MNIH A, KAVUKCUOGLU K. Learning word embeddings efficiently with noise-contrastive estimation. *Advances in Neural Information Processing Systems*, 2013, 26: 2265 – 2273.

[31] LIU P, QIU X, HUANG X. Learning context-sensitive word embeddings with neural tensor Skip–Gram model. *Proceedings of the 24th International Joint Conference on Artificial Intelligence*. Buenos Aires, Argentina: International Joint Conferences on Artificial Intelligence, 2015: 1284 – 1290.

[32] HUANG E H, SOCHER R, MANNING C D, et al. Improving word representations via global context and multiple word prototypes. *Proceedings of the 24th International Joint Conference on Artificial Intelligence*. Stroudsburg: ACL, 2012: 873 – 882.

[33] LEVY O, GOLDBERG Y. Dependency-based word embeddings. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Stroudsburg: ACL, 2014: 302 – 308.

[34] LE Q, MIKOLOV T. Distributed representations of sentences and documents. *Proceedings of the 31st International Conference on Machine Learning*. New York: ACM, 2014: 2931 – 2939.

[35] YU M, DREDZE M. Learning composition models for phrase embeddings. *Transactions of the Association for Computational Linguistics*, 2015, 3: 227 – 242.

[36] MCCANN B, BRADBURY J, XIONG C, et al. Learned in translation: Contextualized word vectors. *arXiv Preprint*. ArXiv: 1708.00107, 2017.

[37] PETERS M E, NEUMANN M, IYYER M, et al. Deep contextualized word representations. *arXiv Preprin*. ArXiv: 1802.05365, 2018.

[38] DEVLIN J, CHANG M, LEE K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv Preprint*. ArXiv: 1810.04805, 2018.

[39] CHELBA C, MIKOLOV T, SCHUSTER M, et al. One billion word benchmark for measuring progress in statistical language modeling. *arXiv Preprint*. ArXiv: 1312.3005, 2013.

[40] TISHBY N, PEREIRA F C, BIALEK W. The information bottleneck method. *ArXiv Preprint*. Physics/0004057, 2000.

[41] LI X L, EISNER J. Specializing word embeddings (for parsing) by information bottleneck. *arXiv Preprint*. ArXiv: 1910.00163, 2019.

[42] SCHNABEL T, LABUTOV I, MIMNO D, et al. Evaluation methods for unsupervised word embeddings. *Proceedings of Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: ACL, 2015: 298 – 307.

[43] TSVETKOV Y, FARUQUI M, LING W, et al. Evaluation of word vector representations by subspace alignment. *Proceedings of Conference on Empirical Methods in Natural Language Processing*. Stroudsburg: ACL, 2015: 2049 – 2054.

[44] CHO K, VAN MERRIENBOER B, BAHDANAU D, et al. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv Preprint*. ArXiv: 1409.1259, 2014.

[45] BOWMAN S R, ANGELI G, POTTS C, et al. A large annotated corpus for learning natural language inference. *arXiv Preprint*. ArXiv: 1508.05326, 2015.

[46] WILLIAMS A, NANGIA N, BOWMAN S R. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv Preprint*. ArXiv: 1704.05426, 2017.

[47] RAJPURKAR P, ZHANG J, LOPYREV K, et al. Squad: 100, 000+ questions for machine comprehension of text. *arXiv Preprint*. ArXiv: 1606.05250, 2016.

[48] RAJPURKAR P, JIA R, LIANG P. Know what you don't know: Unanswerable questions for SQuAD. *arXiv Preprint*. ArXiv: 1806.03822, 2018.

[49] CUI Y, CHE W, LIU T, et al. Pre-training with whole word masking for chinese bert. *arXiv Preprint*. ArXiv: 1906.08101, 2019.

[50] SUN Y, WANG S, LI Y, et al. Ernie: Enhanced representation through knowledge integration. *arXiv Preprint*. ArXiv: 1904.09223, 2019.

[51] JOSHI M, CHEN D, LIU Y, et al. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 2020, 8: 64 – 77.

[52] RAFFEL C, SHAZEER N, ROBERTS A, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv Preprint*. ArXiv: 1910.10683, 2019.

[53] LAN Z, CHEN M, GOODMAN S, et al. Albert: A lite bert for self-supervised learning of language representations. *arXiv Preprint*. ArXiv: 1909.11942, 2019.

[54] WANG Y, HOU Y, CHE W, et al. From static to dynamic word representations: a survey. *International Journal of Machine Learning and Cybernetics*, 2020: 1 – 20.

[55] RUDER S, VULI I, SGAARD A. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 2019, 65: 569 – 631.

[56] UPADHYAY S, FARUQUI M, DYER C, et al. Cross-lingual models of word embeddings: An empirical comparison. *arXiv Preprint*. ArXiv: 1604.00425, 2016.

[57] GUO J, CHE W, WANG H, et al. Learning sense-specific word embeddings by exploiting bilingual resources. *Proceedings of the 25th International Conference on Computational Linguistics*. New York: ACM, 2014: 497 – 507.

[58] GOUWS S, BENGIO Y, CORRADO G. Bilbowa: Fast bilingual distributed representations without word alignments. *Proceedings of the 32nd International Conference on Machine Learning*. New York: ACM, 2015: 748 – 756.

[59] COULMANCE J, MARTY J, WENZEK G, et al. Trans–Gram, fast cross-lingual word-embeddings. *arXiv Preprint*. ArXiv: 1601.02502, 2016.

[60] MULCAIRE P, KASAI J, SMITH N A. Polyglot contextual representations improve crosslingual transfer. *arXiv Preprint*. ArXiv: 1902.09697, 2019.

[61] LAMPLE G, CONNEAU A. Cross-lingual language model pretraining. *arXiv Preprint*. ArXiv: 1901.07291, 2019.

[62] CONNEAU A, KHANDELWAL K, GOYAL N, et al. Unsupervised cross-lingual representation learning at scale. *arXiv Preprint*. ArXiv: 1911.02116, 2019.

[63] SCHUSTER T, RAM O, BARZILAY R, et al. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. *arXiv Preprint*. ArXiv: 1902.09492, 2019.

[64] WANG Y, CHE W, GUO J, et al. Cross-lingual BERT transformation for zero-shot dependency parsing. *arXiv Preprint*. ArXiv: 1909.06775, 2019.

[65] MULCAIRE P, KASAI J, SMITH N A. Low-resource parsing with crosslingual contextualized representations. *arXiv Preprint*. ArXiv: 1909.08744, 2019.

[66] WESTON J, BENGIO S, USUNIER N. Large scale image annotation: learning to rank with joint word-image embeddings. *Machine Learning*, 2010, 81(1): 21 – 35.

[67] USUNIER N, BUFFONI D, GALLINARI P. Ranking with ordered weighted pairwise classification. *Proceedings of the 26th International Conference On Machine Learning*. New York: ACM, 2009: 1057 – 1064.

[68] FROME A, CORRADO G, SHLENS J, et al. Devise: A deep visual-semantic embedding model. *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2013: 2121 – 2129.

[69] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 2012, 25: 1097 – 1105.

[70] KIROS R, SALAKHUTDINOV R, ZEMEL R S. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv Preprint*. ArXiv: 1411.2539, 2014.

[71] KIROS R, ZEMEL R S, SALAKHUTDINOV R. A multiplicative model for learning distributed text-based attribute representations. *arXiv Preprint*. ArXiv: 1406.2710, 2014.

[72] CHEN W, ZHANG Y, ZHANG M. Feature embedding for dependency parsing. *Proceedings of the 25th International Conference on Computational Linguistics*. New York: ACM, 2014: 816 – 826.

[73] MIKOLOV T, KOPECKY J, BURGET L, et al. Neural network based language models for highly inflective languages. *Proceedings of 2009 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Piscataway, NJ: IEEE, 2009: 4725 – 4728.

[74] LU J, BATRA D, PARIKH D, et al. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv Preprint*. ArXiv: 1908.02265, 2019.

[75] SU W, ZHU X, CAO Y, et al. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv Preprint*. ArXiv: 1908.08530, 2019.

[76] SHARMA P, DING N, GOODMAN S, et al. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. Stroudsburg: ACL, 2018: 2556 – 2565.

[77] SUN C, MYERS A, VONDRICK C, et al. Videobert: A joint model for video and language representation learning. *Proceedings of the 17th IEEE/CVF International Conference on Computer Vision*. Seoul, Korea: IEEE, 2019: 7463 – 7472.

[78] PENG K, YIN C, RONG W, et al. Named entity aware transfer learning for biomedical factoid question answering. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2021, DOI: 10.1109/TCBB.2021.3079339.

[79] HAJIAMINSHIRAZI S, MOMTAZI S. Cross-lingual embedding for cross-lingual question retrieval in low-resource community question answering. *Machine Translation*, 2020, 34(4): 287 – 303.

[80] ALHARBI N M, ALGHAMDI N S, ALKHAMMASH E H, et al. Evaluation of sentiment analysis via word embedding and rnn variants for amazon online reviews. *Mathematical Problems in Engineering*, 2021, 2021: 1 – 10.

[81] DOVDON E, BATSUURI S. Text2Plot: Sentiment analysis by creating 2D plot representations of texts. *IEEJ Transactions on Electrical and Electronic Engineering*, 2021, 16(6): 852 – 860.

[82] DONG Y, WANG J, WANG J. Multi-task learning network based on attention for aspect-based sentiment analysis. *IOP Publishing*, 2021, 1827(1): 012173.

[83] SINGH J, SINGH A K. Semrank: A semantic similarity-based tweets ranking approach. *International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)*, 2021, 15(3): 74 – 96.

[84] ZHAO F, ZHU Z, HAN P. A novel model for semantic similarity measurement based on wordnet and word embedding. *Journal of Intelligent & Fuzzy Systems*. 2021(Preprint): 1 – 12.

[85] JIA L, TANG J, LI M, et al. TWE–WSD: An effective topical word embedding based word sense disambiguation. *CAAI Transactions on Intelligence Technology*, 2021, 6(1): 72 – 79.

[86] LI S, PAN R, LUO H, et al. Adaptive cross-contextual word embedding for word polysemy with unsupervised topic modeling. *Knowledge-Based Systems*, 2021, 218: 106827.

[87] YANG Q, XIE H, CHENG G, et al. Pronunciation-enhanced chinese word embedding. *Cognitive Computation*, 2021, 13(3): 688 – 697.

[88] WANG Y, MA H, ALIFU K, et al. Remote sensing image description based on word embedding and end-to-end deep learning. *Scientific Reports*, 2021, 11(1): 1 – 13.

作者简介:

刘建伟 博士, 副教授, 目前研究方向为智能信息处理、复杂非线性系统分析、预测、控制、算法分析与设计, E-mail: liujw@cup.edu.cn;

高 悦 硕士研究生, 目前研究方向为机器学习, E-mail: 940024427@qq.com.