

文本复制检测报告单(全文标明引文)

№:BC202105122141571811839938

检测时间:2021-05-12 21:41:57

检测文献: 新冠肺炎交互数据可视化设计 & 分析

作者: 袁鑫

检测范围: 中国学术期刊网络出版总库

中国博士学位论文全文数据库/中国优秀硕士学位论文全文数据库

中国重要会议论文全文数据库

中国重要报纸全文数据库

中国专利全文数据库

图书资源

优先出版文献库

大学生论文联合比对库

互联网资源(包含贴吧等论坛资源)

英文数据库(涵盖期刊、博硕、会议的英文数据以及德国Springer、英国Taylor&Francis 期刊数据库等)

港澳台学术文献库

互联网文档资源

源代码库

CNKI大成编客-原创作品库

时间范围: 1900-01-01至2021-05-12

检测结果

去除本人文献复制比: 3.2%

去除引用文献复制比: 3.2%

总文字复制比: 3.2%

单篇最大文字复制比: 0.9% (数据挖掘中可视化的探究与应用)

重复字数: [539]

总段落数: [4]

总字数: [17057]

疑似段落数: [3]

单篇最大重复字数: [158]

前部重合字数: [419]

疑似段落最大重合字数: [419]

后部重合字数: [120]

疑似段落最小重合字数: [41]



指标: ☐ 疑似剽窃观点 ☒ 疑似剽窃文字表述 ☐ 一稿多投 ☐ 疑似整体剽窃 ☐ 过度引用 ☐ 重复发表

表格: 0 公式: 没有公式 疑似文字的图片: 0 脚注与尾注: 0

10.9%(419) 10.9%(419) 新冠肺炎交互数据可视化设计 & 分析_第1部分 (总3833字)

1.2%(41) 1.2%(41) 新冠肺炎交互数据可视化设计 & 分析_第2部分 (总3492字)

4%(79) 4%(79) 新冠肺炎交互数据可视化设计 & 分析_第3部分 (总1979字)

0%(0) 0%(0) 新冠肺炎交互数据可视化设计 & 分析_第4部分 (总7753字)



(注释: 无问题部分 文字复制比部分 引用部分)

1. 新冠肺炎交互数据可视化设计 & 分析_第1部分

总字数: 3833

相似文献列表

去除本人文献复制比: 10.9%(419) 文字复制比: 10.9%(419) 疑似剽窃观点: (0)

1	数据挖掘中可视化的探究与应用 吴天昊(导师: 黄善国) - 《北京邮电大学硕士论文》 - 2017-12-01	4.1% (158) 是否引证: 否
2	大数据审计中的可视分析 李强; 谢汶莉; - 《中国内部审计》 - 2016-02-15	4.0% (154) 是否引证: 否
3	数据可视化实践对数据新闻团队的启示 方爱华; 张解放; - 《新闻世界》 - 2014-06-10	1.9% (74) 是否引证: 否
4	2019-nCoV新型冠状病毒肺炎疫情数据可视化分析系统 张铭杨 - 《大学生论文联合比对库》 - 2020-05-31	0.9% (36) 是否引证: 否

5	2019-nCoV新型冠状病毒肺炎疫情数据可视化分析系统 张铭杨 - 《大学生论文联合比对库》 - 2020-06-05	0.9% (36) 是否引证: 否
6	电子与信息工程学院-计算16-3-1606010327-张铭杨-查重正文 张铭杨 - 《大学生论文联合比对库》 - 2020-06-10	0.9% (36) 是否引证: 否
7	统计类数据分析与WebGIS可视化的研究 黎彦宏(导师: 芦鹏飞) - 《北京邮电大学硕士论文》 - 2020-06-30	0.8% (31) 是否引证: 否

原文内容

绪论

研究背景

1.1.1 数据化时代下的疫情

全球领先的管理咨询公司麦肯锡认为：“数据今天已经渗透到每一个行业和业务功能领域，成为重要的生产要素”。当今时代是一个快速发展的时代。人们通过互联网提高了彼此之间的信息交换效率，在这个过程中会产生大量的数据，所谓“大数据”就是这个信息化发展时代的产物。正因如此，当今时代可以称之为‘数据化’时代，互联网每时每刻都在产生数据，而以某一中心词作为核心，其相关数据积累到一定的量时，我们可以利用相关技术对这些数据进行挖掘和分析，回顾其以前的发展状况并预测其以后的发展态势。自2019年12月疫情在武汉爆发以来，在互联网上以‘疫情’为中心词的相关数据每日剧增，截至目前为止，疫情相关数据量已达百万级别，同时其数据种类也十分丰富，包括数值数据、文本数据、多变量时变时间、地理信息数据、点数据等等。在这些数据中有许多是值得我们去深入挖掘和分析的，例如文本数据种类中的舆情数据，我们可以收集到一些高频的舆情信息然后再通过官方平台进行造谣或者证实，再例如地理信息数据中被确诊人员近几日的活动路线数据，我们可以对其密切接触的人和地区采取一些必要的防控措施等等。当然，除了这些数据以外还有很多与疫情相关的一些数据值得我们去进一步挖掘和分析。

1.1.2 传染病领域中的数据可视化

(1) 数据可视化概述

数据可视化是技术与艺术的完美结合，主要目的是利用计算机图形学手段清晰有效地传达关键信息，通过直观的手段来展示一些抽象、不易理解的信息，是理解复杂数据不可缺少的技术。一方面，因为海量的数据，可视化应运而生；另一方面，可视化可以帮助用户从海量的数据中捕获关键信息，从有限的空间中融入更多的数据，因此数据与可视化相辅相成，两者缺一不可。随着计算机技术的成熟和搜索引擎技术的发展，政府信息的开放性和众包模式的兴起，人们获取和解读数据的可能性大大提高，基于数据挖掘、理解数据可视化成为新的发展方向 and 突破口。

(2) 可视化在传染病领域上的应用

可视化技术应用于传染病领域可以追溯到较早的 19 世纪，英国医生通过标记地图发现了霍乱的源头，帮助控制当时的伦敦霍乱疫情，这也被认为是数据可视化的应用典范。随着计算机技术推动可视化技术的不断进步，许多学者开始探索面向大数据的更加多元化的传染病相关领域可视化方法。例如金思辰等通过构建交互式可视分析系统，帮助用户分析疾病的时空分布，利用热力图辅助查看聚类信息，并通过案例对系统进行评估。胡雪芸等利用可视化技术分析肺结核疾病。吴静等分析了与传染病相关的可视化虚拟仿真实验教学。李拓对信息可视化技术在疫情中的应用案例进行分析，以提供有价值的可视化范本。

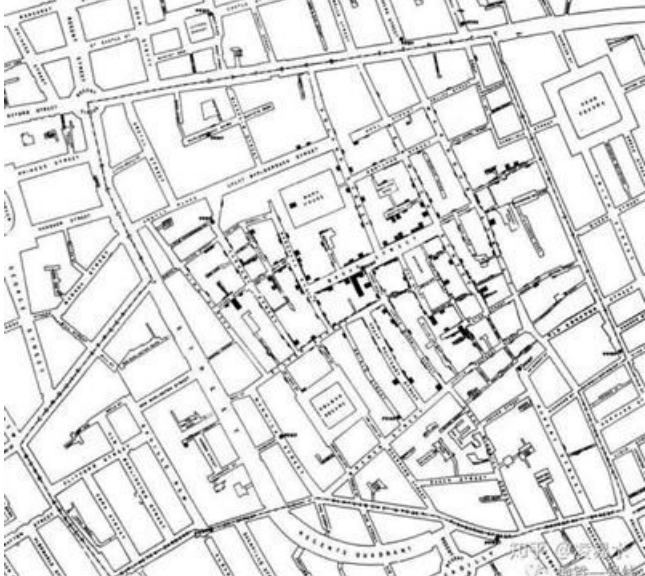


图1.1 英国医生采用“标记地图”的可视化方法发现了霍乱的来源

(3) 国内外疫情可视化研究现状

由于2020年的新冠肺炎疫情的爆发，由疫情产生的大量的有潜在价值的信息进一步推动了数据可视化在传染病领域的运用。中国对全球新冠肺炎疫情信息系统进行了升级,利用地图直观显示全球新冠肺炎疫情的相关信息，实现了一些数据可视化的应用。

丁香园是国内最早对疫情进行数据可视化的平台，其主要展示了疫情的实时数值数据、疫情地图展示以及近一周疫情的发展态势包括确诊趋势、治愈趋势、死亡趋势等，因为该平台主要是应用于移动端，所以其能够提供的交互性很弱，同时平台提

供的数据量不大且大部分的都是实时数据。总的来说该平台主要是起到了疫情预警的作用，并不能够做到疫情大数据的可视辅助分析。继丁香园推出疫情可视化平台后，国内三大厂百度、阿里、腾讯也纷纷上线了疫情可视化平台，这些平台的整体设计与丁香园推出的平台均大同小异。



图1.2 中国丁香园企业推出的疫情数据可视化平台

世界卫生组织在其官方网站上也推出了名为“Coronavirus disease (COVID-19) Situation dashboard”的数据可视化平台，其展示的数据涵盖了全球各国，与此同时该平台针对不同区域的疫情扩散情况进行了一定的可视分析，但是就平台整体而言还是存在着可视交互较弱、视觉映射较差等情况，笔者体验后感觉该平台很难捕获到关键性数据。



图1.3 世界卫生组织在官方网站上推出的疫情数据可视化平台

以上的可视化研究状况是数据可视化在传染病相关领域的有益探索，但大部分工作主要集中在对实时数据的可视化展示，或者面向普通用户的可操作性以及可视化交互性偏弱且不利于用户对数据的理解，或者仅提出相应的设计范式，缺少整合性的平台支持等。

1.2 研究意义

基于上文提到的各平台的缺点与不足，本文主要是提出一系列有关疫情大数据的可视化图表模型并设计出相关的可视化平台，一是能够让用户直观便捷地观察到疫情的动态趋势实现预警于民，二是设计出合理的可视化方案，借助于人眼快速的视觉感知和人脑的智能认知能力，通过可视化交互加深用户对疫情数据的理解并辅助数据的分析，三是结合知识图谱可视化技术，构建出几种新冠知识图谱并进行交互可视化，以图谱的方式向用户普及新冠相关的知识网络。从而做到疫情的实时追踪与预警，辅助预防、分析和把控新冠肺炎疫情的发展。

相关技术及框架

2.1 相关技术

2.1.1 人机交互

人机交互是指人与系统之间通过某种对话语言以某种方式进行信息交换的过程。系统可以是各种机器、计算机和软件。用户界面或人机界面是指人机交互所依赖的媒体和对话界面，通常包括硬件和软件系统。信息可视化的概念最早是在ACM召开的用户界面软件与技术会议上提出的。它的本质是一个交互式图形用户界面范例。一方面，人机交互的发展强调了对智能用户界面的研究，使计算机系统成为一个具有思想、个性和观点的智能机器人。另一方面，要充分利用计算机系统和人的优势，弥补彼此的不足，共同分析和解决问题。本文在描述如何建立可视化模型的过程中，结合可视化交互技术，提出了几个符合平台背景的交互设计原则，从而加深用户对数据的理解，辅助数据分析。

2.1.2 知识图谱

知识图谱（Knowledge map visualization，简称知识域可视化）或图书情报行业中的知识域地图（Knowledge domain mapping map，Knowledge domain mapping map），是一系列表示知识发展过程与知识结构关系的图形。它利用可视化技术描述知识资源及其载体，挖掘、分析、构建、绘制和展示知识及其关系。基于知识图谱和数据可视化分析等相关技术，本文将提出一系列基于新冠肺炎知识图谱数据可视化研究的平台设计方案和原则，通过抓取、采集、清洗、挖掘、对新冠肺炎疫情大数据等相关操作进行分析和知识图谱构建多种可视化模型，然后通过交互可视化等技术在平台上，将疫情相关知识网络以直观、易于理解和操作的方式呈现给用户，展示了新冠肺炎相关知识载体之间的联系，一定程度上为用户普及疫情相关知识，从而进一

步加深用户对疫情相关知识理解，实现对民众的预警。

2.1.3 可视分析

可视分析是融合了科学与信息可视化、人机交互、认知科学、数据挖掘、信息论、决策论等相关研究领域的一门交叉技术。Thomas和cook根据2005年给出的定义，认为可视化分析是一门通过交互式可视界面帮助用户分析和推断大规模复杂数据集的科学和技术。可视化分析的运作过程可以看作是一个数据—知识—数据的循环过程。它经历了两条主线：可视化技术和自动分析模型。从数据中理解知识的过程主要依赖于两条主线的相互作用与合作。在本文中，可视分析技术主要应用于平台设计规范，平台将会根据可视分析技术设计出一套合适交互式可视化界面。

2.2 相关框架

2.2.1 Vue.js

国内一款比较火热的前端框架。Vue框架的最大特点是从下到上逐层应用，因此被称为是渐进式框架。同时Vue的核心库仅关注视图层，通过NPM包管理工具，可以非常容易与第三方库结合，在本平台中IviewUI、Echarts、Swiper等等第三方库都是通过NPM十分便捷地引入到了平台项目中。同时该框架可以在平台中构建单页应用，实现响应组件和视图组件的数据绑定，构建丰富的用户界面视图。在本研究中该框架可用于搭建平台，利用数据驱动实现图表的动态更新与渲染。

2.2.2 Echarts

一个基于JS的开源可视化图表库，包括各种各样的交互式图表，能够满足很多领域图表要求。同时其兼容性较强，能够兼容大多数电脑端和移动端的主流浏览器。Echarts还提供了一个用户作品交流社区Make A Pie（链接：<https://www.makeapie.com/explore.html>），可以在社区里面浏览他人的作品以给自己提供制图思路。在本研究中Echarts主要用于实现新冠肺炎数据的可视化，定制出丰富的可视化图表并结合Vue框架设计出合适的可视化交互方式。

2.2.3 Asynchronous Javascript And XML

AJAX，即“异步JavaScript和XML”（异步JavaScript和XML），在平台中用于网络请求获取对应后端API数据接口，同时结合Vue框架数据驱动的特点，将AJAX异步请求到的数据动态地渲染到网页视图中，这样就实现了网页的局部刷新。在平台中Ajax技术主要用于请求后台疫情数据以实现数据视图的实时更新。

指 标
疑似剽窃文字表述
1. 随着计算机技术的成熟和搜索引擎技术的发展，政府信息的开放性和众包模式的兴起，人们获取和解读数据的可能性大大提高，基于数据挖掘、理解数据可视化成为新的
2. 过程。系统可以是各种机器、计算机和软件。用户界面或人机界面是指人机交互所依赖的媒体和对话界面，通常包括硬件和软件系统。信息可视化的概念
3. 是一个交互式图形用户界面范例。一方面，人机交互的发展强调了对智能用户界面的研究，使计算机系统成为
4. 机器人。另一方面，要充分利用计算机系统和人的优势，弥补彼此的不足，共同分析和解决问题。
5. Thomas和cook根据2005年给出的定义，认为可视化分析是一门通过交互式可视界面帮助用户分析和推断大规模复杂数据集的科学和技术。可视化分析的运作过程可以看作是一个数据—知识—数据的循环过程。它经历了两条主线：可视化技术和自动分析模型。

2. 新冠肺炎交互数据可视化设计_第2部分			总字数：3492
相似文献列表			
去除本人文献复制比：1.2%(41) 文字复制比：1.2%(41) 疑似剽窃观点：(0)			
1	基于云平台架构乘客信息系统技术方案研究 汪晓臣;赵伟慧;黄志威;孙同庆;田源：- 《第十三届中国智能交通年会大会论文集》- 2018-11-07	1.2% (41)	是否引证：否
2	智慧亲子平台 林立伟 - 《大学生论文联合比对库》- 2017-05-19	1.2% (41)	是否引证：否
3	基于微信公众号的智慧亲子供应链平台 林立伟 - 《大学生论文联合比对库》- 2017-05-25	1.2% (41)	是否引证：否
4	基于微服务架构的医疗业务经办系统的设计与实现 李含笑 - 《大学生论文联合比对库》- 2018-05-28	1.2% (41)	是否引证：否
5	基于微服务架构的医疗业务经办系统的设计与实现 李含笑 - 《大学生论文联合比对库》- 2018-06-02	1.2% (41)	是否引证：否
原文内容			

第三章可视化平台的设计

3.1 平台总体架构设计

3.1.1 平台总体架构概述

为实现数据的高效性存储，服务的分层性运行，数据的交互式可视化分析，平台基于浏览器并使用多元编程语言和框架技术来进行前后端交互。考虑到数据的庞大，时间的分布性明显，采用 hive 数据仓库来进行大量高效率存储。同时，为了进行及时稳定的交互，使用mysql关系型数据库来存储实现系统功能的数据。后端采用 MVC 架构风格，分离服务层、控制层、持久层、Dubbo+Springboot 面向微服务架构，将本平台所需数据封装成一个个订阅服务，前端使用 Vue、Echarts、IviewUI等框架，并设计相关请求接口，向后端服务进行数据请求，并将数据进行可视化展示。



图3.1 平台总体架构概述

3.1.2 相关概念

(1) 前后端交互

前后端交互即web前端与web后端的数据交互。前端所需的数据主要是通过网络请求来完成的，前端发送请求，后端接收到前端的请求后，便进行对数据库的操作，返回前端所需要的各项数据，即可完成一次前后端的交互。一次前后端交互的大致流程如下：

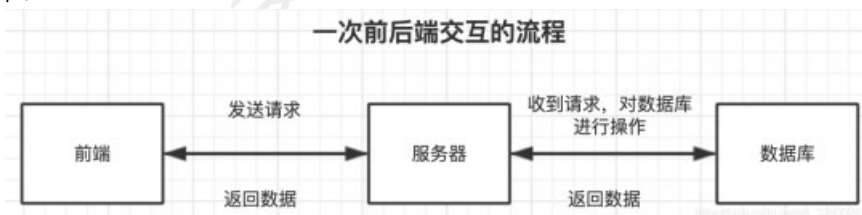


图3.2 一次前后端交互的流程

平台通过前端交互技术具有更好的解耦性，后端只需要将数据封装到一个API中，前端通过网络请求技术调用API即可获取所需的疫情数据，并通过echarts框架渲染成各种交互图表。

(2) 微服务架构

微服务是一种架构的风格，大型复杂软件的设计可以分为一个或多个微服务。所有的微服务都可以独立配置完成相应的块任务功能，所有的微服务都可以独立配置。同时，每个微服务都是松耦合的，而服务的内部是高度内聚的。其出发点是解决测试成本高、可扩展性低、可靠性低、替换难度低、语言水平低等单一架构的问题。

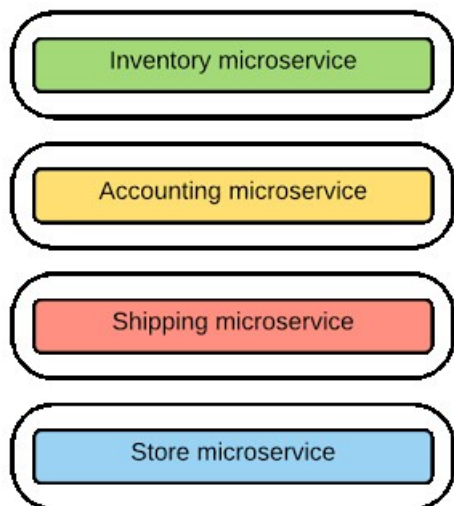


图3.3 微服务架构示意图

本项目中将各个模块功能划分为一个个模块任务，每一个功能任务分装成微服务，将其置于服务器，并设置专门的RestFul API，从而可以供外界访问，其无状态通信具有很强的可读与可用性。

3.2 平台页面结构设计

3.2.1 平台页面结构概述

平台页面是基于Vue-cli脚手架构建的SPA（单页面应用）项目，均是运行在浏览器端的HTML页面，同时平台采用了大量的预加载和懒加载以及交互动画等形式使用户能够畅通无阻地浏览平台的各个页面。平台页面整体分首页、趋势挖掘、时空分析、知识图谱、新闻谣言五个板块。

3.2.2 相关概念

(1) SPA（单页面应用）

SPA即web单页面应用，在本平台中单页面主要是由Vue框架中vue-router路由管理器实现，通过配置相关的路由内容就可以实现不同路由内容的切换，而这些内容的切换都局限于一个页面之中，这样既避免了页面因刷新而导致的静态资源重新加载的问题又在一定程度上避免了用户频繁切换页面而出现的白屏和闪屏等现象，从而进一步提高了用户的浏览体验感。同时由vue-router构建的单页面应用提供了动态路由匹配、嵌套路由、程式导航、命名路由等功能，可以让开发者灵活地制作路由也可

以让用户快速辨别出对应路由页面的内容，提高路由灵活性和可读性。

(2) 预加载

预加载即在所有网页加载之前先加载开发者期望提前加载的一些主要内容，主要目的是为用户提供更好的体验，减少等待时间。否则，如果页面所需加载的内容过大，没有设置预加载功能的页面会很长时间处于空白状态，直到所有内容都加载完毕才能够正常地显示出页面内容，大大地降低了用户的体验感。在本平台中预加载主要是提前加载一些大型的静态资源，然后利用浏览器缓存，在后续加载这些静态资源的时候就不会出现卡顿和空白等情况。

(3) 懒加载

懒加载顾名思义就是在页面初始加载的时候并不会把所有的资源都加载完毕，当用户浏览到相关资源内容的时候再进行加载也就是我们常说的延迟加载。在一些大型的网站平台中，其需加载的资源非常多，如果不采用懒加载的方式就会出现页面初始化资源过久，等待时间较长等问题，同时由于页面内容过多也会造成网页卡顿和网页崩溃等现象。在本平台中懒加载的实现主要是通过IviewUI框架中自带的懒加载组件结合JS的滚轮事件以实现部分资源的延迟加载，提高网页的渲染效率。

3.3 平台数据库设计

3.3.1 平台数据库概述

平台采用 Mysql数据库管理疫情数据信息，疫情数据来源于世界卫生组织官网、Aminer官网、阿里健康官网、丁香园官网且数据统计区间为2020年1月1日至2020年12月31日，主要数据库表格为国内疫情实时数据表(china_real_time)、国内疫情趋势表(china_trend)、国际疫情趋势表(international_trend)、国内疫情统计表(china_statistics)、国际疫情统计表(international_statistics)、疫情相关知识图谱表(knowledge_map)。在知识图谱表格设计中，因为是采用Mysql数据库进行数据管理，所以在表格中设定关键字connection用于存储该节点存在的关系网络，由节点唯一标识id和分隔符“，”构成，关键字connection_name用于存储该节点存在关系网络中对应的关系名，由关系名和分隔符“，”构成。

3.3.2 平台数据库相关表格

(1) 国内疫情实时数据表(china_real_time)

字段名	类型	约束	允许为空	说明
real_time	varchar	无	是	实时时间
area	varchar	无	是	地区
current_confirmed	float	无	是	现存确诊
all_case	float	无	是	累计确诊
all_cured	float	无	是	累计治愈
all_death	float	无	是	累计死亡
current_input	float	无	是	现存境外输入
china_statistics	varchar	无	是	所属大类

字段名类型约束允许为空说明
real_time varchar 无是实时时间
area varchar 无是地区
current_confirmed float 无是现存确诊
all_case float 无是累计确诊
all_cured float 无是累计治愈
all_death float 无是累计死亡
current_input float 无是现存境外输入
china_statistics varchar 无是所属大类

(2) 国内疫情趋势表(china_trend)

字段名	类型	约束	允许为空	说明
time	varchar	无	是	时间
area	varchar	无	是	地区
add_confirmed	float	无	是	现存确诊
add_case	float	无	是	累计确诊
add_cured	float	无	是	累计治愈
add_death	float	无	是	累计死亡
add_input	float	无	是	现存境外输入
china_statistics	varchar	无	是	所属大类

字段名类型约束允许为空说明
time varchar 无是时间
area varchar 无是地区
add_confirmed float 无是现存确诊
add_case float 无是累计确诊
add_cured float 无是累计治愈
add_death float 无是累计死亡
add_input float 无是现存境外输入
china_statistics varchar 无是所属大类

(3) 国内疫情统计表(china_statistics)

字段名	类型	约束	允许为空	说明
time	varchar	无	是	时间
area	varchar	无	是	地区
current_confirmed	float	无	是	现存确诊

all_case	float	无	是	累计确诊
all_cured	float	无	是	累计治愈
all_death	float	无	是	累计死亡
current_input	float	无	是	现存境外输入

字段名类型约束允许为空说明

time varchar 无是时间

area varchar 无是地区

current_confirmed float 无是现存确诊

all_case float 无是累计确诊

all_cured float 无是累计治愈

all_death float 无是累计死亡

current_input float 无是现存境外输入

(4) 国际疫情趋势表(inter_trend)

字段名	类型	约束	允许为空	说明
time	varchar	无	是	时间
area	varchar	无	是	地区
add_confirmed	float	无	是	现存确诊
add_case	float	无	是	累计确诊
add_cured	float	无	是	累计治愈
add_death	float	无	是	累计死亡
add_input	float	无	是	现存境外输入
inter_statistics	varchar	无	是	所属大类

字段名类型约束允许为空说明

time varchar 无是时间

area varchar 无是地区

add_confirmed float 无是现存确诊

add_case float 无是累计确诊

add_cured float 无是累计治愈

add_death float 无是累计死亡

add_input float 无是现存境外输入

inter_statistics varchar 无是所属大类

(5) 国际疫情统计表(inter_statistics)

字段名	类型	约束	允许为空	说明
time	varchar	无	是	时间
area	varchar	无	是	地区
current_confirmed	float	无	是	现存确诊
all_case	float	无	是	累计确诊
all_cured	float	无	是	累计治愈
all_death	float	无	是	累计死亡

字段名类型约束允许为空说明

time varchar 无是时间

area varchar 无是地区

current_confirmed float 无是现存确诊

all_case float 无是累计确诊

all_cured float 无是累计治愈

all_death float 无是累计死亡

(6) 疫情相关知识图谱表(knowledge_map)

字段名	类型	约束	允许为空	说明
id	float	无	否	节点唯一标识
name	varchar	无	是	节点名
connection	varchar	无	是	节点的关系网络
connection_name	varchar	无	是	节点的关系网络中对应的关系名
url	varchar	无	是	节点相关信息网址

字段名类型约束允许为空说明

id float 无否节点唯一标识

name varchar 无是节点名

connection varchar 无是节点的关系网络

connection_name varchar 无是节点的关系网络中对应的关系名

url varchar 无是节点相关信息网址

3. 新冠肺炎交互数据可视化设计_第3部分			总字数: 1979
相似文献列表			
去除本人文献复制比: 4%(79) 文字复制比: 4%(79) 疑似剽窃观点: (0)			
1	汽车转向管柱测试数据可视分析方法研究与实现 李杭芮 - 《大学生论文联合比对库》 - 2019-05-22	4.0% (79)	是否引证: 否
2	农村电商大数据交互展示系统开发 彭威枫 - 《大学生论文联合比对库》 - 2018-05-30	2.1% (42)	是否引证: 否
原文内容			

第四章平台交互式可视化的设计

4.1 交互式可视化设计的目的

可视化主要是由视觉呈现和交互两部分组成。可视化中的交互是从数据中发现信息的重要手段，可以缓解有限的可视化空间与数据过载之间的矛盾。例如，对于高维数据，分组可用于视觉交互中的降维。由新冠肺炎疫情产生的数据中，有数据量大、数据维度高、数据种类丰富等特点，所以必须采取合适的可视交互手段在有限的空间内高效地展示各类别的数据，并从用户的角度出发，针对用户的操作习惯和视觉习惯，使得用户更易理解可视化视图中的各项数据元素。

4.2 交互式可视化的具体方法

4.2.1 颜色映射及区间选择

在数据的展示过程中，我们可以通过视觉映射的方式对不同区间的数据进行颜色上的映射，从而使得用户能够快速感知到不同区间的数据分布情况，同时用户也可以通过区间选择组件选择自己关注的区间来进行展示。在平台中我们在地图组件中应用了这一交互方法。



图4.1 颜色映射及区间选择过滤交互方法展示图

4.2.2 层级下钻

在多层级数据中，用户往往只关注某一层的数据而不想关注其他层的数据或者用户想了解当前层级某一数据中更深层的数据，因此我们在做可视交互的过程中必须保证视图中同一层级的数据有一定视觉关联性又得保证父子层级有一定的交互关联性，所以我们可以采用层级下钻的方式来满足用户的需求。一般层级下钻包括多层级下钻折线图、多层级下钻饼状图、多层级下钻地图等，在平台中层级下钻交互方法主要是用于处理不同地域级别的数据交互展示，例如洲际级与国家级、国家级与省级、省级与市级等。





图4.2 层级下钻地图

4.2.3 鼠标移入悬浮提示

鼠标移入悬浮提示是交互式可视化常用的方法，主要是用于疫情相关数据的名称和详细信息的提示、相关数据的扩展提示、知识图谱中节点的相关解释以及节点与节点间的联系提示等，通过悬浮提示这种方式帮助用户更多了解疫情数据以及其含义。同时在平台的悬浮提示设计中另外加入了形象化的小图标作为标识，利用人眼视觉匹配这一特性，帮助用户快速地在提示的数据与图表的数据之间产生关联。



图4.3 鼠标移入悬浮提示展示图

4.2.4 数据区域缩放

在一些疫情数据的呈现中，我们期望提供数据的总体概览，查看数据中特定部分的信息，或者去除异常值的影响。数据区域缩放主要是对数轴进行控制，通过数据过滤和设置内部的轴显示窗口来达到数据窗口缩放的效果，用户可以选择一个感兴趣的区间来直观分析该区间内的疫情发展情况。

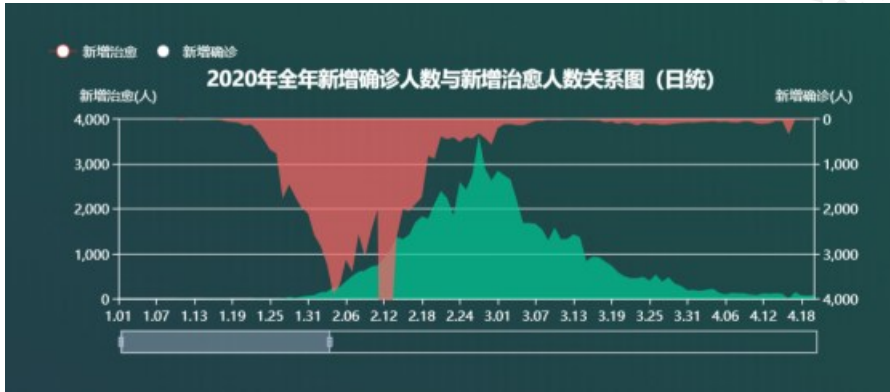


图4.4 数据区域缩放展示图

4.2.5 聚焦选择过滤

对数据进行可视化后，用户在关注某一部分数据时在视觉上往往会受到其他数据的干扰，换言之，在这种情况下，我们需要对用户关注的用户不关注的用户数据进行一定的展示处理。聚焦选择过滤就是通过用户选择的数据节点以及未被用户选择的数据节点进行突出以及淡化等处理。在疫情知识图谱的展示中，我们可以采用这种交互方式，在用户选择某一数据节点的时候高亮该节点和与其有关联的节点以及链接并淡化其他数据节点。



图4.5 聚焦选择过滤展示图

4.2.6 图例标识

在显示图表的过程中，往往有多个数据系列，图例是用不同的符号或图形标记不同的数据系列。图例展现了不同系列的标记，颜色和名字。用户可以通过点击图例控制哪些系列的显隐，可用于单独对比以及交叉分析等。它有着双重任务，既是编制图表时显示图表内容的标准，也是使用图表时不可缺少的阅读指南。

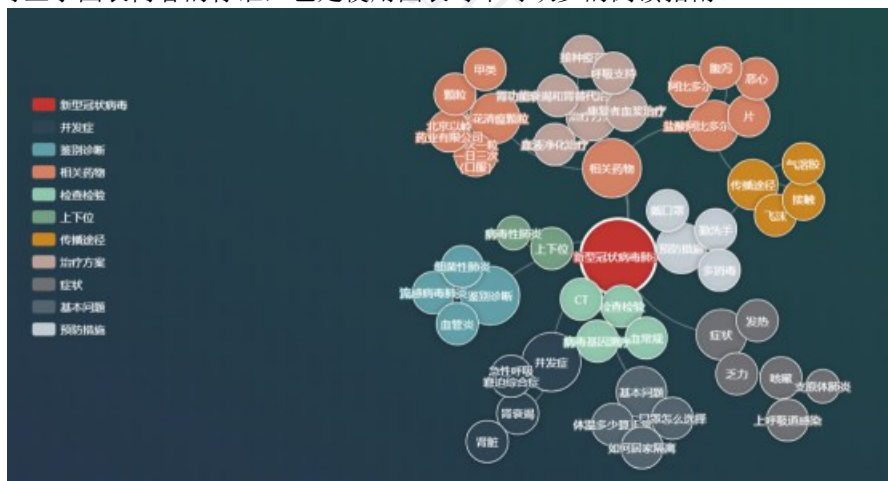


图4.6 图例标识展示图

4.2.7 时间轴自定义动态演示

时间轴是常用于展示时间序列数据的交互方法，支持不同序列数据的切换、播放以及交互等操作，适用于在可视化应用中展示不同时间段的数据变化情况，比如使用时间轴组件实现数据轮播。在传统的时间轴交互中一般都是固定轮播步长以及轮播的起止时间，而在本次交互设计中，我们加入了自定义动态演示这一功能，使用户能够随意对其关注的项目和时间段数据序列进行动态演示。



图4.7 时间轴自定义动态演示展示图

4.3 交互设计存在的不足

以上提到的交互设计基本能够满足疫情相关数据的展示，但是对于用户而言这些交互式可视化的展示还缺少了一定的个性化，即用户的可操作性还是较弱，其展示的图表形式还是较为单一。例如，我们是否可以允许用户进行图表修改或者制作新的配色方案，让用户可以生成符合自己审美风格的图表，同时也在一定程度上也增加了可视化平台的易用性和灵活性。除了配色方

案的修改与制作，图表样式的基本设置是否也可以通过可视化交互的方式来进行修改，这样就可以大大的提升图表的个性化，但是由于在可视化平台的交互设计的过程中往往受到技术框架和运行平台的支持性等限制，因此这些交互设计中存在的不足，我们一时无法进行突破，所以在未来交互设计中我们仍有很漫长的一段道路需要去走完。

指 标
疑似剽窃文字表述
1. 可以缓解有限的可视化空间与数据过载之间的矛盾。例如，对于高维数据，分组可用于视觉交互

4. 新冠肺炎交互数据可视化设计及分析_第4部分	总字数：7753
相似文献列表	
去除本人文献复制比：0%(0)	文字复制比：0%(0) 疑似剽窃观点：(0)
原文内容	

第五章可视化平台的具体实现

5.1 平台搭建准备工作

5.1.1 平台开发技术以及运行环境

分类名称

语言前端 Javascript

后端 Java

开发框架前端 Echarts、Vue、CesiumJS

后端 Springboot、SpringCloud 、Hive

表5.1 平台开发技术

分类	名称	版本	功能
语言	前端		Javascript
	后端		Java
开发框架	前端		Echarts、Vue、CesiumJS
	后端		Springboot、SpringCloud 、Hive

分类	名称	版本	功能
测试编辑环境	Windows系统	windows10	代码程序本地开发
测试及应用环境	Chrome浏览器	2020. 12. 1	疫情知识图谱数据可视化平台运行
实际运行环境	Linux	7. 2	大数据项目及javaee项目包运行及各种框架运行

分类名称版本功能

测试编辑环境 Windows系统 windows10 代码程序本地开发

测试及应用环境 Chrome浏览器 2020. 12. 1 疫情知识图谱数据可视化平台运行

实际运行环境 Linux 7. 2 大数据项目及javaee项目包运行及各种框架运行

表5.2 平台运行环境

5.1.2 平台后端相关API数据接口

请求Url示例	请求类型	返回数据格式	返回数据示例	解释
http://127.0.0.1/real_time/area	GET	JSON	{real_time: "2021/4/18", area: "china", current_confirmed: 529, all_case: 103273,}	国内实时疫情数据API接口，area字段为所需数据的地区
http://127.0.0.1/china_trend/area	GET	JSON	{data: [{time: "2020/1/1", area: "北京", add_confirmed: 120,}, {time: "2020/1/2", area: "北京",}],}	国内疫情趋势数据API接口，area字段为所需数据的地区
http://127.0.0.1/int_trend/area	GET	JSON	{data: [{time: "2020/1/1", area: "中国", add_confirmed: 1120,}, {time: "2020/1/2", area: "中国",}],}	国际疫情趋势数据API接口，area字段为所需数据的地区
http://127.0.0.1/china_statistics/area	GET	JSON	{data: [{time: "2020/1/1", area: "北京", current_confirmed: 89, all_case: 102,}, {time: "2020/1/2", area: "北京",}],}	国内疫情统计数据API接口，area字段为所需数据的地区
http://127.0.0.1/international_statistics/area	GET	JSON	{data: [{time: "2020/1/1", area: "中国", current_confirmed: 8929, all_case: 10201,}, {time: "2020/1/2", area: "中国",}],}	国际疫情统计数据API接口，area字段为所需数据的地区
http://127.0.0.1/knowledge_map/map_name	GET	JSON	{map: [{id: 1, name: "新冠肺炎", connection: [4, 6, 8,], connection_name: ["传播途径", "预防措施", "居家隔离",]}]}	疫情相关知识图谱数据API接口，map_name字段为图谱名

			家隔离 ",...],url:" https://www .163.com/dy/article/G7RV6 RTD055224PI.html"}...],m ap_name:" 疫情预防知识图 谱"}	
--	--	--	--	--

请求Url示例请求类型返回数据格式返回数据示例解释

http://127.0.0.1/real_time/area GET JSON

{real_time: "2021/4/18", area: "china", current_confirmed: 529, all_case: 103273,} 国内实时疫情数据API接口, area字段为所需数据的地区

http://127.0.0.1/china_trend/area GET JSON {data:[{time: "2020/1/1", area: "北京

", add_confirmed: 120,}, {time: "2020/1/2", area: "北京",},]} 国内疫情趋势数据API接口, area字段为所需数据的地区

http://127.0.0.1/int_trend/area GET JSON {data:[{time: "2020/1/1", area: "中国

", add_confirmed: 1120,}, {time: "2020/1/2", area: "中国",},]} 国际疫情趋势数据API接口, area字段为所需数据的地区

http://127.0.0.1/china_statistics/area GET JSON {data:[{time: "2020/1/1", area: "北京

", current_confirmed: 89, all_case: 102,}, {time: "2020/1/2", area: "北京",},]} 国内疫情统计数据API接口, area字段为所需数据的地区

http://127.0.0.1/inter_statistics/area GET JSON {data:[{time: "2020/1/1", area: "中国

", current_confirmed: 8929, all_case: 10201,}, {time: "2020/1/2", area: "中国",},]} 国际疫情统计数据API接口, area字段为所需数据的地区

http://127.0.0.1/knowledge_map/map_name GET JSON {map:[{id: 1, name: "新冠肺炎

", connection: [4, 6, 8, ...], connection_name: ["传播途径", "预防措施", "居家隔离", ...], url: " https://www.163.com/dy/article/G7RV6RTD055224PI.html"}...], map_name: " 疫情预防知识图谱"} 疫情相关知识图谱数据API接口, map_name字段为图谱名

表5.3 平台后端相关API数据接口

5.2 平台的搭建和部署

平台依托web, 前端基于vue2.x版本框架开发, 利用vue-cli脚手架快速搭建平台前端项目, 前端UI组件采用IviewUI框架, 前端页面过渡动画采用animate。在css框架中, 前端交互采用jQuery中的ajax方法, 前端页面路由由vue-router管理, 基于node.js的npm包管理工具导入每个项目所需的依赖项和模块。

图5.1 项目前端结构目录

5.3 平台页面路由的配置

为更好地管理页面路由, 平台采用vue-router路由管理器管理路由, 可以使得页面业务逻辑更加清晰, 不同的路由可以由其对应的单一vue文件来实现, 使各个路由模块解耦, 同时也可以实现路由页面间的互相通信, 路由配置代码及路由结构如下:

	1.	export default new Router({
	2.	routes: [{
	3.	path: '/',
	4.	redirect: '/preloading'
	5.	}, {
	6.	path: '/preloading', //预加载页面
	7.	name: 'preloading',
	8.	component: PreLoading
	9.	}, {
	10.	path: '/main', //导航栏主路由
	11.	redirect: '/main/home',
	12.	name: 'main',
	13.	component: Main,
	14.	children: [{
	15.	path: 'home', //主页面
	16.	name: 'home',
	17.	component: Home
	18.	}, {
	19.	path: 'tab1', //趋势挖掘页面
	20.	redirect: 'tab1/chinaCurrent',
	21.	name: 'tab1',
	22.	children: [{
	23.	path: 'chinaCurrent', //趋势挖掘子页面1
	24.	name: 'chinaCurrent',
	25.	component: chinaCurrent
	26.	}, {
	27.	path: 'chinaTrend', //趋势挖掘子页面2
	28.	name: 'chinaTrend',
	29.	component: chinaTrend
	30.	}],

	31.	component: Tab1
	32.	}, {
	33.	path: 'tab2', //时空分析页面
	34.	name: 'tab2',
	35.	component: Tab2,
	36.	}, {
	37.	path: 'tab3', //知识图谱页面
	38.	redirect: 'tab3/map1',
	39.	name: 'tab3',
	40.	children: [{
	41.	path: 'map1', //知识图谱子页面1
	42.	name: 'map1',
	43.	component: Map1
	44.	}, {
	45.	path: 'map2', //知识图谱子页面2
	46.	name: 'map2',
	47.	component: Map2
	48.	}, {
	49.	path: 'map3', //知识图谱子页面3
	50.	name: 'map3',
	51.	component: Map3
	52.	}],
	53.	component: Tab3
	54.	}, {
	55.	path: 'tab4', //新闻谣言页面
	56.	redirect: 'tab4/news',
	57.	name: 'tab4',
	58.	component: Tab4,
	59.	children: [{
	60.	path: 'news', //新闻子页面
	61.	name: 'news',
	62.	component: News
	63.	}, {
	64.	path: 'rumor', //谣言子页面
	65.	name: 'rumor',
	66.	component: Rumor
	67.	}]
	68.	}]
	69.	}]
	70.	});

```

1. export default new Router({
2. routes: [{
3. path: '/',
4. redirect: '/preloading'
5. }, {
6. path: '/preloading', //预加载页面
7. name: 'preloading',
8. component: PreLoading
9. }, {
10. path: '/main', //导航栏主路由
11. redirect: '/main/home',
12. name: 'main',
13. component: Main,
14. children: [{
15. path: 'home', //主页面
16. name: 'home',
17. component: Home
18. }, {
19. path: 'tab1', //趋势挖掘页面
20. redirect: 'tab1/chinaCurrent',
21. name: 'tab1',
22. children: [{
23. path: 'chinaCurrent', //趋势挖掘子页面1
24. name: 'chinaCurrent',

```

```

25.component: chinaCurrent
26.}, {
27.path: 'chinaTrend', //趋势挖掘子页面2
28.name: 'chinaTrend',
29.component: chinaTrend
30.}],
31.component: Tab1
32.}, {
33.path: 'tab2', //时空分析页面
34.name: 'tab2',
35.component: Tab2,
36.}, {
37.path: 'tab3', //知识图谱页面
38.redirect: 'tab3/map1',
39.name: 'tab3',
40.children: [{
41.path: 'map1', //知识图谱子页面1
42.name: 'map1',
43.component: Map1
44.}, {
45.path: 'map2', //知识图谱子页面2
46.name: 'map2',
47.component: Map2
48.}, {
49.path: 'map3', //知识图谱子页面3
50.name: 'map3',
51.component: Map3
52.}],
53.component: Tab3
54.}, {
55.path: 'tab4', //新闻谣言页面
56.redirect: 'tab4/news',
57.name: 'tab4',
58.component: Tab4,
59.children: [{
60.path: 'news', //新闻子页面
61.name: 'news',
62.component: News
63.}, {
64.path: 'rumor', //谣言子页面
65.name: 'rumor',
66.component: Rumor
67.}]
68.}]
69.}]
70.});

```

路由path	重定向redirect	直接父路由	解释
/preloading	无	无	预加载页面
/main	/main/home	无	导航栏主路由
/home	无	/main	首页
/tab1	/tab1/chinaCurrent	/main	趋势挖掘页面
/chinaCurrent	无	/tab1	趋势挖掘子页面1
/chinaTrend	无	/tab1	趋势挖掘子页面2
/tab2	无	/main	时空分析页面
/tab3	/tab3/map1	/main	知识图谱页面
/map1	无	/tab3	知识图谱子页面1
/map2	无	/tab3	知识图谱子页面2
/map3	无	/tab3	知识图谱子页面3
/tab4	/tab4/news	/main	新闻谣言页面
/news	无	/tab4	新闻子页面
/rumor	无	/tab4	谣言子页面

路由path 重定向redirect 直接父路由解释

/preloading 无无预加载页面
/main /main/home 无导航栏主路由
/home 无 /main 首页
/tab1 /tab1/chinaCurrent /main 趋势挖掘页面
/chinaCurrent 无 /tab1 趋势挖掘子页面1
/chinaTrend 无 /tab1 趋势挖掘子页面2
/tab2 无 /main 时空分析页面
/tab3 /tab3/map1 /main 知识图谱页面
/map1 无 /tab3 知识图谱子页面1
/map2 无 /tab3 知识图谱子页面2
/map3 无 /tab3 知识图谱子页面3
/tab4 /tab4/news /main 新闻谣言页面
/news 无 /tab4 新闻子页面
/rumor 无 /tab4 谣言子页面

表5.4 平台页面路由结构

5.4 平台预加载页面的制作

为使用户有流畅的平台体验，特地制作了预加载页面用于提前加载平台所需的各项静态资源，预加载函数实现如下：

	1.	preload() {
	2.	let imgs = [
	3.	//所需预加载的静态资源
	4.	"../static/img/cured.png",
	5.	"../static/img/death.png",
	6.	"../static/img/sure.png",
	7.	"../static/img/time.png",
	8.	"../static/img/city.png",
	9.];
	10.	this.imgLength = imgs.length;
	11.	for (let img of imgs) {
	12.	//创建资源实例
	13.	let image = new Image();
	14.	image.src = img;
	15.	image.onload = () => {
	16.	this.count++;
	17.	this.progress = Math.floor(this.count / this.imgLength * 100);
	18.	}
	19.	}
	20.	}

```
1. preload() {
2. let imgs = [
3. //所需预加载的静态资源
4. "../static/img/cured.png",
5. "../static/img/death.png",
6. "../static/img/sure.png",
7. "../static/img/time.png",
8. "../static/img/city.png",
9. ];
10. this.imgLength = imgs.length;
11. for (let img of imgs) {
12. //创建资源实例
13. let image = new Image();
14. image.src = img;
15. image.onload = () => {
16. this.count++;
17. this.progress = Math.floor(this.count / this.imgLength * 100);
18. }
19. }
20. }
```

预加载动画特效的实现是基于canvas画布，采用js面向对象的设计方式构建并渲染，最终实现效果图如下：

图5.2 平台预加载页面展示图

5.5 平台首页的制作

在结构目录src文件夹下的components文件夹中创建Main.vue和home.vue文件，Main.vue主要是用于构建全局导航栏以及全局footer栏，通过npm管理工具按照要求导入IviewUI框架以及animate.css动画框架，然后在目录src->router->index.js文件

中注册home.vue路由由即首页路由，通过ajax网络请求获取该页面所需所有的数据，最终实现效果如下：

图5.3 平台首页展示图

5.6 平台趋势挖掘页面的制作

趋势挖掘页面中主要是由Echarts地图、柱状图、关系对比图、日历图、折线图等可视化图表构成，同样地，为实现数据图表的动态更新，后端整合出前端对应所需数据并封装为API，前端采用AJAX技术实现数据的交互请求。该页面中主要有两张可视化视图，第一张视图主要由国内疫情地图与各省市区疫情柱状图构成，同时提供了动态演示配置面板包括自定义演示项目、演示时长、起止时间、演示速度和颜色映射等功能。第二张视图主要是由国内疫情基本数据折线图、每日确诊人数、治愈人数以及死亡人数日历图和确诊人数和治愈人数对比关系图构成，效果图如下所示：

图5.5 平台趋势挖掘页面展示图1

图5.6 平台趋势挖掘页面展示图2

5.7 平台时空分析页面的制作

本页面主要是利用echarts与百度地图实现的一个世界实时疫情数据可视化，包括全球疫情热力映射图、发展趋势动态图，展示了全球各洲以及大部分国家每日的现存确诊人数、治愈率、死亡率，典型国家的疫情态势，国内外疫情发展图等，同时融入了数据交互展示，用户可以通过交互获取到详细的可视化趋势图表。

图5.4 时空分析页面效果图

5.8 平台知识图谱页面的制作

为使广大用户进一步了解疫情相关的知识，通过相关官方网页以及相关权威书籍提供的数据，本页面构建了与疫情相关的知识图谱利用Echarts进行了交互式可视化，主要包括三大知识图谱，分别是疫情基本知识图谱、疫情预防知识图谱以及疫情疫苗知识图谱。疫情基本知识图谱主要是构建了以新冠肺炎为中心词的相关知识网络，例如与其相关的药物、传播途径、上下位等等，与此同时，为加强用户对节点的理解，平台搜集了相关节点解释的网址，通过点击图谱上的相关节点，平台会跳转到对应节点的解释网站。疫情预防知识图谱主要是以疫情预防为中心构建的环状网络关系图，结合了聚焦选择过滤、鼠标悬浮提示等交互手段，通过点击相关节点，页面会展示节点的详细内容以供用户学习。疫情疫苗知识图谱是以新冠疫苗为中心构建的散点关系图，为用户科普疫情疫苗的相关知识网络，通过弹出窗口和悬浮提示等方式来展示节点的详细信息。

图5.7 疫情疫苗知识图谱效果图1

图5.8 疫情疫苗知识图谱效果图2

图5.9 基本知识图谱效果图

图5.10 预防知识图谱效果图

5.9 平台新闻谣言页面的制作

疫情相关的新闻以及疫情相关谣言也是大多数用户所关心的问题，本页面主要是利用swiper轮播组件对疫情相关新闻以及谣言进行了可视化，方便用户关注疫情的新闻以及谣言动态，新闻数据来源于腾讯新闻，请求接口为

<https://cdn.mdeer.com/contentdtos.js?callback=callbackcontentdtos>，谣言数据来源于丁香园，请求接口为

<https://file1.dxycdn.com/2020/0130/454/3393874921745912507-115.json>。

构建swiper轮播效果代码如下：

	1.	var mySwiper = new Swiper(".swiper-container", {
	2.	slidesPerView: 3, //默认是显示4个
	3.	initialSlide: 0, //默认从第几个显示
	4.	spaceBetween: 0, //间距
	5.	speed: 1000, //速度
	6.	prevButton: ".swiper-button-prev", //左右按钮
	7.	prevButton: ".swiper-button-prev", //左右按钮
	8.	nextButton: ".swiper-button-next"
	9.	});

```
1. var mySwiper = new Swiper(".swiper-container", {
2. slidesPerView: 3, //默认是显示4个
3. initialSlide: 0, //默认从第几个显示
4. spaceBetween: 0, //间距
5. speed: 1000, //速度
6. prevButton: ".swiper-button-prev", //左右按钮
7. prevButton: ".swiper-button-prev", //左右按钮
8. nextButton: ".swiper-button-next"
9. });
```

使用ajax请求疫情新闻以及疫情谣言数据代码如下：

	1.	//新闻API数据请求采用jsonp方式
	2.	var that = this;
	3.	var news_url = "https://cdn.mdeer.com/contentdtos.js?callback=callbackcontentdtos&t=";
	4.	\$.getScript(news_url+(new Date()));
	5.	window.callbackcontentdtos = function(res) {
	6.	that.news_data = res;

	7.	}
	8.	//谣言API数据请求
	9.	var rumor_url = "https://file1.dxycdn.com/2020/0130/454/3393874921745912507-115.json?t=";
	10.	\$.ajax({
	11.	url:rumor_url + new Date(),
	12.	type:"get",
	13.	dataType:"json",
	14.	success:function(data) {
	15.	this.rumor_data = data;
	16.	},
	17.	error:function(err) {
	18.	console.log(err);
	19.	}
	20.	})

```

1. //新闻API数据请求采用jsonp方式
2. var that = this;
3. var news_url = "https://cdn.mdeer.com/contentdtos.js?callback=callbackcontentdtos&t=";
4. $.getScript(news_url+(new Date()));
5. window.callbackcontentdtos = function(res) {
6. that.news_data = res;
7. }
8. //谣言API数据请求
9. var rumor_url = "https://file1.dxycdn.com/2020/0130/454/3393874921745912507-115.json?t=";
10. $.ajax({
11.url:rumor_url + new Date(),
12.type:"get",
13.dataType:"json",
14.success:function(data) {
15.this.rumor_data = data;
16.},
17.error:function(err) {
18.console.log(err);
19.}
20. })

```

新闻API接口返回数据格式如下:

```

res:[{abs:"<p>*月*日*时, 31个省, 自治区以及直辖市.....</p>",articleAbstract:"*月*日*时, 31个省, 自治区以及直辖市.....",authorId:1409,authorName:"央视新闻",contentId:210071,gmtCreate:1618792049000,previewUrl:"",publishTime:1618792299000,pv:0,realPublishTime:"2021年04月19日 08:27",title:"31省区市无新增本土确诊 31省区市新增11例确诊均为境外输入",url:"https://m.weibo.cn/status/4627611617857469"},.....]

```

疫情新闻信息可视化效果图如下:

图5.11 疫情新闻信息可视化效果图

谣言API接口返回数据格式如下:

```

data:{code:"success",data:[{body:"疫苗瓶的生产要求确实很高, 按照国际标准...",id:197,mainSummary:"2017 年前, 我国还无法实现疫苗瓶的量产, 但如今已成功突破这项技术瓶颈。",rumorType:0, score:210, sourceUrl:"",summary:"",title:"中国企业造得出疫苗却造不出疫苗瓶?"},...],...}

```

谣言信息可视化效果图如下:

图5.12 疫情谣言信息可视化效果图

结语

本文围绕新冠肺炎交互数据可视化设计相关内容开展工作, 在平台项目中既对开发者有技术能力要求, 同时也有一定的艺术设计要求, 最终的平台可以为广大用户进一步了解新冠肺炎提供辅助性的分析与帮助。从项目开发到结束, 最重要的部分是相关准备工作, 包括UI、UE的设计、交互式设计、数据可视化图表的设计、数据的搜集清洗以及分析、项目模块的确定, 当这些部分确定完成, 后续的内容自然是水到渠成。整个平台既设计出了丰富的可视化图表以及交互手段, 同时也设计出了实时疫情数据展示、疫情新闻谣言等实用性功能, 但由于能力原因, 平台仍然存在着一些不足, 例如知识图谱的构建, 由于本文主要工作是交互式可视化设计, 所以在知识图谱构建中采用了手动构建的方式。也有一些硬性条件的限制, 在数据趋势挖掘页面中只考虑了2020年全年的数据, 同时也存在着上文提到的交互设计的一些不足。整体而言, 平台的综合性较强, 涉及了前端、后端、艺术和技术, 因此对我个人综合能力也有较大的提升, 在后续的工作中会继续添加新的内容, 不忘构建平台的初心!

说明:

1. 总文字复制比: 被检测论文总重合字数在总字数中所占的比例

2. 去除引用文献复制比：去除系统识别为引用的文献后，计算出来的重合字数在总字数中所占的比例
3. 去除本人文献复制比：去除作者本人文献后，计算出来的重合字数在总字数中所占的比例
4. 单篇最大文字复制比：被检测文献与所有相似文献比对后，重合字数占总字数的比例最大的那一篇文献的文字复制比
5. 指标是由系统根据《学术论文不端行为的界定标准》自动生成的
6. 红色文字表示文字复制部分；绿色文字表示引用部分；棕灰色文字表示作者本人文献部分
7. 本报告单仅对您所选择比对资源范围内检测结果负责



✉ amlc@cnki.net

🌐 <http://check.cnki.net/>

👤 <http://e.weibo.com/u/3194559873/>

CNKI科研诚信管理系统研究中心