

教育经历

西北工业大学, 西安, 陕西  
计算机科学与技术专业, 本科

2014/09 - 2018/07

工作经历

北京阿丘机器人科技有限公司, 北京, 中国  
(深度学习算法应用系统研发) 系统工程师

2018/07 - 现在

- 负责项目系统及产品的需求拆解、功能定义、技术选型、架构设计、模块拆解、开发计划制定;
- 负责前后端开发工程师的开发工作计划安排、人力资源调控、关键时间节点把控;
- 负责系统核心架构和核心模块、功能的开发实现;
- 负责组内人员的工作规划、任务安排、日常技术培训等。

(图像处理算法研发及优化) 图像算法工程师

- 负责基于深度学习的图像处理算法的调研、实验及实现和持续优化效果;
- 负责内部深度学习框架的功能开发、优化及维护;
- 负责负责算法应用框架软件的架构设计、开发规划, 以及带领团队进行开发和维护。

项目经历

算法应用框架软件开发  
系统架构师

2021/09 - 至今

- 工作内容:
- 通过总结公司检测产品的在各个项目中的具体落地使用场景及使用方法, 以及调研竞品软件等方式, 总结和梳理算法应用框架软件的核心需求;
  - 基于对各个场景需求的总结, 进行算法应用框架软件的架构设计和模块设计, 以及制定算法应用框架软件的开发计划;
  - 制定框架软件的开发流程、编码规范及相关技术调研和培训;
  - 复制应用算法框架的核心架构开发;
  - 组织团队进行开发, 负责日常团队日常开发工作, 包括需求分析、设计方案评审、Code Review 等;

- 工作成果:
- 梳理和总结了公司原有产品在各个项目中的落地场景和流程, 完成算法应用框架软件的架构设计及各个模块的概要设计;
  - 完成算法应用框架核心功能开发, 为公司其他各产品线的开发提供统一的底层算法应用框架, 为公司端、边、云产品的协同打下了基础;
  - 基于公司原有开发流程及编码规范薄弱的问题, 在本项目中引入 Bazel 进行 C++ 构建规则管理、引入 Clang-tidy 结合 CI 流程进行静态代码检查, 制定测试规范, 借助一系列工具帮助, 简化和规范开发流程, 严控代码质量; 在开发过程中对组内成员进行培训, 提高人员素质, 迭代开发流程和开发规范。
  - 目前完成了初版算法应用框架的开发, 通过提供完整的计算图流程分析、调度、关联数据管理等功能, 支撑了公司核心产品的重大功能升级需求;
  - 通过良好的问题定义和产品抽象, 为公司不同产品线之间的写作提供标准的基础平台, 大幅简化了公司新产品的研发成本, 提高了不同产品线和部门之间的协作效率;
  - 通过在该框架中规范在算法应用流程的表示, 为整体算法流程的优化创造了空间和平台, 使之前一些不太好实现的算法优化方法在该框架下能够比较方便的进行优化;
  - 在项目中通过引入多项新技术, 如利用 jinja 和 python 的反射机制自动生成 C++ 代码, 以及使用 Clang 分析 C++ AST 自动为 C++ 程序生成 C、C#、Python 等跨语言接口等技术, 在研发资源有限的情况下, 提高了项目的开发效率并有效保障了项目的质量。

某客户工厂面板生产线自动缺陷检测系统 (ADC)  
项目研发负责人

2020/12 - 2021/08

- 工作内容:
- 参与调研和理解客户需求, 负责根据客户需求设计检测系统整体系统架构, 进行技术调研和选型, 以及制定研发计划;
  - 负责与客户项目负责人及技术组接洽, 对接系统各种接口协议, 研发进度等事项;
  - 负责设计 ADC 系统整体系统架构及算法方案制定, 根据项目特点及资源情况, 系统方面采用 Django、MySQL 及 TibcoRV 等工具开发后台系统用于管理客户的工程数据以及进行训练、推理任务的调度和管; 算法方面采用 Faster-RCNN 作为基础检测算法, 实现客户对面板缺陷进行检测并分类的需求;
  - 带领团队进行系统开发, 管控系统部分研发进度, 参与算法方案修改、算法迭代及优化;
  - 调研、改进算法方案及参与部分实验, 解决客户关注的异常类 (Out of Distribution) 缺陷检出的问题;

- 工作成果:
- 先后与团队一起完成 ADC 系统两个阶段的开发和上线; 满足客户的功能需求和检测指标要求;
  - 系统已在客户工厂上线运行, 支撑客户工厂 30 多个生产环节每日超过 200 万高清图像的缺陷识别需求。

深度学习框架支持 TensorRT 推理  
算法工程师

2020/08 - 2020/09

- 工作内容:
- 在深度学习框架中添加深度 TensorRT 库支持, 允许推理时根据用户参数设置选择使用 TensorRT 进行推理

- 为 TensorRT 不支持的层实现自动插件派发机制，对于 TensorRT 不支持的层，自动通过插件适配器调用 Caffe 中

的层实现进行推理。 工作成果：

- 算法框架支持所有模型自动转换到 TensorRT 模型并使用 TensorRT 进行推理；提高所有模型推理速度 20% 30%；

深度学习框架添加图编译功能支持

2020/03 - 2020/06

算法工程师

工作内容：

- 调研和学习其他深度学习框架的静态图编译机制及实现方法，基于原 Caffe 的框架接口，设计图编译的实现机制；
- 在 Caffe 框架中引入图编译机制，通过在执行训练和推理前静态分析模型结构、模型中的参数、梯度和及激活值之间的依赖关系，实现智能分配显存和内存空间，通过合理复用不同 Tensor 的存储空间；
- 在图编译阶段引入自动评估算子性能，根据不同算子实现在不同参数上的实际性能表现智能选择合适的算子实现，以提高训练效率；
- 在 Caffe 中基于 NCCL 及 MPI 实现多卡和多机并行训练功能支持；
- 在推理阶段添加算子融合的编译过程，自动合并如 Convolution、BatchNorm、Scale、Eltwise、ReLU 等算子，以提高推理效率。

工作成果：

- 通过引入静态编译机制，使内部所有网络模型训练和推理占用显存均降低 30% 左右；
- 通过引入算子选择功能，优化部分模型训练速度 20% 以上；
- 推理阶段通过自动融合算子，提高各个模块推理速度 10% 左右；

重构算法模块实现流程及应用接口

2019/07 - 2019/11

算法工程师

工作内容：

- 梳理原产品中各个模块的实现方式，总结统一的数据集读取、数据集预处理、模型参数设置、加载网络模型、创建和设置 DataLoader、启动训练、保存模型等算法实现的整体流程；总结和归纳统一的算法实现流程；
- 合理拆解和组合数据集管理、数据集预处理、网络模型参数设置、网络初始化及训练、模型保存等各个流程和模块的关系，设计统一的算法实现框架；
- 按照新的算法实现框架、重构原来的各个算法模块，规范各个算法模块的实现结构，同时优化各个模块的实现细节；
- 开发算法模块串联执行框架，使得无需对各个算法模块的逻辑做任何改造的情况下，即可将前一模块的检测结果区域输入到后一算法模块的做进一步的检测、并自动将检测结果映射回原图坐标系。
- 整理和规范该算法各个算法模块的对外接口，制定和实现统一的标注格式协议、参数接口协议，统一不同算法模块的调用逻辑，提高接口的易用性。

工作成果：

- 优化和统一了原有的各个算法模块的实现逻辑，减少了后续修改和维护的成本；
- 通过提供合理的框架支持，使得在该框架下实现新算法模块的时间由一个月减少到一周；
- 通过规范各个模块的算法实现以及提供自动将不同算法模块串联执行的机制，简化我们的 SDK 接口，降低了用户基于我们的 SDK 进行开发的使用成本，提高了 SDK 的易用性；
- 由于此项工作以及其他贡献，荣获公司年度优秀员工奖。

模型训练及推理速度优化

2019/01 - 2019/04

算法工程师

工作内容：

- 梳理公司原来的各个深度学习算法模块的整体处理流程，分析训练和推理执行的性能瓶颈及原因；
- 综合运用调整算法网络结构、对比 cuDNN 算子实现和自行开发的 CUDA 算子的自行速度并选择合适的执行策略、优化部分网络层的 CUDA 算子实现等方式，在保持原有算法模块指标的前提下提高模型训练及推理速度；
- 针对原来的工程数据前处理、后处理流程复杂、计算量大的问题，从新设计数据集存储格式及标注格式、推动产品端修改产品实现，支持更易于处理、计算量更小的标注格式；并在算法框架中添加统一、规范的处理流程，减少计算量，以及修改 DataLoader 的数据加载机制，提高模型训练及推理速度。
- 在后续的工作中持续对各个算法模型的执行速度进行不断优化。

工作成果：

- 不同网络模型训练时间分别优化 15% 50%，推理时间优化 10% 30%，极大提高公司相关产品算法的训练和推理速度。

其他工作

嵌入式产品算法支持及效率优化

2020/04 - 2020/07

小组负责人

工作内容：

- 指导组员在嵌入式产品中完成深度学习算法移植；
- 帮助组员分析嵌入式产品的算法瓶颈，梳理优化方向；

深度学习云产品（AIDI-Cloud）的产品完善及部署落地

2019/04 - 2020/06

系统架构师，小组负责人

工作内容：

- 完善分布式深度学习云训练产品的功能并部署落地到客户现场；
- 根据在部署实施过程中遇到的实际问题及客户提出的需求，重新设计和重构云训练平台产品后端架构，并进行进行技术调研和方案设计；
- 指导后台工程师完成重构功能开发，支持易扩展的分布式训练任务调度等功能，增强和完善用户管理、任务管理、数据标注等功能。

工作成果：

- 完成公司深度学习云训练产品的分布式训练、任务管理等功能的开发，并配合前端工程师一起完善 Web 端标注功能，优化标注体验；
- 完成云训练产品在客户现场的落地部署，完成产品逻辑验证；

## 个人项目

conf-gen

https://github.com/dhmemi/conf-gen

专为 C++ 开发打造的配置文件读写校验及参数调用接口生成器

- 利用模板和宏开发了一套 C++ 参数结构生成器，只需简单的申明语句即可自动生成参数文件模板、参数文件读写和校验及参数存取接口代码，免却在开发中反复编写无聊的参数配置文件读写及校验之苦；
- 提供统一、简洁的参数项协议（仅 8 项原始参数类型），并且用户可以在声明参数的同时非常方便的声明参数的名称及说明信息，可以基于此非常方便的开发与具体参数内容无关的参数显示编辑 UI 界面，将实际的参数内容与参数的界面开发工作解耦；
- 在 8 项高度归纳的常用原始参数类型的基础上，支持组合、嵌套和引用等特性，支持用户根据实际需求声明高复杂度的参数；
- 参数自动全局同步，避免在实际开发中在不同模块间来回传递和同步参数，同时，也能够通过复制参数方便的全局同步，提供灵活的使用方法。

## 技能

Lorem ipsum.

2333/23 - 2333/23

- **编程语言:** 熟练使用 C/C++, Python, CUDA, 6 年 C++ 和 Python 开发经验, 3 年 CUDA 开发经验, 使用 C++ 和 Python 均独立开发/合作开发过大中型项目, 对其语言特性、开发生态等较为了解; 对 Java, C#, JavaScript 等语言有一定了解。
- **算法框架:** Caffe, Pytorch, TensorRT, 熟悉 Pytorch 及 TensorRT 的使用, 对包括 Pytorch, OneFlow, Megenine等多种深度学习框架的底层实现均有所了解和析, 熟悉 Caffe 的底层实现。
- **算法:** 图像处理, 深度学习, 拥有较为丰富的图像相关算法经验, 了解经典的深度学习算法, 并有复现和改进优化经验, 持续关注相关领域的研究进展。此外, 也了解部分传统图像算法原理, 能熟练使用 OpenCV 等工具。
- **软件架构设计:** 3 年经验, 负责设计了公司内部算法应用框架, 负责带领设计和开发了两套分布式深度学习训练和推理系统, 并均完成了落地。
- **其他技能:** 熟悉 Django, MySQL, Docker, Redis 等后台开发技术栈相关工具和技术; 熟悉 Qt 框架。

Last Updated on September 19, 2023