第十一届“泰迪杯”
数据挖掘挑战赛

优秀作品

作品名称：基于迁移学习与集成学习的招聘与求职双向推荐系统

荣获奖项：特等奖

作品单位：中国地质大学（武汉）

作品成员：张文然 靳博原 何向洋

指导老师：万林

基于迁移学习与集成学习的招聘与求职双向推荐系统

摘要

本文基于 **LDA 主题生成**模型与 **K-modes** 算法，构建了招聘信息画像和求职者画像；依据**岗位胜任力模型**公式分别构建岗位匹配度与求职者满意度模型；基于**迁移学习**与**集成学习**模型构建排序模型，最终通过**贪心策略**实现招聘与求职双向推荐。

针对问题一，本文使用爬虫算法爬取泰迪内推网站的找工作和找人才页面，并将招聘者和岗位的相关信息进行了数据清洗和预处理，使其变为结构化数据，存储在 CSV 文件中。

针对问题二，应用问题一结果信息，针对不同特征，从多个方向建立招聘信息画像与求职者画像。使用 **LDA 主题生成**模型对招聘信息、求职者自我介绍进行主题词生成，得到其主题关键词。通过 **K-modes 聚类**算法，分别对招聘信息、求职者进行聚类，得到每个类别的代表性特征，即分类后的整体画像。

针对问题三，本文基于**岗位胜任力模型**公式，分别构建岗位匹配度模型和求职者满意度模型，从多维度对求职者或招聘岗位进行评估，得到求职者与招聘岗位间的匹配度和满意度。

针对问题四，分为两部分进行求解，首先基于**迁移学习**在公开数据集上对模型进行训练，选出最好的模型 **LightGBM** 后，基于其预测分数获得求职者推荐排序序列。分别通过三种**贪心策略**，选择推荐序列中能够最优化履约率的求职者进行推荐，求职者选择接受最高满意度岗位的 offer。通过对比实验，使用考虑 offer 数量的贪心策略能够得到最接近于最佳结果的履约率，其与本数据集履约率极限值的比值为 **0.7802997**。

关键词 双向推荐系统 人岗匹配 迁移学习 LightGBM 集成学习 贪婪策略

目录

一、 绪论 1
1.1 数据挖掘背景 1
二、 问题分析 1
2.1 问题一的分析 1
2.2 问题二的分析 3
2.3 问题三的分析 3
2.4 问题四的分析 4
三、 基本假设 4
四、 针对问题一的解决方案 5
4.1 工作思路 5
4.2 数据获取 5
4.3 数据清洗 7
五、 针对问题二的解决方案 10
5.1 工作思路 10
5.2 数据处理 11
5.2.1 缺失值处理 11
5.2.2 招聘岗位标签 11
5.2.3 薪资数据 13
5.2.4 提取技能关键词 13
5.2.5 地址数据 14
5.3 模型设计与选择 15
5.3.1 LDA 主题模型 15
5.3.2 K-modes 聚类模型 16
5.4 招聘信息画像 17
5.4.1 招聘岗位 17
5.4.2 薪资待遇 18
5.4.3 学历要求 20
5.4.4 岗位需求量 21
5.4.5 公司类型 22
5.4.6 岗位技能 23
5.4.7 企业工作地点 24
5.4.8 公司规模 25
5.4.9 工作经验需求 26
5.4.10 岗位福利 27
5.5 求职者画像 27
5.5.1 预期岗位 27
5.5.2 薪资需求 28
5.5.3 知识储备 29
5.5.4 学历 30
5.5.5 工作经验 31

5.5.6 居住地和工作地 32
5.6 模型应用 34
5.6.1 LDA 主题模型 34
5.6.2 K-modes 聚类模型 36
5.6.3 招聘岗位聚类 37
5.6.4 求职者聚类 38
六、 针对问题三的解决方案 39
6.1 工作思路 39
6.2 模型设计 39
6.3 建立匹配度模型 40
6.3.1 学历 40
6.3.2 薪资 40
6.3.3 工作经验 41
6.3.4 技能 41
6.3.5 工作地点 41
6.3.6 工作类型 41
6.4 建立满意度模型 42
6.4.1 薪资 42
6.4.2 期望行业 42
6.4.3 工作地点 43
6.4.4 工作类型 43
6.4.5 工作福利 43
6.5 匹配度、满意度求解 43
七、 针对问题四的解决方案 44
7.1 工作思路 44
7.2 排序模型设计 44
7.2.1 迁移学习 45
7.2.2 LightGBM 算法 45
7.3 排序模型应用 46
7.4 贪心策略的推荐模型 47
7.4.1 计算履约率极限值 47
7.4.2 贪心选择 48
八、 模型的评价与推广 49
8.1 模型的优点 49
8.2 模型的缺点 49
九、 参考文献 49

一、 绪论

1.1 数据挖掘背景

在新时代的背景下，随着大学生毕业人数的不断增加，大学生就业问题已经成为了广泛关注的社会热点。此外，受疫情影响，许多企业的招聘已经改为线上进行，脱离了时间和空间的限制，招聘需求不断增加，其中科技研发、数字化和蓝领技能岗位的需求量较大，但也存在不同程度的人才短缺。然而，从人才供给的角度来看，应届生数量正在增加，部分企业的校园招聘活动也已经暂缓或推迟，因此出现了校园招聘需求缩减或冻结的情况。这些因素加剧了应届生就业的严峻形势，使就业竞争压力大、招聘和求职信息不对称等现象变得更加普遍。

泰迪内推平台是一家聚焦于“大数据+”和“人工智能”领域的求职招聘网站。该平台融合了多家企业发布的招聘信息，同时为求职者提供求职信息的展示。为了缓解毕业生的就业压力，同时满足企业对人才的需求，泰迪内推平台会定期为高校学生提供优质岗位推荐，解决毕业生就业问题的同时也缓解企业用人难的问题，为校企之间搭建起资源互换的桥梁，力求实现人才的供需对接和教育资源转化，通过深化产教融合，促进教育链、人才链、产业链和创新链的有机衔接。

因此，对招聘信息进行分析研究，了解不同职业领域的需求特点，挖掘兴起的数据类行业相应的人才需求现状及发展趋势，为广大求职者提供正确的就业指导具有重要意义。

二、 问题分析

2.1 问题一的分析

问题一要求从泰迪内推平台的“找工作”页面和“找人才”页面，爬取所有招聘与求职信息。依据招聘信息 ID 记录、整理每条招聘信息并保存为文件；依据求职者 ID 记录求职者信息并保存。

首先在官网的“找工作”页面找到“售前技术支持”这个岗位，考虑使用 Python 语言的 Requests 爬虫库等工具编写爬虫程序，向该岗位页面发送 HTTP 请求，获取到页面内容后，使用正则表达式或 HTML 解析器从中提取出需要的招聘信息，如职位名称、薪资待遇、公司名称、职位描述、技能要求等。然后整理成可读的格式保存到 CSV 文件中。

对于每个求职者则向“找人才”页面发送请求获取相应的内容，解析其 HTML 代码，并且提取出求职者 ID，最后整理成 CSV 文件。可以考虑使用多线程或异步编程等技术来加速爬取过程。也需要注意如何处理页面内容的编码问题、如何处理页面翻页、如何处理反爬机制等细节。

对于爬取到的所有招聘与求职信息，进行数据清洗和预处理，比如去重、转换数据格式、去除无效数据等。将清洗和处理后的数据存储后，方便后续问题的分析和使用。

尝试从数据获取到数据处理的整个数据分析流程，学习爬虫、数据清洗、数据分析等相关技巧。整体的思路见图 2 所示。


售前技术支持
12000 - 20000·月
全职 | 本科 | 不限 | 6人
收藏
投递简历
2023-04-19 11:22 职位发布
2023-12-31 00:00 投递截止
职位关键词
互联网
软件
技能要求
网络安全
云计算
职位描述
岗位职责：
1、负责配合客户及项目需要，完成技术交流，提供解决方案；
2、负责网络安全、基础网络、云计算等项目的售前支持，包括但不限于售前咨询、整体规划、方案设计、标书制作、投标及应答；
3、负责协助销售人员对客户在项目过程中进行沟通、服务、技术支持；
4、深度挖掘市场需求、发展方向。
任职资格：
1.一年及以上网络安全、数据安全系统集成相关产品售前和技术方案工作经验，本科及以上学历，计算机网络、通信、电子等相关专业；
2、能独立撰写项目投标方案；
3、熟悉深信服、奇安信、绿盟，华为、华三，阿里云、腾讯云等主流厂商及产品方案；
4. 熟悉企业，园区解决方案；
5. 较强的沟通协调能力、学习能力及责任心。
职位福利
餐饮补贴
专业培训
弹性工作
公司简介
众云网
互联网/计算机软件
民营公司
150-500人
职位发布者
超级管理员
HR

图 1 招聘岗位信息页面

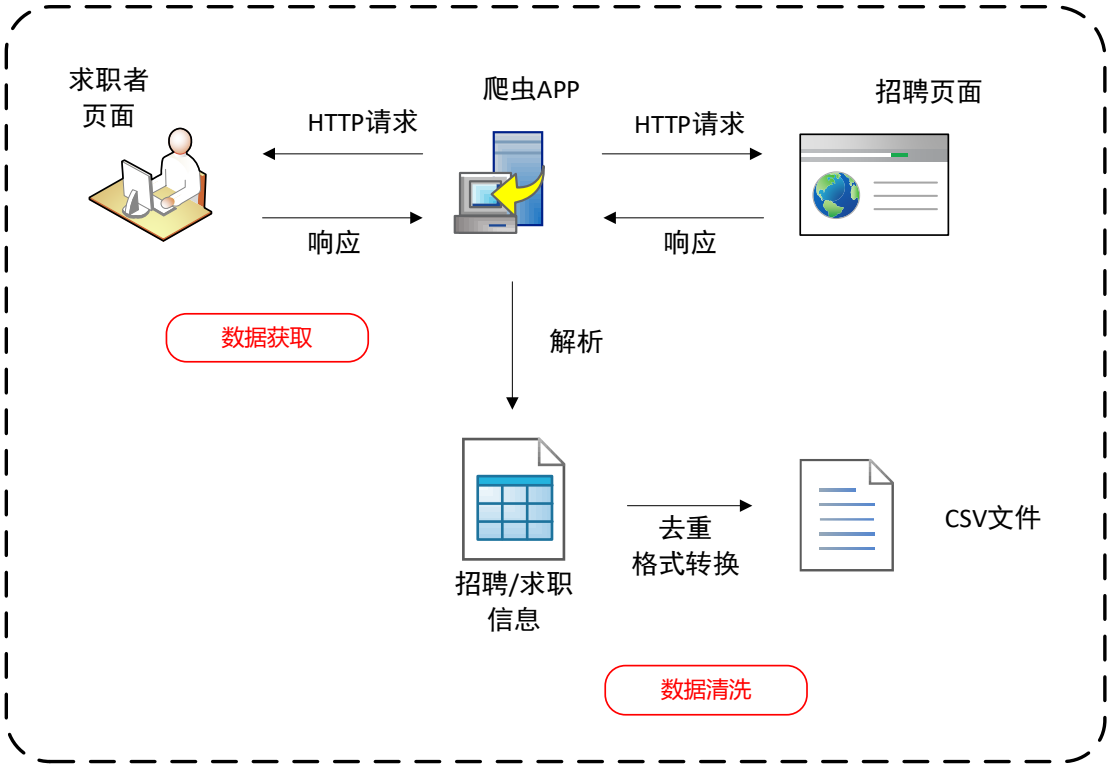

图 2 问题一流程

2.2 问题二的分析

问题二要求应用问题一的招聘信息与求职信息构建画像：根据采集的企业招聘信息，从招聘岗位、学历要求、岗位需求量、公司类型、薪资待遇、岗位技能、企业工作地点等多个方向建立招聘信息画像；根据采集求职者求职信息，从预期岗位、薪资需求、知识储备、学历、工作经验等多个方向建立求职者画像。

首先考虑对问题一中保存的招聘信息和求职信息做进一步数据处理，比如缺失值处理、文本数据可以使用 TF-IDF 等算法提取关键词、使用 One-hot 编码处理可分类的特征、使用归一化处理连续特征等，最后将其转换为结构化数据。

以招聘信息为例，拟从题目要求的多方面提取岗位的特征，如统计每个岗位的招聘人数，统计其类型分布、学历要求分布、薪资待遇分布、工作地点分布等信息。在此基础上进一步分析和挖掘，可以对薪资待遇进行聚类分析，得到不同薪资水平的职位和要求；可视化工作地点的地理分布情况，得到不同城市和地区的招聘和求职热点等。

在后续问题中可以利用得到的企业招聘信息画像和求职者画像，对招聘和求职过程进行分析和优化，制定更精准的招聘和求职推荐策略，提高招聘和求职的效率和质量。

2.3 问题三的分析

问题三要求根据招聘信息与求职者信息，构建岗位匹配度和求职者满意度的模型，并且基于该模型，为每条招聘信息提供岗位匹配度非 0 的求职者，以及为每位求职者提供求职者满意度非 0 的招聘信息。

岗位匹配度是企业考虑求职者掌握的技能、预期岗位、薪资需求、学历和工作经验等个人情况，多方面评估求职者是否符合招聘要求的匹配程度。可以结合该岗位对于不同能力维度的要求权重，通过计算求职者在各个能力维度的水平等级，得到一个

综合的分数以衡量与岗位要求的匹配程度。

求职者满意度则是求职者对于招聘岗位的满意程度，需要考虑招聘岗位与自我预期是否匹配、所提供的薪资待遇和福利等信息。可以结合求职者对于不同需求维度的重视程度，通过计算招聘岗位在各个维度的得分情况，得到一个综合的分数以衡量求职者的满意程度。

综上，模型的目标是基于爬取的信息为每个招聘岗位（求职者）计算一个匹配度（满意度）得分，得分越高表示匹配度（满意度）越高。在模型构建后，可以对每条招聘信息和每个求职者进行匹配度或满意度评估，保存每条招聘信息的匹配度非 0 的求职者和每个求职者满意度非 0 的招聘信息。

2.4 问题四的分析

问题四要求按照给定的流程设计招聘求职的双向推荐模型，通过多轮的 offer 推荐，最终实现求职者与岗位的匹配签约，使得履约率指标达到最高。

首先对题目所给招聘流程中的各个环节进行分析，发现求职者选岗位的原则是依据问题三构建的求职者满意模型，在收到多于一个 offer 时选满意度最高的岗位进行签约。因此，问题四的主要目标是构建向岗位推荐求职者的模型，以最大化招聘流程中的履约率。

通过爬取的招聘岗位信息，我们可以确定参加招聘的求职者人数和拟聘岗位人数之和，这意味着履约率存在一个极限。在每一轮推荐中，我们需要为每个招聘岗位选择合适的求职者，因此需要采用适当的算法来调整推荐顺序，以逼近履约率的最大值。可以采用启发式算法，如贪心算法、遗传算法、模拟退火算法等，以最大化招聘岗位的签约人数，从而提高推荐系统的履约率。

三、 基本假设

1. 假设爬取的招聘信息、求职者信息真实可靠；
2. 不考虑求职者对公司未来发展机遇、公司对求职者潜力培养等潜在因素。

四、 针对问题一的解决方案

4.1 工作思路

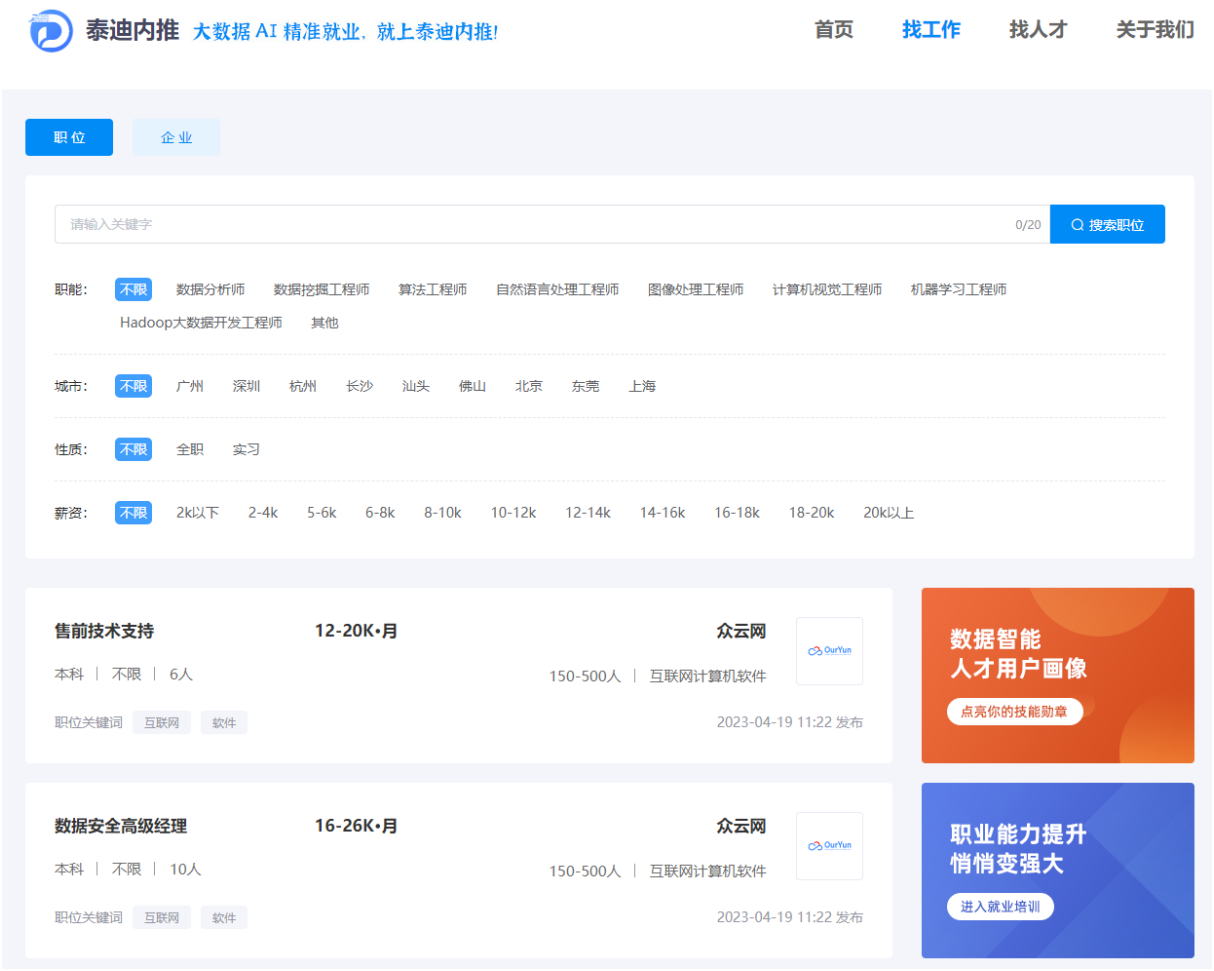

图 3 泰迪内推平台“找工作”页面

以爬取“招聘信息”为例，我们首先使用了 Requests 爬虫库工具编写爬虫程序，向泰迪内推平台的“找工作”页面发送 HTTP 请求，获取到页面内容后，使用解析器从中提取出，保存“招聘信息 ID”“企业名称”“招聘岗位”等其他需要的招聘信息，然后整理成可读的格式保存到 CSV 文件中。其中，我们使用多线程技术来加速爬取过程，通过 pageSize 和 pageNumber 两个查询参数指定每页返回的记录数和要获取的页码数。

最后对于爬取到的招聘与求职信息，进行了数据清洗和预处理，比如去重、转换数据格式、去除无效数据等。

4.2 数据获取

爬虫的工作原理是模拟人类对网站的访问行为，通过发送请求获取网页内容，然后解析网页内容提取需要的数据，所以爬虫技术满足问题一的需求。我们指定爬取“泰迪内推”官网，根据不同的任务分别确定要爬取的页面和数据类型等参数，以便自动化地执行抓取任务。

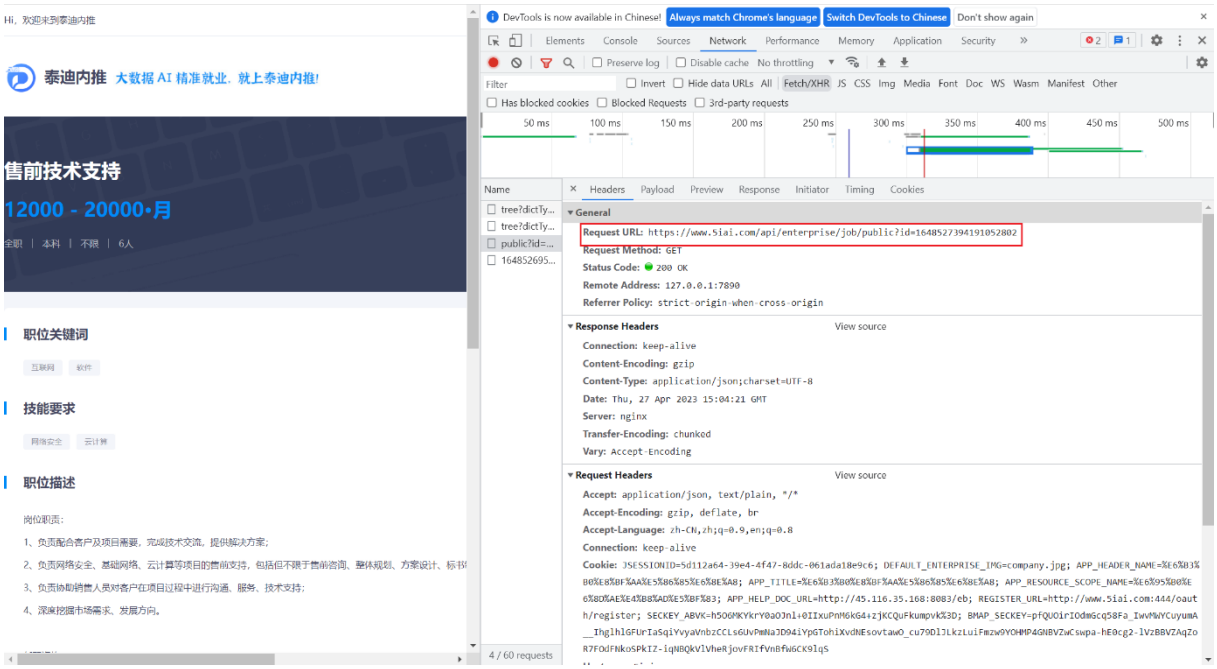

图 4 获取 URL

我们通过网页的开发者工具来获取要爬取的 URL。在开发者工具中，使用网络面板来查看网页请求和响应的详细信息。找到要爬取的 URL 后，设置请求头部信息 headers，以便模拟用户访问网站，使用 Python 的 requests 库发送 HTTP 请求，获取指定页面的招聘信息并进行数据抓取。

print_hi() 函数接收一个参数 page，用于指定要爬取的页面。它首先构造了请求 URL，然后使用 requests 库的 get()方法发送请求并获取响应。响应内容是一个 JSON 格式的字符串，需要使用 json 库的 loads()方法将其转换为字典类型。接下来，对于每个招聘信息，该函数将其提取为一个字典类型的 item，并将其添加到 data 列表中。最后该函数输出 item 并将其添加到 data 列表中。

主函数主要用于控制程序的流程。首先，使用一个 for 循环来遍历每一页的数据，即调用 print_hi()函数并传入相应的页码。接着，将所有爬取到的数据存储在一个名为 df 的 Pandas DataFrame 中，又创建了一个名为 df1 的 DataFrame 用于存储数据的序号，再将这两个 DataFrame 拼接起来，并将结果写入 CSV 文件中。

整个数据获取的流程图见图 5 所示。

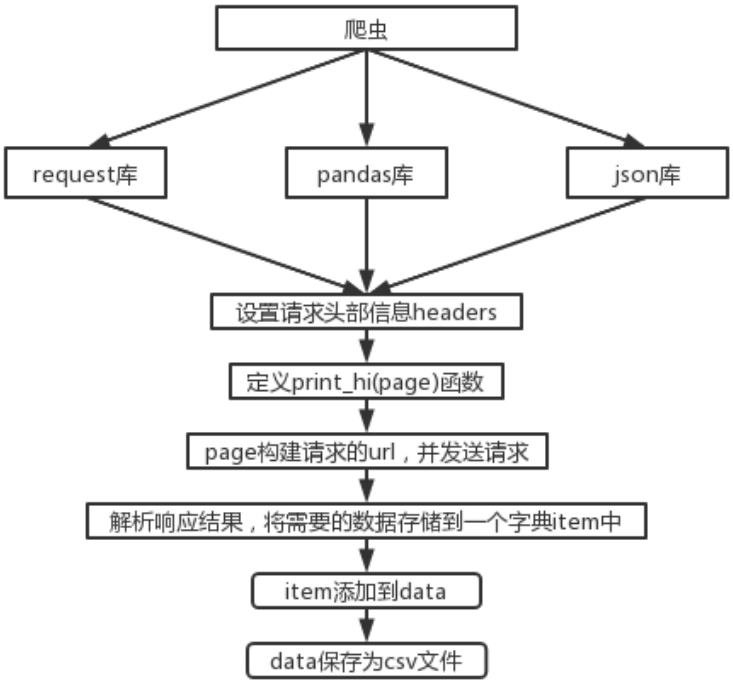

图 5 爬虫流程图

4.3 数据清洗

1. 去重

首先对数据集进行去重。例如，我们爬取到条求职者数据，进行重复性检测发现有 xx 条重复数据。

2. 异常值

以求职者信息为例，使用爬虫得到的“预期岗位”和“期望行业”列中，存在部分乱码数据“\\\”，将其修改为正常形式。

表 1 异常值表

求职者 ID	姓名	预期岗位	…	期望行业
1461606361627492352	朱先生	"\"\\\"\\\\\\\"[\\\\\\\\\\\\\\\"数据分析师\\\\\\\\\\\\\\\"]\\\\\\\"\\\"\""	…	["互联网","信息安全"]
1469220920555077632	刘女士	"\"[\\\"数据分析师\\\",\\\"数据挖掘工程师\\\"]\""	…	"\"[\\\"不限\\\"]\""

3. 缺失值

清洗招聘信息时发现，有 6 条数据的“工作地点”数据缺失。而泰迪内推官网有企业的信息，可以爬取到每个企业的 ID 号，获取企业名称和公司地址。

以“招聘信息 ID”：“1517417343414829056”为例，通过招聘信息中的“企业名称”：“广州交易猫信息技术有限公司”为索引，最后将该招聘信息的地址修改为“江西省上饶市信州区”。

表 2 缺失值表

招聘信息 ID	企业名称	招聘岗位	… …	工作地点
1517417343414829056	广州交易猫信息技术有限公司	游戏客服专员	… …	.
1504651614491901952	广州鼎捷软件有限公司	数据实施顾问	… …	.

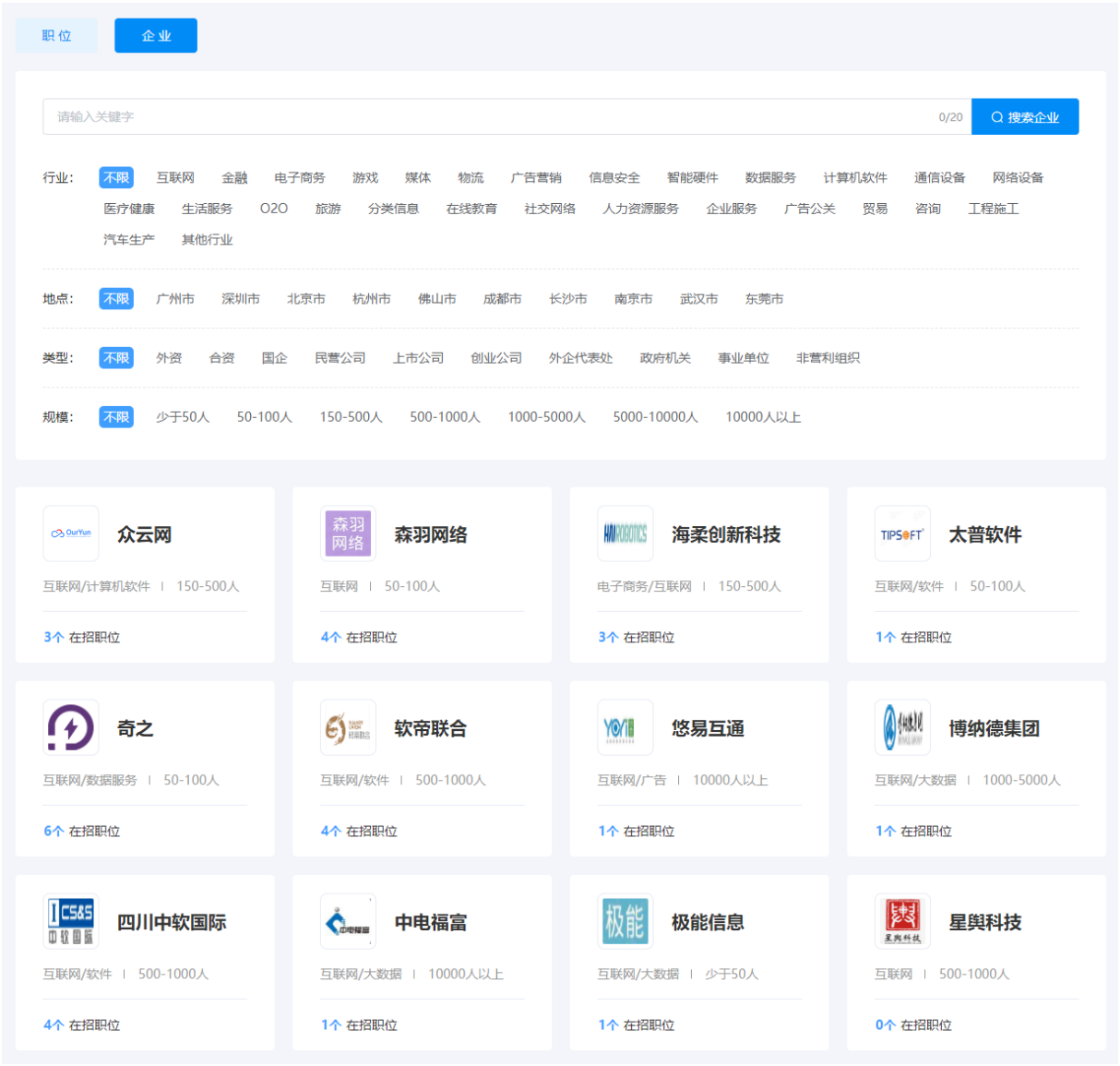

图 6 泰迪内推官网企业信息

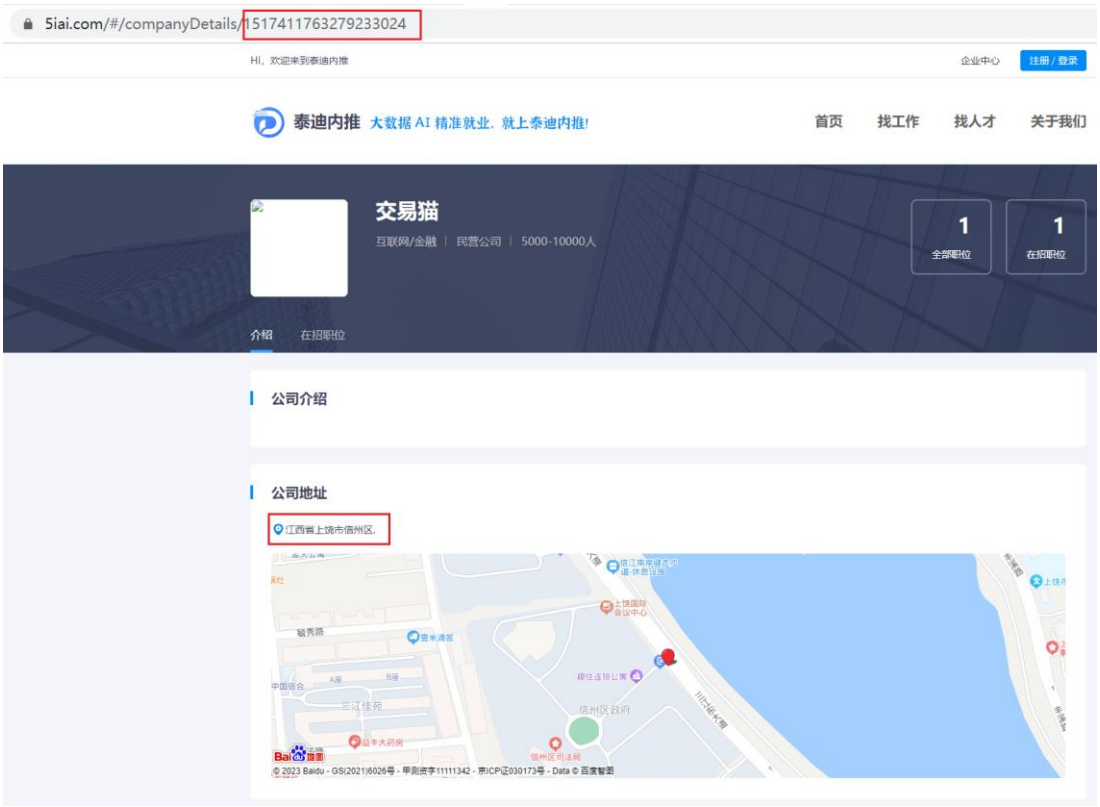

图 7 获取修改信息

4. 占位符转换

以招聘信息的清洗为例，对于“招聘人数”原始 JSON 文件读出来的 0 代表“不限”，“工作类型”中的 0 代表“实习”，2 代表“全职”……查看岗位工作类型，发现均为数字 0、2，查看求职者数据，发现工作性质存在 0、1、2、5 四类；通过泰迪内推网官网可知，岗位工作类型中 0 对应实习，2 对应全职；而求职者工作性质中，1、2、5 均为全职，0 为实习。将数字转换为对应类型。

5. 数据格式转换

对于“职位福利”，原始 JSON 文件读出来的["餐饮补贴","专业培训","弹性工作"]等列表数据转换为字符串“餐饮补贴，专业培训，弹性工作”等。

最终处理完的数据共计 1575 组招聘信息数据，10858 组求职者信息数据。求职者部分数据见表 3，完整数据表见附录。

表 3 数据处理结果（求职者）

序号	求职者 ID	姓名	预期岗位	工作经验	……	自我评价	……
120	146780208146 8481536	蔡女士	数据分析师	无经验	……	熟悉数据分析的相关流程；熟悉 SQL 数据查询语言；熟悉 python 的编程语言；有良好的自驱力,责任心强,具备良好的团队协作能力；熟悉 Word、Excel 等办公软件	……

121	1470307844552261632	曾女士	其他	无经验	……	1、熟悉 python 编程语言、能自己独立编程并更改报错； 2、具备数据库知识，熟练掌握 SQL 数据查询语言； 3、能够自己设计和编辑 PPT 模板、熟练操作 EXCEL 的公式和函数； 4、在企业参加过为时四个月的关于数据分析的培训。	……
122	1468079045689344000	曾女士	数据分析师 数据挖掘工程师	无经验	……	1、熟悉 python 编程语言、能自己独立编程并改报错； 2、具备数据库知识，熟练掌握 SQL 数据查询语言； 3、能够自己设计和编辑 PPT 模板、熟练操作 EXCEL 的公式和函数； 4、在企业参加过为时四个月的关于数据分析的培训。	……

五、 针对问题二的解决方案

5.1 工作思路

问题二要求应用问题一的招聘信息与求职信息构建画像：根据采集的企业招聘信息，从招聘岗位、学历要求、岗位需求量、公司类型、薪资待遇、岗位技能、企业工作地点等多个方向建立招聘信息画像；根据采集求职者求职信息，从预期岗位、薪资需求、知识储备、学历、工作经验等多个方向建立求职者画像。

首先考虑对问题一中保存的招聘信息和求职信息做进一步数据处理，比如缺失值处理、文本数据可以使用 TF-IDF 等算法提取关键词、使用 One-hot 编码处理可分类的特征、使用归一化处理连续特征等，最后将其转换为结构化数据。

以招聘信息为例，拟从题目要求的多方面提取岗位的特征，如统计每个岗位的招聘人数，统计其类型分布、学历要求分布、薪资待遇分布、工作地点分布等信息。在此基础上进一步分析和挖掘，可以对薪资待遇进行聚类分析，得到不同薪资水平的职位和要求；可视化工作地点的地理分布情况，得到不同城市和地区招聘和求职热点。

5.2 数据处理

对于问题一所得到的数据，在经过去重、转化格式等操作之后，仍需要进一步进行数据处理，以便构建求职者和招聘信息的具体画像模型。在求职者信息中，有大量特征空缺且重复的用户数据，下文将这种数据称为“僵尸数据”，在某些情况下，这些“僵尸数据”可能会被去除以提高计算效率和准确性。

除此之外，问题二应用了 K-modes 聚类模型。该模型是一种用于聚类分类属性数据的方法，因此需将数据中的连续特征进行离散化处理。

5.2.1 缺失值处理

整体而言，招聘信息的数据更为完善，各个信息几乎不存在缺失值。然而，求职者信息数据中存在大量缺失值，对于数值数据采取用 0 填充，文本类特征用“未知”填充。

5.2.2 招聘岗位标签

如表 4、表 5 所示，对于每条招聘信息，招聘岗位的形式各不相同，并且与求职者的“预期岗位”无法一一对应，所以我们参考官网的“职能”分类进行打标签。

表 4 招聘信息招聘岗位表

招聘信息 ID	企业名称	招聘岗位
1648527394191052802	深圳市众云网有限公司	售前技术支持
1648527394191052801	深圳市众云网有限公司	数据安全高级经理
1648527394191052800	深圳市众云网有限公司	数据安全项目经理
1648165203084447745	济宁森羽网络科技有限公司	数据挖掘工程师

表 5 求职者预期岗位表

求职者 ID	姓名	预期岗位
1649221278801985536	王先生	数据分析师,数据挖掘工程师,其他
1648221000086716416	特先生	数据挖掘工程师,图像处理工程师
1648848763151843328	王先生	Hadoop 大数据开发工程师,其他,机器学习工程师
1648774046462115840	张先生	数据分析师,数据挖掘工程师,算法工程师

图 8 泰迪内推官网“职能”分类标签

首先对文本数据进行统一小写化处理，然后使用正则表达式去除中文中的空格，考虑到“职能”间无法清晰地划分，可以追加多个标签，最终整理标签。

图 9 以某招聘信息的岗位“NLP 算法 工程师”为例，展示处理的流程。

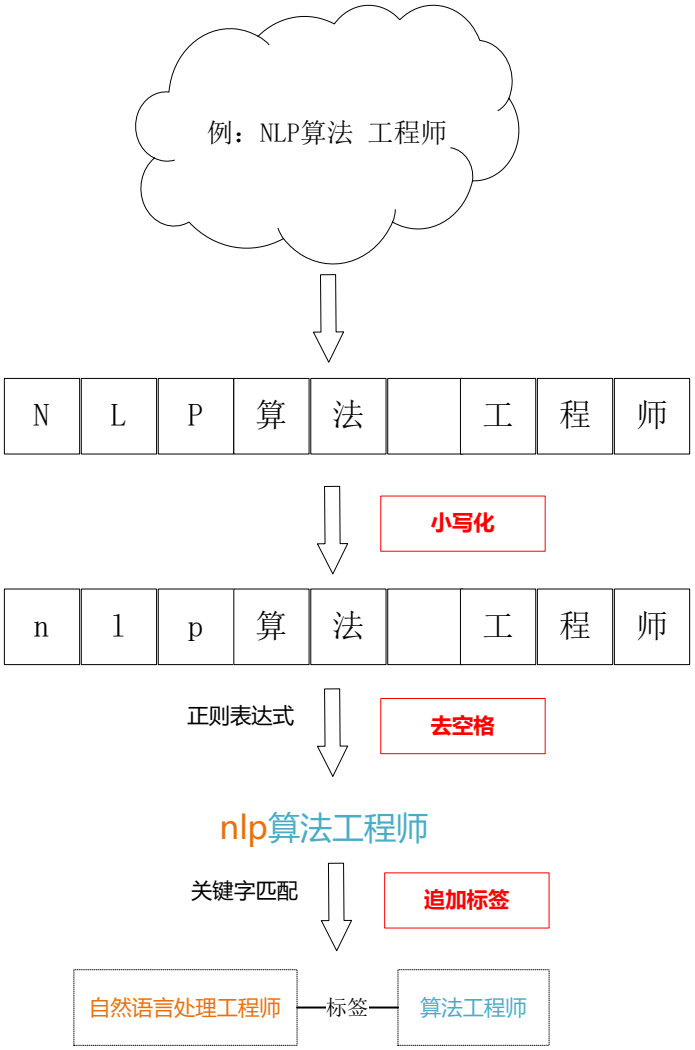

图 9 招聘岗位标签处理流程

具体关键词匹配的流程见图 10。对于招聘岗位“NLP 算法 工程师”而言，最后的标签为“自然语言处理工程师”和“算法工程师”；岗位“会计实习生”最终标签为“其他”。

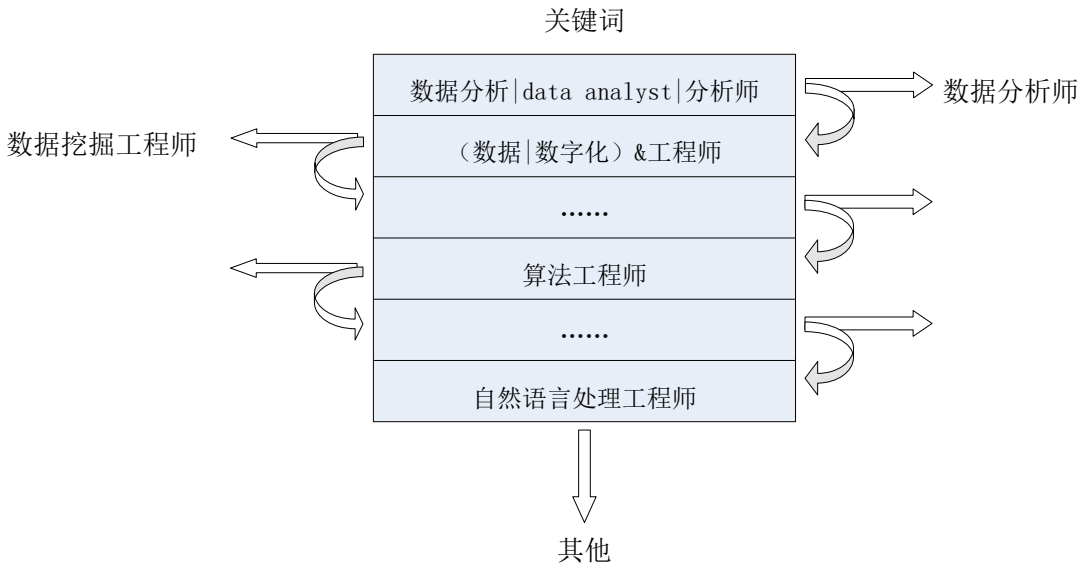

图 10 关键词匹配流程

5.2.3 薪资数据

从爬取的招聘信息可以看出，大部分岗位以“月薪”为结算方式，但少部分岗位给出的是“日薪”和“年薪”，并且求职者的期望薪资也均用月薪表示。为了方便比较岗位与求职者的匹配度，最终将薪资根据“结算频率”统一转换为月薪。

并且为了方便后续的聚类任务，按照泰迪内推官网的标签对薪资划分区间，以岗位提供的最低工资为划分标准，分为如下区间：2k 以下、2k-4k、4k-6k、6k-8k、8k-10k、10k-12k、12k-14k、14k-16k、16k-18k、18k-20k 以及 20k 以上。

5.2.4 提取技能关键词

对于招聘岗位，我们认为“职位关键词”、“技能关键词”以及“职位描述”文本均为岗位技能要求相关描述，因此对三个特征进行统一处理。观察数据可知，“职位关键词”、“技能关键词”已满足分词要求，但对技能要求描述不全面、有所缺失；而“职位描述”文本虽然全面阐述了岗位的技能要求，但非技能词汇过多需要进行筛选。

最终通过整合“职位关键词”和“技能关键词”作为关键词表，对“职位描述”的分词结果进行筛选，具体操作流程如下：

整合全部职位关键词、技能关键词后，进行去重以及英文字母小写化操作，得到主要关键词。表 6 展示了需要进行修改的主要关键词。对修改后主要关键词再次去重，得到最终的主要技能关键词表。

表 6 主要关键词修改对照表

主要关键词	修改后
"c、php"	"c" "php"
"pmp/cmmi"	"pmp" "cmmi"
"procedural language/sql"	"procedural language" "sql"
"hadoop hive sql"	"hadoop" "hive" "sql"
"javascri"	"javascript"

本文使用 jieba 库对岗位的“职位描述”进行分词。为了提高分词结果的准确性，我们使用了清华大学开放中文词库 THUOCL[1]中的 IT 类词汇来指导分词。通过引入这

些专业词汇，我们可以更准确地将文本分成有意义的词语，从而提高后续文本处理和分析的效果。接着将分词英文字母小写化，并使用哈工大停用词表去除无用分词。但是，在处理分词结果时，我们发现其中存在大量非技能性词项。因此，我们使用上文得到的主要关键词表对分词结果进行筛选，若分词存在于主要关键词表中，则认为该分词是当前招聘岗位的技能要求关键词。经过关键词筛选，最终得到了所有招聘岗位的技能要求关键词数据。

流程结构见图 11。

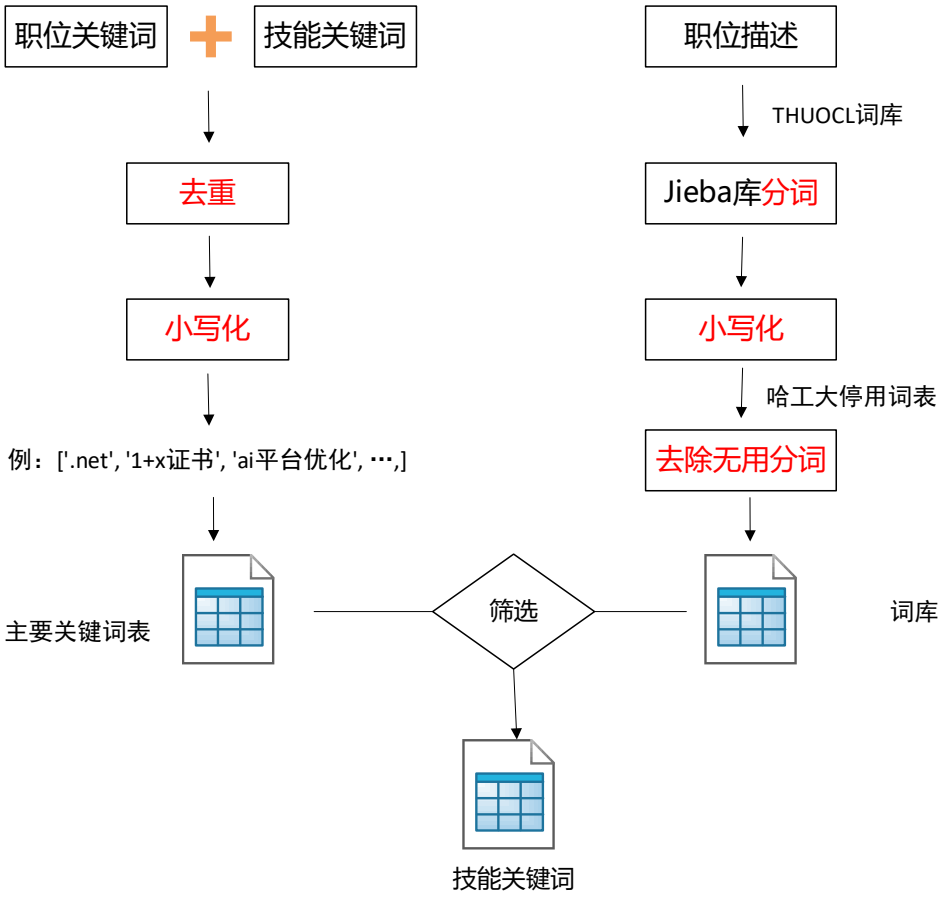

图 11 技能关键词提取流程

对于求职者而言，其个人技能相对于招聘岗位要求而言具有主观性。因此，我们同样使用上文招聘岗位的主要技能关键词表对其进行分词及筛选。观察求职者数据，发现“自我评价”、“简历关键词”以及“学科专业”均可作为技能描述特征。因此，我们将其合并为一段文本后再分词，后续操作与招聘岗位关键词提取方法相同。

5.2.5 地址数据

通过爬虫我们得到的是招聘岗位的详细地址，观察求职者数据发现其地址形式为“省份、城市和区县”，故对招聘岗位进行处理获取同样形式的数据。

为了处理这些地址数据，本文使用 GeocodingCHN 库将不规范或连续的文本地址尽可能转变为标准化的模块。具体而言，将招聘岗位的地址数据标准化为 Address 类的对象，并取省、市、区三个属性作为地址结果进行存储。如果对于省市区未识别出相关数据，则可以赋值为“未知”。

5.3 模型设计与选择

5.3.1 LDA 主题模型

LDA 主题模型是对文本进行分析，用于发现大规模文本语料库中隐藏的主题结构。它是在概率隐性语义索引上扩展得到的三层贝叶斯概率模型，包含词项、主题和文档三层结构。其基本思想是把文档看成其隐含主题的混合，而每个主题则表现为跟该主题相关的词项的概率分布[2]。我们可以认为一篇文章的每个词都是通过“以一定概率选择了某个主题，并从这个主题中以一定概率选择某个词语”这样一个过程得到。LDA 是一种非监督机器学习技术，基于词袋模型，认为文档和单词都是可交换的，忽略单词在文档中的顺序和文档在语料库中的顺序，从而将文本信息转化为易于建模的数字信息。

具体步骤如下：

1. 设定主题数量 K，以及语料库 D 和词项集合 V。

2. 初始化文档主题分布$\theta \sim Dir(\alpha)$和主题词分布$\phi \sim Dir(\beta)$，其中α和β是超参数。

其中，θ是 K 维 Dirichlet 随机变量，其概率密度如下所示：

$$P(\theta \mid \alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \ldots \theta_k^{\alpha_k - 1}$$

3. 对于文档 d 属于 D 中的每个文档：

1) 从主题词分布θ中随机选取一个主题组合z_K。

2) 对于文档中的每个单词 w，从文档主题分布θ中随机选取一个主题，并且从主题词分布ϕ中选取一个单词。

4. 重复以上步骤直到收敛为止。

在每次迭代中，LDA 会计算主题词分布和文档主题分布的概率分布，用于指导下一次迭代中的随机选择。最终得到的主题词分布和文档主题分布可以帮助我们理解文档集合中的主题结构，从而支持文本挖掘和信息检索等应用。

常见的 LDA 模型使用词频来构建矩阵，提取主题词。TF-IDF 是词频和逆文本频率指数的乘积。

词频（TF）表示词条在文本中出现的概率：

$$TF_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}}$$

其中，$n_{i,j}$表示词条t_i在文档 d_j中出现的次数，$TF_{i,j}$ 表示词条t_i在文档d_j中出现的频率。

逆文件频率（IDF）表示关键词的普遍程度，若包含词条i的文档越少，IDF 越大，说明该词条具有很好的类别区分能力。

$$IDF_i = log \frac{|D|}{1 + |j: t_i \in d_j|}$$

其中，$|D|$ 表示所有文档的数量，$\left|j: t_i \in d_j\right|$表示包含词条$t_i$的文档数量。

某一特定文件内的高词语频率，以及该词语在整个文件集合中的低文件频率，可以产生出高权重的 TF-IDF。因此，TF-IDF 倾向于过滤掉常见的词语，保留重要的词语，表达为

$$TF\text{-}IDF = TF \cdot IDF$$

5.3.2 K-modes 聚类模型

K-modes 是一种用于处理离散型数据集的聚类分析算法。与 K-means 算法相同的是，K-modes 也基于中心点，但不同之处在于它使用众数(mode)而不是平均值(mean)来计算聚类中心和样本之间的距离。

K-modes 算法的主要思路是将样本分配到 K 个聚类中心中的一个，并在每个聚类中心中找到出现最频繁的值，即众数，以更新聚类中心。

具体步骤如下：

1. 随机选择 K 个初始聚类中心点；
2. 对每个样本计算与聚类中心之间的距离，并将其分配到最近的聚类中心中；
3. 对每个聚类中的离散型特征，计算众数，并将其作为新的聚类中心；
4. 重复 2 和 3 步骤，直到聚类中心不再发生变化或达到预定的迭代次数。

由此可见，因为使用众数来计算聚类中心点，K-modes 算法能够更好地处理离散数据，这使得 K-modes 算法比其他基于距离的聚类算法更适用于处理分类变量。其次，K-modes 算法不需要对离散变量进行编码或规范化，因此可以减少数据预处理的时间和计算成本。

然而，K-modes 也存在一些基于中心点的算法的通病，即对初始聚类中心点的选择较为敏感，可能会导致聚类结果的不稳定性。

5.4 招聘信息画像

5.4.1 招聘岗位

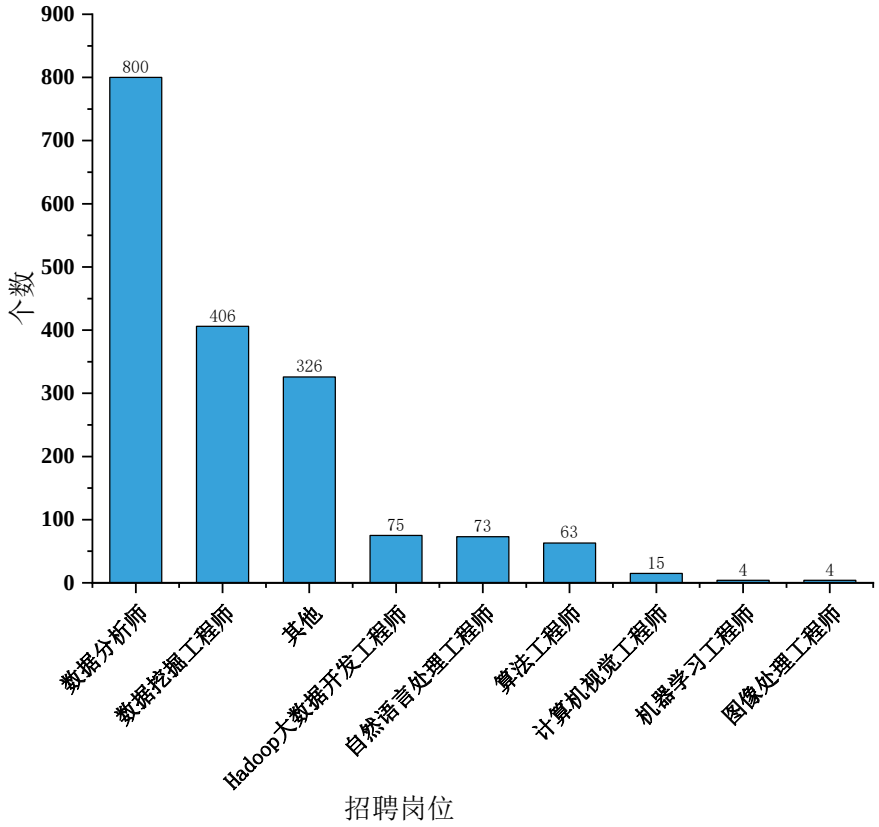

图 12 招聘岗位标签分布（柱状图）

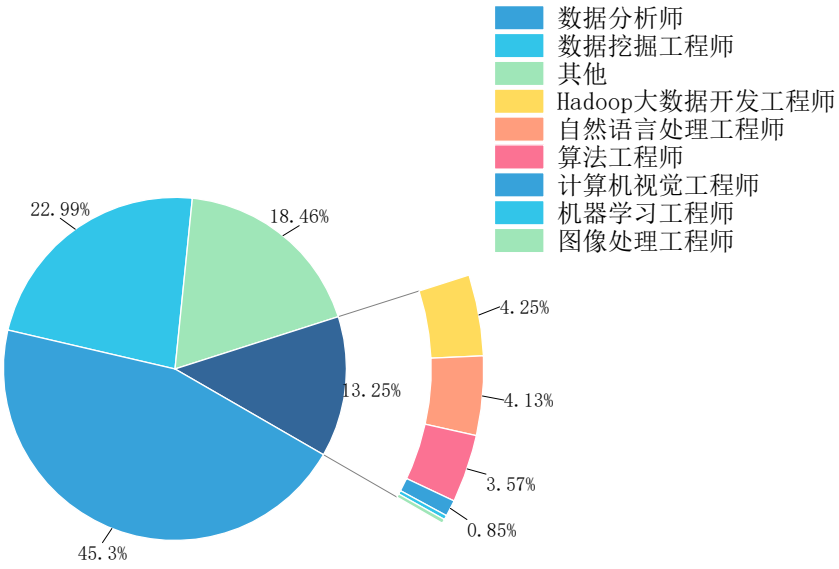

图 13 招聘岗位标签分布（饼状图）

在对招聘岗位进行进一步的数据处理后，我们将每个岗位打上了相应的标签，图 13 显示了这些标签的分布情况。从图表可以看出，大部分招聘岗位属于“数据分析

师”和“数据挖掘工程师”这种互联网行业的大类方向，而对于具体的小方向岗位如“图像处理工程师”和“机器学习工程师”的需求相对较少。

这一结果的分析可能是由于市场上“数据分析”和“数据挖掘”这些大类方向的需求量相对较高，因此相应的招聘岗位也更多。而对于较为具体的小方向岗位，可能需要的技能要求较高，招聘难度也相应增加，因此在市场上需求量相对较少。

5.4.2 薪资待遇

箱线图可用作直观地显示数据的分散情况，反映原始数据分布的特征和多组数据的比较。箱线图有 5 个参数，分别是上四分位数、上边缘、中位数、下边缘、下四分位数。

箱线图

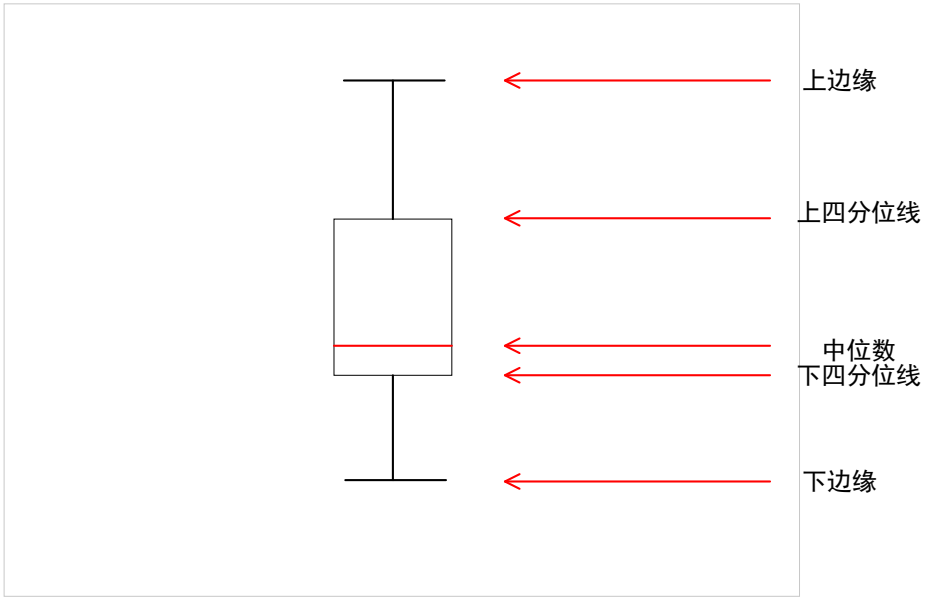

图 14 箱线图示例

箱线图四分位数的含义是，一组数据从小到大排列后，把该组数据四等分的数，称为四分位数。下四分位数 Q_1 和上四分位数 Q_2 分别为样本中第 25%、75%的数字，其差距称为四分位距 IQR。由四分位距可以计算上边缘 UE 和下边缘 LE，计算公式如下：

$$UE = Q_2 + \frac{3}{2} IQR$$

$$LE = Q_1 - \frac{3}{2} IQR$$

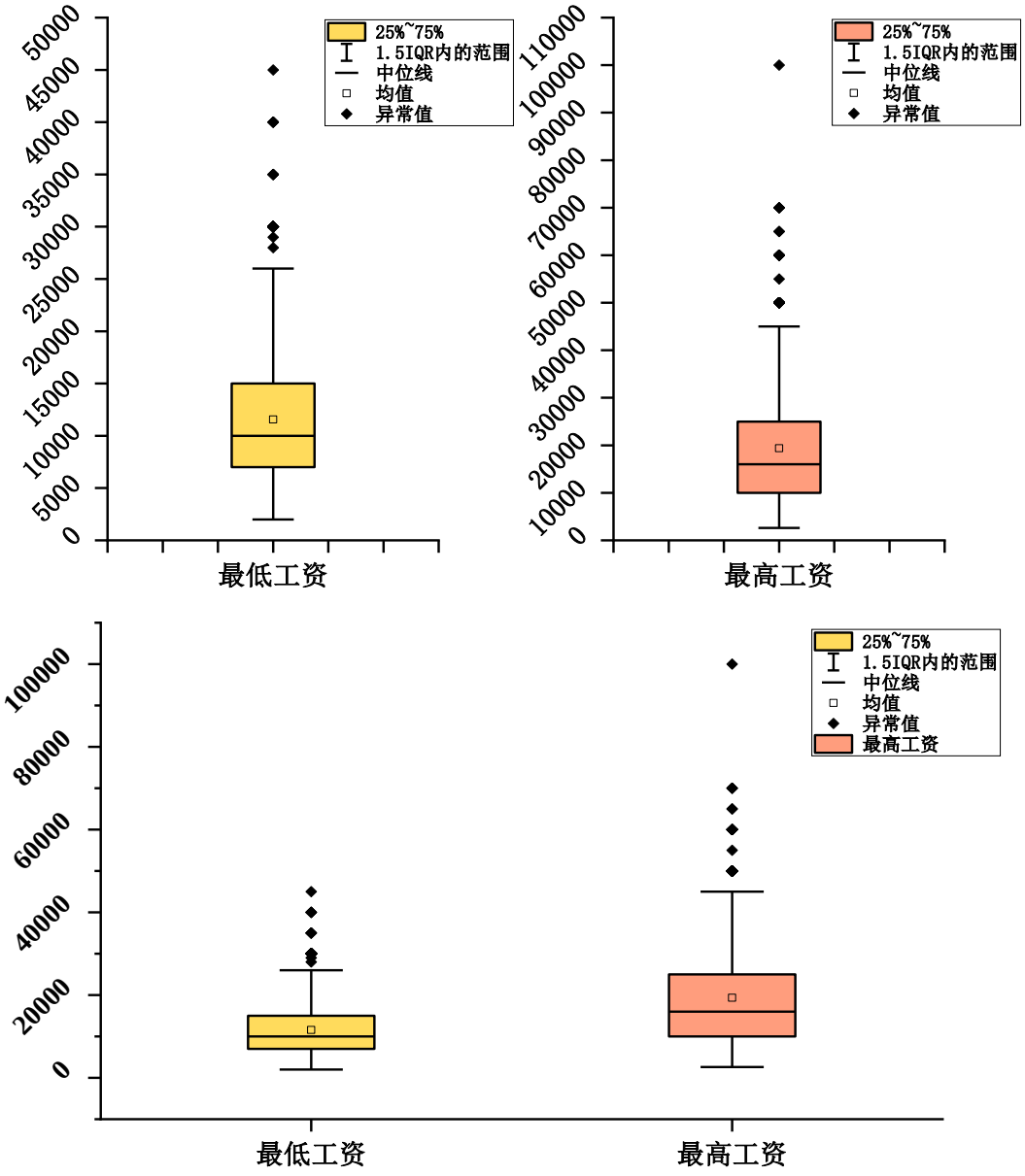

图 15 工资水平箱线图分布

图 15 显示了工资水平的箱线图分布情况。可以看到，大部分岗位提供的最低工资主要集中在 5k-20k 之间，而最高工资则主要集中在 10k-25k 之间。然而，也存在一些岗位提供的工资较高，这些岗位在箱线图中被标记为离群点。接下来，我们将在不同的维度上对工资分布进行更深入的分析，以便更好地从薪资待遇方面构建招聘信息画

基于“泰迪内推”官网提供的分类标准，我们将岗位提供的工资按照区间进行划分，包括'2k 以下', '2k-4k', '4k-6k', '6k-8k', '8k-10k', '10k-12k', '12k-14k', '14k-16k', '16k-18k', '18k-20k', '20k 以上'。可以看出，大多数岗位提供的工资也都集中在 5k-20k 之间，这表明在当前招聘市场上，该工资水平属于主流，也是大多数岗位提供的薪酬水平。

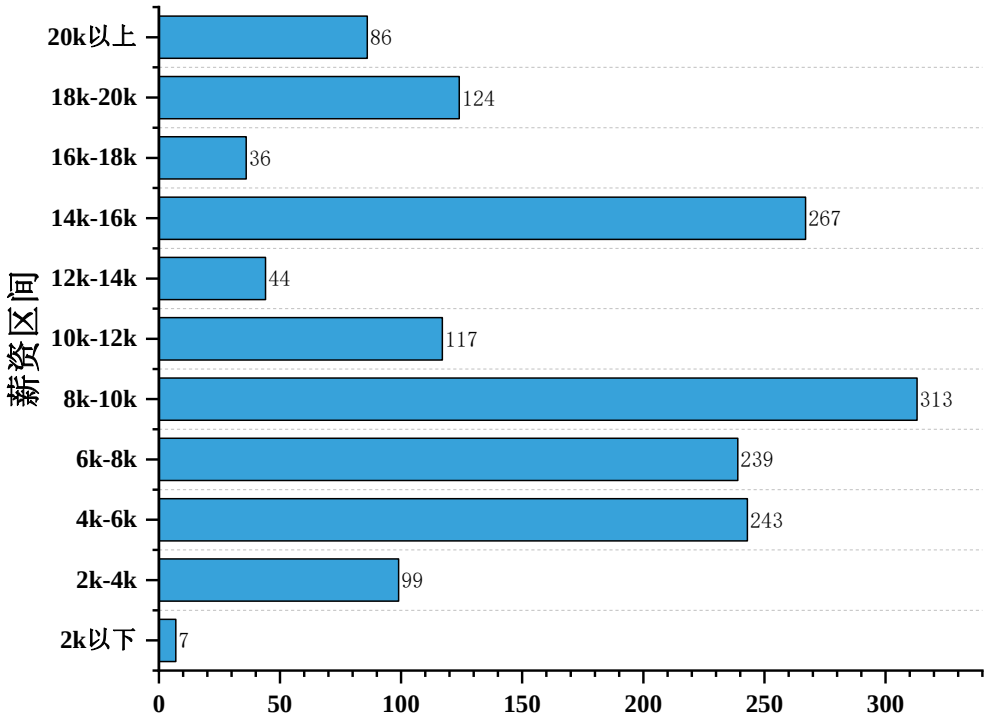

图 16 薪资区间分布

5.4.3 学历要求

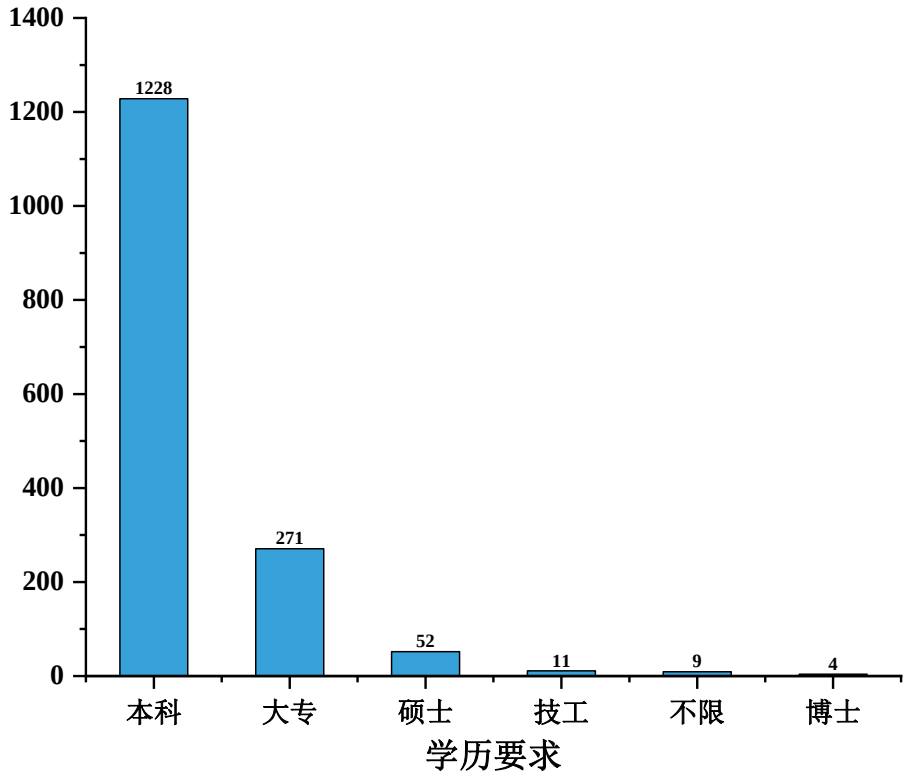

图 17 学历要求分布

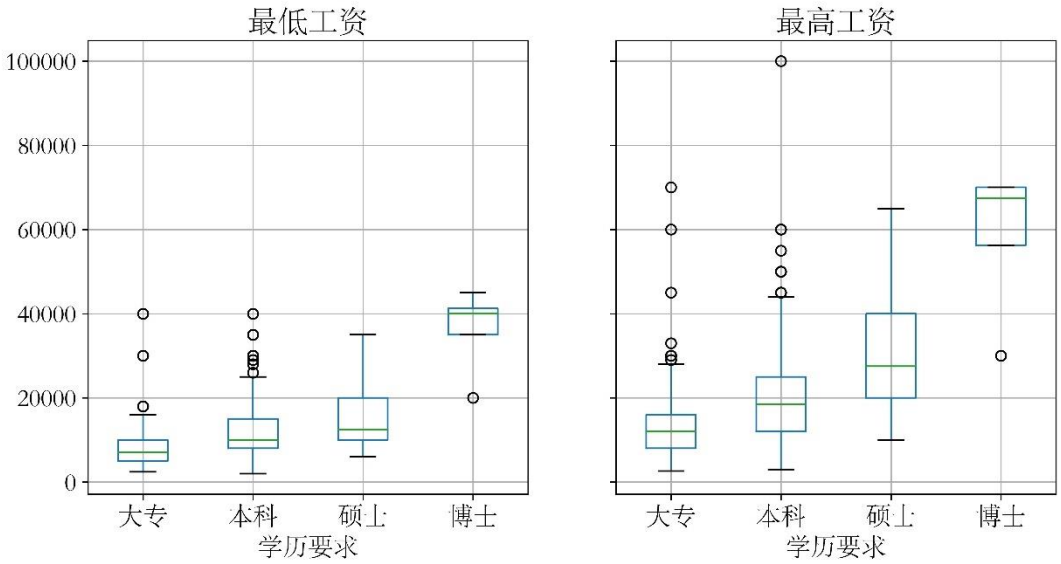

图 18 学历要求-薪资分布箱线图

通过观察图 18，我们可以发现，招聘岗位的学历要求主要集中在“本科”水平，相比之下，对于高学历“硕士”和“博士”，以及低学历“技工”的需求量较低，呈现出一种分布不平衡的趋势。这种趋势也与实际市场情况相符合。

在箱线图中，我们可以看出不同学历要求对应的薪资分布情况。随着学历要求的提高，招聘方所提供的最低和最高薪资主要集中的区域都在增加，这也符合市场情况。整体而言，在不同学历要求的招聘中，存在一定的薪资差异，高学历招聘的薪资水平相对较高，而低学历招聘的薪资水平则相对较低。

5.4.4 岗位需求量

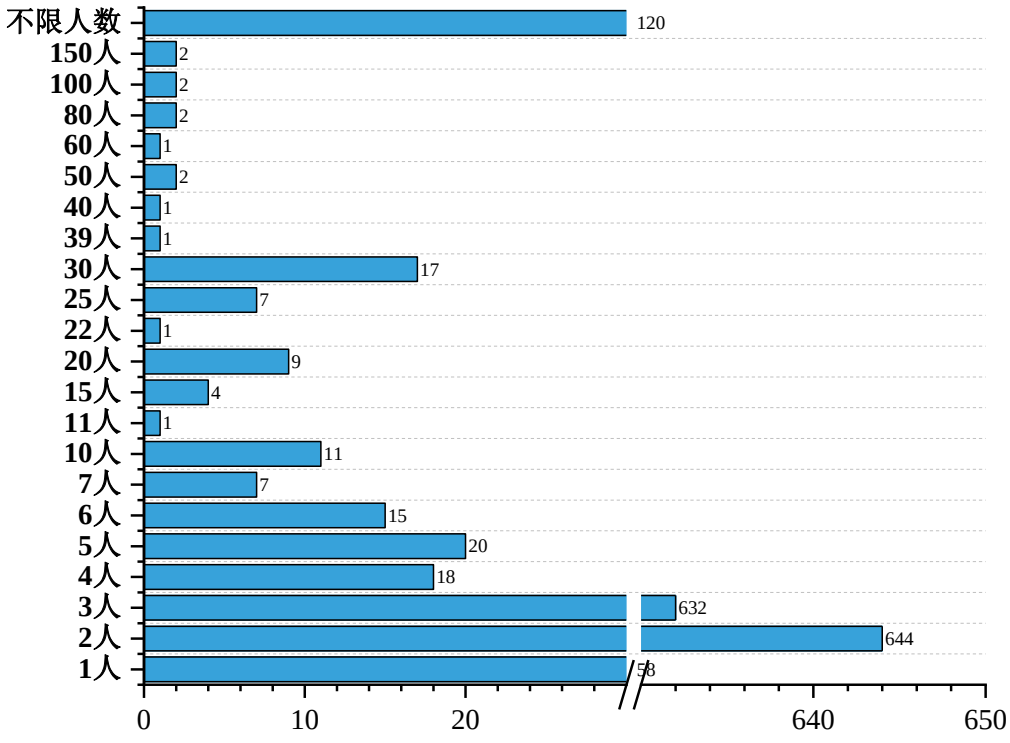

图 19 岗位需求量

按照区间进行划分，招聘人数 10 人以内是“岗位需求量小”，11-50 人是“岗位需求量适中”，50 人以上是“岗位需求量大”。

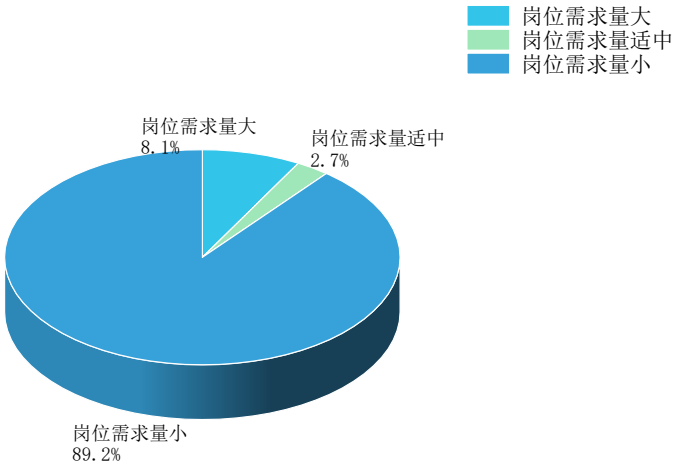

图 20 岗位需求量区间划分分布

获取的“招聘人数”信息可以用来统计各个岗位的需求量，见图 20。根据统计结果，大多数岗位只需要招聘 2-3 人，而很少有岗位一次性招聘 30 人以上，招聘 50 人以上的岗位数寥寥无几。为了更好地描述这些统计结果，我们将招聘人数按照 1-10 人、11-50 人和 50 人以上进行分类，并将其归为岗位需求量小、岗位需求量适中和岗位需求量大三类。根据饼状图的结果，约 90%的岗位需求量被归类为小型需求。

这一结果说明，在泰迪内推网的大多数岗位只需要招聘少数人，而只有很少的岗位需要招聘大量的人才。这也反映出当前经济形势下，企业对于人才需求的谨慎态度，更倾向于进行小规模的招聘，而非一次性招聘大量人才。

5. 4. 5 公司类型

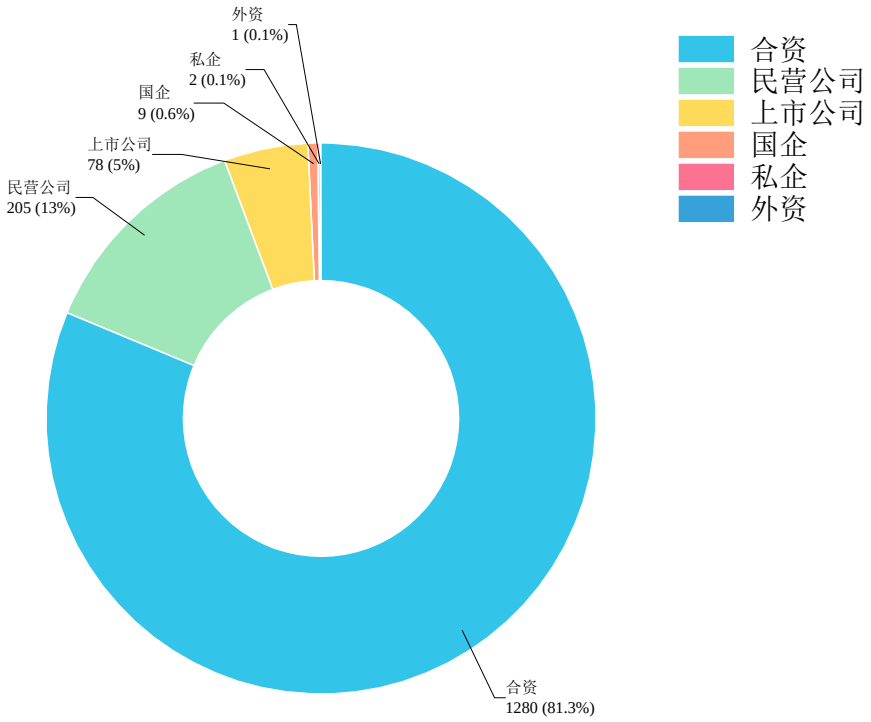

图 21 公司类型分布

招聘公司的类型主要是合资企业和民营企业，两者占据了 90%，而私企、外企的公司非常少。

根据图 21 显示，招聘公司主要分为合资企业和民营企业两种类型，这两种类型占据了总体的 90%。相比之下，私营企业和外企的数量较少。这一结果说明，泰迪内推网的大多数招聘公司属于合资企业和民营企业。由此可见，合资企业和民营企业在互联网行业的经济中扮演着重要的角色，而私营企业和外企的市场份额较小。能作为企业有价值的市场信息和参考依据，更好地制定人才招聘和发展战略，以适应互联网企业环境的发展和变化。

5.4.6 岗位技能

根据 5.2 节数据处理提取的技能关键词，绘制的词云如图 22 所示。可以看到，岗位要求的主要技能是“数据分析”“开发”“数据库”等。

图 22 岗位技能词云图

5.4.7 企业工作地点

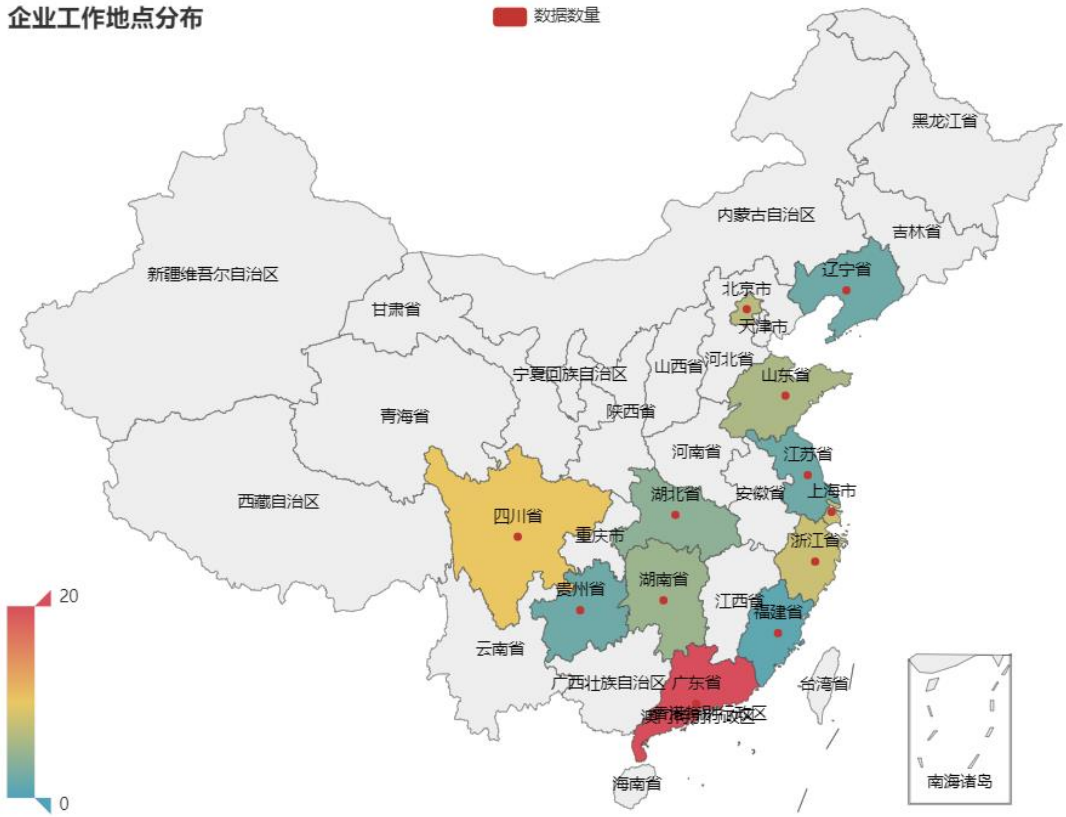

图 23 企业工作地点分布（省）

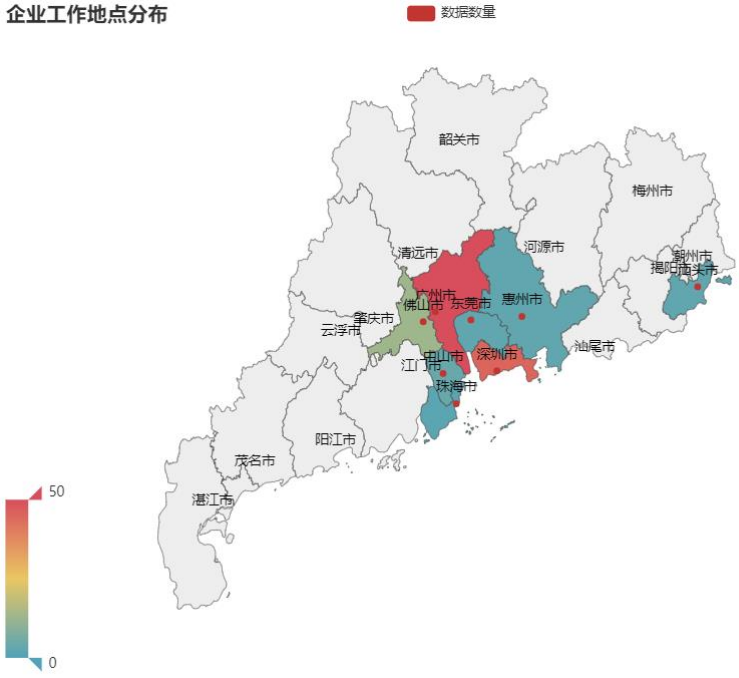

图 24 企业工作地点分布（广东省）

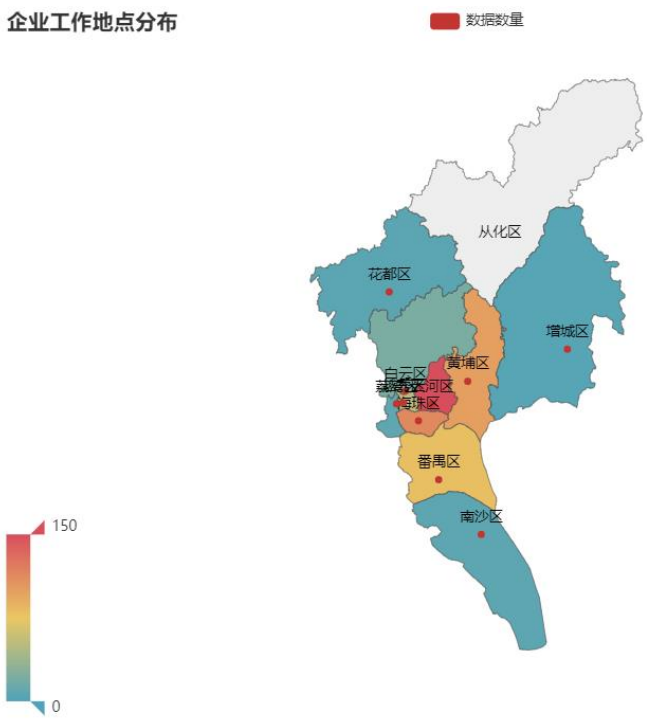

图 25 企业工作地点分布（广州市）

我们使用 pyecharts 库对招聘岗位地址、求职者居住地址及期望工作地址进行可视化，以更好地理解招聘市场的地理分布情况。通过绘制地点分布图，我们可以发现招聘企业主要集中在广东省和四川省。而在广东省内部，主要集中在广州市和深圳。此外，通过观察图表，我们可以看到天河区的企业更为集中，而在市区向外逐渐减少。

5.4.8 公司规模

除此之外，我们可以通过对公司员工数量进行分析，来了解公司的规模情况。根据图 26 所示，约有 80%的公司员工人数集中在 50-100 人之间，这反映了这一规模段的企业较为常见。相比之下，员工人数少于 50 人的公司和员工人数多达 10000 人的公司则较为罕见。说明在互联网行业中，中小型企业数量占主导地位，而规模较大或较小的公司数量相对较少。

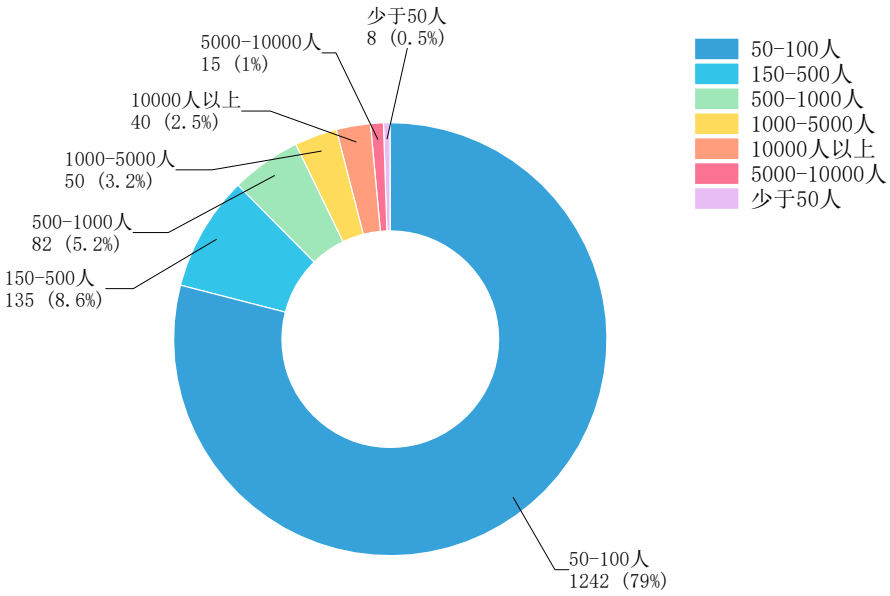

图 26 公司员工数量分布

5.4.9 工作经验需求

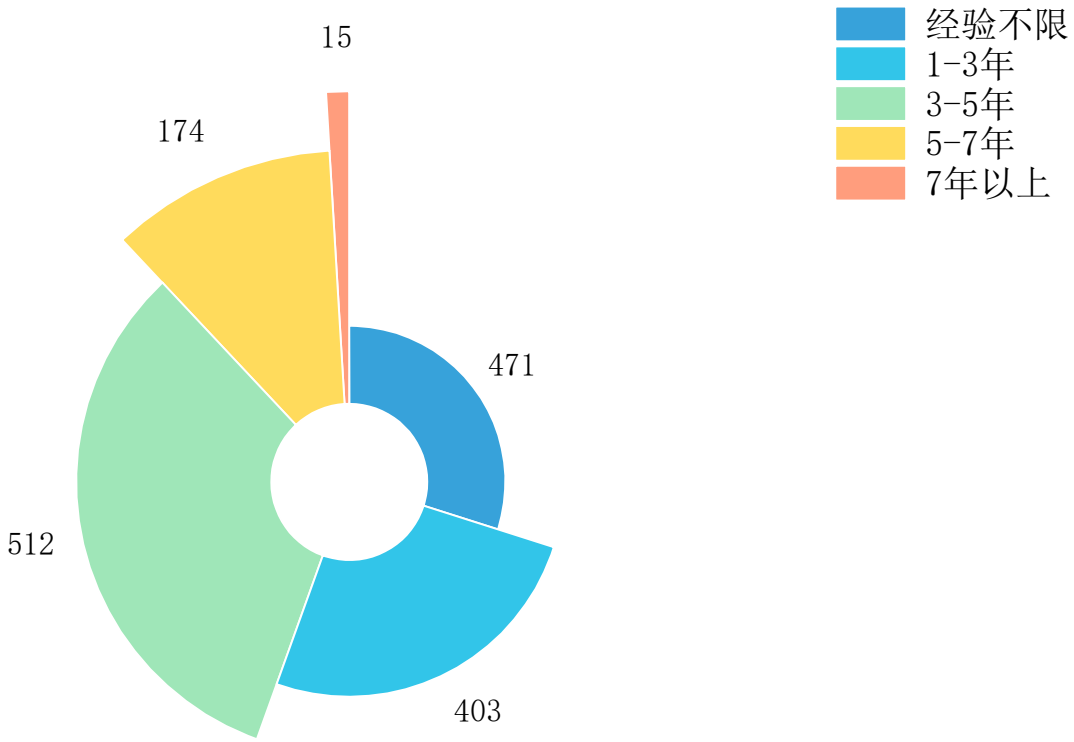

图 27 工作经验需求分布

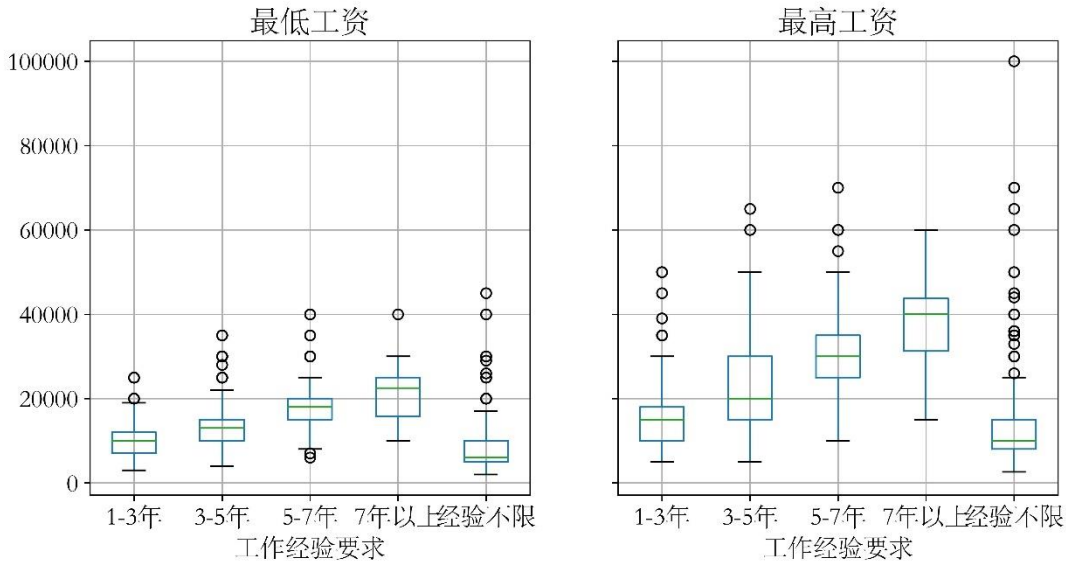

图 28 工作经验要求-薪资分布箱线图

招聘岗位总体的工作经验要求集中在 3-5 年，极少部分公司会要求求职者有多于 7 年的工作经验。图 28 可以看出，随着工作经验要求的增加，岗位提供的最低和最高工资也在提高。

5.4.10 岗位福利

图 29 岗位福利词云图

可以看到企业主要提供专业培训和补贴、奖金等。

5.5 求职者画像

根据采集求职者求职信息，从预期岗位、薪资需求、知识储备、学历、工作经验等多个方向建立求职者画像。

5.5.1 预期岗位

统计求职者预期岗位数据分布，如图 30 所示。由于“僵尸数据”的影响，“数据分析师”和“数据挖掘工程师”的占比最多。

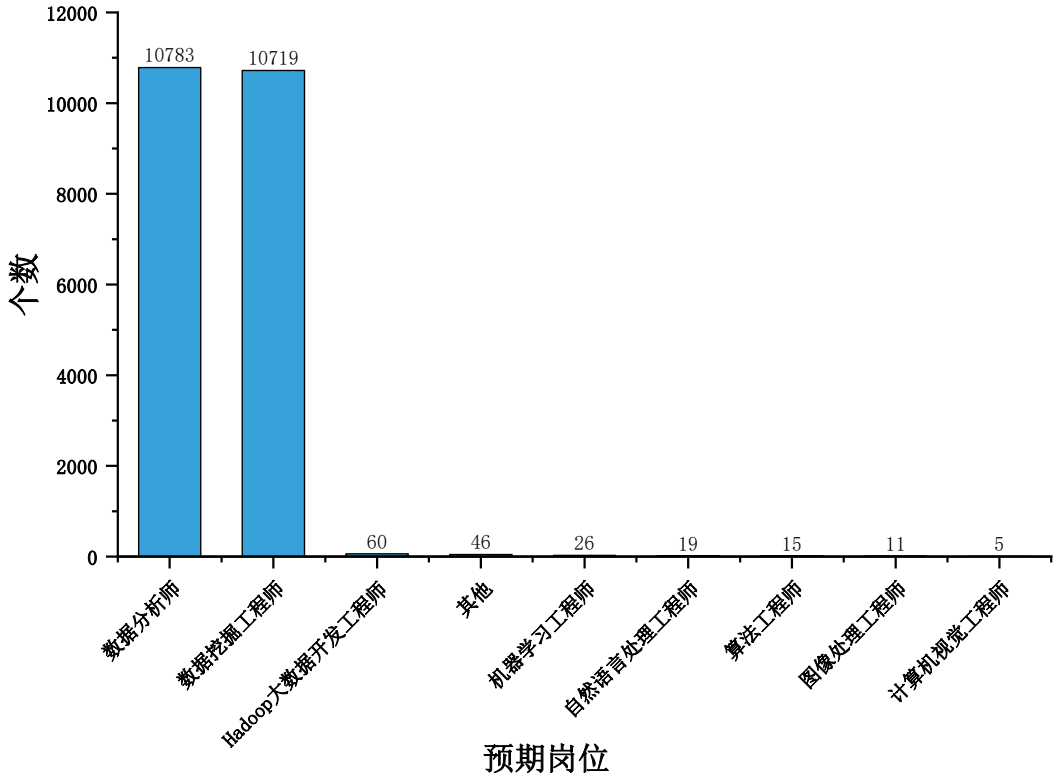

图 30 预期岗位分布（未处理）

处理“僵尸”数据后，预期岗位的分布如图 31 所示。数据分析师、数据挖掘工程师数量最多，而机器学习等算法工程师较少。

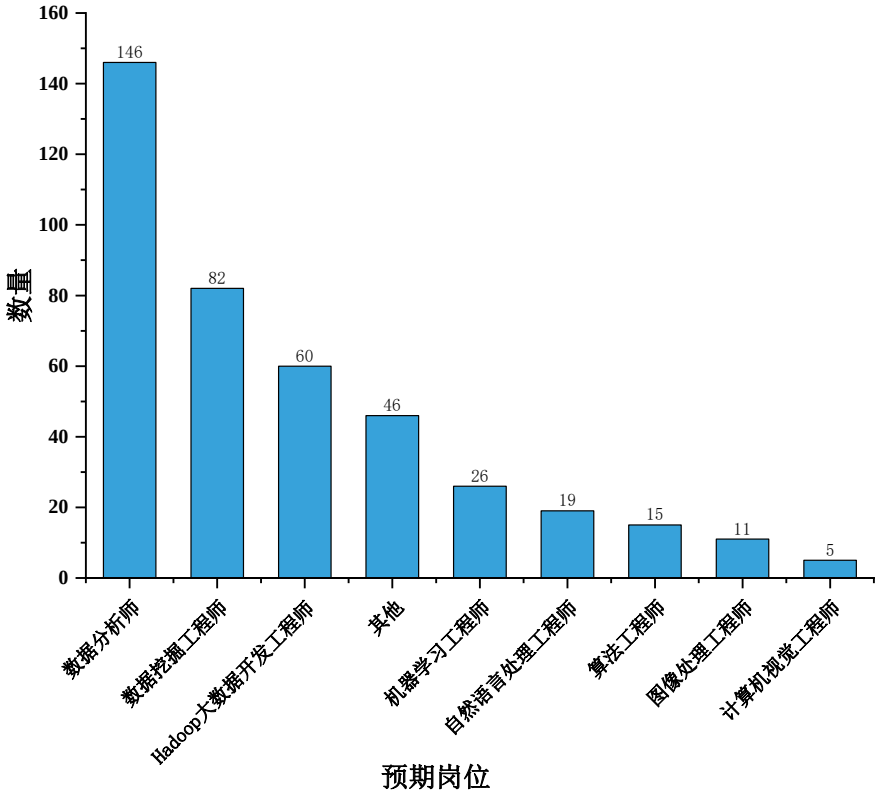

图 31 预期岗位分布（处理后）

5.5.2 薪资需求

可以看到，由于“僵尸”数据的期望薪资最低为 4k，最高为 6k，所以整体箱线图分布几乎为一条直线。

右图去掉后可以观察到求职者的期望薪资大部分在 10k 以内。

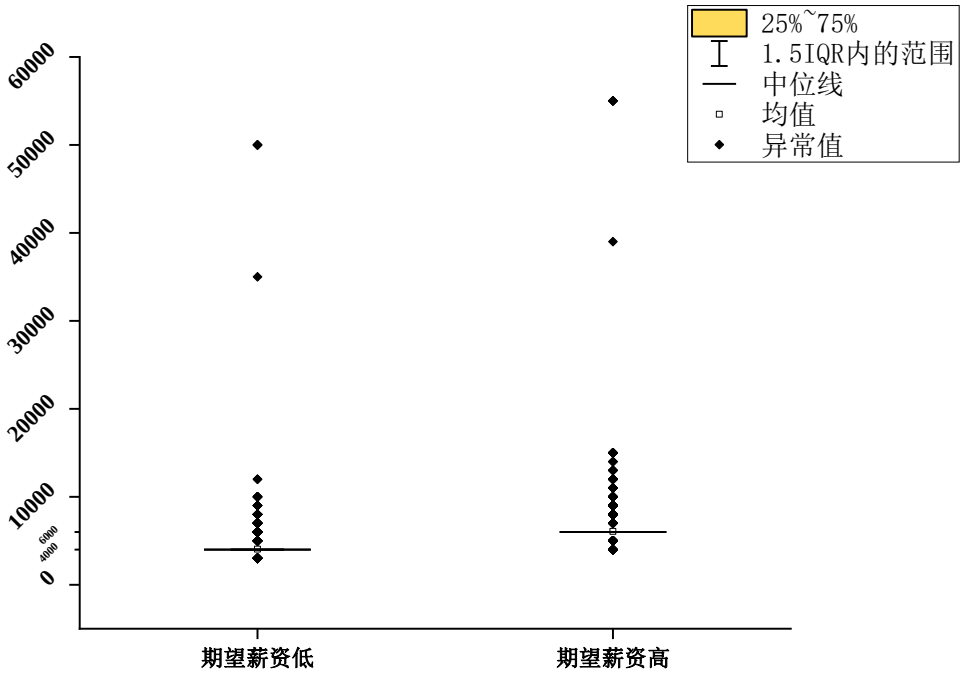

图 32 薪资需求箱线图（处理前）

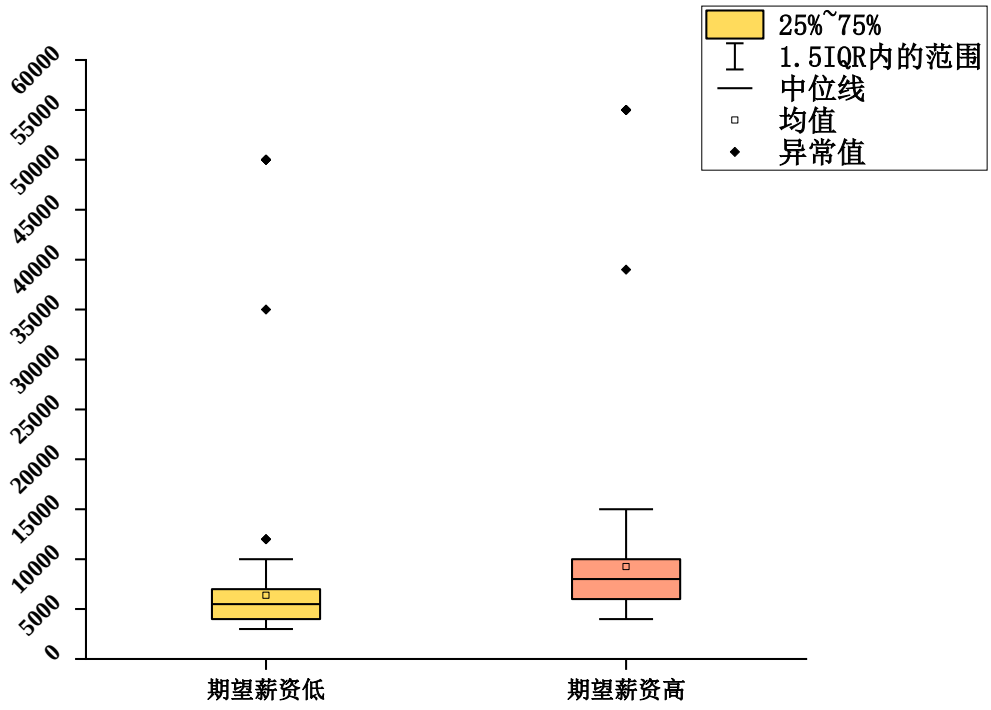

图 33 薪资需求箱线图（处理后）

5.5.3 知识储备

根据 5.2 节中数据处理得到的求职者知识储备关键词，进行词云可视化，结果如图 34 所示。

图 34 求职者知识储备词云图

由词云图可以看出，求职者知识储备多数为“python”、“开发”、“sql”和“数据分析”。这与招聘信息中岗位技能要求也相匹配。

5.5.4 学历

从图 35 可以看出，90%以上的求职者学历在本科及以上，与招聘信息的学历要求比例吻合。

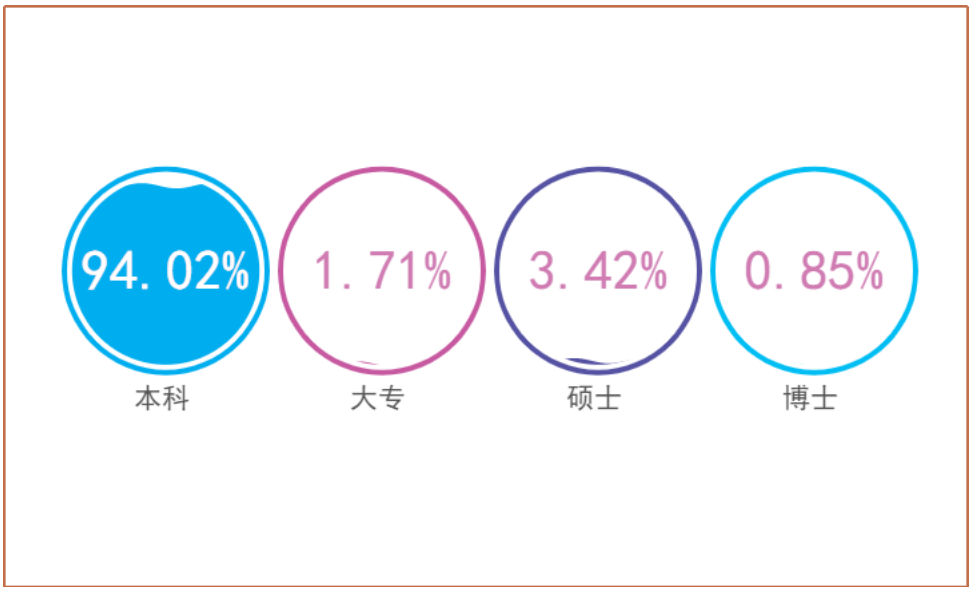

图 35 学历分布水波图

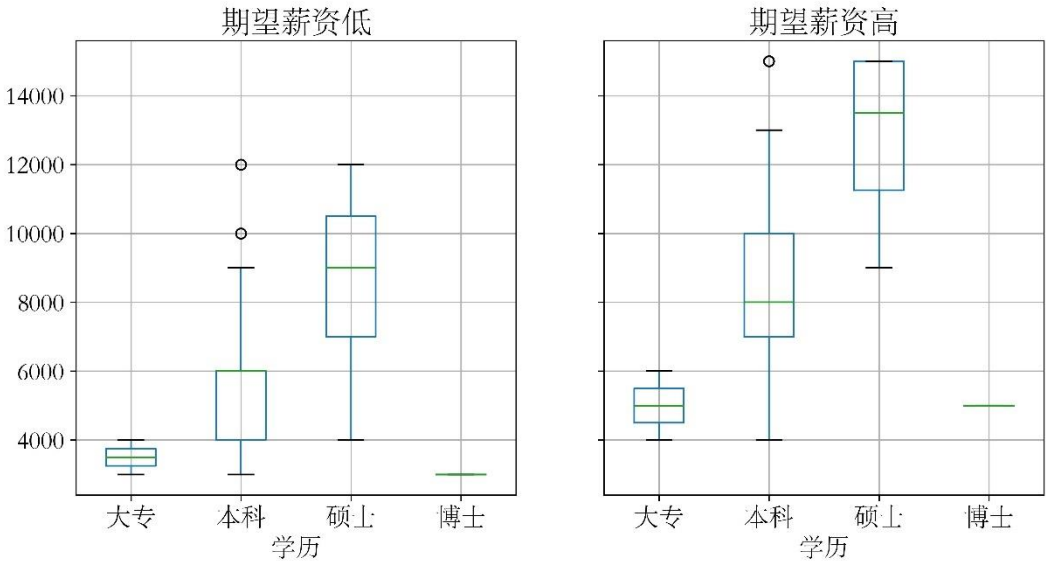

图 36 期望薪资-学历箱线图

根据图 36，除博士数据外，期望薪资随学历的升高而增大，这也符合人们现实中的想法。

5.5.5 工作经验

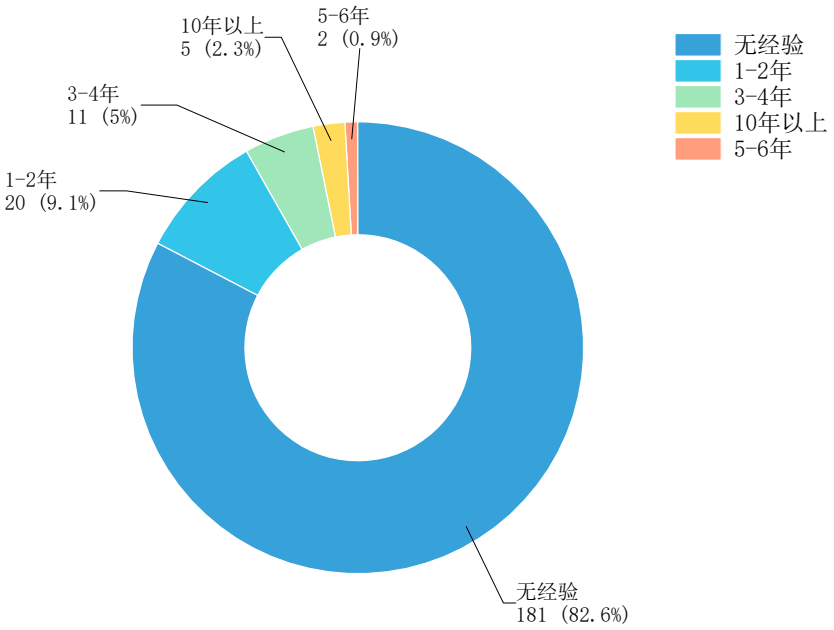

图 37 工作经验分布图

由图 37 可以看出，本数据集中大部分求职者没有工作经验。

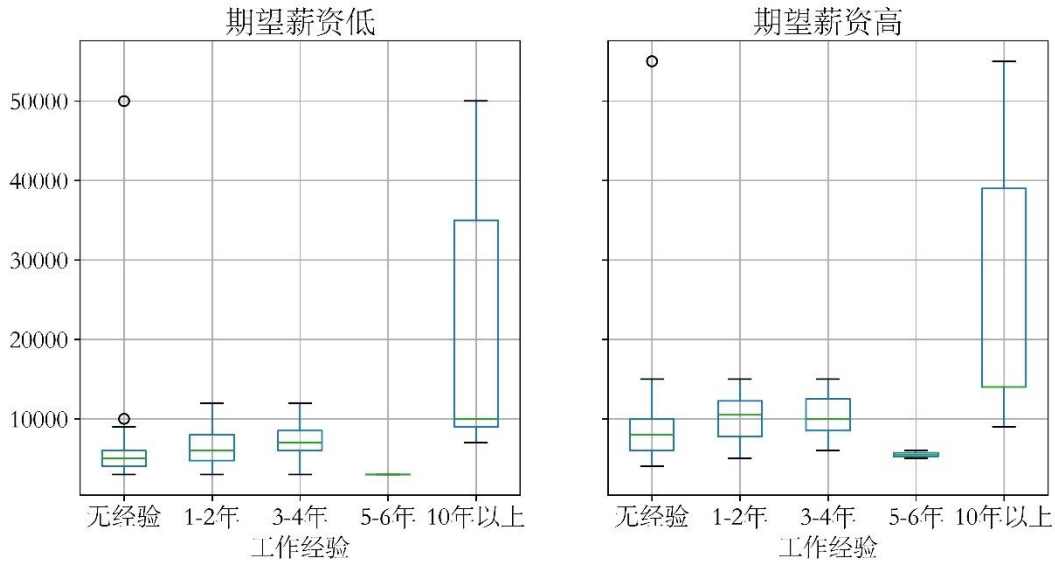

图 38 工作经验-薪资分布箱线图

由图 38 可以看出，除部分离群点外，期望薪资总体随工作经验的增加而提高，这与现实中工作年份越久，越有项目经验，能力越强一致。

5.5.6 居住地和工作地

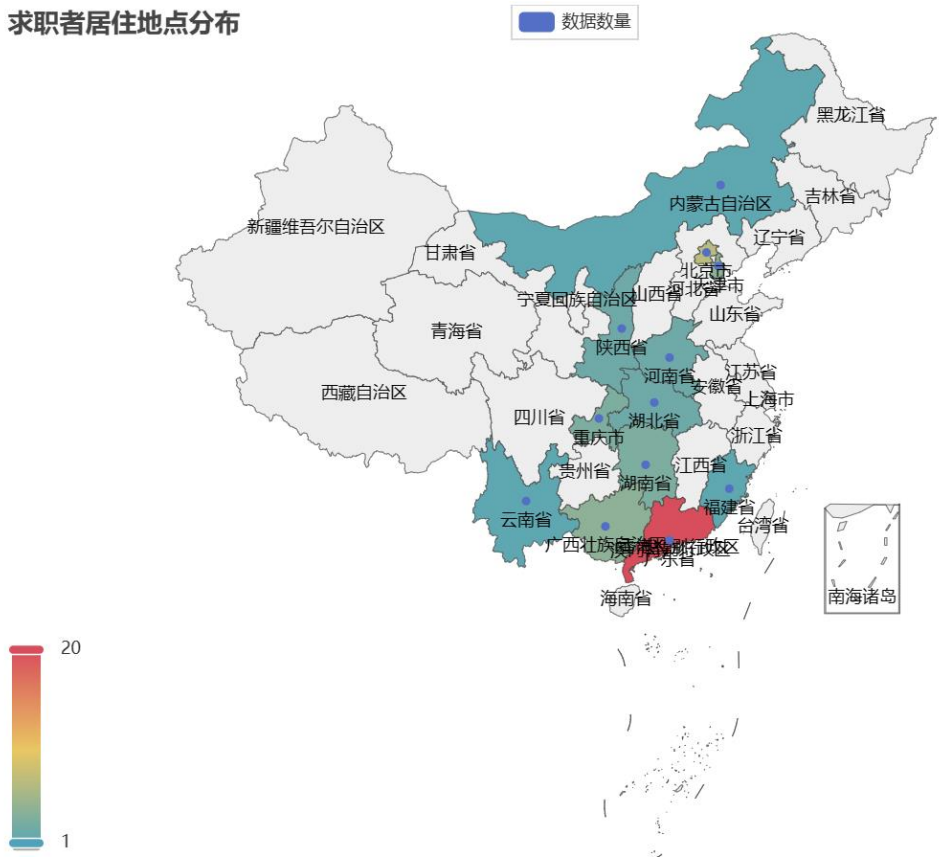

图 39 求职者居住地点（省）

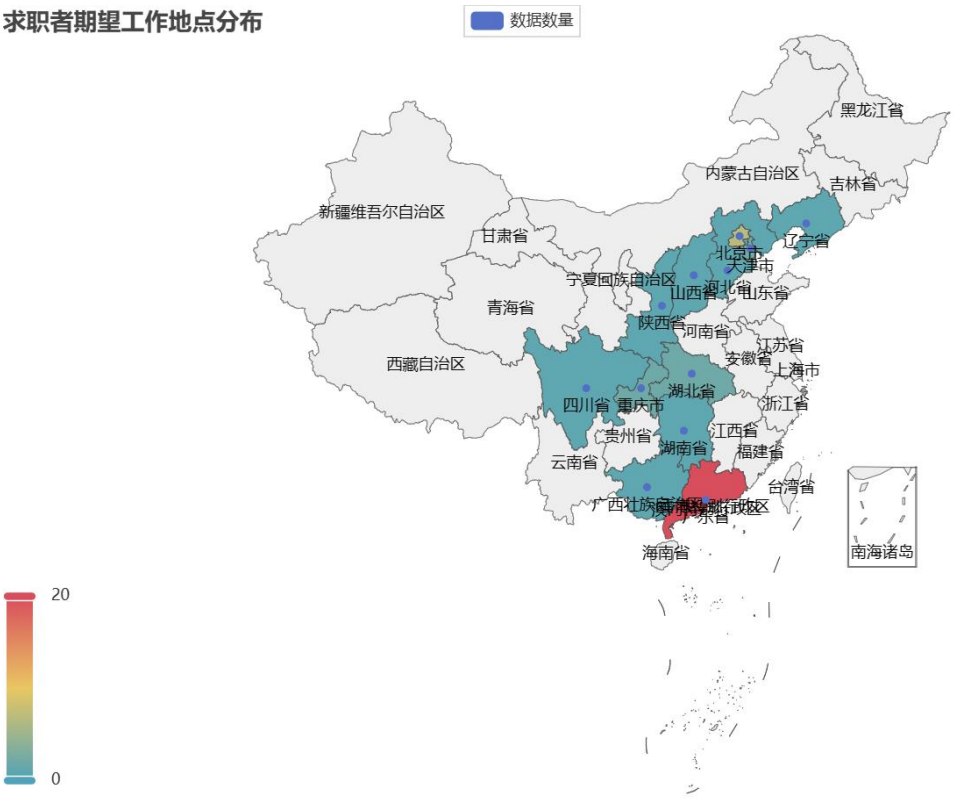

图 40 求职者期望工作地点（省）

从求职者居住地址和期望工作地址的分布图可以看出，求职者的居住地点和工作地点基本吻合，主要集中在广东省，并且求职者的期望工作地点与企业的分布有一定的重合度。下面主要分析求职者的期望工作地点。

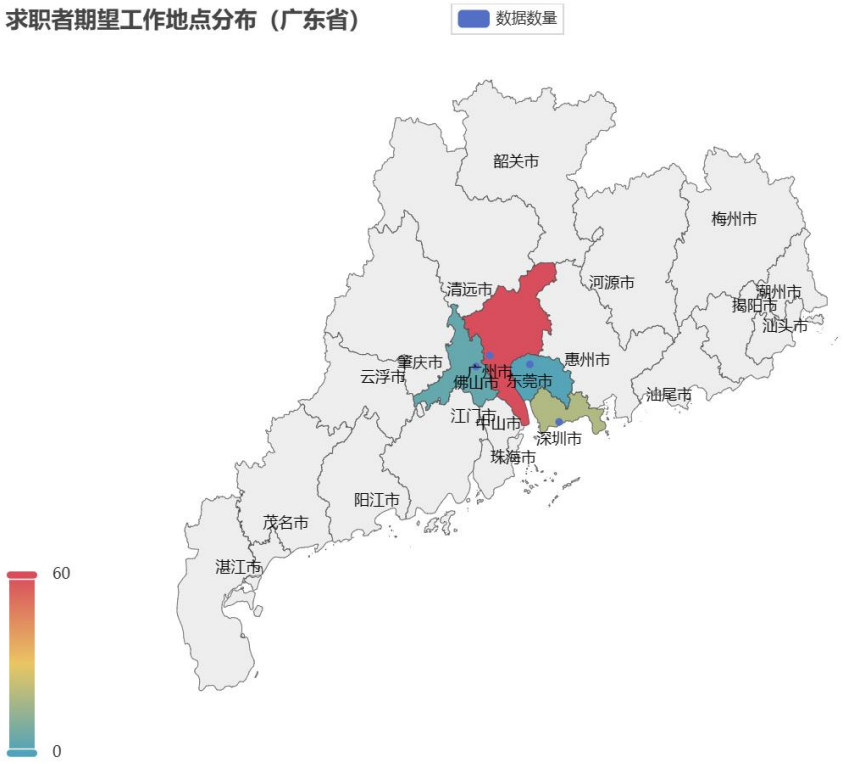

图 41 求职者期望工作地点（广东省）

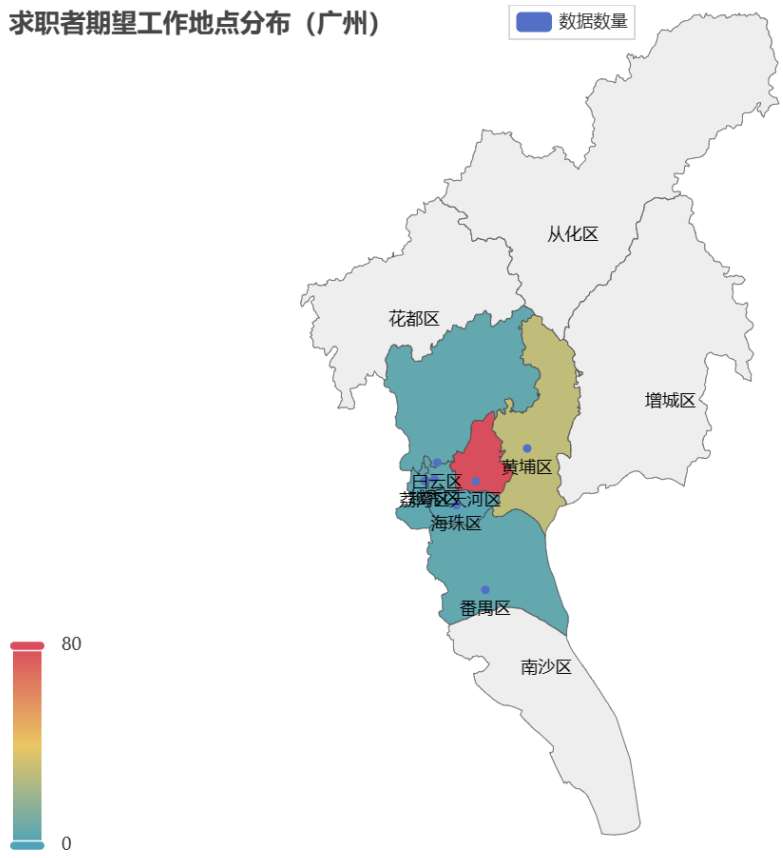

图 42 求职者期望工作地点（广州市）

由图 41 可以看出广东省求职者大部分期望在广州市工作，进一步查看广州市求职者期望工作地点分布，由图 42 可知期望工作地点以天河区为中心，逐渐向四周减少，这也和广州市市区位置以及企业工作地点集中于天河区有关。

5.6 模型应用

5.6.1 LDA 主题模型

本文 LDA 模型使用 5.3.1 节介绍的 TF-IDF 值构建矩阵，以得到更好的主题提取效果。

图 43 岗位 LDA 主题模型

对招聘岗位技能要求关键词进行 LDA 关键词提取。将技能要求关键词转换为主题词袋，并求其 TF-IDF 值，将数据传入 LDA 模型进行主题词提取。通过对不同主题数结果进行比较分析，认为主题数为 4 时的聚类结果具有较高的主题提取效果，对其各主题词类的主题词结果进行词云可视化，可以得出四个主题分别与岗位要求中的 Hadoop 大数据开发工程师、产品经理、数据分析师以及算法工程师一一对应。

表 7 招聘岗位技能主题词

主题词类	主题词 Top4
产品经理	产品设计、项目管理、大数据、需求分析
算法工程师	机器学习、深度学习、自然语言、python
Hadoop 大数据开发工程师	数据库、sql、mysql、hive
数据分析师	数据分析、数据、运营、电商

对求职者个人技能关键词进行 LDA 关键词提取，操作同招聘岗位技能要求关键词，分析不同主题数结果，主题数为 3 时聚类结果具有较高的主题提取效果，各主题词类的主题词词云可视化如下：

图 44 求职者 LDA 主题模型（处理前）

对于“僵尸数据”的处理，仅保留其中一条数据后再次进行 LDA 主题生成，当主题数为 3 时，主题提取效果有了显著的提高，各主题类的主题词词云图如下：

图 45 求职者 LDA 主题模型（处理后）

观察可知，经过处理后的求职者主题中，主题三的开发主题词比重有所降低，更加符合主题类别的语义，因此经过处理后的求职者数据主题提取效果有所提高。

通过招聘岗位和求职者的主题聚类对比，可以看出在泰迪内推网中，算法工程师等相关岗位供不应求，结合数据集与实际分析，算法工程师往往要求硕士及以上学历，而求职者数据集仅有少量数据为硕士及以上学历，造成了以上结果。

表 8 求职者技能主题词

主题词类	主题词 top3
大数据开发	hadoop、java、hive
数据分析	python、数据分析、数据
产品经理	吃苦耐劳、认真负责、细心

5.6.2 K-modes 聚类模型

本文使用 KModes 聚类算法，对招聘岗位数据及求职者数据进行整体聚类，以获取各类别代表画像并进行分析。下面展示招聘岗位数据按照 5.4 节分析后划分的区间：

1. **对于招聘人数**，按 0-10、10-50、50 以上划分为小、适中、大三类。

2. **对于职位福利**，首先统计职位福利数，并将其按 0-3、3-6、6 以上划分为低福利、中等福利和高福利三类。

3. **对于工作经验**，表 9 展示了招聘信息的工作经验要求，发现存在仅有工作年限“数字”、工作年限“区间”以及“不限”工作经验三种情况。对于工作年限区间而言，仅需满足其下限即可符合工作经验要求；对于不限工作经验，相当于工作年限为 0。因此，在工作经验中首次出现的数字即为岗位对工作经验的最低要求。使用正则表达式获取工作经验中首次出现的数字，若没有数字出现即为不限工作经验，置为 0。

表 9 招聘岗位工作经验数据

招聘信息 id	工作经验要求
1482196148415496198	10
1507240831520735232	5-7 年
1506179537359208453	不限

4. **对于工作地点**，由于聚类可能会出现省份、城市、区县互不相符的情况，因此仅选取省份特征作为聚类特征。

5. 此外将上文对**招聘岗位的标签作为岗位关键词特征**，并将 LDA 主题词生成中的主题类别也作为岗位聚类类别特征加入。

处理后的数据如表 10 所示：

表 10 招聘岗位离散化处理结果

招聘信息 ID	薪资区间	工作经验要求	招聘人数	职位福利	…	岗位关键词	岗位聚类类别
1613439889204969472	2k-4k	0	需求大	低福利	…	自然语言处理工程师	算法工程师
1613439536044572672	2k-4k	0	需求大	低福利	…	其他	产品经理
1613439183576236032	2k-4k	0	需求大	低福利	…	其他	算法工程师

由于 K-modes 算法使用众数代替了 K-means 算法中的均值，若存在许多相同的数

据，有可能会影响 K-modes 的聚类效果，故去除求职者数据中的大量“僵尸”数据。求职者数据的处理同招聘岗位，最终数据如下表所示：

表 11 求职者离散化处理结果

求职者 ID	居住省份	期望工作省份	工作经验	期望薪资区间	到岗时间	…	预期岗位
1649221278801985536	广东省	广东省	0	4k-6k	随时到岗	…	数据库工程师
1648221000086716416	河南省	广东省	1	4k-6k	1 周后到岗	…	数据库工程师
1648848763151843328	云南省	广东省	10	20k 以上	时间待议	…	数据库工程师
1648774046462115840	重庆市	重庆市	0	20k 以上	时间待议	…	数据库工程师

5.6.3 招聘岗位聚类

对处理后的招聘岗位数据，使用 K-modes 算法对其进行聚类，并使用 SSE 评估聚类效果。

在 K-modes 聚类算法中，聚类后的误差平方和（SSE）中误差指的是类别匹配度，即海明距离，若两属性值相同即为 1，否则为 0，最后求和即为当前两样本点误差。根据不同 k 值得到的 SSE 绘制岗位聚类的肘部图，如图所示

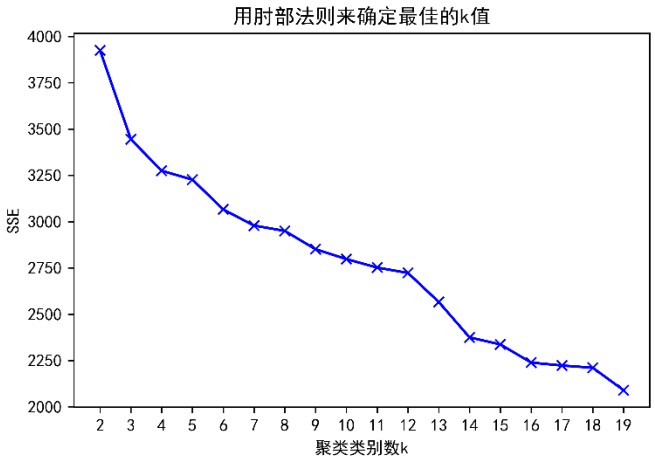

图 46 招聘岗位聚类 SSE 曲线

从图 46 中可以看到，当 k 从 4 增加到 5 时，SSE 下降幅度减缓，故选择将数据分为 4 个类簇，类簇的中心样本，即代表岗位画像如下表所示

表 12 招聘岗位类簇中心

类别	薪资区间	工作类型	学历要求	工作经验要求	招聘人数	职位福利	省份	岗位关键词	岗位聚类类别
0	10k-12k	全职	本科	3	小	中等福利	广东省	数据分析师	数据分析师
1	6k-8k	全职	本科	0	小	中等福利	广东省	其他	数据库工程师

2	14k-16k	全职	本科	5	小	中等福利	广东省	数据挖掘工程师	产品经理
3	20k以上	全职	本科	1	小	中等福利	广东省	算法工程师	算法工程师

可以看到，最终的四个聚类中心，聚类类别与 LDA 主题生成时的主题相对应，说明两模型效果较好，岗位可看作“数据分析师”、“Hadoop 大数据开发工程师”、“产品经理”、“算法工程师”四类，其中算法工程师薪资最高。

5.6.4 求职者聚类

求职者聚类操作同招聘岗位聚类，以下为求职者聚类的肘部图

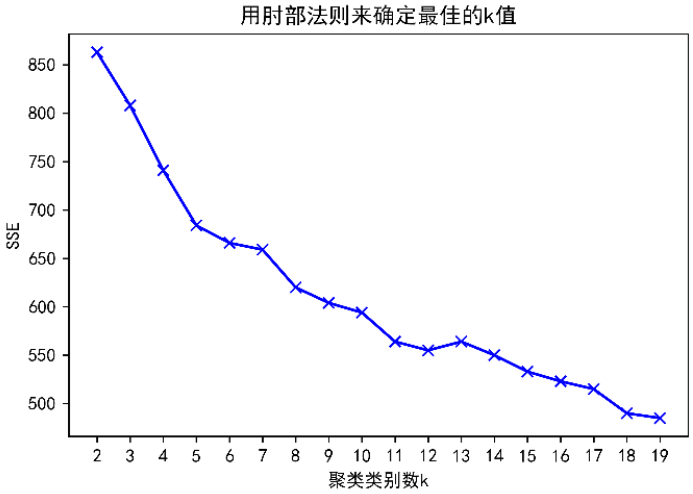

图 47 求职者聚类 SSE 曲线

从图 47 中可以看出，当 k 为 5 时，曲线下降速度明显减小，故将求职者划分为五个类簇。类簇的中心样本如下表所示

表 13 求职者类簇中心

类别	性别	年龄	居住省份	期望工作省份	工作经验	期望薪资区间	期望工作性质	学历	到岗时间	学科专业	求职者聚类类别
0	男性	25岁以下	广东省	广东省	0	6k-8k	全职	未知	时间待议	未知	数据分析师
1	女性	25岁以下	未知	未知	0	4k-6k	全职	本科	时间待议	信息与计算科学	数据分析师
2	男性	25岁-30岁	广东省	广东省	0	4k-6k	实习	本科	随时到岗	信息与计算科学	数据库工程师

3	女性	25岁以下	广东省	广东省	0	4k-6k	全职	本科	时间待议随时到岗	统计学	产品经理
4	女性	25岁以下	广东省	广东省	0	2k-4k	全职	未知	未知	未知	数据库工程师

六、 针对问题三的解决方案

6.1 工作思路

匹配度和满意度是影响岗位和求职者信息匹配程度的潜在因素，它们的计算取决于不同的评价标准。需要注意的是，这两个因素并不存在最优值需要去逼近，因此在问题三中不存在优化目标，也就无法为模型学习提供有效的指导。为了解决这一问题，我们查阅了人岗匹配行业内的模型计算公式，并且结合自己的需求和数据特点，为每个招聘岗位和求职者计算匹配度和满意度得分。

岗位匹配度是企业考虑求职者掌握的技能、预期岗位、薪资需求、学历和工作经验等个人情况，多方面评估求职者是否符合招聘要求的匹配程度。我们结合该岗位对于不同能力维度的要求权重，通过计算求职者在各个能力维度的水平等级，得到一个综合的分数以衡量与岗位要求的匹配程度。

求职者满意度则是求职者对于招聘岗位的满意程度，需要考虑招聘岗位与自我预期是否匹配、所提供的薪资待遇和福利等信息。我们结合求职者对于不同需求维度的重视程度，通过计算招聘岗位在各个维度的得分情况，得到一个综合的分数以衡量求职者的满意程度。

综上，我们基于爬取的信息为每个招聘岗位和求职者计算一个匹配度和满意度得分，得分越高表示匹配度（满意度）越高。这为我们更好地进行岗位推荐和匹配提供了依据。

6.2 模型设计

根据文献[3]，查阅到胜任力模型可以提升组织的特定工作水平，为岗位角色有效服务，已成为人力资源管理的通用工具；胜任力模型强调岗位达到高绩效目标所需要的行为、技能和特定知识，并提供了一个共同的参照标准。其中岗位胜任力模型可计算岗位匹配度，公式如下：

$$岗位匹配度 = \frac{\sum(岗位胜任力要求 \times 个人能力水平)}{\sum(岗位胜任力要求)}$$

其中，岗位胜任力要求指的是该岗位中要求所必须具备的能力、技能、经验等，通过不同岗位间要求的比较确定对应等级；个人能力水平指的是个人在这些能力、技能、经验等方面的表现水平，通过对求职者能力间的不同以及对岗位所需要求的匹配来确定等级。对于岗位匹配度，需综合多个能力维度对求职者进行考虑评估。本文依照上述内容建立岗位匹配度模型。

对于求职者满意度模型，参考岗位匹配度模型，考虑多个需求维度上求职者的期

望要求与岗位的满足程度，建立相应的求职者满意度模型。

为使匹配度模型中不同求职者或满意度模型中不同岗位有一个统一的比较标准，设定个人能力水平以及岗位满足程度的等级最大值为 5，则若各能力维度或需求维度中，个人能力水平或岗位满足程度均为最大值，最终的匹配度值或满意度值即为 5，即最终得分值得范围被限定在 0 到 5。

由于不同岗位的胜任力要求不同，因此每个求职者的岗位匹配度都是针对于某一岗位而言的，且不同求职者的比较仅限于在同一岗位胜任力要求下才可进行比较，因此对同一岗位中的求职者的岗位匹配度值进行归一化，得到最终的岗位匹配度，针对某一岗位的岗位匹配度公式如下：

$$p_{job,i} = \frac{v_{job,i}}{max\ v_{job}}$$

求职者满意度模型同上，计算公式如下：

$$p_{people,i} = \frac{v_{people,i}}{max\ v_{people}}$$

6.3 建立匹配度模型

对于岗位匹配度模型，考虑学历、薪资、工作经验、技能、工作地点以及工作类型六个能力维度。

6.3.1 学历

胜任力要求：对于学历数据，由于学历数据本身就是序数变量，因此可直接对其进行等级赋值，对于岗位而言，学历要求越高代表学历维度的胜任力要求越高，求职者满足时的匹配度更大，因此学历的胜任力等级与学历高低成正比。本文模型直接将学历从 1 到 5 进行赋值，其中不限与技工对学历要求均较低，因此赋值为 1，等级划分如表 14。

表 14 学历等级映射表

学历	学历胜任力等级
博士	5
硕士	4
本科	3
大专	2
技工	1
不限	1

个人能力水平：求职者学历水平评分与胜任力要求中的评分标准相同，对于学历缺失样本将其归入不限类型。

6.3.2 薪资

胜任力要求：薪资数据为连续数据，为使其方便模型计算，将薪资数据离散化为 1 到 5 共五个等级。招聘岗位数据中有最高薪资与最低薪资，薪资区间中位数往往可以较好地反映岗位的薪资水平，因此将其作为胜任力要求的划分标准。获取所有岗位薪资的区间中位数，计算其五分位数，根据五分位数的值给每个岗位的薪资胜任力等级进行赋值。

个人能力水平：对于岗位而言，本文认为求职者的最低期望薪资不会被关注，而只会在意求职者的最高期望薪资；当求职者最高期望薪资超出岗位最高薪资时，表明

求职者自身能力与岗位要求不匹配，因此将求职者最高期望薪资m_{high}^{people}超出岗位最高薪资m_{high}^{job}的部分在求职者期望薪资区间的占比作为衡量个人能力水平是否匹配的标准，计算公式为：

$$m = \frac{m_{high}^{people} - m_{high}^{job}}{m_{high}^{people} - m_{low}^{people}}$$

当求职者最高期望薪资低于岗位最高薪资时，直接赋值为 0。

以上个人能力薪资公式中，当求职者最高期望薪资超出岗位最高薪资占求职者期望薪资区间比例越多时，m越高，匹配度越低。由于限制个人能力水平最高为 5，因此将m归一化为权重，当m大于等于 1，即求职者期望薪资完全高于岗位薪资时，将其置为 1。该权重越大，则匹配度越低，故个人能力薪资水平公式为

$$m_{score}^{people} = (1 - m)5$$

6.3.3 工作经验

胜任力要求：由于工作经验数据本身即为序数变量，故可直接将其作为胜任力等级，观察到工作经验中存在 0，而胜任力要求作为评分权重为 0 时无意义，故对所有工作经验等级加一。

个人能力水平：对于求职者工作经验，观察数据分布，发现仅有 0、1、3、4、10 五个数据值，与个人能力水平等级一一对应。但是工作经验要求在求职中往往是硬性要求，因此当求职者工作经验小于岗位工作经验要求时，直接赋值为 0，否则将求职者工作经验对应等级作为个人能力工作经验水平分数。

6.3.4 技能

胜任力要求：在前文中，经过分词与关键词提取处理，已获得了岗位技能要求关键词，对于岗位来说，技能要求越多，说明匹配难度越高，匹配时的匹配度权重就会越多，故可以直接使用岗位技能要求关键词的词数作为等级。由于数据中存在岗位没有技能要求关键词的情况，同工作经验一样，对所有技能要求等级加一。

个人能力水平：个人能力关键词也已获取，对于岗位要求的技能，求职者个人能力应越满足越好，即两关键词集合交集长度与岗位技能要求关键的的比值应越大越好，以上处理得到一个权重，再归一化至个人能力水平分数区间，公式如下：

$$s_{score}^{people} = \frac{\left|S^{job} \cap S^{people}\right| + 1}{|S^{job}| + 1}5$$

6.3.5 工作地点

胜任力要求：由于不同岗位间的工作地点没有可比较性，因此胜任力等级直接取前几个能力维度的均值。

个人能力水平：工作地点数据为省份、城市、区县，其程度是逐级递增的，因此对于求职者而言，满足程度越大的地点要求，匹配度也越大，故计算个人能力工作地点水平时直接判断个人工作地点与岗位工作地点是否匹配，均不满足时个人能力水平等级为 0，省份匹配时为 1，城市匹配时为 3，区县匹配时为 5。

6.3.6 工作类型

胜任力要求：同工作地点，不同岗位间的工作类型也没有可比较性，因此胜任力

等级直接取前几个能力维度的均值。

个人能力水平：由于工作类型仅有实习与全职两种，且结合现实情况而言，工作类型要求属于硬性要求，因此直接比较求职者与岗位要求是否相同，相同则直接将个人能力水平等级赋为满分 5，否则为 0。

通过以上各能力维度计算，依靠岗位匹配度模型即可得到求职者与招聘岗位对应的匹配度值，针对同一岗位下的应聘者，将其匹配度值归一化，得到最终的匹配度。

6.4 建立满意度模型

对于求职者满意度模型，考虑薪资、期望行业、工作地点、工作类型以及工作福利五个需求维度。

6.4.1 薪资

期望要求：求职者薪资期望要求处理与岗位匹配度模型中岗位薪资胜任力要求相同，不再赘述。

岗位满足水平：对于求职者而言，岗位的薪资越高越好，因此会更加关注岗位所给最低薪资是否符合自己要求；当招聘岗位最低薪资低于求职者最低期望薪资时，表明岗位提供薪资与求职者期望不匹配，因此将求职者最低期望薪资m_{low}^{people}超出岗位最低薪资m_{low}^{job}的部分在求职岗位薪资区间的占比作为衡量岗位是否满足求职者需求的标准，计算公式为：

$$m = \frac{m_{low}^{people} - m_{low}^{job}}{m_{high}^{job} - m_{low}^{job}}$$

当求职岗位最低薪资高于求职者最低期望薪资时，直接赋值为 0。

以上薪资满足水平公式中，当岗位的最低薪资低于求职者最低薪资的部分占招聘岗位薪资区间比例越多时，m越高，满意度越低。后续操作与匹配度模型中薪资维度个人能力水平相同，公式如下：

$$m_{score}^{job} = (1 - m)5$$

6.4.2 期望行业

期望要求：期望行业与岗位匹配度模型中的技能能力维度相似，均为判断关键词数量。观察求职者期望行业数据，知求职者对期望行业进行关键词描述，且其中存在不限要求。同技能能力维度，期望行业关键词越多，说明需求越严格，计算满意度时权重更大，故以其关键词词数作为期望要求等级，并全部加一。但对于关键词中出现不限的，说明该求职者对期望行业要求较低，故出现不限一词的，期望行业要求等级均置为最低值 1，并将关键词直接去除。

岗位满足水平：对于期望行业关键词，招聘岗位有对应企业类型数据，为当前岗位的行业关键词。同岗位匹配度模型中技能能力维度一致，判断两者关键词间交集，当交集为空时，说明求职者对期望行业无要求，故直接将等级赋为满分 5 分，否则根据交集大小与求职者期望行业关键词次数的比值作为权重，得到最终等级，公式如下：

$$f_{score}^{job}\begin{cases}\dfrac{\left|F^{people}\cap F^{job}\right|+1}{\left|F^{people}\right|+1}5 & \left|F^{people}\cap F^{job}\right|\neq 0\\ 5 & \left|F^{people}\cap F^{job}\right|=0\end{cases}$$

6.4.3 工作地点

期望要求：处理同匹配度模型一样，直接取前几个需求维度的均值。

岗位满足水平：处理同匹配度模型中工作地点维度一致，不再赘述。

6.4.4 工作类型

处理同匹配度模型中工作类型维度一致，不再赘述。。

6.4.5 工作福利

期望要求：由于求职者数据集中没有明确的工作福利期望特征，故同工作地点、工作类型一致，使用前几个需求维度的均值作为等级。

岗位满足水平：由于求职者数据中没有对岗位福利的要求，无法对比获取满意度。根据现实实际，工作福利越多，求职者满意度相应会更高，因此直接统计招聘岗位工作福利的福利数，并与岗位福利最大值相比得到权重，归一化到 0 到 5 的岗位满足度分数中。

通过以上各需求维度计算，依靠求职者满意度模型即可得到求职者与招聘岗位对应的满意度值，针对同一应聘者获取的不同招聘岗位 offer，将其满意度值归一化，得到最终的满意度。

6.5 匹配度、满意度求解

在前文中，我们将招聘岗位按照泰迪内推网标签划分为九个类型，在求职者数据中也有期望岗位数据与九个标签相对应。基于实际生活，本文认为招聘岗位类型与期望岗位为最重要的匹配特征，若岗位类型与求职者期望岗位不符，即不满足题目中的“最低要求”，相应的匹配度和满意度赋值为 0。

因此在计算匹配度与满意度时，分别对同一类型下数据进行计算，而不同类型间岗位与求职者的匹配度、满意度认为为 0。将划分好的九个类型分别传入匹配度满意度模型进行求解，得到九个类型下岗位与求职者的匹配度和满意度。

需要注意的是，由于类别标签相互间有所重复，同一招聘岗位可能分到多个类型中，同时，求职者会有多个期望岗位，因此也可能分到多个类型中，这会造成岗位与求职者匹配度、满意度的重复，在这种情况下本文选择匹配度、满意度最高的数值作为结果数据。

将九个类型的数据整合为最终数据，去除岗位与求职者重复的数据，得到最终结果部分数据如下表所示

表 15 岗位匹配度结果表

招聘信息 ID	求职者 ID	岗位匹配度
1374177417123467264	1469233133072285696,	1
1374177417123467264	1468148406491938816,	0.921875
1374177417123467264	1468114577727291392,	0.90625
1374177417123467264	1472869555029278720,	0.828125
1374177417123467264	1550096565136392192,	0.8125
1374177417123467264	1469243148147490816,	0.75

表 16 求职者满意度结果表

求职者 ID	招聘信息 ID	公司名称	求职者满意度
1461512488951611392	1461587935500566528	广州广电运通信息科技有限公司	1
1461512488951611392	1461621784037163008	广东铭太信息科技有限公司	1
1461512488951611392	1461584951286824960	广州思迈特软件有限公司	1
1461512488951611392	1461585230782660608	广州思迈特软件有限公司	1
1461512488951611392	1461595991152132096	艾瑞咨询集团	0.949596774
1461512488951611392	1463081025323270144	中软国际	0.927419355
1461512488951611392	1463027273123889152	中数通信息有限公司	0.927419355
1461512488951611392	1461591923750993920	上海众言网络科技有限公司	0.903225806

七、 针对问题四的解决方案

7.1 工作思路

通过 2.4 节的分析，我们初步确立问题四的主要目标是构建向岗位推荐求职者的模型，以最大化招聘流程中的履约率。

通过爬取的招聘岗位信息，我们可以确定参加招聘的求职者人数和拟聘岗位人数之和，这意味着履约率存在一个极限。在每一轮推荐中，我们需要为每个招聘岗位选择合适的求职者，因此需要采用适当的算法来调整推荐的顺序，以逼近履约率的最大值。因此，我们需要先构建一个求职者的排序模型来实现这一目标。

对于构建排序模型，我们首先采用迁移学习的方法，引入与本数据集高度相似的公开数据集向本数据集进行对齐，作为训练数据来学习特征，以便在目标域推荐中使用。我们比较了多种机器学习方法，如随机森林、决策树以及集成学习模型 LightGBM 来预测排序，发现 LightGBM 在源域数据集上的表现最优。因此，我们在源域数据集上进行模型参数的学习后，将效果最好的预训练模型应用到目标数据集中。

在获得推荐序列后，我们需要考虑到不同岗位可能对应着相同的求职者顺序，这种情况可能导致多个岗位同时向同一批求职者发放 offer，降低整个推荐流程的履约率。为了解决这个问题，我们采用了一种贪心策略来进行求解。具体来说，我们从增加随机数、相邻判别、考虑 offer 数量三个方面提出了解决方案，并取得了较好的结果。

7.2 排序模型设计

对于推荐一个更适合的排序，我们首先尝试直接使用问题三得到的岗位匹配度和

求职者满意度简单的加权求和的结果进行排序，即

$$rank_i = \alpha p_{job,i} + (1-\alpha) p_{people,i}$$

再尝试使用迁移学习的方法提供模型学习特征输出最优排序。

7.2.1 迁移学习

由于本题的招聘岗位、求职者数据集存在缺少标注、数据分布不平衡、质量参差不齐等问题，无法直接用于监督学习模型的训练。本文引入与目标与高度相似的公开数据集作为训练数据，并将学习到的信息应用于目标域推荐中。

7.2.1.1 源域数据集介绍

本文所用源域人岗数据集来源于 2021 年全球开放数据应用创新大赛[4]，脱敏数据由深圳市人才集团有限公司提供。源域数据集给出了训练集和测试集。人岗数据集的输入中有 32 个特征，包括工作年限、最高学历等。表 17 列举了人岗数据集的部分特征字段以及其含义。

表 17 特征字段及含义

字段名	说明
WORK_YEARS	工作年限
HIGHEST_EDU	最高学历
KEY_TECHNOLOGY	关键技术
JOB_TITLE	招聘职位
MAJOR	对应聘者的专业要求
LOCATION	工作地点

7.2.1.2 源域到目标域的迁移

由表 17 可知，源域数据集中包含的特征与目标域中存在特征基本一致，两者均为岗位匹配数据，可以推断源域和目标域的数据分布本身具有较高的相似性。

对源域数据提取两数据集共有特征，并以 5.2 节的数据处理操作对源域数据集特征进行处理，确保模型能从源域数据中学习到目标域的相关特征。

7.2.2 LightGBM 算法

本文使用 LightGBM 算法作为推荐预测算法。LightGBM [5]是由微软亚洲研究院团队开源的基于决策树算法的梯度提升框架。随着网络招聘的流行，岗位数据和人才简历数据的规模变得越来越巨大，传统 GBDT 面对高维大数据的问题，模型的训练速度以及准确性面临极大的挑战。LigthGBM 主要提出了 2 种改良算法来提升训练速度：直方图算法和单边梯度采样算法（Gradient One-Side Sampling，GOSS）[6]。

7.2.2.1 直方图算法

直方图算法是一种将连续的特征值离散化为若干个区间，然后在这些区间上寻找最优分割点的方法，相较于传统的预排序算法，直方图算法减少了分割点的搜索空间和内存占用，提高了计算效率；并利用了直方图做差的技巧，减少重复计算，加速训练过程。

7.2.2.2 单边梯度采样算法

单边梯度采样算法是一种基于梯度大小进行样本采样的方法，它保留了梯度大的样本，舍弃了梯度小的样本，减少了数据量，保持了信息增益的准确性；通过给梯度小的样本加权，平衡数据分布，避免偏差；可与直方图算法结合使用，进一步提升效率。

7.3 排序模型应用

在源域数据的训练集上，使用经过数据处理后的特征对 LightGBM 模型进行训练，将得到的“预测打分”按降序排序，取 15%的人岗匹配标签记为成功。在源域测试集上部分结果如下：

表 18 源域测试集结果

岗位编号	求职者编号	预测打分	预测标签	LABEL
43668705	630374	0.99942	1	1
42542695	5953494	0.99926	1	1
42513834	6235546	0.999191	1	1
43668705	196099	0.999165	1	1
43668705	3354867	0.998923	1	1

最终 LightGBM 模型的准确率达到 93.54%。

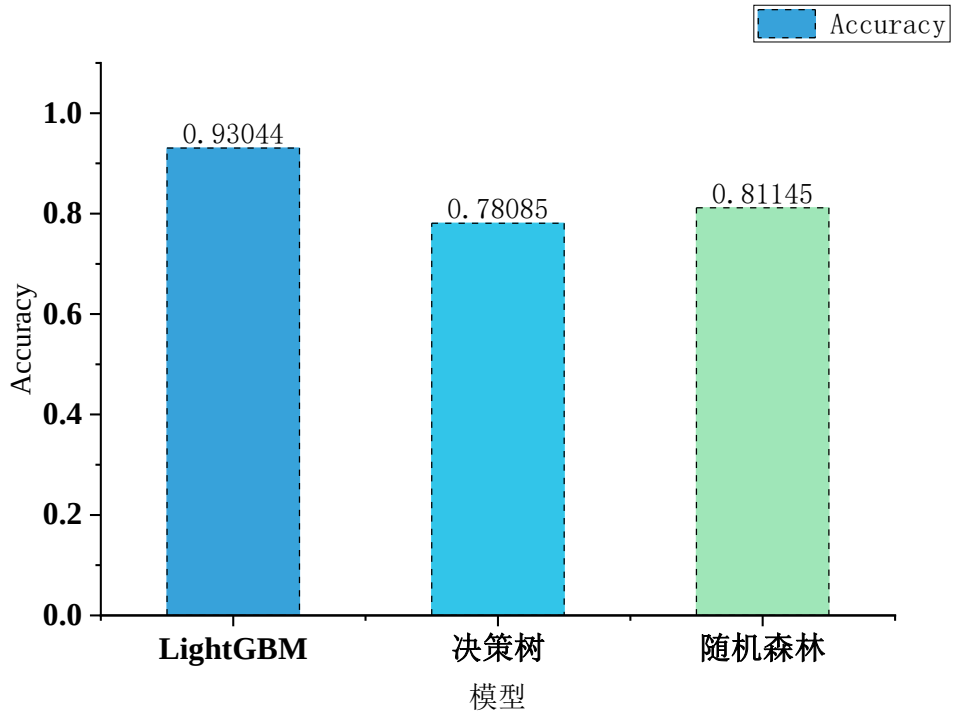

图 48 模型比较

将源域数据向目标域数据进行对齐后，调用不同的模型进行比较。图 48 表明经过特征筛选和处理后的模型在源域数据集上的预测性能也处于较高水准，证明 LightGBM 模型在人岗匹配度预测方面的性能较为良好。

同时，由表 18 可以看出，在源域数据集中训练得到的 LightGBM 模型输出为人岗是否匹配的 0、1 标签，而本任务仅需获得推荐排序即可，故取模型的“预测打分”作为排序标准。使用在源域数据集上预训练的 LightGBM 模型对目标域数据集中的所有岗位及求职者进行匹配预测，根据匹配度结果排序得到每个岗位的求职者推荐序列。部分结果见表。

表 19 求职者推荐序列分数

岗位编号	求职者编号	预测打分
1482195175148224552	1647944223779061760	0.9938862
1482195175148224552	1461530285551255552	0.993370642
1531566491487567872	1647944223779061760	0.993335075
1482195175148224552	1648345891960127488	0.993309941
1482188020965834752	1647944223779061760	0.993150563

将该序列作为人岗双向推荐的推荐序列进行推荐流程模拟，计算得到履约率。

7.4 贪心策略的推荐模型

在得到推荐序列后，我们需要考虑到不同岗位对应的求职者顺序可能相同的情况。这种情况会导致多个岗位同时向同一批求职者发放 offer，从而降低整个推荐流程的履约率。为了解决这个问题，我们采用了一种贪心的策略来进行求解。该算法的基本思想是，每一步都选择当前状态下最优的解决方案，以期望最终得到全局最优解。在我们的实现中，我们会根据每个岗位的需求以及求职者的优先级顺序进行比较，并选择能够最大化推荐效果的求职者来填补 offer。这样做不仅能够避免多个岗位同时向同一批求职者发放 offer 的问题，而且还能够提高整个推荐流程的履约率，从而提高整体的效益。

7.4.1 计算履约率极限值

在推荐流程中，除了招聘信息中的“招聘人数”可以为“不限”这一未知量外，其余的所有参数都是已知的。为了确保招聘流程的顺利进行，我们进行了一系列的消融实验。在这些实验中，我们尝试不同的数值作为“不限”情况下的具体招聘人数，以观察不同取值对于履约率的影响。结果表明，在将“招聘人数”设为 285 时，我们的履约率达到了最高点，这意味着我们可以在这一数值下获得最好的招聘效果。因此，我们将 285 作为我们的超参数的最佳选择，以表示推荐流程中 BEST 模型。

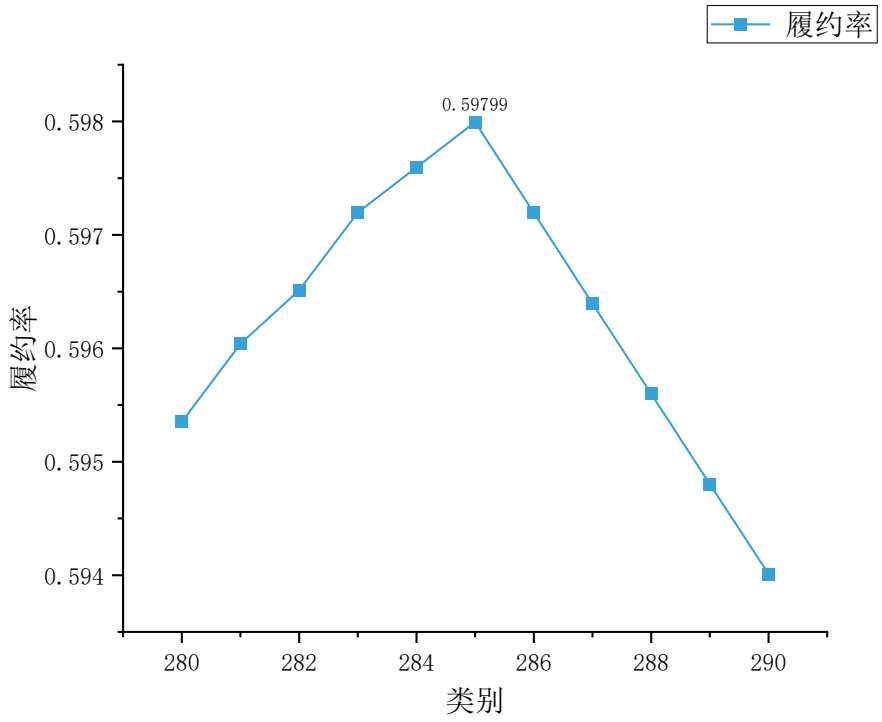

图 49 履约率极限值

7.4.2 贪心选择

在本次实验中，我们对发放 offer 的顺序基于贪心策略使用了多种改进方式，

首先是对其进行用户满意度和岗位匹配度进行了加权聚合，这样排序后发现招聘需要许多轮才达到了最终的终止条件，我们思考后发现是当权重相同时，会按照一个固定的方式进行排序，导致多个公司将 offer 发给了同一个人手中，但这个人对这几所公司的满意度都相同，从而导致整个流程非常的缓慢。

7.4.2.1 增加随机数

接着，我们在此基础上，对于用户满意度和岗位匹配度进行了加权聚合后相同的数据排序时加入了随机数，通过随机一个 1-10 之间的随机数，判断相同数据的前后顺序，取得了较好的结果，但是依然是出于较低的水平线。

理论履约率为:0.597994917 最好结果为：0.37291419,占比：0.623607626。

7.4.2.2 相邻判别

然后，我们在上述的基础上加入了相邻判定的流程，对于相邻两个岗位，如果他们发放给相同的招聘者，而这个招聘者对于两个岗位的满意度和匹配度都相同时，后一岗位将不会对这名招聘者发放 offer，在一定程度上避免了不必要的 offer 分发。取得了较好的结果，但还有更进一步的可能。

理论履约率为:0.597994917 最好结果为：0.37345679,占比：0.624514991。

7.4.2.3 考虑 offer 数量

最后，我们经过思考，决定加入招聘者收到 offer 的数量作为排序的指标，当用户满意度和岗位匹配度的加权聚合相同时，比较招聘者接受到的 offer 数量，对于接收到更多 offer 的招聘者将其置后，如果接受到的 offer 数量相同时，就进行随机数比较，经过这样的改进，我们取得了较好的结果，也是目前最好的结果。

总结，我们通过贪心选择，不断改进招聘流程中的发放 offer 的顺序，取得了较好的实验结果，理论履约率为:0.597994917 最好结果为：0.466615249,占比：0.780299691。

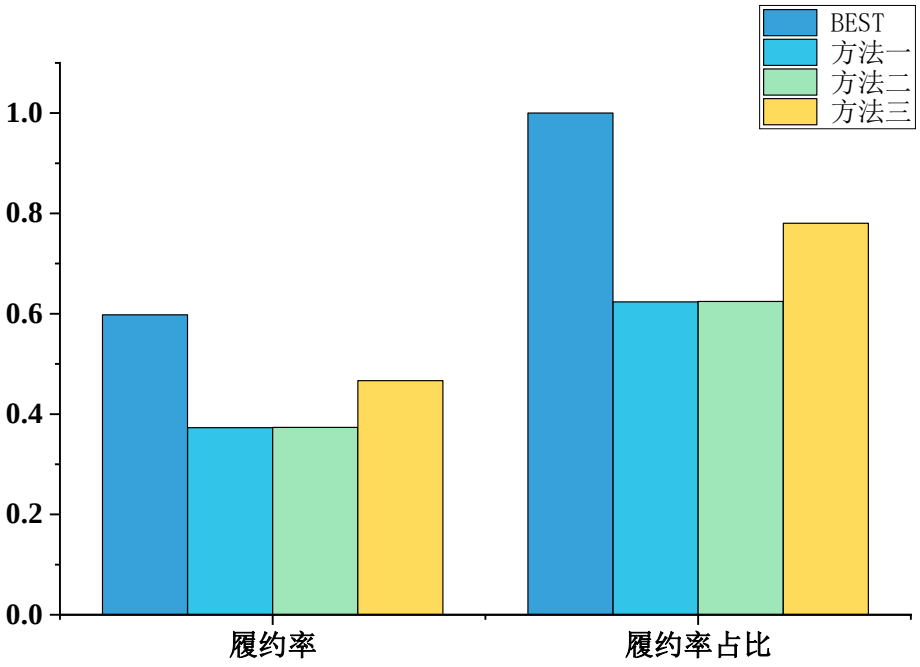

图 50 贪心策略履约率比较

由图 50 可以看出，方法一（增加随机数）与方法二（相邻判别）履约率差别不大，这是因为随机数带来的影响较小，方法三（考虑 offer 数量）相较于前两种方法有明显的提升，表明方法三综合考虑 offer 数量的贪心策略对于双向推荐系统的效果更好。

八、 模型的评价与推广

8.1 模型的优点

使用 LDA 主题生成模型和 K-modes 聚类算法构建招聘信息画像与求职者画像；可以从大量的文本数据中自动地抽取潜在的主题，从而反映招聘信息或求职者自我介绍的主要内容；可以根据主题分布进行文档聚类，从而发现招聘信息或求职者间的相似性或差异性；根据主题词分布进行关键词提取，提高相关信息的可理解性。KModes 算法可以直接处理离散型数据，快速聚类分析得到每个类别的代表性特征值，可作为该类别的整体画像。

对于基于贪心策略的推荐系统，我们执行效率高，每一次的改进最终的目标都是使得招聘流程中的轮数减少，从而提高执行效率以及履约率。并且更加贴近实际的招聘流程，无论是相邻判别还是考虑招聘者的接收的 offer 数量，都是正常招聘流程中 HR 所用考虑的实际情况，因此取得较好的结果。贴合问题，我们所定义的的贪婪算法是对于这道题我们所面临的问题所提出的，因此在不断改进的过程中，与我们所解决的问题越发的匹配。

8.2 模型的缺点

需要指定聚类个数，但并没有一个明确的方法来确定最优的聚类个数；LDA 模型基于词袋，没有考虑词语之间的顺序、语法、语义等高级特征，可能会忽略一些重要的信息，或引入噪声；KModes 算法对离散型数据之间的距离度量有一定的局限性，不能较好地反映数据之间的相似性或差异性；KModes 模型对初始聚类中心敏感，可能陷入局部最优解。

九、 参考文献

[1] 韩世依, 张钰晖, 马云山, 涂存超, 郭志芃, 刘知远, 孙茂松. THUOCL：清华大学开放中文词库. 2016.

[2] 邹晓辉,孙静.LDA 主题模型[J].智能计算机与应用,2014,4(05):105-106.

[3] 徐峰.人力资源绩效管理体系构建:胜任力模型视角[J].企业经济,2012,1:68-71.

[4] 广东省政务服务数据管理局、深圳市人民政府.人岗精准匹配模型[EB/OL].(2021-04-20)[2021-10-12]

[5] KE G L,MENG Q,FINLEY T,et al.LightGBM: A highly effiencient gradient boosting decision tree [C]//Proceedings of the 31st International Conference on Neural Information Processing Systems.2017:3149-3157.

[6] 基于 SMOTE 和贝叶斯优化的人岗匹配算法 Adj-LightGBM 刘付谦，秦华妮，赖惠慧