

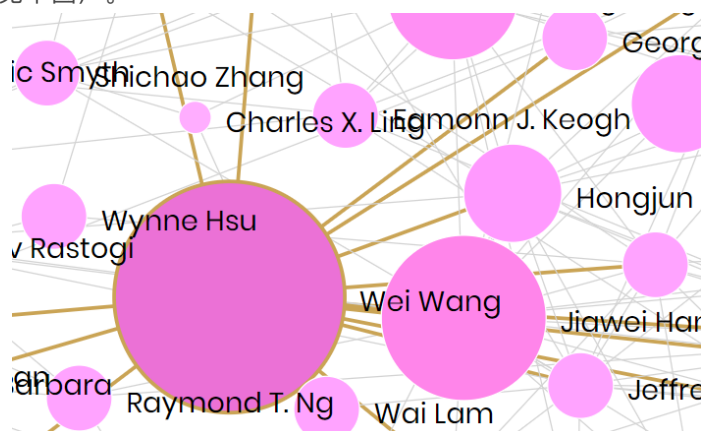
Midway Report

项目简介

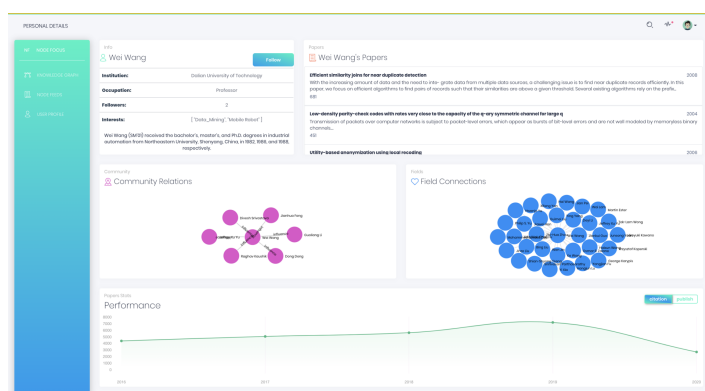
NodeFocus: 深度挖掘学术信息

作为学生或者科研工作者，人们常常希望能够快速了解某个学术领域的发展动态、发现领域内较为有影响力的学者，甚至与其开展合作。虽然类似Google Scholar的学术搜索网站已经能够提供特定学者的详细信息，但是却没有很好地挖掘“学术群体”的潜在价值。

NodeFocus是一个基于领域和影响力的大规模学者关系图谱及其可视化网站，我们的目的是将其打造成类似**LinkedIn**（一个职场社交平台）的学术社交网站。基于可视化的图谱，用户不仅可以快速检索到相关领域的知名学者，查看学者的详细信息，还可以对与其关系密切的合作者一目了然（见下图）。



如果用户对某位学者感兴趣，可以选择“Follow”该学者，便可以跟踪其最新动态。同时我们挖掘了每位学者所在的“学术圈”（主要包括与其合作紧密的学者、其受影响最大以及影响最大的学者、导师、学生等），帮助用户了解其所在学术群体的信息（见下图）。同时，**NodeFocus**还提供“学者影响力比较”功能，从多方面对学者进行影响力评估。



基于以上功能，**NodeFocus**希望为寻找暑研、导师的学生群体以及有意进行学术社交的研究人员提供切实帮助。

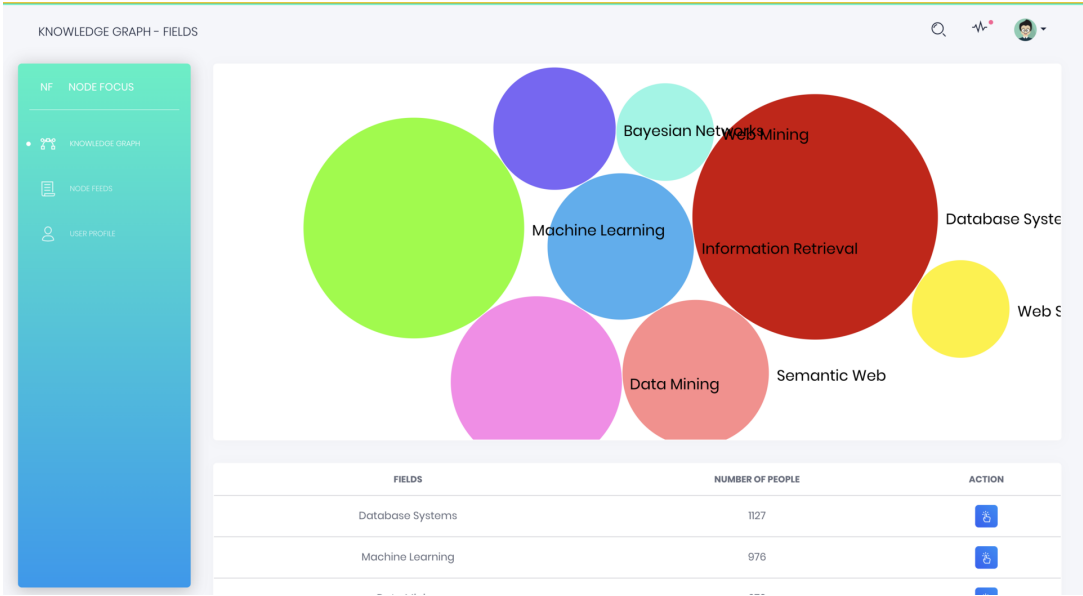
中期总结

已完成功能

前端：

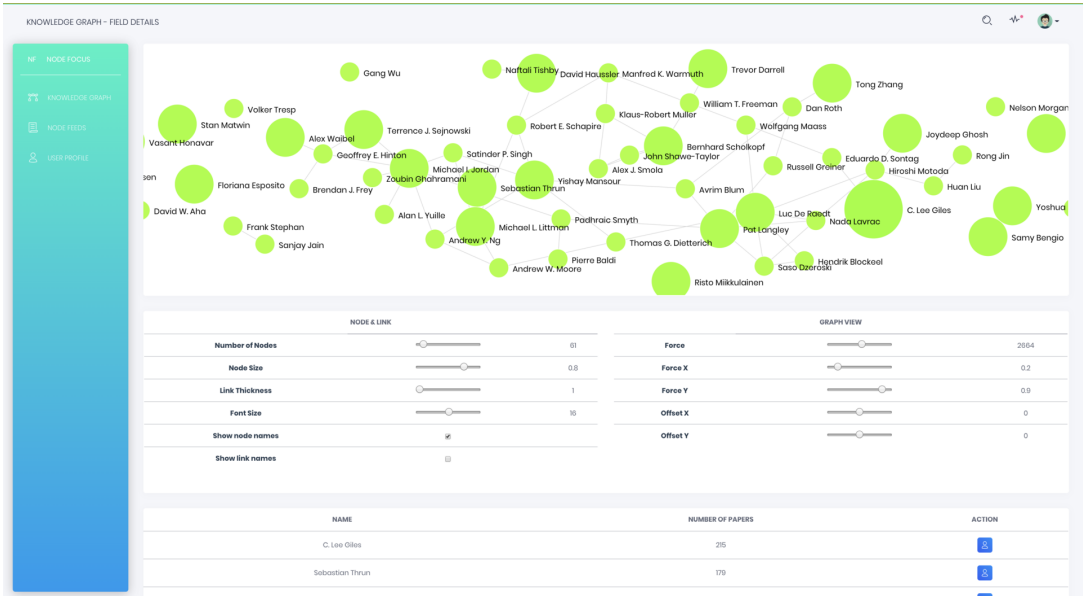
- Fields Page:

可以显示领域节点，并且根据领域热度调整节点大小。可以进行节点/列表的选择，并且相应的列表/节点会被highlight。点击列表的Action按钮会被链接到相关领域的details。



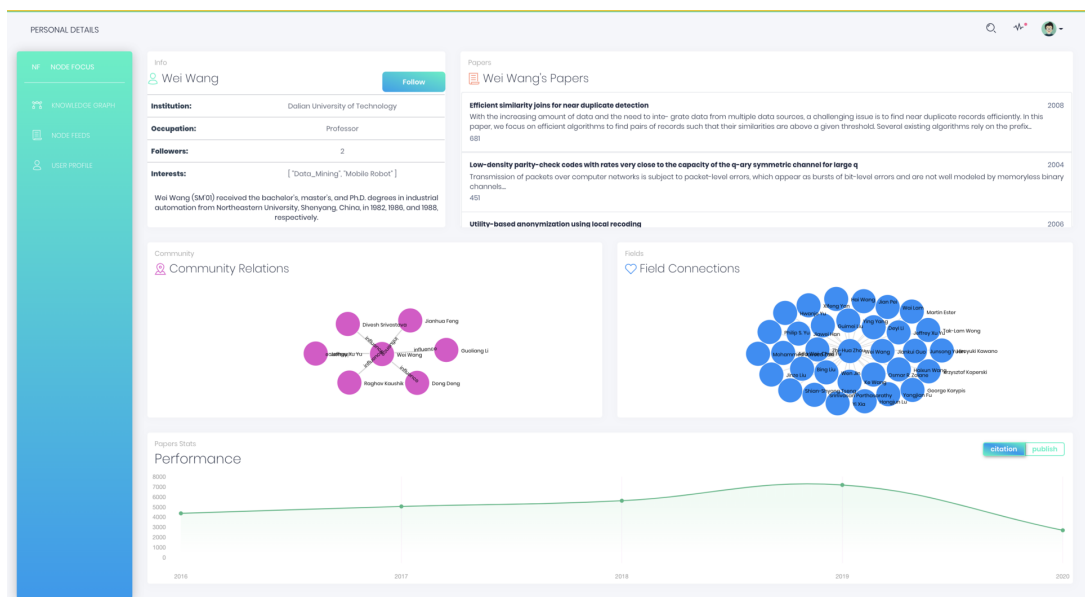
- Field Details Page:

领域的details页面，在这里会显示该领域的学者们及他们之间的关系图。
节点的大小是根据学者的影响力来决定的，而这里的影响力是根据我们自己的算法来算出来的。
同时该图还支持节点大小、force大小、节点数目等一些图的参数的调整。
并且同样支持和Fields Page一样节点/列表的选择功能。
点击Action下面的人图标就可以进到Personal Details Page



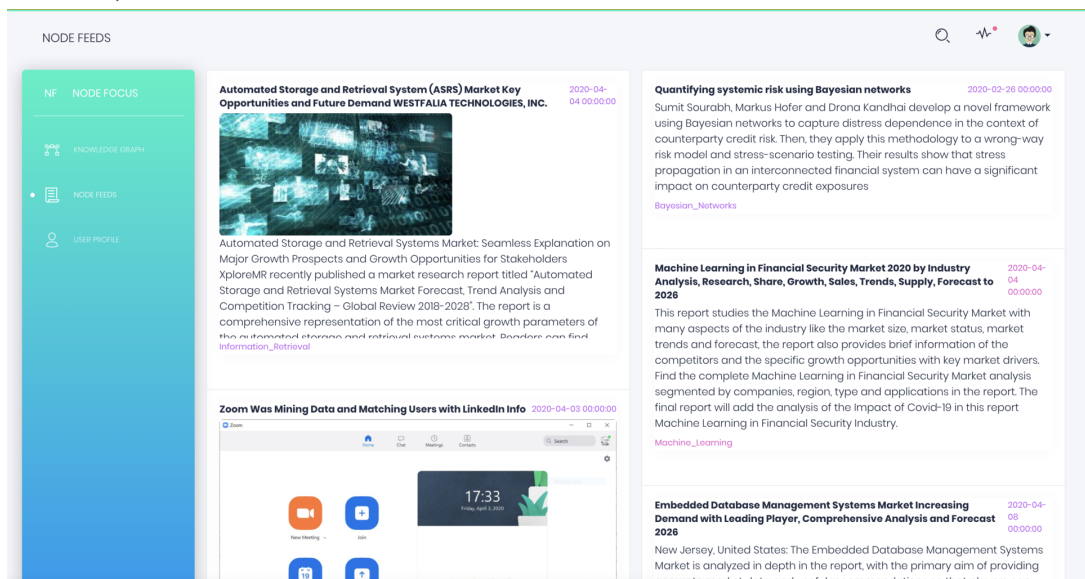
- Personal Details Page:

该页面主要是用于显示个人信息，主要有：Personal Brief Info、Papers、Community Relations（用来显示有什么社交群体）、Field Connections（用来显示在同一领域里面的关系）、以及Papers的引用次数、发布数量。
还有一个Follow的功能（待添加），即用户可以Follow这个学者，并在NodeFeeds页面看关注学者的最新动态。



- Node Feed Page:

新闻页面，用于显示Follow之后节点的最新动态。



后端：

网站部署以及前端提供数据获取的api。

主要api有：

- `baseurl+'/'fields'`，获取领域视图下的节点和关联数据。
- `baseurl+'/'fields/' + query`，获取特定领域下符合要求的节点和关联数据
- `baseurl + '/'person/' + query` 获取特定学者的详细数据
- `baseurl+'/'news/' + query` 获取满足要求的新闻数据

团队分工

前端：成镇宇

后端：李振宇

遇到的困难和解决方案

前端：

- 前端遇到的困难主要都是跟使用的vue-d3-net work库有关，因为这个库的不稳定因素较多，并且性能较差，所以在实现的过程中需要通过不少“弯路”来解决一些库不能实现的问题。比如节点之间的碰撞关系，必须得给库添加自己的custom force方案，而且必须是某种格式的方案才可以用，所以功能非常有限。
- 同时由于是第一次接触bootstrap和sass，所以在学习、尝试、及debug上花了很多的功夫。
- 还有就是前端的视觉因素，到现在配色方案还是一个很大的问题。如何实现统一颜色色调贯彻整个前端，这个对项目的系统化开发有较高的要求，但又由于对语言、框架、工具的不熟悉，所以暂时实现的不太好看。还需要改进。
- 还有js的shallow copy和deep copy的问题，这个问题在vuejs框架中就显得非常不清晰，有的时候shallow copy就可以实现component之间的prop传输，但有的时候就只有deep copy才能够更实现。

后端：

- 由于之前没有接触过Flask和mongodb，所以很多知识和操作都是第一次尝试，虽然学习的过程中花费了些时间也遇到了很多小bug，但是好在目前用到的操作还比较简单，网上的教程也比较详细，所以最终困难也都被一一克服。
- 另外一点是关于数据的问题，由于网站所需要的数据比较多元，包括学者的详细信息、论文数量变化、引用量变化、所在学术群体等信息，原来使用的Aminer上的一个小数据集是不够用的，所以还需要使用爬虫从Semantic Scholar、Micorsoft Academic等网站上爬取，这个过程中还涉及到数据一致性问题。所以直到目前网站后端的数据仍然没有齐全，这也是正在尽快处理的事务。

整体：

- api：一般api开发是前端成镇宇会在api convention中写下需要的api的request和response的格式，并给李振宇来进行后端的api开发。但在实际开发中，因为前端显示总会有一些参数变化，所以会要求后端重写或是修改api，这个对效率非常不友好。所以决定后端的api也由成镇宇来开发，让李振宇着重于后端数据收集和整理。

工作的优缺点

根据小组互评的结果，整体评论算是不错的。大多数都表示前端开发较为完整，需要的就是例如字的大小、节点的可阅读性、可操作性等的一些细节上的改进。

下阶段工作计划

下一阶段的工作主要可以分为以下5点：

- 个性化服务：
 - a. 用户注册登陆等用户机制
 - b. 用户之间互相follow，建群等用户之间的交互机制
- Knowledge Graph：
 - a. 节点的可视性
 - b. 节点对比
 - c. 节点之间的dijkstra
- NodeFeeds：

- a. 爬取并显示Facebook, Twitter等大平台的feed
 - b. 后端自动化信息爬取
- 节点搜索功能：

根据关键词生成相应的KnowledgeGraph

- 美学：
 - a. 整体颜色方案
 - b. 元件之间的布局等
- 高并发(optional)

时间表：

Time	Goals
9th	美学 && NodeFeeds
10th	个性化服务 && NodeFeeds
11th	Knowledge Graph && 节点搜索功能
12th Progress Report 2	gap week for backup / or work on Concurrency
13th	Add more personal data and display them
14th	
15th	Maybe try mobile frontend if possible
16th	Final