

# 基于大数据技术的电商用户画像可视化设计

曾德真

南宁学院人工智能学院, 南宁, 530200

**摘要** 电商用户数据画像可视化技术是人们对用户数据进行分析建模可视化的重要手段。针对电商用户画像可视化需求, 研究电商用户画像可视化的关键技术, 提出基于大数据技术的用户画像可视化的技术方案。在该方案中, 利用 Flume 技术对电商系统中的用户信息进行采集, 实现对数据 ETL 迁移、标签模型 Oozie 调度、标签存储与计算; 通过 Kmeans 算法对数据标签进行挖掘; 最后通过 Hue 和 Echarts 可视化技术实现电商用户画像的可视化。

**关键字** 电商用户, 画像, 建模, 可视化技术

## Design E-commerce User Portrait Visualization Based on Big Data Technology

Zeng Dezhen

School of Artificial Intelligence  
Nanjing University  
Nanning 530200, China;  
984374899@qq.com

**Abstract**—E-commerce user data portrait visualization technology is an important means for people to analyze, model and visualize user data. In view of the needs of e-commerce user portrait visualization, this paper studies the key technologies of e-commerce user portrait visualization, and puts forward the technical scheme of user portrait visualization. In the scheme, the Flume technology is used to collect the user information in the e-commerce system, and ETL migration, label model Oozie scheduling, label storage and calculation are performed. And then, the data label is mined by Kmeans algorithm. Finally, the visualization of e-commerce user portraits is realized through Hue and Echarts visualization technology.

**Key words**—E-commerce users, Portraits, Modeling, Visualization technology

## 1 引言

用户画像首先在互联网公司发展起来, 是基于互联网数据的开放性、竞争性及互联网公司开发灵活性。随着社交平台针对用户画像后, 实现了对用户准确推送其兴趣内容, 到后来电商行业借助社交平台的画像能力展开营销推广, 再转变为电商行业自行根据用户消费习惯、消费能力进行主动画像与营销。用户画像基于大数据分析, 驱动行业主动了解用户, 使用户得到了良好体验, 同时推荐消费、关联消费、兴趣消费等新型消费因素带来的经济价值, 又促进了行业公司不断优化数据分析与用户画像模型, 形成了良好的循环机制<sup>[1][2]</sup>。

在国外, 以 Facebook、Twitter 等为代表的主流社交网站, 及 Ebuy、Amazon 等为代表的商城类网站, 都已拥有了成熟可靠的用户画像应用。在我国, 用户画像在电商行业、电力行业及三大运营商都有较为成熟的应用。在电商行业中, 京东、阿里这些以 TB 计的高质量、多维度的数据记录着用户大量的网络行为,

用户画像就是对这些数据的分析而得到的用户基本属性、购买能力、行为特征、社交网络、心理特征和兴趣爱好等方面的标签模型<sup>[2][3][4]</sup>, 从而指导并驱动业务场景和运营, 发现和把握在海量用户中的巨大商机。

目前, 有关用户画像可视化方法的研究成果取得了一些研究成果<sup>[3][4][5][6]</sup>, 但是不是太多, 还需要做比较深入和广泛的研究。本文对用户画像可视化关键技术进行研究, 提出对用户数据画像可视化技术的方案。

## 2 系统设计

### 2.1 构建用户画像

首先, 对用户画像的相关建模方法进行分析; 接着, 分别从用户的自然属性、兴趣属性、地理信息、设备信息、隐含属性五个维度对用户画像的刻画进行描述; 然后, 讨论了建立标签体系的方法, 将标签体系分为四级逐级细化; 最后, 对构建用户画像的基本流程进行讲解<sup>[9]</sup>。

(1) 用户画像的建模方法

在用户画像建模过程中，由于画像的需求和使用场景不同，所使用的建模数据也不同。获取到的数据中可能包括比较明确的数据，如：性别、年龄、操作方法、浏览页面链接等，也可能包括一些意图不明确的文本信息。所以，在用户画像建模的过程中所用到的方法根据实际需求也有所不同。如果原始数据中包含大量用户的基本属性信息，我们可能需要用到分类和聚类算法对用户进行群组划分并用以预测新用户。如果原始数据包含用户对网页的操作行为，则可能需要使用序列模式挖掘算法来发现用户的行为轨迹并根据用户的操作规则优化网站结构。如果原始数据中包含大批的文本信息数据，一般文本的意图不太明确，想要从中挖掘出关键词作为用户画像的标签，就需要用到文本挖掘相关算法来对文本的语句进行切分，再从语句中切分出关键词，按照出现的次数对关键词排序，以找出合适的标签。如果将用户画像用于产品营销领域，就需要用到推荐算法，根据用户最近的访问行为，挖掘出与该行为相关联的信息及时推送给用户，要注意信息的及时性，以达到精准营销。

构建用户画像的核心是为标签建模，过程如下：

- ① 获取原始数据。在本研究中所用到的数据为用户访问系统的行为日志和用户的基本属性，通过采集日志数据，得到用户的行为信息；
- ② 对数据进行预处理，挖掘出事实标签。对用户数据进行过滤、清洗、简化表示，从用户的基本属性信息可以得到用户的机构编号、等级、名称、第一次登录时间等，这些信息属于事实标签；
- ③ 分析用户行为信息，构建用户画像的模型标签。通过对用户的行为进行分析，得到用户访问页面和操作的规律，构建出用户的行为模型；
- ④ 通过模型标签进行预测，完善用户画像。根据用户的行为模型可以预测出用户的操作行为。在构建用户画像的过程中，主要建模方法如表 1 所示。

表 1 构建用户画像的主要建模方法

类型	相关算法	任务
文本挖掘	TextRank、TopicModel、LDA、TF-IDF、VSM	处理和清洗数据；匹配和标识用户数据、提取事实数据信息
分类/聚类	KNN、贝叶斯网络、神经网络、SVM 支持向量机	预测新用户；分析有相同特征的群体信息，对用户进行划分
机器学习	回归、决策树、EM	挖掘出画像标签
相似度计算	欧式距离、皮尔逊相似度、余弦相似度	辅助分类/聚类的特征划分；计算特征数据偏差值
推荐算法	SVD 矩阵分解、FPTree、Apriori	根据用户访问情况推荐出兴趣标签，完善画像
预测算法	回归预测、监督学习、线性规划	用户兴趣程度、忠诚度、流失预测，实现个性化服务

## (2) 多维度刻画用户画像

要想全面的挖掘用户信息，更加细致的构建用户画像，就需要从不同维度去刻画用户画像。如今用户画像已逐渐应用在各行各业，不同的领域所关注的内容有所区别，刻画用户画像的维度也不同，所以需要结合各领域的实际情况来划分用户画像的维度，建立多维度有侧重性的标签体系。本文将对教育统计平台的用户建立画像，根据该平台的使用场景和用户的具体需求，将从用户的自然属性、兴趣属性、位置属性、设备属性和隐含属性五个维度来构建标签体系。

第一、用户的自然属性主要指用户的一些静态特征信息数据，比如：用户注册时填写的姓名、性别、年龄、邮箱、电话、职业等。自然属性可以作为用户画像标签的基本特征，在挖掘兴趣标签时，可以根据基本特征信息对标签的层级、用户的群组进行划分。比如，不同职业的用户在使用系统时感兴趣的功能点不同；不同年龄阶段的用户常用的操作方式各异等。用户自然数据信息的标签比较平稳，标签系统将在建立后长时间不会变更。

第二、用户的兴趣属性主要指用户的行为偏好信息，包括访问的页面、使用的功能、访问路径等，还包括用户隐含的兴趣，这些信息需要从用户的操作日志中挖掘。构建用户画像的关键是为用户行为打标签。兴趣属性标签会随着时间的变化而不断变更，标签的时效性也比较强，标签系统需要定时更新，所以我们应该考虑到兴趣标签的衰减。做兴趣衰减常见的方法就是计算标签权重，后面的研究中我们会用到文本挖掘中的 TF-IDF 算法和 BM25 算法来计算标签权重。

第三、地理信息主要指用户的位置分布，对于用户移动轨迹来说，位置会不断变更，标签也要不断更新，对于常驻用户位置会比较固定，标签一般不用更新。针对教育统计平台用户分布的特征，我们有五台服务器，不同区域的用户在访问系统时会连接到不同的服务器，用户的地理位置与服务器的地理位置，同一时间段登录的用户分布情况等都与地理属性相关。

第四、设备属性主要指用户登录系统所使用的浏览器类型，电脑主机的 IP，连接到的服务器的 IP 等。用户登录系统所用的电脑不同，主机的 IP 也不同，从 IP 地址可以找到用户所在的区县。用户使用的浏览器不同与网站显示时所用的组件的兼容性有关，不同的浏览器对组件的兼容性处理方式不同。

第五、隐含属性指从用户的基本信息、行为信息等数据中发掘出用户信息中隐含的规律或偏好。本文将从用户上线的频率计算出用户的活跃等级，活跃等级可以反映出用户对系统的粘性。

### (3) 标签体系的建立

建立一套完整的标签体系是构建用户画像的前提。针对不同的行业,构建出的用户画像的使用场景和目标不同,标签的设定就需要结合具体的用户需求、产品的使用场景和使用目标来对标签的维度进行刻画。在标签体系中,标签是对客户特征信息的抽象化展示,它可以是任何文字或符号,但标签要体现出一定的群体性,需要从某些层面上反映和总结出事物的特性同时,标签系统的构建需要遵循 MECE (Mutually Exclusive Collective Exhaustive) 原则,即标签需要彼此独立且详尽无遗在建立标签体系时,应注意:

① 用户需求和场景会不断更新,所以标签体系是一个不断完善的过程,不可能一次性建立完成;

② 不同领域的用户需求和业务场景不同,用户画像的标签系统也不同。关键是要从不同层面更透彻地分析特定领域用户的决策行为。建立标签系统时,应根据具体业务情况进行切实分析;

③ 根据具体需求合理划分标签的体系结构,一般需要让标签体系有结构框架,呈现出一定的层级关系便于标签管理,也可以增加标签间的联系。

标签的体系结构一般包括三种类型:结构化标签系统、半结构化标签系统、非结构化标签系统。结构化标签系统组织良好,有明确的层级关系;半结构化标签系统关注的是标签系统的展示成果,对组织结构要求不高,但是不能太过混乱,否则在实践应用中比较困难;非结构化标签系统的标签间没有规整的结构,也没有层级关系,主要应用于搜索关键词。本文选择了一种非结构化标签体系的建立方法:关键词法。关键词法是通过实际的业务需要确定出标签体系的维度,并将具体的标签对应成为可以体现标签含义的关键词。然后使用计算文本相关度的方法计算出关键词在用户相关标签文本中的分布情况,在进一步确定标签对应关键词的权重。对标签体系进行层次划分,可分为:原始数据层、事实标签层、模型标签层、业务标签层。

### (4) 构建用户画像流程

目前,流行的用户画像方法是数据-用户标签映射方法,它是用数据来驱动用户画像的生成。首先,收集数据以获得用户的静态信息(性别,班级,邮箱,地区等)和动态信息(行为特征,社会特征,心理偏好特征等);接着,使用相关的算法(行为序列模式挖掘、文本挖掘、权重等)将用户信息转换为标签信息;然后,按照标签信息对用户画像建模;最后,对建模结果进行分析,用数据可视化技术将用户画像进行展示。用户画像的构建过程如图 1 所示。

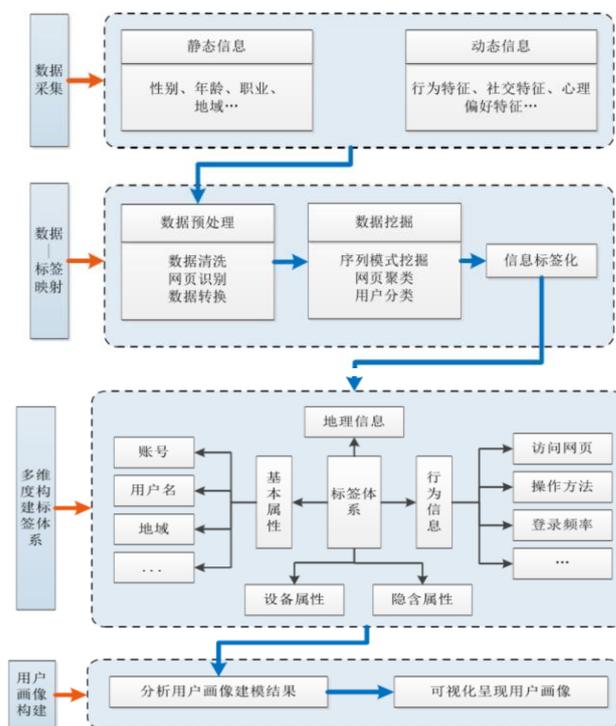


图 1 用户画像的构建过程示意图

## 2.2 系统模块层设计

系统模块分为数据层、业务层、展示层。其中:

数据层主要负责数据清洗、采集与导入,结构化数据从数据库或数据库文件中直接获取,需要对某些空字段、乱码、多余字符等进行处理;非结构化数据需要对文本内容进行格式化预定义处理,比如定长截取、字符分割等等,最终把两种数据转化为结构化可用数据。

业务层主要负责标签的分析、计算与生成,数据采集处理完成后,根据人工初步定义的特征标签词典,开始进入数据分类、关联、聚类分析。首先需要对数据进行切词,可以利用词频计算等方式,初步得到分词词典。其次根据词典,判断各个特征词语在某一客户数据中的权重,因为各词语在每个客户身上的权重是不一样的,此处使用 TF-IDF 计算权重,生成标签结果。当有一定对象或元组数据后,采用 K-means 算法对生成的用户画像结果进一步进行聚类分析。

展示层主要负责将业务处理层得到的数据结果进行可视化展示,通过对不同系统角色权限人的判断,提供不同的菜单功能。展示层主要使用 Javascript 图表技术表示数据结果,通过 Json 串由 Ajax 变量获得数据传入 Javascript 对应的 data 域中,实现可视化。

三层之间的逻辑处理流程图如图 2 所示。

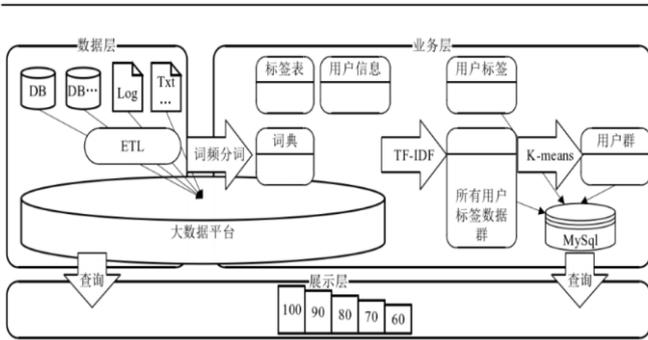


图 2 系统模块层的逻辑处理流程图

### 3 系统实现

#### 3.1 标签体系的建立

标签体系是根据已注册用户的偏好、行为习惯和人口属性等不同的领域进行建立的，按领域可以分为人口属性、商业属性、行为属性和用户价值四类。按具体的实现方式分为规则标签、统计标签和挖掘标签。在本项目中标签体系按照业务类型划分为基础标签和组合标签<sup>[10]</sup>。

标签按照级别可以划分为 5 个级别标签，总体来说分为三大级别：

(1) 第一级别为类别标签 :1 级、2 级、3 级标签属于类别标签。

(2) 第二级别为业务标签。4 级标签就属于业务标签，标签规则是：计算此标签的业务数据来源及字段信息。

(3) 第三级别为属性标签。5 级标签就属于属性标签，是业务标签的具体的值。可关注标签的 ID 和标签规则 rule，进行打标签。

标签发掘流程与算法如图 3 所示。对于挖掘类型标签开发来说，分为两步完成：

(1) 构建算法模型。可以从业务数据中获取算法特征数据 (features)。如果是监督学习算法，则需要标注数据，即 label。相关的建模过程如下：

业务数据 -> 特征工程 -> 训练模型 -> 最佳模型 -> 保存模型

标注数据、特征转换、特征提取

算法超参数、模型评估

(2) 模型预测值。加载模型 (算法模型提取训练好，保存起来)，封装方法 loadModel，如果模型不存在，使用数据训练，获取最佳模型，并且保存起来。

利用 predictionDF 结合属性标签规则，给每个用户打上具体的标签值。模型预测流程如图 4 所示。

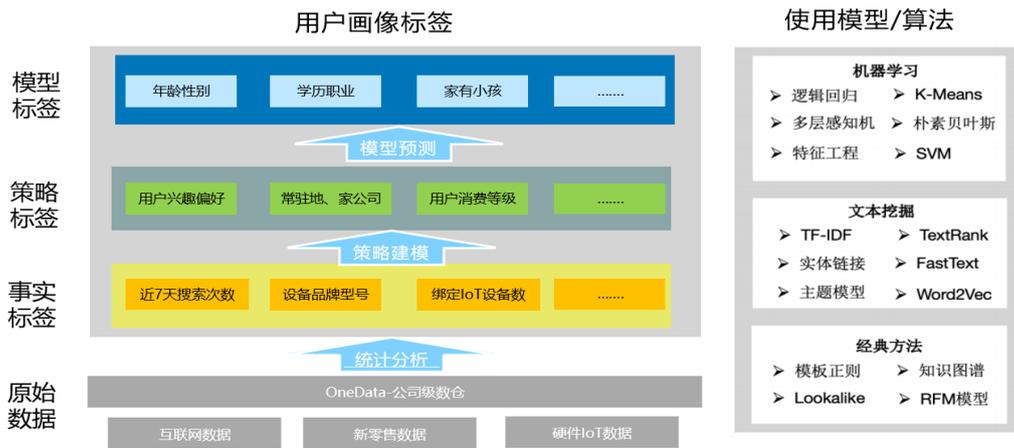


图 3 标签发掘流程与算法示意图

#### 3.2 标签体系的存储

每个标签 (4级标签、业务标签) 开发时涉及到的相关数据如下：

(1) 标签数据。标签管理平台中，每个标签开发时，新建标签：4级标签业务标签和5级标签 (属性标签)。其中业务标签对应标签模型，每个标签模型就是Spark Application，运行程序可以给用户打

上标签：TagName。在模型表中存储spark application 运行时参数设置。核心数据包括：tagName -> tagRule，标签规则等。

(2) 业务数据。依据每个业务标签 (4级标签) 的标签规则rule，获取业务数据。其中，inType判断业务数据的数据源，然后解析参数为Meta，加载业务数据 (SparkSQL)。值得注意的是，整个项目业务数据主要存储在HBase表中。

(3) 构建标签。使用业务数据和标签数据（属性标签对应tagName和rule）计算标签，得到modelDF，将其保存到HBase表中。其中，画像标签表为tbl\_profile。存储标签数据时，也将标签数据存

储同步存储到Elasticsearch索引中，方便使用标签进行查询用户。实现时，基于Elasticsearch为HBase表构建二级索引。图4标签体系的存储的关联图。

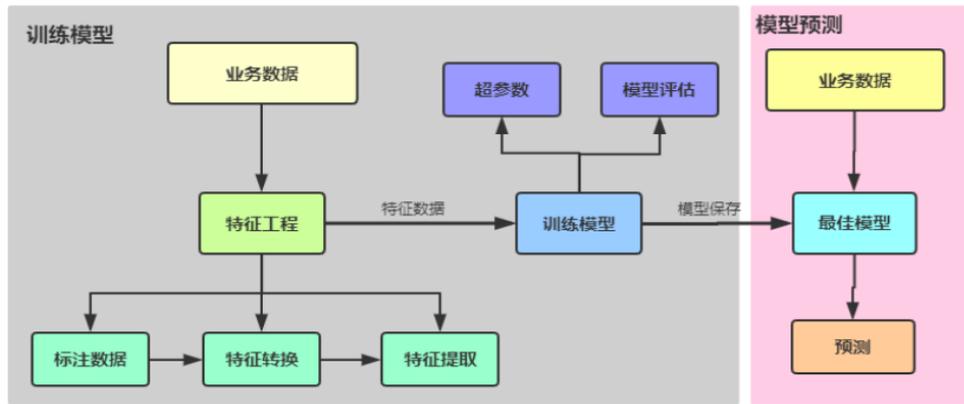


图 4 模型训练与预测的流程图

### 3.2 标签体系的存储

每个标签（4级标签、业务标签）开发时涉及到的相关数据如下：

(1) 标签数据。标签管理平台中，每个标签开发时，新建标签：4级标签业务标签和5级标签（属性标签）。其中业务标签对应标签模型，每个标签模型就是Spark Application，运行程序可以给用户打上标签：TagName。在模型表中存储spark application运行时参数设置。核心数据包括：tagName -> tagRule，标签规则等。

(2) 业务数据。依据每个业务标签（4级标签）的标签规则rule，获取业务数据。其中，inType判断业务数据的数据源，然后解析参数为Meta，加载业务数据（SparkSQL）。值得注意的是，整个项目业务数据主要存储在HBase表中。

(3) 构建标签。使用业务数据和标签数据（属性标签对应tagName和rule）计算标签，得到modelDF，将其保存到HBase表中。其中，画像标签表为tbl\_profile。存储标签数据时，也将标签数据存储同步存储到Elasticsearch索引中，方便使用标签进行查询用户。实现时，基于Elasticsearch为HBase表构建二级索引。图4标签体系的存储的关联图。

### 3.3 标签模型开发流程

展示每个标签模型在实际开发时主要流程如下：

Step1: 通过标签管理平台新建标签

- 1.1: 新建主分类标签（1、2、3级标签）；
- 1.2: 新建业务标签（4级标签）；

1.3: 设置相关属性，包含标签的属性字段的值和对应模型字段的值；

1.4: 标签模型对应Spark Application名称，及标签模型分类，尤其关键为标签规则；

1.5: 新建属性标签（5级标签）。

Step 2: 开发标签模型。如何开发标签模型及测试功能，完成以后需要打成jar包。

Step 3: 标签模型的任务调度执。

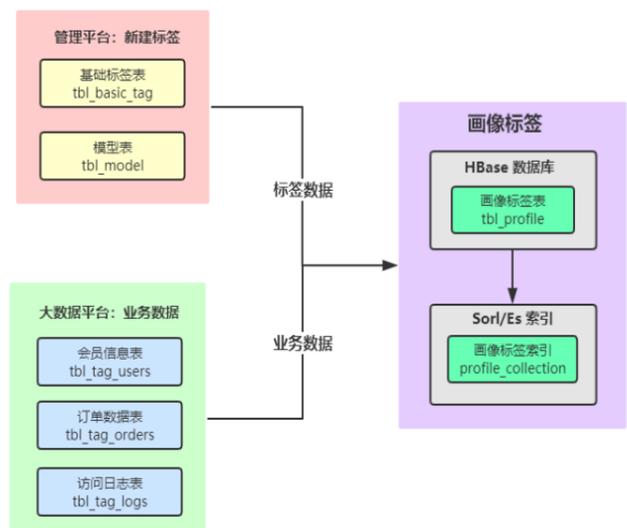


图 4 标签体系的存储的示意图

### 3.4 标签调度

本模块主要是基于Oozie实现Web管理平台和Yarn

计算平台的调度,方便管理的计算任务,Oozie在这里扮演一个公共的角色,所有的标签(模型应用)都需要使用Oozie来进行调度执行计算标签。

此模块单独为一个Maven Module进行开发,当做工具类库使用。将提供Maven Module导入值Maven Project中。标签模型jar包及J1时存储路径如下设置:

```
jars: "/apps/temp/jars"
model-base: "/apps/tags/models"
hdfs dfs -mkdir -p /apps/temp/jarshdfs dfs -mkdir -p
/apps/tags/models
#=====#
测试代码模型jar包地址
/apps/tags/models/tag_0/lib/model.jar
hdfs dfs -mkdir -p /apps/tags/models/tag_0/lib/
hdfs dfs -put /export/servers/spark-2.2.0-bin-
2.6.0-cdh5.14.0/examples
/jars/spark-examples_2.11-2.2.0.jar/apps/tags/
models/tag_0/lib/model.jar
```

主要的工具类为:OozieUtils,提供上传属性文件和提交任务执行,测试类代码OozieUtilsTest如下:

```
package cn.demo.tags.test
import cn.demo.tags.up.OozieParam
import cn.demo.tags.up.OozieUtils.{genProperties, start, store,
uploadConfig}
object OozieUtilsTest {
/* 调用方式展示 */
def main(args: Array[String]): Unit = {
val param = OozieParam(0,
"org.apache.spark.examples.SparkPi",
"hdfs://bigdatacdh01.demo.cn:8020/apps/tags/models/tag_0/li
b/model.jar",
"",
"2020-02-04T19:20+0800",
"2020-02-04T19:25+0800"
)
val prop = genProperties(param)
uploadConfig(param.modelId)
store(param.modelId, prop)
start(prop)
} }
}
```

测试完成以后,需要将此模块进行install安装,由于标签管理平台依赖此模块工具类,需调度标签模型。

### 3.5 标签管理平台

本模块前端使用Vue,后端使用SpringBoot进行开发,这个管理平台主要负责对标签的创建/查询等进行操作,并且负责对标签运行状态等进行管理。开发人员可以使用平台将标签计算应用JAR包上传,启动标签计算任务,方便标签管理。

在项目开发中,使用SpringBoot和Vue开发简易版WEB Platform,可以新建业务标签和属性标签,方便开发使用。

将提供开发完成Web系统的代码导入Maven

Project,数据库中创建相关表,运行应用即可。其用到的代码如下:

- (1) 后端代码,使用java开发 src/java
- (2) 前端页面代码,使用Vue resources/static
- (3) 属性配置文件 application.properties

标签管理平台Web Platform使用MySQL数据库进行数据存储,其中[基础标签表:tbl\_basic\_tag]和[模型表:tbl\_model]用于创建DDL语句及初始化1、2、3级标签,DML语句运行【WebApplication】类中main方法,浏览器输入地址: http://localhost:8081即可。

### 3.6 标签模型计算

标签模块计算主要根据原始表数据以及MySQL中的设置好的标签规则进行相应的计算,比如对规则匹配型、统计型和数据挖掘型等标签相关的计算操作,最终得到用户的最终的标签结果,并将其存储到HBase中。这里需要注意的是,在保存到HBase的时候,本次的保存一定不能把上次计算的标签结果进行覆盖,要将历史的标签数据和新生成的标签数据进行合并操作,这样才能保证数据的保存完全,不丢数据。

在系统中,每个标签(业务标签,4级标签)对应一个模型,每个模型就是一个SparkApplication。标签模型SparkApplication有一个核心的方法doTag,当标签模型应用程序运行以后,会给所有用户打上对应的标签的值。每一个业务标签都会有一个标签规则rule,标签规则决定标签计算时数据源。有了规则后,先获取标签规则rule值,解析完成后依据inType判断数据源,加载对应的业务数据。加载完成后,进行画像标签存储到HBase,通过协处理器将HBase的数据同步到Elasticsearch索引,这一过程就是画像标签数据。

每个标签模型计算时与存储系统交互过程如下:

**Step1:** 加载MySQL表的数据,加载标签数据,获取标签规则rule。每个业务标签都有规则,表示此标签计算时业务数据存储的地方(业务数据数据),比如存储HBase表中。

**Step2:** 加载HBase表的数据,加载业务数据。给用户打标签,需要获取标签的值,从业务数据中获取,比如性别标签,性别信息gender存储tbl\_tag\_users。这一步就是要读取业务数据,比如用户基本信息数据、订单数据等。

**Step3:** 打标签。结合属性标签数据和业务数据,给用户打上标签的值。使用标签的标识符TagId时要使用属性标签中标签规则rule。

**Step4:** 保存用户标签数据至HBase表。即将每个用户标签数据保存到HBase的表tbl\_profile中。

### 3.7 可视化

用户画像模型封装基于PG(关系型数据库)和大数据平台(hive、impala)包含基础标签与分析类知识标签,实现用户特征全貌刻画:多种封装角度,分用户类型、渠道内容、业务场景进行封装配置。接口数据实时推送,实现用户画像数据实时更新至运营及营销统一视图(WeMeta、WeDate、WeSearch等)中进行展现,并实时反馈运营及营销信息问题,保证数据应用的时效性。展现UI封装依托用户画像,将推荐信息配置应用端进行可视化展现,集中活动运营,实现千人千面的运营效果<sup>[7]</sup>。

系统最终展示效果如图5所示。用户数据画像可视化界面,涉及年龄分布,消费占比,行业区分比例,新增会员信息,消费记录,所属行业分布,用户偏好,精准营销,地区分布可视化功能模块,为企业提供了足够的信息基础,能够帮助企业快速找到精准用户群体以及用户需求等更为广泛的反馈信息。用户画像是

在解客户需求和消费能力,以及客户信用额度的基础上,寻找潜在产品的目标客户,并利用画像信息为客户开发产品。

用户消费数据可视化界面,涉及用户交易信息,交易量排名,商品分类占比,商品销量排行,个平台占比,以及城市排行可视化功能板块,可从多个维度了解用户的消费数据。实现了对用户准确推送其兴趣内容,到后来电商行业借助社交平台的画像能力展开营销推广,再转变为电商行业自行根据用户消费习惯、消费能力进行主动画像与营销<sup>[11]</sup>。

电商可根据用户消费习惯、消费能力进行主动画像与营销。在了解客户需求和消费能力,以及客户信用额度的基础上,寻找潜在产品的目标客户,并利用画像信息为客户开发产品。只有对用户的各个行为指标进行有效的分析,才能设计出符合用户心意的产品,最终把精准的用户画像创建出来<sup>[8]</sup>。



图4 系统展示效果图

## 4 结束语

本文对电商用户画像技术进行研究,提出了对电商用户画像技术解决方案。在该方案中,首先对数据ETL迁移、标签模型Oozie调度、标签存储与计算。接着对标签模型开发、规则匹配引擎。然后通过SparkSQL开发、外部数据源配置、Hbase应用、统计开发、Hbase条件过滤、数据条件动态过滤动态加载、通过Kmeans算法进行挖掘数据标签。最后,通过Hue和Echarts可视化技术实现用户画像可视化。

## 参考文献

- [1] 孙振兴. 基于用户画像的移动图书馆精准化服务研究[D]. 吉林: 吉林大学, 2022
- [2] 侯璠. 基于Spark的用户画像系统的设计与实现[D]. 上海: 华东师范大学, 2021

- [3] 岳怡然. 面向知识服务的用户画像构建与应用研究[D]. 北京: 中国农业科学院, 2021
- [4] 徐妙. 基于用户画像的YN移动APP运营策略研究[D]. 云南: 云南财经大学, 2021
- [5] 陈伟宁. 基于校园网络行为日志的用户画像研究与实现[D]. 天津: 天津大学, 2020
- [6] 马晓冬. 基于银行客户数据分析的用户画像系统设计与实现[D]. 四川: 电子科技大学, 2020
- [7] 王菊艳. 基于WEB日志的用户画像及可视化分析[D]. 西安: 西安理工大学, 2019
- [8] 李斯. 大数据背景下面向运营商精准营销的用户画像研究[D]. 大连: 大连理工大学, 2019
- [9] 王晓霞, 刘静沙, 许丹丹. 运营商大数据用户画像实践[D]. 北京: 电信科学, 2018
- [10] 任志斌. 基于电信数据的用户内容偏好画像补全[D]. 北京: 北京邮电大学, 2018
- [11] 鲁墨. 基于用户画像的推荐系统的设计与实现[D]. 沈阳: 东北大学, 2016