

项目报告

21371036 郝嘉霖

1. 研究目的

在当前的职场环境中，多个行业都呈现出不同程度的招聘需求。信息技术领域的需求持续增加，尤其是软件开发、数据科学和网络安全领域。招聘需求中对于个人能力和需求有着不同的要求，本文的研究主要目的就是详细分析这些能力需求，从未来应聘者的角度看待招聘市场，从人才需求的角度出发，充分利用网络招聘文本信息，研究其中蕴含的对人才的要求。希望为人才培养单位洞悉市场需求现状，并依此改善学科专业建设提供参考。同时有关人才可以根据本文的研究结果，在高校统一的课程安排之下，结合自身特点和职业规划有意识的安排重点学习内容，早做准备。

此外对于不同行业 and 不同职位，我们也对薪资、地点等关键属性进行了分析，了解多个行业的招聘就业需求、薪资水平和地点可以帮助个人做出更明智的职业规划和就业决策。不同的行业和地点提供了多样化的机会，可以根据个人的兴趣和技能来选择最合适的职业路径。

2. 研究过程

2.1 招聘数据的获取

在我们的研究中，获取招聘数据的方式是通过网络爬虫，通过python来编写爬虫爬取第三方招聘平台上的招聘数据。

2.1.1 第三方招聘平台

概述

我国的第三方招聘平台数量众多，种类多样。目前，我国的第三方招聘平台主要有综合招聘和垂直招聘两种主要模式。最早出现的是综合模式，这类平台覆盖多个行业和地理区域，具有很强的综合竞争力。前程无忧是这类平台的典型代表。而垂直招聘网站则更具针对性，专注于特定地域或领域，因此岗位匹配度较高。此外，一些分类信息网站主要面向蓝领人群，并融入了社交元素。最新的招聘模式将求职培训与岗位推荐相结合，通常规模较小但专业性很强，例如牛客网。

实现

为了获取到更加多元化的招聘数据，我们主要采用了综合招聘网站例如智联招聘、前程无忧和猎聘等，辅以垂直招聘网站特别是一些企业的官方招聘网站例如华为、百度等，这些大型企业的招聘通常比较有针对性。

2.1.2 网络爬虫介绍

概述

爬虫是一种用于获取到网络数据的技术，由于高速发展的互联网使得很多信息都出现在了网络上，所以诞生了爬虫，爬虫可以模拟人类点击网站，访问内容并获取到页面内的全部数据，然后通过进一步的对于页面内的数据进行解析就可以提取到数据，最后对于数据进行处理就可以保存在数据库内。因此整体流程可以被写作如下：

1. 模拟人类点击发送请求
2. 获取整体网页数据
3. 解析数据，获取需要的部分

4. 处理数据，保存进入数据库

而想要爬取的目的网络也已经在上面确定了，就是智联、前程无忧、猎聘、华为、百度等招聘网站，我们只需要分析页面结构，按照指定的方法对其中的数据爬取、解析、保存即可。

实现

本研究中使用了python的request库和selenium包，前者可以模拟浏览器给对应的网站发送请求，后者则可以模拟人类行为点击和移动界面中的元素，由于现在的网站存在较为复杂的反爬手段，像以前一样单纯的使用request发送请求无法获取到有效的数据，想要获取到目标数据有时候会经过“机器人验证”等过程，也就是拖动滑块，这时候就需要selenium来完成目标元素的移动，selenium可以完成反爬验证。

通过selenium和request的有机结合，很快就能通过验证并爬取到对应的数据，最终获取到了约3.7w条数据。



图2-1 招聘网站示例

2.2 招聘数据的处理

在我们的研究中，除去获取数据，还需要对数据进行筛选和处理，通过python来完成数据清洗的过程

2.2.1 数据清洗

爬取到的数据中，由于发布方的问题，可能有一些数据中有一些字段不存在或者模糊不清、格式混乱，这样的数据在分析过程中会导致程序无法解析出现错误，所以首先要被剔除。因此数据首先需要被清洗一遍，对于一些无法被解析的字段需要清洗筛选掉。

我们使用了python的pandas包进行数据处理，pandas包是一个强大的数据处理包，对于处理二维表格等有着非常高的效率，通过pandas包中自带的函数如 `dropna()` 等可以很快的清洗掉不合法的数据

爬取到的数据中，由于许多第三方的招聘网站对于格式要求非常不严谨，所以各个招聘信息的格式五花八门，在清洗掉不合格的数据之后，还需要对全部数据进行格式化处理，让他们有着相同的格式，便于后续的工作和分析。例如：在招聘数据中给出的薪酬字段，一些数据会写：“10k/月”，另一些数据会写：“120元一天”。所有的薪酬字段为了便捷最终会被格式化以K/月为单位的字段。

我们仍使用pandas包，对于不同列上的数据进行不同的处理，根据其中的关键词来判断是以日、月、年作为单位，并识别其中的数字字段，按照换算关系换算成统一的单位。

对于地区，我们构建了一个区域-省-市的关系表，然后根据招聘数据给出的具体地区，通过关系表就能很快地找到区域和省、市，获取到这三个字段。

经过数据清洗之后，我们仍保留了近2.7w条招聘数据。

大类专业	大类岗位	岗位名称	薪酬	地点-区域	地点-省	地点-市	学历要求	经验要求	企业名称	企业规模	岗位详情
软件工程	前后端	前端开发主	7.0K/月	华东	江苏省	南京	本科	3-5年	瑞仪光电	1000-9999	任职要求：1. 本科学历，计算机、软件等相关专业，良好的英文能力 2. 熟练使用c#语言，掌握Winform、MVC、WebService开发技术3. 熟悉SQL语言，ORM框架，掌握SQLServer等主流数据库开发技术4. 熟悉前端语言，掌握HTML、CSS、Javascript技术，熟悉jQuery、echarts等常用前端库福利待遇：工资待遇：另加绩效奖金、年终奖金等，如有加班按法律规定进行结算福利政策：提供五险一金，提供食宿休假制度：享有双休，法定假日，婚、丧假及产假，法定带薪年假等教育培训：提供良好的教育训练及在职进修体系
软件工程	前后端	前端开发工	13.0K/月	华东	上海市	上海	本科	3-5年	国以贤智能	100-499人	岗位职责1、负责前端界面构建，各类交互设计实现等;2、完成产品(网页端)和管理平台的前端开发，编写可复用的用户界面组件;3、持续优化前端体验和页面响应速度，并保证兼容性和执行效率;4、和后端工程师紧密配合，保证产品的用户体验及稳定性;5、负责项目的数据交互等业务，独立完成项目并解决遇到的技术问题; 6、具有较强的性能优化意识并发挥积极作用。任职资格：1、本科及以上学历，3年及以上前端开发经验; 2、精通easyui、javascript、html、css、js等前端开发技术; 3、熟悉ES6标准、主流的JS库和开发框架，例如:Node.js、Vue、Webpack且有实际项目经验; 4、工作态度积极主动，认真负责岗位职责。

图2-2 处理后的数据示例

2.2.2 数据分词

对于数据中“岗位详情”这一个字段，发布方会发布岗位的任职资格、任职资历以及福利等信息，我们的目的是研究具体的任职需求，因此为了删除所有的噪音文字例如“的”、“和”等与需求无关的无意义的字还需要对其进行筛选。此外为了后续的文本向量化，除了筛选无意义的词语还需要把一段文字划分为一个个的单词，这样才能够对这段话进行数据分析，所以这里需要一次分词。

我们使用了jieba包进行分词，由于jieba包虽然自带大型字典，但其中都是常用词，对于招聘信息这样有着大量专业词汇的特殊文本，需要给词典中加入一些专业词汇，否则会出现分词错误的情况，我们通过大语言模型生成了目标行业内主要的专业词汇并构造成词典dict.txt，搭配jieba包中的一般词汇一同作为词典进行分词。

```
# 掌握机器学习算法，熟练使用python、计算机相关专业
'''
'''
添加自定义字典之前：['掌握', '机器', '学习', '算法', '熟练', '使用', 'python', '计算机', '相关', '专业']
'''
'''
添加自定义字典之后：['掌握', '机器学习算法', '熟练', '使用', 'python', '计算机相关专业']
'''
```

图2-3 分词示例

此外对于常见的停用词如“的”、“和”等，我们在csdn上找到了常见的中英文停用词，构造了一个停用词字典，在此基础上我们还添加了一些常见的对于需求没有意义的动词和形容词，例如“至少”，“非常”，“强”，“有着”等无意义的词语，旨在尽可能保留和专业需求相关的名词性词语，依次构造了stopwords.txt，在分词的基础上剔除了上述这一类无效的词语。

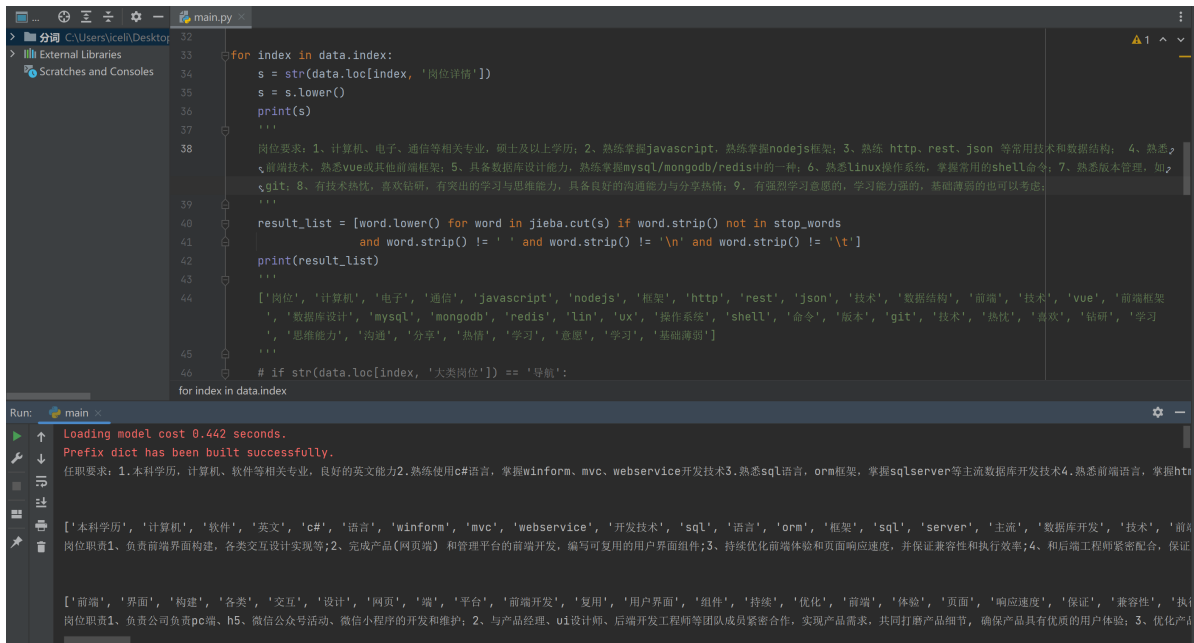


图2-4 分词结果示例

2.3 招聘数据的分析

这是研究过程中最重要的一个部分，就是如何对这些招聘数据进行处理，获取到不同类别的不同数据。我们的目标是对同一个专业下的所有招聘数据进行分类，根据全部招聘数据的“岗位详情”描述字段，对招聘主题进行分类，这样就可以获取到该专业下有哪些主要的招聘方向，例如“软件工程”下往往有“后端”，“前端”等分类。分类完成后再分析不同招聘主题下的薪资、地区等数据，得到最终的结果并展示。

2.3.1 LDA模型

概述

Latent Dirichlet Allocation (LDA) 是一种用于文本分析和主题建模的机器学习模型。它的基本思想是：每篇文档都由多个主题组成，每个主题又由一组单词组成。LDA的目标是从给定的文档集合中推断出主题和每个文档与主题的关系。

LDA的工作过程包括以下步骤：

1. 初始化：为每个文档中的每个单词分配一个随机的主题。
2. 迭代更新：在多次迭代中，LDA根据文档中的单词分布和主题分布来更新主题的分配，直到达到稳定状态。
3. 输出结果：最终，LDA输出每个文档的主题分布以及每个主题的单词分布。

LDA的应用包括文本分类、信息检索、主题发现等领域。它有助于理解文本数据中的潜在结构，从而使文本数据更容易分析和利用。通过LDA，我们可以识别文本中的关键主题，为信息检索和知识管理提供有力支持。

确定困惑度

我们使用了python的gensim包里的IdaModel来实现我们的LDA模型，首先我们根据LDA模型对某一个专业下的所有招聘数据的主题数量进行确定，对于所有招聘数据的“岗位详情”字段，由于招聘数据本身的混杂性和数据量较为庞大，人为确定主题数量是不现实的，所以我们通过LDA的困惑度来确定主题数量，这样的结果可以更加准确。

我们首先对主题的范围进行一个确定，让LDA从1-35个主题上，以5为步长进行迭代计算，从而确定主题从1-35之间的最佳区间，结果如下：

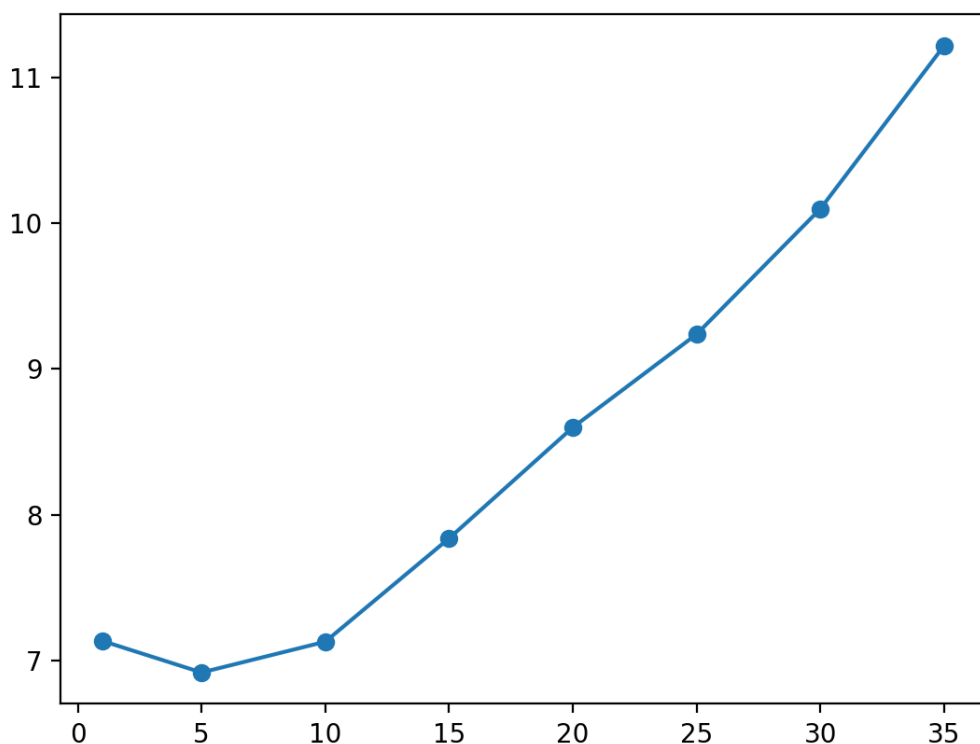


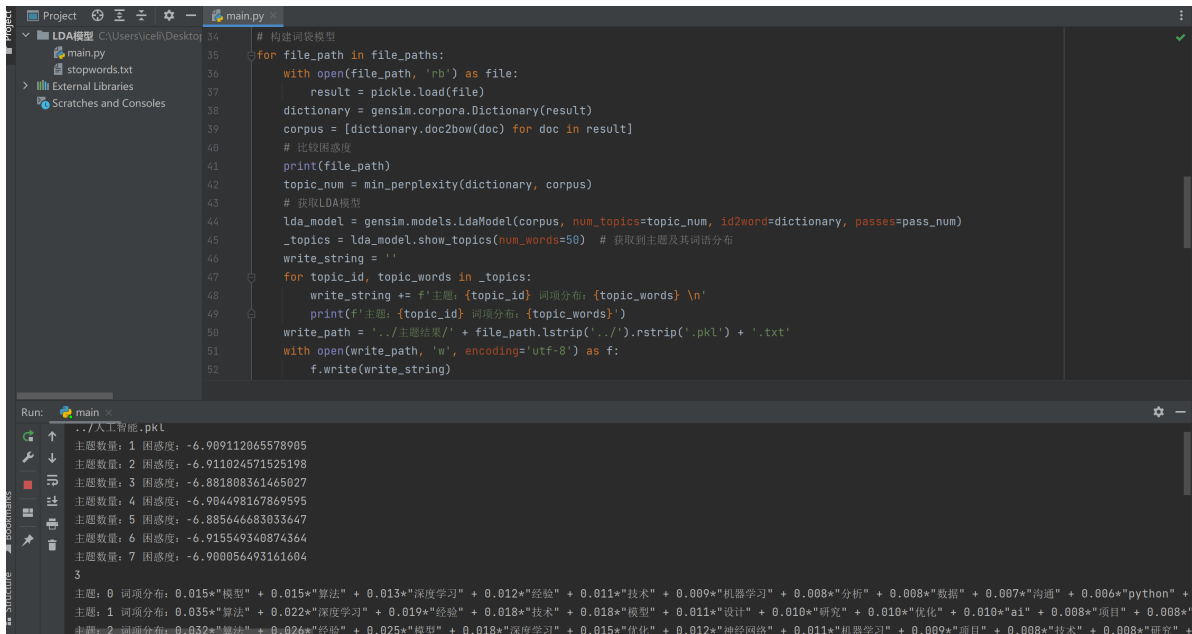
图2-5 困惑度大小示例

可以看到，主题数过多导致模型反而无法集中注意力，困惑度太高，最佳的主题数量在1-10之间，因此我们之后将会遍历1-10，找到最佳的主题数量，并以此来训练LDA模型。

训练LDA模型

在遍历和确定主题数量之后，我们使用LDA模型按照主题数量来获取到具体的主题内容，LDA的迭代次数我们选择了2次，过少的迭代次数会导致欠拟合，而过多的迭代次数会导致过拟合，根据我们自己的数据量，最终选择了这个迭代次数。

接下来，我们分专业进行分析，对于某一个专业下的所有招聘信息，我们把“岗位详情”在分词结果中处理得到的向量通过 `doc2bow` 方法根据词典转化为向量，这样我们就得到了所有招聘信息的向量化结果，第*i*的位置上的元素是*n*就表示第*i*个词在这个招聘文本中出现了*n*次，然后我们把全部向量传递给LDA模型，通过困惑度确定好的主题数量以及设定好的迭代次数进行迭代，最终完成训练。



```
34 # 构建词袋模型
35 for file_path in file_paths:
36     with open(file_path, 'rb') as file:
37         result = pickle.load(file)
38         dictionary = gensim.corpora.Dictionary(result)
39         corpus = [dictionary.doc2bow(doc) for doc in result]
40     # 比较困惑度
41     print(file_path)
42     topic_num = min_perplexity(dictionary, corpus)
43     # 获取LDA模型
44     lda_model = gensim.models.LdaModel(corpus, num_topics=topic_num, id2word=dictionary, passes=pass_num)
45     _topics = lda_model.show_topics(num_words=50) # 获取到主题及其词语分布
46     write_string = ''
47     for topic_id, topic_words in _topics:
48         write_string += f'主题: {topic_id} 词项分布: {topic_words} \n'
49         print(f'主题: {topic_id} 词项分布: {topic_words}')
50     write_path = '../主题结果/' + file_path.lstrip('../').rstrip('.pkl') + '.txt'
51     with open(write_path, 'w', encoding='utf-8') as f:
52         f.write(write_string)
```

Run: main

../人工智能.pkl

主题数量: 1 困惑度: -6.989112865578985
主题数量: 2 困惑度: -6.911824571525198
主题数量: 3 困惑度: -6.881808361465027
主题数量: 4 困惑度: -6.984498167869595
主题数量: 5 困惑度: -6.885646683033647
主题数量: 6 困惑度: -6.915549340874364
主题数量: 7 困惑度: -6.980856493161604

3

主题: 0 词项分布: 0.015*“模型” + 0.015*“算法” + 0.013*“深度学习” + 0.012*“经验” + 0.011*“技术” + 0.009*“机器学习” + 0.008*“分析” + 0.008*“数据” + 0.007*“沟通” + 0.006*“python” +
主题: 1 词项分布: 0.035*“算法” + 0.022*“深度学习” + 0.019*“经验” + 0.018*“技术” + 0.018*“模型” + 0.011*“设计” + 0.010*“研究” + 0.010*“优化” + 0.010*“ai” + 0.008*“项目” + 0.008*
主题: 2 词项分布: 0.032*“算法” + 0.026*“经验” + 0.025*“模型” + 0.018*“深度学习” + 0.015*“优化” + 0.012*“神经网络” + 0.011*“机器学习” + 0.009*“项目” + 0.008*“技术” + 0.008*“研究” +

图2-6 LDA训练结果示例

训练结束后我们获取到了每个专业下有几个主题以及每个主题的词项分布，接下来按照这些主题进行聚类。

2.3.2 TF-IDF词袋模型

文本向量化仅仅用于LDA模型的训练，对于大规模的招聘数据，除了频率向量化，更好的办法是基于TF-IDF词袋模型的word2vec向量化，在频率的基础上，TF-IDF还添加了逆文档频率，使得向量更加特征化，对于聚类等都有着更好的效果帮助。

我们在这里使用了python中gensim包里的TF-IDF的模型实现word2vec，得到了每个招聘数据最后的向量表示，用于后面的PCA和kmeans处理。

```
# 获取TF-IDF模型
tf_idf_model = gensim.models.TfidfModel(corpus)
tf_idfs = tf_idf_model[corpus]
dict_num = len(topics[0]) # 所有主题的语料库中词语是一样的，选择第一个主题计算即可
vectors = [] # vectors数组是一个普通数组，里面的每一项都是一个文章的tf-idf词项分布
for tf_idf in tf_idfs:
    result_list = [0] * dict_num
    for (index, num) in tf_idf:
        result_list[index] = num
    vectors.append(result_list)
```

图2-7 TF-IDF模型示例

2.3.3 PCA降维

概述

主成分分析（Principal Component Analysis, PCA）是一种常用的数据降维技术，旨在通过线性变换将高维数据集转换为低维数据集，同时最大程度地保留原始数据的信息。PCA的目标是找到一个新的坐标系，其中数据的方差最大，从而减少数据的冗余性。

PCA的基本步骤如下：

1. 数据标准化：首先，将原始数据进行标准化，以确保所有特征具有相同的尺度。这可以通过减去每个特征的平均值并除以标准差来实现。

2. 计算协方差矩阵：接下来，计算标准化后的数据集的协方差矩阵。协方差矩阵描述了不同特征之间的相关性和方差。
3. 计算特征值和特征向量：对协方差矩阵进行特征值分解，得到特征值和对应的特征向量。特征值表示了新坐标系中每个主成分（主特征）的方差大小，而特征向量则是每个主成分的方向。
4. 选择主成分：根据特征值的大小，选择要保留的主成分数量。通常，可以根据特征值的比例来确定要保留的主成分数量，以保留足够的数据信息。
5. 投影数据：将原始数据投影到选定的主成分上，从而将高维数据转换为低维数据。

通过这些步骤，PCA可以用来降维，减少数据的维度，同时保留数据的主要信息，以便更容易进行数据分析、可视化和建模。它在数据压缩、特征选择、图像处理等领域都有广泛的应用。

实现

由于招聘文本通常都比较短，而招聘文本的总数量比较大，所以招聘文本向量化之后是一个稀疏向量，其中包含了很多的0，所以一般的PCA效果并不是太好，我们在这里使用了python中sklearn包里的Sparse PCA降维，它是PCA的一个变种，更适合用于处理稀疏向量的降维，其本质上的计算原理和PCA非常相似，只是额外增加了一些和稀疏向量处理相关的方法。

除此之外，由于招聘文本中总是包含了一些共同的文字特征，例如对于岗位的通用要求“负责”，“经验”，“理解”等词汇，大部分的招聘文本都包含了这些词语，因此如果直接进行聚类将会不可避免地导致聚类结果混乱，因为聚类会按照这些共同的通用词汇进行分类，而不是按照每条招聘信息里特有的需求词汇进行分类。

使用PCA降维的原因是原本的招聘文本向量维度比较大，取决于其词典中词语的数量，通常在6000-10000维，并且包含了大量的0，所以在聚类之前必须要用PCA进行降维之后提取出关键的特征之后在进行聚类，这样的方法可以让聚类结果更准确并且效率更高，如果不降维聚类的运行速度将会非常慢甚至不可行。

另外降维的最终维度也是一个超参数，需要手动设定，使用SPCA中的 `explain_variance_ratio` 能够直接查看到最终的降维结果效果，这个结果给出了降维后的各个维度上的评分，我们只需要选择前n个评分较高的维度即可，对于评分很低的维度可以直接舍弃，因为这种维度对于特征化降维的作用微乎其微。

```
[0.02245145 0.01782119 0.01199621 0.01021026 0.00931065 0.00895806
0.00863744 0.00835335 0.00797295 0.00741988 0.00709579 0.00689983
0.00629303 0.00622933 0.00603443 0.00590527 0.00585204 0.00557854
0.00554798 0.00550415 0.00538363 0.00532848 0.00521559 0.00516259
0.00503562 0.00490761 0.00484803 0.00474119 0.00472062 0.0046949
0.0045311 0.00440439 0.00438909 0.00429429 0.00424068 0.00414804
0.00404866 0.00400788 0.00398648 0.00391229 0.00388553 0.00386005
0.00384555 0.00377379 0.00372628 0.00362851 0.00357396 0.00351802
0.00350315 0.00349149 0.00344885 0.00341039 0.00335284 0.00332152
0.0032823 0.0032683 0.00325664 0.0031988 0.00316569 0.0031006
0.00308182 0.00303776 0.00300036 0.0029545 ]
```

图2-8 前64维评分示例

最终我们选择了64维作为降维目标维度，使用 `fit` 和 `transform` 方法完成PCA降维。

2.3.4 kmeans聚类

概述

K均值聚类（K-means clustering）是一种常用的无监督学习算法，用于将数据集中的样本划分成K个不同的簇或群组。其基本原理是通过迭代的方式，将数据点分配到K个簇中，使得每个数据点与其所属簇的中心点之间的距离最小化。

K均值聚类的工作流程如下：

- 1. 随机选择K个初始簇中心点。
- 2. 将每个数据点分配到离其最近的簇中心点。
- 3. 计算每个簇的新中心点，即簇内所有数据点的平均值。
- 4. 重复第2和第3步，直到簇中心点不再发生显著变化，或达到预定的迭代次数。

K均值聚类的优点包括简单、高效、易于理解和实现。然而，它对初始簇中心点的选择非常敏感，可能会陷入局部最优解，因此通常需要多次运行算法，以选择最优的聚类结果。此外，K值的选择也需要一定的领域知识或通过一些启发式方法来确定。总之，K均值聚类是一种强大的聚类算法，适用于各种数据分析和模式识别任务。

实现

我们在之前获取了主题的数量以及每个主题的词项分布，并且还通过PCA获取到了每个文章的向量，由于LDA模型虽然能够给出主题，但是不能够给出某一篇文章属于哪个主题，因此我们需要聚类来根据主题的词项分布向量，把招聘信息的向量通过kmeans聚类向其靠拢，这样能够更好的把招聘向量根据主题完成聚类。

我们使用了python中sklearn里的kmeans包完成了kmeans聚类分组，通过降维后的向量完成聚类，最终获取到了每一个招聘信息所在的主题分类，并添加到csv文件里，至此我们就完成了对于招聘数据的分类分析。

4	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	id	大类专业	大类岗位	岗位名称	薪酬	地点-区域	地点-省	地点-市	学历要求	经验要求	企业名称	企业规模	岗位详情	分组
2		1 软件工程	前后端	前端开发主	7.0K/月	华东	江苏省	南京	本科	3-5年	瑞仪光电	<1000-9999	任职要求：1. 本科学历，计算机、软件等相关专业，良好的英文能力2. 熟练使用c#语言，掌握Winform、MVC、WebService开发技术3. 熟悉SQL语言，ORM框架，掌握SQLServer等主流数据库开发技术4. 熟悉前端语言，掌握HTML、CSS、Javascript技术，熟悉jQuery、echarts等常用前端库福利待遇：工资待遇：另加绩效奖金、年终奖金等，如有加班按法律规定进行结算福利政策：提供五险一金，提供食宿休假制度：享有双休，法定假日，婚、丧假及产假，法定带薪年假等教育培训：提供良好的教育训练及在职进修体系	1
3		2 软件工程	前后端	前端开发工	13.0K/月	华东	上海市	上海	本科	3-5年	国以贤智能	100-499人	岗位职责1、负责前端界面构建，各类交互设计实现等；2、完成产品（网页端）和管理平台的前端开发，编写可复用的用户界面组件；3、持续优化前端体验和页面响应速度，并保证兼容性和执行效率；4、和后端工程师紧密配合，保证产品的用户体验及稳定性；5、负责项目的数据交互等业务，独立完成项目并解决遇到的技术问题；6、具有较强的性能优化意识并发挥积极作用。任职资格：1、本科及以上学历，3年及以上前端开发经验；2、精通easyui、javascript、html、css、js等前端开发技术；3、熟悉ES6标准、主流的JS库和开发框架，例如:Node.js、Vue、Webpack且有实际项目经验；4、工作态度积极主动，认真负责岗位职责。	2

图2-9 聚类分组结果示例

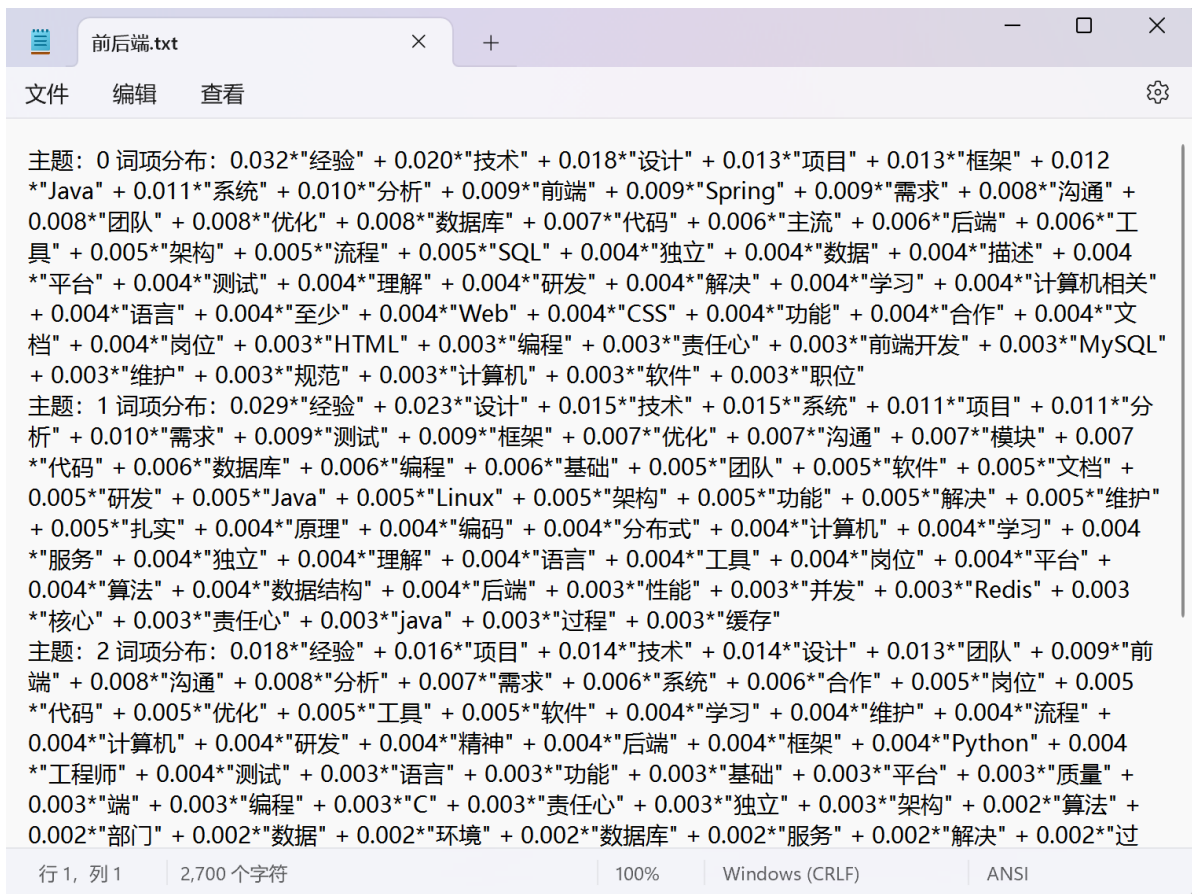


图2-10 主题词项分布示例

3. 研究结果

每一个专业都会通过LDA和kmeans被聚类称为多个类，我们将会根据每个类的词项分布和对应的所有的招聘信息的岗位名称、岗位详情来给每个类取一个合理的名字。

在这里我们的数据展示图中给出了一个“经验不限”的字段，通常招聘者也会在招聘信息里看到“经验不限”，但其通常不是真的经验不限，只是为了增加应聘者数量，背后有着自己的筛选规则。通过我们的研究和数据展示，可以直观地给招聘者展示“经验不限”一般对应了什么段位的薪资，通过和其他经验的薪资的对比，招聘者可以大致了解到所谓的“经验不限”背后的潜规则。

3.1 软件工程专业

3.1.1 前后端

前后端通过LDA和kmeans被聚成了7类

1. python开发

词项分布: 0.017 经验 / 0.015 数据 / 0.015 python / 0.012 技术 / 0.011 算法 / 0.010 平台 / 0.008 ai / 0.008 模型 / 0.008 云 / 0.007 研发 / 0.007 工具 / 0.007 工程师 / 0.007 团队 / 0.007 运维 / 0.006 系统 / 0.006 机器学习 / 0.005 服务 / 0.005 提升 / 0.005 职位 / 0.005 设计 / 0.005 沟通 / 0.004 分析 / 0.004 研究 / 0.004 优化 / 0.004 项目 / 0.004 建设 / 0.004 语言 / 0.004 人工智能 / 0.004 描述 / 0.004 爬虫 / 0.004 ic / 0.003 方向 / 0.003 计算机 / 0.003 深度学习 / 0.003 golang / 0.003 框架 / 0.003 学习 / 0.003 自动化 / 0.003 需求 / 0.003 效率 / 0.003 容器 / 0.003 企业 / 0.003 解决方案 / 0.003 数据分析 / 0.003 生产 / 0.003 k8s / 0.003 网络 / 0.003 岗位 / 0.002 大数据 / 0.002 广告

数据展示

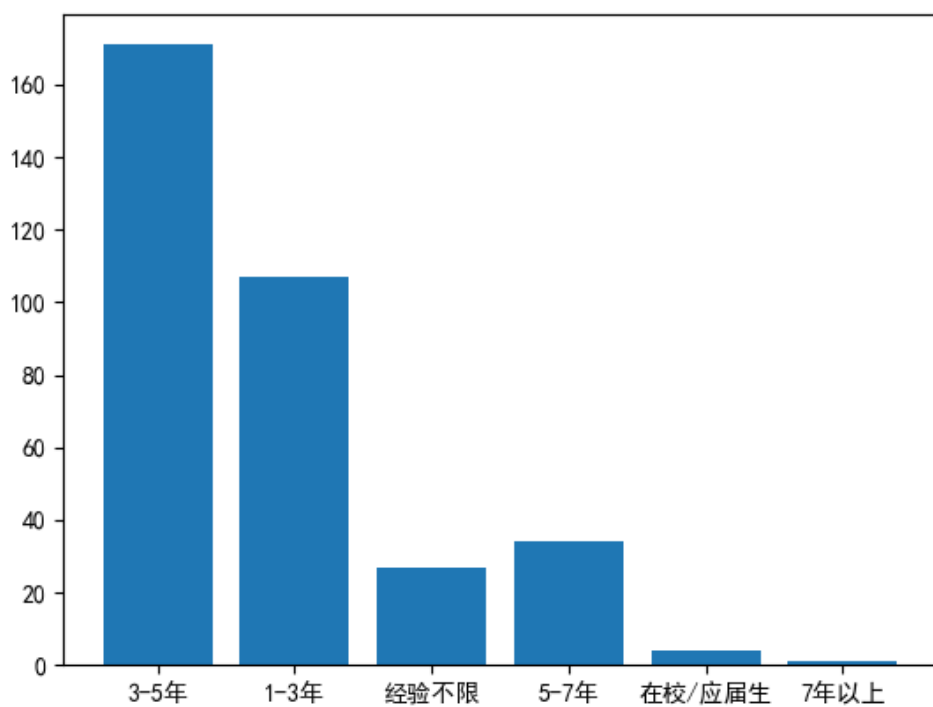


图3-1-1-1 python开发下的经验要求

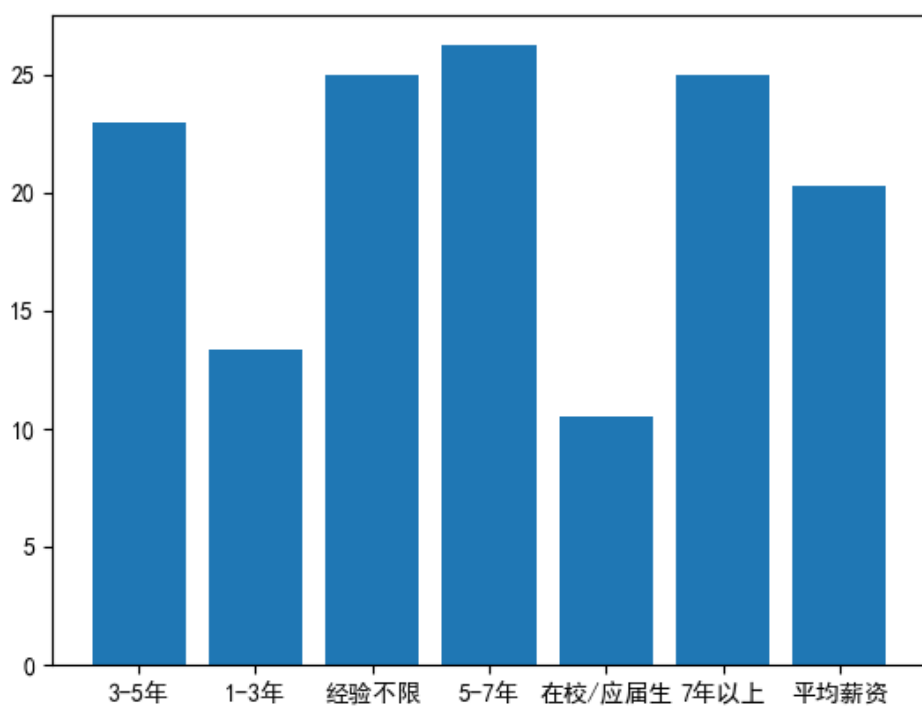


图3-1-1-2 python开发下的经验薪资

这里我们可以看到，对于3-5年的开发工程师经验需求最多，反而是对于7年以上开发经验的工程师需求很少。

薪资水平从应届生到1-3年开发经验到5-7年开发经验逐步递增：从1-3年到3-5年之间有飞跃提升。1-3年和应届生因为经验不足所以薪资水平较低。3-5年经验平均薪资可以达到23w，这时有着较强的耐力和思维敏捷度，也有一定的经验，相当于开发师的“壮年”。5-7年可以达到27w左右，这时有着更多的经验但是由于年龄问题可能缺少了思维敏捷和耐力，相当于“中年”。反而7年以上的薪资更

低了，其一可能是因为需求太少样本不足，恰巧样本中给出的薪资较低，其二更可能是因为7年以上对应的年龄更大了，思维敏捷度降低，面临着程序员的“中年危机”，被裁员和降薪等问题。

从此可以得出的结论就是：python开发岗位对于3-5年经验需求最多，并且薪资水平也非常高；对于5-7年经验的需求不多，但是薪资水平达到了顶峰；对于7年以上的经验需求更少了，薪资反而略低。

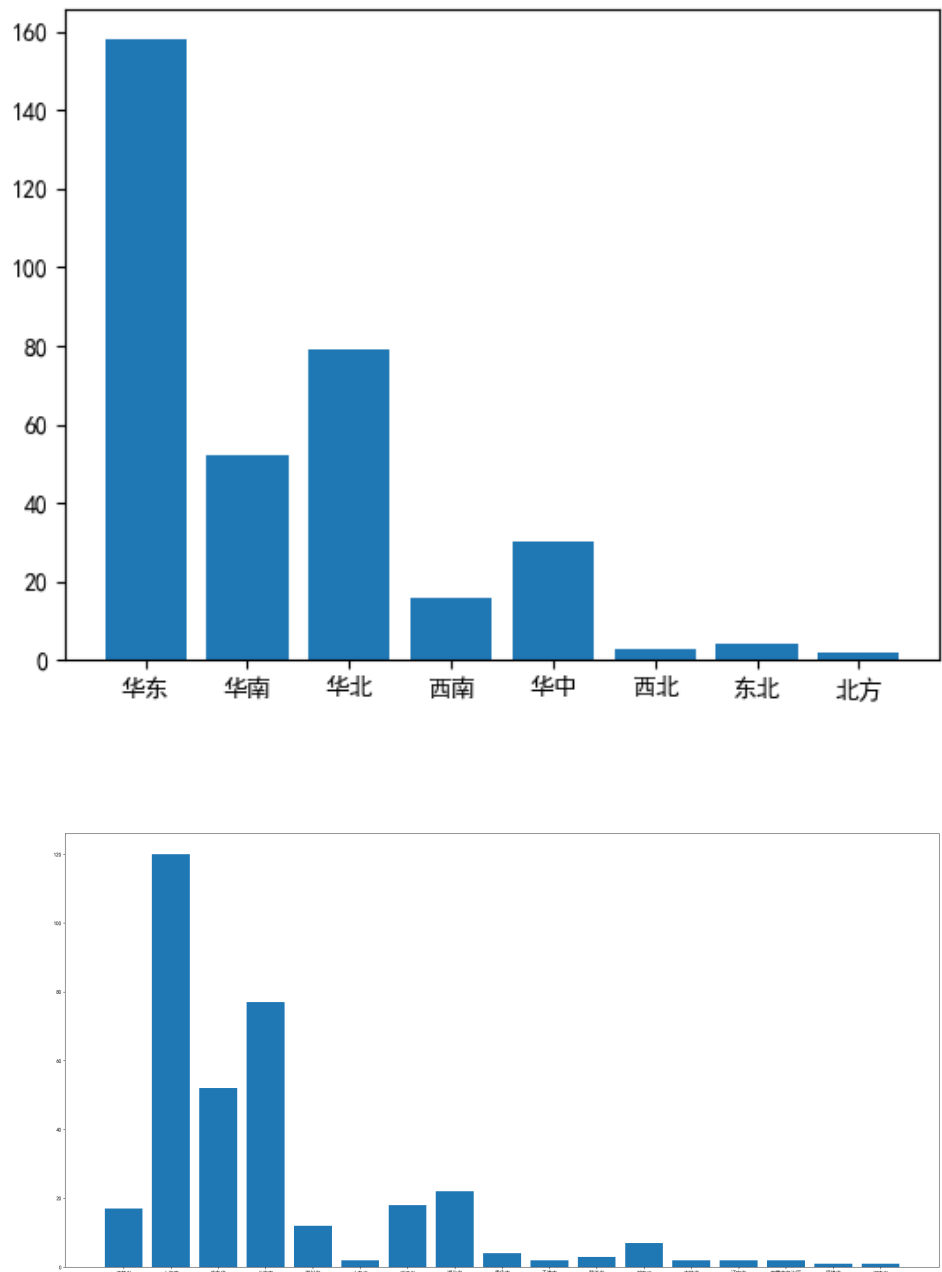


图3-1-1-3 python开发岗位的地区分布

这里我们可以看到，python开发岗位主要分布在华东、华北、华南地区，主要省份即北上广并且上海占据了主要部分，python开发工程师主要在超一线城市就业，在一些比较发达的省份例如江苏、浙江、湖北也有分布。

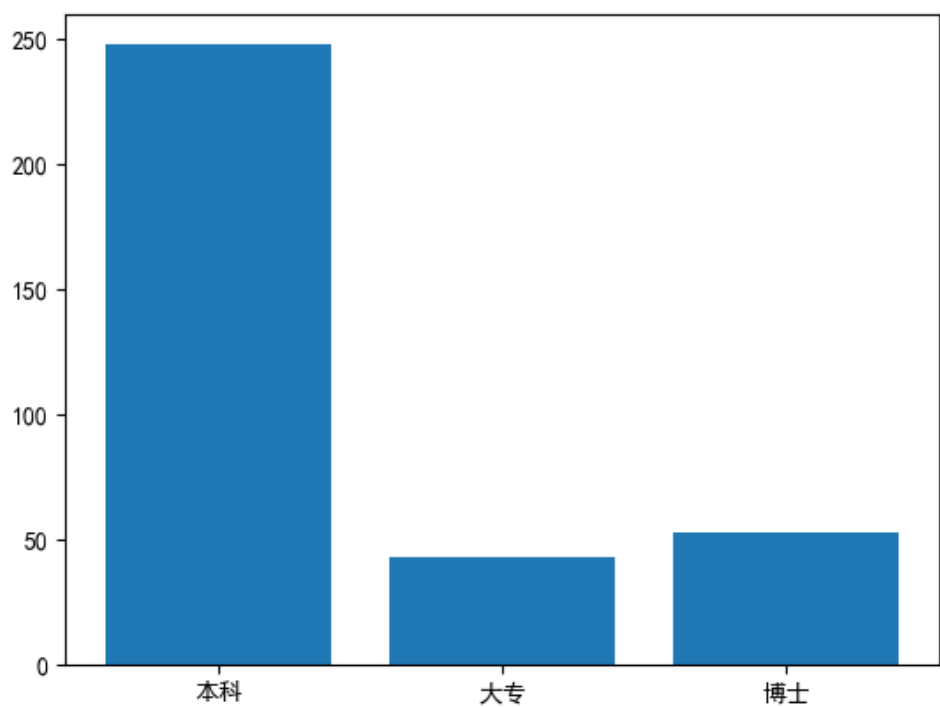


图3-1-1-4 python开发岗位的学历要求

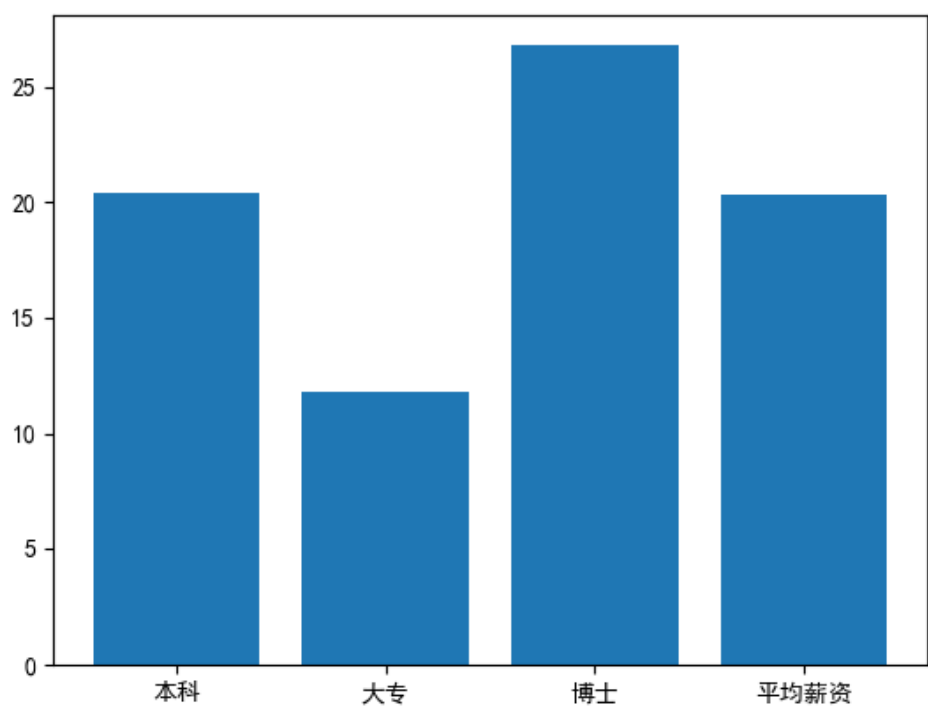


图3-1-1-5 python开发岗位的学历薪资

这里我们可以看到，在所有python开发岗位中，没有对于硕士的要求，只有本科、大专和博士的要求。并且要求应聘者是“本科”的数量最多，博士和大专相比之下都很少。

薪资水平从大专到本科到博士逐渐递增，本科平均薪资达到20w左右，博士可以达到28w左右，从大专到博士逐级递增，符合常识。

从这里可以得出的结论是：python开发岗大部分对于应聘者的需求是本科学历，没有硕士要求，有少量博士，博士的薪资较高。

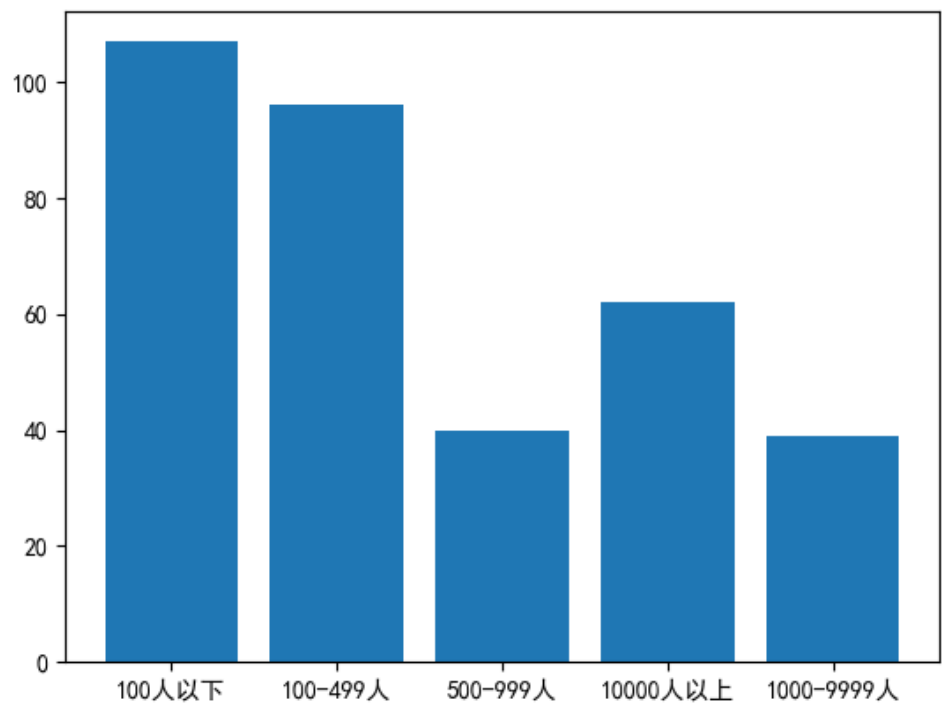


图3-1-1-6 python开发岗位的企业规模

这里可以看到，招聘python开发的大部分公司是小规模的500人以下的公司，中型公司对于python开发的招聘数量不大，大型公司反而招聘的更多了一些。因为python的框架对应的都是比较小型的软件，大型软件通常使用Java编写，所以小公司对于python的需求较大。

2. Java开发

词项分布：0.034 java / 0.032 经验 / 0.020 项目 / 0.019 设计 / 0.019 技术 / 0.017 spring / 0.015 系统 / 0.014 框架 / 0.013 需求 / 0.013 数据库 / 0.012 工程师 / 0.010 分析 / 0.009 测试 / 0.009 sql / 0.009 沟通 / 0.009 软件 / 0.009 代码 / 0.009 文档 / 0.009 mysql / 0.007 团队 / 0.007 独立 / 0.006 软件开发 / 0.006 模块 / 0.006 mybatis / 0.006 cloud / 0.006 主流 / 0.006 acl / 0.005 架构 / 0.005 编程 / 0.005 维护 / 0.005 优化 / 0.005 语言 / 0.005 解决 / 0.004 springboot / 0.004 功能 / 0.004 计算机 / 0.004 mvc / 0.004 redis / 0.004 过程 / 0.004 学习 / 0.004 合作 / 0.004 责任心 / 0.004 编码 / 0.004 工具 / 0.004 计算机相关 / 0.004 开源 / 0.003 linux / 0.003 规范 / 0.003 研发

数据展示：

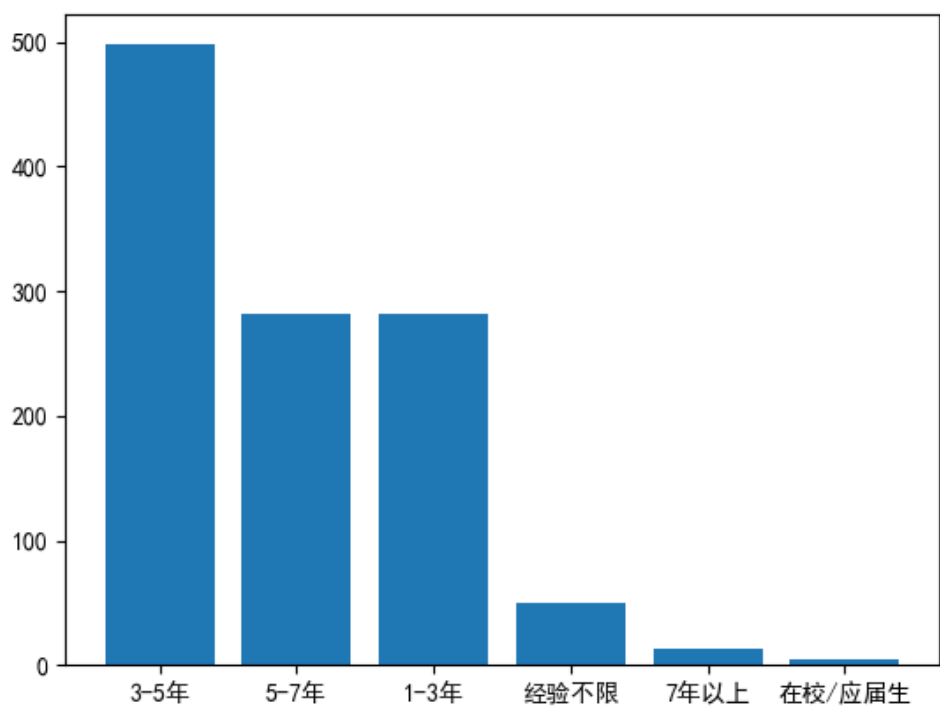


图3-1-1-7 java开发岗位的经验要求

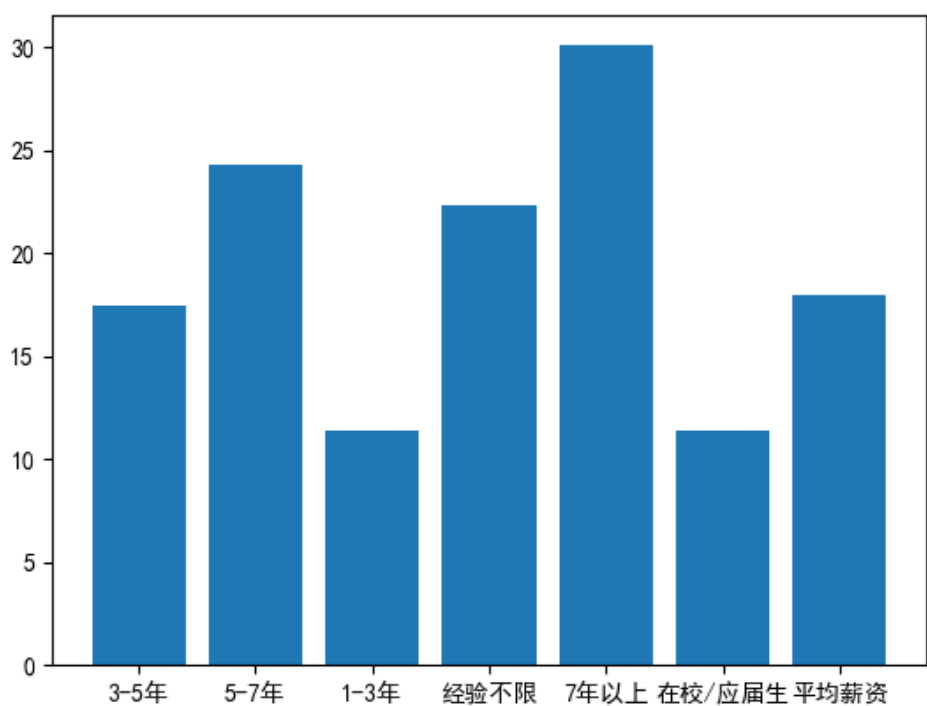


图3-1-1-8 java开发岗位的经验薪资

这里可以看到，java开发岗位对于经验的要求也是大多数是3-5年，以1-3年和5-7年为辅，对7年以上的需求较少。

薪资水平从应届生到7年以上逐级递增，最高可以达到30w的平均薪资。

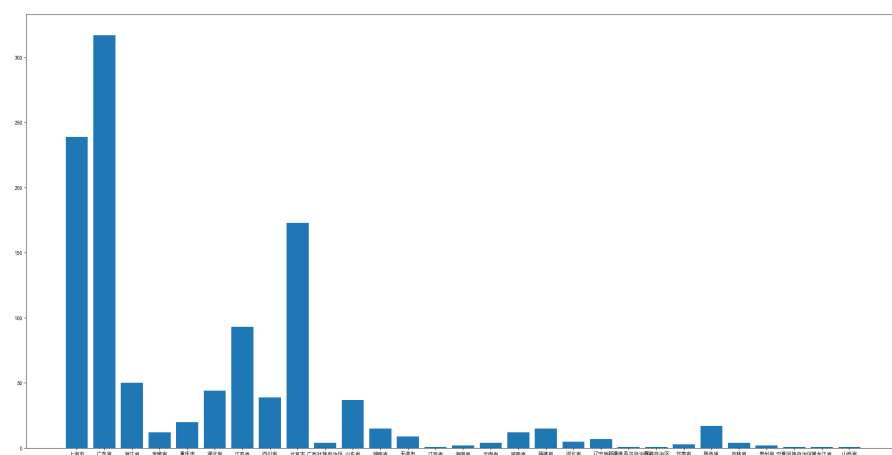
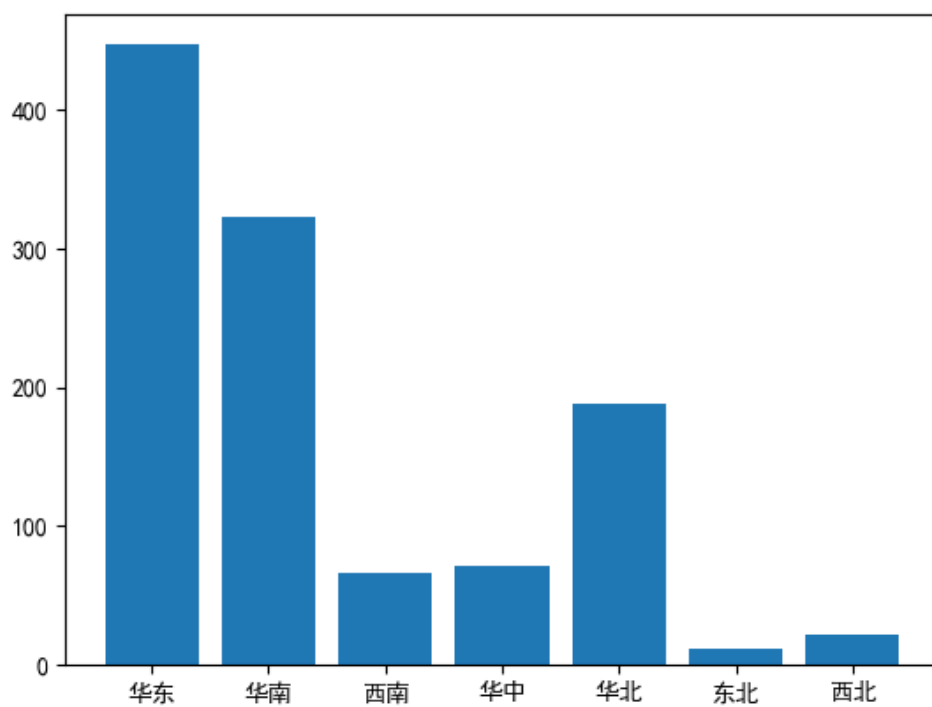


图3-1-1-9 java开发岗位的地区分布

Java的开发岗位也是主要分布在了华东、华南和华北，其中仍是北上广占据了主流，并且广东数量最多。

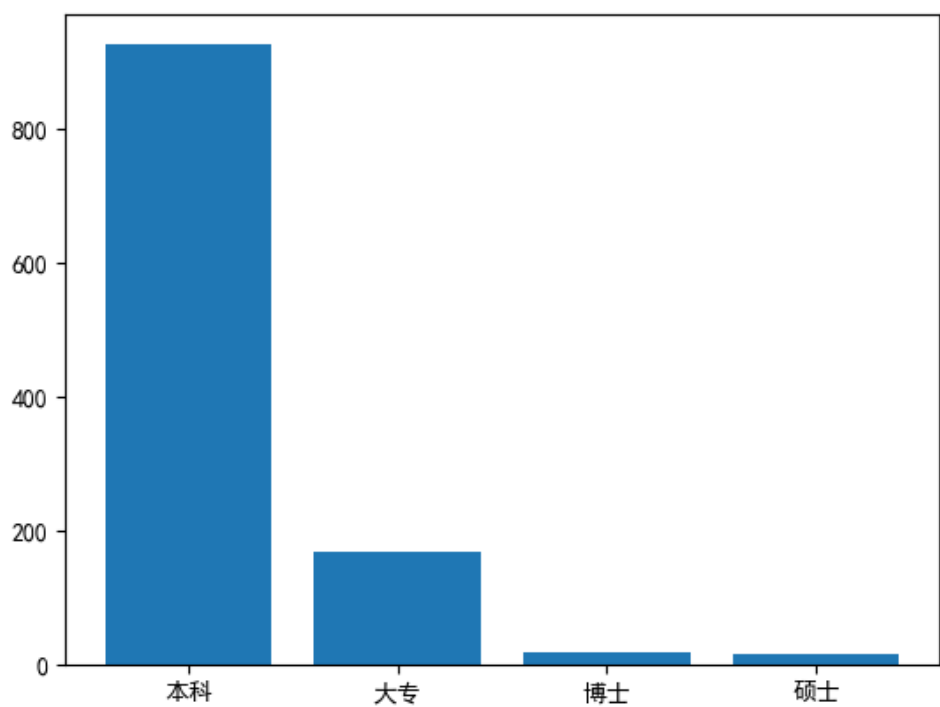


图3-1-1-10 java开发岗位的学历要求

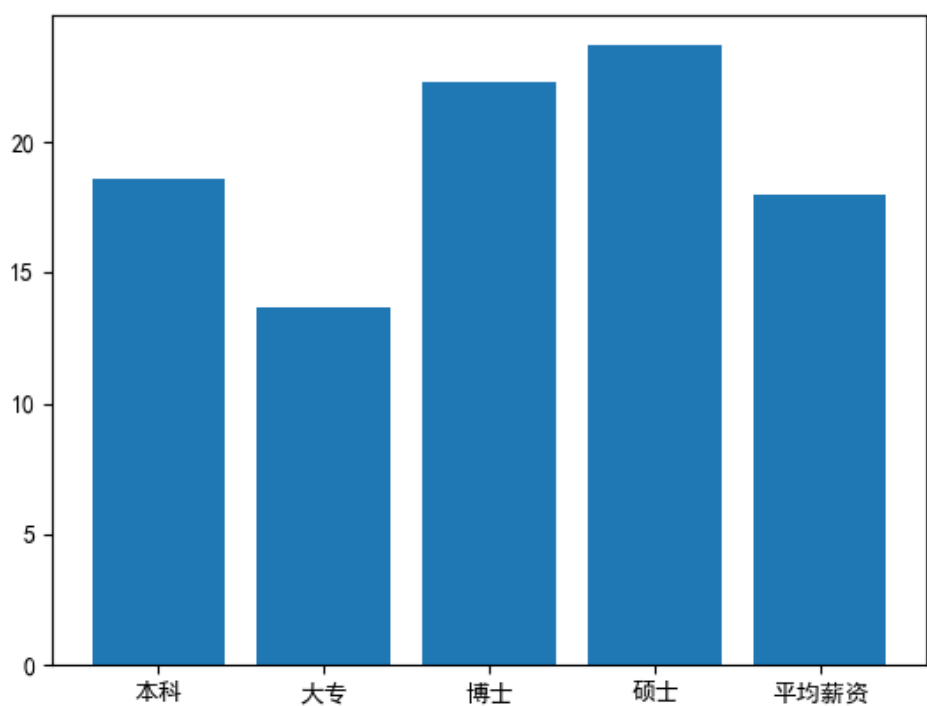


图3-1-1-11 java开发岗位的学历薪资

java的开发对于博士和硕士的要求都非常少，也很符合Java开发岗的特点，主要收纳本科生，硕士和博士也很少会进入java开发岗就业，所以其薪资也没有比本科生高很多，反常的是博士薪资反而还略低，可能是因为很少有博士进入Java开发岗位就业，所以样本数量很少，恰巧样本中对应的岗位薪酬较低。

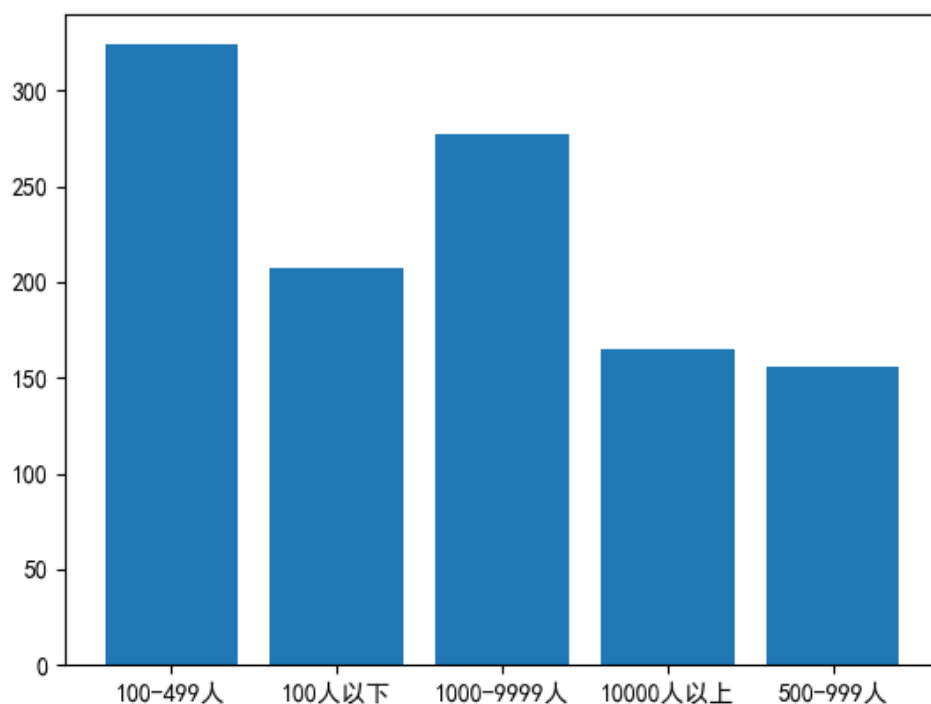


图3-1-1-12 java开发岗位的企业规模

Java开发的企业小中大都包含了并且对应的数量分布都较高，因为大部分企业都需要Java来进行后端开发。

3. 前端开发：

词项分布： 0.032 前端 / 0.021 经验 / 0.019 web / 0.019 前端开发 / 0.017 技术 / 0.015 css / 0.014 es / 0.014 html / 0.011 vue / 0.010 javascript / 0.009 工程师 / 0.009 框架 / 0.009 react / 0.008 js / 0.008 页面 / 0.007 ui / 0.007 项目 / 0.006 端 / 0.006 团队 / 0.005 ar / 0.005 设计 / 0.005 浏览器 / 0.005 ni / 0.005 交互 / 0.004 优化 / 0.004 代码 / 0.004 用户体验 / 0.004 jquery / 0.004 工具 / 0.004 前端框架 / 0.004 学习 / 0.004 angular / 0.004 沟通 / 0.003 组件 / 0.003 数据 / 0.003 ajax / 0.003 主流 / 0.003 需求 / 0.003 岗位 / 0.003 理解 / 0.003 合作 / 0.003 程序 / 0.003 性能优化 / 0.003 webpack / 0.003 微信 / 0.003 app / 0.003 系统 / 0.003 h5 / 0.003 ip / 0.003 ic

数据展示：

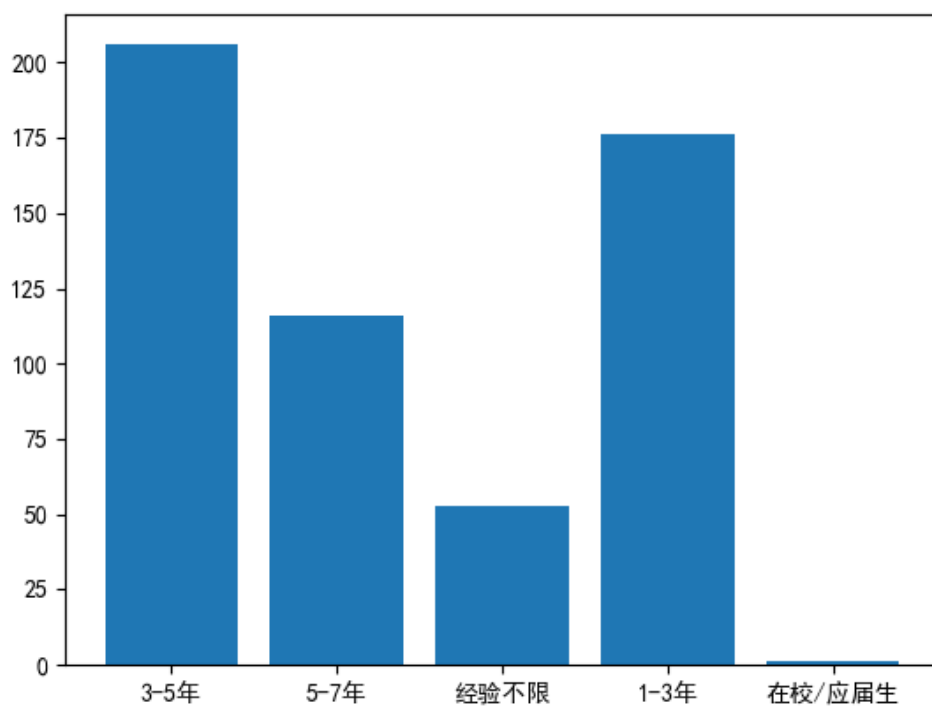


图3-1-1-13 前端开发岗位的经验要求

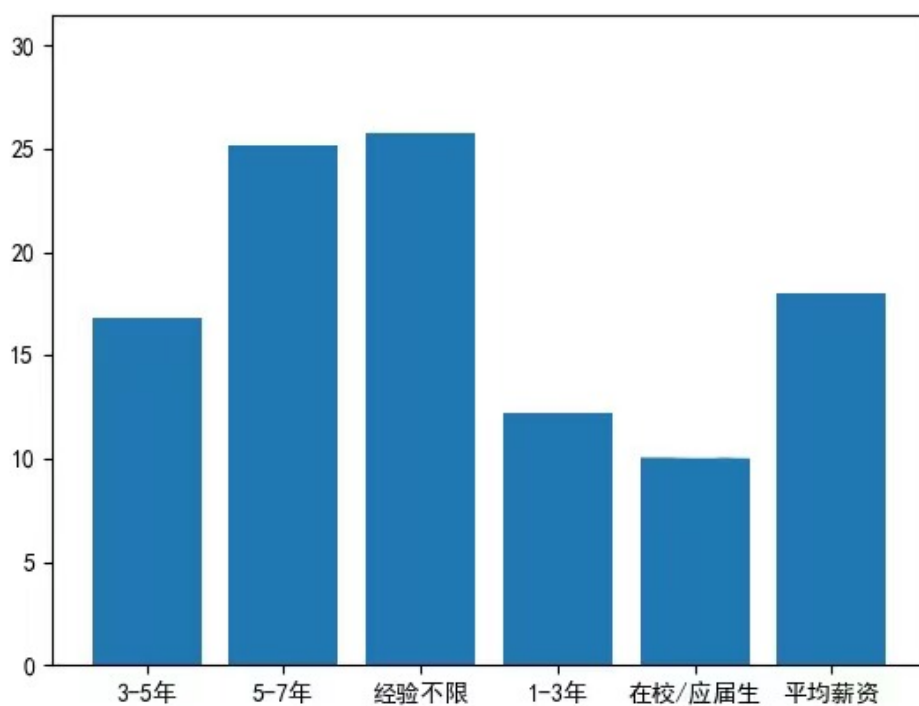


图3-1-1-14 前端开发岗位的经验薪酬

这里可以明显地看到，前端对于经验的要求非常宽松，大多是1-3年到3-5年工作经历的应聘者，没有7年以上的要求。

薪资也是随着资历进行递增但是往往都不是非常高，就算是5-7年的经验，其薪资也仅仅在25w左右。

这里就可以看出来前端开发的岗位对于应聘者的要求比较宽松，进而反映出来前端相比于其他技术岗更好上手。

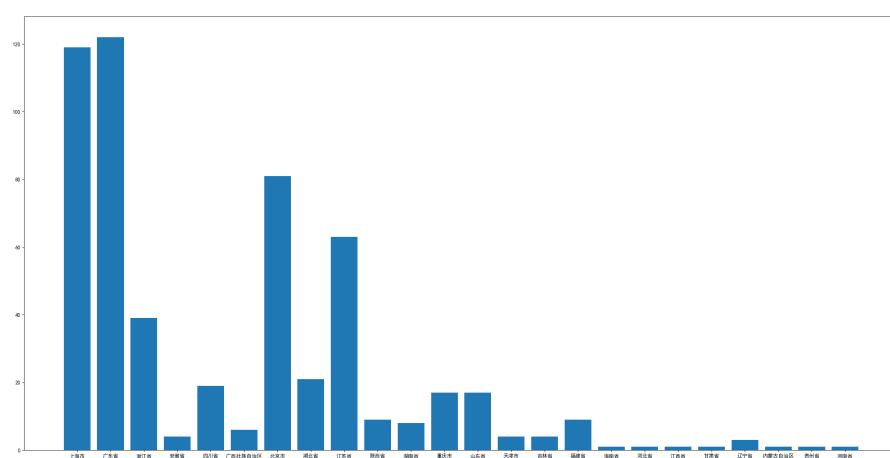
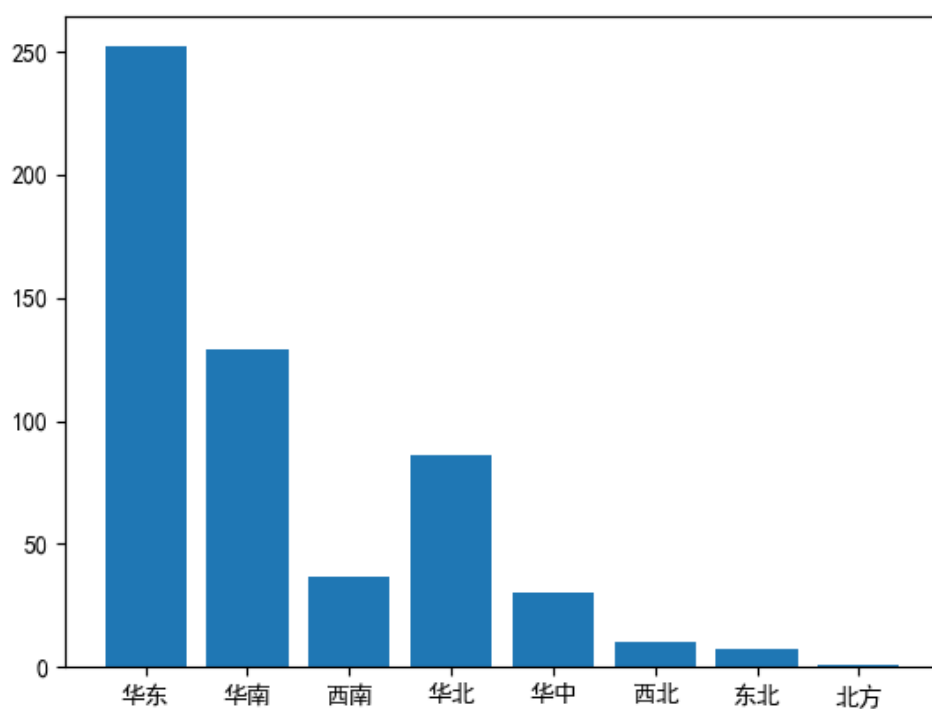


图3-1-1-15 前端开发岗位的地区分布

前端开发仍然主要分布在华东、华南和华北，仍以北上广为领头羊，在超一线城市就业，此外江苏的前端开发岗位也存在一些。

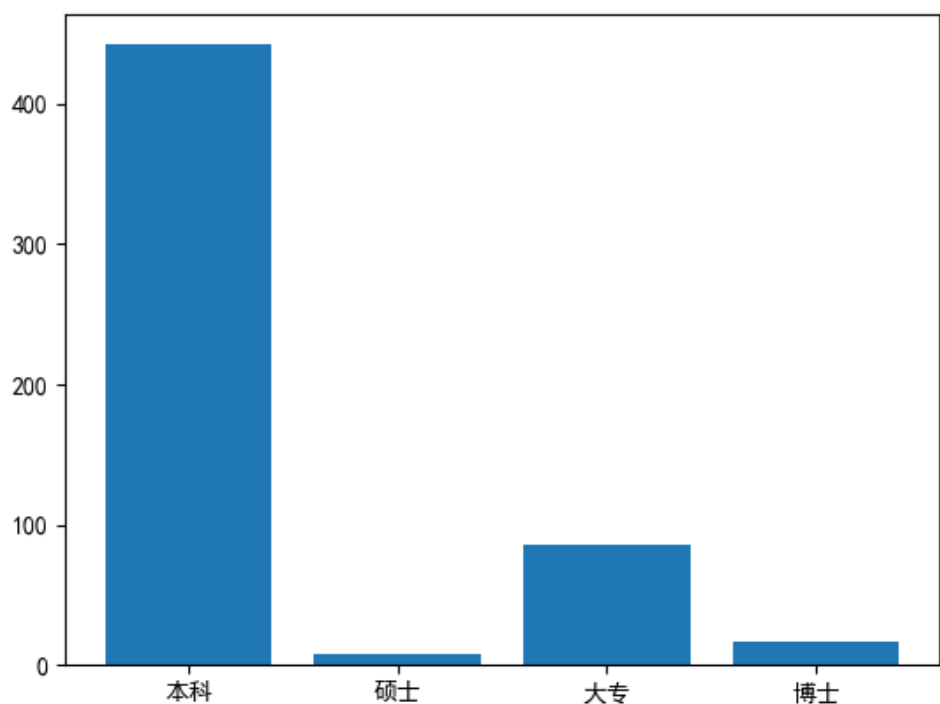


图3-1-1-16 前端开发岗位的学历要求

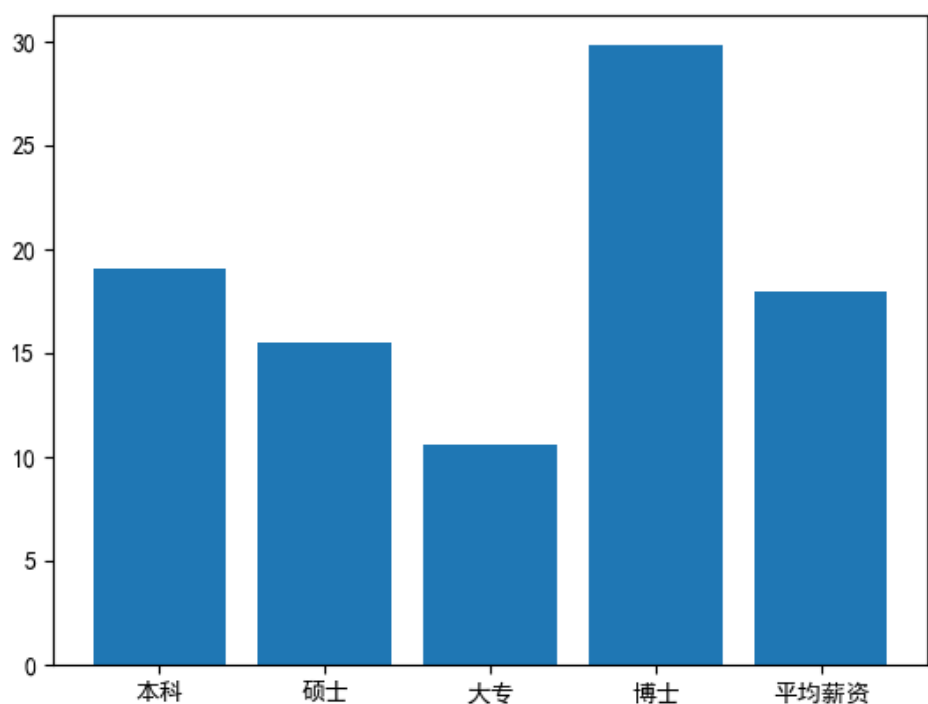


图3-1-1-17 前端开发岗位的学历薪资

这里可以看出来，前端开发岗位几乎不需要硕士和博士，以本科生为主，因为前端易于上手等技术较为简单，所以也很少有硕士和博士生参与到前端开发岗位中。

前端对于博士的要求少但是薪资非常高，通过查看招聘信息发现这是因为少有的大型企业和高新技术仍需要全面性高技术人才，主要的集团是“美团”和“蚂蚁集团”，需要对大平台的前端页面进行开发，由于大部分前端由于易于上手都是本科应聘者，所以前端中的高技术力人才比较少，对于大平台的多技术栈要求就需要博士这种高技术力人才，所以前端的学历中出现了断档，也就是低端技术

人才和高级技术人才的两个极端，很少有硕士学历的中端技术人才。因此给到硕士的学历薪资并不高，因为很少有对应的岗位。

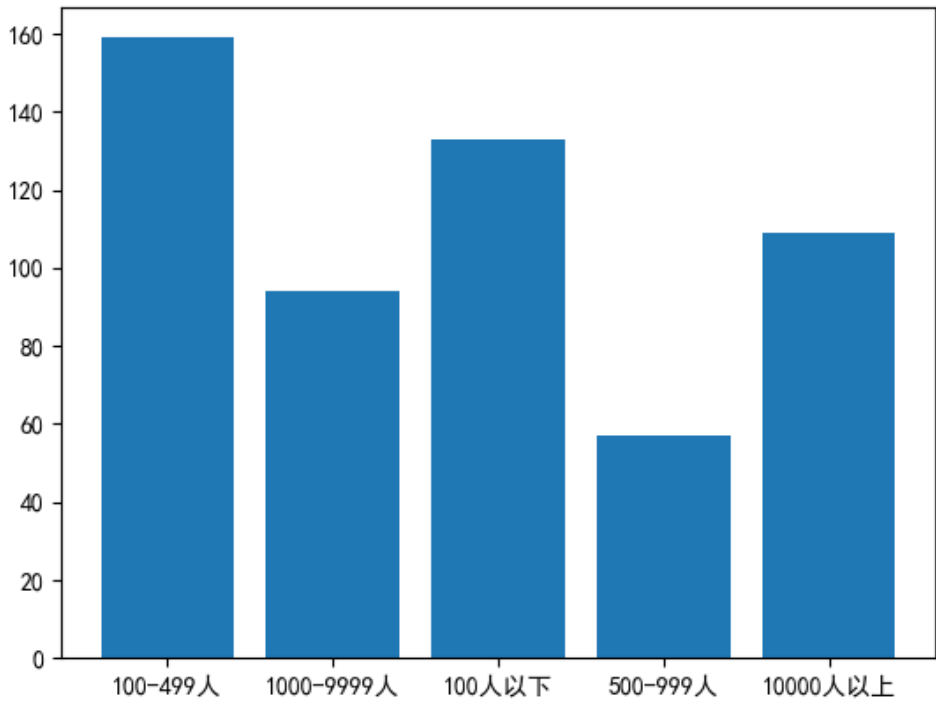


图3-1-1-18 前端开发岗位的企业规模

前端开发岗位和Java开发岗位非常相似，从小中大三个规模的企业都按比例需要人才，因为大部分企业都需要前端。

4. 应用技术栈开发：

应用技术栈不仅仅再限于Java、python或c、golang等某一门语言，而往往需要技术栈结合，完成特定需求。

例如某一条招聘信息中写道：“熟悉常见的后端编程语言，如Java、Python、C#等。熟悉常用的后端框架，如Spring、Django、ASP.NET等。对数据库设计和优化有良好的理解，熟悉常见的关系数据库和SQL语言。熟悉Web开发技术，如HTTP协议、RESTful API设计等。具备良好的问题解决能力和调试技巧。具备团队合作精神，能够与其他开发人员、产品经理和设计师紧密合作。技能要求：熟悉常见的后端开发工具和版本控制系统，如Git。了解云计算平台，如AWS、Azure或Google Cloud等。熟悉容器化技术，如Docker和Kubernetes。理解前端开发技术和相关的前后端交互。熟悉软件开发生命周期和敏捷开发方法。有良好的代码编写习惯和文档编写能力。对系统性能优化和安全性有基本的认识。”

需要一定的技术栈，多种编程能力的结合，主要体现了编程思维。这样的应用开发岗位切面比较广，从互联网公司到各种软件开发、车载系统、充电桩开发、量化交易等，行业范围非常广泛，对技术栈的要求比较分散，但是通常都需要多技术栈，综合应用开发。

词项分布： 0.024 项目 / 0.020 技术 / 0.019 经验 / 0.017 设计 / 0.016 团队 / 0.014 沟通 / 0.012 需求 / 0.010 研发 / 0.009 工程师 / 0.008 代码 / 0.007 前端 / 0.007 学习 / 0.006 分析 / 0.006 系统 / 0.006 优化 / 0.006 理解 / 0.005 软件 / 0.005 框架 / 0.005 c++ / 0.005 计算机 / 0.005 合作 / 0.005 独立 / 0.004 组件 / 0.004 维护 / 0.004 岗位 / 0.004 平台 / 0.004 解决 / 0.004 责任心 / 0.004 持续 / 0.004 善于 / 0.004 golang / 0.004 编程 / 0.004 意识 / 0.004 数据结构 / 0.004 编码 / 0.004 算法 / 0.003 架构 / 0.003 发展 / 0.003 规范 / 0.003 质量 / 0.003 提升 / 0.003 端 / 0.003 模块 / 0.003 测试 / 0.003 解决问题 / 0.003 描述 / 0.003 功能 / 0.003 知识 / 0.003 后端 / 0.003 协调

数据展示：

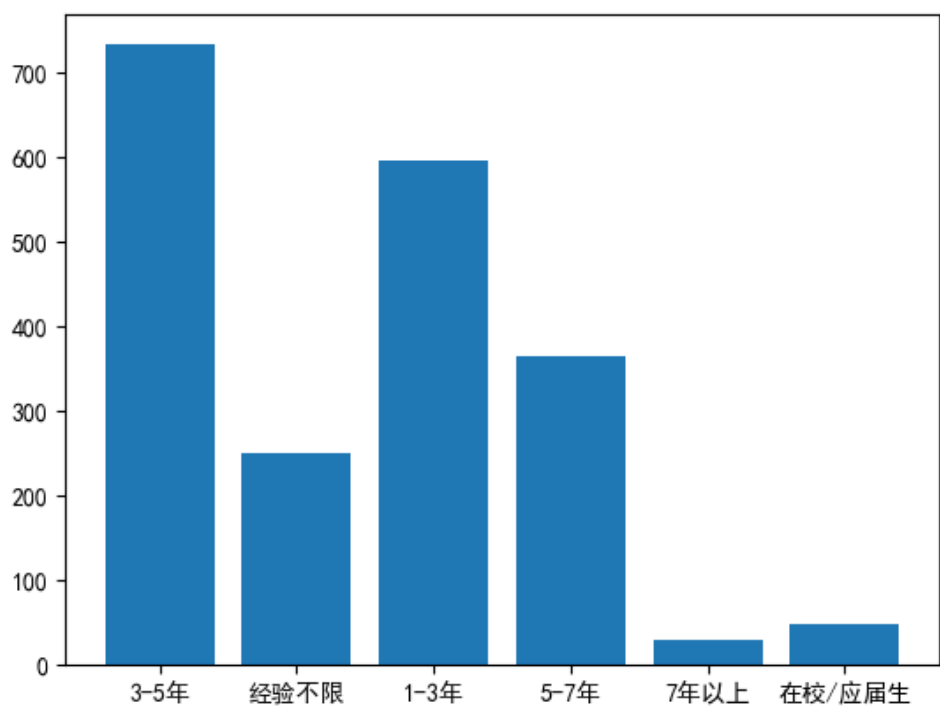


图3-1-1-19 应用开发岗位的经验要求

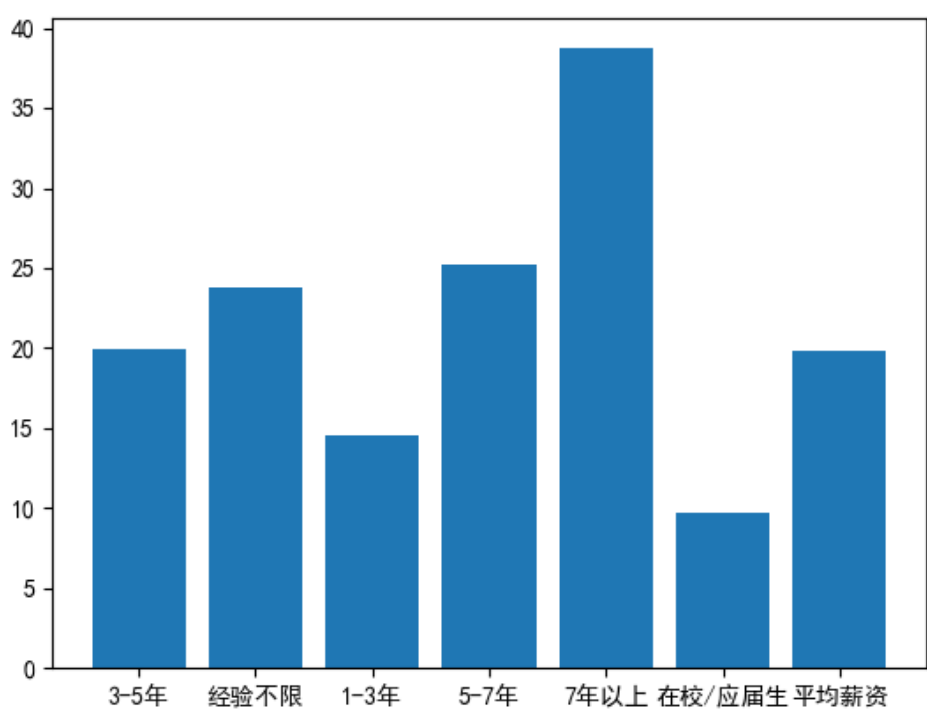


图3-1-1-20 应用开发岗位的经验薪资

可以看到应用开发岗位的经验要求多是有工作经验的应聘者，并且随着工作年限的增加薪资也爱一直增加，这是因为应用开发岗位一般都比较有针对性，例如车载系统的开发往往需要应聘者精通相关领域的综合技术栈，仅仅只会几门编程语言是不够的，所以随着工作年限的增加薪资也在增加，因为在这一领域内逐渐变得更加精通了。

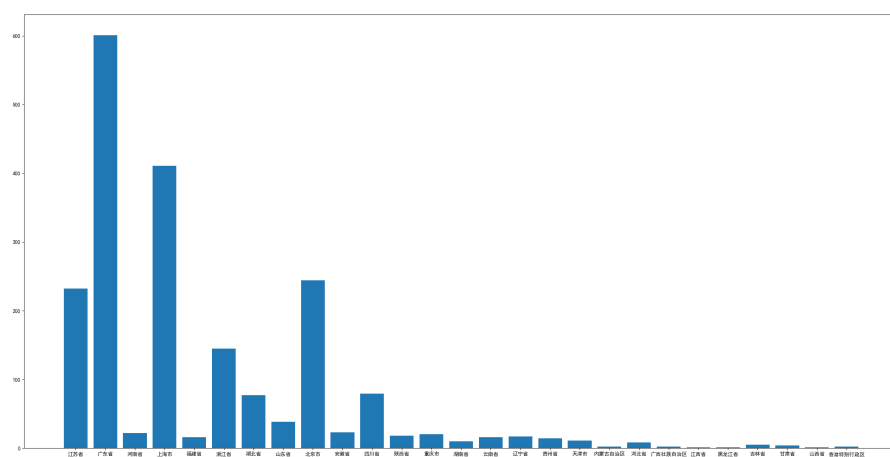
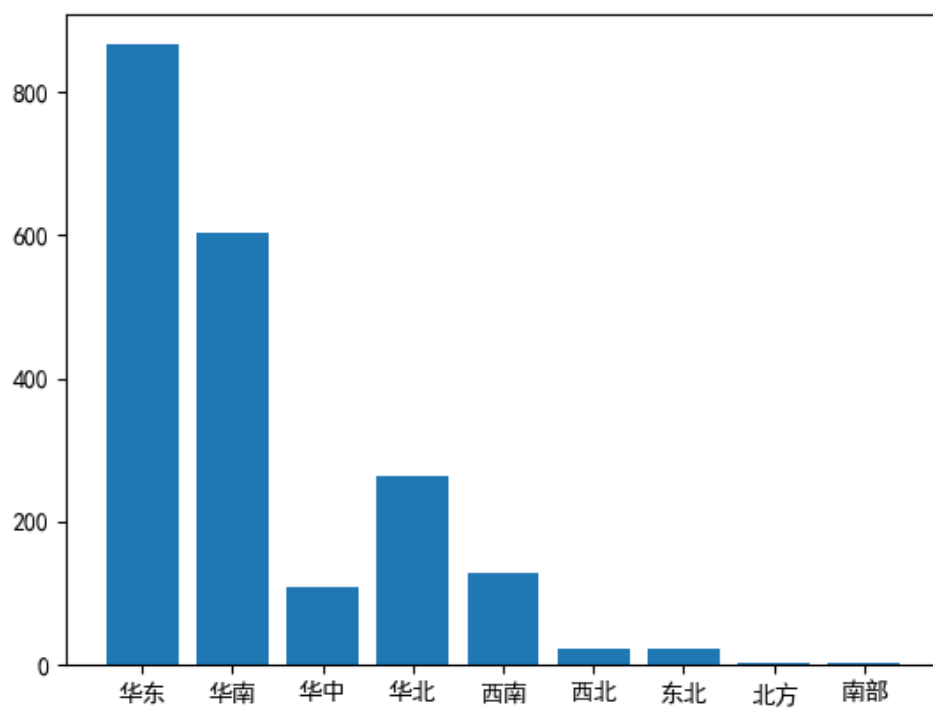


图3-1-1-21 应用开发岗位的地区分布

这里可以看到仍然是在华东、华南和华北地区为主，北上广街领的超一线城市和一线城市就业。

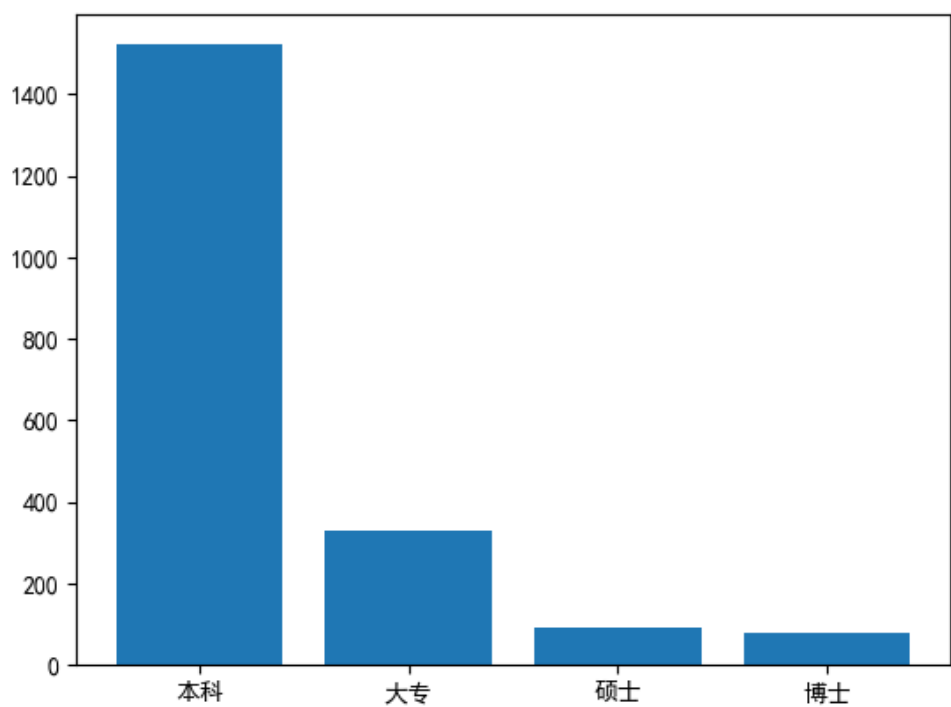


图3-1-1-22 应用开发岗位的学历要求

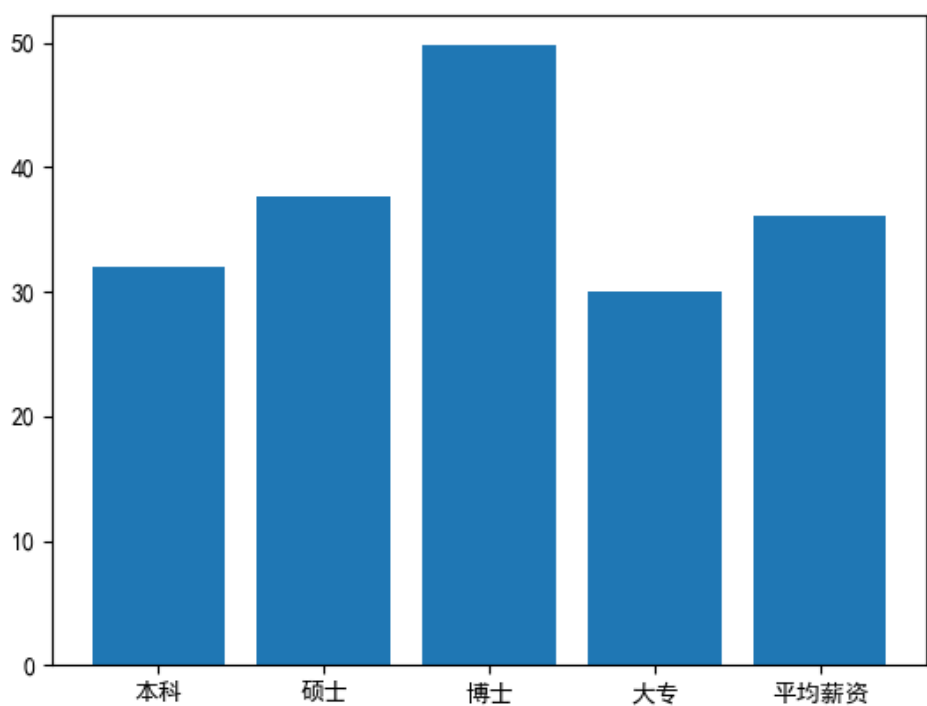


图3-1-1-23 应用开发岗位的学历薪资

应用开发岗位主要的学历要求还是本科，但是在薪资方面，随着学历的增加薪资有着明显的增加，并且普遍薪资比较高。这都是因为工作岗位比较有针对性，在相关领域长久从事磨练出来的精通专业知识、硕士博士深入某一相关领域的研究带来的专业针对性综合技术应用带来的高薪。

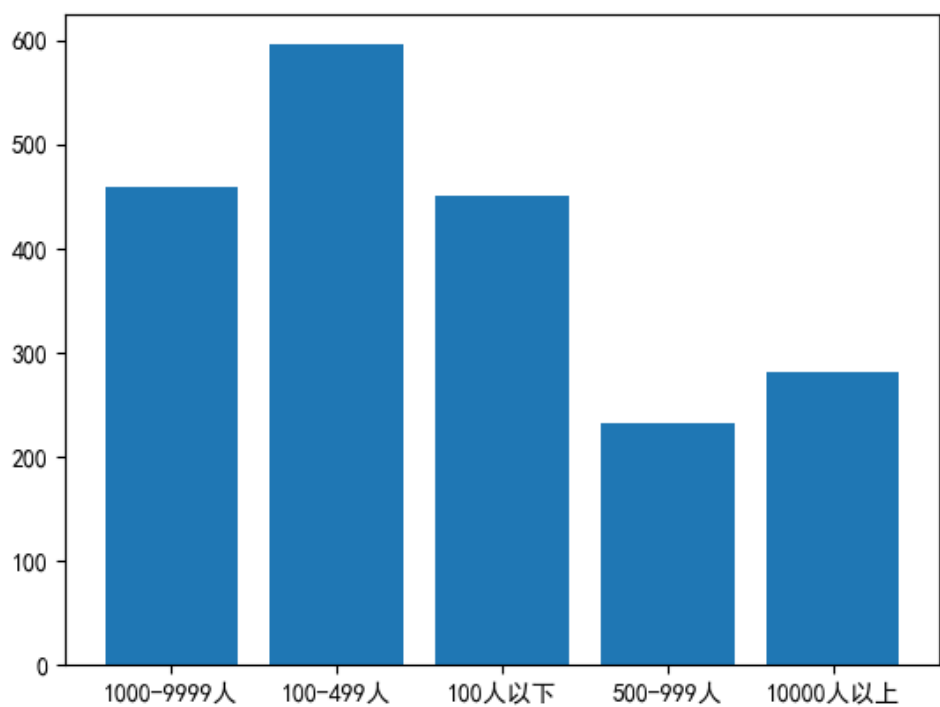


图3-1-1-24 应用开发岗位的企业规模

5. C/C++开发:

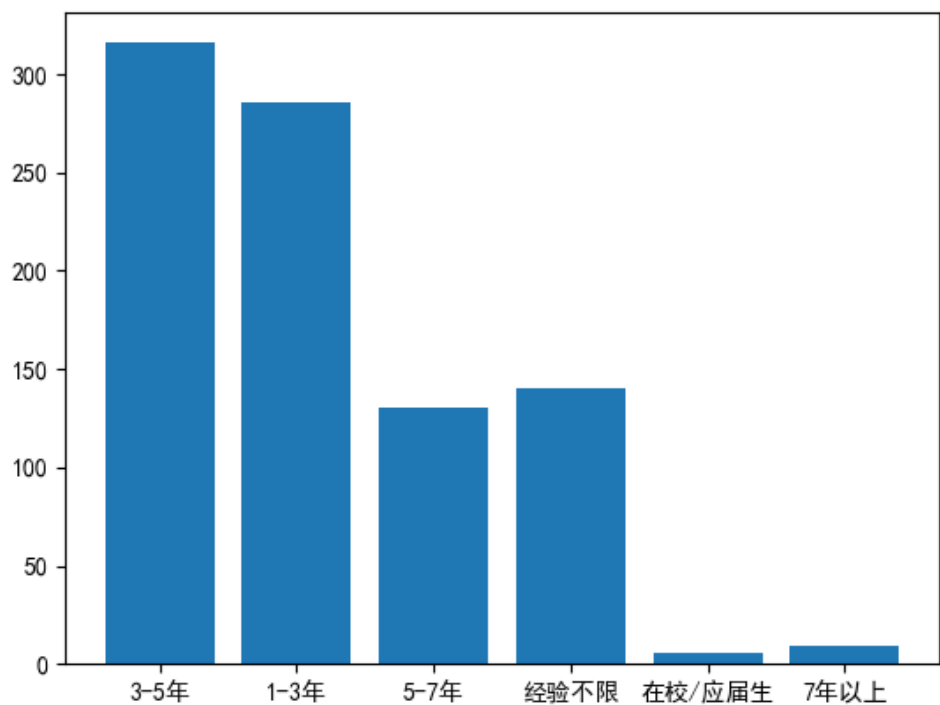


图3-1-1-25 C/C++开发岗位的经验要求

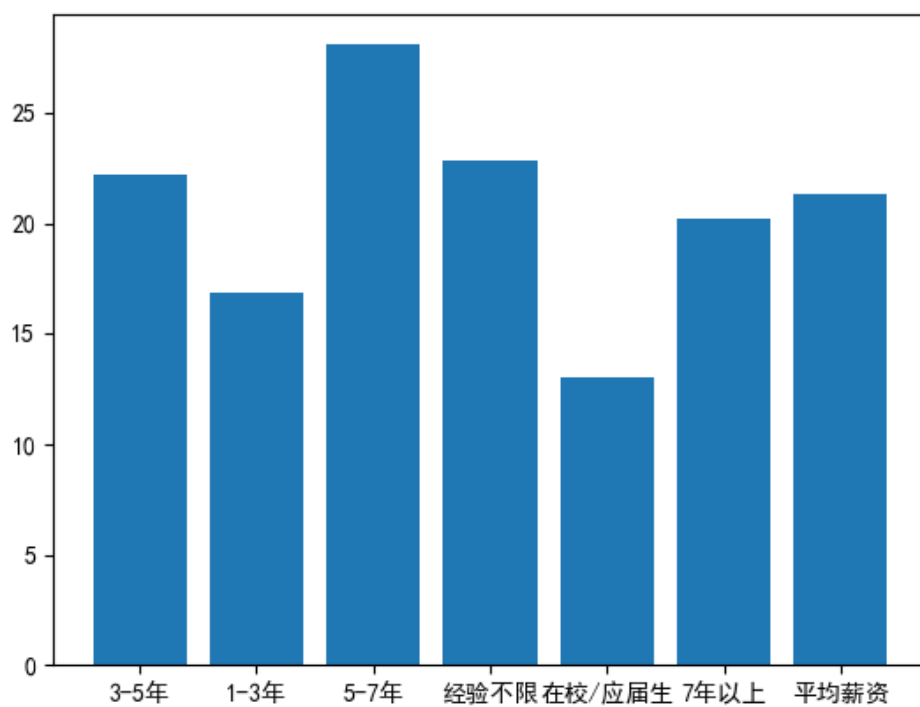
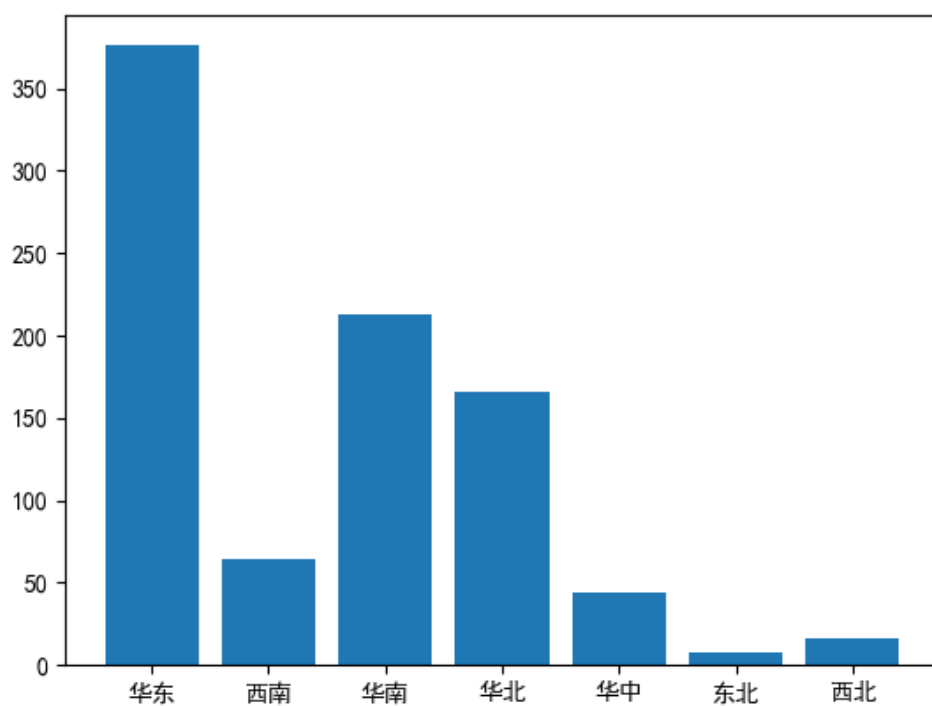


图3-1-1-26 C/C++开发岗位的经验薪资



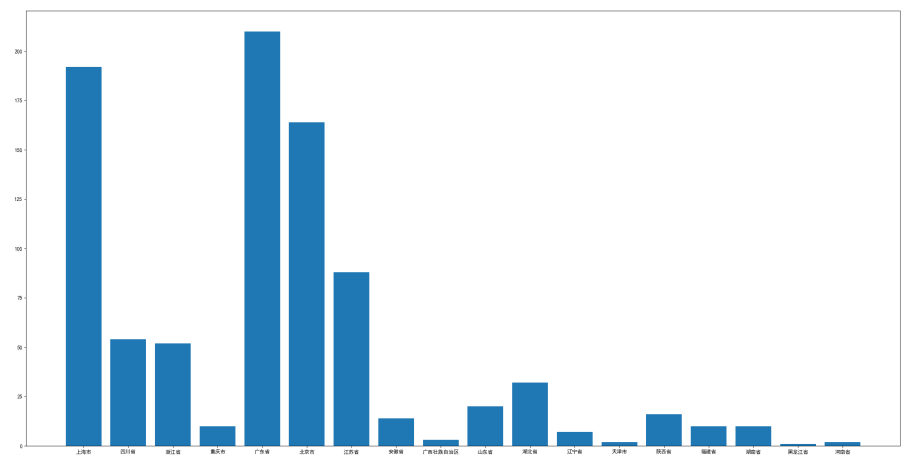


图3-1-1-25 C/C++开发岗位的地区分布

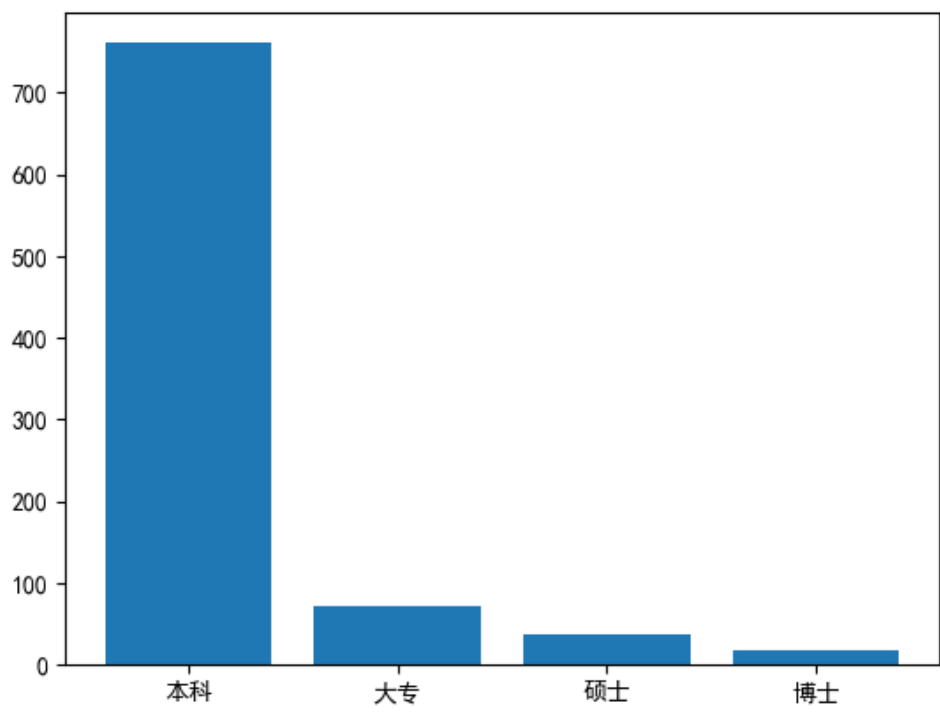


图3-1-1-26 C/C++开发岗位的学历要求

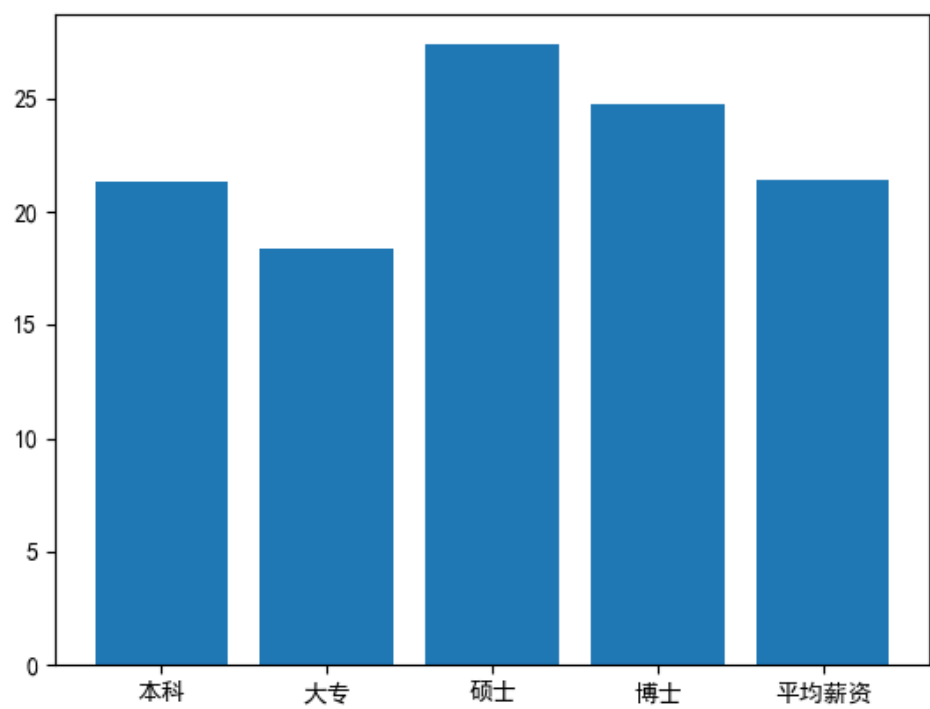


图3-1-1-25 C/C++开发岗位的学历薪资

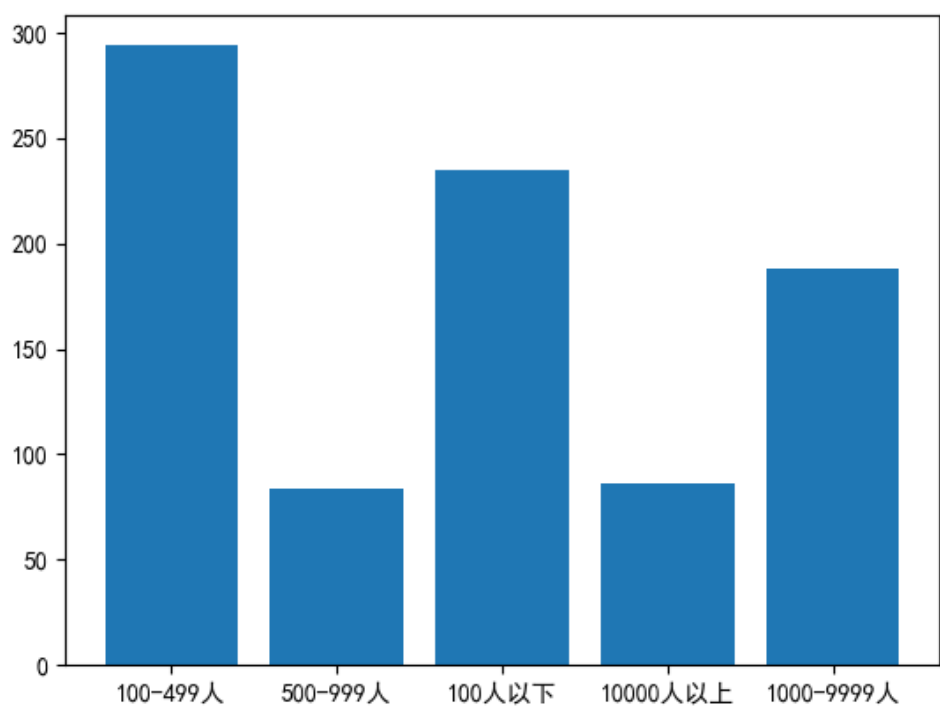


图3-1-1-26 C/C++开发岗位的企业规模

3.1.2 软件测试

3.2 通信工程专业

3.2.1 通信技术

3.3 电子科学与技术专业

3.3.1 光电

3.3.2 数字电子

3.4 电子信息工程专业

3.4.1 导航

3.4.2 电气控制

3.4.3 嵌入式

3.5 计算机专业

3.5.1 人工智能

3.5.2 算法

3.6 网络工程专业

3.6.1 网络安全

3.7 微电子与集成电路专业

3.7.1 微电子与集成电路

3.8 数据管理专业

3.8.1 数据分析

3.8.2 数据开发