第三届 Apache Flink 极客挑战赛暨 AAIG CUP 攻略发布！

原创

Apache Flink 于 2021-09-26 21:30:00 发布 56 收藏
文章标签： 算法 分布式 大数据 编程语言 python

▼ 关注「**Flink 中文社区**」，获取更多技术干货 ▼

摘要：第三届 Apache Flink 极客挑战赛暨 AAIG CUP 自 8 月 17 日上线以来已有 4000+ 参赛队伍报名。针对赛题“电商推荐‘抱大腿’攻击识别”涉及的知识点及相关工具使用已在每周直播课程中分享，本文将通过以下几点对赛题进行详细解读，帮助选手更好的了解赛题核心内容。

Tips：点击「**阅读原文**」即可查看更多技术文章～

 GitHub 地址 

欢迎大家给 Flink 点赞送 star~

一、赛题背景

随着互联网的发展，网购成为越来越多人的选择，据阿里巴巴财报显示，2020 财年阿里巴巴网站成交总额突破一万亿美元，全球年度活跃消费者达 9.60 亿。

为了满足不同用户的个性化需求，电商平台会根据用户的兴趣爱好推荐合适的商品，从而实现商品排序的千人千面需求。推荐系统常见的召回路径有 U2I (User-Item)、I2I (Item-Item) 等。其中，user-to-item 是指通过用户的 profile 信息为用户进行商品的推荐，而 item-to-item 推荐策略则根据用户的商品点击列表为用户推荐关联的商品。

推荐系统的目的是基于不同用户的偏好进行千人千面的推荐。传统的离线推荐系统基于用户历史的行为数据进行加工处理，形成特征样本，然后离线训练模型，并且在线部署进行服务。然而用户的偏好是多元的、用户的行为分布会随着时间而变化，离线的模型无法刻画这种动态的用户偏好，因此需要进行实时的特征更新与模型参数更新，从而能够更好的捕获用户的行为偏好。在推荐场景中，为了更好的提升推荐的时效性与准确性，平台会基于全网的用户行为信息进行实时的 U2I 及 I2I 的更新，并且基于用户最近的行为信息进行相关性的推荐。

为了获取更多的平台流量曝光，将自己的商品展现在更多的消费者面前，部分商家通过 HACK 平台的推荐机制从而增加商品的曝光机会。其中一种典型的手法为“抱大腿”攻击，该方法通过雇佣一批恶意用户协同点击目标商品和爆款商品，从而建立目标商品与爆款商品之间的关联关系，提升目标商品与爆款商品之间的 I2I 关联分。商家通过这种方式诱导用户以爆款的心理预期购买名不符实的商品，不仅损害了消费者的利益，降低其购物体验，还影响了平台和其他商家的信誉，严重扰乱了平台的公平性。因此，我们需要用一个风控系统来过滤掉这些可能的恶意流量，避免它们对推荐系统的模型造成干扰。

由于所有用户行为在输入推荐系统之前，都会首先经过风控系统的过滤，所以如果想要做到推荐系统的实时性，风控系统就必须同样做到实时性。实时拦截此类行为，有助于在保证推荐的时效性的同时，保护实时推荐系统不受恶意攻击影响。

实时风控系统对数据安全的要求较高，如果系统的拦截算法意外泄漏，HACK 平台将得以针对性地加强恶意流量的伪装能力，增大平台监控恶意流量的难度，因此，此类系统有必要部署在加密的可信环境中。

综上所述，为了保障实时推荐系统的准确性，比赛要求选手实现一个保证了数据安全的实时风控系统。

二、数据说明

给定恶意点击、正常点击及对应的“商品”、“用户”相关的属性信息 (用户本地调试可以从网上下载)，选手实现实时的恶意点击识别分类算法，包括模型训练和模型预测。在大赛评测系统中，系统使用 100 万条数据用于模型训练、10 万条数据用于模型预测。另外，比赛提供给选手 50 万条数据的数据集用于算法的本地调试。

比赛提供如下格式的数据用于训练与预测。所有数据均采用 csv 格式保存在文件中，即以下数据格式的各列之间以逗号分隔。每条数据代表一次用户点击商品的行为，它的特征主要来源于其所关联的用户与商品。

uuid	visit_time	user_id	item_id	features	label

uuid：每条数据的 id。该 id 在数据集内具有唯一性。

visit_time：该条行为数据的发生时间。实时预测过程中提供的数据的该值基本是单调递增的。

user_id：该条数据对应的用户的 id。

item_id：该条数据对应的商品的 id。

features：该数据的特征，包含 N 个用空格分隔的浮点数。其中，第 1 ~ M 个数字代表商品的特征，第 M+1 ~ N 个数字代表用户的特征。

label：值为 0 或 1，代表该数据是否为正常行为数据。

训练数据包括上述所有列的数据，预测数据包括除了 label 之外的所有列。

模型文件的输入输出格式

对于只希望在算法层面加以优化的选手，仅需保证保存的模型文件的输入输出为如下格式即可。我们提供的示例镜像的代码能够预处理输入数据的格式，解析 Tensorflow 模型的推理结果，并最终生成符合评测程序要求的 CSV 格式的文件。

预测模型输入 tensor 格式。其中 N 为 feature 的个数。

Tensor("input:0", shape=(?, N), dtype=float32)

预测模型输出 tensor 格式。输出值为 0 或 1，表示输入行为数据是否为恶意行为。

Tensor("output:0", shape=(?, 1), dtype=float32)

三、Demo 解析

本次赛题注重算法和工程的结合，解答赛题大概要经过以下几个阶段：模型训练、模型预测、最优阈值选取、在线预测并判定类别。

模型训练：训练集中的数据都是结构化的，不需要进行特征抽取阶段，可以直接使用模型进行训练。在 demo 里，构建了一个前向反馈网络进行模型的训练，直接拟合样本的标签；

模型预测：为了将训练与预测阶段做到更好的分离，在模型预测阶段，使用的是 cluster serving 的形式，因此预测只需要直接加载训练好的模型，便可以进行预测；

阈值选取：线上使用的是直接判定类别，而不是输出一个概率，这个是非常符合实际业务场景的。但是直接输出类别的情况下，阈值的选取对于模型的线上效果影响特别大，因此需要进行阈值最优选择，找到在验证数据中最优的阈值作为线上判定的阈值，目前 demo 使用的阈值为 0.5；

在线预测并判定类别：在最终输出的时候通过对于当前预测概率与最优阈值的大小，从而确定当前样本的预测类别 (是否作弊)。

四、Demo 优化

实时特征：目前提供的只有用户/商品的偏静态的特征，但是数据中还包括了 用户-商品 的点击关系，用户可以考虑基于点击关系构建实时的特征，比如统计当天截止目前用户/商品的点击量，用户的平均商品点击数、商品的平均用户点击数等；不过需要注意的是，当预测阶段使用了实时特征，则在训练阶段也需要配套相同的实时特征，否则训练与预测使用的特征不一致会导致模型报错或者效果变差的情况；此外，训练集中已经知道哪些商品/用户是有过作弊行为的，这些信息也可以作为模型的特征进行构建；

模型训练：业界有很多成熟的 DNN 模型，目前 demo 使用了 3 层的结构，选手可以考虑使用更复杂的模型进行训练，从而达到更好的拟合效果；此外，我们不应该局限于某个 "超级模型"，而是可以考虑基于集成学习的方式混合多个模型/策略进行预测。

最优阈值选择：目前 demo 中使用的阈值为 0.5，但是最优阈值选取需要基于模型的在验证集中的预测情况进行选择，其实我们可以写一个脚本，通过验证集找到最优的阈值；

在线预测：线上 demo 模型对于全部的流数据均会进行预测，然而一但出现某个样本的预测出现高延迟，可能会导致后续的样本预测也会出现连带的延迟，从而导致整体线上延迟严重。除了优化算法与工程、尽量降低延迟之外，选手也可以尝试对延迟进行监控，以缓解长尾现象的影响。

五、评分指标

选手提交结果的分数由两方面评分的乘积来决定，两方面分别代表选手提交结果的算法与工程方面的表现。用一个公式表示即如下所示：

score=F1 ∗valid_latency

在算法方面，比赛根据推理结果的 F1 参数来评分，即推理结果的准确率与召回率的调和平均数。

在工程方面，由于比赛模拟实时风控场景，所以比赛对实时推理过程中的延迟做出限制。选手的程序需要为 Kafka 中出现的实时数据流提供推理服务，并在数据流的流量不超过给定阈值的情况下，单条数据的延迟不超过 500ms。

选手部署的推理服务需要从 Kafka 中读取待推理数据，并将推理结果写入 Kafka。数据的延迟的定义即为待推理数据及其推理结果在 Kafka 中的时间戳的差值。上述公式中的 valid_latency，即为延迟符合要求的数据占所有数据的比例。延迟超过 500ms 的数据不仅会影响到 valid_latency 的值，进而影响到分数，而且也不会参与 F1 参数的计算过程。

六、技术介绍

Apache Flink 是一个在无界和有界数据流上进行状态计算的框架和分布式处理引擎。Flink 已经可以在所有常见的集群环境中运行，并以 in-memory 的速度和任意的规模进行计算。

在 Flink 的基础上，Flink AI Flow 作为兼顾流计算的大数据 + AI 顶层工作流抽象和配套服务，提供了机器学习的端到端解决方案。

Analytics Zoo 及 BigDL 是英特尔®开源的统一大数据分析和 AI 平台，支持分布式 TensorFlow 及 PyTorch 的训练和推理，通过 OpenVINO 工具套件和 DL Boost 指令集等，提升深度学习工作负载的性能。Cluster Serving 是 Analytics Zoo/BigDL 的分布式推理解决方案，可以部署在 Apache Flink 集群上进行分布式运算。

Occlum 是蚂蚁集团基于 Intel SGX 的开源 LibOS，使得 Linux 应用程序在只修改少量代码或者完全不修改代码的情况下运行于 Enclave 安全环境中，保证数据处于加密和强隔离状态，确保数据安全与用户隐私。

参考资料

基础镜像使用说明与相关技术介绍：

https://code.aliyun.com/flink-tianchi/antispam-2021/tree/master

Flink 1.11 中文文档：

https://ci.apache.org/projects/flink/flink-docs-release-1.11/zh/

Flink AI Flow Wiki：

https://github.com/alibaba/flink-ai-extended/wiki

Analytics Zoo Cluster Serving Programming Guide：

https://github.com/intel-analytics/analytics-zoo/blob/master/docs/docs/ClusterServingGuide/ProgrammingGuide.md

Occlum Github Repo：

https://github.com/occlum/occlum

学习资料

学习论坛：

https://tianchi.aliyun.com/competition/entrance/531925/forum

学习视频：

https://flink-learning.org.cn/activity/detail/99fac57d602922669b0ad11eecd5df01

大赛答疑交流钉钉群：35732860

热点推荐

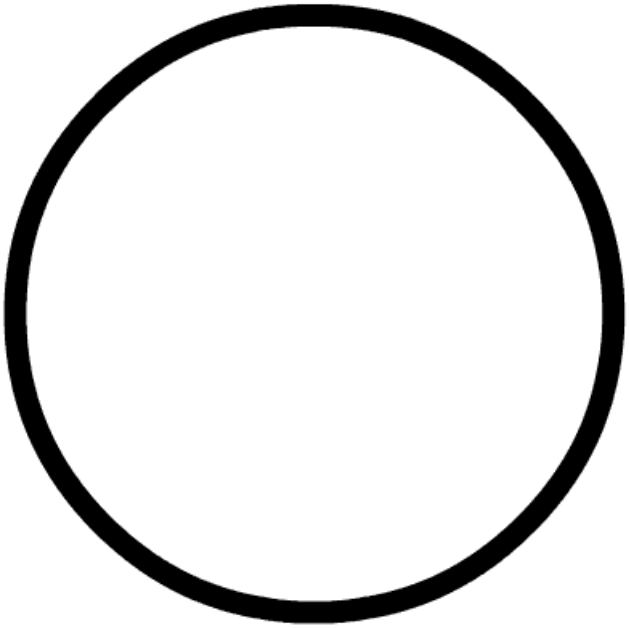
Flink Forward Asia 2021 正式启动！议题火热征集中！

第三届 Apache Flink 极客挑战赛暨 AAIG CUP：Cluster Serving 概况

Flink 1.14 新特性预览

更多 Flink 相关技术问题，可扫码加入社区钉钉交流群～

【③群】 Apache Flink C... 全员



戳我，查看更多技术文章！