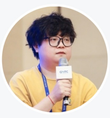

专题出品人：樊中恺

百度 资深研发工程师、文心一言 APP 技术负责人

樊中恺，百度资深研发工程师，文心一言APP技术负责人，2008 年接触前端开发，2012 年开始移动端开发至今，曾先后负责百度浏览器、文库、阅读、百度 APP 前端技术架构、搜索前端架构、推荐前端架构、Paddle.js 等研发工作，对于端智能、工程化、前端架构等方向有较为丰富的经验。QCon 北京 2023 「明星讲师」。

专题：LLM 时代的大前端技术

地点：纽约厅（二层）

本专题将探讨 LLM 时代的大前端技术发展趋势与企业级应用开发的机遇和挑战。

LLM赋能声明式前端框架调试的实践与思考

by 涂旭辉

华为

公共开发部/Web前端技术专家

随着 AI 技术的快速发展，ChatGPT 的亮相进一步提高了人们对生成式 AI 的期待，大语言模型赋能千行百业的时代已经到来。

本次演讲将介绍如何将大语言模型赋能前端调试领域，结合 record & replay 对声名式框架进行交互式调试。开发者通过调试聊天框与模型互动，大模型对缺陷库进行学习增强程序分析推理能力并基于时间戳给出调试建议，开发者结合经验执行调试给出反馈，可以帮助开发者高效准确定位问题根因，为开发者带来全新开发调试范式。

演讲提纲：

1. 背景与趋势
   - 前端框架发展next
   - LLM 赋能千行百业
2. AI 赋能前端领域洞察
   - why AI for debug
   - 技术选型
3. 程序分析技术在前端调试的应用
   - 程序切片技术
   - 程序分析 x LLM
4. 人机交互调试解决方案
   - 传统声名式前端框架调试流程
   - record & replay 交互式调试流程
   - 整体技术架构
   - 实践问题经验分享
5. 未来与展望
   - AI 赋能前端开发全场景

听众收益点：

- 传统前端调试 vs AI 赋能调试
- 程序切片在前端调试的应用
- LLM 对前端开发提效的思考

WebNN，Web 端侧推理的未来

by 胡宁馨

英特尔

软件与先进技术事业部/首席工程师

AI PC 以及 AI Mobile 的新兴时代已经到来，越来越多的设备集成了强大的神经处理单元 NPU，以实现高效的人工智能加速，这对需要端侧推理的应用至关重要。除了通过 CPU 和 GPU 进行推理之外，Web Neural Network API (WebNN) 提供了 Web 应用访问此类专有 AI 加速器 NPU 的途径，以获得卓越性能及更低功耗。

本次演讲将会给大家分享 WebNN API 的 W3C 标准进度，对 CNN，Transformer 以及更广泛的生成式 AI (Generative AI) 模型的支持情况和计划，以及在 Chrome，Edge 等浏览器的实现进展。作为 JavaScript ML 框架的后端，WebNN 将会在几乎不更改前端代码的前提下，为 Web 开发者及他们的产品带来相较于 Wasm，WebGL 更为优异的性能体验。

演讲提纲：

1. 当前 Web AI 发展概况
2. 主流硬件加速器的发展（CPU，GPU，NPU）
3. WebNN 设计与架构
4. WebNN 代码演示
5. WebNN 浏览器（Chromium）实现
6. WebNN 机器学习框架集成（ONNXRuntime 和 TensorFlowLite）
7. WebNN Transformers 支持
8. WebNN 性能

听众收益点：

- 了解 Web 平台对异构处理器的支持
- 了解基于 Web 的机器学习模型硬件加速
- 了解 Chromium 实现内部细节

AI Native 化的大前端开发模式

by 周廷帅

百度

App产品研发部/前端资深工程师

在以大型语言模型（LLM）为背景的时代，对话流已经成为大部分交云互动的主流方式。面对这一趋势，传统的大前端技术如何与强大的模型相融合？除了用户界面（UI），前端技术还能在哪些领域发挥作用？

在本次的分享中，我将向大家展示如何将传统的交互方式与对话流相结合，这包括上下文和状态流转的设计策略。除了交互设计本身，我们还会探讨特定场景下的 PatternPlugin，这将涉及如何将状态机技术应用于肉鸽游戏和活动设计。此外，我还将分享如何利用大型模型来进行业务监控和效果评估。

演讲提纲：

1. 背景：AI Native时代下的大前端开发概述
2. UI扩展插件（UI Extended Plugin）
   - 概览
   - GPTs与文心一言APP：差异与联系
   - 详细设计
3. 模式插件（Pattern Plugin）
   - 状态机与记忆系统
   - 执行解析与流程管理
   - 典型应用场景分享
4. 评估驱动开发（EDD）
   - 性能评估与监控机制
   - 训练数据管理
5. 未来一些想法：
   - LLM模型能力提升对开发流水线及架构工程的影响
   - 大前端的新机遇探索

听众收益点：

- 了解传统交互与 LLM 对话的结合方案
- 了解 PatternPlugin 状态机的设计
- 基于大模型的 EDD 设计思想

利用 LLM 改善研发过程里答疑体验

by 段潇涵

字节跳动

产研&工程部门研发工程师

在当前快速发展的大型语言模型（LLM）时代，像 GPT 系列这样的技术正在改变我们对人工智能应用的理解，特别是在智能客服领域。ChatGPT 的问世更是为我们提供了全新的处理自然语言的方法。在研发领域，不论是提出问题的一方还是解答问题的工程师，答疑工作都是一个避不开的重要环节。

本次演讲将聚焦于研发答疑这一场景，探讨如何利用 LLM 技术协助值班人员提升回复效率，同时帮助提问者更快地找到问题的解决方案，从而整体优化答疑体验。我将结合具体的实际应用场景，深入讨论知识管理、模型交互和效果评测等关键技术方案的实施与应用。

演讲提纲：

1. 答疑场景遇到的体验问题：
   - 提问者视角：等待排队对及时性的影响，了解新技术时遗读文档的成本问题
   - 值班人视角：接到过多的简单重复问题，常用知识沉淀的成本问题
2. 利用 LLM 改善答疑体验思路：
   - 建设知识库，涵盖已有沉淀的实践文档、手册文档、组件库文档站等内容，为常用的咨询类问题利用 LLM + 知识库自动提供回复
   - 利用 LLM 提供答疑过程总结，帮助值班人员快速沉淀 FAQ
   - 通过自动回复前置拦截 → 未解决进入人工答疑 → 自动沉淀 FAQ 闭环，逐步完善，提升自动答疑的覆盖范围
3. 技术方案：
   - 构建知识库
   - Splitting 环节
   - 文本切割方案，利用模型分析文本内容归类，依据文本类型选择更合适的切割方案
   - 针对文档内出现的表格、代码块等进行切割优化
   - 分析文档元信息、预设问题，辅助完善文档解析内容
   - Embedding & Recall
   - 多路召回召回方案，提升检索效果
   - 利用精排模型，二次提升检索准确率
   - 对话
   - 会话维度积累历史，压缩成为临时记忆，减少对 Token 的消耗
   - 会话历史沉淀给模型进行训练优化
4. 效果评测分析：
   - 积累答疑记录，持续沉淀为模型训练数据
   - 通过答疑前置拦截率，评价等关注智能答疑效果
   - 利用答疑记录分析可能的效果提升途径，如：知识覆盖度提升

听众收益点：

- 了解 RAG 理念，基于 LLM 的相关工程落地思路，技术要点
- 了解目前领域的难点与挑战，以及智能答疑领域后续可能发展思路