

周子杰

电话: +86 18930851256 | 微信: zhouzijie1256 | 邮箱: zhou.zijie@northeastern.edu
个人网站: <https://zijiezhou.me> | **GitHub**: github.com/Zijie000



教育背景

东北大学(NEU)

Software Engineering System, M.S.

Boston, U.S

May 2025

亚利桑那州立大学(ASU)

Computer Science, B.S.

Tempe, U.S

May 2022

技术栈

语言: Python, Java, Scala, Go, JavaScript, SQL, Lisp

框架: Spring/Spring boot, Gin, Spark

基础设施即代码: Terraform, Packer

持续集成部署工具: GitHub Actions, Jenkins

云技术&容器技术: Amazon Web Services(AWS), Google Cloud Platform(GCP), Kubernetes, Docker, Kafka

学术项目

实时社交媒体关键词情感分析系统(Spark)

地点: Boston

2025 一月 ~ 2025 五月

设计并实现了一个支持实时分析的分布式社交媒体情感分析系统, 支持用户指定关键词与时间区间, 针对 X(原 Twitter)、Reddit 等平台的内容进行舆情分析。系统具备良好的可扩展性、容错性与流处理能力。

- 在数据采集层, 基于 **Akka Actor** 构建并行采集器, 通过社交媒体 API 实时抓取符合关键词和时间范围的内容, 并作为 **Kafka** 发布者将数据发布至 **Kafka Broker**
- 在数据处理层, 使用 **Spark Streaming** 对 **Kafka** 流数据进行规范化处理, 接入 **Hugging Face NLP** 模型进行情感分析, 并使用 **Spark MLlib** 的 **TF-IDF** 算法提取相关态度词并进行分类
- 在存储层, 采用 **PostgreSQL** 数据库存储处理后的结构化信息, 设计特定 **Schema** 支持关键词维度的高效查询, 并定期清理历史数据
- 在可视化层, 展示关键词对应推文的平均情感趋势变化折线图、词云(代表性情绪词)、以及推文所属的类别与对应情绪词分布
- 整体系统使用 **Scala** 编写, 严格遵循函数式编程范式, 使用不可变数据结构与 **Future** 进行并发控制, 并采用 **Actor** 模型管理状态, 确保并行与异步场景下的安全性

云计算维护/云原生

地点: Boston

2024 九月 ~ 2024 十二月

设计更新并维护了一个用户信息系统, 使用了 **Go** 和 **Gin**, 同时还负责管理他们的云基础设施。

- 使用 **Golang**、**Gin** 和 **GORM(ORM)** 开发和维护了一个 **RESTful API** 用户管理系统, 提供了高效且可扩展的解决方案
- 使用 **Terraform** 定义 **VPC**, 其中包含多个私有和公共子网。**RDS** 数据库位于私有子网中, 禁止直接连接互联网, 从而保证数据库的安全性。公共子网中部署了核心应用程序。
- 应用程序的负载均衡器(**ELB**)和自动伸缩集群(**Autoscaling**)配置在公共子网中, 其域名通过 **Route 53** 配置 A 记录, 使用 **TLS/SSL** 实现 **HTTPS** 加密, 确保通信的安全性
- 为用户图片存储配置了 **S3** 存储桶, 并通过 **AWS Lambda** 部署了邮件验证程序, 优化用户体验和系统交互流程。
- 使用 **Packer** 编写 **HCL** 文件来构建 **EC2** 镜像。该镜像包含预装的 **Golang Gin RESTful API** 应用程序, 确保高效的应用交付
- Golang** 应用程序的源代码和 **Packer** 文件都托管在 **GitHub** 仓库中, 通过 **GitHub Actions** 工作流实现 **CI/CD** 流程。每个代码变更需要通过集成测试, 过后才允许合并, 并触发 **Packer** 构建过程, 将 **EC2** 镜像上传到 **AWS**

实习经历

尼尔森 NIQ | 电商平台数据采集自动化 RPA

地点: 上海

2023 五月 ~ 2023 八月

开发了一种机器人流程自动化 (**RPA**) 解决方案, 用于自动化从淘宝和饿了么平台的大规模数据采集。该 RPA 系统能够持续获取这些零售平台的实时商品数据, 确保业务分析部门获得最新、准确的信息

- 使用 **Java** 结合 **Selenium** 和 **Appium** 设计并实现了 **RPA**, 支持从网页和移动端界面自动提取数据
- 通过 **Appium** 将 **RPA** 集成到外部 **Android** 设备中, 使系统能够与移动平台的图形用户界面 (**GUI**) 进行交互
- 优化了 **RPA** 工作流程, 高效处理大批量的商品数据, 最大限度地减少停机时间, 确保系统持续运行

奖项

美国数学建模大赛 成功参与奖 (2021 年)

New American University Scholar 奖学金 (2017 年-2021 年)