

文章编号:1674-4578(2024)03-0084-04

基于 Docker 的分布式疫情数据可视化平台管理系统*

巩如悦¹, 巩熠², 李泽沁¹, 吴宏晶¹, 刘建文¹

(1. 广东东软学院, 广东 佛山 528000; 2. 青岛科技大学, 山东 青岛 266061)

摘要:新冠疫情数据可视化平台以可视化的方式为人们实时展现疫情动态,给使用者提供便捷,但随着项目体系的复杂化、模块层次增多,加大了运维管理和开发成本。因此,提出基于 Docker 的分布式疫情数据可视化平台管理系统,将网络爬虫模块、数据清洗模块、可视化模块存放在不同的容器中,各容器相互独立并发运行。系统设置生命周期管理模块、监控与警告模块、访问控制模块用于管理容器,以提高平台的健壮性,更好的提供可视化服务。

关键词: Docker; 容器管理; 分布式; 疫情数据; 可视化平台管理系统

中图分类号: TP311.13 **文献标识码:** A

Docker-based Distributed Epidemic Data Visualization Platform Management System

Gong Ruyue¹, Gong Yi², Li Zeqin¹, Wu Hongjing¹, Liu Jianwen¹

(1. Nersoft Institute Guangdong, Fushan Guangdong 528000, China;

2. Qingdao University of Science and Technology, Qingdao Shandong 266061, China)

Abstract: The novel coronavirus data visualization platform displays real-time epidemic dynamics in a visual way and provides convenience to users. However, with the complexity of the project system and the increase of module levels, operation and maintenance management and development costs are increased. Therefore, a distributed epidemic data visualization platform management system based on Docker is proposed in this paper. The web crawler module, data cleaning module and visualization module are stored in different containers, and each container runs independently and concurrently. The system sets life cycle management module, monitoring and warning module, and access control module for container management so as to improve the robustness of the platform and provide better visual services.

Key words: Docker; container management; distributed; epidemic data; visualization platform management system

0 引言

传统的软件开发过程中,开发人员在项目开发、测试和维护各个环节都投入诸多精力,同时,测试人员和维护人员在测试和运维部署过程中需要解决各种版本软件包管理、环境问题,存在重复作业现象。Docker 是基于 Linux 容器创建的一个容器引擎^[1],解决容器快速构建、部署和分享问题,隔离性强、高移植性,可实现轻量级的虚拟化^[2]。自开源以来,以快速应用部署的魅力成功赢得各界的高度关注和

广泛应用。前期开发新型冠状病毒肺炎疫情数据可视化智慧平台也是采用了传统的软件开发流程,为了提高系统的可用性和易用性,本文基于 Docker 容器设计并实现了一套成本低、耗时少、模块化的分布式疫情数据可视化平台管理系统^[3]。

1 管理系统研究

1.1 系统结构框架设计

系统结构框架包括容器应用和容器管理。容器应用模块负责疫情可视化数据的采集、处理和可视

收稿日期:2023-05-15 修回日期:2024-05-01

基金项目:广东大学生科技创新培育专项资金资助项目(Pdjh2022b0607)

作者简介:巩如悦(1990-),女,山东滨州人,讲师,硕士研究生,研究方向:智能信息系统。

化等业务;容器管理模块负责对各个容器进行管理,确保系统高效有序运行。系统体系结构如图 1 所示。系统总体流程如下:

- 1) 业务层:创建 5 个容器,分别用于定时爬取疫情数据、疫情数据存储、数据处理、数据分析、数据可视化。
- 2) 客户端:用户通过客户端可以查看可视化的数据,方便了解疫情的实时动态。
- 3) 服务器端:主要负责对容器的管理。

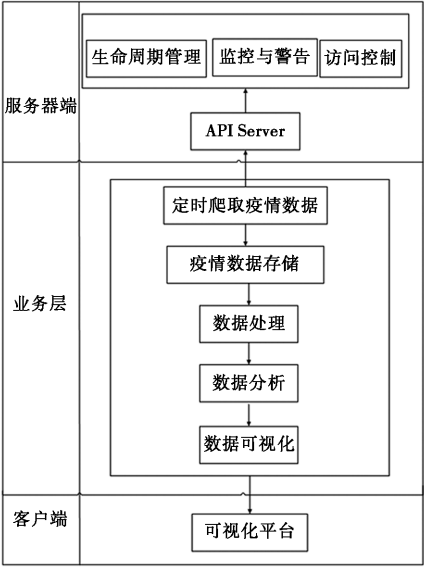


图 1 系统体系结构图

1.2 容器应用模块

应用模块通过各容器并发运行提供可视化服务,容器名称和用途如表 1 所示。

表 1 容器名称及用途

名称	用途
C1	主要负责定时爬取疫情数据,并将数据存储在 C2 中
C2	用于存储疫情数据,包括原始数据、处理后的数据等
C3	主要对抓取的数据进行分词、清洗工作,并将处理好的数据放入 C2 中
C4	对处理好的数据进行分析提取,获取需要可视化的数据存储在 C2 中
C5	主要通过热力图、柱形图、折线图等对数据进行可视化展示

1.3 容器管理模块

系统主要通过 API Server 模块和三个容器管理模块对各容器进行管理^[4]。

- 1) API Server 模块:接收用户请求执行相应模块的功能,包括容器状态查询、生命周期管理、监控指标与告警阈值设置、容器访问规则设置等。
- 2) 生命周期管理模块:根据用户请求查询容器状态或执行相应的生命周期管理操作,包括部署、删

除、终止、在容器内执行特定命令等。流程图如图 2 所示。

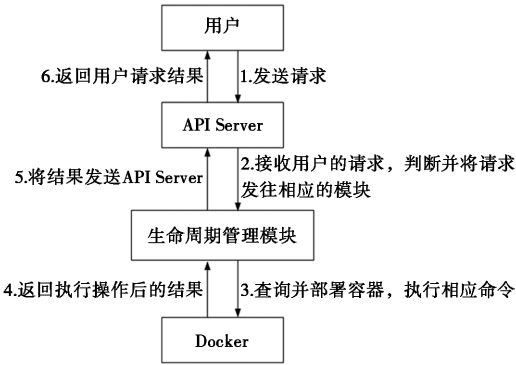


图 2 容器生命周期管理流程图

2 管理系统设计与实现

2.1 系统环境与架构

开发本系统需要安装虚拟机 VMware,操作系统 Windows10,虚拟机镜像 CentOS 7,终端交互软件 MobaXterm,利用 MobaXterm 远程登录实现对各容器的操作。开发语言 python,使用开发软件为 PyCharm 3.6,利用 flask 搭建服务器,通过 PostMan 返回应答结果。

2.2 系统设计

可视化平台管理系统分为客户端和服务端。客户端主要负责可视化疫情数据,该部分已在可视化智慧平台实现,本文不在赘述。服务器端主要实现对容器的管理,包括容器生命周期管理模块、监控与告警模块和访问控制模块等。

2.3 系统功能实现

2.3.1 生命周期管理模块

1) 容器相关函数

对容器的配置需要导入 docker 模块,同时创建名为 DockerHelper 的类。在初始化函数中,根据指定的 ip 地址和端口号创建客户端,连接远程的 Docker 程序。需要定义容器相关函数,包括获取容器信息、某项属性值、判断容器是否存在、容器管理、获取列表镜像、判断镜像是否存在及其获取镜像相关信息。其中,容器管理方法有 start(), stop(), restart() 和 remove()。

2) 主机相关函数

导入模块 SSHClient、AutoAddPolicy 和 os,在初始化函数中,创建 ssh 连接和 sftp 连接。需要定义

的函数包括:获取主机的信息、对服务器进行相关操作、判断本地文件状态、将本地(windows10)目录上的文件传到某个容器的指定目录等。

2.3.2 API Server 模块

搭建网页服务器,通过“IP 地址 + 端口 + 某些字符串”方式访问特定的网页,实现 API Server 模块的构建。需要导入外部模块 Flask、request 和生命周期管理模块 DockerHelper、NodeHelper,并实例化生命周期管理模块。采用图 3 的方式来接收用户发过来的特定请求。

```
@server.route("/")
def get_root():
    return "欢迎来到容器管理平台"
```

图 3 接收用户的特定请求图

2.3.3 监控与告警模块

1) 监控器相关函数

首先需要创建 __init__(self) 函数,该函数可初始化一个存放所有监控器的字典、初始化一个存放所有监控器的队列的字典、初始化一个邮件助手。除此之外,还需创建监控器操作相关函数,包括:创建监控器、查询正在运行的监控器、判断某个监控器

是否仍在运行、监控器执行函数、删除所有监控器函数等。

2) 邮件发送助手相关

a) def __init__(self):

创建内部的变量

self.mail_server = "smtp.qq.com",。

self.qq_user = "容器平台的 QQ 邮箱"

self.qq_password = "容器平台的 QQ 邮箱凭证"

self.receiver = ["用户的 QQ 邮箱"]

b) def send(self, content)

功能:邮件发送函数

2.3.4 访问控制模块

该模块主要是对 docker 网络进行相关操作,需要定义 docker 网络操作函数,包括:获取当前的所有网络、判断网络是否存在、创建某个网络、查询连接到某个网络的所有容器、判断某个容器是否连接到某个网络、将某个容器连接/断开到某个网络上、判断两个容器是否互通等。

2.4 系统功能测试

利用 Postman 工具对各模块进行测试,各模块均正常工作。同时,服务器端设有监控界面,实时动态显示容器相关信息,包括当前带宽、内存、CPU 的使用率,如图 4 所示。



图 4 容器实时信息图

3 总结

将新型冠状病毒肺炎疫情数据可视化智慧平台

部署到 Docker 容器环境中,将各模块封装在不同的容器中,通过生命周期管理、监控与告警、访问控制

3 个模块实现对各个容器的监测和控制,其运行速度更佳,充分体现了容器轻量级、可移植性强、兼容性好等优势。

参考文献

- [1] 周鹏. Docker 集群管理系统的研究与实现 [D]. 南京:东南大学,2018.
- [2] 郑远. 基于 Docker 的服务器虚拟化实践 [J].

福建电脑,2019(4):108-110.

- [3] 郝慧杰. 基于 Docker 的 VNF 生命周期管理系统的设计与实现 [D]. 南京:南京邮电大学,2020.
- [4] 郝慧杰,肖建,张粮,等. VNF 生命周期管理系统设计与实现 [J]. 计算机技术与发展,2020,30(7):12-16.

(上接第 61 页)

这是因为 Seq2Seq 模型能够处理较长序列间的依赖关系,并且具有更强的记忆能力。相比之下,LSTM

模型可能难以处理相对较长的信道状态变化,容易出现“梯度消失”等问题,导致预测效果变差。

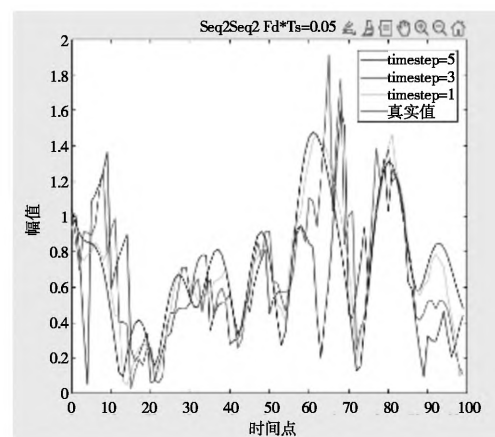
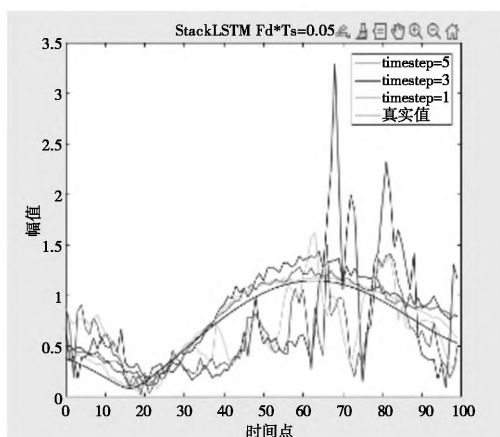


图 6 $Fd * Ts = 0.05$ 下两种预测模型的预测结果

5 结论

针对无线时变信道,传统的正弦波叠加法和成型滤波法用于信道仿真,存在精度低、容错性差、不宜进行大数据量计算等缺点。而神经网络具有极强的学习能力,能在复杂情况下与各非线性系统相逼近。本文利用传统方法产生的大量多普勒功率谱样本数据,训练 BPNN 对时变信道进行学习预测,训练好的网络结构仿真来的结果频域符合多普勒功率谱的特征,呈 U 型谱。时域自相关函数满足第一类贝塞尔函数的特征,很好地符合 Jakes 模型的时频域条件。与传统的正弦波叠加法、成型滤波法仿真结果对比,神经网络的仿真效果更接近理论值,说明其效果优于传统方法,同时具有很好的容错性。

参考文献

- [1] 王智宁,江虹,彭满祺. 改进 GA-BP 神经网络的无线信道预测方法 [J]. 数据采集与处理,2022,37(6):1268-1279. DOI: 10.16337/j.1004-9037.2022.06.008.
- [2] 董子奇. 基于混合神经网络的无线电信号识

别 [D]. 西安:西京学院,2022.

- [3] 邱杰凡,贾逸哲,华宗汉,等. 基于机器学习的跨平台缓存划分方法研究 [J]. 计算机学报,2023(10):2097-2116.
- [4] 李翠然,杨茜,谢健骊,等. 基于 SWIPT 的能量收集 WSN 吞吐量性能分析及优化 [J]. 西南交通大学学报,2024. DOI: 10.3969/j.issn.0258-2724.
- [5] 王晶晶. UFMC 系统的干扰分析和接收机重构算法研究 [D]. 杭州:浙江工商大学,2022.
- [6] 李聪. 侧信道攻击方法综述 [J]. 现代信息科技,2022,6(14):82-85. DOI: 10.19850/j.cnki.2096-4706.2022.014.020.
- [7] 李宝安,刘翔,王宗辉,等. 融合关键信息的科技文献创新点生成方法 [J]. 计算机工程与设计,2023,44(4):1267-1273.
- [8] 张峰. 基于预训练语言模型的无监督文本简化研究与应用 [D]. 扬州:扬州大学,2022.