

盛泽

(+1) 267-319-4543 · osanzas2002@gmail.com · wechat:rsldlk · <https://github.com/OwenSanzas>

教育背景

德州农工大学 (USNews 排名 23), 计算机科学 博士	2024.9 - 2028.6
宾夕法尼亚大学 (QS 排名 6), 计算机科学 & 系统工程, 硕士	2021.9 - 2024.6
计算机专业排名前 5%, 系统工程专业排名 1, 全满分毕业	
加州大学戴维斯分校 (QS 排名 59), 计算机科学与技术, 暑期交换生项目 排名 1, Academic Perfection (学术完美)	2018.7 - 2018.9
电子科技大学, 电子信息工程, 学士	2016.9 - 2021.6
专业排名 1/254, 国家奖学金, 学业奖学金, 优秀学生奖学金, 发表两篇 EI 会议论文, IEEE 学生会员	

技术能力

- 编程语言: Java, Python, Go, JavaScript (ECMAScript, Node.js), HTML/CSS, SQL, C/C++, Shell
- 操作系统, 数据库与工程构建: Linux/macOS, Echo/Go, Spring/Java, Nodejs/Javascript, PostgreSQL/MongoDB

工作/实习经历

德州农工大学人工智能与系统安全实验室, 助理研究员 2024.7-至今

- 探索当今大语言模型在软件安全, 系统安全的应用, 实现基于 GPT-4o 的全自动软件维护系统: 该项目在美国国防部举办的 AIxCC 比赛中获得了**全球第五名以及两百万美元奖金**。
- 独立负责大模型在 Linux 内核的安全性分析, 包括建立基于 **RAG** 的 CVE 历史数据库, 建立数据分析层和代理层来对目标代码库进行区分和分流, 设计数据提取和简化流程提高提示词的信息密度。

辉固美国科技有限公司 | **Fugro**, 软件工程师 (休斯顿, 德克萨斯州) 2023.6 - 2023.9

- 使用 C#, 在公司自研的基于 **ASP.NET** 软件框架上, 开发了海底机器人控制逻辑, 实现了障碍规避、深度维护和海底测绘功能, 并优化了其多线程数据管道, 将处理时间缩短到 50 毫秒以内。
- 将 NSubstitute 和 xUnit 优化为 NMock 和 NUnit, 并重构了测试用例。利用依赖注入来解耦组件, 通过 NuGet 管理依赖项, 并解决了 .NET Framework 和 .NET Core 之间的不兼容性, 确保了可维护性。
- 维护了公司的标准业务库, 包括优化线程安全类, 从而提高 WPF 的安全性。将部分组件封装为 Interface, 实现组件的高效复用性。
- 采用 MVC 和 SOA 范式重构重复的算法模块到服务层, 减少了代码冗余和耦合, 提高了可扩展性。
- 基于测试驱动开发 (TDD), 利用 TestFlow 平台在模拟环境中设计了机器人功能的测试场景。通过单元测试提高了系统的可靠性并减少了潜在风险。

数据林科技有限公司 | **DataLynn**, 软件工程师 (纽约, 纽约州) 2023.4 - 2024.1

- 使用基于 Node.js 的 REST API 设计并实现了一个 AI 教育平台的后端, 管理存储在 MongoDB 中的数据。实现了基于 JWT 和 OAuth 的登录方法和基于 Stripe 的支付中间件, 用于处理用户订阅和支付。
- 使用 SGLang 和 GPT 在 Azure OpenAI Studio 上开发了一个 AI 驱动的模拟面试服务, 利用 Tavily 和 prompt chaining 动态生成代码面试问题并提供评分反馈。并实现了一个由 Kubernetes 上的 Docker 容器管理的多用户在线代码编译和执行组件。
- 使用 TypeScript 和 React 创建了一个 Chrome 插件, 在 Colab Notebooks 中嵌入了聊天框, 实现了用户代码和 Markdown 的实时读取, 提高了代码生成的效率和准确性。链接在此。
- 通过 FastAPI 和 Redis 优化了 API 响应时间至 300 毫秒, 使用会话令牌隔离了模拟面试用户并保存历史数据, 并通过数据库快照实现了数据持久化。

狮子金融 | **Lions Financial**, 软件工程师 (纽约, 纽约州) 2022.6 - 2022.9

- 使用 Java Spring Boot 和 MyBatis 构建了一个基于微服务架构的内容管理系统, 利用 PostgreSQL 作为数据库, 并结合 AWS S3 进行对象存储, 为库存管理和销售预测提供了一个集中的平台。
- 使用 Gradle 管理分布式代码库中的依赖关系, 实现了组件的平滑集成和确保了不同环境的兼容性。
- 协助开发了一个协同过滤推荐算法, 优化了购买建议, 提高了转化率和客户满意度。
- 使用 Log4j 进行日志记录, 使用 Postman 进行 API 测试, 使用 JUnit 和 Mockito 进行后端单元测试, 使用 Cypress 进行前端测试, 并使用 ngrok 小规模部署进行端到端测试, 确保应用程序在生产部署前的可靠性和稳定性。
- 使用 Jenkins on Kubernetes 构建了自动化 CI/CD 管道, 并通过 Groovy 和 Bash 在 GCP App Engine 上实现了可扩展的部署, 确保了持续交付的平稳进行, 并降低停机时间。