会议总结

2020-8-6

**报告总结：**

1. 对于OC-tree及其可视化部分

(孔德旭)在数据结构方面，我们根据分词结果构造了前缀树，并合并多余分支以达到压缩的效果。

设计的数据结构包含结点信息和员工职位信息。

在节点信息中包含以下字段：Node\_id, name, count(对于叶节点而言是该职位对应的简历数目，非叶节点该项目为0), score(结点重要程度，目前根据对应子树上的count和), resume\_id:(uid, rid)<该项对于非叶节点也为空>, interval。

在员工职位信息部分，则记录了员工的user\_id及按照时间存储的对应年份职位的依序展开节点的node\_id列表。

(彭志远)在数据结构的基础上，根据echart插件开发出了基于时间变化的职位树状图，该图具有以下特点：

1. 在拖动时间轴的时候树状图的展开状态可以得到保持，变化过程相对平滑。
2. 树状图可以根据选定的员工项展现对应年份该员工的任职情况，该情况也可随着时间轴的拖动进行变化，并且也会在时间轴变化中保留展开情况。
3. 保留了放大缩小，拖拽等功能。
4. (刘志威)对于标签的设计涉及多个领域，我们将其划分为不同的组，组是一个或多个标签的集合，同组内的标签存在互斥关系，根据不同组来识别标签目前作为履历主题的初级划分。

目前是对不同组的标签以关键字匹配的判断规则得到目标标签，目前视标签为履历主题的最细粒度划分。

首先我们观察输入数据中是否存在isexist内的关键词，若存在，则选择此标签组。接着判断输入数据是否匹配上remove中的元素，若匹配上则取消选择本组标签，如果通过前两次关键词匹配，则依序匹配各addition key，若匹配上了则返回对应标签，若均未匹配上，则返回default标签。

目前运行情况，运行速度较快，并且匹配结果保持一定的准确度和信息量。

1. （李可）在分词部分，首先使用jieba分词。为了弱化通用分词器倾向于将分词碎片化的特点，我们决定通过以下方法：
2. 通过jieba预分词。
3. 通过预分词结果建立OC-tree并进行压缩
4. 将OC tree压缩后的前缀加入结巴的词典中
5. 在使用jeiba 进行再分词

在此过程中，我们使用precision，recall和f1-score去量化模型的优劣。通过本方法可以发现，模型的准确度和f1-score有明显的进步。

为了不局限于jieba此类基于匹配的分词方法，还使用了HMM和PKUseg Toolkit(CRF)两种算法进行分词，并结合上述两部分词探寻新的分词器，但目前如何提升模型准确度还值得研究。

**讨论问题：**

1. 由于大部分实际标签需要符合多个搜索条件以及时间参数，如何将这些条件与参数包含在知识库里，尽量避免在新增标签或修改规则时修改代码。
2. （王钺老师）需要明确标签学习面向业务的应用场景。例如进行岗位画像：根据每个岗位历届任职人员的经历标签反向分析该岗位所需的标签。利用该信息分析岗位性质，或帮助用人单位选择岗位的人选。
3. （张林老师）现阶段标签搜索系统的应用场景是“自然语言-标签搜索引擎-候选人数据库”这一沙漏型系统的中间部分。
4. 未来研究方向：参考基于OpenAI的GPT-3模型的自然语言MySQL指令生成算法，通过自然语言自动进行标签搜索
5. 在拥有标签搜索引擎的情况下，可以进行个人能力图谱，职位图谱和班子图谱的建立。

**下一步工作计划：**

1. 可视化：
	1. 调研echart组件能否支持50000个点以上的展示以及平滑变动
	2. 以用户为中心展开的个人职业轨迹子树。
2. 基于知识的标签学习系统
	1. 建立多层级的知识库：在基于文本识别的底层标签的基础上，定义由多个底层标签的逻辑组合定义的高层标签，并在查询query中加上时间这一可选参数。
3. 分词系统
	1. 在当前测试数据中分析基于OC-Tree分词结果降低测试集recall的原因。
	2. 进一步研究如何提升分词准确度，例如结合多种分词方法的混合模型
4. 先锋数据-社交网络分析
	1. 明确community detection等社交网络问题在先锋数据应用场景中的研究意义