指导文档

文件介绍

1.数据拆分

```
sh run_workflow.sh
```

首先的话需要执行 run_workflow.sh ,然后获得一个新的csv文件，这个文件也就是利用 api 提取工作内容、岗位要求、技术栈。 （待优化，接口准确但是比较慢）

2.数据合并

```
python job_info_trans.py        #涉及到时间、省份新增列

python job_merge.py          #设计工作岗位合并
```

然后的话执行一个 python脚本完成相似岗位的合并，这里主要是做了一个替换，将同一类的岗位名称修改为一个，方便后续进行岗位数量、工作地点等方面的统计。

3.星图绘制

```
python draw_skills.py
```

这个脚本主要是开启一个窗口，用户可以在下滑窗口中选中一个，然后绘制一个该岗位的 技术栈 星图。同理下面的几个命令：

```
python draw_contents.py

python draw_requires.py
```

contents代表工作内容，requires代表岗位需求

优化工作

1. 工作地点，工作岗位的加入

- [x] 首先来说的话，这个应该放置在第二点数据合并之后进行处理，更为关键的是怎么去处理呈现呢？
  - [x] 我把时间拆分成 三列 年 月 天。把省份 ， 市 作为两列， 补全空白公司信息，设置默认公司岗位招聘人数为1。 然后写入到一个新的csv文件中。
    这个转换的话是在 job_info_trans.py
  - [x] 数据的呈现的话涉及到图像、岗位合并。然后同类统计呈现。

2. 数据切割

- [ ] 优化方向：争取不调用大模型接口，利用小模型或者规则
  - [x] 实验了一个qwen2-7B的模型也没有快，现在的话大概一千条三分钟左右。
  - [ ] 规则化的提取岗位职责、岗位能力

3. draw_*.py

- [x] tk窗口中文编码问题
- [ ] 多文件整合
- [x] 星图添加排行榜

提取岗位相关数据，包括年度、需求企业数量、人才需求量、按省需求量分布等。 排行榜显示

- [x] 模型读取整理

4. 6.19 修复模型加载问题

- [x] 使用模型加载的绝对路径，
  其中涉及到的文件有 draw.py draw_*.py

5. 数据修复

- [x] 去掉前缀工作内容：\n 岗位职责：\n

  修复后的文件：Boss直聘_skills.csv

6. 规则指定呈现

- [x] 设立一个 def filter_dataframe(df, job_name=None, province=None, start_time=None, end_time=None, top_n=None, sort_by=None) 的函数，用来排序。当参数被赋予的时候，该参数固定，否则该参数为该参数的全集。 就比如来说限制了时间，那么就是只看那段时间的数据。反之不限制，就得看所有时间的数据。

7. 构建新的合并表

- [ ] 构建新的合并表，
  - [ ] 合并项

    先合并所有相似岗位名称->新表（一类一名），
  - [ ] 必要性?

    我做这个表是因为每次绘制星图的时候都要去计算相似度研究top10,这个开销其实可以给到存储。

    然后待定的点是，做一个怎样的新表？做一个只有类别的怎么样？也就是说这个表的形式大概是下面这样的:

岗位类别	岗位地点	岗位薪资	岗位发布时间	岗位招聘人数	岗位需求	岗位能力	岗位内容
  - [ ] 代价

    之前构建的新表策略不能再用，那个新表构建的时候可以保留各自的岗位名称，但是要新增一列 cluster， 要通过这个与 上面构想表有联系。然后查询的时候可以通过 cluster过渡到上表。
  - [ ] 要不要构建技能表，内容表

    又有了一个新的思路，每一列映射为一张表，然后的话大概简单一些，两列就行了

content	cluster

然后这样的话，就是说，对于每一个元素都可以去通过映射，然后得到各自的一个类，也就是把一张文字表，变为一张数字表（也可以是文字表了）

大概规划变为：

| 职位名称 | 工作地址 | 学历要求 | 工作年限要求 | 招聘人数 | 薪资待遇 | 公司行业 | 公司性质 | 公司规模 | 融资阶段 | 招聘状态 | 职位类型 | 岗位描述 | 公司介绍 | 公司工商信息 | 简历详情页地址 | 更新日期 | 工作内容 | 任职资格 | 技术栈 | 日期 | 市 | 省 |