

## 基于知识图谱的大学生职位推荐方法

徐红艳,王碧莹,冯 勇,王嵘冰\*

(辽宁大学 信息学院 辽宁 沈阳 110036)

**摘 要:** 近年来,我国大学生的就业压力一直较大.我国经济的发展虽然创造出大量的职位,但应届大学生由于缺乏工作经验和社会阅历,面对大量的职位招聘时很难作出正确和理性的选择.为更好地推动大学生就业,本文提出了一种基于知识图谱的大学生职位推荐方法.该方法首先运用知识图谱技术构建职位知识图谱,使用图形数据库 Neo4j 完成知识的存储工作,通过将知识图谱中的实体与关系进行向量化表示,计算简历与职位描述间的相似度;然后通过用户对职位的行为矩阵计算得到职位间的相似度;最后将2种相似度进行融合,生成职位推荐列表.对比实验结果表明,本文所提方法在准确率和召回率等指标上均有显著提升,相较于主流职位推荐方法具有更高的精准度.

**关键词:** 职位推荐;知识图谱;大学生就业;向量化表示

中图分类号: TP 319 文献标志码: A 文章编号: 1000-5846(2022)02-0107-09

## A Method of Job Recommendation for College Students Based on Knowledge Graph

XU Hong-yan, WANG Bi-ying, FENG Yong, WANG Rong-bing\*

(College of Information Liaoning University, Shenyang 110036, China)

**Abstract:** In recent years, the employment pressure of college students in our country has been great. Although the development of the national economy has created a large number of jobs, it is difficult for recent college students to make accurate and rational choices in the face of a large number of jobs due to lack of work and social experience. In order to better promote the employment of college students, this paper proposes a job recommendation method for college students based on knowledge graphs. This method first constructs a job knowledge graph, uses graph data base to complete the knowledge storage work, and vectorizes the entities and relationships in the knowledge graph. Meanwhile calculate the similarity between the resume and the job description; then calculate the similarity between the positions through the user's behavior

收稿日期: 2021-09-20

基金项目: 辽宁省教育厅科学研究基金面上项目(LJKZ0085); 辽宁省普通高等教育本科教学改革项目(2021-22); 辽宁大学本科教学改革项目(2021-10); 辽宁省经济社会发展研究课题研究成果基金项目(2022lslwzzkt-018)

作者简介: 徐红艳(1972-),女,辽宁沈阳人,硕士,教授,研究方向: 个性化推荐,大数据技术.

\* 通讯作者: 王嵘冰, Email: wrb@lnu.edu.cn.

matrix for the positions; finally, the two similarities are merged to generate a job recommendation list. The results of comparative experiments show that the method proposed in this paper has a significant improvement in indicators such as accuracy rate and recall rate and has higher accuracy than other job recommendation methods.

**Key words:** job recommendation; knowledge graph; employment of college students; vectorized representation

## 0 引言

在国内经济稳定、高速发展及国际重视中国市场的背景下,各行业都在扩大生产经营规模,藉此产生了更多的职位需求。在信息技术的推动下,这些职位需求数据在网络空间迅猛增长和快速传播,进而引发信息过载问题<sup>[1]</sup>。在海量的职位信息中找到适合的职位对求职者来说是一件复杂而又耗时的事情,尤其对缺乏工作和社会经验的应届大学生,职位选择显得愈发困难。

当前的职位推荐系统大多是热门职位推荐或是基于关键词匹配的推荐,然而热门职位推荐缺乏针对性,基于关键词匹配的推荐不能满足用户的个性化职位需求<sup>[2]</sup>。知识图谱作为一种新兴知识处理技术,它可以从知识层面表述知识实体及知识间的关联,将其应用于职位推荐领域可以更好地处理职位供需间的匹配,带来更为准确的职位推荐。面对互联网中海量的招聘数据,基于知识图谱的职位推荐不仅能满足求职者对所需职位的高覆盖率,也能体现出职位推荐的个性化特征。因此,本文提出了一种基于知识图谱的大学生职位推荐方法,通过借助知识图谱工具显示招聘资源间的结构关系以及职位的分布情况,根据知识图谱中展现的关系,将与求职大学生有着直接或间接关系的职位推荐给他们,这样有助于提高大学生的就业率。本文通过引入知识图谱相关技术,开发出来的职位推荐系统能为高校毕业生提供一个简洁高效的职位推荐平台,使学生可以更精准地获取到适合自己的职位信息,这样可以加快毕业生的求职速度,有效地缓解学生的就业压力,也能为企业节约时间和招聘成本。

## 1 相关工作

### 1.1 知识建模

知识建模是知识图谱构建技术的基础。有 2 种搭建知识图谱的方法:自顶向下(top-down)和自底向上(bottom-up)。自顶向下是预先定义好本体和数据模式,然后把实体加进知识库系统中,这种搭建方法必须采用当前的部分结构化知识库系统作为基础知识库系统,FreeBase 知识库就是采用这种方法构建而成的。自底向上是从一些对外界开放的数据信息中获取实体,选择可靠性比较高的放到知识库系统中,然后对顶层的本体模式进行搭建,很大一部分知识图谱就是依靠自底向上的方法搭建出来的,这里面最具代表性的是 Google 和微软构建的知识库,这也符合互联网数据内容知识产生的特点。

### 1.2 知识抽取

知识抽取中最重要的是如何从异构数据库中自动抽取信息内容以获得替代的知识单元。相对于

垂直领域的知识图谱,其数据通常采用2种方式获得:1)业务本身的生成数据信息,这类数据大体上是指企业内部的数据库表,并将其以结构化形式存储起来;2)以网页形式存在的非结构化数据,这类数据一般是从Internet上抓取的公开数据。第1种类型的数据通常经过简单的预处理就可以用作AI系统后续的输入数据,而后1种类型的数据就要依靠自然语言的处理技术来获得结构化的数据信息。这一点恰好是知识抽取部分的一个难题,它涉及的核心技术包括实体抽取、关系抽取以及属性抽取<sup>[3]</sup>。

### 1.2.1 实体抽取

实体抽取指的是从文本数据中自动检索出命名实体<sup>[4]</sup>,它的目的是在知识图谱中构建节点。实体抽取的质量很大程度上影响着后面知识获取的质量和获取速度,所以说实体抽取是知识抽取环节中最基础也是最重要的模块。

### 1.2.2 关系抽取

文本语料在前面的实体抽取后获得的是一系列分散开来的命名实体,为了更好地从中获得语义信息,在把这些实体和概念关联在一起之前,必须从与之对应的语料中获取出实体间的关联信息,这样才能得到知识系统的网状结构。关系抽取技术,归根结底就是从文本语料中抽取实体之间关联关系的一项技术<sup>[5]</sup>。

### 1.2.3 属性抽取

属性抽取的最终目的是要从不一样的信息源中收集指定实体的属性相关信息,以达到详细描述实体属性这一目的<sup>[6]</sup>。如针对某款笔记本电脑,可以从互联网中获取多源异构的数据,从中得到其品牌、配置等信息。

如果把实体的属性值看作是一种特殊的实体,那么属性抽取实际上也是一种关系抽取。百科类网站提供的半结构化数据是通用领域属性抽取研究的主要数据来源,但具体到特定的应用领域,涉及大量的非结构化数据,属性抽取仍然是一个巨大的挑战<sup>[7]</sup>。

## 1.3 知识表示

知识表示旨在通过计算机来展现知识的可行性和有效性,它想要达到数据结构与控制结构的统一,因此,不仅要注意知识存储上容易出现的状况,也要想到知识使用方面的问题。知识表示可以看作是叙述一组事物的约定,从而将知识表示为能够被机器识别处理的数据结构<sup>[8]</sup>。

## 2 基于知识图谱的大学生职位推荐方法描述

本文提出了一种基于知识图谱的职位推荐方法,推荐方法的框架如图1所示。通过将知识图谱的概念引入到职位推荐中,以此来弥补当前职位推荐方法中存在的推荐精度不高、缺乏个性化等问题。所提方法的思想:首先将知识图谱映射到低维稠密向量空间里,以获取职位信息中的职位相似度;接下来依据职位信息生成用户的行为矩阵,进而计算出职位之间的相似度;然后通过融合这2种类型的职位相似度,得到融合后的职位相似度矩阵,并以该矩阵作为基础,得到用户对职位库中所有职位的估算分值;最后依据这个分数得出推荐给用户的职位排序列表。下面分别对方法中的核心环节加以详细描述。

### 2.1 向量化表示

关于知识建模等构建知识图谱的相关技术和方法已经在第1节中详细介绍,下面介绍知识图谱

的向量化表示.

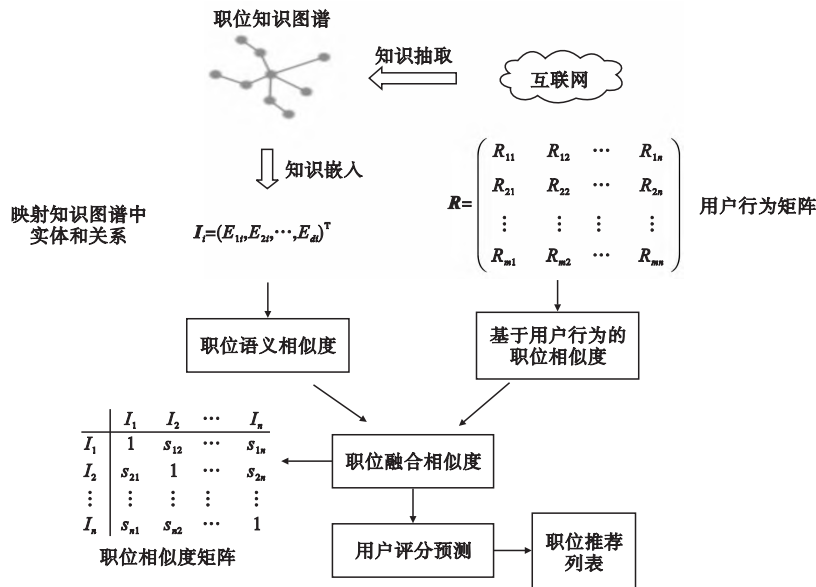


图1 基于知识图谱的大学生职位推荐方法框架

这里使用 TransE 算法将知识图谱中的实体与关系嵌入到低维向量空间中<sup>[9]</sup>, TransE 算法的主要思想是将三元组  $(h, r, t)$  中的关系看作从实体 head 到实体 tail 的翻译. 其中  $h$  是头实体的向量表示,  $r$  是关系的向量表示,  $t$  是尾实体的向量表示. 通过不断调整三者的向量表示, 使得  $h + r$  尽可能与  $t$  相等, 即  $h + r = t$ , 那么就说明知识图谱中的三元组成立. 如果成立, TransE 算法就必须达到公式 (1) 和公式 (2) 的要求:

$$\text{对正样本三元组: } h + r \approx t \quad (1)$$

$$\text{对负样本三元组: } h + r \neq t \quad (2)$$

在上面两式中  $h + r$  与  $t$  的关系表示与向量相似度的近似等价程度. 这里向量相似度通过计算欧氏距离实现, 得分函数的公式如式 (3) 所示:

$$f(h, r, t) = \|h + r - t\|_{L1/L2} \quad (3)$$

得分函数值越小, 越有利于正样本三元组; 得分函数值越大, 对负样本三元组越有利<sup>[10]</sup>. 接着, TransE 算法通过损失函数测试表示学习的效果, 损失函数的计算如公式 (4) 所示:

$$L = \sum_{(h, r, t) \in S} \sum_{(h', r', t) \in S'_{(h, r, t)}} [\gamma + f(h, r, t) - f(h', r', t)]_+ \quad (4)$$

其中  $S$  是正样本的集合,  $S'_{(h, r, t)}$  是三元组  $(h, r, t)$  的负样本. 知识图谱中三元组的负样本是通过随机替换头实体  $h$  或尾实体  $t$  后经过大量训练得到的. 以  $x$  表示公式 (4) 中方括号内的式子, 则  $[x]_+$  指的是  $\max(0, x)$ ;  $\gamma$  表示函数间隔的大小, 且  $\gamma > 0$ . TransE 算法通过不断地训练和优化算法, 损失函数的值可以达到最小. 算法通过不断地迭代来更新参数, 直到收敛或者达到最大迭代次数, 这时知识图谱中的实体就会被映射到向量空间中的相应位置<sup>[11]</sup>.

## 2.2 相似度计算

本小节主要对系统中职位的相似度进行计算, 分为 2 个部分: 基于简历的职位相似度计算和基于用户行为的职位相似度计算.

本文所提方法的相似度计算有2个来源:1)通过提取用户简历中的信息计算其与职位的相似度;2)由用户在系统上对职位作出查看或收藏等用户行为而得到的职位相似度.最后融合这2种来源的相似度,以此获得最终的职位信息相似度.

### 2.2.1 基于简历的职位相似度

前文介绍了知识图谱中富含大量的语义信息,可以从语义的层面描述职位之间的相似度<sup>[12]</sup>.2.1节里又详细地阐明了知识图谱中实体向量化表示的具体步骤,因此获得了实体在高维空间里的向量表示.接下来根据职位的嵌入向量,介绍职位语义相似度的详细计算.

首先,把知识图谱上职位信息的实体和关系都映射到 $d$ 维空间里去,职位 $I_i$ 嵌入一个 $d$ 维的向量如公式(5)所示:

$$\mathbf{I}_i = (E_{1i} E_{2i} \cdots E_{pi})^T \quad (5)$$

这里的 $E_{pi}$ 指的是职位 $I_i$ 在第 $p$ 维上嵌入向量的数值.

然后使用欧几里得距离计算后获得的评分函数,对知识图谱中的实体和关系做嵌入操作<sup>[13]</sup>.还需使用它测算向量的语义相似度,以此来精准表达用户的简历与职位信息之间的相似性.在这一步之前,需要先用公式(6)来计算职位之间的距离:

$$d(\mathbf{I}_i, \mathbf{I}_j) = \sqrt{\sum_{k=1}^d (E_{ki} - E_{kj})^2} \quad (6)$$

通过公式(6)的计算最终欧几里得距离大于等于零,再经过公式(7)的计算,可以把它控制在区间 $(0, 1]$ :

$$\text{sim}_{\text{knowledge\_graph}}(\mathbf{I}_i, \mathbf{I}_j) = \frac{1}{1 + d(\mathbf{I}_i, \mathbf{I}_j)} = \frac{1}{1 + \sqrt{\sum_{k=1}^d (E_{ki} - E_{kj})^2}} \quad (7)$$

这个式子的计算结果越大,就表明用户的简历信息与职位信息之间的语义相似度数值越大.如果该数值是等于1的,简历信息与职位信息之间的语义相似度就是最大的,可以认定两者相似程度很高;反之,如果该值是趋近于0的,就认定两者之间毫无关联.

### 2.2.2 基于用户行为的职位相似度

根据用户行为得到的所有信息可以看出用户自身对于职位信息的关注<sup>[14]</sup>.基于物品的协同过滤推荐的主要思路也是如此.基于物品的协同过滤推荐可实现领域范围之内的推荐,是依据访问系统的全部用户对物品做出的用户行为进而得到物品之间的相似度,接着把认定是相似的物品按照用户行为的历史数据进行推荐.本文将用户对职位信息的用户行为划分成许多个类别,这里以最为寻常的评分信息作参照对职位信息进行向量化表示.

假设本文实现的系统里有访问系统的 $m$ 个用户 $U = (U_1, U_2, \cdots, U_m)$ 和被单位发布出来的 $n$ 个职位 $I = (I_1, I_2, \cdots, I_n)$ ,那么用户对职位的评分信息可以表示为如公式(8)所示的 $m \times n$ 矩阵 $R_{m \times n}$ :

$$R = \begin{pmatrix} R_{11} & R_{12} & \cdots & R_{1n} \\ R_{21} & R_{22} & \cdots & R_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ R_{m1} & R_{m2} & \cdots & R_{mn} \end{pmatrix} \quad (8)$$

接下来,要把职位 $I_i$ 表示成如公式(9)所示的一个 $m$ 维的向量,它的数值意味着在每一个维度

上都有相应的求职者对职位的一个打分.

$$\mathbf{I}_i = (R_{1i} \ R_{2i} \ \cdots \ R_{mi})^T \quad (9)$$

这里利用余弦相似度公式得到基于用户行为的职位信息相似度<sup>[15]</sup>, 具体的计算如公式(10)所示:

$$\text{sim}_{\text{user\_behavior}}(\mathbf{I}_i, \mathbf{I}_j) = \frac{\mathbf{I}_i \cdot \mathbf{I}_j}{\|\mathbf{I}_i\| \|\mathbf{I}_j\|} = \frac{\sum_{k=1}^m R_{ki} \times R_{kj}}{\sqrt{\sum_{k=1}^m R_{ki}^2} \times \sqrt{\sum_{k=1}^m R_{kj}^2}} \quad (10)$$

公式(10)计算出来的数值越大, 就代表职位  $\mathbf{I}_i$  与职位  $\mathbf{I}_j$  之间被用户行为的影响程度越深. 一旦计算结果等于 1, 就可以说这 2 个职位从某种程度上来讲是一样的; 如果计算结果等于 0, 就意味着这 2 个职位之间完全无关. 另外, 如果这个用户  $u$  没有给职位  $i$  打出评分数据, 那么  $R_{ui}$  的数值就是零.

### 2.2.3 相似度融合

根据获取到的简历信息对于职位相关信息的嵌入向量, 可以获得基于简历信息的职位相似度  $\text{sim}_{\text{knowledge\_graph}}$ . 同时, 依据用户行为而生成的职位相关的评分矩阵, 可以获得基于用户行为的职位相似度  $\text{sim}_{\text{users\_behavior}}$ . 这里用加权的方法对这 2 种相似度进行融合计算, 可以获取所需的最终相似度数值, 详细的计算方式如公式(11)所示:

$$\text{sim}(\mathbf{I}_i, \mathbf{I}_j) = \alpha \cdot \text{sim}_{\text{knowledge\_graph}}(\mathbf{I}_i, \mathbf{I}_j) + (1 - \alpha) \text{sim}_{\text{users\_behavior}}(\mathbf{I}_i, \mathbf{I}_j) \quad (11)$$

这里  $\alpha$  代表 2 种相似度算法的融合度, 它的区间在  $[0, 1]$ , 表示最终职位相似度中根据简历信息计算的职位相似度的占比大小.  $\alpha$  的数值等于 1 说明最终相似度的结果全部是根据简历中用户信息获取的职位相似度数值;  $\alpha$  的数值等于 0 说明最终相似度结果全部是根据用户行为计算出来的相似度值. 职位相似度融合之后的区间也是  $[0, 1]$ .

对全部用户信息与职位信息之间的相似度进行融合以后, 基于职位信息的相似度通过矩阵的方式展现如下:

$$\begin{array}{c|cccc} & \mathbf{I}_1 & \mathbf{I}_2 & \cdots & \mathbf{I}_n \\ \hline \mathbf{I}_1 & 1 & s_{12} & \cdots & s_{1n} \\ \mathbf{I}_2 & s_{21} & 1 & \cdots & s_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{I}_n & s_{n1} & s_{n2} & \cdots & 1 \end{array}$$

这里  $s_{ij}$  表示职位  $\mathbf{I}_i$  和职位  $\mathbf{I}_j$  之间的相似度  $s_{ij} = s_{ji}$ . 另外, 如果  $i$  等于  $j$ ,  $s_{ij}$  的值为 1, 即这里全部的职位信息和它们自己的相似度数值都是 1.

### 2.3 推荐列表生成

这里帮助求职者进行偏好职位推荐的基本思想: 首先根据 2.2 节计算获取到的简历信息与职位之间生成的相似度矩阵以及求职者对于职位的用户行为生成的矩阵, 然后通过预测来获得每个求职者给未评价过的职位内容的打分数值; 最后系统根据求职者对每个职位内容的预测评分生成一个按照评分由高到低排序的职位列表.

生成职位相似度矩阵之后, 这里通过  $p_{ui}$  指代预估出来的用户  $u$  对职位  $i$  的分值, 借助公式(12)就可以计算出  $p_{ui}$  的数值.



$$p_{ui} = \frac{\sum_{j \in N(u) \cap S(i, k)} (s_{ij} \times R_{uj})}{\sum_{j \in N(u) \cap S(i, k)} s_{ij}} \quad (12)$$

其中  $s_{ij}$  表示职位  $I_i$  和职位  $I_j$  之间的相似度,  $R_{uj}$  指代现存评分矩阵里用户  $u$  给职位  $j$  打出的分值,  $N(u)$  是用户  $u$  已经给出评分的职位信息的全部集合,  $S(i, k)$  指代和职位  $i$  相关性最高的  $k$  个职位的全部集合。

经过公式(12)计算得出的预测评分数值越高,代表该职位与用户过往评分数值很大的职位相似程度越高,其在系统中职位推荐列表里的位置排列更靠上些。

在本文实现的职位推荐系统中,系统给每位求职者都推荐一个含有  $N$  个职位在内的详情列表。系统给求职者作出推荐的依据如下:如果求职者对职位的预测评分数值很高,那么就有理由认定该用户对这个职位偏好强烈,这个职位信息在列表中的位置就靠前,也更易被访问系统的求职者注意到;反之,这个职位的信息可能出现在列表中相对靠后的位置,甚至可能因为数值过低没有出现在列表中。按照这个思路,系统最终会生成一个包括  $N$  个职位信息在内的、依据数值由高至低排列的职位推荐列表。

### 3 实验分析

#### 3.1 数据集与实验环境介绍

本系统通过网络爬取技术抓取互联网招聘数据<sup>[16]</sup>,数据包括 23 213 条节点数据,11 种关系类型在内的关系数据 126 559 条。下面将其按照 7:3 的比例分为训练集和测试集,并针对这 2 个数据集进行实验。

本文实验环境如下:

- 1) 操作系统: Windows 7
- 2) CPU: 频率 2.60 GHz
- 3) 编程工具: PyCharm Community Edition 2019.2.3 和 Python
- 4) 显卡信息: Intel(R) HD Graphics 4000

#### 3.2 指标和参数选取

本文实现的职位推荐系统是向用户推荐偏好的职位信息,因此针对系统生成的职位推荐列表,选取 2 个数据集采用交叉验证的方法进行实验分析,从准确率(Precision)和召回率(Recall) 2 个指标评价职位推荐系统的推荐结果。

为了更加清楚地解释准确率和召回率,本文定义一些符号如下:

**TP**( True Positive) 代表推荐系统已经推荐,且实际的用户行为也确实发生了的样本;

**FP**( False Positive) 代表推荐系统已经推荐,但实际的用户行为并没有发生的样本;

**FN**( False Negative) 代表推荐系统没有推荐,但实际的用户行为发生了的样本。

准确率的定义如公式(13)所示:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (13)$$

准确率计算公式的含义是,在所有被职位推荐系统预测推荐的样本中实际推荐的概率。

召回率的定义如公式(14)所示:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (14)$$

召回率计算公式的含义是,在职位推荐系统实际推荐的样本中被预测推荐的职位概率。

这里融合度的变化范围是  $[0, 1]$ , 设定融合度的变化数值为 0.1, 共做 11 组实验, 即观察融合度数值以 0.1 为步长的情况下准确率和召回率的变化情况。融合度数值等于 0, 代表职位推荐方法全部采用基于简历信息与职位信息的相似度算法; 融合度数值等于 1, 代表推荐方法使用的都是基于用户行为的相似度算法。随后, 用数据集对这些算法做实验分析, 以此得到指标状况最佳时的融合度数值。下面对不同融合度数值下的各个算法进行实验分析。

图 2 为 2 种相似度算法在不同融合度数值下准确率指标变化情况, 图 3 为相同条件下召回率的变化情况。当融合度数值为 0 时, 表示基于简历信息与职位信息的相似度算法的指标情况。由图 2 和图 3 可以发现, 当融合度值在 0.6 的时候, 融合相似度算法在 2 个指标上都达到了最高点, 表明在这个融合度下, 该融合算法在准确率、召回率上的表现都是最佳的。

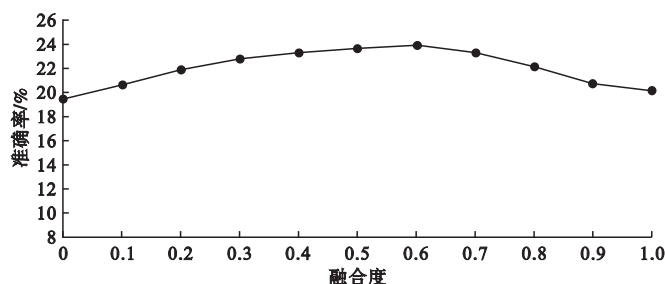


图2 不同融合度下的准确率

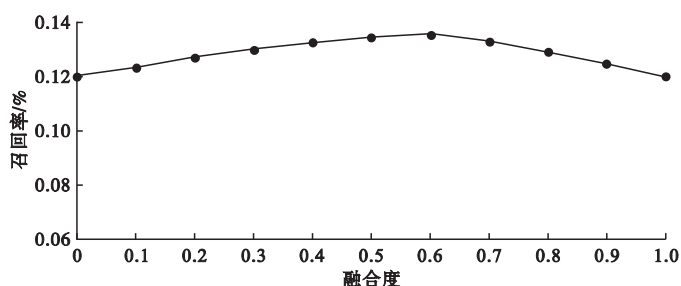


图3 不同融合度下的召回率

### 3.3 实验结果

本文选取使准确率和召回率都到达最理想效果的融合度 0.6, 验证当融合度值确定时本方法在不同近邻数  $K$  下的准确率和召回率, 使用主流应用的基于内容的推荐方法、协同过滤推荐方法以及混和推荐方法作为本文提出的推荐方法的对比方法。通过对比实验比较这 4 种方法在近邻数的不同选值下准确率和召回率的情况。近邻数  $K$  的值分别选取 50, 70, 90, 110 和 130, 实验 10 次取得平均值作为实验的最终结果。

从图 4 和图 5 中可以看出, 本文提出的推荐方法在准确率和召回率上表现良好, 说明在引入知识图谱概念的基础上, 对 2 种相似度进行融合, 推荐的效果比基于内容的推荐方法和协同过滤推荐方法相比要更好。当近邻数为 100 时, 本文提出的方法在准确率上提升了约 5%, 在召回率上提升了约 3%, 弥补了当前推荐方法的不足, 同时也说明了本文提出的职位推荐方法的优越性。



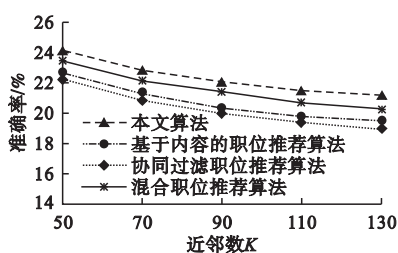


图4 不同近邻数K下的准确率

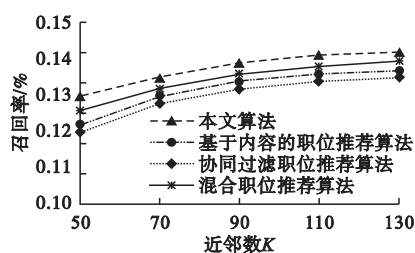


图5 不同近邻数K下的召回率

## 4 结束语

针对大学生面对海量招聘信息时的职位选择困惑,本文提出了基于知识图谱的职位推荐方法,通过使用图形数据库 Neo4j 构建一个求职招聘领域知识图谱,将知识图谱中的实体映射到低维稠密向量空间里,对其中的实体和关系进行向量化表示,计算用户的简历信息与职位信息之间的相似度,同时根据用户行为计算其与职位间的相似度,通过融合相似度计算得到用户对职位的预测评分,进而生成职位推荐列表,以提高推荐系统的精准度。经对比实验验证,本文所提职位推荐方法能够更好地满足当前大学生的就业需求。

## 参 考 文 献:

- [1] 蔺丰奇,刘益.网络化信息环境中信息过载问题研究综述[J].情报资料工作,2007(3):36-41,48.
- [2] 李兆军.高校舆情个性化推荐的研究与系统实现[D].济南:济南大学,2020.
- [3] 陈宗良.基于知识图谱的中文古诗词问答系统研究与实现[D].重庆:西南大学,2020.
- [4] 林旺群,汪淼,王伟,等.知识图谱研究现状及军事应用[J].中文信息学报,2020,34(12):9-16.
- [5] 张彬,徐建民,吴姣.大数据环境下基于知识图谱的用户兴趣扩展模型研究[J].现代情报,2021,41(8):36-44.
- [6] Bollacker K D, Cook R P, Tufts P. Freebase: A shared database of structured general human knowledge [C]// Proceedings of the twenty-second AAAI conference on artificial intelligence. Vancouver: AAAI, 2007: 1962-1963.
- [7] Bizer C, Lehmann J, Kobilarov G, et al. DBpedia-A crystallization point for the web of data [J]. Journal of Web Semantics, 2009, 7(3): 154-165.
- [8] Suchanek F M, Kasneci G, Weikum G. YAGO: A large ontology from wikipedia and WordNet [J]. Journal of Web Semantics, 2008, 6(3): 203-217.
- [9] Resnick P, Varian H R. Recommender systems [J]. Communications of the ACM, 1997, 40(3): 56-58.
- [10] 王兴旺,张欣.基于知识图谱的国际搜索引擎研究历程与热点演进[J].科技管理研究,2014,34(22):124-128,133.
- [11] 汤建民.学科知识图谱的绘制及在学科发展监测与评价中的应用[J].情报理论与实践,2009,32(10):55-59.
- [12] 宗乾进,袁勤俭,沈洪洲,等.2001—2010年国内情报学研究回顾与展望——基于知识图谱的当代学科发展方向研究[J].情报资料工作,2012(1):10-15.
- [13] 杜泽宇,杨燕,贺禄.基于中文知识图谱的电商领域问答系统[J].计算机应用与软件,2017,34(5):153-159.
- [14] 单佩佩.基于知识图谱的个性化推荐研究[J].电脑知识与技术,2020,16(36):177-178,184.
- [15] 徐锦阳,张高煜,王曼曦,等.招聘网站职位与简历的双向匹配相似度算法[J].信息技术,2016,40(8):43-46,51.
- [16] 阮光册,樊宇航,夏磊.知识图谱在实体检索中的应用研究综述[J].图书情报工作,2020,64(14):126-135.

(责任编辑 郑绥乾)