Yong Wei

Increase production capacity and happniess for every one

Email : weiyong1024@gmail.com
Mobile : (+86) 173-7657-1024

Education

- **Master - EE - Beihang University** Beijing, China
  Exempt Entry Exam; Finish courses and research publishing mission during the 1^{st} *semester* *Sep. 2016 – Jan. 2019*
- **Bachelor - EE - Beihang University** Beijing, China
  Exempt NCEE; Honors College(top 50 among 3000+ freshmen); GPA 3.7 *Sept. 2012 – Jun. 2016*

Competitive programming

- **Google Code Jam Kickstart 2017**
  Top 5%, rank 108^{th} *globally.* ***scoreboard, id:WeiYong1024***

Experience

- **Alibaba Cloud** Hangzhou, China
  Senior Software Engineer - middleware, web-backend, lead Eng team *Aug 2020 - May 2022*
  - **Apsara stack - Standard deployment and delivery**: The original distributed deployment process is too heavy based on K8S. To solve this problem, I lead the team to provide this new single-machine solution based on docker-compose. As the united environment initialization scripts are provided, we manipulate client-sites system by maintain different configurations. Base on this solution, a well trained partner engineer can delivery/update 20+ services in 2 workdays, and the monthly deliveries increase to double-digit from single-digit for the team.
  - **Apsara stack - Private cloud management system**: Lead the team to develop the gov scenario private cloud management system. As the in charge character for app-ops-center, I build basic functionalities like resource monitoring, utility statistics, biz inspection, and also key expert competencies such as slow SQL analysis and biz-health check report. The system is deployed in more than 6 major cities in PRC and support several significant businesses such as school entrance of Zhengzhou, China and the national nucleic acid test.
  - **Middleware - Cluster traffic scheduling**: Landing project in Alibaba. As a member of high-availability team, I am in charge of the middleware for cluster traffic scheduling used by the entire group. During my tenure, this system was promoted to empower all Alibaba core services and more than 400K containers are covered by it. The system monitors traffic for every single container, finds outliers in seconds. And abnormal traffic is scheduled and recovered in minutes. Providing escort for 2020 e-commerce festival at 11.11.
- **Pony.ai** Beijing, China
  Software Engineer - Infrastructure *Feb 2019 - Mar 2020*
  - **Voice Logging Pipeline**: Provide the voice logging pipeline to enable single-onboard operator. In this solution, a pair of microphone(for recording) and hand button(for trigger) is equipped for each self-driving vehicle (SDV). First importing the opensource library evdev with BSD license as the input event interface. Then writing a daemon processing based on evdev and ALSA driver from Linux to check the hardware, who listens to signal from the hand button and FIFO connected with the onboard main process, triggers or stops the microphone, and saves the audio files during trips. For the data processing part, first building the speech-to-text service on Google Cloud Platform. Then extract the content from audio files into text. Finally serializing the meta-info by Google Protobuf and create issues by current issue-reporting pipeline.
  - **Car Sound Workflow**: The original Car Sound Module uses Pico library by Google to convert the hardcoded content into audio-stream and then play it out, which is a waste of computing resources and is limited to single Language support. The new car sound module is built on a file-based pipeline, aiming to break those limitations. First, build a internal Python command-line tool (CLI) for engineers to add/update and manage sound suite version: First, write CloudFormation template on AWS to build the Text-to-Speech service, and add a related option in the CLI. Then, add uploading/downloading interface to the storage server in the CLI and enabling local cache. For the onboard part, adding sound names into the sound-request schema, load the audio stream into memory when initializing Car Sound Module, then play the sound when receiving the request.
- **Airbnb** Beijing, China
  Software Engineer Intern - Web Full Stack *Jun 2018 - Sep 2018*
  - **Host Retrospect Page**: Build retrospect page which is build on Ruby On Rails. Also participant in the design work.
- **Megvii (Face++)** Beijing, China
  Research Intern *Jan 2018 - Jun 2018*
  - **Model Search**: Participant in the project for VIVO X21 phone, where my job is to optimize the CNN model. First, by adding a bottleneck layer before the Inception-ResNet module, the performance is improved by 14% without additional FLOPS(float operations per second) overhead. Then, after replacing some $N \times N$ convolutional kernels by a sequence of $1 \times N$ and $N \times 1$ kernels,the $\frac{1}{10{,}000}$ passing rate(ROC value when $x = \frac{1}{10{,}000}$) is improved by 1% on most benchmarks.

Skill Stack

- **System experience**: Cloud computing, cluster high availability, DevOps, self driving vehicle, WEB system, face recognition
- **Programming languages**: Java, C++ and Python are the top 3 most commonly used

魏雍

让每个人更幸福地提高产能

邮箱: weiyong1024@gmail.com
Mobile : (+86) 173-7657-1024

教育背景

- **硕士 - 北航 - 电子与通信工程** 中国，北京
  保研; 研一上发核心期刊达毕业标准; 北京市优干 *2016 年 1 月 – 2019 年 1 月*
- **本科 - 北航 - 电子信息工程** 中国，北京
  物理竞赛保送; 沈元荣誉学院 (入学 top50/3000+); 北京市三好; GPA 3.7 *2012 年 9 月 – 2016 年 6 月*

编程竞赛

- **Google Code Jam Kickstart (谷歌 2017 全球校招赛)**
  全球 108th, 中国 18th, 前 5% **计分板链接** ***(id - WeiYong1024)***

工作经历

- **阿里云** 中国，杭州
  高级工程师 - 中间件、web 后端、带虚线团队 *2020 年 8 月 - 2022 年 5 月*
  - **专有云 - 标准化部署交付**: 大组（P9）层面的技术产品输出原先使用基于 K8S 的分布式部署方案，前期资源申请流程过重。为解决该问题我带虚线团队推出基于 docker-compose 的单机轻量化交付流程。该方案纳管组内 20+ 个服务的交付，基于通用的环境初始化脚本与交付步骤文档，仅通过维护不同客户现场的环境变量即可做到中心化管控，单个经过培训的外包可以在 2 日内交付/升级一个现场。基于该方案，团队技术产品簇的单月商业化输出能力从个位数据点上升到两位数。
  - **专有云 - 专有云管理系统**: 带虚线团队建设政务场景下的专有云管理系统。该系统由应用运维、资源运营、总集管理三大主要职能。本人作为应用运维中心一号位，负责日常迭代开发、产品版本发布管控、并在组内主推工程规范和 CodeOwner 机制严控项目代码质量。应用运维中心具备云资源监控巡检、资源列表、资源使用统计、多种业务巡检、等基础能力，同时具备慢 SQL 统计分析、业务大盘、体检报告等专家能力。该系统目前在全国超过 6 个局点商业化输出，重保护航 2021 某省会城市小学入学等项目、2022 春节核酸检测等项目。
  - **集团中间件 - 流量调度**: 来阿里的 landing 项目，作为高可用团队的一员，负责阿里云内部流量调度中间件。任职期间，推广产品纳管了阿里集团全部核心应用，支撑 40W+ 容器，提供集群单容器维度秒级流量监控、离群点发现、分钟级流量调度与恢复链路。护航 2020 年双十一双峰平稳进行。
- **小马智行** 中国，北京
  工程师 - 基础架构 *2019 年 2 月 - 2020 年 3 月*
  - **行车录音工具链**: 增加通过语音记录 issue 的流程供司机使用以支持单人单车运营。为此新增麦克风和按钮作为硬件方案，首先在 Bazel 项目中引入基于 BSD 软件许可的输入设备接口库 evdev，并基于该库和 Linux 的 ALSA 音频驱动实现负责硬件检测、监听来自外部按钮的信号和车载系统进程的管道信息、触发与停止麦克风录音的守护进程 sound_recorder，在行车过程中将 issue 信息以音频文件的形式存储下来。数据处理阶段，在谷歌云上搭建语音转文字服务，提取音频文件内容，序列化后对接 Issue 汇报流程将相关问题发至 QA。
  - **车载语音系统**: 原有的车载语音模块使用 Google 的 Pico 文字转语音工具库在行车过程中将硬编码的语音内容文本实时播放，从而导致文字转语音的过程需要耗费车载计算资源，同时语言被限制只能使用英文。新的车载语音系统使用基于音频文件的工作流程，旨在减少计算量并支持语音的 I18N。为此首先用 Python 编写内部 click 命令行工具 car_sound_utils(以下简称 CLI)，供工程师用于添加和升级现有语料库、管理语音包版本：首先，在 AWS 上编写 CloudFormation 模板搭建语音转文字的 Web 服务，并在 CLI 中添加对应的调用命令和相应参数，用以在本地生成多语言语音语料。然后，在 CLI 中添加上传、下载语音文件和上传语音包到内部 storage 服务器的接口，并加入本地缓存机制加速下载。对车载系统部分，在播放语音请求的消息原型中加入语音名称，初始化语音模块时使用 Linux 的 ALSA 驱动将音频流加载进内存，并在语音模块接到播放请求消息时播放。
- **爱彼迎** 中国，北京
  实习工程师 - Web 全栈 *2018 年 6 月 - 2018 年 9 月*
  - **房东回顾页面**: moWeb 场景，Ruby On Rails 框架。参与页面设计，独立前后端编码。
- **旷视科技** 中国，北京
  实习炼丹师 - CV 算法 *2018 年 1 月 - 2018 年 6 月*
  - **模型搜索**: 背靠 vivoX21 人脸识别项目，研究移动端 CNN 模型，先通过加入 bottleneck 层在不提高运算开销的前提下提升模型 14% 的原始图片处理能力，后通过使用 2 个串联的 1 维卷积核替代原 N 维卷积核将模型万一通过率（ROC 曲线上 $\frac{1}{10,000}$ 对应纵坐标的值）从 72% 提升至 73%。

技术栈

- **系统经历**: 云计算、集群高可用、DevOps、自动驾驶、Web 系统、人脸识别
- **编程语言**: Java, C++, Python 为主要工作语言