文章编号：2095-2163(2023)04-0047-06 中图分类号：TP391.1 文献标志码：A

基于深度学习的端到端人岗匹配模型

朱 瑜，魏嘉银，卢友军，王 琳，江 漫
（贵州民族大学 数据科学与信息工程学院，贵阳 550025）

摘 要：针对现有人岗匹配推荐算法主要采用人工评估求职者与职位的匹配度，存在招聘速度慢、成本高且易受主观判断所误导等问题，提出一种基于深度学习的端到端人岗匹配模型 BATPJF。首先，运用 TextCNN 提取简历和职位描述数据的局部特征。同时，运用 BiLSTM 提取简历和职位描述文本数据的上下文特征，再将 BiLSTM 隐藏层产生的特征作为 Attention 层的输入，利用注意力机制对 BiLSTM 层提取的特征采用加权的方式体现不同的经历和能力对岗位能力需求重要程度的影响。然后，将 2 种模型提取到的特征进行融合。最后，通过全连接层进行预测。实验结果表明，与其他 5 种人岗匹配模型对比，本文提出的模型可以更有效地匹配工作要求和简历文本信息。

关键词：人岗匹配；注意力机制；招聘分析

End to end person-job matching model based on deep learning

ZHU Yu, WEI Jiayin, LU Youjun, WANG Lin, JIANG Man
(School of Data Science and Information Engineering, Guizhou Minzu University, Guiyang 550025, China)

【Abstract】Aiming at the problems of low recruitment efficiency, high cost and being easily misled by subjective judgment, the existing person position matching recommendation algorithm mainly uses manual evaluation of the matching degree between job seekers and positions, and proposes an end-to-end person-job matching model BATPJF based on deep learning. First of all, the paper uses TextCNN to extract the local features of the resume and job description data. At the same time, BiLSTM is used to extract the contextual features of resume and job description text data, and then the features generated by the BiLSTM hidden layer are used as the input of the Attention layer. The attention mechanism is used to weigh the features extracted by the BiLSTM layer to reflect the impact of different experiences and abilities on the importance of job competency requirements. Secondly, the features extracted from the two models are fused. Finally, the prediction is conducted through the complete connection layer. The experimental results show that the model proposed in this paper can match job requirements and resume text information more effectively than the other five person-job matching models.

【Key words】person-job matching; attention mechanism; recruitment analysis

0 引 言

为了找到满足岗位需求的人才，传统方法是由招聘人员手动审查求职者的简历，以决定是否提供面试机会。然而，面对海量的简历，招聘人员不得不花费大量的时间和精力筛选简历，优中选优以便能够找到合适的求职者。传统的简历审查方式存在招聘速度慢、成本高等问题。因此，如何从简历中挖掘出求职者自身的价值并将其与已有的职位相匹配成为一个亟待解决的问题，这个问题则称为人岗匹配问题。

职位推荐作为人才招聘中的一项重要任务，已经有许多学者对其进行研究。早期研究者根据用户的学历以及用户在每个职位上的点击、浏览时间等交互信息，采用协同过滤等推荐算法向用户推荐职位[1]。早期方法忽视了工作和简历文档的文本语

基金项目：贵州省科技计划项目(黔科合基础[2018]1082，黔科合基础[2019]1159)；贵州省教育厅自然科学研究项目(黔教技[2022]015号)；贵州省教育厅自然科学研究项目(黔教技[2022]047 号)。

作者简介：朱 瑜(1998-)，女，硕士研究生，主要研究方向：推荐算法；魏嘉银(1986-)，男，博士，副教授，主要研究方向：推荐算法、自然语言处理；卢友军(1985-)，男，博士，副教授，主要研究方向：复杂网络；王 琳(1998-)，女，硕士研究生，主要研究方向：推荐算法；江漫(1997-)，女，硕士研究生，主要研究方向：计算机视觉。

通讯作者：魏嘉银 Email：weijiayin05@ sina.com

收稿日期：2022-11-09

义信息,因此,为了充分利用简历和职位要求中丰富的文本语义信息,大多数研究将人岗匹配任务视为文本匹配任务,就是将工作描述和简历内容表示为维数相同的隐藏向量,然后计算 2 个向量的匹配得分,并据此预测简历与职位的匹配程度[2]。

CNN 作为近年来最流行的深度学习算法之一,已被广泛应用于人岗匹配领域。Nasser 等学者[3]将简历分为不同的类别,并提出了一个 CNN 模型,将简历与工作配对。Zhu 等学者[4]提出了一个 PJFNN 模型,将简历和职位描述中嵌入的每个词分别用 2 个 CNN 模型进行建模,并利用简历和职位之间的余弦相似度计算匹配分数。Khatua 等学者[5]使用 Twitter 中的双卷积网络来匹配招聘人员和求职者。虽然 CNN 模型提取局部特征效果较好,但是容易忽略单词之间的顺序和关系,导致语义特征提取不够准确[6]。LSTM 可以更有效地处理文本信息,更高效地挖掘文本潜在的语义信息,缓解梯度爆炸问题。于是,Zhou 等学者[7]将 LSTM 应用于文本分类领域,提高了文本分类的准确度。Qin 等学者[8]将分层 RNN 模型应用于工作文档,提出了一种基于分层能力感知注意力机制的循环神经网络结构来学习文本的语义表示。Jiang 等学者[9]通过 LSTM 模型学习求职者和招聘人员的隐含意图,结合语义生成求职者和招聘人员的有效表示。为了充分发挥 CNN 与 RNN 提取特征时的优势,许多研究者将 CNN 与 LSTM 结合使用,以便提高模型提取特征的能力。如:李超凡等学者[10]提出了一种基于注意力机制结合 CNN-BiLSTM 的文本分类模型,解决了中文电子病历文本高度稀疏且分类效果不佳的问题。吉兴全等学者[11]使用 CNN 与 LSTM 对短期电价进行预测,提高了电价预测的精度及预测效率。任建吉等学者[12]针对电网数据具有非线性和时序性的特点,将 CNN 与 BiLSTM 结合提取数据本身的时空特征,提高了模型的预测精度。以上模型虽然提升了模型提取特征的效果,但大多采用递进式网络结构,导致提取到的信息向后传递时容易发生梯度消失或梯度爆炸的问题,同时递进式网络结构提取文本特征时只用到单一网络的优势,无法融合 CNN 和 RNN 提取文本信息的优势,因此最终效果有待提升。

为了提高人岗匹配的效果,本文提出一种端到端的人岗匹配模型 BATPJF,该模型采用并列式网络结构,充分发挥了 CNN 提取局部特征的优势与 BiLSTM 记忆功能的优势,有效改善了模型的整体结构,提升了人岗匹配的效果。

1 模型构建

1.1 问题定义

令 $\boldsymbol{job}_i = \{\boldsymbol{job}_{i,1}, \boldsymbol{job}_{i,2}, \cdots, \boldsymbol{job}_{i,p}\}$ 为一条岗位招聘信息,其中 $\boldsymbol{job}_{i,j}(j \in [1,p])$ 为具体的岗位需求或职责,令 $\boldsymbol{job}_{i,j} = \{job^1_{i,j}, job^2_{i,j}, \cdots, job^s_{i,j}\}$ 表示岗位 $\boldsymbol{job}_{i,j}$ 的信息中包含的 s 个词。又令 $\boldsymbol{r}_i = \{\boldsymbol{r}_{i,1}, \boldsymbol{r}_{i,2}, \cdots, \boldsymbol{r}_{i,q}\}$ 表示一份简历,其中 $\boldsymbol{r}_{i,j}(j \in [1,q])$ 表示该简历包含的具体工作经历,再令 $\boldsymbol{r}_{i,j} = \{r^1_{i,j}, r^2_{i,j}, \cdots, r^u_{i,j}\}$ 表示工作经历 $\boldsymbol{r}_{i,j}$ 的信息中包含的 u 个词。给定一组数据 $\boldsymbol{S} = \langle \boldsymbol{J}, \boldsymbol{R}, \boldsymbol{Y} \rangle$,其中,$\boldsymbol{J} = \{\boldsymbol{job}_1, \boldsymbol{job}_2, \cdots, \boldsymbol{job}_m\}$ 为招聘信息,$\boldsymbol{R} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \cdots, \boldsymbol{r}_n\}$ 为简历信息,$\boldsymbol{Y}$ 为招聘结果标签。本文的目标是训练一个匹配函数 M,并根据 M 来快速精准地预测一份简历 $\boldsymbol{r}_i$ 与 $\boldsymbol{job}_j$ 之间的匹配结果。由于人岗匹配问题很难直接得到一个绝对的匹配分数,且很可能会导致过拟合和模型偏差[13],因此采用 Top-K 的方法优化排名。

1.2 模型概述

本文提出的模型 BATPJF 主要由 TextCNN 网络、BiLSTM-Attention 网络、融合层和匹配预测层构成。模型的总体框架如图 1 所示。

考虑到职位描述和简历中包含大量描述职位要求或求职者经历的词语,而 TextCNN 在捕捉文本数据的层次关系和局部语义方面效果较好,因此本文运用 TextCNN 模型用于提取数据中的关键性词语。卷积层与池化层的交替结构对于数据特征的提取是有效的,因此在 TextCNN 中,使用卷积层和最大池化层以交错堆叠的方式自动提取职位描述和简历文本数据的局部特征,并将所得特征向量输出。

TextCNN 模型通过卷积核提取输入文本的局部特征,但是滤波器的大小限制了模型学习文本数据前后的依赖关系。所以本文使用 BiLSTM-Attenion 模型用于提取文本上下文依赖关系并分别对不同的词和句子分配相应的权重。具体如下:首先使用 BiLSTM 分别获取简历和职位描述的词级别的文本表征,然后将文本表征作为注意力机制层的输入来预估每个单词的重要性,接着根据每个能力要求的隐藏状态和所有能力要求的上下文向量之间的相似性计算出每个能力要求的重要性并输出。

最后,将 TextCNN 模型捕捉到的局部特征,以及经过并行的 BiLSTM-Attenion 模型提取到的上下文特征进行特征拼接后,输入到全连接层,并经由 *Softmax* 层输出结果。

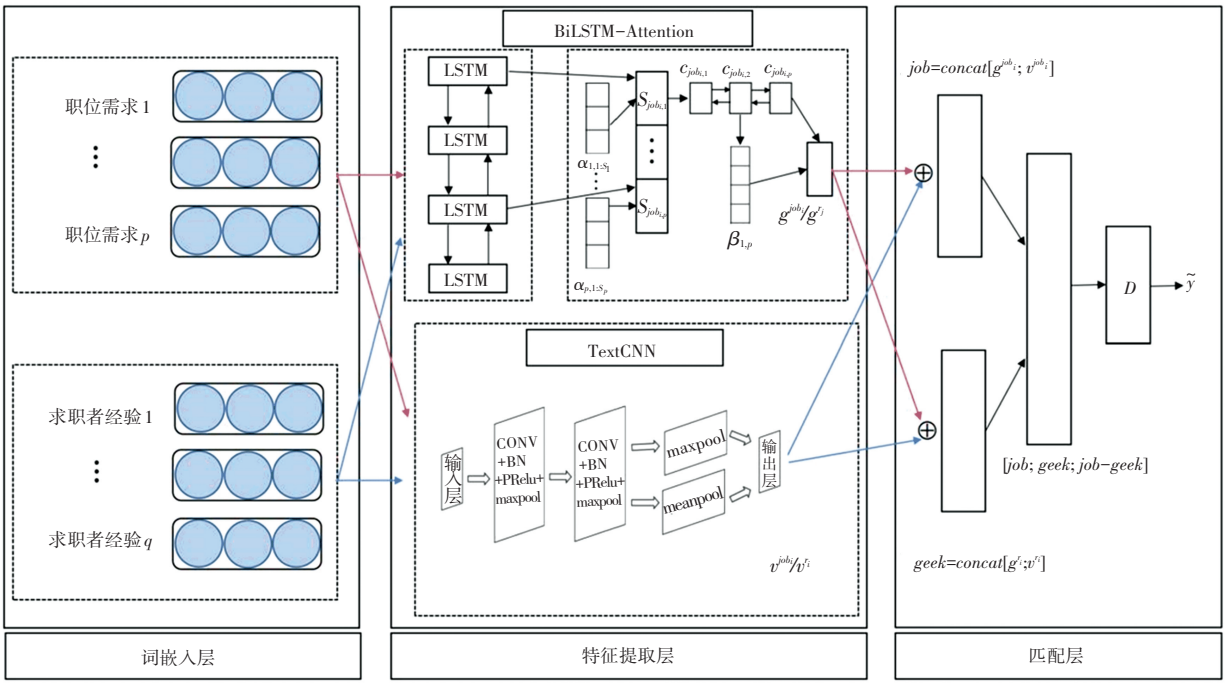

图 1　BATPJF 模型图

Fig. 1　Structure diagram of BATPJF

1.3　TextCNN 层

TextCNN 主要由输入层、卷积层、归一化层和池化层构成。本文以岗位描述部分为例进行说明。TextCNN 层模型如图 2 所示。

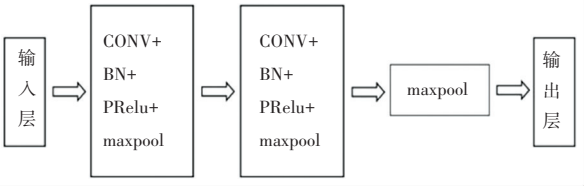

图 2　TextCNN 层

Fig. 2　TextCNN layer

对于 $\boldsymbol{job}_{i,j}$ 中的第 l 条需求中的第 t 个词的 d_0 维词向量 $\boldsymbol{w}_l^t \in R^{d_0}$，于是第 l 项需求对应的矩阵可表示为 $\boldsymbol{v}_{job_{i,l}} = [\boldsymbol{w}_l^1, \boldsymbol{w}_l^2, \cdots, \boldsymbol{w}_l^s]$，$\boldsymbol{v}_{job_{i,l}} \in R^{d\times s}$。首先，使用卷积层提取第 l 项需求的文本特征，然后对卷积层的输出应用批归一化处理以降低训练成本，接着运用 *Prelu* 激活函数对输出值做非线性变换操作，最后使用池化层压缩提取到的特征以减少模型的计算量，同时增加模型识别特征的抗干扰能力。在此基础上，将所有需求项的向量通过最大池化层投影到一个向量上：$\boldsymbol{v}^{job_i} = [\max(\boldsymbol{v}_{job_{i,1}}), \max(\boldsymbol{v}_{job_{i,2}}), \cdots, \max(\boldsymbol{v}_{job_{i,p}})]$，以表示职位描述。

简历部分的模型与岗位部分的相似。唯一不同的是最后一层使用均值池化将求职者经验表示集成到简历表示中。对此可用如下公式进行描述：

$$v^{r_i} = \frac{(v_{r_{i,1}} + v_{r_{i,2}} + \cdots + v_{r_{i,q}})}{n} \tag{1}$$

1.4　BiLSTM-Attention 层

1.4.1　BiLSTM

LSTM 模型是 Hochreiter 等学者[14]为了解决 RNN 模型因处理信息过多导致的梯度消失或梯度爆炸问题提出的模型。BiLSTM 作为 LSTM 的一种变体，由一个正向 LSTM 和一个反向 LSTM 模型拼接而成。其结构如图 3 所示。

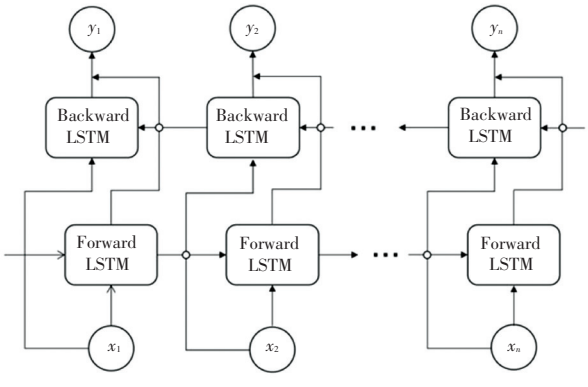

图 3　BiLSTM 结构图

Fig. 3　Structure diagram of BiLSTM

首先获得招聘信息中 $\boldsymbol{job}_{i,j}$ 的词向量表示：$\boldsymbol{w}_l^t = \boldsymbol{W}_e \boldsymbol{job}_{i,l}^t$，$\boldsymbol{w}_l^t \in R^{d_0}$，其中，$\boldsymbol{w}_l^t$ 表示第 l 条需求中第 t 个词的 d_0 维词嵌入，$\boldsymbol{W}_e$ 是参数矩阵，$\boldsymbol{job}_{i,l}^t$ 表示 $\boldsymbol{job}_i$ 中第 l 条需求的第 t 个词向量。对 $\boldsymbol{job}_{i,l}$ 中的每个

词,计算对应的语义表征可表示为:$\{\boldsymbol{h}_l^1,\boldsymbol{h}_l^2,\cdots,\boldsymbol{h}_l^s\}$,$\boldsymbol{h}_l^t=BiLSTM(w_l^{1:s},t)$,$\forall t\in[1,\cdots,s]$。同理,可得 $\boldsymbol{R}$ 对应的语义表征:

$$\boldsymbol{h}_{l'}^{t'}=BiLSTM(\boldsymbol{w}_{l'}^{1:u},t')\quad \forall t'\in[1,\cdots,u] \tag{2}$$

1.4.2 注意力层

在自然语言处理领域,注意力机制被用来为不同重要性的词或句子分配权重,权重越大的词越重要。将经过 BiLSTM 处理后的语义表征 $\{\boldsymbol{h}_l^1,\boldsymbol{h}_l^2,\cdots,\boldsymbol{h}_l^s\}$ 作为全连接层的输入,计算字符级别文本向量间的关联度,然后使用 *Softmax* 函数计算注意力分数 α,即:

$$\boldsymbol{\alpha}_{l,t}=\frac{\exp(\boldsymbol{e}_l^t)}{\sum_{z=1}^{s}\exp(\boldsymbol{e}_l^z)} \tag{3}$$

$$\boldsymbol{e}_l^t=\boldsymbol{v}_\alpha^{\mathrm{T}}\tanh(\boldsymbol{W}_\alpha\boldsymbol{h}_l^t+\boldsymbol{b}_\alpha) \tag{4}$$

其中,$\boldsymbol{v}_\alpha$、$\boldsymbol{W}_\alpha$ 和 $\boldsymbol{b}_\alpha$ 分别表示训练过程中的可学习参数;$\boldsymbol{v}_\alpha$ 表示 $\boldsymbol{job}_{i,l}$ 的上下文向量。接着通过式(5)计算词级别的岗位需求表征:

$$\boldsymbol{s}_{job_{i,l}}=\sum_{t=1}^{s}\boldsymbol{\alpha}_{l,t}h_{l,t} \tag{5}$$

再将词级别的岗位需求表征 $\{\boldsymbol{s}_{job_{i,1}},\boldsymbol{s}_{job_{i,2}},\cdots,\boldsymbol{s}_{job_{i,p}}\}$ 作为 BiLSTM 层的输入,并可推得:

$$\boldsymbol{c}_{job_{i;t}}=BiLSTM(\boldsymbol{s}_{job_{i;p}},t)\quad \forall t\in[1,2,\cdots,p] \tag{6}$$

运算得到隐层状态向量 $\{\boldsymbol{c}_{job_{i,1}},\boldsymbol{c}_{job_{i,2}},\cdots,\boldsymbol{c}_{job_{i,p}}\}$,此后根据每个能力要求的隐藏状态和所有能力要求的上下文向量间的相似性计算出每个能力要求的重要性 $\boldsymbol{\beta}_t$,即:

$$\boldsymbol{\beta}_t=\frac{\exp(\boldsymbol{f}_t)}{\sum_{z=1}^{p}\exp(\boldsymbol{f}_z)} \tag{7}$$

$$\boldsymbol{f}_t=\boldsymbol{v}_\beta^{\mathrm{T}}\tanh(\boldsymbol{W}_\beta\boldsymbol{c}_{job_{i,t}}+\boldsymbol{b}_\beta) \tag{8}$$

其中,$\boldsymbol{W}_\beta$、$\boldsymbol{b}_\beta$ 和 $\boldsymbol{v}_\beta$ 是可学习参数,最后句子级别的岗位需求表征可用式(9)计算:

$$\boldsymbol{g}^{job_i}=\sum_{t=1}^{p}\beta_t\boldsymbol{c}_{job_{i,t}} \tag{9}$$

同理可得简历的需求表征为:

$$\boldsymbol{g}^{r_i}=\sum_{t=1}^{q}\delta_t\boldsymbol{c}_t^{r_i} \tag{10}$$

1.5 匹配预测层

将 TextCNN 层输出的职位需求向量 $\boldsymbol{v}^{job_i}$ 与通过 BiLSTM-Attention 表示的职位向量 $\boldsymbol{g}^{job_i}$ 进行融合,即 $\boldsymbol{job}=concat[\boldsymbol{g}^{job_i};\boldsymbol{v}^{job_i}]$,同理可以得到简历的特征为 $\boldsymbol{geek}=concat[\boldsymbol{g}^{r_i};\boldsymbol{v}^{r_i}]$。为了预测彼此之间的匹配程度,将其输入全连接网络预测人岗匹配程度,即:

$$\boldsymbol{D}=Leakyrelu(\boldsymbol{W}_d[\boldsymbol{job};\boldsymbol{geek};\boldsymbol{job}-\boldsymbol{geek}]+\boldsymbol{b}_d)$$

$$\tilde{\boldsymbol{y}}=Sigmoid(\boldsymbol{W}_y\boldsymbol{D}+\boldsymbol{b}_y) \tag{11}$$

其中,$\boldsymbol{W}_d$、$\boldsymbol{b}_d$、$\boldsymbol{W}_y$、$\boldsymbol{b}_y$ 是可学习参数。

2 实验结果与分析

2.1 数据集描述

为验证 BATPJF 模型的有效性,实验使用智联招聘人岗匹配数据集。为保护用户隐私,所有简历都做了脱敏处理。原始数据集包含 4 500 份简历、269 534 份招聘信息和 700 938 条申请记录,在剔除职位描述为空、没有成功申请的数据之后,最终数据集见表 1。将数据集按 8∶1∶1 的比例划分为训练集、测试集和验证集。本文为每个正样本均匀地抽取一个负样本组成训练集。

表 1 数据集的基本统计信息

Tab. 1 Basic statistics of data set

统计信息	值
简历	4 228
招聘信息	202 319
申请	485 811
成功	19 989
失败	465 822
稀疏性/%	0.056

由表 1 的交互(申请、成功、失败)记录可知,成功率仅约为 4%,这从侧面反映了人才招聘工作的困难,而这正是本文工作的意义与价值所在。

2.2 实验描述

本文以申请成功的工作-简历对作为正样本,以未申请成功的工作-简历对作为负样本对模型进行训练。以下是实验中的一些基础设置:*batch_size* 设置为 128,*epoch* 设置为 300。测试集的大小设置为 128,验证集的大小设置为 1 024,若验证集上的评估结果连续 10 个 *epoch* 没有增加,训练将提前停止。为了尽可能地避免过拟合现象的发生,将 *drop_out* 设置为 0.5,并在 TextCNN、BiLSTM 中分别选择 *Prelu*、*LeakyRelu* 作为激活函数,以提高模型的非线性表达能力并加速模型收敛速度。为了更好地学习模型参数,选择 Adam 作为优化器进行训练。模型的具体参数见表 2。

表 2 参数设置表

Tab. 2 Parameters setting

LSTM 参数		TextCNN	
参数	值	参数	值
词向量维度	128	通道数	20
隐藏层大小	64	卷积核大小	(5,1)(3,1);(5,1)(5,1)
学习率	0.001	隐藏层大小	64

2.3 实验结果

为验证本文提出的 BATPJF 模型性能，将其与 BPR[15]、BPJFNN[16]、APJFNN[16]、LightGCN[16]、PJFFF[17]模型进行比较，评价指标主要采用命中率（Hit Rate @ k，*HR*）和归一化折损累计增益（*NDCG@k*）来综合判断模型的性能。实验结果见表 3。表 3 中，带“*”符号的表示对比模型中的最佳结果。

表 3　不同模型在数据集的结果

Tab. 3　Results of different models in the dataset

模型	指标	
	HR@5	*NDCG*@5
BPR	0.393 9	0.305 3
BPJFNN	0.576 3	0.403 2
APJFNN	0.579 5*	0.426 5
LightGCN	0.428 8	0.350 7
PJFFF	0.512 6	0.376 9
BATPJF	**0.591 0**	**0.436 7**

由表 3 可知，本文提出的模型相比参照模型中最好的模型 APJFNN 在指标 *HR*@5 和 *NDCG*@5 上分别提高了 1.98%和 2.39%。由此可见，BATPJF 模型能充分发挥 TextCNN 提取局部特征的优点与 BiLSTM 具有记忆功能的优点，且注意力机制的加入，可以计算工作要求对不同工作经验的重要性以及工作经验对不同工作要求的贡献，因此性能优于其他对比模型。

3　结束语

本文针对人岗匹配问题提出了一种基于深度学习的端到端人岗匹配模型，模型首先通过词嵌入将文本表示成低维词向量矩阵，接着利用 TextCNN 和 BiLSTM 分别提取职位描述和个人经历文本中的局部关键信息和上下文信息，最后将得到的结果进行融合，提高了人岗匹配的效果。通过与其他模型进行对比，证明了本文模型 BATPJF 的有效性。由于人岗匹配包括结构化和非结构化数据的匹配，而本文仅考虑了非结构化文本数据的匹配，所以下一阶段的工作是将结构化的信息也考虑进模型中，以便进行更好的人岗匹配。

参考文献

[1] ZHANG Yingya, YANG Cheng, NIU Zhixiang. A research of job recommendation system based on collaborative filtering[C]//2014 Seventh International Symposium on Computational Intelligence and Design. Hangzhou, China:IEEE, 2014, 1: 533-538.

[2] YAN Rui, LE Ran, SONG Yang, et al. Interview choice reveals your preference on the market: To improve job-resume matching through profiling memories[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. Anchorage, AK, USA:ACM, 2019: 914-922.

[3] NASSER S, SREEJITH C, IRSHAD M. Convolutional neural network with word embedding based approach for resume classification[C]//2018 International Conference on Emerging Trends and Innovations In Engineering And Technological Research (ICETIETR). Ernakulam, India:IEEE, 2018: 1-6.

[4] ZHU Chen, ZHU Hengshu, XIONG Hui, et al. Person-job fit: Adapting the right talent for the right job with joint representation learning[J]. ACM Transactions on Management Information Systems (TMIS), 2018, 9(3): 1-17.

[5] KHATUA A, NEJDL W. Matching recruiters and jobseekers on twitter[C]//2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). The Hague, Netherlands: IEEE, 2020: 266-269.

[6] METLAPALLI A C, MUTHUSAMY T, BATTULA B P. Classification of Social Media Text Spam Using VAE-CNN and LSTM Model[J]. Ingénierie des Systèmes d Inf., 2020, 25(6): 747-753.

[7] ZHOU Chunting, SUN Chonglin, LIU Zhiyuan, et al. A C-LSTM neural network for text classification[J]. arXiv preprint arXiv:1511.08630, 2015.

[8] QIN Chuan, ZHU Hengshu, XU Tong, et al. An enhanced neural network approach to person-job fit in talent recruitment[J]. ACM Transactions on Information Systems (TOIS), 2020, 38(2): 1-33.

[9] JIANG Junshu, YE Songyun, WANG Wei, et al. Learning effective representations for person-job fit by feature fusion[C]//Proceedings of the 29th ACM International Conference on Information & Knowledge Management. New York, USA: ACM, 2020: 2549-2556.

[10] 李超凡，马凯. 基于注意力机制结合 CNN-BiLSTM 模型的电子病历文本分类[J]. 科学技术与工程，2022，22(6)：2363-2370.

[11] 吉兴全，曾若梅，张玉敏，等. 基于注意力机制的 CNN-LSTM 短期电价预测[J]. 电力系统保护与控制，2022，50(17)：125-132.

[12] 任建吉，位慧慧，邹卓霖，等. 基于 CNN-BiLSTM-Attention 的超短期电力负荷预测[J]. 电力系统保护与控制，2022，50(08)：108-116.

[13] LE Ran, HU Wenpeng, SONG Yang, et al. Towards effective and interpretable person-job fitting[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. Beijing, China: ACM, 2019: 1883-1892.

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

[15] RENDLE S, FREUDENTHALER C, GANTNER Z, et al. Bayesian personalized ranking from implicit feedback[C]//Proc. of Uncertainty in Artificial Intelligence. [S.l.]: AAAI, 2014: 452-461.

[16] QIN Chuan, ZHU Hengshu, XU Tong, et al. Enhancing person-job fit for talent recruitment: An ability-aware neural network approach[C]//The 41st international ACM SIGIR Conference on Research & Development in Information Retrieval. Ann Arbor, MI, US: ACM, 2018: 25-34.

（下转第 59 页）