第十一届“泰迪杯”
数据挖掘挑战赛

优
秀
作
品

作品名称：泰迪内推平台招聘与求职双向系统的构建

荣获奖项：特等奖并获泰迪杯

作品单位：华南师范大学

作品成员：符文智 廖俊林 刘广生

泰迪内推平台招聘与求职双向系统的构建

摘要

在新时代背景下，大学生求职问题已成为广泛关注的社会热点，特别是疫情过后，多种社会因素加剧了应届生就业的严峻形势。对招聘信息与求职者信息进行分析研究，挖掘各类兴起行业相应的人才需求现状及发展趋势，有助于为广大求职者提供正确的就业指导。本文围绕招聘信息与求职信息，运用**自然语言处理技术**进行数据分析挖掘，基于**半结构化文本**构建 **Topsis-Fuzzy** 模型以评估岗位匹配度和求职者满意度，并结合**自由文本**构建**空间向量模型**来修正评估结果，最后根据**互惠概率、贪心优化和动态规划**分别构建了三种双向推荐模型。

针对问题一，本文首先运用 python 中 **Request 模块**获得，读取“找工作”，”找人才”以及各岗位信息和求职信息所在网址的 url 并分析其规律；然后，基于规律利用 json 库 **loads 函数**将截至 2023 年 4 月 17 日各求职者和各招聘岗位的信息以字典形式爬取下来；最后，运用 panda 库中 dataframe 函数将字典形式转化为表格形式，在此基础上运用 **concat 函数**分别完成对各招聘岗位和各求职者关键指标信息之间的横向拼接，并将其保存到对应的 csv 文件。经过上述爬取步骤，一共获得 10920 条求职者信息，1575 条招聘信息。

针对问题二，本文首先运用 **word2vec 模型**、**TF-IDF 算法**对所爬取信息展开文本预处理，包括**文本清洗**，**字段提取**，**停用词过滤**及**自由文本关键词提取**，在此基础上通过对各指标的词频统计分别绘制求职者和招聘岗位的整体画像，探究内推平台中求职者和招聘岗位的共同特征。考虑到仅仅采用整体画像对招聘信息和求职信息展开画像描述缺乏个性化描述，难以精准表达招聘岗位和求职者的需求，本文运用 **k-means++算法**对招聘岗位和求职者进行聚类，并基于不同类别求职者和招聘信息在聚类指标上的差异可视化分析对各类别求职者和招聘信息进行命名。最后，针对各类求职者和招聘信息的特征通过词频统计分别绘制画像，并从双向匹配的角度对各类画像展开分析，结论表明：“策马奔腾型岗位”与“目标求职者”相匹配，“经验丰富型岗位“和”智慧前沿型岗位”与“高薪求职者”相匹配，而“持续探索型岗位”与“入门求职者”相匹配。

针对问题三，本文综合考虑了半结构化文本和自由文本来评估岗位匹配度和求职者满意度。一方面，本文针对半结构化文本构建评价体系，然后运用 **AHP**

法和熵权法对评价指标进行组合赋权，考虑到部分离散型指标数据可能导致评价结果不准确，本文引入模糊理论构建 **Topsis-Fuzzy 评价模型**来评估岗位匹配度和求职者满意度，将离散型指标映射为模糊隶属度函数，使得评估结果更加可靠。另一方面，由于自由文本蕴含重要的求职与招聘信息，本文结合 **TF-IDF** 权重法和**向量空间模型(VSM)**计算自由文本中招聘信息与求职者信息的特征向量，然后结合**余弦相似度、皮尔逊相关系数、Jaccard 相似度、欧几里得距离**和**曼哈顿距离来**计算招聘信息与求职者信息的综合相似度，以综合相似度指标来刻画招聘企业与求职者的匹配程度和求职者对招聘企业的满意程度，最后利用相似度对基于半结构化文本得出的匹配度和满意度进行修正。

针对问题四，本文根据题中所给的招聘流程逐步构建了三个招聘求职双向推荐模型。为了提高双向推荐的效率，本文首先对**互惠概率**进行定义（代表企业 i 给求职者 j 发送 offer 的实际概率），然后构建基于互惠概率的招聘求职双向推荐模型，根据互惠概率是否大于阈值α来给招聘企业优先推荐求职者。为了使得履约率指标尽可能达到最高，本文构建了基于**贪心优化**的招聘求职双向推荐模型，主要思路是在每一轮发送 offer 时选择签约人数最多的策略以实现最终所有岗位的签约总人数最大的目标。考虑到贪心优化往往会导致局部最优解，本文基于**动态规划**的招聘求职双向推荐模型以期得出全局的最优推荐策略。最后本文对比三种模型的结果发现，三种模型的履约率均随着α的增加呈阶梯式下降；当$\alpha \in [0,0.2501]$时，基于**动态规划**的招聘求职双向推荐模型的履约率(60.65%)是最高的,需要发送 offer 的轮数(3)是最小的，说明该模型能以最低的成本达到最优的履约率结果，是相对最优的双向推荐模型。

关键词：双向推荐；自然语言处理技术；k-means++算法；Topsis-Fuzzy 模型；向量空间模型(VSM)；贪心优化；动态规划

目录

表目录

图目录

一、引言

1.1 研究背景及意义

在近年来新冠疫情的背景下，我国企业招聘与大学生求职方式在很大程度上从校园线下形式转变成线上形式网络招聘数据量呈现爆炸式增长。在这种变化下，如何有效提取并利用线上方式的招聘数据，实现线上企业成功招聘所需人才与大学生成功找到理想职业的双向盈利已成为数据分析与挖掘领域的研究热点[1]。

随着科技的不断进步与互联网的日益发展，近年泰迪内推、猎聘网等线上求职平台正逐渐走进大学生的视野里。线上求职平台的产生在一定程度上为企业寻找人才和求职者成功求职找到一条可行之径，但由于其求职者信息，招聘信息多种多样，在求职平台找到符合需求的人才与职业往往需要一定的时间成本。若能够通过对线上平台招聘与求职信息构建双向推荐系统，将会极大提高企业的人才查找效率及求职者的职业探索效率，对于缓解和解决毕业生求职问题和企业找人难问题具有重大意义。

1.2 挖掘目标

基于以上背景，本文利用泰迪内推平台这一聚焦于“大数据+”和“人工智能”领域的求职招聘网站，通过对其求职信息与招聘信息的爬取与分析建立泰迪内推平台的招聘与求职双向推荐系统，具体挖掘目标可分为以下 4 个部分：

- **招聘和求职信息爬取：**

基于泰迪内推平台“找工作”和“找人才”页面的共同特征，爬取所有的招聘与求职信息并依据招聘信息和求职者 ID 分别整理保存到对应的 csv 文件；

- **招聘和求职信息的预处理和分析：**

在通过数据清洗，文本分析对招聘和求职信息 csv 文件进行数据预处理的基础上，根据招聘信息与求职信息为招聘者和求职者分别构建用户画像；

- **构建岗位匹配度和求职满意度的模型**

本文大体可以分为两个步骤。一是构建岗位最低要求和求职者最低要求，筛选出不满足岗位最低要求的求职者和不满足求职者最低要求的岗位并分别定义其岗位匹配度和求职满意度为 0，二是构建评价模型，对未筛选求职者的岗位匹配度和岗位的求职满意度进行评分；

- **构建招聘求职双向推荐模型：**

基于图 1 的招聘流程，为平台设计招聘求职双向推荐模型，使得履约率

（履约率 $=\frac{\text{所有岗位签约人数之和}}{\text{所有拟聘岗位人数之和}}$）达到最高。

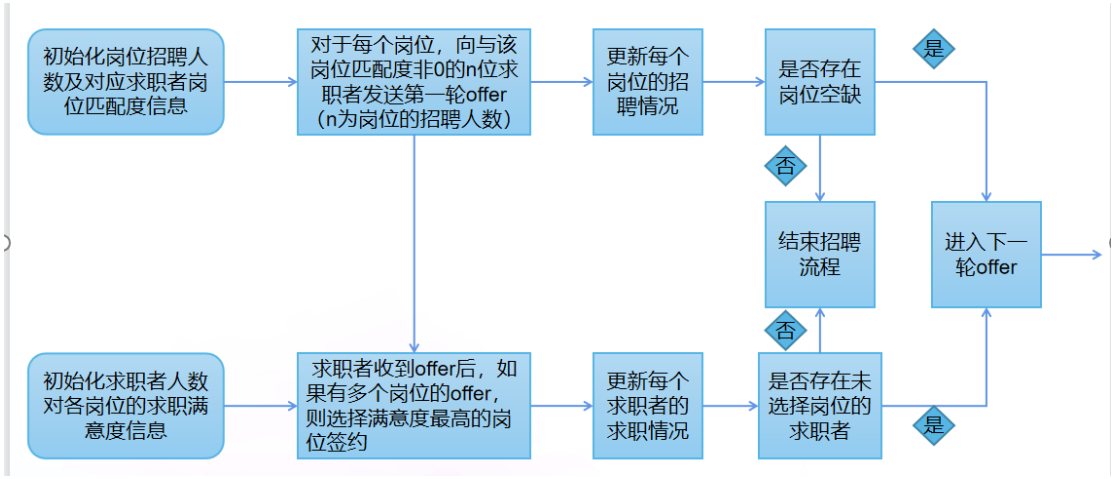


图 1 泰迪内推平台招聘流程

1.3 研究思路

本文针对上述问题进行初步的分析，拟定整体问题求解的思维导图如图 2 所示：

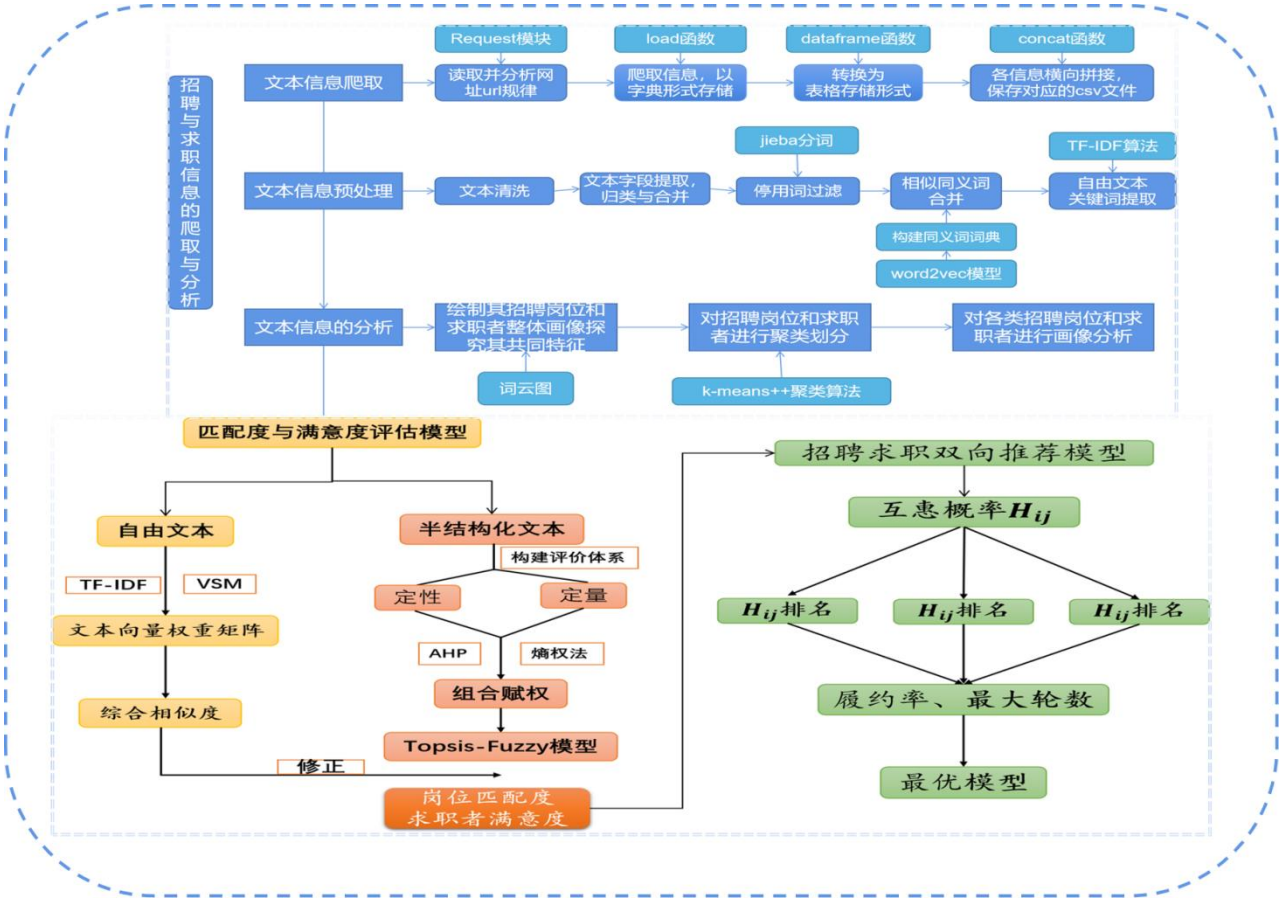


图 2 全文的解题思路图

二、招聘与求职信息爬取

基于题目所提供的泰迪内推平台的数据，本文利用 python 软件对其**截至 2023 年 4 月 16 日相关招聘岗位与求职信息**展开爬取。

2.1 网络爬虫方法设计

网络爬虫是一个从网页上自动抓取数据的程序。网络爬虫[2]通过程序模拟浏览器上网,从一个或若干个 URL 开始,抓取初始 URL 网页上的信息,在抓取网页数据的过程中,不断地从当前页面得到新的 URL 放入队列,不停地抓取 URL 队列中的网页数据,直到满足一定的终止条件。

常用的爬虫方法有通用型爬取方法和聚焦型爬虫方法[3]。考虑到相比于通用性爬取方法，聚焦性爬取方法在实施网页抓取时会对内容进行处理筛选，能够显著提高数据的爬取效率和爬取质量。在本文中本文主要采取聚焦性爬取方法展开招聘和求职信息的爬取。聚焦性爬取方法采集数据的流程如下图 3：

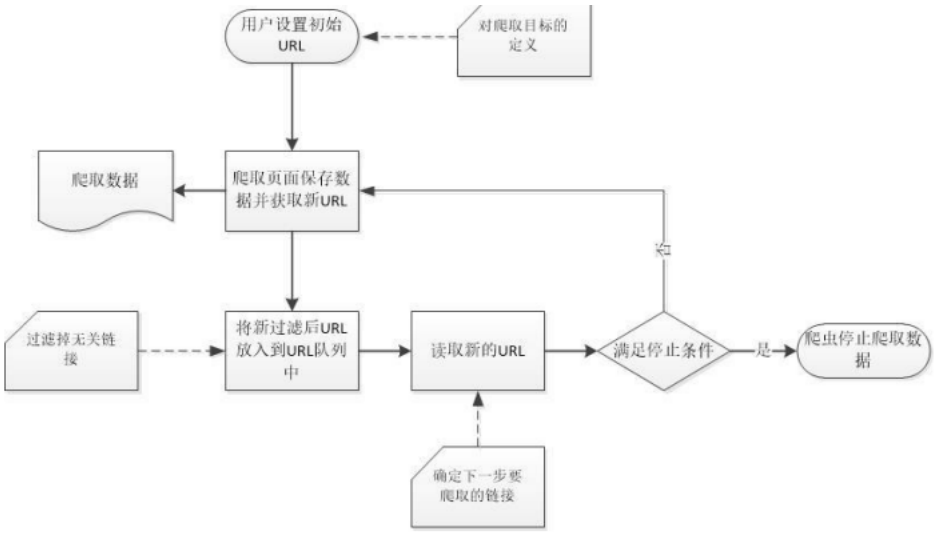


图 3　聚焦性爬取方法的采集数据流程图

2.2 招聘与求职信息的爬取过程

（1）导入模块：

在爬取数据前首先导入与爬虫相关的模块，包括 Requests 模块、json 模块以及 Pandas 模块。其中，Request 模块用于从网址获得并读取相关的招聘信息和求职信息，json 模块主要用于 request 所爬取信息的解析，pandas 模块主要用于爬取数据 excel 文件的导出。

（2）分析 url：

- “找工作”和“找人才”界面的 url 分析

在泰迪内推平台进入“找工作”和“找人才”的页面后，会有各个岗位和各个的求职者的关键指标信息，岗位关键指标信息包括职位、薪酬、公司规模等，求职者指标关键信息包括期望关键信息，期望职能等。

通过分别对“找工作”界面各岗位招聘信息和“找人才”各求职者 URL 的观察与对比,发现招聘页面各职业和求职页面的各求职者信息的 URL 的差异均主要体现在**页码**的不用，因此可设置**循环语句**，通过改变不同的页码,可以读取到平台上的所有招聘信息与求职信息。其重要代码（以招聘信息读取为例，求职信息读取同理）如下：

```
for i in range(1, 159):
    url = f'https://www.5iai.com/api/enterprise/job/public/es?pageSize=10&pageNumber={i}
' \'&willNature=&function=&wageList=%255B%255D&workplace=&keyword='
    response = requests.get(url, headers=headers)
```

- “求职岗位”和“求职者信息”界面的 url 分析

由于“找工作”和“找人才”里蕴含的“求职岗位”和“求职者信息”界面涵盖职位描述，求职者简介等更多丰富的半结构数据以及自由文本数据。为了进一步丰富求职岗位信息和求职者信息，使下文的数据分析更加地充分且有说服力，打开每一个岗位信息和求职者信息界面对原未爬取的数据展开进一步补充。观察 URL 发现，各岗位信息和求职信息的 URL 仅仅是网络路径后端 id 的不同。因此，可利用招聘信息和求职信息 id 遍历访问各求职岗位和求职者信息的子页面，爬取自由文本数据。其重要代码如下（以招聘信息为例）：

```
for i in data['招聘信息 id']:
    url = 'https://www.5iai.com/api/enterprise/job/public?id={}'.format(i)
    response = requests.get(url, headers=headers)
```

（3）解析数据

利用 json 库 loads 函数对 request 模块获取的数据进行解析，将不同岗位和不同求职者关键指标的信息以字典（键：值=关键指标：对应信息）的形式存储下来，其重要代码如下（以招聘信息为例）：

```
    html = json.loads(response.text)
    for i in html["data"]["content"]:
        item = {'招聘信息 id': i["id"], '公司地址': i["enterpriseAddress"]["detailedAddres
s"].replace('\xa0', ''), 'enterpriseId': i["enterpriseAddress"]['enterpriseId'],'provinceCode': i["
enterpriseAddress"]['provinceCode'],'cityCode':i["enterpriseAddress"]['cityCode'], 'regionCode
': i["enterpriseAddress"]['regionCode'], '招聘岗位': i["positionName"], '员工数量': i["enter
priseExtInfo"]["personScope"], '最低薪资': i["minimumWage"], '最高薪资': i["maximumWa
```

```
ge"],'公司类型': i["enterpriseExtInfo"]["econKind"], '学历': i["educationalRequirements"], '岗位经验': i["exp"], '企业名称': i["enterpriseExtInfo"]["shortName"],'企业类型': i["enterpriseExtInfo"]["industry"],'职业状态':i['willNature'],'招募人数':i['count']，’招聘要求’：’i['demand']’}
            print(item)
            data.append(item)
```

（4）整理与保存数据

在运用 json 库将不同岗位和求职关键指标信息以字典形式存储的基础上，运用 panda 库中 **dataframe 函数**，以关键指标作为表头，将招聘岗位和求职者各关键指标的信息转换为数据框形式，进而用 panda 库中 **concat 函数**分别完成对各招聘岗位和各求职者关键指标信息之间的横向拼接，并将其保存到对应的 csv 文件中。

对招聘与求职爬虫数据整理保存的关键代码如下（以招聘信息为例）：

```
    df = pd.DataFrame(data)
    df1 = pd.DataFrame(range(1, len(df)+1), columns=['序号'])
    pd.concat([df1, df], axis=1).to_csv('数据/result1-1.csv', encoding='utf-8-sig', index=None)
```

三、招聘与求职信息爬取文本的预处理

3.1 文本清洗

由于泰迪内推招聘平台上不同主体在诸如求职岗位的招聘要求等以自由文本区域填写的内容上发布的格式不同，且同一主体可能每天会发布相同的求职信息或招聘信息，这会使爬取的信息非常杂乱，需要在分析前进行文本清洗。因此，本文基于所爬取出的招聘信息和求职信息文件，分别依次查找并处理其重复值、缺失值、异常值。

3.1.1 重复值查找与处理

运用 excel 中数据项中的**“删除重复项”**功能查找求职信息和招聘信息中**相同 id 且其他信息基本相同**的样本，并对重复样本进行剔除处理。重复值剔除前后样本量的变化如表 1：

表 1 重复值剔除前后样本量的变化

表格名称	剔除前样本量	剔除后样本量
招聘信息	1574	1574
求职信息	10919	8263

3.1.2 缺失值查找与处理

通过 excel 的**“筛选”功能**，发现招聘信息和求职信息的部分指标存在缺失值。出现缺失的指标及对应的确实个数如表 2 所示。

表 2 缺失值检查记录

表格名称	指标名称	缺失个数	指标名称	缺失个数	指标名称	缺失个数
招聘信息	公司地址	6	省、市编码	1347	地区编码	1347
求职信息	性别	6502	预期地区	8501	学历	8140
	技能	8180	到岗时间	8066	预期职位状况	1820

针对表 2 存在缺失值的指标，结合各个指标缺失情况，与其他指标的实际关系对缺失值分别对各个指标进行缺失值处理。

从表 2 中可知，招聘指标的缺失指标全都与地区相关，而地区信息可以从企业对应的官网查找。因此，针对招聘信息的缺失指标，可以根据企业名称所在官网查找对应的公司具体地址，进而对招聘信息的所有指标进行填充。

对于求职信息中学历、技能、性别指标存在的缺失值，考虑到若直接删除，可能会导致在双向推荐中忽略未填全信息但符合岗位需求的潜在求职者，但由于其缺失值太多，无法运用其他已有的信息对其进行填补，因此对于这个三个指标保留其缺失值。

对于求职信息中预期职位状况，预期到岗时间及预期地区中的缺失值，将其归入成“无要求”这一新类；对于工作经验指标中的缺失值，将其归入“无工作经验”这一类。

3.1.3 异常值的查找与处理

在上文运用 excel 中的“筛选功能”进行缺失值查找的过程中，发现部分指标存在异常值，具体如下：

（1）求职信息的技能指标出现如“哈哈”等**不属于技能含义范围的样本数据**。这类数据可以间接反应出填写者在填写求职信息的态度不佳，因此可对此类样本数据剔除处理。

（2）招聘信息的最低最高薪资指标量纲不统一，绝大部分均是以千元/月，元/月来计算，但少数是以元/日来计算。为了减少消除量纲对后文数据挖掘的影响，本文按照式（1）中的换算公式将按日计算的招聘岗位薪酬换算成按月计算的数据。

$$p_{month} = p_{day} \times 30$$

其中，p_{day}, p_{month}分别表示为按日、月计算招聘岗位薪酬

3.2 文本字段提取，归类与合并

为了在后文的爬取数据挖掘以及招聘鱼鳅痣双向系统的构建中能够尽可能运用更多维度的指标展开分析，在本节对无明显或有重复划分的结构化文本指标分别展开相应地字段提取与归类。

经观察，招聘信息的部分结构化文本指标出现文本种类过多，且又少数指标出现文本种类表达语义有交集的现象（如招聘信息：岗位经验要求：5-10 年，5-7 年）。直接运用原始文本来进行区分会使由于词频数太少无法从用户画像中展现该指标的特征。因此，需对这些指标进行字段提取或归类。

招聘信息和求职信息各类出现此类现象指标的归类形式如表 3 所示。

表 3 指标文本字段的归类规则

	指标	**岗位经验**	**企业类型**
招聘岗位信息	归类形式	“0”，“不限”，“经验不限”→“经验不限” “1”，“3”，“5”“1-3”“3-5”→“短期工作经验” “10”，“5-7”，“7-10”，“7 年以上”→“长期工作经验”	“互联网+其他行业”（e.g[["电子商务","互联网"])→“互联网+”
	指标	**员工数量→公司规模**	
	归类形式	“少于 50”，“50-100”，→’微型企业’ “150-500”，“500-1000”→“小型企业” “1000-1500”,“1000-5000”→“中型企业” “5000-10000”，“10000 以上”→“大型企业”	
	指标	**薪资待遇 x（最低、最高薪酬的平均值）**	**招聘人数 y**
	归类形式	0<x<1000→低收入需求水平 1000<x<3000→中低收入需求水平 3000<x<8000→中收入需求水平 x>8000→高收入需求水平	y<10→10 人之内 10<y<50→10-50 人 50<y<80→50-80 人 y>80→80 人以上

	指标	工作经验	期望行业
求职者信息	归类形式	“n 年工作经验”→“短期工作经验”（n1-5） “10 年工作经验”→“长期工作经验”	文本出现“不限”→“不限” “互联网+其他行业”（e.g["互联网","电子商务","金融"])→“互联网+” 文本未出现“互联网”→“非互联网行业”
	指标	**预期薪资 x（预期最低薪资/预期最高薪资）**	
	归类形式	0<x<1000→低收入需求水平 1000<x<5000→中低收入需求水平 5000<x<8000→中收入需求水平 x>8000→高收入需求水平	

对于地区指标，考虑到总体来说题中所爬取的城市不同地区经济水平、地理环境、政治文化等基本无显著差异，仅提取出**每个招聘信息所在企业地址对应的城市**和**求职信息预期地区对应的城市**。

3.3 文本停用词过滤

对于带有自由文本或部分结构化文本的求职信息和招聘岗位指标，由于其存在标点符号等大量的停用词，对后文文本分析以及岗位匹配度和求职满意度的计算会造成一定的干扰。因此，本文需对文本存在的停用词进行过滤。

采用 python 中的 jieba 库，对招聘岗位信息和求职者信息的停用词进行过滤，并将自由文本进行中文分词，具体步骤[4]如下：

①结合“哈工大停用词表”、“百度停用词表”、“中文停用词表”、“四川大学智能实验室停用词库”生成停用词集合，并将其储存在文本文件中；

②根据自定义词典，使用 jieba 库剔除文本集中无具体意义的词，并将自由文本进行中文分词

四、招聘与求职信息爬取文本的分析

为充分挖掘内推平台求职信息与招聘信息的求职现状和招聘现状，本文需要通过爬取的招聘岗位数据与求职信息分别绘制其对应的招聘岗位和求职者画像。

为解决以上问题，本文在对爬取数据进行文本预处理的基础上，首先对于岗位信息的自由文本运用 **TF-IDF 算法**筛选出能够代表该文本信息的关键词，并运用 **word2vec 模型**对原始自由文本读取意思相近的词语，结合《同义词词林》词典[5]以及实际情况构造同义词词典，进而通过同义词词典对自由文本意思相近的关键词进行合并，其次基于关键指标构造用户标签，通过对关键指标文本及关键词在各岗位信息与求职信息的词频统计绘制词云图，构造求职者和招聘岗位的整体用户画像。考虑到整体用户画像包含大量的矛盾信息，缺乏针对性，因此，根据原始求职和招聘数据选择相对较全且可比较的指标，在对预处理后指标数值化编码的基础上，运用 **k-means++聚类算法**分别对求职者和招聘岗位进行划分，进而针对各类求职者和招聘岗位绘制更加精确的用户画像。

4.1 相关原理介绍

4.1.1 word2vec 模型

word2Vec 模型[6-7]是一种基于神经网络的自然语言处理技术，它通过将单词嵌入到一个高维向量空间中，将自然语言文本转换为数学形式，进而完成语义分析、情感分析、文本分类等任务。其中，文本中的同义词查找是 Word2Vec 模型的一个常见应用。

word2Vec 模型有两种主要的训练方式：CBOW 和 Skip-gram。CBOW 模型尝试从上下文中的单词预测当前单词，而 Skip-gram 模型尝试从当前单词预测上下文中的单词。这两种模型都使用神经网络来训练，最终得到一个词向量矩阵，其中每一行代表一个单词的向量表示。考虑到相比于 Skip-gram 模型，CBOW 模型运行速度较快，且在训练时能够充分运用上下文的信息，本文主要运用 CBOW 模型对文本进行训练。

在经 CBOW 模型训练得到单词向量后，利用向量空间中的距离来计算单词之间的相似度，进而判断两个单词是否语义相似。如果两个单词的向量非常接近，则它们在语义上很可能相似，可以认为是同义词。

4.1.2 TF-IDF 算法

TF-IDF[8-10]算法又称“词频－逆文档频次算法”是一种基于统计的计算方法，常用于评价在一个文档集中一个词对某份文档的重要程度。TF-IDF 算法的主要思想是：如果某个词在一篇文章中出现的频率高，在其它文章中出现的频率低，则认为该词在该文章具有代表性。

TF-IDF 的计算公式如下：

$$W = \frac{n_{i,j}}{\sum n_{i,j}} \times \lg \frac{|D|}{|1 + \{j: t_I \in d_j\}|}$$

其中，$n_{i,j}$为词语t_I在招聘信息/求职者信息d_j中出现的次数，$\sum n_{i,j}$为招聘信息/求职者信息d_j中出现的次数；$|D|$为招聘信息/求职者信息总数；$\{j: t_I \in d_j\}$为包含词语t_I的招聘信息/求职者信息数。

4.1.3 聚类算法

（1）算法介绍

作为一种无监督学习方法，聚类分析[11]是指在未知分类类别与分类数的前提下根据数据对象之间的相似性对大数据集展开分组分类的过程，而 k-means 聚类[12]作为聚类分析中基于划分方法的算法，一般使用欧式距离这一衡量数据之间相似度之间指标对数据进行划分。

相比于其他聚类算法，k-means 聚类算法[13]简单高效且对聚类结果拥有较好的解释度，在用户群体的划分得到广泛地应用。但是，由于 k-means 算法选取聚类中心的方式具有随机性，其聚类效果在遇到聚类中心过于密集的情况时会显得不太稳定，从而容易陷入局部最优的情况，导致聚类结果偏差过大。基于 k-means 聚类算法对初始聚类中心的依赖性，以及随机选取初始聚类中心对算法所造成不利影响的考虑，本文提出 k-means++聚类算法对 k-mean 均值算法中的聚类初始中心的确定作出相应的改进。

k-means++算法[14]是一种基于质心的聚类算法，它在 k-means 算法的基础上优化了初始聚类中心的选择方式，使初始聚类中心相互间距离尽可能得远。其具体的算法流程如图 4 所示：

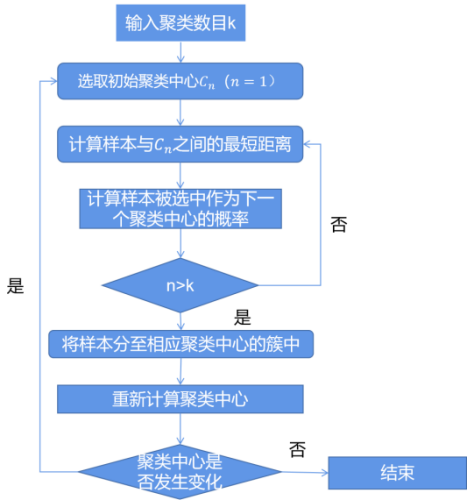


图 4　k-means++聚类算法的聚类流程图

（2）算法评价指标

聚类的目标是通过选取一种相似性度量方式，将数据进行划分，使得同一类

簇的数据尽可能相似[15]。为了验证聚类在题中招聘岗位和求职者的划分是否具有合理性，引入聚类的相关评价指标对聚类算法进行评价。

常见的聚类评价指标有内部有效性指标和外部有效性指标，考虑到在对招聘岗位和求职者划分未有一个明确的划分标准，外部有效性指标不太适用，本文主要运用内部有效性指标对聚类效果进行评价，包括**轮廓系数**和 **DB 系数**。

- **轮廓系数(SC)**

轮廓系数(SC)是用于衡量聚类结果准确度和分离度的指标，其值的范围在[-1,1]之间。一般来说，轮廓系数越大，其聚类效果越好。轮廓系数(SC)的计算公式如下：

$$SC = \frac{\sum_{i=1}^{n} \frac{b(x_i) - a(x_i)}{\max[a(x_i), b(x_i)]}}{n}$$

其中，n 为样本的个数，$b(x_i)$为样本与其他簇样本平均距离的最小值，$a(x_i)$为样本与同簇其他样本的平均距离。

- **DB 系数**

DB 系数是用于评估聚类结果质量的一个指标，其值越小，代表聚类效果越好。DB 系数的计算公式如下；

$$DB = \frac{1}{k}\sum_{i=1}^{n} \max_{j=1}\left(\frac{\overline{C_i} + \overline{C_j}}{||w_i - w_j||}\right)$$

其中，$\overline{C_i}$，$\overline{C_j}$分别表示第 i 类和第 j 类的类间平均距离，$||w_i - w_j||$表示第 i 类和第 j 类两个类之间的距离。

4.2 自由文本指标的文本处理

4.2.1 同义词词典的构造

首先，运用 **gensim 库**加载并使用 word2vec 模型对预处理后的岗位信息自由文本数据展开训练，进而计算出每个词中的词向量，关键代码如下：

```
import gensim
model = gensim.models.KeyedVectors.load_word2vec_format('招聘信息爬取数据（预处理）.xlsx', binary=True)
```

在此基础上，设定最大距离阈值，得出初步同义词词典如表 4：

表 4 初步的同义词词典(共 86 条)

词语	同义词
数据采集	数据收集，数据查找
...	...

基于 word2vec 模型得出的 86 对同义词信息，结合《同义林词典》查找以及

实际情况对 63 对同义词展开核查，最终保留 63 对有效同义词信息。

4.2.2 对招聘信息自由文本的关键词提取

运用 TF-IDF 算法对各自由文本各词语在该文本的重要程度进行计算，根据计算得出 TF-IDF 值从大到小**保留前 25%的关键词**代表此本文的重要指标，用于后面该指标的词频统计。考虑到文本间原词语若运用同义词表达时会使原词语的 TF-IDF 值会变少，在计算 TF-IDF 前基于 Word2vec 构造的同义词词典将同义词统一词语表达（如“数据采集”“数据收集”“数据查找”这一对同义词统一表达为数据收集），进而使文本的关键词提取更加准确、科学。

部分样本自由文本提取出的关键词如表 5 所示：

表 5 部分样本自由文本提取出的关键词

样本 id	关键词 (TF-IDF 值)
1526028377780256769	游戏（0.61），美术（0.32），系统（0.18），交互（0.14），...
1524956737138982913	医疗器械（0.80），生物医学（0.14），全球（0.10），...
.....	.....

4.3 招聘与求职信息画像构建与分析

4.3.1 招聘与求职信息画像标签体系的建立

用户标签是指在绘制用户画像时为用户赋予的特定属性或标签，在用户画像的构建中起到至关重要的作用[16]。在招聘与求职信息的用户画像构建，构造良好的用户画像评价体系会使得用户画像变得更加清晰，进而使招聘岗位负责人员和求职者通过用户画像更加了解彼此的信息。因此本文根据求职者信息和招聘岗位信息的结构文本指标以及招聘岗位的自由文本指标含义，将指标归入相应的标签层中，从而分别构建出相应招聘岗位和求职者用户画像的标签体系。

表 6 招聘岗位用户画像标签体系

标签层	指标层
岗位性质	公司类型、企业类型、企业工作地点、员工数量
招聘需求	招聘岗位名称、招聘人数
求职者待遇	薪资待遇（最低薪资，最高薪资）
求职者要求	学历要求、职业状态要求、岗位工作经验要求、招聘要求

表 7 求职者用户画像标签体系

标签层	指标层
个人情况	性别、技能，工作经验、到岗时间
求职需求	预期岗位、预期岗位个数、预期薪资需求、申请职位状况、期望行业、期望工作地区

4.3.2 求职者和岗位信息的整体用户画像绘制

通过对经文本处理后求职者信息和招聘岗位信息各指标文本的词频统计，以词云图的形式分别绘制求职者和招聘岗位信息的整体画像。

内推平台招聘岗位的整体画像如图 5 所示：

图 5 内推平台招聘岗位整体画像

从图 5 可以看出，泰迪内推平台这招聘岗位大多数在广州市且与互联网相关，其公司类型以合资企业、民营企业为主。平台的所有岗位大多数注重学生数据分析和数据挖掘的能力以及本科以上的学历要求，但是大部分岗位对工作经验的要求相对比较低，同时也会提供中高水平的薪资待遇以及一些福利和补贴，以吸引更多优秀的求职者。

图 6 内推平台求职者整体画像

通过图 6 可以看出，从求职者的个人情况来看，该平台的求职者用户男女比例均衡，且大部分均具有数据分析、数据挖掘等大数据行业必须掌握的基本能力，但是这些求职者的工作经验相对较少，这可能会在一定程度上影响他们在求职市场的竞争力。

通过对该平台用户的求职需求进行分析，可以发现大部分求职者希望在一线城市，如广州、深圳、北京、上海等地工作。此外，求职者期望获得中等水平的薪资，并从事数据分析与数据挖掘相关的工作。这表明，这些求职者具有一定的专业背景和技能，但他们需要更多的实践机会和经验积累，以提高他们在求职市场的竞争力。

综合来看，平台中提供的求职者与招聘岗位均与大数据行业有关。其中，求职者大多数具有一定的大数据处理技能但缺乏工作经验，期望通过全职或实习工作获得中收入水平的薪酬，而招聘岗位在对有相关技能和实战经验的大数据人才有需求的同时，普遍提供招聘者高收入水平的薪酬，说明企业招聘岗位和求职者在某些指标高度匹配。因此，泰迪内推平台可以为企业与求职者搭建了一个高效的双向匹配平台，通过对信息的输入得出对应符合要求的招聘信息和求职者的信息，从而促进了企业和求职者之间的互动与合作，进而提升了整个大数据行业的竞争力。

4.4 招聘岗位与求职信息的聚类分析

4.4.1 聚类指标的选取及数据预处理

在招聘平台中，每个招聘岗位和求职者都具有独特的特点和需求，仅仅通过整体画像来描述所有用户的信息和行为难免存在一定的偏差和误差。这种标签上文本的矛盾会影响到平台的服务质量，无法满足用户的个性化需求。为了更好地满足用户的需求和提供更加精准的个性化服务，招聘平台可以选取适当的聚类指标对所有招聘岗位和求职者信息进行聚类。通过聚类的方法，可以将相似的招聘岗位和求职者归为一类，并为每一类用户群体绘制更加精准的画像，进而平台在针对每一类用户群体提供更加个性化的服务，帮助他们更好地匹配职位或求职者，提高招聘的成功率和效率的勇士，也可以帮助平台更好地了解用户的需求和特点，优化平台的服务和经营策略，提高平台的竞争力和市场份额。

因此，本文选取相应的聚类指标对招聘岗位和求职者信息进行聚类划分。考虑到一方面从指标含义上看部分指标之间具有一定的相关性，另一方面预处理后经缺失值填充后的指标文本可能会有一定的偏差，在本文综合考虑最终确定的聚类指标如表 8 所示。

表 8 招聘信息和求职信息选取的聚类指标

	聚类指标
招聘岗位	公司规模、薪酬福利、岗位类型、招聘人数、职业状态、工作经验要求、公司类型
求职者信息	预期岗位个数、工作经验、预期职业状态、期望行业、预期收入

由于文本类数据在聚类分析中不能直接进行数值计算，需要结合各指标文本的实际含义对各指标编码从而使其数值化。各指标的数值化形式如表 9，表 10 所示.

表 9 招聘信息聚类指标编码表

聚类指标	编码形式
公司规模	“微型企业”→“0”，“小型企业”→“1”，“中型企业”→“2”，“大型企业”→“3”
薪酬福利	“低收入”→“0”，“中低收入”→“1”“中收入”→“2”，“高收入”→“3”
岗位类型	“非互联网行业”→“0”，“互联网行业”→“1”“互联网+行业”→“2”，
招聘人数	“10 人之内”→“0”，“10-50 人”→“1”，“50-100 人”→“2”，“100 人以上”→“3”
职业状态	“实习”→“0”，“全职”→“1”

工作经验要求	“不限”→“0”，“短期工作经验”→“1”，“长期工作经验”→“2”
公司类型	“国企”→“0”，“合资”→“1”，“民营公司”→“2” “上市公司”→“3”，“私营”→“4”，外资→“5”

表 10 求职信息聚类指标编码表

聚类指标	编码形式
预期收入	“低收入”→“0”，“中低收入”→“1” “中收入”→“2”，“高收入”→“3”
期望行业类型	“非互联网行业”→“0”，“互联网行业”→“1” “互联网+行业”→“2”，“不限”→“3”
预期职业状态	“实习”→“0”，“全职”→“1”，“均可接受”→“2”
工作经验	“不限”→“0”，“短期工作经验”→“1”，“长期工作经验”→“2”

在对聚类进行编码的基础上，考虑到各类聚类指标具有量纲差异，本文对各类聚类指标进行归一化处理。

4.4.2 基于肘部法对聚类个数的确定

聚类个数 k 的确定[15]，是进行聚类划分的首要前提，k 的大小过多或过低均会降低聚类的效果。常见的聚类个数确定的方法有经验确定法，肘部法，考虑到经验确定法对聚类个数的确定具有很强的主观性，容易使聚类结果不太稳定，在本文中运用肘部法确定聚类个数。

通过计算招聘岗位和求职者不同聚类个数的误差平方和 SSE,绘制出招聘岗位信息和求职者信息的肘部图如图 7 所示。

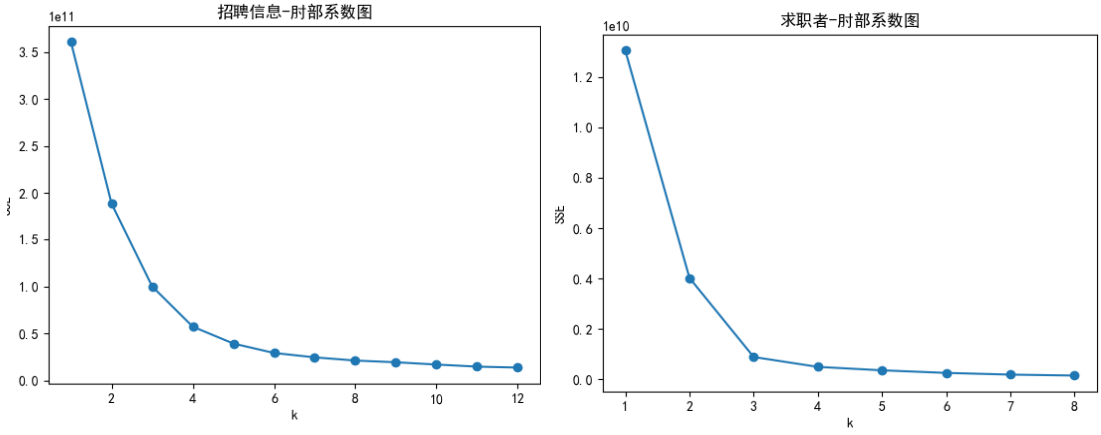


图 7 招聘岗位和求职岗位的肘部系数图

通过图 7 可以看出，当招聘岗位信息 k=4 以及求职者信息 k=3 时，继续增

加聚类个数使 SSE 的变化会趋于平缓。因此，招聘岗位和求职者的最佳划分种类分别为 4 类，3 类。

4.4.3 聚类算法的应用与评价

基于肘部法确定的最佳聚类个数，运用 k-means 以及 k-mean++算法对招聘岗位和求职者信息展开聚类。招聘岗位和求职者在两种聚类算法下的评价指标如表 11 所示：

表 11 招聘岗位和求职者在两种聚类算法的评价指标

聚类对象	聚类算法	轮廓系数	DB
招聘岗位	k-means 聚类算法	0.49394	0.74992
	k-means++聚类算法	0.6438	0.5323
求职者	k-means 聚类算法	0.6966	0.4876
	k-means++聚类算法	0.7424	0.4281

通过表 11 可以看出：

（1）无论 k-means 还是其改进后的聚类算法，其轮廓系数均大于 0，说明 k-means 聚类算法在用户群体的划分具有良好的效果。

（2）相比于 k-means 聚类算法，运用 k-means++算法对求职者和招聘岗位聚类轮廓系数与 DB 系数表现出的聚类效果均比 k-mean 算法表现良好，这进一步验证本文对聚类算法改进的合理性和有效性。

4.5 基于聚类分析对用户画像的扩充

4.5.1 不同种类求职者和招聘岗位之间的可视化分析

在运用 K-means++算法对求职者信息和招聘岗位进行聚类划分的基础上，通过对不同聚类指标在各类型文本词频的可视化分析直观探究不同种类求职者之间和不同种类招聘岗位之间的差异，进而为各类型进行命名。

通过对不同类别岗位聚类指标的各文本出现词频展开统计，分别绘制在各类招聘岗位各类聚类指标文本词频的百分比堆积柱状图。

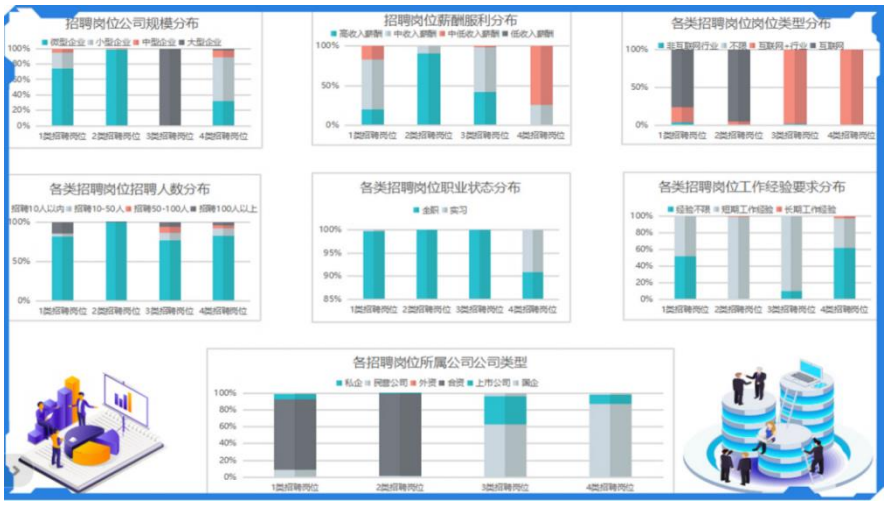


图 8 不同类别招聘岗位聚类指标差异的可视化大屏

通过图 8 可以看出，各类别的招聘人数普遍较少，大部分都是全职行业，说明大部分企业均想通过平台找到能够长期为企业服务的大数据人才，来补强公司在市场上中的竞争力。

通过可视化大屏对各类招聘岗位展开差异性分析，可以得出：在岗位公司规模中，第 3 类招聘岗位的公司规模相对较大，其余三类均为小中型规模企业；在薪酬福利和工作经验要求中，第 2 类招聘岗位和第 3 类招聘岗位对工作经验有一定的要求，但提供以中高收入的岗位福利，而第 1 类和第 4 类中对工作经验多半没有需求，相对来说岗位福利上比 2 类和 3 类要少；在岗位类型分布和所属类型中，第 1 类和第 2 类企业主要为合资，互联网企业，而第 3 类和第 4 类主要为国企或上市，互联网+行业。

基于上述差异性分析，可将 4 类招聘岗位分别命名为：**策马奔腾型新生互联网合资企业，经验丰富型互联网合资明珠企业，智慧前沿型国有互联网+翘楚企业，持续探索型国有互联网+革新企业**。

通过对不同类别求职者聚类指标的各文本出现词频展开统计，分别绘制在各类求职者早各类聚类指标文本词频的百分比堆积柱状图。

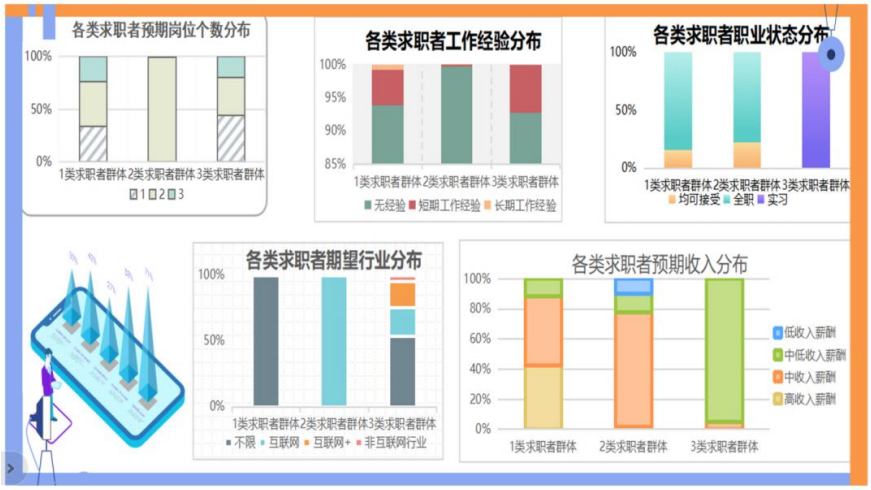


图 9 不同类别求职者聚类指标差异的可视化大屏

通过对图 9 的可视化分析可以得出：1 类求职者群体有一部分人在工作方面有一定的经验，对期望行业的种类普遍接受度较高，但是这类需求者对预期收入的要求较高，力图找到一个高薪，稳定且可长期工作的全职岗位；2 类求职者群体基本没有工作经验，对预期收入要求相对没有太高，在期望行业中有一定的目标和方向，力图找到一个理想，稳定的全职岗位；3 类求职者群体普遍均是实习生，基本没有工作经验，有一部分人在期望行业也有目标，力图通过平台找到一个容易入门，适合自己的工作。

基于上述分析，可将求职者群体分为**“高薪求职者”，“目标求职者”，“入门求职者”三类。**

4.5.2 各种类求职者和招聘岗位的画像绘制

在对求职者和招聘岗位进行划分和命名的基础上，通过对经文本处理后各类求职者信息和各类招聘岗位信息所有指标文本的词频统计，以词云图的形式分别绘制各类求职者和各类招聘岗位信息的画像。

各类招聘岗位的画像如图 10 所示：

策马奔腾型新生互联网合资企业岗位

经验丰富型互联网合资明珠企业岗位

智慧前沿型国有互联网+翘楚企业岗位

持续探索型国有互联网+革新企业岗位

图 10 各类招聘岗位画像

各类求职岗位的画像如图 11 所示：

高薪求职者　　**目标求职者**　　**入门求职者**

图 11 各类求职者画像

从招聘者和求职者画像可以看出，在招聘岗位和求职者满足学历，工作经验等基本要求双向符合的前提下，策马奔腾型新生互联网合资企业岗位在招聘岗位时对岗位技术以及相关技能掌握要求不高，但对求职者学习态度和工作态度十分注重，力求通过对求职者的不断培训培养出公司独特的高技术人才。由于该类型企业大多处于初创阶段，相应地职位可能会多样会，涵盖了金融、算法设计等各个大数据行业，为求职者提供广阔的岗位选择空间，与目标求职者的就业需求高度匹配；而持续探索型国有互联网革新企业岗位大多数对求职者没有的要求较低，会给实习生一个良好的工作平台，让其接受到更多的新技术和业务模型，是入门求职者的不二之选。

在高薪求职者这一类群体当中，经验丰富型互联网合资明珠企业岗位和智慧前沿型国有互联网翘楚企业岗位是两个相对较好的选择，但是两者不同的是，经验丰富性互联网合资明珠由于规模较少，招聘人数十分有限，企业在招聘岗位时更加注重求职者的综合能力，力求通过全方面发展人才的查找与培养实现公司的全方面发展，而由于智慧前沿型国有互联网翘楚企业岗位企业规模较大，涉及的行业、业务范围也比较广泛，所以该类型企业在招聘时更加看重应聘者的技术培养，力求通过大规模的专业人才做到公司全方面的发展。因此，高薪求职者可以适当根据自己的个人情况综合权衡利弊，选择两者中较为适合自己的岗位。

五、构建岗位匹配度和求职者满意度模型

通过对招聘岗位和求职者的聚类结果初步对各类求职者和对应招聘岗位的双向匹配关系分析，可根据招聘岗位或求职者的所属类别分别找到对应求职者或招聘信息类别的相关信息，进而实现一个求职者和岗位信息的双向匹配。显然这种招聘系统简单高效，为岗位匹配度和求职者满意度提供了一个良好的思路，但是由于在聚类后同一类样本之间各个文本指标也会存在微小差异，此方法构造出的双向招聘系统缺乏全面性，难以满足招聘岗位和求职者的个性化需求，而量化衡量岗位对各求职者的匹配程度以及各求职者对岗位的满意度相关个性化指标恰好可以很好地解决无法衡量同一类样本之间差异的问题。因此，在下文中引入“岗位匹配度”和“求职者满意度”两个量化指标，通过综合评价方法对两个指标值展开计算，为下一章求职者与岗位的个性化双向推荐作铺垫。

在前文部分已经爬取并处理了招聘企业与求职者相关的文本信息，包括半结构化文本和自由文本。半结构化文本提供了招聘企业与求职者的较为完整且具有固定结构的相关指标文本信息，可以根据对应指标运用评估模型来确定招聘信息的匹配度以及求职者的满意度。而自由文本提供的信息并没有固定的文本结构，尽管不能直接使用常规的评价模型来评估满意度和匹配度，但其涉及的语义内容、关键词特征相对来说更加丰富（比如说专业技能要求等），因此在评估满意度和匹配度不能忽略自由文本的信息。为了综合考虑半结构化文本与自由文本的信息，运用向量空间模型计算自由文本中招聘信息与求职者信息的相似度，以相似度指标来刻画招聘企业与求职者的匹配程度和求职者对招聘企业的满意程度，然后利用相似度对基于半结构化文本得出的匹配度和满意度进行修正，最终得到岗位匹配度和求职者满意度的结果。

5.1 基于半结构化文本构建匹配度与满意度评估模型

在这部分，主要通过五个步骤来构建基于半结构化文本的匹配度与满意度评估模型，分别是“基于最低要求对样本的筛选”“定义评价指标”“构建评价体系”、“组合赋权”、“构建 topsis-fuzzy 模型”。

5.1.1 基于最低要求对样本的筛选

根据题目要求，预先设置岗位最低要求和求职者最低要求的标准，对于不满足岗位最低要求的求职者，将其岗位匹配度定义为 0；对于不满足求职者最低要求的岗位，将其求职者满意度定义为 0，在此基础上将筛去岗位对应匹配度为 0

的求职者和求职者对应满意度为 0 的招聘信息，保留匹配度非 0 的求职者和满意度非 0 的招聘信息。

通过查询相关资料并联系实际情况，确定了招聘岗位的最低要求和求职者的最低要求分别如表 12 和表 13 所示。

表 12 招聘岗位的最低要求筛选原则

判断指标	具体标准
招聘岗位	求职者“预期岗位”应与企业招聘信息的“招聘岗位”有交集。
薪资	求职者的“预期最低薪资”应不大于企业招聘信息的“最高薪资”。
工作经验	求职者的“工作经验”年数应不小于企业招聘信息中要求的“工作经验”最低年数。
职位状况	求职者的“职位状况”应与企业招聘信息要求的“职业状态”相符合
行业类型	求职者的“期望行业”应与企业招聘信息的“行业类型”有交集，特别说明，若求职者的“期望行业”是“不限行业”，那么该求职者与所有招聘信息的“行业类型”有交集。
学历	对于填写学历信息的求职者，其学历应满足招聘信息学历的最低要求，否则求职者的岗位匹配度为 0；对于没有填写学历信息的求职者，应满足上述所有招聘岗位、薪资、工作经验、职业状况、行业类型的最低要求。

基于表 12 的筛选原则，对各企业企业匹配度为 0 的求职者进行初步筛选。若求职者 i 不满足 j 岗位最低要求的任意一个，那么对于企业 j 来说，求职者 i 的岗位匹配度为 0。

表 13 求职者岗位的最低要求筛选原则

判断指标	具体标准
预期工作地级市	企业公司地址所在的地级市应与求职者“预期工作地级市”相同。
招聘岗位	企业招聘信息“预期岗位”应与求职者的“招聘岗位”有交集。
薪资	求职者的“预期最低薪资”应不大于企业招聘信息的“最高薪资”。
职位状况	求职者的“职位状况”应与企业招聘信息要求的“职业状态”相同。
行业	求职者的“期望行业”应与企业招聘信息的“行业类型”有交集，特别说明，若求职者的“期望行业”是“不限行业”，那么该求职者与所有招聘信息的“行业类型”有交集。

基于表 13 的最低要求筛选规则，对各求职者求职满意度为 0 的岗位进行初步筛选。若企业招聘信息 j 不满足求职者 i 最低要求的任意一个，那么对于求职者 i 来说，j 的求职者满意度为 0。

5.1.2 评价指标的定义

从实际生活的角度出发，在众多岗位匹配度非 0 的求职者中，招聘企业会更偏好那些薪资低、工作经验丰富、学历高的求职者，同理在众多满意度非 0 的招聘信息中，求职会更偏好那些薪资高的企业。

基于以上考虑，本文选取以下几个指标作为求职满意度以及岗位匹配度的评价指标。

- $\boldsymbol{YG_i/ZK_i/HY_i}$（i=1 或 2）：这三个指标分别代表企业和求职者预期岗位/职业状况/行业类型的匹配结果，其中 i=1 代表企业，i=2 代表求职者，对于岗位匹配度非 0 的求职者或满意度非 0 的招聘企业有：$YG_i = ZK_i = HY_i = 1$
- **企业的最低薪资落差**$(\Delta \boldsymbol{D_q})$:该指标等于匹配度非 0 的求职者的“预期最低薪资”与企业招聘信息的“最低薪资”之差，对**企业**而言，ΔD_q越小越好。
- **企业的最高薪资落差**$(\Delta \boldsymbol{G_q})$：该指标等于匹配度非 0 的求职者的“预期最高薪资”与企业招聘信息的“最高薪资”之差，对**企业**而言，ΔG_q越小越好。
- **工作经验附加值**$(\Delta \boldsymbol{Y_q})$：该指标等于匹配度非 0 的求职者的“工作经验”年数与企业招聘信息的“工作经验”年数之差，对**企业**而言，ΔY_q越大越好。
- **学历附加值**$(\Delta \boldsymbol{L_q})$：该指标等于匹配度非 0 的求职者的“学历”编码数值与企业招聘信息的“学历”编码数值之差，对**企业**而言，ΔL_q越大越好。
- **求职者的最低薪资落差**$(\Delta \boldsymbol{D_z})$:该指标等于求职者的“预期最低薪资”与满意度非 0 的企业招聘信息的“最低薪资”之差，对**求职者**而言，ΔD_z越小越好。
- **求职者的最高薪资落差**$(\Delta \boldsymbol{G_z})$:该指标等于求职者的“预期最高薪资”与满意度非 0 的企业招聘信息的“最高薪资”之差，对**求职者**而言，ΔD_z越小越好

5.1.3 评价体系的构建

在构建岗位匹配度评价体系时本文主要考虑两类指标，分别是定性指标和定量指标。定量指标包括ΔD_q、ΔG_q、ΔY_q、ΔL_q，在众多岗位匹配度非 0 的求职者

中，招聘企业往往会更偏好那些薪资低、工作经验丰富、学历高的求职者，定量指标便是衡量不同求职者的岗位匹配度差异之处的关键指标；定性指标主要用于刻画基础的岗位匹配度，包括YG_i、ZK_i、HY_i。根据前述定义，岗位匹配度非 0 的求职者的指标YG_i、ZK_i、HY_i均为常数 1，故称其为定性指标。由于到企业对岗位匹配度非 0 的求职者都会有一个基础的偏好程度（常数），忽略定性指标会使岗位匹配度损失该方面的信息。为使岗位匹配度的构建更加科学，全面，本文将定量指标和定性指标均考虑到评价体系的构造当中，岗位匹配度的评价体系如图 12。

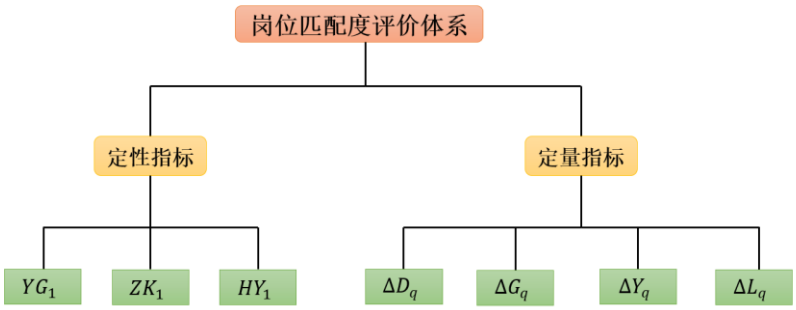
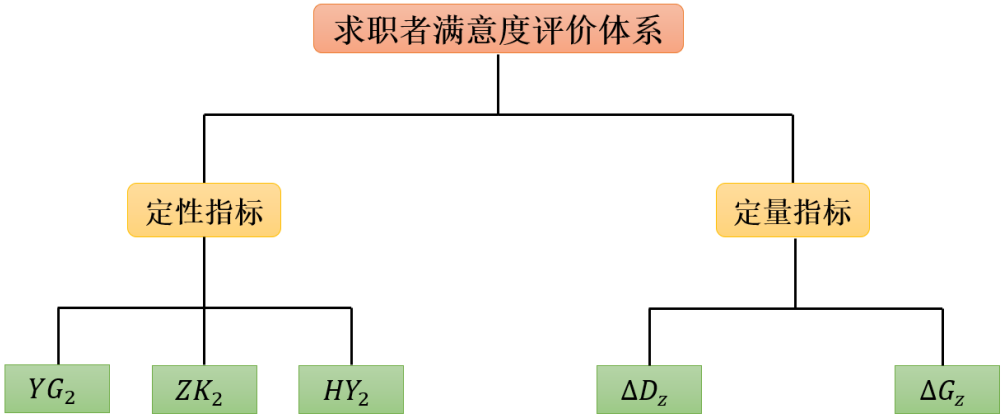


图 12 岗位匹配度评估体系

同理，可以得出对应求职匹配度的评价体系如图 13。



：

图 13 求职者满意度评估体系

5.1.4 指标组合赋权

- **组合赋权原理**

AHP（Analytic Hierarchy Process）[17]和熵权法[18]是常用的指标赋权方法，两者均被广泛地应用到解决评价类问题中。AHP 方法考虑了不同指标之间的相对

重要性，可以根据指标的重要性对不同指标进行赋权，提高综合评价结果的准确性；但是 AHP 方法往往需要专家参与，专家主观因素可能影响到赋权结果的准确性。熵权法不需要专家参与，可以直接使用指标数据进行赋权，具有较高的客观性和准确性，但是熵权法只考虑了指标之间的差异性，没有考虑指标之间的关系和相互依存性，可能导致赋权结果的不准确性。

因此，为了能够充分利用两种方法的优点，同时避免各自可能存在的缺陷和局限性，提高指标赋权的准确性和可信度，本文将利用 AHP 和熵权法进行组合赋权对评价体系中的指标进行组合赋权。由于两个评价体系确定指标权重的思路基本相同，本文主要以确定岗位匹配度评价指标的权重进行分析。

- **基于 AHP 的指标赋权**

AHP 层次分析法给岗位匹配度评价指标赋权的基本步骤如下：

（1）建立判断矩阵

本文站在招聘企业的角度，根据实际经验进行决策，建立指标之间的判断矩阵，即两两比较各层次因素之间的相对重要性。对于每个判断矩阵进行归一化处理，即将其每行元素的和除以行数，得到一个单位矩阵如表 14。

表 14　AHP 单位矩阵

指标	YG_i	ZK_i	HY_i	ΔD_q	ΔG_q	ΔY_q	ΔL_q
YG_1	1	1	1	0.25	0.5	0.2	0.222
ZK_1	1	1	1	0.25	0.5	0.2	0.222
HY_1	1	1	1	0.25	0.5	0.2	0.222
ΔD_q	4	4	4	1	2.5	0.667	0.8
ΔG_q	2	2	2	0.4	1	0.556	0.625
ΔY_q	5	5	5	1.5	1.8	1	1.25
ΔL_q	4.5	4.5	4.5	1.25	1.6	0.8	1

（2）计算指标权重。

根据判断矩阵计算各层次因素的权重，主要的方法包括特征向量法、最大特征值法和一致性指标法等，在这里本文采用方根法求取特征向量，具体步骤如下：

第一步：计算判断矩阵每行元素之积的 n 次方根，如下式：

$$M_i = \sqrt[n]{\prod_{j=1}^{n} a_{ij}}$$

第二步：将M_i进行归一化处理，如下式:

$$W_i = \frac{M_i}{\sum_{i=1}^{n} M_i}$$

第三步：计算判断矩阵的最大特征根

$$\lambda_{\max}=\frac{1}{n}\sum_{i=1}^{n}\frac{(AW)_i}{W_i}$$

运用 python 编程，本文计算得出岗位匹配度评价体系中的 7 个指标的权重结果，具体见表 15。

表 15 最大特根法结果

指标	特征向量	权重(%)	最大特征根	CI 值
$\boldsymbol{YG_1}$	0.476	5.449		
$\boldsymbol{ZK_1}$	0.476	5.449		
$\boldsymbol{HY_1}$	0.476	5.449		
$\boldsymbol{\Delta D_q}$	1.887	21.595	7.041	0.007
$\boldsymbol{\Delta G_q}$	1.015	11.615		
$\boldsymbol{\Delta Y_q}$	2.372	27.133		
$\boldsymbol{\Delta L_q}$	2.038	23.312		

（3）一致性检验。

使用一致性检验判断所构建的判断矩阵是否存在逻辑错误，若不通过，则需重新构建判断矩阵。一致性检验的具体步骤如下：

第一步：首先需要计算一致性指标 CI，其公式为：

$$CI=\frac{\lambda_{\max}-n}{n-1}$$

第二步：计算平均随机一致性指标 RI，具体的 RI 值可取自 AHP 一般矩阵所应有的随机一致性指标表。

第三步：计算一致性比率 CR，公式为：

$$CR=\frac{CI}{RI}$$

若 CR 指标值小于 0.1，则通过了一致性检验，表示判断矩阵一致性较好，即权重确定方法是合理的。

表 16 一致检验结果

最大特征根	CI 值	RI 值	CR 值
7.041	0.007	1.341	0.005

从表 16 中一致性检验的计算结果显示，最大特征根为 7.041，根据 RI 表查到对应的 RI 值为 1.341，根据公式可以得出一致性比率 CR 等于 0.005<0.1，故通过一致性检验，说明构建的判断矩阵是合理的，岗位匹配度评价体系各项指标的 AHP 权重结果见表 17。

表 17 岗位匹配度指标的 AHP 赋权结果

指标	YG_1	ZK_1	HY_1	ΔD_q	ΔG_q	ΔY_q	ΔL_q
权重(%)	5.449	5.449	5.449	21.595	11.615	27.133	23.312

同理可以计算得出求职者满意度的 AHP 权重结果，具体结果见表 18。

表 18 求职者满意度 AHP 赋权结果

指标	YG_2	ZK_2	HY_2	ΔD_z	ΔG_z
权重(%)	9.524	9.524	9.524	47.619	23.81

- **基于熵权法的指标赋权**

熵权法是一种基于信息熵理论的赋权方法，它通过计算各指标的信息熵和信息熵权重，得出各指标的权重，可以处理指标之间存在非线性关系的情况，且不需要预先设定权重的范围和先验知识。其具体步骤如下：

（1）**构造指标矩阵：**将所有待评价指标按照其对决策目标的影响程度进行量化，并将其构成一个矩阵，其中每一行代表一个指标，每一列代表一个样本，即被评价对象。然后将指标矩阵中的每个元素归一化到[0,1]区间内，使得各指标在计算信息熵时具有可比性。

（2）计算信息熵：对于每个指标，计算其信息熵，公式为：

$$E_j = -\frac{1}{\ln(n)}\sum_{i=1}^{n} p_{ij}\ln(p_{ij})$$

其中 n 为样本数量，p_{ij}为第 j 指标在第 i 个样本中所占比重。

（3）计算信息熵权重：对于每个指标，计算其信息熵权重，公式为：

$$w_j = \frac{1-E_j}{n-\sum_{i=1}^{n}(1-E_i)}$$

（4）对信息熵权重进行归一化：将信息熵权重进行归一化，使得各指标权重之和为 1，归一化后各指标的权重即为熵权法确定的权重。

运用 SPSS 软件，计算得出岗位匹配度和求职满意度评价体系基于熵权法指标的赋权结果分别如表 19 和表 20 所示。

表 19 岗位匹配度指标的熵权法赋权结果

指标	YG_1	ZK_1	HY_1	ΔD_q	ΔG_q	ΔY_q	ΔL_q
权重(%)	0	0	0	0.702	1.228	97.794	0.275

表 20 求职满意度指标的熵权法赋权结果

指标	YG_2	ZK_2	HY_2	ΔD_z	ΔG_z
权重(%)	0	0	0	36.385	63.615

- **组合赋权方法对各指标权重的最终确定**

设置了权重系数α，对各指标 AHP 权重和熵权法权重进行加权平均，进而得到组合权重。各个指标的组合权重结果表达式为：

$$w_i = \alpha w_{i,AHP} + (1-\alpha) w_{i,\text{熵权法}} \qquad i = 1,2,\ldots,7$$

如果 AHP 权重更重要，则可以将 AHP 权重赋予更大的权重系数；如果熵权法权重更重要，则可以将熵权法权重赋予更大的权重系数。

考虑到 AHP 是站在招聘企业的角度并依据经验进行赋权的方法，更能体现招聘企业对岗位匹配度的要求，因此本文决定给 AHP 权重赋予更大的权重系数，设置$\alpha = 0.8$，那么可计算得出岗位匹配度评价体系中各个指标的组合权重结果，见表 21.

表 21 岗位匹配度评价体系中各个指标的组合权重

指标	YG_1	ZK_1	HY_1	ΔD_q	ΔG_q	ΔY_q	ΔL_q
权重(%)	4.3592	4.3592	4.3592	17.4164	9.5376	41.2652	18.7032

类似地，设置求职满意度评价体系中的$\alpha = 0.5$，同样可以得到求职者满意度评价体系中各个指标的组合权重结果，见表 22.

表 22 求职满意度评价体系中各个指标的组合权重

指标	YG_2	ZK_2	HY_2	ΔD_z	ΔG_z
权重(%)	4.762	4.762	4.762	42.002	43.712

5.1.5 构建 topsis-fuzzy 模型

TOPSIS 模型是一种适用于连续型指标的多指标决策方法[19]，它的基本思想是通过计算各个评价对象与最优和最劣解之间的距离，然后根据距离计算出每个评价对象与最优解的相对接近度。但评价体系中存在离散型指标，由于离散型指标并不具备可比性和可加性，因此无法直接使用距离计算方法来衡量它们之间的差异和相似度。直接将离散型指标作为连续型指标来处理会忽略它们的本质差异，导致评价结果不准确。

模糊理论是一种处理不确定性和模糊性问题的数学工具[20]，它能够对含有模糊或不确定性信息的数据进行描述和分析。考虑到岗位匹配度评价体系和求职者满意度评价体系中的离散型指标可能会对评估结果产生影响，为了更好地处理这些问题，本文可以将模糊理论引入到 TOPSIS 模型中，构建构建 topsis-fuzzy 模

型，将离散型指标映射为模糊隶属度函数，使得评估结果更加可靠。

TOPSIS-Fuzzy 模型的建模思路与 TOPSIS 相似，主要分为以下三个步骤：

- **确定评价指标和评价对象，并将其表示为评价矩阵 X**；对评价矩阵进行标准化，将其转化为归一化矩阵 X_normalized；确定各个指标的权重，将其表示为权重向量 W；计算加权标准化矩阵 X_weighted，即将归一化矩阵与权重向量相乘；
- **计算正理想解 ideal 和负理想解 anti_ideal，分别为各个指标在评价矩阵中的最大值和最小值**；计算各个评价对象到正理想解和负理想解的距离 D_pos 和 D_neg，分别表示各个评价对象到正理想解和负理想解的欧氏距离；
- **根据距离 D_pos 和 D_neg 计算模糊综合评价得分，即采用模糊理论对不确定信息进行综合评价。**具体方法包括计算到负理想解的距离与到正理想解的距离之比，然后采用模糊逻辑运算得到模糊隶属度函数 mu_neg 和 mu_pos，最后采用模糊化解的方法得到模糊综合评价得分。

利用 python 导入模糊系统库“skfuzzy”来构建 TOPSIS-Fuzzy 模型，并运用其中的“defuzz”模块来计算 TOPSIS-Fuzzy 模型的评分结果。

部分岗位匹配度和求职者满意度的 TOPSIS-Fuzzy 模型的评分结果分别如表 23 和表 24 所示。

表 23 岗位匹配度的部分结果

招聘信息 id	求职者 id	岗位匹配度
1613439889204969472	1469877468583297024	0.278254
1613439889204969472	1467793287824932864	0.133659
1604731457065058305	1468183620412899328	0.275755
1604731457065058305	1487091440004759552	0.128396
1604731457065058305	1470313042016337920	0.276954
1604731457065058305	1473188657572741120	0.277968
1604731457065058305	1469191500528222208	0.282537
1604731457065058305	1473128370643533824	0.142863
1604731457065058305	1471662236597616640	0.130503
1604731457065058305	1470309657343033344	0.131519

表 24 求职者满意度的部分结果

求职者 id	招聘信息 id	公司名称	求职者满意度
1639549004641599488	1599949951281004544	极能信息	0.094492
1639549004641599488	1561971141940215809	漪畔网络	0.162098
1639549004641599488	1536666675389267968	侨益物流	0.091646

1639549004641599488	1534374988734398464	小蝌蚪电商	0.117599
1639549004641599488	1527491662845181952	大牛教育	0.259971
1639549004641599488	1526834259040534529	凝新科技	0.089721
1639549004641599488	1525006403364847617	麦贝科技	0.07696
1639549004641599488	1524956737138982913	科迈医疗	0.297012
1639549004641599488	1523505369161269248	六度科技	0.080609
1639549004641599488	1482196338836897804	佰聆数据	0.231231

5.2 基于自由文本对匹配度与满意度结果的修正

由于在向量空间模型（VSM）中，每个文本被表示为一个向量，而向量的每个维度对应一个特征（比如特征词），向量空间模型可以应用于各种文本处理任务，如文本分类、信息检索、推荐系统等。因此，为更好地对招聘信息与求职信息中的自由文本数据进行处理，本文引进向量空间模型，将招聘信息与求职信息中的自由文本数据表示为向量，从而可以进行基于向量的文本相似度计算。

假设招聘信息中的自由文本数量为 m，求职者信息中的自由文本数量为 n，招聘信息与求职信息文本相似度计算的基本步骤如下：

1.构建词汇表：首先本文基于招聘信息的聚类结果，将招聘信息分成 4 类，对每类招聘信息下的自由文本基于 TF-IDF 进行特征词抽取，选取 TF-IDF 值排名前 200 的词语作为特征词组成该类别招聘信息的词汇表，词汇表中的词语都会作为向量空间中的一个维度，每个类别招聘信息所提取的部分特征词如下表 25 所示：

表 25 特征词提取结果（部分）

策马奔腾型企业特征词	经验丰富型企业特征词	智慧前沿型企业特征词	持续探索型企业特征词	排名
游戏	医疗器械	游戏	银联	1
财务管理	ERP	视觉	工程师	2
催收	数据安全	PLM	动画	3
欺诈	商品	工程师	视觉	4
商品	游戏	bi	票据	5
大气	支付	算法	人力资源	6
气象	欺诈	爬虫	产品	7
支付	模板	金融	开发	8
发布	政务	人力资源	数据库	9
oracle	医疗器械	软件开发	支付	10

2.构建文本向量：对于每一则招聘信息，以及求职者信息中的自由文本，根据词汇表中的特征词出现情况构建文本向量，其中向量的每个维度对应一个特征词，向量的值表示该特征词在对应文本中的 TF-IDF 值，则对于每一则招聘信息自由文本与所有求职者信息中的自由文本，其文本向量矩阵可表示如下：

$$\begin{matrix} w_1 \\ \vdots \\ w_{200} \end{matrix}\begin{pmatrix} X_{1,1} & \cdots & Y_{1,n+1} \\ \vdots & \ddots & \vdots \\ X_{200,1} & \cdots & Y_{200,n+1} \end{pmatrix}$$

其中，w_i（i=1...200）表示提取的 200 个特征词，$X_{i,1}$（i=1...200）则表示每个特征词在该招聘信息自由文本中的 TF-IDF 值，$Y_{i,j+1}$（i=1...200，j=1...n）表示第 i 个特征词在第 j 个求信息自由文本中的 TF-IDF 值，因此总共生成 m 个文本向量矩阵。

以策马奔腾型企业某条招聘信息为例，可建立该招聘信息与求职信息的文本向量矩阵如表 26 所示。

表 26 某则招聘信息与求职信息的文本向量矩阵（部分）

	招聘信息	求职 1	求职 2	求职 3	求职 4	求职 5	...
游戏	0	0	0.08825	0	0.053297	0	...
财务管理	0.734877	0.05954	0	0.020031	0	0.138138	...
催收	0.730863	0.043661	0.08825	0.05554	0	0.166621	...
欺诈	0	0	0.083079	0	0.1765	0	...
商品	0	0.14848	0	0.069069	0.045986	0.051391	...
大气	0.706223	0.03862	0.08825	0.055059	0	0.076565	...
气象	0.70415	0.05554	0	0.08825	0.07424	0	...
支付	0.703525	0	0.041176	0.039848	0.068695	0.08656	...
发布	0	0.08825	0.045738	0	0.03636	0	...
oracle	0.681158	0.112081	0	0.091779	0	0.03862	...

3.计算相似度：将每一则招聘信息以及求职者信息中自由文本的文本向量表示为向量空间的点，可以使用各种相似度度量方法计算两个向量之间的相似度，进而比较两个文本之间的相似程度，即每一则招聘信息与求职者信息的自由文本匹配度。

其中常见衡量相似度的指标有余**弦相似度，皮尔逊相关系数：Jaccard 相似度，欧几里得距离，曼哈顿距离**，各个指标的简要介绍如下：

※余弦相似度

余弦相似度是一种广泛应用于文本相似性比较的度量方法。它通常用于判断文本在语义上的相似度。余弦相似度通过将两个文本分别转化成对应的词向

量，然后计算两个向量之间的余弦夹角来确定它们的相似度，其计算公式如下：

$$\cos(\theta) = \frac{\vec{A} \cdot \vec{B}}{|\vec{A}||\vec{B}|}$$

余弦相似度的优点是易于计算和实现，同时它在很多情况下表现良好，特别是在短文本的情况下。但是，余弦相似度无法捕捉词汇顺序的信息，同时也不能很好地处理语义上的相似度，因此在处理长文本和语义复杂的文本时，其表现可能会有所下降。

※皮尔逊相关系数

皮尔逊相关系数是一种用于计算两个变量之间相关性的度量方法。它通常用于计算两个数值型变量之间的相关性。皮尔逊相关系数通过将两个变量的均值和标准差计算出来，然后计算它们之间的协方差和标准差之积的比值来确定它们之间的相关性,其计算公式如下：

$$r_{AB} = \frac{\sum_{i=1}^{n}(A_i - \bar{A})(B_i - \bar{B})}{\sqrt{\sum_{i=1}^{n}(B_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(B_i - \bar{y})^2}}$$

其中A_i，B_i分别表示两个向量中的元素；$\bar{A}$，$\bar{B}$分别表示两个向量元素的均值。

皮尔逊相关系数的优点是可以处理多种类型的数据，同时也能够处理噪声数据。它是一种可靠的方法，可以很好地反映出变量之间的相关性。但是，皮尔逊相关系数的局限性也很明显，它只能反映出线性关系，而不能很好地处理非线性关系。

※Jaccard 相似度

Jaccard 相似系数是一种用于计算两个集合之间相似度的度量方法。它通常用于计算两个文档、两个图像或两个音频文件之间的相似度。Jaccard 相似系数通过计算两个集合的交集和并集的比值来确定它们之间的相似度，其计算公式如下(其中，A，B 分别表示两个向量各自元素所组成的集合)：

$$J(\vec{A},\vec{B}) = \frac{|A \cap B|}{|A \cup B|}$$

Jaccard 相似系数的优点是可以处理多种类型的数据，特别是对于处理文本和图像数据时非常有效。同时，它对于数据中存在噪声和离群点的情况也很鲁棒。但是，Jaccard 相似系数也存在一些缺点，比如无法区分不同的重要性和权重，并且不考虑两个集合之间的距离。

※欧几里得距离

欧几里得距离是一种用于计算两个向量之间的距离的度量方法。它可以用于多维空间中的点之间的距离计算。欧几里得距离通过计算两个向量之间的每

个对应维度的距离的平方和的开平方来确定它们之间的距离，其计算公式如下（其中，A_i，B_i分别表示两个向量$\vec{A}$，$\vec{B}$中的元素）：

$$d(\vec{A},\vec{B}) = \sqrt{\sum_{i=1}^{n}(A_i - B_i)^2}$$

欧几里得距离易于理解和计算，并且可以用于处理多维数据。然而在高维空间中，欧几里得距离可能不太适用，因为距离的定义会变得模糊，这也就导致了所谓的“维数灾难”，同时欧几里得距离对噪声敏感。

※曼哈顿距离

曼哈顿距离是一种用于计算两个向量之间距离的度量方法。它通常用于计算在城市中两个位置之间的距离，因为这种距离的计算方式类似于在城市中的行走距离。曼哈顿距离通过计算两个向量之间的每个对应维度之差的绝对值之和来确定它们之间的距离，计算公式如下（其中，A_i，B_i分别表示两个向量$\vec{A}$，$\vec{B}$中的元素）：

$$D(\vec{A},\vec{B}) = \sum_{i=1}^{n}|A_i - B_i|$$

曼哈顿距离的优点是易于理解和计算，同时也可以处理多维数据。与欧几里德距离不同的是，曼哈顿距离可以很好地处理离散数据和噪声数据。但是，它不能很好地处理连续数据，同时在高维空间中，曼哈顿距离也会遭受“维数灾难”的问题。

综合考虑到每个相似度指标的优缺点，本文将上述 5 个相似度指标进行**线性加权**得到一个**综合相似度指标**，并以此来考量招聘信息与求职信息中自由文本的相似度。考虑到在衡量自由文本相似度时各指标衡量标准的一致性，本文需要对计算所得的欧几里得距离与曼哈顿距离进行如下处理（其中$md(\vec{A},\vec{B})$，$mD(\vec{A},\vec{B})$分别表示基于欧几里得距离与曼哈顿距离所计算的$\vec{A}$，$\vec{B}$的相似度）：

$$md(\vec{A},\vec{B}) = \frac{1}{1 + d(\vec{A},\vec{B})}$$

$$mD(\vec{A},\vec{B}) = \frac{1}{1 + D(\vec{A},\vec{B})}$$

综合相似度计算公式为：

$$SIM(A,B) = \frac{\cos(\theta) + r_{AB} + J(\vec{A},\vec{B}) + md(\vec{A},\vec{B}) + mD(\vec{A},\vec{B})}{5}$$

将计算得出的相似度分别与岗位匹配度和求职者满意度进行加权平均处理，即可得到最终的岗位匹配度和求职者满意度结果，具体结果见附件“result3-1.csv”和“result3-2.csv”。

六、招聘求职双向推荐模型的建立

为了尽可能保证高履约率，本文基于上文计算得出的求职满意度和岗位匹配度逐步构建了三个招聘求职双向推荐模型，并通过对比各个模型的求解结果选出最优的招聘求职双向推荐模型。

6.1 基于互惠概率的招聘求职双向推荐模型

根据实际情况，双向推荐系统在给企业推荐求职者的时候，不应仅仅依据求职者的岗位匹配度大小，还应当考虑到求职者对招聘信息的满意度大小。这是因为企业固然更乐意给岗位匹配度较高的求职者发送 offer，但是可能存在求职者岗位匹配度高而求职者对企业招聘信息的满意度为 0 的情况。在这种情况下，若是企业仅仅参考岗位匹配度发送 offer，那么就可能导致履约率较低，并且耗费较大的人力成本和时间成本。

为了综合考虑岗位匹配度和求职者满意度来构建推荐模型，本文定义了以下变量：

- $\boldsymbol{P_{ij}}$:代表在只考虑岗位匹配度大小的情况下，招聘企业 i 给岗位匹配度非 0 的求职者 j 发送 offer 的概率，其定义表达式为(G_{ij}表示求职者 j 关于企业 i 的岗位匹配度，n_i表示招聘企业 i 岗位匹配度非 0 的求职者个数)：

$$\boldsymbol{P_{ij}} = \frac{G_{ij}}{\sum_{j=1}^{n_i} G_{ij}}, \quad i = 1,2,\dots,1573$$

- $\boldsymbol{p_{ij}}$:代表在求职者 j 的所有满意度非 0 的招聘企业给 j 发送 offer 的情况下，求职者 j 选择企业 i 的概率，其定义表达式为(M_{ij}表示求职者 i 对招聘企业信息 j 的满意度，m_j表示求职者 j 满意度非 0 的招聘企业信息个数)：

$$\boldsymbol{p_{ij}} = \frac{M_{ij}}{\sum_{i=1}^{m_j} M_{ij}}, \ j = 1,2,\dots 8623$$

- **互惠概率($\boldsymbol{H_{ij}}$)**：其定义表达式为：

$$\boldsymbol{H_{ij}} = P_{ij} \times p_{ij}$$

互惠概率$\boldsymbol{H_{ij}}$同时考虑了岗位匹配度和求职者满意度，代表着招聘企业 i 给岗位匹配度非 0 的求职者 j 发送 offer 的有效概率，避免了企业给岗位匹配度高而满意度较低或为 0 的求职者发送 offer 的情况，因此根据互惠概率构建推荐系统，能够有效提高推荐效率和降低推荐成本。

对于岗位匹配度为 0 的求职者和求职者满意度为 0 的招聘信息，分别有$P_{ij} = 0$、$p_{ij} = 0$，此时H_{ij}也为 0。当$H_{ij} = 0$时，招聘企业 i 不需给求职者 j 发送 offer。考虑到企业往往偏好匹配度较高的求职者，本文设置阈值$\alpha \in [0,1]$,只给招聘企业推荐$H_{ij} > \alpha$的求职者，且招聘企业只会给$H_{ij} > \alpha$的求职者发送 offer。

假设某岗位拟聘 x 人，当H_{ij}>α的求职者人数不小于 x 时，泰迪内推平台向企业推荐互惠概率最大的 x 位求职者发出第一轮 offer；当H_{ij}>α的求职者人数小于 x 时，泰迪内推平台向企业推荐H_{ij}>α的求职者发出第一轮 offer。然后平台根据当前各招聘信息的剩余岗位数，向后续被推荐求职者发出第二轮 offer，如此继续，直到招聘人数已满或者向所有拟推荐求职者均已发出 offer 为止。

基于上述模型确定的策略，本文统计出所有岗位的签约总人数以便后文比较不同模型的优劣性。

6.2 基于贪心优化的招聘求职双向推荐模型

所谓贪心优化[21]，就是在每个步骤中选择最优的决策，以期望最终结果是全局最优的。为了使得履约率指标达到最高，本文在互惠概率的基础上，进一步利用贪心优化算法构建基于贪心优化的招聘求职双向推荐模型。本文的主要思路是在每一轮发送 offer 时选择签约人数最多的策略，以期望最终所有岗位的签约人数之和最大。

- **目标函数**：每一轮发送 offer 使得每一轮岗位的签约人数达到最大，其对应数学表达式为：

$$\max \sum_{i=1}^{1573} \sum_{j=1}^{8623} x_{ij}$$

假设每一轮求职者 j 收到的若干 offer，这些 offer 用集合U表示，那么$\{p_{ij}(U)\}$代表求职者 j 选择这些 offer 的概率集合，此时x_{ij}的取值条件为：

$$x_{ij}=\begin{cases}1, & H_{ij}>\alpha 且 p_{ij}=max\{p_{ij}(U)\}\\ 0, & 其它\end{cases}$$

- **约束条件：**

1.每个招聘信息能够招聘到的求职者是有数量上限的，其对应的数学表达式为：

$$\sum_{j=1}^{8623} x_{ij}\le R_i,\quad i=1,2,\dots,1573$$

其中R_i代表招聘信息 i 的拟聘员工数量。

2.每位求职者最多只能选择接受一份 offer，其对应数学表达式为：

$$\sum_{i=1}^{1573} x_{ij}\le 1,\quad j=1,2,\dots,8623$$

自此本文将问题转化为使每一轮的签约人数最大的优化问题，即整合成以下的优化问题：

$$\max \sum_{i=1}^{1573}\sum_{j=1}^{8623} x_{ij},$$

$$s.t.\begin{cases}\sum_{j=1}^{8623} x_{ij}\le R_i, i=1,2,\dots,1573\\ \sum_{i=1}^{1573} x_{ij}\le 1,\quad j=1,2,\dots,8623\\ x_{ij}=\begin{cases}1, & H_{ij}>\alpha 且 p_{ij}=max\{p_{ij}(U)\}\\ 0, & 其它\end{cases}\end{cases}$$

需要说明的是，由于上式是每一轮贪心优化问题的表达形式，所以其中的某些参数需要根据上一轮发送 offer 的结果进行更新，例如参数R_i，若招聘信息 i 还未招满人且在上一轮已经成功签约一名求职者，那么新一轮优化问题当中的R_i需对应地减少 1 人。本文将会运用 python 当中的 cvxpy 库求解上述优化问题，并统计所有岗位的签约总人数与发送 offer 的最大轮数。

6.3 基于动态规划的招聘求职双向推荐模型

考虑到本题的双向推荐有多个轮次，每轮求职者选择岗位的决策会影响后续轮次的选择，属于多阶段决策问题，因此本文决定在前面两个双向推荐模型的基础上，使用动态规划来解决这个问题。此问题转化为动态规划问题的基本思路是将其分解为子问题，并且需要考虑到当前状态对未来的影响。本文将这个问题分为多个阶段，每个阶段处理一个招聘轮次，即向求职者发出 offer 的过程。对于每一轮招聘，本文需要考虑的是当前阶段的状态，以及如何从上一轮招聘的状

态转移而来。

- **目标函数**

本文将招聘流程建模成一个多阶段决策过程，每个阶段代表每轮 offer 的发出过程。假设一共有 k 个阶段（即一共发出 k 轮 offer），那么目标函数就是 k 个阶段签约的总人数最多，履约率最高，其对应的数学表达式为：

$$\max \sum_{c=1}^{k} \sum_{i=1}^{1573} \sum_{j=1}^{8263} x_{c,ij}$$

其中决策变量为$x_{c,ij}$，代表在第 c 个阶段求职者 j 选择招聘信息 i 的 offer 结果，假设每一轮求职者 j 收到的若干 offer，这些 offer 用集合U表示，那么$\{p_{c,ij}(U)\}$代表在第 c 个阶段求职者 j 选择这些 offer 的概率集合，此时x_{ij}的取值条件为：

$$x_{c,ij} = \begin{cases} 1, & H_{ij} > \alpha \text{且} p_{c,ij} = max\{p_{c,ij}(U)\} \\ 0, & \text{其它} \end{cases}$$

- **约束条件**

1. 每个招聘信息能够招聘到的求职者是有数量上限的，其对应的数学表达式为：

$$\sum_{c=1}^{k} \sum_{j=1}^{8623} x_{c,ij} \le R_i, \quad i = 1,2,\dots,1573$$

其中R_i代表招聘信息 i 的拟聘员工数量。

2. 每位求职者最多只能选择接受一份 offer，其对应数学表达式为：

$$\sum_{c=1}^{k} \sum_{i=1}^{1573} x_{c,ij} \le 1, \quad j = 1,2,\dots,8623$$

- **状态转移方程**

令$f_{c,ij}$表示第 i 轮发出招聘 offer 时，求职者 j 接受招聘信息 i 的 offer 后最多的签约人数，经分析，最终的状态转移方程可确定为：

$$f_{c,ij} = max(f_{c-1,ij} + \sum_{i=1}^{8263} S_{c,p})$$

其中，$D_{c,i}$表示第 c 轮中求职者 i 可以接受 offer 的候选岗位集合，$S_{c,p}$表示第 c 轮中岗位 p 对履约率的贡献，即签约人数的增加值。

自此可将问题转化整合成以下的动态优化问题：

$$\max \sum_{c=1}^{k} \sum_{i=1}^{1573} \sum_{j=1}^{8263} x_{c,ij}$$

$$s.t. \begin{cases} \sum_{c=1}^{k} \sum_{j=1}^{8623} x_{c,ij} \le R_i, \quad i = 1,2,\dots,1573 \\ \sum_{c=1}^{k} \sum_{i=1}^{1573} x_{c,ij} \le 1, \quad j = 1,2,\dots,8623 \\ x_{c,ij} = \begin{cases} 1, & H_{ij} > \alpha \text{且} p_{c,ij} = max\{p_{c,ij}(U)\} \\ 0, & \text{其它} \end{cases} \\ f_{c,ij} = max(f_{c-1,ij} + \sum_{\substack{i=1 \\ p \in D_{c,i}}}^{8263} S_{c,p}) \end{cases}$$

6.4 结果对比分析

本文将基于互惠概率的招聘求职双向推荐模型、基于贪心优化的招聘求职双向推荐模型、基于动态规划的招聘求职双向推荐模型分别简称为模型 A、模型 B、模型 C。根据履约率的定义：履约率=所有岗位的签约人数之和/所有拟聘岗位人数之和，考虑到部分招聘信息的拟聘人数为“不限”，为了便于计算履约率，本文将这部分招聘信息实际签约的人数为“拟聘人数”。本文计算得出当参数α取不同值时每种模型的履约率结果，并绘制成折线图，具体结果见图 14.

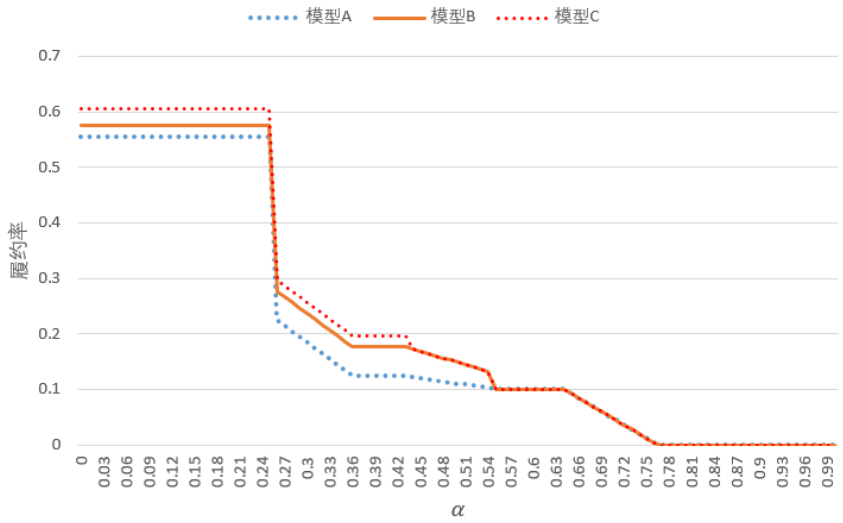


图 14 三种模型的履约率结果

根据图 14 进行分析，本文可以得到以下四条结论：

- **总体来看，履约率是随着α的增加呈阶梯式下降，基于动态规划的求职双向推荐模型是最优的。**当$\alpha \in [0,0.2501]$时，模型 A、B、C 的履约率结果均达到最高值，分别为：55.45%、57.66%、60.65%。这是因为基于互

惠概率的招聘求职双向推荐模型和基于贪心优化的招聘求职双向推荐模型的求解思路容易陷入局部最优解，难以为本文提供履约率最高的策略；基于动态规划的招聘求职双向推荐模型运用动态规划算法的思想得到全局的最优解，可以为本文提供履约率最高的推荐策略。另一方面，设置较低的α，意味着企业给求职者发送 offer 的条件放低了，这导致三种模型的履约率相同且均达到了最高

- **当$\alpha > 0.4312$时，模型 B 和模型 C 的履约率结果折线重合，即此时贪心优化的结果和动态规划的结果相同。**经分析，本文得出了其中原因：当$\alpha > 0.43$时，由于满足$H_{ij} > \alpha$条件的求职者较少，因此只需发一轮 offer 即可触发“停止发送 offer”的条件，在这种情况之下，动态规划和贪心优化均是单阶段决策模型，所以两者的结果相同。
- **当$\alpha > 0.5423$时，三种模型的履约率结果折线重合**。较高的α意味着企业向求职者发送offer的门槛变高，达到发送offer标准的求职者减少，不需要设置复杂的推荐策略来提高履约率，使用模型 A 可以简单高效地得到最优的推荐结果。
- **当$\alpha > 0.7609$时，三种模型的履约率结果均为 0**，这是因为不存在满足该条件的求职者，企业可发送的 offer 数量为 0.

在构建招聘求职双向推荐系统时，不仅仅需要考虑履约率的结果，还需要考虑发送 offer 的轮数，因为企业发送 offer 的轮数越大，所需耗费的人力成本和时间成本往往就越大。为了分析不同模型的推荐策略成本，本文统计了当α为 0 时，三种模型下需要发送 offer 的最大轮数 k 和履约率结果。

表 27 三种模型的最大轮数与履约率结果

模型	模型 A	模型 B	模型 C
最大轮数 k	4	3	3
履约率	55.45%	57.66%	60.65%

从表 27 结果显示，当$\alpha = 0$时，模型 A 的履约率最低，需要发送 offer 的最大轮数是三种模型中最多，结果为 4；而模型 B 和模型 C 的最大轮数相对较少，均为 3。综合来看，基于动态规划的双向推荐模型是高效的，它能以最低的成本达到最优的履约率结果。因此，本文最终确定最优模型为当$\alpha = 0$时的基于动态规划的招聘求职双向推荐模型，该模型的全部推荐结果见附件“result4.csv”，在这里本文附上部分结果，如表 28 所示。

表 28 模型 C 的部分推荐结果

招聘信息 id	求职者 id	岗位匹配度	求职者满意度
1461593160642854912	1469191201306574848	0.294919476	0.193403963
1461593160642854912	1469191500528222208	0.293442523	0.18977832
1461593160642854912	1470313042016337920	0.286311621	0.144114343
1461593160642854912	1468183620412899328	0.284732315	0.129524156
1461593160642854912	1498147738557218816	0.279469752	0.091646057
1463066895702949888	1466233482740105216	0.295860258	0.200882263
1463339842803990528	1463685809768103936	0.202892868	0.174469547
1482192491158568977	1461534785997504512	0.346314588	0.445627709
1482192491158568977	1461530285551255552	0.34271749	0.350312053
1482194394978320389	1554782002648055808	0.325002502	0.360358961
1482194394978320389	1471283827795165184	0.319372121	0.333309888

七、参考文献

[1]张长华.大数据视域下网络招聘数据信息挖掘的研究[J].科学技术创新，2021（10）：114-115.

[2]汤飞弘.基于 Python 爬虫的招聘信息数据可视化分析[J].软件,2023,44(01):176-179.

[3]刘雷,胡文利.高职院校教育信息爬取与数据分析研究[J].网络安全技术与应用,2021,No.250(10):69-70.

[4]栗琛.基于用户画像的虚拟求职社区推送服务研究[D].上海师范大学,2022.DOI:10.27312/d.cnki.gshsu.2022.001118.

[5]惠义禹.基于 GA-KNN 分类模型在船期数据分析中的研究与应用[D].电子科技大学,2016.

[6]Mikolov, T., Chen, K., Corrado, G., and Dean, J. , Efficient estimation of word representations in vector space. In Proceedsings of Workshop at ICLR, 2013a.

[7]Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. ,Distributed representations of words and phrases and their compositionality. In NIPS, pp. 3111– 3119, 2013b.

[8]李明.基于文本挖掘的招聘信息分析和职位画像[D].苏州大学,2021.DOI:10.27351/d.cnki.gszhu.2021.002617.

[9]李楚.基于 TF-IDF 算法的文本量化方法及作者识别应用[J].现代信息科技,2022,6(19):1-6+12.DOI:10.19850/j.cnki.2096-4706.2022.19.001.

[10]卓佳怡,于劲松,张力文等.基于 TF-IDF 算法的公文用户画像[J].办公自动化,2020,25(17):61-64.

[11]黄虹霞. 基于频繁项集挖掘和用户聚类的协同过滤算法研究[D].南昌大学,2021.DOI:10.27232/d.cnki.gnchu.2021.002852.

[12]李晓丽,苏钦,吴博等.基于 K-means 聚类算法的百货商场用户价值分析[J].山西师范大学学报(自然科学版),2023,37(01):7-13.DOI:10.16207/j.cnki.1009-4490.2023.01.012.

[13]曾如明. K-means 聚类算法的改进及其应用研究[D].西华师范大学,2022.DOI:10.27859/d.cnki.gxhsf.2022.000453.

[14]顾明星,黄伟建,黄远等.结合用户聚类与改进用户相似性的协同过滤推荐[J].计算机工程与应用,2020,56(22):185-190.

[15]曾如明,李云飞.K-means 聚类算法的一种改进方法研究[J].邵阳学院学报(自然科学版), 2021,18(02):8-14.

[16]吉峰,许淑亚.驱动、困境与展望：论用户画像技术在计算广告中的应用[J].教育传媒研究,2023,No.43(02):59-64.DOI:10.19400/j.cnki.cn10-1407/g2.2023.02.006.

[17]叶珍. 基于 AHP 的模糊综合评价方法研究及应用[D].华南理工大学,2010.

[18]章穗,张梅,迟国泰.基于熵权法的科学技术评价模型及其实证研究[J].管理学报,2010,7(01):34-42.

[19]信桂新,杨朝现,杨庆媛等.用熵权法和改进 TOPSIS 模型评价高标准基本农田建设后效应[J].农业工程学报,2017,33(01):238-249.

[20]侯志东,吴祈宗.基于 Hausdauff 度量的模糊 TOPSIS 方法研究[J]

[21]董军军.动态规划算法和贪心算法的比较与分析[J].软件导刊,2008,No.64(02):129-130.