基于深度学习的中文网络招聘文本中的技能词抽取方法

文益民[1,2]，杨 鹏[1]，文博奚[3]，蔡 翔[3]

（1.桂林电子科技大学 广西可信软件重点实验室，广西 桂林 541004；
2.桂林电子科技大学 广西图像图形智能处理重点实验室，广西 桂林 541004；
3.桂林电子科技大学 商学院，广西 桂林 541004）

摘 要：为了能够充分利用领域知识来提升技能词的抽取性能，提出了一种基于深度学习与语料特征相结合的技能词抽取方法。将技能词抽取转化为序列标注问题，以序列标注的基本模型Bi-LSTM-CRF为基础，在输入层中加入语料特征，并将输入层的输出与Bi-LSTM输出连接在一起作为CRF层的输入。实验结果表明，提出的技能词抽取方法效果提升明显，加入的语料特征有利于提升技能词抽取的准确率，并能够缓解标注数据的稀缺。

关键词：网络招聘；技能词；序列标注；深度学习

中图分类号：TP391 **文献标志码**：A **文章编号**：1673-808X(2020)04-0338-11

DOI:10.16725/j.cnki.cn45-1351/tn.2020.04.014

Skill word extraction from Chinese online recruitment texts based on deep learning

WEN Yimin[1,2], YANG Peng[1], WEN Boxi[3], CAI Xiang[3]

(1. Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, China;
2. Guangxi Key Laboratory of Image and Graphic Intelligent Processing,
Guilin University of Electronic Technology, Guilin 541004, China;
3. School of Business, Guilin University of Electronic Technology, Guilin 541004, China)

Abstract: In order to make full use of domain knowledge to improve the performance of skill word extraction, a method of skill word extraction based on the combination of deep learning and corpus features is proposed. Skill word extraction is transformed into a sequence tagging problem. Based on the basic model of sequence tagging, Bi-LSTM-CRF, which adds corpus features to the input layer, and connects the output of the input layer with the output of the Bi-LSTM layer as the input of the CRF layer. The results of a large number of experiments show that the effect of the proposed skill word extraction method has improved significantly. The added corpus features can help improve the accuracy of skill word extraction and reduce the effort of data annotations.

Key words: online recruitment text; skill words; sequence labeling; deep learning

近年来，随着我国高等教育的迅猛发展，大学毕业生日益增多。尽管就业岗位的数量也在不断增加，但中国劳动力市场的供需失配的结构性问题依然非常严重。根据《中国劳动力市场技能缺口研究》报告，仅2016年中国就有超过1 200万的本科和高职专科毕业生。虽然中国拥有如此庞大的劳动力供应，但许多企业仍然很难招聘到合适的人才。因此，如何准确地了解企业需求，提高人才培养针对性是当务之急。随着互联网的普及，网络招聘成为企业招聘人才的主流方式。招聘信息中含有企业对所招岗位专业能力的具体描述，反映了企业的需求。对网络招聘数据进行分析，以提取工作岗位要求中的专业能力需求，可以解

收稿日期：2020-04-29

基金项目：国家自然科学基金（61866007，71463010）；广西自然科学基金（2018GXNSFDA138006）；广西学位与研究生教育改革课题（JGY2017055）；教育部人文社会科学研究项目（17JDGC022）

通信作者：文益民（1969－），男，教授，博士，研究方向为机器学习、模式识别、媒体和图像挖掘、教育数据挖掘。E-mail:ymwen2004@aliyun.com

引文格式：文益民，杨鹏，文博奚. 基于深度学习的中文网络招聘文本中的技能词抽取方法[J]. 桂林电子科技大学学报，2020，40(4)：338-348.

决这一难题。如今,一些研究尝试利用网络招聘信息分析企业招聘岗位对专业能力的需求。在国外,Kim等[1]通过对数据科学家岗位的招聘信息进行人工分析,总结出企业对数据科学家这个岗位所需的专业能力以及学历等各方面的要求。De 等[2]对 2 700 条大数据相关岗位信息进行分析,结合专家判断,将大数据相关岗位划分为 4 个岗位簇,并对每个岗位簇所需的技能进行评估。在国内,黄崑等[3]从智联招聘网站上收集了 2 615 份有关数据分析、数据管理和数据挖掘岗位的招聘信息,从岗位基本信息、岗位职责、任职要求 3 个角度分析了这些岗位所招聘人才需要掌握的知识和能力要求,并对图书馆情报学科人才培养方案提出建议。

然而,网络招聘信息往往为非结构化文本,自动、准确地从招聘文本中抽取技能词并非易事。为简洁起见,将岗位对所需人才的专业知识和专业能力的要求称为技能词。技能词可以看成是特定专业领域内的命名实体[4]或者术语[5]。

因此,网络招聘文本技能词的抽取任务可以借鉴命名实体识别或者术语抽取的方法。但是,通过对网络招聘文本中技能词的分析,可看出技能词抽取与命名实体识别和术语抽取相比,具有其自身的特殊性与复杂性。具体表现在:1)部分技能词类似于术语,用于表达各专业领域的特殊概念,仅仅在该领域特定上下文中才表示为技能词,如“操作系统”“自然语言处理”“机器学习”等。2)部分技能词具有与命名实体相似的特点,是文本中具有很强特定意义的实体,并且不受上下文影响,在任何场景下出现都表示为技能词,如“Java”“Linux”“算法”等。3)技能词不同于一般的命名实体(人名、地名、机构名等),没有明确的关于技能词的定义,不能清晰地界定技能词的边界,并且技能词形式多变,表现在长度、组成模式等方面,如“C#”“数据库/表结构索引设计”,还有类似于“Deep Learning 框架”和“Neo4j”等中英文混合、数字英文混合词语等形式。4)大量技能词在表述时多采用英译词或英文缩写,如“J2EE”和“JavaEE”等。5)已有词表或者技能词图谱不足以涵盖全部技能词,而且随着技术的不断进步,新的技能词会不断出现,如“联邦学习”“胶囊网络”等。6)技能词存在嵌套形式,如“Linux 操作系统”,其中“Linux”和“操作系统”又分别作为技能词出现。此外,不同于正式文本,招聘信息通常很不规范,常常包含许多拼写错误的技能词,如将“Excel” 错拼为“Excle”,“Linux”错拼为“Liunx”等。如果直接按照词频进行精确匹配,将不能识别拼写错误的技能词,从而影响技能词抽取的准确率。目前,虽然命名实体识别的相关工作很多,但重点都是在正式文本中识别人名、地名和机构名,而术语抽取的相关工作主要针对特定领域内的术语识别,缺乏通用性和可移植性。由此可见,从招聘文本中抽取技能词仍然是一项颇为艰巨的任务。

选取 IT 类网络招聘数据作为研究对象,分析了技能词的特点,制定了相应的标注规则,对从招聘网站上获取的招聘语料预处理后进行人工标注,将技能词抽取任务转化为序列标注问题,充分挖掘招聘语料的各类型特征,并分析了各类语料特征对技能词抽取结果所起的作用。

本研究的主要贡献包括 4 个方面:

1) 提出了一种基于深度学习的序列标注模型与语料特征相结合的网络招聘文本中的技能词抽取方法;

2) 与经典的序列标注模型相比,本模型引入了更多的语料特征,并将输入层的输出与 Bi-LSTM 输出连接在一起作为 CRF 层的输入,模型的性能得到了极大提高;

3) 通过实验评估了本模型中加入的语料特征是否能够缓解模型对大量标注数据的依赖;

4) 进行了大量实验,实验结果表明,本模型能够从网络招聘文本中自动、准确地抽取技能词。

1　相关工作

本研究将技能词抽取任务转化为序列标注问题,类似于命名实体识别和术语抽取。因此,回顾与技能词抽取密切相关的研究,这些相关研究工作包括不同种类技术的命名实体识别和术语抽取方法。

命名实体识别是自然语言处理中一项基本任务,旨在识别特殊实体并将其分类为预定义类别的任务,例如产品名称、旅游景点名称、新闻领域的人名或生物医学领域的疾病名称[4]。术语抽取也是自然语言处理中一项基础任务,指从文本中自动发现术语的过程[4],它可以应用于信息检索、关系抽取、对话生成等复杂任务领域。随着信息技术的不断发展,各领域数据不断扩张,产生大量的领域术语。这些领域术语在该领域中具有很强的特定意义,是构成该领域专业文本的信息主体[5]。与术语识别类似,命名实体识别也是从一段自然语言文本中识别特定类型的实体。因此,命名实体识别也逐渐由识别通用对象转向识别特定领域术语,特定领域术语抽取也逐渐采用命名识别中的方法。另外,从本质上讲,如果仅仅抽取文本中术语的名称或简称,不标注术语的具体表述或评价,术语抽取可被视为一类命名实体识别[6]。当前,用于命名实体识别和术语抽取的方法可简单地分为传统

方法和基于深度学习的方法。

传统方法大致可分为 3 类：基于规则、无监督学习和基于特征的监督学习方法。基于规则的命名实体识别方法依赖于人工制定的规则，而规则的设计一般是基于句法、语法、词汇的模式以及特定的领域知识等。例如可通过结合中文姓氏词典与词性特征来识别中文人名“张三”[7]。同样，基于规则的术语抽取方法也是主要通过分析语料，制定术语抽取规则。这些规则的制定依赖于领域知识，需要人工编制，很大程度上受限于人所具有的知识。Chen 等[8]根据上下文信息和术语组成规则来实现术语抽取。由于不同领域的命名实体和术语组成方面有所差异，具有不同的特征，需要构建各式各样的规则，因此这类方法移植性很差。无监督学习的命名实体方法利用上下文语义相似性，从聚集的词组中提取命名实体。Zhang 等[9]提出了一种借助语料库统计信息（例如逆文档频率和上下文向量）和浅层语法知识（例如名词短语分块）来推断命名实体。基于无监督学习的术语抽取方法通常先从语料库中选取候选术语，然后利用统计信息（例如使用词频和词长度、互信息等）计算候选术语成为术语的可能性。Pantel 等[10]提出了一种基于语料库统计的术语抽取方法，通过计算 2 个词语或 2 个词语组成的词串之间的互信息和似然比相结合的方法，对候选术语进行评分，从而实现对术语的抽取。这类方法不仅依赖中文分词的准确性，而且存在无法识别低频命名实体和术语的缺陷。基于特征的有监督学习方法是目前常用的方法，利用监督学习将命名实体识别和术语抽取任务转化为序列标注问题。根据标注好的数据，研究者应用领域知识与工程技巧设计复杂的特征（如单词特征、上下文特征、词典及词性特征等）来表征每个训练样本，然后应用机器学习方法（如支持向量机[11-12]（support vector machine，简称 SVM），隐马尔科夫模型[13-14]（hidden Markov model，简称 HMM），条件随机场[15-16]（conditional random field，简称 CRF）等训练模型对词语进行分类。这类方法需要精细的人工特征以及大量的领域知识，很难识别新的较长的组合型命名实体和术语。

近年来，深度神经网络已成为有效提取命名实体或术语潜在特征的主流方法。标准的做法是运用不同的神经网络模型（卷积神经网络，convolutional neural networks，简称 CNN；长期短期记忆，long short term memory，简称 LSTM）提取不同类型的表示形式（字符级或单词级），将学习到的特征表示馈入顶部的 CRF 层中以进行序列标签预测，从而实现命名实体识别或术语抽取。Collobert 等[17]利用 CNN 网络学习单词级特征表示，将学习到的特征表示馈入 CRF 进行序列标签预测。Lample 等[18]利用 Bi-LSTM 网络结构提取字符的上下文潜在特征表示，并利用 CRF 层进行标签解码。Huang 等[19]提出了一个与 LSTM-CRF 类似的模型，但是加入了人工构造的特征（例如，单词是否以大写字母开头，字母的前缀和后缀等）。首先通过 Bi-LSTM 网络提取了字符的上下文潜在特征表示，然后将其与人工构造特征相连接，最后将其输入到 CRF 层中。CNN-CRF 模型与 LSTM-CRF（bidirectional long short term memory-conditional random field）模型的区别在于，CNN 网络对提取词的形态信息（例如词的前缀和后缀）非常有效，主要适用于英文处理，英文单词是由更细粒度的字母组成，这些字母潜藏着一些特征，而中文单字无法分解，在中文中可能并不适用。但 CNN 网络在长序列输入上特征提取能力弱，而 LSTM 提供了长距离的依赖，拥有较强的长序列特征提取能力。因此，有研究者为了充分利用这 2 个模型的优点，将两者结合。Ma[20]提出了 Bi-directional LSTM-CNNs-CRF 网络架构。在该架构中，CNN 网络首先用于提取单词的字符级特征表示，将 CNN 网络提取出的字符级特征表示与预训练的词嵌入连接起来，然后再馈入 LSTM 网络以提取单词级的上下文潜在特征表示，最后进入 CRF 层进行标签解码。这些基于神经网络的方法不需要特定于任务的特征工程，便可获得良好的性能。

深度学习也逐渐应用在术语抽取研究工作中。闫兴龙等[21]提出了一种基于带条件随机场的双向长期短期记忆（Bi-LSTM-CRF）的新型网络安全实体识别模型，以从非结构化文本中提取与安全相关的概念和实体。赵东玥等[22]同样采用了 Bi-LSTM 模型对科技文献进行术语抽取。赵洪等[23]以 Bi-LSTM-CRF 模型为基础框架，融入了理论术语的语料特征（例如理论术语的尾词特征、术语的构词特征等），构建了基于深度学习的理论术语抽取模型，并提出一种自训练算法，以实现模型的弱监督学习。

另外，目前针对中文技能词提取的研究工作非常少。俞琰等[24]从前途无忧招聘网站中抓取了 10 000 条计算机领域和 30 000 条非计算机领域的招聘信息，利用依存句法分析从计算机领域的招聘信息中选取候选技能，再利用非计算机领域的招聘信息计算候选技能中每个词的领域相关性并与候选技能的词频、词长等统计信息相结合得出融合领域相关性的 C-value 值，按值降序排列。最终选取前 N 个候选技能信息作为被抽取的技能词进行人工判定是否正确。

鉴于深度学习模型在序列标注问题上的优异性能,拟将深度学习方法应用于技能词抽取,并根据技能词的特点,充分挖掘语料的各类型特征与深度学习方法相结合,构建面向招聘领域的技能词抽取模型。由于是首次开展基于深度学习抽取技能词的研究,采用 Bi-LSTM-CRF 模型[15]作为本研究技能词抽取模型的基础框架。但与之相比,文献[15]针对英文正式文本的命名实体识别,而本研究的工作是针对中文招聘文本的技能词抽取属于非正式文本,具有更强的干扰性。此外,在输入层中加入语料特征,并将输入层的输出与 Bi-LSTM 层的输出相连接作为 CRF 层的输入。而文献[15]并未将人工构造的特征经过输入层和 Bi-LSTM 层,而是直接连接到 Bi-LSTM 的输出。赵洪等[23]的工作与本研究最为相似,同样采用 Bi-LSTM-CRF 模型[15]作为基础框架。但不同之处在于,本研究是针对招聘领域的技能词抽取,输入层中加入的语料特征的选取更多的是考虑到结合与技能词所相关的特性,所构造的是基于字符级的语料特征。文献[23]所选取的理论术语的语料特征更多是从术语的构词特征考虑,并且基于词语级的特征。此外,在进入 CRF 层前,还要将输入层的输出与 Bi-LSTM 层的输出进行拼接以更好地提升模型的预测性能。俞琰等[24]针对网络招聘数据的技能词抽取,但采用的是无监督学习方法,需要利用词频、词长等统计信息来抽取技能词,并且还需要依靠人工判定抽取出的技能词是否准确。而本研究是采用有监督学习方法,通过学习训练文本特征,构造模型来抽取技能词,能够弥补无监督学习方法无法识别低频技能词的缺陷。

2　基于深度学习的技能词抽取模型

提出的技能词抽取模型如图 1 所示。模型由 4 个层次的模块组成,即输入层、Bi-LSTM 层、特征拼接层和 CRF 层。在输入层中,将每个输入语句转换为一系列字符特征向量,再与输入语句中各个字符的位置特征(Seg)、输入语句分词后的词性特征(Pos)和技能词的上下文特征(Con)进行拼接,将其输入到 Bi-LSTM 层中,以将上下文信息顺序编码为固定长度的隐藏向量,接着在特征拼接层中将输入层的输出与 Bi-LSTM 层的输出连接在一起,作为 CRF 层的输入。最后由 CRF 层预测出最佳标签序列作为整个网络的输出。

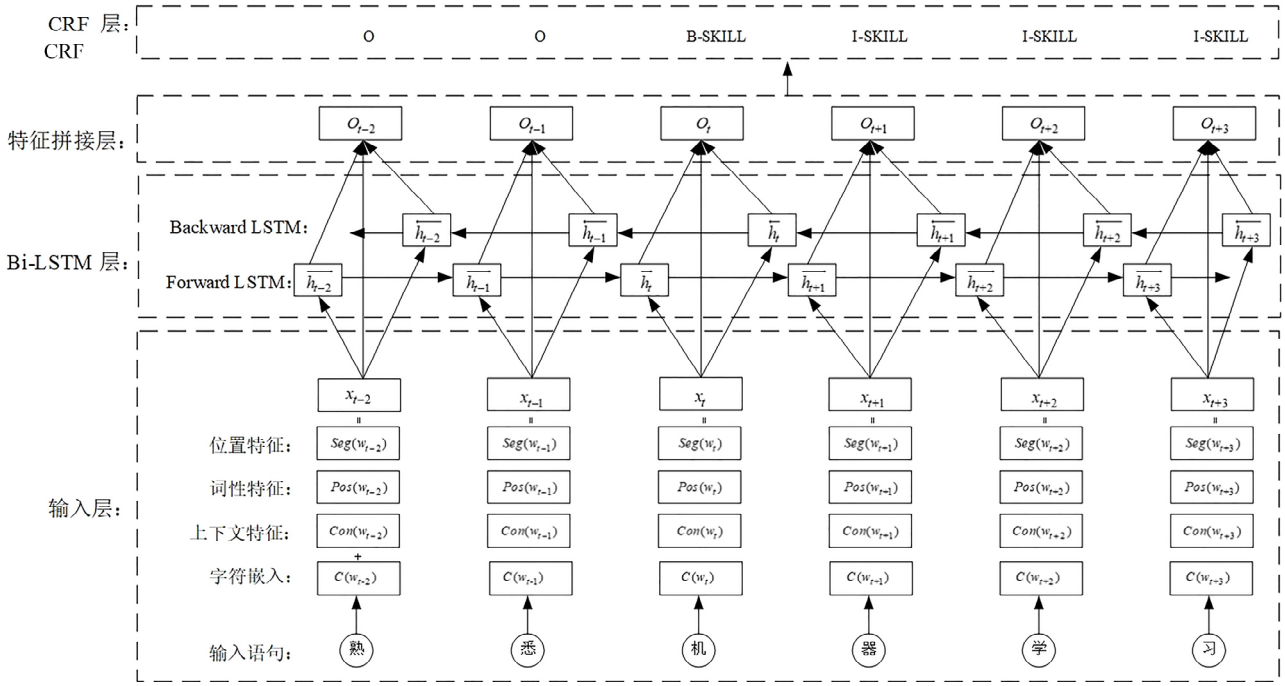

图 1　基于深度学习的技能词抽取模型

2.1　输入层

在输入层中分为 2 个步骤。1)将输入语句转换为字符级密集向量序列。先生成包含语料库中所有字符的字典,再用一个嵌入矩阵 $\boldsymbol{M} \in \mathbf{R}^{D\times V}$ 将每个字符映射为一个密集矢量,其中 D 为嵌入向量维度,V 为字典中所有字符的总量。输入句子表示为 $\boldsymbol{S}=[w_1,w_2,\cdots,w_n]$,其中 n 是句子的长度,$w_i \in \mathbf{R}^V$ 是输入语句中第 i 个字符在字典中的 one-hot 表示。句子的字符嵌入向量表示为 $[\boldsymbol{c}_1,\boldsymbol{c}_2,\cdots,\boldsymbol{c}_n]$,其中 $\boldsymbol{c}_i \in$

$\mathbf{R}^D$。

2)加入网络招聘信息中字符的各类语料特征,与字符嵌入向量拼接。语料特征主要由 3 种特征构成:位置特征(Seg)、词性特征(Pos)和上下文特征(Con)。

位置特征(Seg)是指对输入句子进行 jieba 分词后,每个字符与所在词语的相对位置。如“操作系统”是分词后得到的词,那么“操”的位置特征标记为“0”,“作”的位置特征标记为“1”,“系”的位置特征标记为“2”,“统”的位置特征标记为“3”。词性特征(Pos)是指将输入语句进行 jieba 分词后将每个字符的词性标记为所在词语所对应的词性。如“具备”的词性为“动词”,则“具”“备”的词性都记为“动词”。根据术语特性可知,有些词只在本领域流通。而本模型所抽取的技能词大多都属于特定领域,因此可考虑在技能词抽取时加入词语的位置特征。同样,通过对技能词的分析可知,虽然技能词的构成模式有很多种,但是大部分是名词性短语,可见词性对于技能词识别是另一个重要的特征。

上下文特征(Con)是根据技能词的上下文特点构造的特征。首先,通过分析招聘语料库,随机抽取了 1 000 多条网络招聘文本后发现:包含技能词的文本通常为动宾结构,且技能词大多数为“名词/名词性短语”,如“熟悉关系型数据库”。技能词在句子中的位置主要位于动词、形容词/形容词短语或“和”“或”以及“、”等之后。例如“了解自然语言处理”“常用的机器学习算法”“掌握文本挖掘、实体抽取、词性标注等技术”。表 1 统计了网络招聘语料中技能词出现的位置。通过分析技能词出现的下文可发现,其下文使用了较多的习惯语,如“掌握 XX 能力”和“具有 XX 经验”等。

因此,在标注上下文特征时,首先将输入语句进行 jieba 分词,提取每个词的词性,然后依据如下规则进行标注。

表 1 网络招聘语料中技能词出现的位置

语料数量	技能词数量	动词后	形容词/形容词短语后	“和”“或”“、”等并列形式	其他位置
1 006	8 126	2 739	1 979	2 831	577

1)若“动词”之后出现“名词”,则将该“动词”词语组成的字符都标注为“1”,该“名词”词语组成的字符都标注为“0”。如“掌握”之后出现“名词”,则“掌”和“握”都标注为“1”。

2)若“动词”之后出现其他词性词语,则将该“动词”和其他词性词语组成的字符都标注为“0”。

3)若“形容词/形容词短语”之后出现“名词”,则将该“形容词/形容词短语”词语组成的字符都标注为“2”,该“名词”词语组成的字符都标注为“0”,如“常用的”之后出现“名词”,则“常”、“用”和“的”都标注为“2”。

4) 若“形容词/形容词短语”之后出现其他词性词语,则将该“形容词/形容词短语”和其他词性词语组成的字符都标注为“0”。

5)若以并列形式,连续出现 2 个或 2 个以上的“名词”,则将并列形式的连接字符,如“、”与“和”等都标注为“3”。

6)若以“动词 ＋名词＋名词”形式出现,则将后一个“名词”词语组成的字符都标注为“4”,该“动词”词语组成的字符都标注为“1”,第一个“名词”词语组成的字符都标注为“0”,如“具有 XX 能力”,则“具”、“有”都标注为“1”,“能”、“力”都标注为“4”。

7)若非以上诸情况,输入语句中的其他字符都标注为“0”。关于如何给每个字符标注上下文特征,举一个例子,如“具备数据库和数据结构基础。”标注为“具/1 备/1 数/0 据/0 库/0 和/3 数/0 据/0 结/0 构/0 基/4 础/4。/0”。具体技能词的上下文特征词如表 2 所示。

表 2 技能词的上下文特征词

技能词出现的位置	特征词
动词后	熟悉、具有、具备、使用、精通、熟练、理解、编写、绘制、挖掘、进行、实现、设计、掌握、通晓、处理、解决、学会…
形容词/形容词短语后	常用的、主流、扎实的、相关、优秀的、常见的、丰富的、实际的、复杂的、较好的、基本的、良好的、常识性、通用的…
“和”“或”“、”等	或、\、并、“、”、和、与、及、等…
下文特征	优先、工具、经验、实现、基础、能力、经历、成果、效果、技术、理论、操作、系统、工作、用法、原理、方向…

最后,输入层的输出由各输入语句序列中每个节点的字符特征向量、位置特征向量(Seg)、词性特征向量(Pos)和上下文特征向量(Con)四组特征向量构成,即 $\boldsymbol{X}_i=[c_i,\mathrm{seg}_i,\mathrm{con}_i]$。例如:输入语句为“具备数据库和数据结构基础”,词性特征向量表示为“0,0,1,1,1,2,1,1,1,1,1,1”,其中“0”代表动词,“1”代表名词,“2”代表连词。位置特征向量表示为“0,1,0,1,2,0,0,1,2,3,0,1”,其中“0”代表所在词语的第一个字符,“1”代表所在词语的第二个字符,……。上下文特征表示为“1,1,0,0,0,3,0,0,0,0,4,4”。

2.2 Bi-LSTM 层

LSTM 是一种特殊类型的递归神经网络(recurrent neural network,简称 RNN),它可以捕获长距离序列信息,并且在序列数据建模方面功能强大。与标准 RNN 的区别在于,LSTM 在隐藏层的神经元中加入细胞状态和输入门、遗忘门、输出门。细胞状态更新时需要同时使用输入门和遗忘门结果。具体实现为:

$$\boldsymbol{i}_t=\sigma(\boldsymbol{W}_{xi}\boldsymbol{V}_t+\boldsymbol{W}_{hi}\boldsymbol{h}_{t-1}+\boldsymbol{W}_{ci}\boldsymbol{c}_{t-1}+\boldsymbol{b}_i); \quad (1)$$

$$\boldsymbol{f}_t=\sigma(\boldsymbol{W}_{xf}\boldsymbol{X}_t+\boldsymbol{W}_{hf}\boldsymbol{h}_{t-1}+\boldsymbol{W}_{cf}\boldsymbol{c}_{t-1}+\boldsymbol{b}_f); \quad (2)$$

$$\boldsymbol{c}_t=\boldsymbol{f}_t\otimes\boldsymbol{c}_{t-1}+\boldsymbol{i}_t\otimes\tanh(\boldsymbol{W}_{xc}\boldsymbol{X}_t+\boldsymbol{W}_{hc}\boldsymbol{h}_{t-1}+\boldsymbol{b}_c); \quad (3)$$

$$\boldsymbol{o}_t=\sigma(\boldsymbol{W}_{xo}\boldsymbol{X}_t+\boldsymbol{W}_{ho}\boldsymbol{h}_{t-1}+\boldsymbol{W}_{co}\boldsymbol{c}_t+\boldsymbol{b}_o); \quad (4)$$

$$\boldsymbol{h}_t=\boldsymbol{o}_t\otimes\tanh(\boldsymbol{c}_t)。 \quad (5)$$

其中:σ 为 logistic sigmoid 激活函数;$\otimes$为元素的乘积。在 t 时刻,i、f、o 和 c 分别代表输入门、遗忘门、输出门和细胞状态。输入门、输出门和遗忘门由 sigmoid 激活函数实现,细胞状态由 3 个门控制。权重矩阵 W 下标表示每个门之间的连接,b 为偏置。例如,W_{x_i} 为输入节点 X_t 与输入门之间权重矩阵;W_{h_i} 为 $t-1$ 时刻隐藏层状态 h_{t-1} 与输入门之间权重矩阵;W_{c_i} 为 $t-1$ 时刻细胞状态 c_{t-1} 与与输入门之间权重矩阵。

但是,LSTM 只能看到 t 时刻前的历史信息,而不能看到 t 时刻后的将来信息。而 Bi-LSTM 可从全局上下文中学习字符的隐藏特征表示。对于从输入层输出的包含 n 个字符的序列 $(\boldsymbol{X}_1,\boldsymbol{X}_2,\cdots,\boldsymbol{X}_n)$。将 $\overrightarrow{LSTM}$ 代表 LSTM 网络从左到右扫描句子,则 $\overrightarrow{LSTM}$ 学会的隐藏表示可以表示为$[\overrightarrow{h_1},\overrightarrow{h_2},\cdots,\overrightarrow{h_n}]$,类似地,$\overleftarrow{LSTM}$ 代表 LSTM 网络从右到左扫描句子,$[\overleftarrow{h_1},\overleftarrow{h_2},\cdots,\overleftarrow{h_n}]$ 是由$\overleftarrow{LSTM}$ 学习到的隐藏表示。第 i 个字符隐藏表示是通过将拼接其$\overrightarrow{LSTM}$ 和$\overleftarrow{LSTM}$ 下文表示得 $\boldsymbol{h}_i=[\overrightarrow{h_i},\overleftarrow{h_i}]$。Bi-LSTM 层的输出为 $\boldsymbol{h}=[\boldsymbol{h}_1,\boldsymbol{h}_2,\cdots,\boldsymbol{h}_n]$,其中 $\boldsymbol{h}_i\in\mathbf{R}^{2T}$,$T$ 是 LSTM 网络中隐藏特征表示的维数。Bi-LSTM 输出的是每个字符的上下文特征拼接后的向量。

2.3 特征拼接层

特征拼接层将 Bi-LSTM 的输出与前面输入层中的字符嵌入特征向量、位置特征(Seg)、词性特征(Pos)和技能词上下文特征(Con)拼接在一起,以期更好地提升模型的识别准确率。拼接后,第 i 个字符的特征表示为$\boldsymbol{A}_i=[\boldsymbol{h}_i,\boldsymbol{X}_i]$。其中,$\boldsymbol{X}_i$ 为输入层中第 i 个字符的输出,$\boldsymbol{h}_i$ 为 Bi-LSTM 层中第 i 个字符的输出。此外,为了避免简单的线性组合,增强神经网络模型的非线性因素,提出在特征拼接层输出前,将 Bi-LSTM 层与输入层的输出拼接后的向量 $\boldsymbol{A}_i$ 映射成 L 维向量。其中,L 是技能词标签集中标签的数量。即 $\boldsymbol{O}_i=\tanh(w*\boldsymbol{A}_i+b)$,$O_i\in\mathbf{R}^L$。其中,tanh 为激活函数,$w$ 是映射的权重,b 是偏置,将 $\varphi=\{w,b\}$ 记为特征拼接层中的参数集。最终,特征拼接层的输出为 $\boldsymbol{O}=[\boldsymbol{O}_1,\boldsymbol{O}_2,\cdots,\boldsymbol{O}_n]$。

2.4 CRF 层

从特征拼接层输出的向量可以直接用作特征,通过 softmax 函数输出每个字符预测的分类标签。但是,在序列标注任务中,每个输入字符的标签都涉及上下文语义关系,相邻标签通常具有很强的依赖性。而 softmax 计算依据的是字符分类标签的状态概率,未考虑序列本身的全局最优问题,来为每个输入字符输出最可能的标签。例如,用 BIO 定义字符序列的标签集表示,“B”表示该字符是技能词的开头,“I”表示字符位于技能词的中间位置,“O”表示该字符不属于技能词的一部分,“I”标签通常在“B”或“I”之后,但不可能在“O”之后。CRF 层可以约束最终的预测标签以确保它们有效。这些约束可以由 CRF 层在训练过程中自动从训练数据中学习。

用 $\boldsymbol{y}=[\boldsymbol{y}_1,\boldsymbol{y}_2,\cdots,\boldsymbol{y}_n]$ 表示句子 $\boldsymbol{S}$ 的标签序列,其中 $\boldsymbol{y}_i\in\mathbf{R}^L$ 是第 i 个字符标签的 one-hot 表示。CRF 层的输入是特征拼接层的输出,CRF 层的输出是标签序列 y。输入 $\boldsymbol{O}$ 的标签序列 y 的条件概率计算如下[25]:

$$p(y\mid\boldsymbol{O},\theta)=\frac{\prod_{i=1}^{n}\Psi(\boldsymbol{O}_i,y_i,y_{i-1})}{\sum_{y'\in Y(S)}\prod_{i=1}^{n}\Psi(\boldsymbol{O}_i,y'_i,y'_{i-1})}。 \quad (6)$$

其中:$Y(S)$ 为句子 s 的所有可能标签序列的集合;$\Psi(\boldsymbol{O}_i,\boldsymbol{y}_i,\boldsymbol{y}_{i-1})$ 为势函数,

$$\Psi(\boldsymbol{O}_i,\boldsymbol{y}_i,\boldsymbol{y}_{i-1})=\mathrm{e}^{(\boldsymbol{y}_i^{\mathrm{T}}\boldsymbol{E}^{\mathrm{T}}\boldsymbol{O}_i+\boldsymbol{y}_{i-1}^{\mathrm{T}}\boldsymbol{F}_{y_i})}。 \quad (7)$$

其中:$\boldsymbol{E}\in\mathbf{R}^{L\times L}$,$\boldsymbol{F}\in\mathbf{R}^{L\times L}$ 为 CRF 层的参数;θ 为参数集,即 $\theta=\{\boldsymbol{E},\boldsymbol{F}\}$。CRF 层的损失函数:

$$f_{\mathrm{Loss}}=-\sum_{S\in\mathrm{corpus}}\log(p(y_S\mid\boldsymbol{O}_S,\theta)), \quad (8)$$

其中 corpus 是训练数据集中的所有语句。

2.5 训练过程

本研究使用学习率为 0.001 的小批量自适应矩估计(Adam)优化算法,以端到端的方式训练技能词抽取模型。在训练期间,使用每批次 20 条语句训练 100 个周期。换句话说,在每个周期中,语料库中

1 006 条语料被随机分为 51 个批次进行训练，每个批次不超过 20 个句子。对于每一批次，采用预先训练的字符嵌入而非随机初始化的嵌入作为输入字符的嵌入特征向量。在 CRF 层中，使用动态编程来计算式(6)的结果并预测标签序列。最后，反向传播时根据 CRF 层预测出的标签序列与真实标签序列之间的误差，依次更新 CRF 层中的参数集 θ、特征拼接层中的参数集 φ 以及 Bi-LSTM 层中的所有的权重矩阵和网络参数并保存模型。

3 实验分析

为了验证提出的技能词抽取模型的有效性，将本模型与主流方法进行比较。在实验中详细描述数据集、标注策略、预训练的字符向量、模型参数设置、结果分析。

3.1 数据集与评价指标

智联招聘是国内最大的在线招聘网站之一。它拥有 400 万个合作公司，每天在各个领域发布大量的招聘职位。为了有效地利用这些信息，设计了一种具有布隆过滤器的计算机自动网络爬虫程序，系统地检索招聘网页并将其存储在数据库中，然后从检索的网页中提取特定信息。提取的信息包括职位名称、发布时间、岗位职责要求、职位链接、职位类别、招聘人数、学历要求、经验要求、薪资和福利待遇、工作地点、公司所在地和招聘公司名称。本研究于 2017 年 10 月开始收集数据，一直持续至今。截至 2018 年 12 月 31 日，已获得超过 1 300 万条无重复的岗位招聘数据。其中，获得 IT 行业类别数据 1 364 874 条。收集的数据涵盖了中国 336 个城市、13 个行业类别和 930 个职位类别。

图 2 为网络招聘文本示例，其中“C＋＋”“Linux”“推荐系统”等为岗位所要求的专业知识或专业能力。在此实验中，选择 IT 行业类别的岗位招聘数据作为语料库。将每条职位招聘数据中的“岗位职责要求”视为一条语句。由于时间和手工标注成本的限制，仅标注了 1 006 条“岗位职责要求”作为实验语料。在标注语料时，尝试选择句法结构标准的句子作为研究语料，并将语句中的专业课程名称、专业知识点和相关专业工具标记为技能词，例如“C 语言程序设计”“堆排序”“SpringMVC 框架”等。因此，一条语句将被标记为：“熟/ O 悉/ O 机/ B-SKILL 器/ I-SKILL 学/ I-SKILL 习/ I-SKILL 与/ O 自/ B-SKILL 然/ I-SKILL 语/ I-SKILL 言/ I-SKILL 处/ I-SKILL 理/ I-SKILL。/O”。

任职要求：1.熟练运用C++编程，具有良好的编程习惯。2.有较强的进行代码调试和解决技术问题的能力。3.对于主流Deep Learning框架有了解。对于主流Deep Learining框架的内部框架有深入了解的优先。4、熟悉Linux开发环境，熟悉Python/C++语言；5、熟悉自然语言处理常见算法与模型（语言模型、MaxEnt/CRF, pLSA/LDA, w2v, CNN/RNN等）；6、参与过NLP项目（如中文分词、文本分类、文本聚类）；7、对推荐系统、大数据挖掘， deep learning等方向有浓厚的兴趣，有很强的自学能力的优先；8、计算机科学、机器学习、人工智能等专业职位工作经验优先；9、3-5年实际项目开发经验；10、有较强的团队精神和沟通交流能力。11、热爱运动喜欢跑步。

图 2 网络招聘文本示例

由于没有明确的标准如何将数据集划分为训练/验证/测试集，在实验中将数据集进行两轮交叉验证。首先将整个数据集划分成 5 份，选择其中的 80％作为训练数据，而将其余 20％用作测试数据。然后，再次将训练数据划分成 10 份，选择其中的 90％作为最终训练数据，并将其余的 10％用作验证数据。换句话说，将全部数据的 72％作为训练数据，将 8％作为验证数据，将 20％作为测试数据。每次实验的训练周期为 100 个周期，通过验证集找出训练集在这 100 个周期内最佳网络模型参数后再使用测试集进行测试，以获得本次验证的结果。表 3 中分别展示了用于训练、验证和测试数据集的句子数、技能词数量。值得注意的是，由于每个句子中包含的技能词数量不同，训练/验证/测试数据集中的技能词数量会随着每次划分而动态变化，因此，技能词数量的范围也在表中。

实验采用机器学习中常用的准确率(precision)、召回率(recall)指标评价技能词抽取模型的性能，并采用 F1 值指标评价其综合性能。

表 3 数据集的统计信息

数据集	数据类型	技能词数量	语句数量
招聘数据	训练集	[5 836,6 586]	725
	验证集	[593,817]	80
	测试集	[1 577,1 903]	201

3.2　模型参数设置

为了考虑批量训练样本量大小(batch)和学习率对本模型的影响,通过实验进行超参数选择。首先,根据已有的参考文献中批量训练样本量大小和学习率的使用,确定出 batch 大小和学习率的选择范围,分别为[20、50、100]和[0.001、0.002]。其次,在交叉验证的实验方案基础上,尝试使用不同批量训练样本量大小(batch)和学习率的参数组合,结果如图 3 所示。

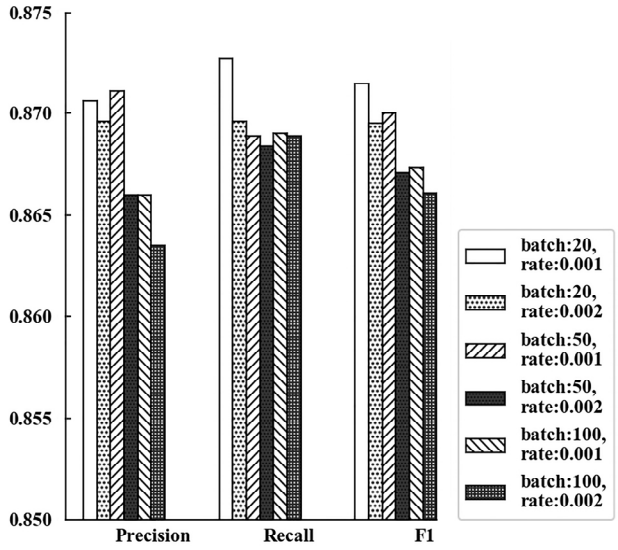

图 3　采用不同批次大小和学习率时模型的准确率、召回率、F1 值

从图 3 可看出,当批量训练样本量大小设置为 20,学习率设置为 0.001 时,该模型可获得最佳总体效果。因此可认为,最合适的批次大小为 20,最佳学习率为 0.001。

模型的其他超参数的初始值设置:字符嵌入特征向量设置为 100。本研究使用的是 word2vec 中的 CBOW 模型对中文维基百科语料预训练的字符向量[26],epoch 次数固定为 100,隐藏层维度固定为 100,其他网络参数包括特征拼接层中的参数集 φ 和 Bi-LSTM 层中所有权重矩阵的初始值,以及 CRF 层的参数集 θ 在[−1,1]范围内随机地均匀初始化。

3.3　实验结果与分析

实验 1　为了验证提出的技能词抽取模型中所加入的各类语料特征对技能词抽取有效性的影响,进行了本组实验。具体实验设置如下:选择基于字符级的 Bi-LSTM-CRF 模型作为基线对比实验,此时只将字符嵌入特征输入网络,并不输入字符的语料特征。然后,分别在 Bi-LSTM-CRF 模型的输入层中加入不同类型的语料特征,如 Model_1 代表在 Bi-LSTM-CRF 的输入层中加入字符的位置特征(seg)、但并未将输出层的输出与 Bi-LSTM 层的输出进行拼接,Model_4 代表在 Model_1 基础上加入技能词的上下文特征(con),即在 Bi-LSTM-CRF 的输入层中加入所有语料特征。Model_8 代表在 Bi-LSTM-CRF 的输入层中加入所有语料特征,并且将输出层的输出与 Bi-LSTM 层的输出进行拼接,即最终的技能词抽取模型。

从表 4 可看出,在 Bi-LSTM-CRF 模型的输入层中分别加入位置特征(seg)、词性特征(pos)和技能词的上下文特征(con)时,相比于 Bi-LSTM-CRF 模型的 F1 值分别提升了 0.44%、0.35%和 7.66%。其中,加入上下文特征获得的提升最大,这是因为:由于招聘语料的句法结构比较单一,技能词的上下文特征较为固定,充分挖掘技能词上下文特征能较好地反映技能词在语料中出现的位置,能有效地抽取出"……具备数据库开发能力……""……常用的 Java、C、python 等编程语言……"这类具有明显句法结构的技能词,从而使训练出的模型更具有泛化能力。而字符的位置特征(seg)和词性特征(pos)的加入,虽然提高了 F1 值,但提升的幅度不是很大。其中可能的原因是:词语的位置特征(seg)是通过对句子进行 jieba 分词,再提取每个字符与所在词语的相对位置而得到的。中文分词结果的不准确性影响了字符位置特征的提取,从而对技能词的抽取带来了一定程度的干扰。对于词性特征(pos),则可能是因为 jieba 分词无法标注出英文字符的词性,以及技能词的构成在词法特征上规律性不强。

表 4　各种语料特征加入后技能词抽取的性能

模型编号	模型特征	Precision	Recall	F1	提升
Model_0	Bi-LSTM-CRF	0.789 0	0.789 9	0.789 2	—
Model_1	Model_0+Seg	0.791 9	0.795 8	0.793 6	+0.44%
Model_2	Model_0+Pos	0.790 5	0.795 1	0.792 7	+0.35%
Model_3	Model_0+Con	0.866 2	0.865 5	0.865 8	+7.66%
Model_4	Model_0+Con	0.867 6	0.867 5	0.867 5	+7.83%
Model_5	Model_0+Seg +Con	0.867 6	0.866 6	0.867 0	+7.78%
Model_6	Model_0+Seg +Pos	0.791 4	0.797 3	0.794 2	+0.5%
Model_7	Model_0+Seg +Pos +Con	0.871 6	0.869 8	0.870 6	+8.14%
Model_8	本技能词抽取模型	0.870 6	0.8727	0.871 5	+8.23%

从表 4 中还可看出：如果同时将字符的位置特征(seg)和词性特征(pos)加入到 Bi-LSTM-CRF 模型的输入层中，相比于 Bi-LSTM-CRF，模型的 F1 值提升了 0.5%；如果同时将字符的位置特征(Seg)和技能词的上下文特征(con)加入到 Bi-LSTM-CRF 模型的输入层中，相比于 Bi-LSTM-CRF，模型的 F1 值提升了 7.83%；如果同时将词性特征(pos)和技能词的上下文特征(con)加入到 Bi-LSTM-CRF 模型的输入层中，相比于 Bi-LSTM-CRF，模型的 F1 值提升了 7.78%；如果同时将字符的位置特征(Seg)、词性特征(Pos)和技能词的上下文特征(Con)都加入到 Bi-LSTM-CRF 模型的输入层中，模型的 F1 值能得到更进一步的提升，从 0.789 2 提高到了 0.870 6，提升幅度达到 8.14%。因此，在模型中加入的语料特征越多，越有利于模型的技能词抽取。

另外，在 Bi-LSTM-CRF 的输入层中加入所有语料特征的同时，将输入层的输出和 Bi-LSTM 层的输出拼接，即为最终提出的技能词抽取模型。相比于只加入语料特征的情况，又进一步提升了模型的 F1 值，提升幅度达 8.23%。最终可得出结论，本模型能够有效进行技能词抽取，并且抽取性能得到了极大提高，加入各类语料特征也有利于技能词抽取性能的提升。

实验 2 为了验证本模型中所加入的各类语料特征，在不同规模训练集下也是否都有利于模型抽取性能的提升，以及评估所加入的丰富的语料特征是否能够缓解模型对大量标注数据需求的依赖，在实验 1 的基础上，进一步从训练集中抽取 25%、50%和 75%的样本，同时保持测试集不变，进行实验，实验结果如表 5 所示。

样本抽取的具体方法如下：在每次使用两轮交叉验证方法实现对数据的划分后，再将训练数据平均分成 N 份，每次选择其中若干份作为最终训练集，并重复进行多次实验。25%训练集的抽取方案为：将训练集划分为 4 份，每次选择 1 份作为最终训练集，实验重复 4 次；50%的训练集抽取方案为：将训练集划分成 2 份，每次选择 1 份作为最终训练集，实验重复 2 次；75%的训练集抽取方案为：将训练集划分成 4 份，每次选择 3 份作为最终训练集，实验重复 4 次。本组实验在不同训练集比例下，同样也是选择 Bi-LSTM-CRF 模型作为基线对比实验。

表 5 不同训练集比例下加入各类语料特征后的技能词抽取性能

模型	训练集比例		
	25%	50%	75%
	Precision，Recall，F1，提升	Precision，Recall，F1，提升	Precision，Recall，F1，提升
Model_0	0.714 6，0.740 2，0.726 7，—	0.754 9，0.775 0，0.764 6，—	0.776 9，0.784 8，0.780 7，—
Model_1	0.723 4，0.751 2，0.736 7，+1.00%	0.760 1，0.778 3，0.768 8，+0.42%	0.784 1，0.788 2，0.785 9，+0.52%
Model_2	0.719 5，0.750 0，0.734 1，+0.74%	0.757 3，0.776 1，0.766 4，+0.18%	0.783 1，0.788 3，0.785 5，+0.48%
Model_3	0.812 4，0.824 4，0.817 2，+9.05%	0.841 3，0.845 6，0.843 3，+7.87%	0.851 1，0.856 2，0.853 5，+7.28%
Model_4	0.815 5，0.829 3，0.822 1，+9.54%	0.845 1，0.852 9，0.848 9，+8.43%	0.861 6，0.855 1，0.858 3，+7.76%
Model_5	0.815 8，0.822 7，0.819 1，+9.24%	0.844 3，0.848 1，0.846 1，+8.15%	0.856 3，0.857 3，0.856 7，+7.60%
Model_6	0.724 7，0.756 3，0.739 9，+1.32%	0.761 0，0.777 5，0.769 0，+0.44%	0.781 8，0.789 1，0.785 3，+0.46%
Model_7	0.819 5，0.834 1，0.826 7，+10.0%	0.845 9，0.852 5，0.849 1，+8.45%	0.863 5，0.861 3，0.862 3，+8.16%
Model_8	0.829 1，0.838 2，0.8336，+10.7%	0.849 8，0.853 7，0.8517，+8.71%	0.864 4，0.864 8，0.8646，+8.39%

如表 5 所示，在不同规模的训练集下，本模型抽取性能相比于 Bi-LSTM-CRF 模型仍有较大提升，在 25%、50%和 75%训练集比例下，F1 值分别由 0.726 7、0.764 6 和 0.780 7 提高到了 0.833 6、0.851 7 和 0.864 6。另外，在不同比例的训练集下，加入的各类型语料特征也依然有利于模型抽取性能的提升，并且可以得出与实验 1 中使用全部训练集同样的结论，即在模型中加入的语料特征越多，越有利于模型的技能词抽取。例如：在 25%、50%和 75%训练集比例下，仅加入词性特征(Pos)，相比于 Bi-LSTM-CRF 模型的 F1 值分别提升了 0.74%、0.18%和 0.48%。若同时将字符的位置特征(Seg)和词性特征(Pos)加入 Bi-LSTM-CRF 模型的输入层，相比于 Bi-LSTM-CRF 模型的 F1 值分别提升了 1.32%、0.44%、0.46%。而同时将字符的位置特征(Seg)、词性特征(Pos)和技能词的上下文特征(Con)都加入 Bi-LSTM-CRF 模型的输入层，模型的 F1 值又能得到更进一步的提升，分别从 0.726 7、0.764 6 和 0.780 7 提高到了 0.826 7、0.849 1 和 0.862 3，提升幅度分别达到 10.0%、8.45%和 8.16%。

从表 5 还可看出：在 Bi-LSTM-CRF 模型的输入层加入丰富的语料特征确实减轻了可用标注数据的不足。例如，训练集比例为 75%时，Bi-LSTM-CRF 模型的输入层仅使用字符嵌入特征情况下的 F1 值为 78.07%，而训练集比例为 50%时，在 Bi-LSTM-CRF 模型的输入层加入位置特征(Seg)和词性特征

(Pos)情况下的 F1 值为 76.90%;训练集比例为 100%时,Bi-LSTM-CRF 模型的输入层仅使用字符嵌入特征情况下的 F1 值为 78.92%,而训练集比例仅为 25%时,在 Bi-LSTM-CRF 模型的输入层加入技能词的上下文特征(Con)情况下的 F1 值便可达 81.72%。因此,可得出结论,加入丰富的语料特征本模型能够缓解模型对大量标注数据的依赖。

实验 3　为了说明提出的技能词抽取模型的有效性,选取了目前主流的序列标注模型 BERT-Bi-LSTM-CRF 和 IDCNN-CRF 模型进行对比。虽然本模型所选择的对比方法在英文数据集上进行实验取得不错的效果,但该类型框架具有通用性,受语言差异影响较小,并且在进行实验时采用同样的数据处理方式。

方法 1　BERT-Bi-LSTM-CRF:BERT(bidirectional encoder representations from transformers)是一种由 Devlin 等[27]等提出的以 Transformers 为主要框架的双向编码表征模型。BERT-Bi-LSTM-CRF 是在预训练的 BERT 模型的顶部添加了用于序列标记的 Bi-LSTM 层和 CRF 层,并且通过招聘语料数据对预训练的 BERT 模型的参数进行调整。

方法 2　IDCNN-CRF:IDCNN-CRF(iterated dilated convolutional neural network-conditional random field)模型是由 Emma 等[28]提出的,类似于 Bi-LSTM-CRF 模型,采用深度学习模型进行特征提取,再放入 CRF 层解码出标注结果。但不同于 Bi-LSTM 网络,该模型用 4 个结构相同的膨胀卷积(DCNN)提取语句特征,称之为 IDCNN。

数据集的划分同样采用交叉验证的实验方案。同样选择 Bi-LSTM-CRF 模型作为基线对比模型,实验结果如表 6 所示。方法 1 中 BERT-Bi-LSTM-CRF 模型的超参数设置如下,初始学习率为 0.001、epoch 次数为 50、隐藏层维度为 100 和批量训练样本量为 32。方法 2 中 IDCNN-CRF 模型的超参数设置如下,字符嵌入维度为 100、初始学习率为 0.001、epoch 次数为 100、卷积核大小为 1*3 和批量训练样本量为 20。

表 6　不同网络模型抽取性能对比

模型	Pre	Recall	F1	提升
Bi-LSTM-CRF	0.789 0	0.789 9	0.789 2	—
BERT-Bi-LSTM-CRF	0.815 6	0.818 4	0.817 0	+2.78%
IDCNN-CRF	0.804 8	0.796 7	0.800 9	+1.17%
技能词抽取模型	0.870 6	0.872 7	0.871 5	+8.23%

从表 6 可看出,本技能词抽取模型的 F1 值远优于方法 1 的 BERT-Bi-LSTM-CRF 模型和方法 2 的 IDCNN-CRF 模型的 F1 值。此外,考虑到 BERT 模型的训练使用了非常庞大的公共资源训练语料库,而本技能词抽取模型的训练仅使用少部分人工标注的语料。因此,可得出结论,本技能词抽取模型可有效地抽取技能词,并且加入的语料特征更有利于模型的抽取性能提升。

4　结束语

技能词是分析大规模招聘数据与劳动力市场供求关系的基础。提出了一种基于深度学习的技能词抽取方法,实验结果表明,充分挖掘招聘语料的内部特征可以提高技能词抽取的准确性。实验结果还表明,引入丰富的语料特征能够缓解模型对大量标注数据需求的依赖问题。目前,通过预训练的字符嵌入作为输入字符的形式化表示,但字符嵌入的预训练受训练语料的语义影响,未来的工作包括如何利用动态词嵌入来更好地提取技能词,以及如何利用大量未标记的语料库或迁移学习方法来减少对标注数据的依赖性。

参考文献:

[1] KIM J Y, LEE C K. An empirical analysis of requirements for data scientists using online job postings[J]. International Journal of Software Engineering and its Application, 2016, 10(4): 161-172.

[2] DE MAURO A, GRECO M, GRIMALDI M, et al. Human resources for big data professions: a systematic classification of job roles and required skill sets[J]. Information Processing and Management, 2018, 54(5): 807-817.

[3] 黄崑, 王凯飞, 王珊珊, 等. 数据类岗位招聘需求调查及对图情学科人才培养的启示[J]. 图书情报知识, 2016, 6(1): 42-53.

[4] 赵京胜, 朱巧明, 周国栋, 等. 自动关键词抽取研究综述[J]. 软件学报, 2017, 28(9): 2431-2449.

[5] 冯鸾鸾, 李军辉, 李培峰, 等. 面向国防科技领域的技术和术语识别方法研究[J]. 计算机科学, 2019, 46(12): 231-236.

[6] 陈锋, 翟羽佳, 王芳. 基于条件随机场的学术期刊中理论的自动识别方法[J]. 图书情报工作, 2016, 60(2): 122-128.

[7] QUIMBAYA A P, MúNERA A S, Rivera R A G, et al. Named entity recognition over electronic health records through a combined dictionary-based approach[J]. Procedia Computer Science, 2016, 100(100): 55-61.

[8] CHEN Y J, ZHOU C L, SHI X D. Automatic extraction of Chinese terms[C]//International Conference on Natural Language Processing and Knowledge Engineering. Washington, USA: IEEE, 2005: 281-286.

[9] ZHANG S, ELHADAD N. Unsupervised biomedical named entity recognition: Experiments with clinical and biological texts[J]. Journal of Biomedical Informatics, 2013, 46(6): 1088-1098.

[10] PANTEL P, LIN D. A statistical corpus-based term extractor[C]//14th Biennial Conference of the Canadian Society for Computational Studies of Intelligence. Berlin, German: Springer, 2001: 36-46.

[11] ZHANG J, SHEN D, ZHOU G, et al. Enhancing HMM-based biomedical named entity recognition by studying special phenomena[J]. Journal of Biomedical Informatics, 2004, 37(6): 411-422.

[12] 蒋婷, 孙建军. 基于 SVR 模型的中文领域术语自动抽取研究: 面向图书情报领域[J]. 情报理论与实践, 2016, 39(1): 24-31.

[13] LI L S, MAO T T, HUANG D G, et al. Hybrid models for Chinese named entity recognition[C]//Proc of the Fifth SIGHAN Workshop on Chinese Language Processing. Stroudsburg, USA: ACL, 2006: 72-78.

[14] 王昊, 王密平, 苏新宁. 面向本体学习的中文专利术语抽取研究[J]. 情报学报, 2016, 35(6): 573-585.

[15] DUAN H Z, ZHENG Y. A study on features of the CRFs-based Chinese named entity recognition[J]. International Journal of Advanced Intelligence, 2011, 3(2): 287-294.

[16] 何宇, 吕学强, 徐丽萍. 新能源汽车领域中文术语抽取方法[J]. 现代图书情报技术, 2015, 31(10): 88-94.

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

[18] LAMPLE G, BALLESTEROS M, SUBRAMANIAN S, et al. Neural architectures for named entity recognition[C]//Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, USA: ACL, 2016: 260-270.

[19] HUANG Z H, XU W, YU K. Bidirectional LSTM-CRF Models for Sequence Tagging[EB/OL]. de. arxiv. org/pdf/1508. 01991.

[20] MA X, HOVY E. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF[C]//54th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, USA: ACL, 2016.

[21] 闫兴龙, 刘奕群, 方奇. 基于网络资源与用户行为信息的领域术语提取[J]. 软件学报, 2013, 24(9): 113-124.

[22] 赵东玥, 杜永萍, 石崇德. 基于 BLSTM 的科技文献术语抽取方法[J]. 情报工程, 2018, 4(1): 67-74.

[23] 赵洪, 王芳. 理论术语抽取的深度学习模型及自训练算法研究[J]. 情报学报, 2018, 37(9): 67-82.

[24] 俞琰, 陈磊, 姜金德. 网络招聘文本技能信息自动抽取研究[J]. 图书情报工作, 2019, 63(13): 105-113.

[25] WU F, LIU J, WU C, et al. Neural Chinese named entity recognition via CNN-LSTM-CRF and joint training with word segmentation[C]//Conference of the World Wide Web. New York, USA: ACM, 2019: 3342-3348.

[26] DONG C H, ZHANG J, ZONG C Q, et al. Character-based LSTM-CRF with radical-level features for Chinese named entity recognition[C]//5th CCF Conference on Natural Language Processing and Chinese Computing and the 24th International Conference on Computer Processing of Oriental Languages. Berlin, German: Springer, 2016: 239-250.

[27] DEVLIN J, CHANG M, LEE K, et al. Bert: pre-training of deep bidirectional transformers for language understanding[C]//Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, USA: ACL, 2019: 4171-4186.

[28] EMMA S, PATRICK V, DAVID B, et al. Fast and accurate entity recognition with iterated dilated convolutions[C]//Conference on Empirical Methods in Natural Language Processing. Stroudsburg, USA: ACL, 2017: 2670-2680.

编辑: 梁王欢