支付风控系统设计：风控数据仓库建设（二）

这篇文章是支付风控系统设计的第二篇，重点介绍支持支付风控的数据仓库建设。关于支付系统在风控上的具体需求，可参见上一篇文章 《支付风控场景分析》。

支付风控系统在数据存储设计上和其它业务不同的地方在于数据获取与使用的流程。一般业务系统会先确定系统数据需求，再设计如何在业务流程中采集数据，以及数据的格式怎么定义。而支付风控面临的是一个无法预知的场景，需要在实践中根据当前运行情况不断调整。它会先把数据采集过来，之后才能从中发现可能存在的问题，并针对该问题制订风控规则。也就是风控是先采集数据，再使用数据。

风控分析不仅要看交易数据，还得研究所有相关联的数据，这才能全面分析出来风险的根源，推断出需要采取的措施。因而数据采集工作对风控系统建设和演化是非常重要的。本文分析风控所需要的数据，如何采集和存储数据，建立支持风控的数据仓库。

一、数据来源

一笔交易的风险等级的计算需要考虑到多个维度。未成年人购买高档酒、促销期间羊毛客刷单、在洗钱高发地区的商户销售的物品成交价格远超实际价格。这些可疑交易的识别，仅依靠支付系统本身是无法完成的。用户的年龄、商品特点(是否高档酒)、是否促销、羊毛号的识别等，需要从各业务系统，甚至公司外部收集和用户、商品、商家、地区、手机号相关的数据，通过对这些数据进行分析，提取特征，识别潜在的风险。

1. 内部数据

风控几乎需要收集所有相关系统的数据。 用户系统需采集用户的静态信息，姓名、性别、年龄等。风控系统不仅仅关注这些静态信息，还需要重点关注用户的行为信息，包括注册、密码修改、修改个人信息等操作，需要收集这些操作的时间、地点、设备等信息。 此外，用户之间的关系，也是风控系统需要关注的数据。

商户系统：除了采集机构的基本信息，如成立时间、注册时间、人员规模、营业额、销售额、经营范围、注册地点等， 还需要考虑到该商户关联的用户，包括法人代表、公司组织结构、主要员工信息等。
商品系统：商品的静态信息，包括类型、价格、上架时间、库存等信息； 商品的浏览、放入购物车、购买、评论、退货等用户操作，包括这些操作的时间、地点、设备等信息。

社交数据，包括评论、论坛、留言等。
业务系统，如视频系统中的观影记录、类型偏好、时间、地点、设备等信息。

当然，支付数据是风控最重要基础数据。用户在支付系统中涉及到的数据都需要收集整理来支持风控分析。包括但不限于账户数据、订单数据、交易数据、优惠券数据和账务流水等。这些数据在支付数据库中也存在，风控所需要的数据和业务数据略有不同。除了业务数据外，风控还关心如下数据：

用户当前上下文环境，包括用户所用设备的类型、操作系统、IP 地址、设备 ID、所在地等，而这些数据往往并不是业务所关心的。而且记录太多的上下文数据也影响性能。
账户，订单等操作实体的状态。在业务数据库中一般仅保留实体的最终状态，比如账户是否已锁定、订单是否已支付等。 而风控需要关心这些状态变更的时机，以及变更的时间间隔。例如，用户频繁更改交易密码，超正常频率提交订单等，就不是一个正常的状态。

这些数据一般可以从日志中采集。

2. 外部数据

对于大部分业务单一和用户量不大的公司来说，其数据有限而且单一，需要使用外部数据来辅助完成风控计算。

常用的外部数据包括：

公安部的实名认证数据，包括用户姓名、身份证号信息；
央行发布的各种名单，如洗钱区域，恐怖组织名单等。
央行信用报告，这个查询可是要真金白银的。
微博数据，一个人经常了解如何养卡，套现等内容并不是太好的事情。
工商局提供的公司信息。
招聘网站上的公司招聘信息。公司一直有招聘说明业务还不错。
芝麻信用，这个需要申请。

二、采集方式

一般来说，风控的非实时数据采集，不能直接从线上的数据库中读取，这会把数据库打死。主要的数据采集方式有从库采集，日志采集和 pingback 三种方式。

1. 数据库从库

主流数据库，如 Hbase，Mysql 都提供同步数据进从库的功能，读取从库不会影响主库操作。但如上所述，采用从库有如下问题：

分析所需数据和业务数据不同，还需要从其他途径补充数据。
将风控所需数据和业务数据紧耦合起来了。一旦业务有变更，风控系统也需要调整。

2. 日志

这是风控数据采集的主要方式。 业务方可以将风控所需要的数据输出到日志中，风控系统对接日志来异步采集数据。这使得数据采集不会影响业务处理主流程。 这种方式风险在于：

需要规范日志的格式，否则每个系统一套日志格式，会导致对接工作量巨大。
保持日志的稳定性。一旦代码被修改，打印日志的代码被删除了，会导致日志数据无法采集的风险。
需要注意日志采集系统的可靠性。目前主流的采集框架都有可能会丢失日志。虽然从我们使用的情况来还未发生这种事情，但不排除这个风险。

从技术上来说，日志采集的框架主要框架有

ELK（Elastic + Logstash + Kibana）， Logstash 驻留在日志输出端采集日志，并发送到 Elastic 服务器上。 Kibana 则是一个日志分析的工具；
Flume + Kafka + Elastic 。 通过 Flume 进行采集，输出到 Kafka，汇总到 Elastic 进行存储。日志分析可以在 Elastic 上离线非实时进行，也可以直接对接 Kafka 准实时分析，即流处理。 使用 Storm 或者 Spark 都可以。

3. pingback

Pingback 指在页面上埋入脚本来监测用户的操作，特别是点击操作和键盘操作，将检测到的用户行为异步发送到服务器端。这可以侦测到用户在页面停留时间，鼠标点击的区域等信息，由此可以推断用户偏好，情绪等信息。 pingback 的挑战在于如何在服务器端应对流量洪峰。pingback 数据一般不直接入库，可以先写入 Kafka，风控系统对接 Kafka 来分析 pingback 数据。

三、数据特征

用于支持风控计算的最终数据，在静态与动态数据为基础计算出来的带置信度的推算数据为主的离散数据，有点绕口，我们详细分析下这里涉及到的几个概念，来说明最终用来支持风控计算的数据有什么特征。

1. 静态数据与动态数据

上述采集到的数据，大部分是静态数据。也就是这些数据一旦产生，一般不会被修改。但在分析时，还需要一些易变的动态数据来，比如用户的 年龄，每天的访问量，每天消费金额等。

2. 原始数据与推算数据

不管静态还是动态数据，他们都是从用户输入或者系统采集的方式产生。但我们知道，互联网的数据可靠性是有问题的。网上千娇百媚的姑娘，在现实中可能是一位抠脚大汉。虽然系统中设计了复杂的表格来收集用户信息，但会提供全部信息的用户还是很少，大家对隐私内容还是捂得很紧。

所以，在进行风险计算前，还需要对数据进行验证和补充。这都需要借助其他数据来进行推算，这些数据被称为推算数据。推算数据和原始数据不同之处在于它会有多个可能取值，每个值都带有置信度。完全可信为 100%，不可信为 0。置信度总和为 1。比如正常情况下，用户的性别要么男，要么女。假如有个用户注册时选择性别女，但经常买刮胡刀，衬衣，没有买过女性用品，那实际性别为男的置信度就非常高。

3. 离散数据与连续数据

这是从属性值的取值范围来评估。比如用户每天的订单额，一般来说是连续分布的。而性别，职业，爱好等，是离散值。一般来说，离散值更容易做分析处理，刻画特征，所以在分析前，需要对连续数值做离散化处理。

四、名单数据

名单数据是支付风控数据仓库中最重要的内容。 风控系统数据仓库建设，也一般都从名单数据开始。 名单加上简单的拦截规则，已经可以解决绝大部分风控的问题。就算在更先进的风控系统中，名单仍然是风控中的基础数据。在评估事件风险时，名单往往是用来执行第一道拦截时所用的数据。比如用户交易时使用的手机是黑名单中的手机，则必须终止本次交易。

1. 黑白灰名单

大家都熟知黑名单与白名单，一个是必须阻止，一个是必须放行。 除此之外，还有灰名单。灰名单用于对一些高风险的用户进行监控。 这些用户的行为不是直接阻止，而是延迟交易，经人工确认无问题后再放行。

2. 更新周期

相对其它数据来说，名单数据的更新频率不高，按天、周、月更新都有，很少有需要实时更新的内容。对于手机号，证件号等名单，一般可以采取人工更新的策略。每天评估风控数据，对确认有问题的号码，加入到黑名单中。如果采用的是第三方名单，则需要按照第三方的要求对名单做更新。

3. 名单列表

一般来说，风控系统需要配置的名单列表有：

（1）个人名单

如下名单是必备的（后续会及时更新）：

央行的反洗钱恐怖分子名单
公安部的通缉犯名单
全国法院失信被执行人名单信息公布与查询

（2）IP 名单

没有权威的 IP 名单。这需要在运行中积累。建立 IP 名单需要注意如下事项：公司内部 IP，合作伙伴 IP 可以列入白名单列表；手机运营商的 IP 也要做到白名单中，封一个 IP 等于封掉一大批手机号；代理服务器可以列入灰名单；访问量大的 IP 也可能大公司的外网 IP，不能仅依赖访问量来识别黑 IP。

（3）公司名单

必备名单包括央行反洗钱制裁公司名单和工商局失信企业名单

（4）手机号名单

这也没有权威数据，电信运营商也不会提供此类服务。支付宝正在推广这个服务，但还没有公开。黑名单数据需要自主收集。

（5）地域名单

央行公布的联合国反洗钱地区名单是必须在风控时考虑的名单，其他地域名单也需要自主收集。

（6）协查名单

公检法协查名单，接收到协查请求后，将人员全部信息拉黑。

4. 名单数据存储

名单数据在使用上的特点：

使用频率高，实时性要求高。各种名单匹配基本都需要在线上做实时计算。
数据粒度小，总量大小不一，但存储空间需求都不高。大部分名单都是一些号码表，几个G的空间都能存储。
更新频率低。名单数据一般都比较稳定，按天更新

在使用中，名单数据一般直接存储在内存中，或者使用内存数据库（Redis，Couchbase）。关系型数据库可以用来保存名单数据，但不会直接被线上应用所访问，它无法满足高访问量的需求。

五、画像数据

名单数据能够快速发现用户在某个维度上的异常行为。在实际使用中，存在过于简单粗暴，一刀切的问题。比如如果限制单次购买金额为5000元，这个规则被试探出来后，攻击者会选择4999元来规避这个限制。画像技术则是尝试从多个维度来评估当前事件的风险。比如画像刻画某用户平时主要在北京地区登录，购买习惯在10~300元之间。某一天突然发生一笔在东莞的4999元额度的消费，那这笔交易就非常可疑了。而这种交易通过规则比较难发现出来。 支付风控涉及的画像包括用户、设备、商品、地域、操作行为等。 这里重点介绍用户、设备和商品的画像。

1. 用户画像（persona）

用户画像是从用户的角度来刻画其背景和行为习惯，为判定某交易的风险等级提供支持。用户画像的内容包括但不限于：

人口信息：一般就叫基本信息，主要包括：姓名、性别、出生日期、出生地、民族、星座

等。
联系方式：家庭地址、工作地址、手机、固定电话、紧急联系人、QQ、微信号等。
资产特征：月工资、年收入、工资外收入、房产、车等
家庭特征：婚姻状况、是否有小孩、小孩关联、家庭成员等
交易偏好：交易频率（总计、年、月、日）、交易金额（总计、年、月、日）、常用账户、交易时间偏好、交易地点偏好、交易所使用设备、交易物品、交易物品所属类别等。
行为特征，这是和业务相关的特征。比如对于电商，关注 用户浏览的物品、浏览的物品类别、购买的物品等。而对于视频网站，则关注用户查看的视频、观影时长、类别偏好、观影地点偏好等信息。

对于已登录用户，可以使用用户 ID 来识别并做画像，但对未登录用户，系统需要通过设备来识别。

2. 设备画像

一个用户配备多台智能设备已经是很常见的事情了。手机，PAD，笔记本，台式机，都是常用的设备。用户在不同的设备上的行为往往是不一样的。有人偏好在电脑上寻找要购买的商品，却最终使用手机来下单，因为手机支付更便捷。 对设备进行画像，和用户画像类似，实际上是刻画使用设备的用户的特征。 此外，对于未登录用户，由于无法标识，也只能通过设备来代表这个用户。设备画像关注如下信息：

设备信息，包括设备类型、型号、屏幕大小、内存大小、CPU 类型、购买时间、购买时价格、现在价格等。
交易偏好，同用户画像；
行为特征，同用户画像。

对设备画像来说，生成一个能唯一识别该设备的标识，即设备指纹，是数据采集中的一个挑战。设备指纹具有如下特点

唯一性，每台机器的指纹都不同，不能重复。
一致性，机器指纹在一台机器上是唯一的，不同应用，不同登录用户中取到的指纹都是一样的。
稳定性，指纹不会随时间变更，不会由于外围设备变更而变更。重装应用，重装操作系统也应该保持不变。

我们将在专门的主题中介绍如何生成设备指纹。

3. 商品画像

商品画像是从商品的角度来刻画购买或者拥有该商品的人的特性。

基本特征：名称，价格，类别，是否虚拟资产，上架时间，下架时间等
促销信息：价格，开始时间，截止时间
购买者特征：偏离这个特征越多，风险越大。购买时间分布，地点分布，价格分布，数量分布，年龄分布，性别分布等。

4. 画像数据存储

画像数据有如下特点：

数据粒度大。一个用户的画像数据，成百上千个维度都正常。
大部分数据都是推算数据，也就是数据格式是带置信度的，比如｛性别： 男，80%；女，20%｝;
每个维度的数据一般最终都需要离散化，比如年龄，虽然 0~150 的取值区间还不算稀疏，一般还会将年龄再分段。
数据量大。考虑到匿名用户和设备，上千万规模的注册用户，匿名用户和设备会在数十亿规模的量级。
数据结构不稳定。 根据业务需要会频繁添加新的数据维度，甚至添加新实体进来。
数据更新频繁。采用推算数据，每天不仅仅要计算新增数据，也需要重新计算现有数据的维度权重。
数据访问频率高。交易时计算权重，也需要使用画像数据。

很难有一个数据库能够同时满足上述的需求。画像数据存储需要综合采用多种数据库来满足不同应用上的需求。

数据写入库， 需要支持数据批量、快速地写入，Hbase 是个不错的选择。
数据读取库，需要支持数据高速读取， couchbase 可以满足这个需求。但 couchbase 不能存储所有数据，这样成本太高。 可以把 couchbase 作为 HBase 的缓存来使用。
写库和读库之间的数据同步。可以根据业务量选取合适的消息队列。每天更新的数据规模在百万及其以下，ActiveMQ 可以满足需求；而上千万的数据，则需要使用 Kafka。

六、知识图谱

画像是从群体和个体的统计角度评估事件的风险，而图谱则更进一步，从关系的角度来评估风险。 知识图谱是由 Google 提出来并应用到搜索引擎上，其后在多个领域都得到很好的应用。 交易是一种社会行为，所以从关系的角度来评估这个行为，能够更精确的了解行为中存在的风险。一个简单的例子，如果发现 A 是高风险的用户，而通过社交图谱分析，发现 A 经常和 B 有交易关系， 那 B 的风险等级也相应地会被调高。

图谱在本质上是一个语义网络， 是一种基于图的数据结构， 它由点和边组成的。点代表一个实体，如人、公司、电话、商品、地址等，边代表实体之间的关系。

如上所示， 如果 A 和 B 两人之间是夫妻关系， 则在图中， A 和 B 分别被用一个节点来标识， 称为实体，他们的关系是 is_wife_of。对电话、出生日期、出生地点、公司等，也可以使用这种方式来表示。 图谱的表达能力，不仅在于描述实体之间的关系，而且通过关系还可以推理出潜在的进一步关系。 比如 A 是 B 的母亲， A 是 C 的妻子， 则有很大的概率可以推断出来 C 是 B 的父亲。 支付风控需要像建立画像一样建立图谱，需要支持包括人，机构，地区，日期，电话，手机号，设备，商品等实体，以及实体之间的关系。图谱数据源也是和画像一样。此外，还有一些互联网数据也有利于建立图谱 百度百科，有很不错的公司，明星，电影，音乐等信息，一般仅限于国内或者中文版本的资料。由于编审并不严谨，数据质量不高。 wiki，有各种语言的版本，提供各种领域的实体，参与的专业人士多，质量较高。 各专业数据库，

知识图谱是基于图的数据结构，它的存储主要是使用图数据库。关系型数据库和 Hbase 等 nosql 数据库在处理图的关系以及关系计算上性能较差，需要专用的图数据库，当前主要的图数据库有 neo4j，Titan,Jena 等。neo4j 是使用最多的图数据库，而且可以和 spark graph 集成，方便对图谱数据做处理。

七、总结

总结一下，本文将风控系统所需要的数据分为名单、画像和图谱三个主题，这三个主题也对应了风控系统发展的不同的阶段。这里列出了每个阶段所需要的典型数据，以及这些数据会如何存储。风控系统会如何使用这些数据，将下一篇博文中分享。

系列文章

支付风控系统设计：支付风控场景分析(一)

作者：凤凰牌老熊，程序员 & 架构师