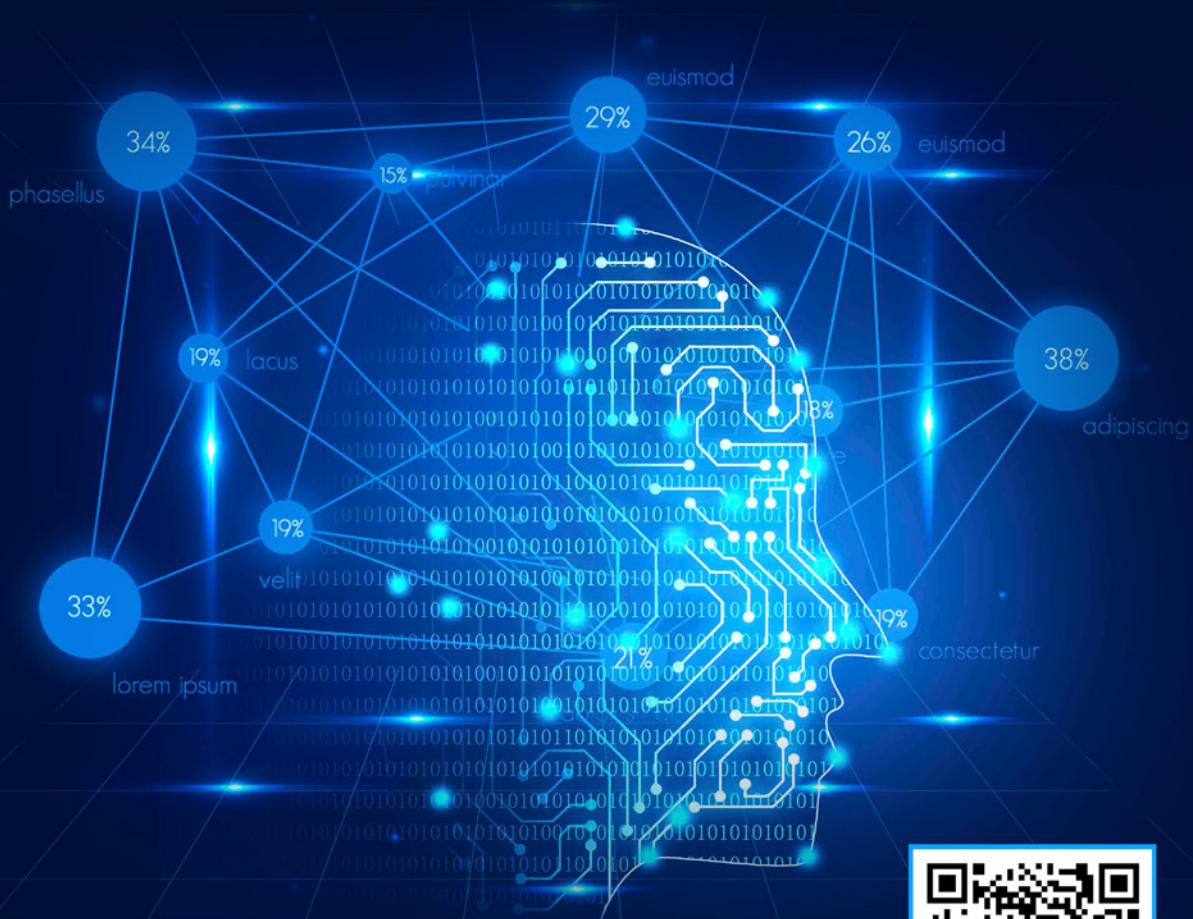




2018年01月刊

A I - F R O N T



关注落地技术，探寻AI应用场景



聚焦最新技术热点 沉淀最优实践经验

[北京站]2018

北京·国际会议中心

演讲：2018年4月20-22日 培训：2018年4月18-19日

精彩案例 先睹为快

《Netflix的工程文化：是什么在激励着我们？》

Speaker: Katharina Probst

Netflix 工程总监

《Apache Kafka的过去，现在，和未来》

Speaker: Jun Rao

Confluent 联合创始人

《人工智能系统中的安全风险》

Speaker: 李康

360网络安全北美研究院负责人，IoT安全研究院院长

《从C#看开放对编程语言发展的影响》

Speaker: Mads Torgersen

微软 C#编程语言Program Manager

《Lavas：PWA的探索与最佳实践》

Speaker: 彭星

百度 资深前端工程师

《浅谈前端交互的基础设施的建设》

Speaker: 程劭非（寒冬）

淘宝 高级技术专家

《深入Apache Spark流计算引擎：Structured Streaming》

Speaker: 朱诗雄

Databricks软件开发工程师, Apache Spark PMC和Committer

《AI大数据时代电商攻防：AI对抗AI》

Speaker: 苏志刚

京东安全 硅谷研究中心负责人

《QUIC在手机微博中的应用实践》

Speaker: 聂永

新浪微博 技术专家

《阿基米德微服务及治理平台》

Speaker: 张晋军

京东 基础架构部服务治理组负责人，架构师

8折 优惠报名中，立减1360元
团 购 享 受 更 多 优 惠

访问官网获取更多前沿技术趋势

2018.qconbeijing.com

如有任何问题，欢迎咨询

电话：15110019061，微信：qcon-0410



ArchSummit 全球架构师峰会

2018 · 深圳站

从2012年开始算起，InfoQ已经举办了9场

ArchSummit全球架构师峰会，有来自Microsoft、Google、Facebook、Twitter、LinkedIn、阿里巴巴、腾讯、百度等技术专家分享过他们的实践经验，至今累计已经为中国技术人奉上了近千场精彩演讲。

● 2017.07.07-08 深圳站

how to use sagas to maintain data consistency in a microservice architecture

--Chris Richardson, *POJOs in Action* 作者, 知名微服务专家

● 2017.12.08-11 北京站

《创新是人类的自信》

--王坚博士, 阿里巴巴集团技术委员会主席

● 2018.7.06-09 深圳站

限时**7折报名中**, 名额有限, 快快抢购。

7折报名中
名额有限, 快快抢购

华南地区架构领域最有影响力的会议，届时有哪些专题和演讲，敬请扫描右方二维码浏览官网。



卷首语：AI 应该是温暖的

刘志勇

如今，我们已经被无数个人工智能包围。从导航软件，到智能推荐系统，再到自动客服……这一些的幕后都由人工智能技术驱动。人工智能默默为人们的工作、生活提供服务，而人们浑然不觉，就像春雨润物细无声一样，人工智能的触角早已渗透日常生活。

自2017年以来，纵观网络各家媒体，人工智能俨然已成为高频词，已经从科研人员的论题发展到全民关心的主流。鉴于此，极客帮科技去年就确定了“AI1 Around AI”的内容战略，并重磅推出了AI前线栏目。

这些变化，在刚刚举办的CES 2018上更是体现得淋漓尽致。CES (International Consumer Electronics Show, 国际消费类电子产品展览会) 有着51年的历史，每一年展会上亮相的黑科技产品，都引领着一个时代。历次CES上，主流都是硬件设备和产品，但今年CES 2018上，由人工智能赋能的创新产业有如春笋怒发。

在这次展会上，人工智能在这些领域大放异彩：语音助手、无人驾



驶、智能家居、无人店铺等等，这些领域有来自Google、Amazon、Intel等国际老牌大厂，也有来自中国本土的百度、阿里巴巴、华为、苏宁、大疆等国内标杆性企业，更有中国的传统家电巨头如TCL、海尔、海信等，据CES官方数据统计，在全部参展厂商中，就有1551家来自中国，整体占比高达33%，在CES展会上刮起了中国风。他们用一个个的发明，改善着人们美好的生活，推动着历史的进程。

这次展会推出的令人眼花缭乱的人工智能创新产品，和以前的产品有所不同，它们为人们带来了真正的、懂人心的消费级人工智能产品，而非以前那种只依靠程序对应和简单搜索，缺少人工智能必备的“学习”元素，只有交互、没有交流的伪AI产品，如以前的聊天机器人经常以“你在说什么？我听不懂。”而终结，用户体验极差，给人一种冷冰冰的感觉，没有一丝温度。作为消费者，人工智能就应该是温暖的。这些创新产品，也提供了新的思路：人工智能在Consumer的落地，首要解决的不是技术的

升级，而是人机关系的变革。人工智能让产品拥有学习能力，不仅可以为人们解决效率问题，更可以为人们带来情感交互。

在我的理想中，Consumer的人工智能产品应该像《超能陆战队》里的大白那样，具备情感计算的能力，通过人工智能的计算机视觉技术，辨识、分析用户的情绪波动，反馈给用户不一样的内容，与用户情感化地交互，带动用户的情绪，产生情感连接，从而产生用户黏度。

而这一些，都离不开六十年来积累的人工智能尖端算法技术和大量数据，在计算机后台形成了丰富的资源。这些数据如何更好地服务人类，应该以怎样的形式和方法，才能最大化数据和技术的价值，这方面的探索，才刚刚开始。

要达到这一雄心壮志的目标，中国人工智能行业的从业者们，任重而道远。程序员不仅仅要掌握基础的开发工具、数据结构，还需要学习更多的数学技术、概率统计、神经网络和理论，学习机器学习的各种算法，还要读论文。现在每年顶尖的学术大会发表的上千篇论文，都可能会对工业界产生影响，因此，程序员要对自己的知识结构进行巨大的升级。请记住，我们AI前线，会和你们在一起，在人工智能征途上，乘长风破万里浪。

我们希望，中国的企业，能够在这一波AI大潮中抢跑，打造一个从技术、到场景应用、再到情感满足的升级，深刻影响人工智能产品的发展。

我们希望，这一天能够尽早到来：人工智能不再是神秘的概念，机器人不再是冰冷的机器，而是带着温度走进千家万户。

AI就应该是温暖的。

AI 前线

InfoQ 中文站 AI 月刊 2018 年 1 月

生态评论

8 AI 前线重磅出品：2017 中国人工智能产业链研究报告

重磅访谈

12 红豆 Live 推荐算法中召回和排序的应用和策略

落地实践

18 视频推荐中用户兴趣建模、识别的挑战和解法

28 携程个性化推荐算法实践

推荐阅读

42 阿里数据库进入全网秒级实时监控时代

精选论文导读

52 伯克利团队解读未来 AI 系统面临的挑战和机会

AI 前线重磅出品：2017 中国人工智能产业生态链研究报告

作者 徐川 陈思



AI 前线导读：2016 年，是人工智能元年

AlphaGo 取得的成功让一部分人看到了人工智能的非凡能力，也让另一部分人看到了机会。这个已经发展了 60 多年，在无数科幻电影里以各种形象示人的神秘事物，终于真真切切地来到了人类的身边，并且变得无处不在。

在中国，人工智能技术已经深入到人们生活的各个角落：就拿最平常的网上购物来说，打开购物网站或 App 的时候，映入眼帘的是根据用户购物习惯，通过智能算法推荐的折扣商品；在网络电商平台购物的时候，为用户答疑解惑的是 AI 智能客服；在购物结束之后，用户可以使用指

纹、声纹，甚至面部识别进行支付。

于是，在潜移默化之中，人工智能真的来了。在国内优秀的科技企业的齐心努力下，中国的人工智能已经逐渐形成了一套中国独有的 AI 产业链。首先请看一下这张图：

我们根据目前中国人工智能发展的现状，绘制出了这套人工智能产业链的层级划分图。

根据麦肯锡 2017 年的调查，金融、医疗、制造等行业应用发展迅速，人工智能领域的全球风投从 2012 年的 5.89 亿美元猛增至 2016 年的 50 多亿美元。麦肯锡预计，至 2025 年人工智能应用市场总值将达到 1270 亿美元。

人工智能的兴起，不仅让科技圈从“互联网 +”时代直接进入了“AI+”时代，更是让投资界的兴趣大为高涨。面对着如此庞大的市场，在投身于 AI 大潮之前，到底人工智能包含哪些内容？如何对人工智能进行分类？相信这些问题不仅困扰着投资界的各位投资人，更多想要踏入人工智能领域的人更是希望能得到一个专业的解释。

于是，这份《中国 AI 生态链报告》应运而生。

我们将人工智能分为这样几个领域

基础设施层

这一层是人工智能技术的基础，提供计算服务以及智能芯片。除了以前老牌的超级计算机仍然在 AI 领域发挥重要作用，分布式计算与当前的人工智能算法相得益彰，得到广泛应用，而最前沿的量子计算也不甘寂寞，有望在人工智能领域大放光彩。在芯片上，国内稍显稚嫩，但也头角峥嵘，夺得第一个手机上智能芯片的殊荣。

数据层

这一层我们又细分为：数据应用、云端大数据、数据平台，三个方向。每个方向都有技术非常突出的企业，我们在这其中选取了几家有代表

性的企业进行了分析。之后的报告中，读者可以看到这几家企业的报告及解读。

算法层

中国在算法方面一直表现突出，据调查，世界人工智能学术界，有 40% 左右的论文是由华人科学家产出，我们熟知的深度学习框架：MXNet，Caffe 等等，都是由华人科学家团队牵头制作的。而在中国，有这样一款深度学习框架，在世界范围内同样引起了巨大的关注，就是来自百度的 PaddlePaddle，同样，我们会有报告在稍后放出。

感知层与认知层

感知智能，包括视觉、听觉、触觉等 AI 能力，让 AI 拥有看、听的能力，于是图像处理技术和自然语言处理技术在这一层就变得尤为重要。在中国，有这样两家企业在这两方面表现可以说走在了世界的前列，一家是已经发展多年的专注智能语音技术的科大讯飞，还有一家是最近势头正猛的商汤科技，关于此二家企业的报告将发布在后面的内容中。

而认知层的认知智能是当前人工智能的热门话题与研究方向，大多数应用 AI 的消费级企业都试图建立知识图谱或用户画像，同时从事语言或文字处理的 AI 公司则希望加强机器对语言的理解。我们将简单介绍这一层的概念及发展现状。

应用和服务层

最上层，也就是应用和服务层，有前面这些技术层提供各项基础技术，方便了许多人工智能初创公司能够加快在各自领域研究的步伐。这一层我们选取了在自动驾驶、智能语音助手等落地应用场景的代表企业进行调研。

这份报告中，我们对人工智能的各个层级进行了详细地划分与分析。在调研过程中，我们也发现了一些目前中国人工智能产业中出现的一些问题，并根据我们目前已有的经验，对一些问题提出了建议。

这是 AI 前线首次尝试此类报告的撰写，也许会有很多不足之处，希望各位读者在阅读完整的报告之后能够与我们分享您的看法，并把这份报告分享出去，让更多人与我们一起，在未来能够完成更加完整、更具有指导意义的人工智能报告，让我们共同推动中国人工智能产业的发展！

完整版报告下载请关注 AI 前线公众号（ID: ai-front），后台回复关键字：“2017 报告”获取下载地址！

红豆 Live 推荐算法中召回和排序的应用和策略

作者 胡南炜



AI 前线导读

有人曾说，“语音直播产品红豆 Live 的突然出现，让沉寂了一段时间的语音知识付费市场又重新燃起了生机”，让语音直播这个小众市场重新吸引了大众的注意力，让声音爱好者找到知音和志同道合之友。但红豆 Live 也用了 AI 这个事实，你知道吗？用到了哪些 AI 技术？推荐算法如何帮助它在众多语音直播产品中脱颖而出？对有意采用 AI 技术的公司有何启示？InfoQ 将在这篇文章中揭开这些问题的答案。

InfoQ 编辑对微博机器学习计算和服务平台负责人胡南炜进行了采访，询问了关于微博旗下的语音直播平台——红豆 Live 应用 AI 技术的

详细情况，以及他对 AI 的深入了解和趋势预测。

红豆 Live 的 AI 布局

据该产品官网数据显示，2017 年 1 月，红豆 Live 面向大众全面开放，KOL 入驻量达 5000+，主播总数量 4 万人，开启了一个全民语音直播的时代。而这款产品的成功，按照该公司的说法，是“AI 发挥的作用不可忽略”。那么，红豆 Live 中究竟采用了哪些 AI 技术？这家公司在 AI 技术方面是否有着深远的布局呢？



The screenshot shows the Red Bean Live homepage. At the top, there's a navigation bar with links for '首页' (Home), '全部直播' (All Broadcasts), '操作指南' (Operation Guide), '有读' (Reading), '下载' (Download), and '充值中心' (Top-up Center). A search bar is also present. Below the navigation, a large banner for a live broadcast titled '听声音的故事' (Listening to Stories) is displayed, featuring two hosts and the date '12月21日 16:30-18:00'. To the right of the banner, there's a promotional section for '上红豆Live' (Go to Red Bean Live) with a QR code. The main content area is divided into several sections: '热门推荐' (Hot Recommendations) with cards for '声优唱见·进来喝杯茶' (Voice Actor Singer · Come In for a Cup of Tea) and '2017互联网第八届牛耳奖颁奖典礼' (2017 Internet Eighth Bull Ear Award Ceremony); '最热豆咖' (Hottest Dukou) featuring profiles like '【获】火力全开... 64.9万' (Achieved full power... 64.9 million) and '彼此~『声C招... 59.1万' (Mutual ~『声C招... 59.1 million); and a '教师资格证面试无忧辅导' (Teacher Qualification Interview Guidance) section.

从技术层面讲，红豆 Live 在 AI 领域使用了语音识别、推荐排序等深度学习技术；其中在推荐排序中红豆 Live 经历了三次算法迭代，从协同过滤到基于内容的推荐，最后到基于音频谱图隐藏特征的深度学习预测模型的演进。“每次的算法迭代都是为了解决用户发现更多优质主播以及提高语音直播内容传播的目标。”胡南炜说道。

众所周知，企业采用 AI 技术需要高昂的成本，在采用这些技术后究竟能产生多大的效果，这是人们非常关心的问题。胡南炜表示，红豆 Live 的推荐模型目标是发现更多主播、用户留存、平均收听时长 3 项。

在应用深度学习预测模型后，从数据表现上，该平台的主播发现率较人工运营时提高了 135%，用户留存率提升 20%，平均收听时长增长 80%。这款产品在应用 AI 后三个重要指标均有较大上涨，因此可以说，深度学习模型对于其业务是有着明显影响的。

语音直播相对来说受众数量较小，那这类产品靠什么来吸引用户呢？

胡南炜认为，虽然语音直播受众数量较小，但确实有效解决了一部分垂直用户的痛点需求。在他看来，直播主要可以满足用户两个方面的需求：娱乐需求和价值需求。顾名思义，娱乐需求是指人们对于娱乐的追求以获得精神满足，直播等视听感受结合的形式可以满足大众的娱乐需求；而满足价值需求，是指直播能给用户带来专业的知识、实用的技能、思路的启发等具有实际意义的东西，解决现实问题。在这方面，他认为语音直播更具优势。另外，音频直播可以更好的将用户的注意力聚焦在内容本身上并降低直播成本，AI 可以帮助忠粉和垂直用户更便利、更有针对性的获取到自己所喜欢的语音内容，从而解决内容获取的痛点。

推荐系统的技术支持详情

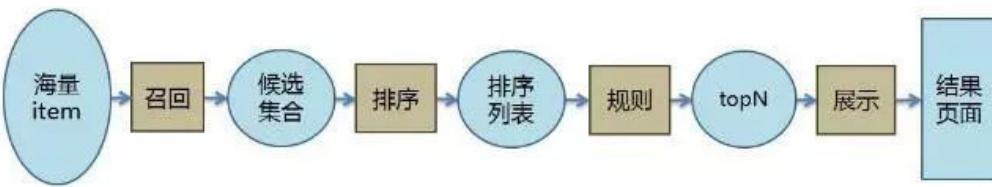
推荐系统的成功离不开背后的技术支持，而部署 AI 更需要强大的技术来做支撑。

红豆 live 推荐系统中使用 CNN+LSTM 用在标签服务里面，把直播间中一些隐藏特征自动化的提取、关联、抽象出来，准备率比起传统机器学习算法大大提高。在 Wide & Deep 排序中，使用宽深度学习网络结合 LR，不仅仅使特征工程的工作量工程量大为降低，而且排序模型的记忆能力和泛化能力比单独使用 LR 提高不少。”从中我们可以看到，推荐系统的算法支持使得红豆 Live 的业务能力显著提高。

然而，没有任何算法是完美无缺的。“红豆 Live 推荐系统主要的缺陷是，推荐系统中的冷启动问题。对于新用户，我们无法获取他们的行为日志和 query 日志。“而针对这个问题，他们有弥补的方法，”因为红豆 live 用户和微博用户重合度为 90%，可以利用该用户的微博兴趣标

签，解决用户的冷启动问题。”胡南炜说道。

关键技术召回和排序的作用和策略



红豆 Live 推荐系统中的两个关键技术分别是召回和排序，其中在召回层用到的策略，是基于 item 的协同过滤，基于用户 query 的 CTR 进行召回，和基于用长短期兴趣的进行召回。而在排序层，则使用 Wide & Deep 网络，主要基于召回层的 item 进行融合、排序，最终选出 top N 个 item 推荐用户。

召回层的作用在于根据用户的不同兴趣，从海量 item 中选出数百个用户感兴趣的 item。而排序层的作用则是基于用户的一些特征，对召回层的 item 再次进行打分排序，更精准地选出用户感兴趣的 item。

具体工作流程

此外，胡南炜还为我们揭示了红豆 Live 推荐算法的具体工作流程：

第一，对用户的行为日志进行利用 JStorm 实时收集，并定时更新基于 item 的协同过滤内容。

第二，对直播间内容进行利用 JStorm 实时收集，实时为直播间打上分类标签、topic、主题词等标签，并定时更新用户画像内容。

第三，对用户 query 日志利用 JStorm 实时收集，定时计算用户 query 的 CTR。

最后，当用户进行刷新时，利用召回策略进行召回，再根据排序策略选择 top N 呈现给用户。

AI 识别“少儿不宜”内容准确率提高

“三俗”内容识别一直是正规内容平台严格把关的方面，AI 能够在这一方面发挥更大的作用。红豆 Live 由于采用了可以提取更丰富特征的新算法，对“三俗”内容进行过滤，因此准确率相较传统机器学习算法有了很大提升。为了保障用户体验，其针对“三俗”内容分别训练模型以及使用敏感词的策略，在对用户进行推荐前，对推荐内容进行实时过滤。

过拟合问题是最大挑战

而被问及红豆 Live 的推荐系统在开发应用过程中遇到的最大困难是什么时，胡南炜表示，任何 AI 技术应用的过程中，神经网络的过拟合问题都是让人头疼的问题，红豆 Live 也不例外，在开发过程中遇到的最大挑战就是它。而他们解决这个问题的思路主要有三点：添加 dropout 层、进行正则化，以及当 loss 和 acc 稳定即停止训练，这或许对我们有所启发。

对 AI 发展趋势的预测

最后，InfoQ 请胡南炜对 AI 行业在未来的发展趋势进行了预测，单就语音直播领域来说，胡南炜认为 AI 技术在语音直播内容分发，以及满足用户个性化语音内容需求等方向会带来深远的影响。“如果说用户碎片时间主要被社交、阅读、音视频等 APP 占据，那么不久的将来也一定会增加语音直播类。”他说道。

而在 2018 年 AI 将有什么样的发展趋势这一问题上，他认为 AI 技术的应用将更加垂直化，AI 技术深入到用户日常生活的每一个方面，比如语音直播。

胡南炜表示，非监督类学习将是红豆 Live 下一步的探索，“我们有这方面的摸索计划，比如在没有标注数据的前提下，我们通过聚类算法将语音直播内容形成一个个的簇，从而做一些粗粒度的随机推荐。”

注：本文观点仅代表受访者本人意见，与受访者所在公司无关。

讲师简介

胡南炜，毕业于北京航空航天大学计算机科学和工程系，在这里完成博士学业之后多年从事软件工程研发和互联网，个人技术专长为大数据、云计算技术和机器学习。他于 2014 年加入微博，负责微博机器学习计算和服务平台开发。在此之前，曾经在 IBM、Yahoo 等公司工作。

视频推荐中用户兴趣建模、识别的挑战和解法

作者 李玉



AI 前线导读

优酷每天为上亿用户推荐上亿的视频，机器学习模型如何更好的描述与捕捉用户的兴趣成为一大挑战。相比电商、新闻等领域用户对于视频内容的兴趣要更为复杂、感性、微妙、纬度多样，用户的兴趣也会逐渐演进、变化、细分，对于惊喜度（serendipity）与多样性（diversity）的要求也更高。用户的行为数据稀疏、分布偏差大、时域上分布规律也复杂多样。12月9日 ArchSummit 北京架构师大会上，优酷数据智能部总监李玉博士，就优酷在《视频推荐中用户兴趣建模、识别的挑战和解法》进行演讲，跟大家分享优酷视频的个性化搜索推荐里跟用户兴趣相关一些问

题和思考。

以下为演讲实录。

一、个性化服务在优酷

本文将介绍一下优酷个性化搜索推荐的服务，优酷在视频个性化搜索推荐里用户兴趣个性化表达碰到的挑战和问题，当前工业界常用的方法，以及我们针对这些问题的尝试。

首先优酷已经非常大量的全面的采用了大量的个性化的搜索推荐技术，今天优酷为几个亿用户提供的服务是全面的，千人千面的个性化服务。在优酷的首页，所有用户看到的内容、推荐的内容都是根据用户个性化的兴趣匹配的完全不一样的内容。在优酷各个垂直频道，像电影频道、综艺频道，也完全采用个性化技术做分发。优酷有大量短视频，优酷短视频信息流的场景，也是大量的采用个性化技术在做分发。优酷的用户和内容都用大量算法做匹配。

今天优酷有多一半视频播放是通过个性化搜索推荐技术做分发的，个性化搜索推荐技术对于优酷 CTR，包括人均播放量，人均时长，留存率都有非常大的提升；更重要的是个性化的算法对于帮助用户发现好的内容，帮助互联网优质内容可以精准触达适合它的受众，在这一点上是贡献更大的，可能比单纯看 CTR 提升多少，或者人均播放量提升多少更重要。而且有时候帮助用户发现好的内容和帮助好的内容触达用户，有时候并不适合简单的业务指标，像 CTR、人均播放量不一定是一致的，所以怎么把用户的兴趣识别更准，怎么把内容推送的更准，这个事情比单纯的关注点击率、人均播放量提升更重要。

二、视频推荐里个性化兴趣表达的挑战

做视频个性化的兴趣的识别还是有非常多的挑战。

第一，优酷业务模式最核心的重点是一些头部内容，像电影、电视剧、综艺、动漫这些核心的头部内容，头部内容用户的选择成本特别高，

用户要追几十集电视剧的话，它要考虑很多问题，很难真正选择开始追一个剧或者一个综艺，所以推荐的成功率往往偏低的，比较困难能够让客户真正推荐方式追一个剧。用户来使用优酷服务的时候目的性往往是较强的，他带着比较强的比较精准的意图过来，发现和浏览逛的心智偏低。

第二，长视频的节目选择空间往往比较有限，算法更适合分发大量的长尾的内容，但是对于优酷这样的场景，选择空间有限的情况下，怎么把推荐这个事情个性化的服务做得更好，这也是比较大的挑战。

第三，头部节目用户行为往往是比较稀疏的，很多头部节目有大量的用户是不够活跃，每个月只有小于三个视频播放的行为。如果再看优酷短视频信息流的场景，用户可能有几百个行为。同样一屏会推荐比如 30 个，在短视频的场景可能对这个用户的了解是几百个观看行为，我推荐 30 个，但是长视频头部的节目里我只知道用户看过三个相关的视频，三个头部的视频，要推荐 30 个节目，所以是两个完全不同的问题，可能需要完全不同的做法，完全不同的模型和算法去解。

另外数据的噪声很多，数据的分布往往比较趋热，传统常见的模型甚至复杂的 DAN 的模型往往效果不好，因为数据的分布噪声非常大，数据的稀疏性比较大。从数据本身角度来看，视频的兴趣非常感性和微妙的，非常复杂，我们刚刚开始做优酷个性化服务的时候，一开始想把电商很成功的做法和系统搬过来，发现会碰到很多问题。对比一下电商，用户的兴趣非常简单明确，我想买一个电视或一条牛仔裤是非常明确的，他的意图是高度结构化，比如类目体系非常清晰，但视频是非常感性和微妙，比如有些用户喜欢武侠片，但是并不喜欢成龙这一类的武侠片，他是存在某种非常复杂的因素在里面。而且视频内容的兴趣往往是非常动态的，不是静态的，不断的演进，不断的发展。比如科幻的兴趣，有的是轻度科幻，也有中度的，也有重度的，是逐渐发展过来的。很多时候用户视频的内容兴趣还体现了很多亚文化的角度，比如二次元的角度，比如文艺青年，这些角度用户的观看兴趣是不同的。有时候用户视频兴趣体现用户个人的认同，视频维度非常多样，非常正交，越来越细分和多样化。比如我们有时

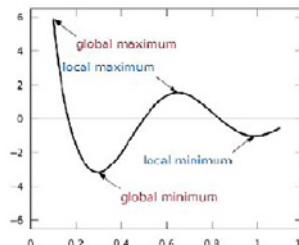
候看一些案子，发现有的客户什么类型都看，也会看魔幻，也会看动作片、武打片，也会看新的，也会看几年前。后来发现他看所有的东西都是大制作，都是制作成本很高的，大制作可能也是兴趣的维度；还有前一段时间《白夜追凶》的剧，很多人描述是美剧质感，这是一个很好的维度，很多用户会喜欢这个维度。很多时候你的视频就在于你怎么梳理类目体系，包括用户对于内容的兴趣是不喜欢重复，识别出来适合他喜欢的还不够，因为用户对兴趣度和多样性的要求是远远高于其他的品类。

我们在不断思考的是用户这些内容的兴趣怎么通过传统的推荐的技术能不能表达好？能不能把这么复杂多样的微妙的用户在视频观看的兴趣表达出来？我们的模型有没有表达的能力表达出来这么复杂的规律？我们的特征有没有足够强的特征表达这些事情。

识别用户的兴趣是非常重要的，往往一个实际产品的问题不能用简单一个方式去表示。

识别、表达用户兴趣的重要性

- Retargeting（看了又看）：
 - 推荐用户有过交互的内容（看了又看）
 - 成功率高，长期价值低
 - 局部提升非全局提升（抢其他渠道流量）
 - 通过ROI价值衡量
 - 容易陷入局部最优
- 热点推荐
 - 推荐近期热点
 - 容易陷入局部最优
- 个性化兴趣推荐
 - 推荐符合每个用户兴趣的内容
 - 更具长期价值
 - 短期收益可能小，但容易长期收敛
- 推荐命中成功率：retargeting > 热点 > 个性化发现
- 推荐命中价值：个性化发现 > 推荐热点 > retargeting



大部分传统的推荐算法都是用点击率预估去训练一个模型，推荐的内容可能有几种类型，一种类型是看了又看，推荐用户看过的，有过行为的东西，做过广告的人知道一个概念，就是用户有过交互的东西。第二类是热点，比如统计 CTR 很高的东西，除了这两类之外，才是真正去猜测

和预估兴趣用户，根据用户兴趣去做推荐的。一般的模型会推这三类东西，这三类东西不一样，如果做过这个事情的人都知道，最有效的点击率最容易高的是推用户有过行为的东西往往很容易有效，但是推这些类型的价值率不高，推荐率高是因为难度比较低，成功率容易高，真正个性化的内容通过猜测用户的兴趣点去做推荐，往往你的成功率偏低，所以点击率偏低，所以从成功率来讲推荐有过行为的最高。推荐命中或者不命中的价值，都是个性化推荐是更高的。即使你推荐的某种给用户没有命中，也会提供一个很负样本信息，你对这个用户的兴趣点了解更深入，知道这个兴趣不感兴趣什么。相反如果你推的都是成功率很高的东西，你的模型长期来讲很容易陷入一个局部自由，因为你收到正负样本没有什么变化，你没有真正探索到用户的兴趣。

三、常见的工业界的做法

针对上边这些视频推荐里的挑战，我们来看看常见的工业界的做法，基本流程作为召回，然后筛选一部分候选，然后做一个排序模型，很多大部分的公司都是这样做的。排序模型里会有一些统计特征、用户的画像，画像里包括 DEMO 的特征，还有基于内容标签的用户画像的特征，包括高微组合特征等等。如果从结构来讲的话，可以分为四个层次，最下面是数据层，然后是召回层，然后是一些特征工程的层次。在数据层面往往是 ETL 数据处理、采样。优酷做了大量不同的召回，经典的像 CF，包括 DNA 的 CF，还有 Slim，还有基于明星的召回，基于热度热点和趋势的召回。特征是比如说 Item 的特征。包括用户画像常用用户搜索记录、浏览记录。模型也有各种各样的。这些都是蛮常见的做法。

这些做法对于刚才我描述的视频兴趣的表达还是有很多问题，我们可以具体看一下。比如从特征的角度，使用这些 DEMO 往往有一个特征，用户视频兴趣往往和用户的年龄、性别和地域关系不大，比如三线城市 50 岁男性和一线城市 30 岁的女性看的东西是差不多的。基于用户标签画像，常见的做法是基于这个内容的标签去生成一些用户的画像，基于一些

统计方法，针对这些特征做一些高纬的组合，比如用 DNN 也好，在视频推荐场景里，特别是头部内容推荐的场景里，行为过于稀疏，数据噪声比较大的时候，在高纬空间组合特征往往效果不是特别好。因为所有空间是很大的，往往很复杂的模型并不是一个全局收敛，往往收敛到一个局部，会发现组合出来的特征是不太准确的。在噪声大，数据稀疏的产品 i2i 里往往超过很好，在场景噪声很大的时候，对于数据降纬的处理。当我们把空间投影到 I 空间里，通过时间积累有很多用户产生行为，这时候数据量就会大很多。所以识别出来之间相似度就会好很多。我们可以理解成某种降纬的降噪对于数据稀疏性和噪声过大一种降噪的处理。基于这个相似度再作为特征放到排序模型里。

导致一个问题就是很容易丢失很多重要信息，比如很多用户看有的是因为喜欢这个主演，有的是因为我喜欢这个类型，也有可能因为这个剧比较热，所以去看，但是并不喜欢这个类型。这个剧热度过了之后就不看了。所以在这种情况下没有办法很好的表述这些信息。另外一个问题，不同的用户群体对于全局也很难特点好的表示不同用户群体的信息。下面这个图是 16 年解释的现象，当你去计算 C 和 I 之间相似度的时候，在 A 的用户群体里相似度比较高，但是在 B 这个群体里相似度是零，因为没有看过 I。当你算一个全局相似度的时候，它是这两个相似度的平均，对于 B 的相似度计算是不准确的，但是也只能解决一部分这样的问题。另外往往力度过细，没有像标准这样有一定的扩展性，趋热是很常见的现象。比如哈利波特，像这样的大热片跟所有的相似度都很高，怎么做热点的打压也是一个问题。

四、个性化推荐在优酷视频的应用

下面介绍一下针对这些问题我们的一些尝试。首先还是认为要非常精细地去做好用户的画像非常重要，传统做法以内容标签为基础产出大量内容标签、类目体系、导演演员，然后根据用户观看内容的行为，对于各个内容的标签分类产生一些用户的画像。传统的做法是基于统计的方法，

并没有很细致的解决好这个问题。

为什么我们觉得做好用户兴趣画像很重要？我们解这个问题很大的挑战在于数据的稀疏性很高，数据里面的噪声很大。对于稀疏性高、噪声很大的问题，如果你用一个很复杂的模型去解这样 N2N 的问题，模型很容易受到噪声的干扰，所以这个效果往往不好。右边这个图一般做法是你的输入是用户的行为，用户的数据，输出就是给用户推荐。如果简单用一个 MEDOL 解决这个问题很容易受到噪声影响。因为现在虽然技术发展很快，但是很多模型没有我们想象那么强大。我们做法是把这些问题拆解成若干个子问题，不是把一个 N2N 的问题 MEDO，而是拆解成若干个子问题，让 MEDO 解决更容易的问题，然后把人工的业务理解加入进去对这个问题做降纬和降噪，我们希望学到一个很准确的用户兴趣的表达，然后把用户的行为首先表现在用户兴趣表达上，把这个问题拆成两个问题，基于这个兴趣表达再去做推荐，拆成两个子问题，可以进行人工空间的整理、系统的建设、类目的建设，包括降纬降噪的处理。

这个思路并不新，学术界过去有很多工作，比如像 CTR，基于这样模型可以更精细的把用户画像解的更准。近几年也有很多围绕这个方向有很多进展，我们尝试有几个工作，像 CTPF 工作，还有下面的工作，都是效果蛮不错的。

介绍一下在这个方向上的进展和尝试。它把概率分布的假设换成播送分布，更多假设这个数据分布更稀疏，更符合实际用户在视频观看里的一些规律，因为用户在视频观看的时候，所谓时间是有限，不论我对这个兴趣再高，我只能看有限的视频，所以分布是偏稀疏。播送分解的思路效果更好。

我们实际应用的时候和学术界的假设还是不太一样，我们除了视频文本信息，因为视频文本信息在实际工业运用里往往噪声比较大，我们视频文本信息，比如视频的抬头，视频的简介。往往多视频是有点标题党，抬头并不能很好表示这个视频所讲的。实际应用里有大量编辑团队对这个视频打了标签，标签的问题有时候偏主观，也有随机性的噪声里面，我们可

以把这些因素用一个概率模型去 MEDO 出来，我们认为这种标签也有某种随机分布。

包括视频热度的维度，我们希望把热度维度单独拆开，希望能够看到哪些用户是比较喜欢热点的，因为这个东西热才去看的，哪些用户是真正因为这个东西的类型才去感兴趣看的。按照这样的思路，我们可以把视频文本的维度、标签的维度、搜索词的维度，比如主演、主角、配角、导演这些信息都可以进行一个独立的分布。EM 迭代并不是全局收敛的方法，往往收敛到局部最优，我们需要做大量人工审核工作，做一个降噪的工作。审核以后把并不好的去掉，做第二次初始值，然后再做迭代，迭代以后会收敛。我们对这个方法做了各种实现上的性能上的优化，基于分布式架构，改造成一个分布式，让它可以处理优酷几亿视频和几亿用户的行为。

用户内容兴趣有复杂的规律，有些兴趣是长期，有些星期是短期的，当兴趣点满足了之后更追求多样性，过一旦时间又希望再看到这一类的内容，我们用了 GRU 的结构希望把用户兴趣时间维度上的信息能够学出来。网络输入是用户观看的 ID 序列，包括观看序列里标签的序列，经过 GRURECNET，再经过多个输出接层。我们做了一个改进，用户观看序列是非常不等距的，有的用户观看是一个月只有一个行为，有的一天有很的行为，跟 GRU 假设不一样。除了增加采样的间隔以外，把 GRU 改成 Time gatek。

刚才提到一个挑战很多用户的行为是比较有限的，我们对这些用户兴趣并不能识别很准，所以我们用了传染病模型建模的方式。中度活跃用户早期也是行为比较少的，我们可以看用户是怎么从早期行为稀少的阶段逐渐演变到中度活跃度比较高的情况。我可以计算一个用户各种兴趣点演进概率和演进的方式，基于这个预测用户将来会对什么感兴趣，根据这个去建模，把这个作为特征放到 MODEL 里，然后基于这个预测的概率做对用户兴趣的捕捉。

关于数据稀疏性的问题，最直接的解法就是通过随机流量有意识收

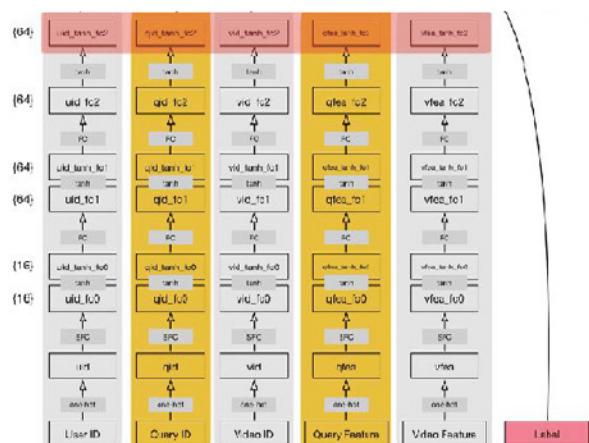
集更多的数据，行为数据，包括交互数据。这里有一个问题，当其实数据是很稀疏的时候，当 $N \times N$ 的 I2I 的矩阵有很多元素很稀疏的，explore 收集数据需要很多流量，代价很高。所以我们考虑 NystromCUR， $N \times N$ 矩阵可以用这个矩阵当中的 C 行，C 是远远小于 N 的。所以沿着这个思路，通过 statistical leverage score 选择 C 个 item。重点 explore 对于 c 个 item 有过观看的用户。只有 C 行是比较稠密的，C 行数据是很准确的。当 C 行算的比较准的时候，还可以乘以 $N \times N$ 的矩阵。

另外可以构建一个 HIN 图，每个节点就是一个标签，按照标签的相似度去做迭代，通过迭代会收敛，下面是一个例子，可以像汽车建筑队、迷你卡车之类的，都是给小朋友看的视频，但是都是表达一类用户的兴趣点，可以通过这种方式找到更多用户的兴趣点。

特征交叉也是一个比较有趣的问题，算法模型能力有限，End2end 模型精准 capture 个性化特征能力有限。最优解在非常高纬空间中，由于噪音与模型收敛能力问题，需人工辅助降低搜索维度。使用交叉特征的统计值，效果好于使用离散交叉裸 id 特征。结合业务理解，辅助模型更好 capture 个性化特征。结合统计量的 variance 进行噪声过滤。

个性化排序在优酷视频搜索-特征域划分及编码

- query user video id 域 统计域 用户观看序列 标签兴趣 文本
- 超高维的稀疏编码来表征独立个体
- 利用神经网络来拟合个体共性
- 视频表达是基础
- 深度特征的组合表达能力
- 按特征的重要度和关联性分域



这是在优酷个性化搜索排序里的模型，这个模型会分成若干个域，域内部是全连接，然后还有 concat，还有域内信息的二次编码，还有稀疏全连接。

这个模型是当你的数据没有那么稀疏的时候，你的做法很简单，就是用足够深的表达能力足够深度的模型就可以把这个问题解的很好。输入里像用户的域是用用户所有观看记录，所有的视频观看的 ID 全部都放进去，每一个用户就是一个 ID 序列。视频维护有视频各种文本、标题，包括视频各种标签分类组合起来表达这个视频。这个问题就可以用这个模型解决很好。但是规模太大，参数已经是上亿级别，特征维度太高，这个模型虽然表达能力非常强，可以把落后裸 id 特征学习很好，但是从模型存储、离线的训练，包括在线网络的预估过程不能响应时间的要求，所以我们很大量的工作在这个领域做模型的压缩和编码的压缩。对于输入层稀疏 ID 的表达做各种压缩，包括域的选择为什么要拆开若干个域，对于离线和在线的评估也好、训练也好可以做一些效率上的提升。包括在离线版本和在线预估的版本，这两个也是有特征模型压缩的区别。包括在优酷搜索里的召回阶段，包括索引阶段，都会把深度表征学习大规模的模型建到索引里。

作者简介

李玉博士，花名谈志，优酷数据智能部总监，负责优酷的个性化推荐、搜索、泛内容 AI 平台、视频 AI 理解等。加入阿里前曾在美国 Uber 负责个性化智能定价、补贴、拼车规划等工作；在京东任京东数据云总监；在美国雅虎负责雅虎的 DSP 广告平台、广告 Targeting 等工作。

携程个性化推荐算法实践

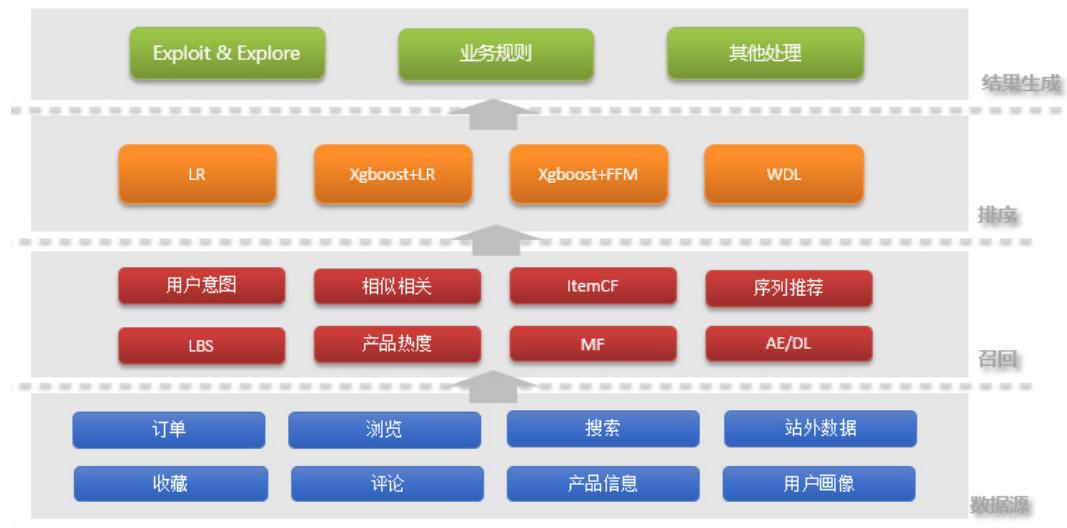
作者 携程基础业务研发部



AI 前线导读

携程作为国内领先的 OTA，每天向上千万用户提供全方位的旅行服务，如何为如此众多的用户发现适合自己的旅游产品与服务，挖掘潜在的兴趣，缓解信息过载，个性化推荐系统与算法在其中发挥着不可或缺的作用。而 OTA 的个性化推荐一直也是个难点，没有太多成功经验可以借鉴，本文分享了携程在个性化推荐实践中的一些尝试与摸索。

推荐流程大体上可以分为 3 个部分，召回、排序、推荐结果生成，整体的架构如下图所示。



召回阶段，主要是利用数据工程和算法的方式，从千万级的产品中锁定特定的候选集合，完成对产品的初步筛选，其在一定程度上决定了排序阶段的效率和推荐结果的优劣。业内比较传统的算法，主要是 CF^{[1][2]}、基于统计的 Contextual 推荐和 LBS，但近期来深度学习被广泛引入，算法性取得较大的提升，如：2015 年 Netflix 和 Gravity R&D Inc 提出的利用 RNN 的 Session-based 推荐^[5]，2016 年 Recsys 上提出的结合 CNN 和 PMF 应用于 Context-aware 推荐^[10]，2016 年 Google 提出的将 DNN 作为 MF 的推广，可以很容易地将任意连续和分类特征添加到模型中^[9]，2017 年 IJCAI 会议中提出的利用 LSTM 进行序列推荐^[6]。2017 年携程个性化团队在 AAAI 会议上提出的深度模型 aSDAE，通过将附加的 side information 集成到输入中，可以改善数据稀疏和冷启动问题^[4]。

对于召回阶段得到的候选集，会对其进行更加复杂和精确的打分与重排序，进而得到一个更小的用户可能感兴趣的产品列表。携程的推荐排序并不单纯追求点击率或者转化率，还需要考虑距离控制，产品质量控制等因素。相比适用于搜索排序，文本相关性检索等领域的 pairwise 和 listwise 方法，pointwise 方法可以通过叠加其他控制项进行干预，适用于多目标优化问题。工业界的推荐方法经历从线性模

型 + 大量人工特征工程^[11] → 复杂非线性模型 → 深度学习的发展。Microsoft 首先于 2007 年提出采用 Logistic Regression 来预估搜索广告的点击率^[12]，并于同年提出 OWLQN 优化算法用于求解带 L1 正则的 LR 问题^[13]，之后于 2010 年提出基于 L2 正则的在线学习版本 Ad Predictor^[14]。Google 在 2013 年提出基于 L1 正则化的 LR 优化算法 FTRL-Proximal^[15]。2010 年提出的 Factorization Machine 算法^[17]和进一步 2014 年提出的 Filed-aware Factorization Machine^[18]旨在解决稀疏数据下的特征组合问题，从而避免采用 LR 时需要的大量人工特征组合工作。阿里于 2011 年提出 Mixture of Logistic Regression 直接在原始空间学习特征之间的非线性关系^[19]。Facebook 于 2014 年提出采用 GBDT 做自动特征组合，同时融合 Logistic Regression^[20]。近年来，深度学习也被成功应用于推荐排序领域。Google 在 2016 年提出 wide and deep learning 方法^[21]，综合模型的记忆和泛化能力。进一步华为提出 DeepFM^[15] 模型用于替换 wdl 中的人工特征组合部分。阿里在 2017 年将 attention 机制引入，提出 Deep Interest Network^[23]。携程在实践相应的模型中积累了一定的经验，无论是最常用的逻辑回归模型（Logistic Regression），树模型（GBDT, Random Forest）^[16]，因子分解机（Factorization Machine），以及近期提出的 wdl 模型。同时，我们认为即使在深度学习大行其道的今下，精细化的特征工程仍然是不可或缺的。

基于排序后的列表，在综合考虑多样性、新颖性、Exploit & Explore 等因素后，生成最终的推荐结果。本文之后将着重介绍召回与排序相关的工作与实践。

数据

机器学习 = 数据 + 特征 + 模型

在介绍召回和排序之前，先简单的了解一下所用到的数据。携程作为

大型 OTA 企业，每天都有海量用户来访问，积累了大量的产品数据以及用户行为相关的数据。实际在召回和排序的过程中大致使用到了以下这些数据：

- 产品属性：产品的一些固有属性，如酒店的位置，星级，房型等。
- 产品统计：比如产品一段时间内的订单量，浏览量，搜索量，点击率等。
- 用户画像：用户基础属性，比如年纪，性别，偏好等等。
- 用户行为：用户的评论，评分，浏览，搜索，下单等行为。

值得注意的是，针对统计类信息，可能需要进行一些平滑。例如针对历史 CTR 反馈，利用贝叶斯平滑来预处理。

召回

召回阶段是推荐流程基础的一步，从成千上万的 Item 中生成数量有限的候选集，在一定程度上决定了排序阶段的效率和推荐结果的优劣。而由 OTA 的属性决定，用户的访问行为大多是低频的。这就使得 user-item 的交互数据是极其稀疏的，这对召回提出了很大的挑战。在业务实践中，我们结合现有的通用推荐方法和业务场景，筛选和摸索出了几种行之有效的方法：

Real-time Intention

我们的实时意图系统可以根据用户最近浏览下单等行为，基于马尔科夫预测模型推荐或者交叉推荐出的产品。这些候选产品可以比较精准的反应出用户最近最新的意愿。

Business Rules

业务规则是人为设定的规则，用来限定推荐的内容范围等。例如机票推酒店的场景，需要通过业务规则来限定推荐的产品只能是酒店，而不会推荐其他旅游产品。

Context-Based

基于 Context 的推荐场景和 Context 本身密切相关，例如与季候相关的旅游产品（冬季滑雪、元旦跨年等）。

新年打折季，去血拼吧

[更多 >](#)



香港 ¥320起

冬季打折季从圣诞一直持续到来年1月份，商品品类，名列世界之最。



东京 ¥504起

东京绝对无愧于"地球最高的城市"。

冬日养生温泉，要惬意

[更多 >](#)



东京 ¥504起

黑川温泉，箱根温泉，去温泉乡体验露天温泉与田园风光的美妙结合。



台北 ¥524起

舒展在一池温暖的水山谷的宁静。

新年打折季，去血拼吧

[更多 >](#)

香港 ¥320起

冬季打折季从圣诞一直持续到来年1月份，商品品类，名列世界之最。



东京 ¥504起

东京绝对无愧于"地球最高的城市"。

冬日养生温泉，要惬意

[更多 >](#)

东京 ¥504起

黑川温泉，箱根温泉，去温泉乡体验露天温泉与田园风光的美妙结合。



台北 ¥524起

舒展在一池温暖的水中的宁静。

LBS

基于用户的当前位置信息，筛选出的周边酒店，景点，美食等等，比较适用于行中场景的推荐。地理位置距离通过 GeoHash 算法计算，将区域递归划分为规则矩形，并对每个矩形进行编码，筛选 GeoHash 编码相似的 POI，然后进行实际距离计算。

Collaborative Filtering

协同过滤算法是推荐系统广泛使用的一种解决实际问题的方法。携程个性化团队在深度学习与推荐系统结合的领域进行了相关的研究与应用，通过改进现有的深度模型，提出了一种深度模型 aSDAE。该混合协同过滤模型是 SDAE 的一种变体，通过将附加的 side information 集成到输入中，可以改善数据稀疏和冷启动问题，详情可以参见文献^[4]。

Sequential Model

现有的矩阵分解 (Matrix Factorization) 方法基于历史的 user-item 交互学习用户的长期兴趣偏好，Markov chain 通过学习 item 间的 transition graph 对用户的序列行为建模^[3]。事实上，在旅游场景下，加入用户行为的先后顺序，从而能更好的反映用户的决策过程。我们结合 Matrix Factorization 和 Markov chain 为每个用户构建个性化转移矩阵，从而基于用户的历史行为来预测用户的下一行为。在旅游场景中，可以用来预测用户下一个目的地或者 POI。

除此之外，也可以使用 RNN 来进行序列推荐，比如基于 Session 的推荐^[5]，使用考虑时间间隔信息的 LSTM 来做下一个 item 的推荐等^[6]。

此外，一些常见的深度模型 (DNN, AE, CNN 等)^{[7][8][9][10]} 都可以应用于推荐系统中，但是针对不同领域的推荐，需要更多的高效的模型。随着深度学习技术的发展，相信深度学习将会成为推荐系统领域中一项非常重要的技术手段。以上几种类型的召回方法各有优势，在实践中，针对不同场景，结合使用多种方法，提供给用户最佳的推荐，以此提升用户体验，增加用户粘性。

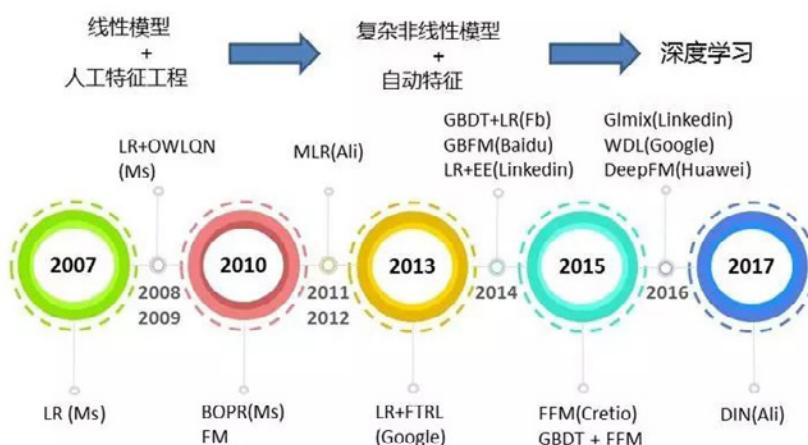
排序

以工业界在广告、搜索、推荐等领域的实践经验，在数据给定的条件下，经历了从简单线性模型 + 大量人工特征工程到复杂非线性模型 + 自动

特征学习的演变。在构建携程个性化推荐系统的实践过程中，对于推荐排序这个特定问题有一些自己的思考和总结，并将从特征和模型这两方面展开。

Model

个性化排序模型旨在利用每个用户的历史行为数据集建立其各自的排序模型，本质上可以看作多任务学习 (multi-task learning)。事实上，通过加入 conjunction features，也就是加入 user 和 product 的交叉特征，可以将特定的 multi-task 任务简化为单任务模型。梳理工业界应用的排序模型，大致经历三个阶段，如下图所示：



本文并不准备详细介绍上图中的算法细节，感兴趣的读者可以查看相关论文，以下几点是我们的一些实践经验和体会。

- 在实践中选用以 LR 为主的模型，通过对数据离散化、分布转换等非线性处理后使用 LR。一般的，采用 L1 正则保证模型权重的稀疏性。在优化算法的选择上，使用 OWL-QN 做 batch learning，FTRL 做 online learning。
- 实践中利用因子分解机 (Factorization Machine) 得到的特征交叉系数来选择喂入 LR 模型的交叉特征组合，从而避免了繁杂的特征选择工作。一般的受限于模型复杂度只进行二阶展开。对于

三阶以上的特征组合可以利用基于 mutual information 等方法处理。已有针对高阶因子分解机 (High Order FM) 的研究，参见文献^[24]。

- 对于 Wide and Deep Learning，将 wide 部分替换 gbdt 组合特征，在实验中取得了较好的效果，并将在近期上线。后续的工作将针对如何进行 wide 部分和 deep 部分的 alternating training 展开。

Feature Engineering

事实上，虽然深度学习等方法一定程度上减少了繁杂的特征工程工作，但我们认为精心设计的特征工程仍旧是不可或缺的，其中如何进行特征组合是我们在实践中着重考虑的问题。一般的，可以分为显式特征组合和半显式特征组合。

显式特征组合

对特征进行离散化后然后进行叉乘，采用笛卡尔积 (cartesian product)、内积 (inner product) 等方式。

在构造交叉特征的过程中，需要进行特征离散化；针对不同的特征类型，有不同的处理方式。

numerical feature

无监督离散化：根据简单统计量进行等频、等宽、分位点等划分区间

有监督离散化：1R 方法，Entropy-Based Discretization (e.g. D2, MDLP)

ordinal feature (有序特征)

编码表示值之间的顺序关系。比如对于卫生条件这一特征，分别有差，中，好三档，那么可以分别编码为 $(1, 0, 0), (1, 1, 0), (1, 1, 1)$ 。

categorical feature (无序特征)

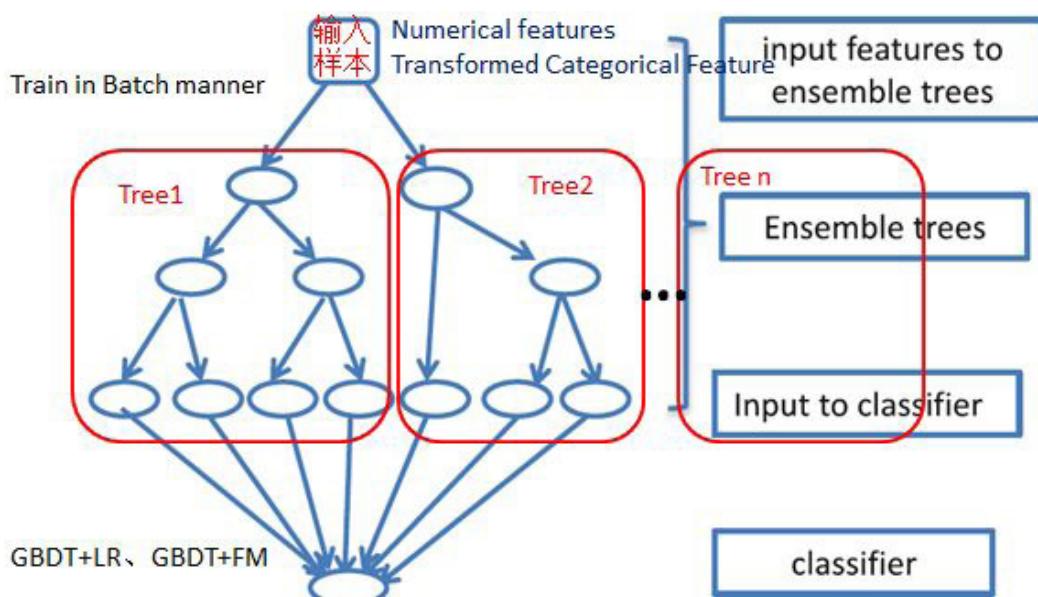
- 离散化为哑变量，将一维信息嵌入模型的 bias 中，起到简化逻辑回归模型的作用，降低了模型过拟合的风险。
- 离散特征经过 OHE 后，每个分类型变量的各个值在模型中都可以看作独立变量，增强拟合能力。一般的，当模型加正则化的情况下约束模型自由度，我们认为 OHE 更好。
- 利用 feature hash 技术将高维稀疏特征映射到固定维度空间

离散化方法	具体做法
OHE(one hot encoding)	用h个变量代表h个level
Dummy Encoding	将一个有h个level的变量变成h-1个变量
Hash Trick	转化为固定长度的hash variable

半显式特征组合

区别于显式特征组合具有明确的组合解释信息，半显式特征组合通常的做法是基于树方法形成特征划分并给出相应组合路径。

一般做法是将样本的连续值特征输入 ensemble tree，分别在每颗决策树沿着特定分支路径最终落入某个叶子结点得到其编号，本质上是这些特征在特定取值区间内的组合。ensemble tree 可以采用 Gbdt 或者 random forest 实现。每一轮迭代，产生一棵新树，最终通过 one-hot encoding 转化为 binary vector，如下图所示。



以下几点是我们在实践中的一些总结和思考。

- 在实验中发现如果将连续值特征进行离散化后喂入 gbdt, gbdt 的效果不佳, AUC 比较低。这是因为 gbdt 本身能很好的处理非线性特征, 使用离散化后的特征反而没什么效果。Xgboost 等树模型无法有效处理高维稀疏特征比如 user id 类特征, 可以采用的替代方式是: 将这类 id 利用一种方式转换为一个或多个新的连续型特征, 然后用于模型训练。
- 需要注意的是当采用叶子结点的 index 作为特征输出需要考虑每棵树的叶子结点并不完全同处于相同深度。
- 实践中采用了 Monte Carlo Search 对 Xgboost 的众多参数进行超参数选择。
- 在离线训练阶段采用基于 Spark 集群的 Xgboost 分布式训练, 而在线预测时则对模型文件直接进行解析, 能够满足线上实时响应的需求。此外, 在实践发现单纯采用 Xgboost 自动学到的高阶组合特征后续输入 LR 模型并不能完全替代人工特征工程的作用; 可以将原始特征以及一些人工组合的高阶交叉特征同 xgboost 学习到的特征组合一起放入后续的模型, 获得更好的效果。

总结

完整的推荐系统是一个庞大的系统, 涉及多个方面, 除了召回、排序、列表生产等步骤外, 还有数据准备与处理, 工程架构与实现, 前端展现等等。在实际中, 通过把这些模块集成在一起, 构成了一个集团通用推荐系统, 对外提供推服务, 应用在 10 多个栏位, 60 多个场景, 取得了很好的效果。本文侧重介绍了召回与排序算法相关的目前已有一些工作与实践, 下一步, 计划引入更多地深度模型来处理召回与排序问题, 并结合在线学习、强化学习、迁移学习等方面的进展, 优化推荐的整体质量。

作者简介

携程基础业务研发部 - 数据产品和服务组，专注于个性化推荐、自然语言处理、图像识别等人工智能领域的先进技术在旅游行业的应用研究并落地产生价值。目前，团队已经为携程提供了通用化的个性化推荐系统、智能客服系统、AI 平台等一系列成熟的产品与服务。

参考文献

- [1] Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 42.8 (2009).
- [2] Sedhain, Suvash, et al. "Autorec: Autoencoders meet collaborative filtering." Proceedings of the 24th International Conference on World Wide Web. ACM, 2015.
- [3] Rendle, Steffen, Christoph Freudenthaler, and Lars Schmidt-Thieme. "Factorizing personalized markov chains for next-basket recommendation." Proceedings of the 19th international conference on World wide web. ACM, 2010.
- [4] Dong, Xin, et al. "A Hybrid Collaborative Filtering Model with Deep Structure for Recommender Systems." AAAI. 2017.
- [5] Hidasi, Balázs, et al. "Session-based recommendations with recurrent neural networks." arXiv preprint arXiv:1511.06939 (2015).
- [6] Zhu, Yu, et al. "What to Do Next: Modeling User Behaviors by Time-LSTM." Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17. 2017.
- [7] Barkan, Oren, and Noam Koenigstein. "Item2vec: neural item embedding for collaborative filtering." Machine Learning for Signal Processing (MLSP), 2016 IEEE 26th International Workshop on. IEEE, 2016.
- [8] Wang, Hao, Naiyan Wang, and Dit-Yan Yeung. "Collaborative deep learning for recommender systems." Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015.
- [9] Covington, Paul, Jay Adams, and Emre Sargin. "Deep neural networks for youtube recommendations." Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016.
- [10] Kim, Donghyun, et al. "Convolutional matrix factorization for document context-aware recommendation." Proceedings of the 10th ACM Conference on

Recommender Systems. ACM, 2016.

[11] <https://mli.github.io/2013/03/24/the-end-of-feature-engineering-and-linear-model/>

[12] Richardson, Matthew, Ewa Dominowska, and Robert Ragno. "Predicting clicks: estimating the click-through rate for new ads." Proceedings of the 16th international conference on World Wide Web. ACM, 2007

[13] Andrew, Galen, and Jianfeng Gao. "Scalable training of L 1-regularized log-linear models." Proceedings of the 24th international conference on Machine learning. ACM, 2007.

[14] Graepel, Thore, et al. "Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft's bing search engine." Omnipress, 2010.

[15] McMahan, H. Brendan, et al. "Ad click prediction: a view from the trenches." Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013.

[16] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, 2016.

[17] Rendle, Steffen. "Factorization machines." Data Mining (ICDM), 2010 IEEE 10th International Conference on. IEEE, 2010.

[18] Juan, Yuchin, et al. "Field-aware factorization machines for CTR prediction." Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016.

[19] Gai, Kun, et al. "Learning Piece-wise Linear Models from Large Scale Data for Ad Click Prediction." arXiv preprint arXiv:1704.05194 (2017).

[20] He, Xinran, et al. "Practical lessons from predicting clicks on ads at facebook." Proceedings of the Eighth International Workshop on Data Mining for Online Advertising. ACM, 2014.

[21] Cheng, Heng-Tze, et al. "Wide & deep learning for recommender systems." Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. ACM, 2016.

[22] Guo, Huifang, et al. "DeepFM: A Factorization-Machine based Neural Network for CTR Prediction." arXiv preprint arXiv:1703.04247 (2017).

[23] Zhou, Guorui, et al. "Deep Interest Network for Click-Through Rate

-
- Prediction." arXiv preprint arXiv:1706.06978 (2017).
- [24] Blondel, Mathieu, et al. "Higher-order factorization machines." Advances in Neural Information Processing Systems. 2016.
- [25] <http://breezedeus.github.io/2014/11/20/breezedeus-feature-hashing.html>
- [26] https://en.wikipedia.org/wiki/Categorical_variable
- [27] <https://www.zhihu.com/question/48674426>

阿里数据库进入全网秒级实时监控时代

作者 吴必良（未立）



AI 前线导读

2017 双 11 再次创下了 32.5 万笔 / 秒交易创建的纪录，在这个数字后面，更是每秒多达几千万次的数据库写入，如何大规模进行自动化操作、保证数据库的稳定性、快速发现问题是一个巨大的难题，这也是数据库管控平台要完成的任务。

随着阿里巴巴数据库规模的不断扩大，我们建设数据库管控平台也经历了很多阶段，从脚本化、工具化、平台化到目前的 DBPaaS，DBPaaS 在今年双 11 中，首次全面覆盖集团、各子公司下的本地数据库、公有

云、混合云等多种场景。今年双 11，数据库已经全面实现容器化部署，弹性使用离线资源、公有云资源支持大促。全面优化的监控采集链路，实现了全网所有数据库实例的秒级采集、监控、展现、诊断。每秒实时处理超过 1000 万项监控指标，让异常无所遁形。DBPaaS 也持续在数据库管理的自动化、规模化、数字化、智能化等方向进行突破。

在这其中，关于数据库监控系统建设比较典型。

在业务平时运行态，线上系统出现故障，在数万数据库中，如何发现异常、快速诊断亦是一件非常具有挑战的事情。在双十一全链路压测中，系统吞吐量未达预期或业务出现了 RT 抖动，快速诊断定位数据库问题是一个现实课题。此外，对于复杂数据库故障事后排查故障根源、现场还原、历史事件追踪也迫使我们建设一个覆盖线上所有环境、数据库实例、事件的监控系统，

做到：

1. 覆盖阿里全球子公司所有机房。
2. 覆盖阿里生态包含新零售、新金融、新制造、新技术、新能源所有业务。
3. 覆盖所有数据库主机、操作系统、容器、数据库、网络。
4. 所有性能指标做到 1 秒级连续不间断监控。
5. 全天候持续稳定运行。

DBPaaS 监控双 11 运行概况

2017 年双 11，DBPaaS 平台秒级监控系统每秒平均处理 1000 万项性能指标，峰值处理 1400 万项性能指标，为线上分布在中国、美国、欧洲、东南亚的、所有数据库实例健康运行保驾护航。做到了实时秒级监控，也就是说，任何时候，DBA 同学可以看到任何数据库实例一秒以前的所有性能趋势。

DBPaaS 监控系统仅使用 0.5% 的数据库资源池的机器，支撑整个采

集链路、计算链路、存储、展现诊断系统。监控系统完美记录今年每一次全链路压测每个 RT 抖动现场，助力 DBA 快速诊断数据库问题，并为后续系统优化提供建议。

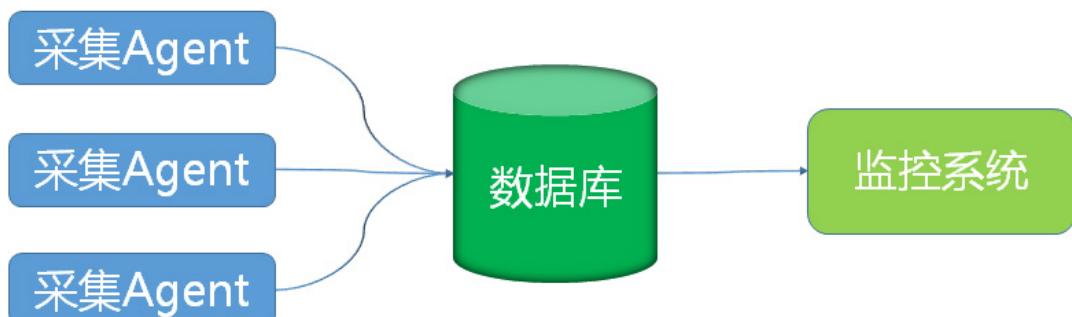
在双 11 大促保障期间，我们做到机器不扩容、服务不降级，让 DBA 同学们喝茶度过双 11。在日常业务运行保障，我们也具备 7*24 服务能力。

我们是如何做到的

实现一个支持数万数据库实例的实时秒级监控系统，要解决许多技术挑战。都说优秀的架构是演进过来，监控系统的建设也随着规模和复杂性增加不断迭代，到 2017 年，监控系统经历了四个阶段改进。

第一代监控系统

第一代监控系统架构非常简单，采集 Agent 直接把性能数据写入数据库，监控系统直接查询数据库即可。



随着数据库集群规模扩大，简易架构的缺点也非常明显。

首先，单机数据库容量扩展性不足，随着监控的数据库规模扩大，日常性能指标写入量非常大，数据库容量捉襟见肘，长时间积累的监控历史数据经常触发磁盘空间预警，我们经常被迫删除远期数据。

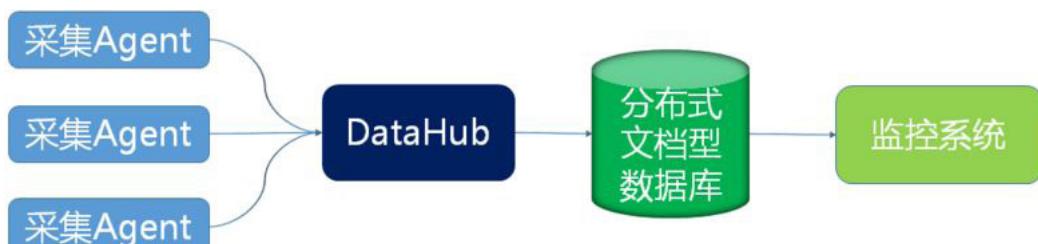
其次，监控指标的扩展性不足。一开始数据库监控项只有十几项，但是很快就发现不够用。因为经常有人拿着 MySQL 的文档说，我想看这个，我想看那个，能不能放到监控系统里。性能指标展现的前提是存储，

在存储层的扩展性缺陷让我们头痛不已。对于这种功能需求，无论是宽表还是窄表，都存在明显的缺陷。如果用宽表，每新增一批性能指标，就要执行一次 DDL，虽然预定义扩展字段可以缓解，但终究约束了产品想象空间。窄表在结构上解决了任意个性能指标的存储问题，但是它也带来了写入数据量放大和存储空间膨胀的弊病。最后，系统整体读写能力也不高，而且不具备水平扩展性。

以上所有原因催生了第二代监控系统的诞生。

第二代监控系统

第二代监控系统引入了 DataHub 模块和分布式文档数据库。数据链路变成由采集 Agent 到 DataHub 到分布式文档数据库，监控系统从分布式文档。



采集 Agent 专注于性能数据采集逻辑，构造统一数据格式，调用 DataHub 接口把数据传输到 DataHub，采集 Agent 不需要关心性能数据存在哪里。DataHub 作为承上启下的节点，实现了采集与存储的解耦。第一，它对采集 Agent 屏蔽了数据存储细节，仅暴露最简单数据投递接口；第二，DataHub 收到根据存储引擎特性使用最优写入模型，比如使用批量写入、压缩等方式；第三，使用 LVS、LSB 技术可以实现 DataHub 水平扩展。分布式文档数据库部分了解决扩展性问题，水平扩容用于解决存储容量不足的问题，schema free 的特性可以性能指标扩展性问题。

随着监控系统持续运行，数据库实例规模扩大，性能指标持续增加，监控系统用户扩大，又遇到新的问题。第一，DBA 同学常常需要查看数据库跨越数月的性能趋势，以预估数据库流量未来趋势，这时系统查询速度

基本不可用。第二，存储长达一年的全量性能数据，成本变得越来越不可承受，每年双 11 压测时，DBA 同学总会问起去年双 11 的性能趋势。第三，DataHub 存在丢失采集数据的隐患，由于采集原始数据是先 buffer 在 DataHub 内存中，只要进程重启，内存中的采集数据就会丢失。

第三代监控系统

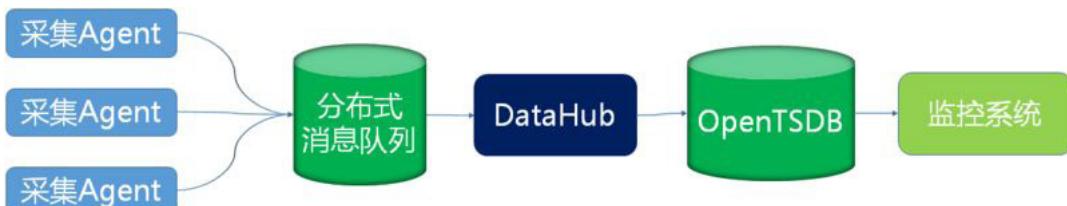
关于查询速度慢的问题，文档型数据库和关系型数据库一样，都是面向行的数据库，即读写的基本数据，每一秒的性能数据存储一行，一行 N 个性能指标，性能指标被存储在以时间为 key 的一个表格中。虽然同一时刻的所有性能指标被存在同一行，但是它们的关系却没那么紧密。因为典型的监控诊断需求是查同一个或几个指标在一段时间的变化趋势，而不是查同一时刻的指标（瞬时值），比如这样的：

时间	指标1	指标2	指标3	指标4	指标5	指标6	指标7	指标8	指标9
t0	1	3	1	6	1	0	1	3	1
t1	2	1	2	5	2	1	2	1	2
t2	3	2	1	3	1	0	1	2	1
t3	1	3	2	1	2	1	2	2	1
t4	2	1	2	2	1	0	1	0	1
t5	2	2	1	2	2	1	2	1	2
t6	2	1	2	1	3	2	3	2	3
t7	3	2	3	2	5	3	1	0	1
t8	1	2	1	2	2	3	3	2	1

数据库存储引擎为了查出某个指标的性能趋势，却要扫描所有指标的数据，CPU 和内存都开销巨大，显而易见，这些都是在浪费。虽然 Column Family 技术可以在一定程度上缓解上面说的问题，但是如何设定 Column Family 是个巨大挑战，难道要存储层的策略要和监控诊断层的需求耦合吗？这看起来不是好办法。

所以，我们把目光投向列式数据库，监控性能指标读写特征非常合适列式数据库，以 OpenTSDB 为代表的时序数据库，进入我们考察视野。

OpenTSDB 用时间线来描述每一个带有时间序列的特定对象，时间线的读写都是独立的。毫无疑问，OpenTSDB 成为第三代监控系统架构的一部分。



为了消除 DataHub 稳定性隐患，引入分布式消息队列，起到削峰填谷作用，即使 DataHub 全线崩溃，也可以采用重新消费消息的方式解决。分布式消息队列，可以选择 Kafka 或 RocketMQ，这些分布式消息队列已经具备了高可用能力。

第三代架构相比过去有巨大的进步，在 2016 年双 11 实现了全网数据库 10 秒级监控，核心数据库集群 1 秒级监控。

随着阿里生态扩大，全球化深入，各类全资子公司业务全面融合到阿里体系，除了中国大陆，还有美国、欧洲、俄罗斯、东南亚的业务。同时在阿里数据库领域的技术应用层出不穷，单元化部署已经成为常态，容器化调度正在覆盖全网，存储计算分离正在不断推进，同一个业务数据库集群，在不同单元的部署策略可能也不同。与之对应的，DBA 团队的规模并没有相应扩大，一个 DBA 同学支持多个子公司业务是常态，有的 DBA 还要兼任新技术推广等工作。在数据库性能诊断这个环节，必须为 DBA 争效率，为 DBA 提供从宏观到微观到诊断路径显得越来越迫切：从大盘到集群、到单元、到实例、到主机、容器等一站式服务。

在这样的诊断需求下，第三代监控架构有点力不从心了，主要表现在查询：

1. 高维度的性能诊断查询速度慢，以集群 QPS 为例，由于 OpenTSDB 里存储的每一个实例的 QPS 数据，当需要查询集群维度 QPS 就需要对扫描集群每一个实例的 QPS，再 group by 时间

戳 sum 所有实例 QPS。这需要扫描大量原始数据。

2. OpenTSDB 无法支持复杂的监控需求，比如查看集群平均 RT 趋势，集群平均 RT 并不是 avg(所有实例的 RT)，而是 sum(执行时间)/sum(执行次数)。为了实现目标只能查出 2 条时间线数据，在监控系统内部计算完后再展现在页面中，用户响应时间太长。
3. 长时间跨度的性能诊断速度慢，比如 1 个月的性能趋势，需要扫描原始的秒级 2592000 个数据点到浏览器中展现，考虑到浏览器展现性能，实际并不能也没必要展现原始秒级数据。展示 15 分钟时间精度的数据就够了。

上述提到的预算问题，OpenTSDB 也意识到，其 2.4 版本开始，具备了简陋预算能力，无论从功能灵活性还是系统稳定性、性能，OpenTSDB 都无法满足 DBPaaS 秒级监控需求。

DBPaaS 新一代架构

新一代架构，我们把 OpenTSDB 升级为更强劲的 HiTSDB，同时基于流式计算开发的实时预聚合引擎代替简单的 DataHub，让秒级监控飞。

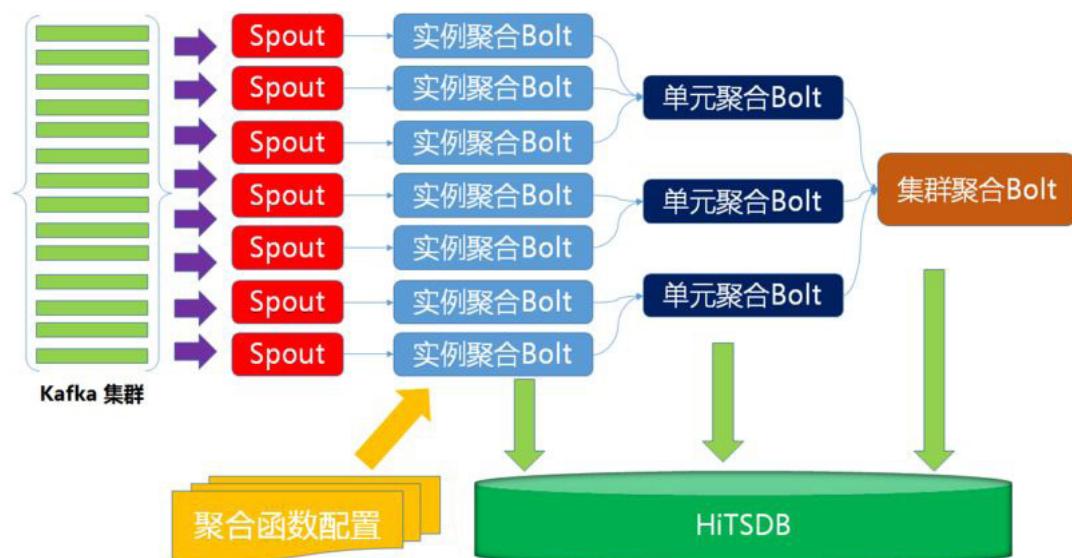


在职责界定上，监控诊断需求的复杂性留给实时预聚合引擎来解决，对时序数据库的查询需求都限定在一条时间线内。这要求时序数据库必须把单一时间线性能做到极致，由兄弟团队开发的阿里巴巴高性能时序数据库 HiTSDB 做到了极致压缩和极致读写能力，利用时序数据等距时间戳和数值小幅变化的特征，它做了大量压缩。同时它全面兼容 OpenTSDB 协议，已经在阿里云公测。

新架构让我们放开双手专注思考监控与诊断需求，不再受存储层的束缚。第一，为了高维度性能趋势查询性能，预聚合引擎做到了预先按业务数据库集群、单元、实例把性能指标计算好，写入 HiTSDB。第二，建立性能指标聚合计算函数库，所有性能指标的聚合计算公式都是可以配置的，实现了自由的设定监控指标。第三，事先降时间精度，分为 6 个精度：1 秒、5 秒、15 秒、1 分钟、5 分钟、15 分钟。不同时间精度的性能数据，才有不同的压缩策略。

实时计算引擎

实时计算引擎实现了实例、单元、集群三个维度逐级聚合，每一级聚合 Bolt 各自写入 HiTSDB。流式计算平台的选择是自由，目前我们的程序运行在 JStorm 计算平台上，JStorm 让我们具备天生的高可用能力。



实时计算引擎性能

实时计算引擎使用了数据库总机器规模 0.1% 的资源，实现了全网秒级监控数据的计算，平均每秒处理超过 1000 万项性能指标，平均写入 TPS 600 万，峰值 TPS 1400 万，下图是双 11 期间 HiTSDB TPS 趋势曲线。



关键优化点

用这么少的计算资源就实现了这么高吞吐量，必然用上了许多黑科技。

1. 在预算算中，我们使用增量迭代计算，无论是 5 秒精度的数据，还是 15 分钟精度数据，我们不需要等时间窗口内所有的性能指标收集满了，再开始计算，而是来多少性能数据，就算多少，仅保留中间结果，极大的节省内存。这项优化，相比常规计算方法至少节省 95% 内存。
2. 采集端，针对性能数据报文进行合并，把相似和相邻的报文合并在一起上报到 kafka，这样可以让 JStorm 程序批量处理数据。
3. 利用流式计算的特性实现数据局部性，同一个集群单元的实例采集到的数据在同一个 kafka 分区。这样可以减少计算过程的网络传输及 java 序列化 / 反序列化。这一项可以减少 50% 的网络传输。有兴趣的朋友可以想想为什么不能按实例分区或按集群分区，会有什么问题呢？
4. 使用 JStorm 自定义调度特性，让具有数据相关性的计算 Bolt 调度在同一个 JVM 中，这个是为了配合上面第二步，实现数据流转尽量发生在同一个 JVM 里。
5. 对于不得不发生的 Map–Reduce 数据传输，尽量使用批量传输，并对传输的数据结构进行复用、裁剪，少传输重复数据，减少序列化、反序列化压力。

未来展望

阿里 DBPaaS 全网秒级监控让数据库管控实现了数字化，经过这一年，我们积累了许多有价值的结构化数据。随着大数据技术、机器学习技术的发展，为数据库管控进入智能化提供了可能性。

1. 智能诊断，基于现有全方位无死角的监控，结合事件追踪，智能定位问题。
2. 调度优化，通过分析每个数据库实例的画像特征，让资源互补性的几个数据库实例调度在一起，最终节省成本。
3. 预算估计，通过分析数据库历史运行状况，在每次大促前，根据业务交易量目标，确定每一个数据库集群容量需求，进而为自动化扩容提供依据。

伯克利团队解读未来 AI 系统面临的挑战和机会

作者 马卓奇



摘要：这篇立场论文解读了伯克利AI团队对于未来10年AI系统的研究方向的观点。人工智能在过去二十年内取得了显著的进展，带来了一场“完美风暴”，而它成功的背后离不开：（1）海量的数据，（2）可扩展的计算机和软件系统，（3）先进技术的可及性。不过，我们仍然需要人工智能系统能够在不可预知的环境中做出及时、安全的决策，对复杂的对抗样本具有强大的鲁棒性，并且可以在不损害机密性的情况下处理跨组织和个人不断增加的数据。随着硬件发展逐渐走向摩尔定律的末端，目前的技术可以存储和处理的数据量最终会受到限制，这将加剧对AI系统的挑战。本文针对系统、架构和安全领域，提出了9个未来AI的开放性研究方

向。

AI 系统发展趋势与挑战

关键任务AI系统

趋势：AI推动了越来越多的关键任务在生活中的应用，例如自动驾驶、机械辅助手术、家庭自动化，与人类的福祉和生命息息相关。

挑战：AI系统需要通过与动态环境交互持续学习，并且做出及时、鲁棒，以及安全的决策。

个性化AI系统

趋势：从虚拟助理、自动驾驶到政治运动，为用户提供量身定做的决策正日益成为AI系统设计的关注焦点。个性化AI系统要考虑用户的行为和喜好。

挑战：设计能够提供个性化应用程序和服务的AI系统，但不能损害用户的隐私和安全性。

跨组织AI系统

趋势：越来越多的组织在利用第三方数据来增强他们的人工智能服务。例如医院共享数据以防止疫情爆发，金融机构共享数据以提高防欺诈能力。这种应用场景的普及将带来从数据仓库（一个公司收集数据，处理数据，并提供服务）到数据生态系统（AI应用可以使用不同组织拥有的数据进行学习和决策）的过渡。

挑战：设计出能够在由不同组织所拥有的数据集上进行训练的AI系统，而不影响组织之间的数据保密性，并且在这个过程中能够跨越组织之间的潜在竞争障碍。

AI的需求超过摩尔定律

趋势：能够处理和存储海量数据是AI取得成功的一个重要前提，然而技术的发展将越来越难追赶上数据产生的速度。首先，数据正在持续指数

增长。其次，数据的激增恰巧发生在我们曾经飞速改善的硬件技术面临停滞的时候。

挑战：开发特定领域的架构和软件系统，以满足后摩尔法则时代，未来AI应用程序的性能需求，包括适用于AI工作负载的定制芯片、边缘云系统以有效处理边缘数据，以及简化和采样数据的技术。

针对这些挑战，本文在3个主要领域（动态环境中的行为、安全AI，以及AI特定的体系结构）中确定了9个未来的研究方向（R1–R9）。

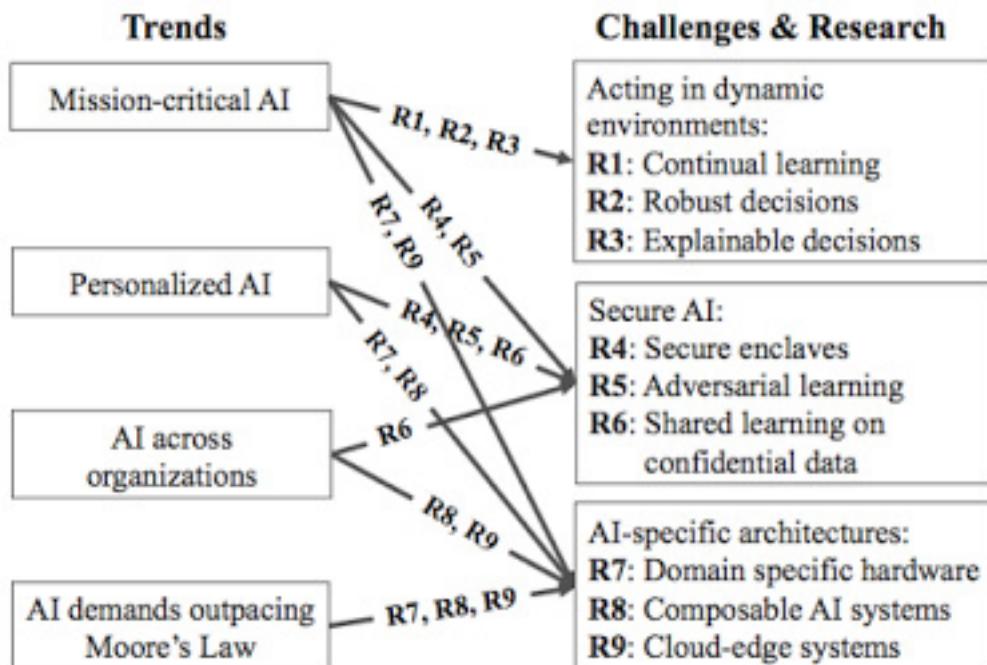


Figure 1: A mapping from trends to challenges and research topics.

趋势、挑战和研究主题之间的关系

动态环境中的行为

未来的大部分AI系统都将在动态环境中运行，这就要求AI系统能够快速安全作出反应，即使对于之前从来没有遇到的场景。

R1：不断学习

在动态环境下学习的AI系统一般使用强化学习（Reinforcement Learning, RL）框架。尽管最近强化学习与深度神经网络的成功结合开发出了能在多种环境下工作的AI系统（例如AlphaGo打败世界象棋冠军），强化学习并没有得到大规模的实际应用。作者认为，强化学习算法的进步与系统设计的创新结合，将推动新的强化学习应用程序的发展。

研究方向：

1. 构建能够充分利用并行性的强化学习系统，同时允许动态任务图，满足毫秒级延迟，并在严格的要求时间内在异构硬件上运行。
2. 构建能够完全模拟真实环境的系统，因为真实环境会不断产生难以预料的变化，而且运行速度要超过实时。

R2：鲁棒决策

越来越多的AI应用程序正在代替人类做出决策，尤其是在关键任务中。一个重要的标准是它们需要对输入和反馈中的不确定和错误保持鲁棒性。

AI系统中最重要的两个鲁棒性概念是：

1. 在有噪声和对抗反馈的情况下能够进行鲁棒学习。
2. 在不可预见的和对抗输入的情况下给出鲁棒决策。

研究方向：

1. 在AI系统中建立细粒度的源头支持，将结果（例如奖励或状态）变化与引起这些变化的数据源连接起来，并自动学习出因果的、特定于源的噪声模型。
2. 为开发系统设计API和语言支持，使系统能够维护制定决策的置信区间，特别是标记不可预见的输入。

R3：可解释决策

除了进行黑箱预测和决策，AI系统也需要为他们的决策提供人类能够理解的解释。因果推断领域在未来AI系统的应用中是十分重要的，并且该

领域与数据库中的系统诊断和源思想有着自然联系。

研究方向：

1. 构建能够支持交互式诊断分析的AI系统，能够重现之前的运行过程，并能够确定负责特定决策的输入特征，一般方法是通过对之前的扰动输入重新执行决策任务。
2. 为因果推理提供系统支持。

安全 AI

AI系统的安全问题可以分为两类：第一类是攻击者破坏决策过程的完整性。第二类是攻击者学习用于AI系统训练的机密数据，或学习保密模型。

R4：安全飞地

防止这些攻击的方法是提供安全飞地（secure enclaves）。安全飞地是指安全的执行环境，它保护飞地内部运行的应用程序，防止受到飞地外运行的恶意代码的危害。

研究方向：

构建利用安全飞地来确保数据机密性、用户隐私和决策完整性的AI系统，可以通过将AI系统的代码分割为在飞地内运行的最小代码库，以及在飞地以外运行的代码。确保飞地内的代码不会泄露信息，也不会影响决策的完整性。

R5：对抗学习

机器学习算法的自适应性使学习系统易受到新类型的攻击，这些攻击通过恶意改变训练数据或决策输入来破坏决策的完整性。主要有两种类型的攻击：闪避攻击（evasion attack）和数据中毒攻击（data poisoning attack）。

闪避攻击发生在系统推理阶段，攻击者试图产生被学习系统错误分类的数据。数据中毒攻击发生在训练阶段，攻击者将中毒数据（例如错误标

签的数据)注入到训练数据集中，导致学习系统学习出错误模式。

研究方向:

构建在训练和预测阶段对对抗性输入鲁棒的AI系统，可以通过设计新的机器学习模型和网络结构，利用源追踪虚假数据源，并在消除虚假数据源后重新进行决策。

R6：机密数据的共享学习

如今，公司与企业通常各自收集数据、分析数据，并使用这些数据来实现新的特性和产品。然而，并不是所有的组织都拥有与大型AI公司相同数量的数据。我们期望越来越多的组织能够收集有价值的数据，有更多的第三方数据服务可用，并从多个组织的数据中共享学习。

共享学习的主要挑战是如何在跨组织数据上学习模型，同时保证训练过程中不会泄露相关信息。主要有三种方法：

1. 将所有数据汇集到硬盘飞地，然后学习模型。
2. 使用安全多方计算技术（secure multi-party computation）。
3. 使用差分隐私（differential privacy）技术。

研究方向:

构建AI系统：

1. 能够跨数据源学习，同时在训练或测试过程中不泄露数据源的信息。
2. 激励潜在的竞争组织共享他们的数据或模型。

AI 特定的架构

AI系统的需求会驱动系统和硬件架构的创新。这些新架构的目标不只是提升性能，而且要通过提供丰富的、易组合的模块库简化下一代AI应用的开发。

R7：域特定的硬件

在数据持续指数性增长时，40年来一直推动着计算机产业发展的“性

能-成本-能源”技术进步已经接近终点，唯一能够继续改进处理器的方法就是开发域特定的处理器。

研究方向：

1. 设计域特定硬件架构来提升性能，并大幅度降低AI应用的能量消耗，并增强这些应用的安全性。
2. 设计能够利用域特定架构、资源分解架构，以及未来的非易失性存储技术的AI软件系统。

R8：可组合的AI系统

模块化和组合是提高人工智能开发速度和应用的关键，它使AI更容易在复杂系统中集成。

研究方向：

设计能够以模块化、灵活的方式组合模型和动作的AI系统和API，并利用这些API开发丰富的模型库和可选项，以极大地简化AI应用的开发。

R9：云边缘系统

目前大量AI应用服务，例如语音识别和语言翻译，均部署在云上。我们期望未来AI系统的跨度可以连接云和边缘设备。首先，部署在云的AI系统可以将部分功能移至边缘设备以提高安全性、隐私性、低延迟和安全性。其次，部署在边缘的AI系统可以分享数据，并利用云的计算资源来更新模型。

研究方向：

设计云边缘AI系统：

1. 利用边缘降低延迟，提升安全性，实现智能数据保持技术。
2. 利用云在跨边缘设备上分享数据和模型，训练复杂的计算密集型模型，并且采取高质量的决策。

延伸思考

（评论来自纽约州立大学布法洛分校Murat Demirbas教授）

1. 2009年，伯克利发表了一篇类似的关于云计算的立场论文（Above the Clouds: A Berkeley View of Cloud Computing）。这篇论文对云计算思想进行了很好的总结、整理。但是8年过去了，那篇文章中的研究计划进行的并不是很理想。计划是无用的，但计划是必不可少的。学界所感兴趣的区域一直在随时间变化，研究方向也在相应变化。在CS领域，几乎不可能完全计划和管理探索性研究（或许在生物学和科学领域是可能的）。

Table 1: Quick Preview of Top 10 Obstacles to and Opportunities for Growth of Cloud Computing.

	Obstacle	Opportunity
1	Availability of Service	Use Multiple Cloud Providers; Use Elasticity to Prevent DDOS
2	Data Lock-In	Standardize APIs; Compatible SW to enable Surge Computing
3	Data Confidentiality and Auditability	Deploy Encryption, VLANs, Firewalls; Geographical Data Storage
4	Data Transfer Bottlenecks	FedExing Disks; Data Backup/Archival; Higher BW Switches
5	Performance Unpredictability	Improved VM Support; Flash Memory; Gang Schedule VMs
6	Scalable Storage	Invent Scalable Store
7	Bugs in Large Distributed Systems	Invent Debugger that relies on Distributed VMs
8	Scaling Quickly	Invent Auto-Scaler that relies on ML; Snapshots for Conservation
9	Reputation Fate Sharing	Offer reputation-guarding services like those for email
10	Software Licensing	Pay-for-use licenses; Bulk use sales

论文中提出的第4, 5, 6项研究方向进展良好，剩下的进展平淡，项目2和9进展甚微。下面几个研究方向虽然在这份研究计划中没有提及，但它们实实在在重塑了云计算领域的发展进程。

- 云中机器学习工作负载的优势
- 新SQL系统的崛起，一致分布式数据库的增多，协调组、Paxos算法、ZooKeeper服务在云中的重要性
- 开发在线内存数据流和流处理系统，如Spark，来自伯克利
- 通过容器和函数作为服务实现细粒度虚拟化的竞争
- SLA受到更多的重视

即使伯克利提出的AI系统研究计划很有道理，我们还是应该关注未来几年内这些计划的进展，以及AI系统领域会带来怎样令人意想不到的研究机遇。

2. 斯坦福在今年早些时间也发表了一篇类似的立场论文，不过他们的论文是关于机器学习中可复用架构的问题和见解。斯坦福的DAWN项目旨在

建立端到端的机器学习工作流，加入领域专家的力量，并进行端到端的优化。下图总结了他们对于可复用机器学习架构的想法：

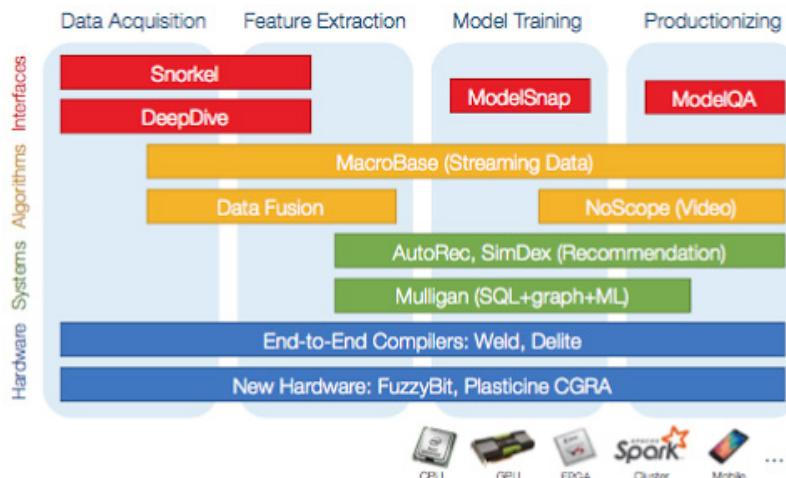


Figure 1: The DAWN Stack for Usable Machine Learning: In the Stanford DAWN project, we are addressing the need for infrastructure for usable ML by building a research stack of software and tools spanning each stage of the ML lifecycle and abstractions from new interfaces to new hardware. We believe this parallel end-to-end and interfaces-to-hardware approach is necessary to fully realize the potential of more usable ML.

当然，这也无可避免地反映了斯坦福团队的优势和弊端：他们更擅长于数据库、数据科学、以及生产方面的研究。看起来与伯克利论文中的“AI特定架构”部分有一些共同点，但是双方针对相同的问题提出了不同的方法。

3. 对于文中提出的R2鲁棒决策这一研究方向，似乎是想说形式化方法——建模、基于不变的推理，是有用的，尤其是当并发控制成为分布式机器学习部署中的一个问题时。

论文原文：[A Berkeley view of systems challenges for AI](#)

参考资料：[Paper summary: A Berkeley view of systems challenges for AI](#)



在微信上关注我们



InfoQ

国内最好的原创技术社区，一线互联网公司核心技术人员提供优质内容。订阅 InfoQ，看全球互联网技术最佳实践。做技术的不会没听过 QCon，不会不知道 InfoQ 吧？——冯大辉从事技术工作，或有兴趣了解 IT 技术行业的朋友，都值得订阅。——曹政



关注「InfoQ」回复“二叉树”，看十位大牛的技术初心，不同圈子程序员的众生相。



聊聊架构

以架构之“道”为基础，呈现更多的务实落地的架构内容。

关注「聊聊架构」
和百位架构师共聊架构



细说云计算

探讨云计算的一切，关注云平台架构、网络、存储与分发。这里有干货，也有闲聊。

关注「细说云计算」
回复“群分享”，
看云计算实践干货分享文章



AI前线

提供最新最全AI领域技术资讯、一线业界实践案例、业界技术分享干货、最新AI论文解读。

关注「AI前线」
回复“AI”，下载《AI前线》
系列迷你书



前端之巅

紧跟前端发展，共享一线技术，不断学习进步，攀登前端之巅。

关注「前端之巅」
回复“京东”，看京东
如何做网站前端监控



移动开发前线

关注移动开发领域最前沿和第一线开发技术，
打造技术分享型社群。

关注「移动开发前线」
回复“群分享”，看移动
开发实践干货文章



高效开发运维

常规运维、亦或是崛起的DevOps，探讨如何
IT交付实现价值。

关注「高效开发运维」
回复“DevOps”，四篇精品
文章领悟DevOps

