# 一种用户画像系统的设计与实现

王 洋[1,2] 丁志刚[2,3,4] 郑树泉[2,3,4] 齐文秀[1,2]

[1](上海市计算技术研究所 上海 200040)

[2](上海产业技术研究院 上海 201206)

[3](上海计算机软件技术开发中心 上海 201112)

[4](上海嵌入式系统应用工程技术研究中心 上海 201112)

**摘 要** 用户画像系统通过结合用户浏览行为日志以及爬取数据作为补充,构成用户浏览行为的完整数据集。通过以 Hadoop 分布式集群为基础的大数据平台结合数据分析算法对该数据集进行清洗、规范化、分析与处理,分析出用户兴趣偏好,为用户标记不同权重的标签,使得企业更了解用户以及为之后针对用户精准推荐铺平道路。此外,针对 K-means 算法依赖初始化中心的缺陷进行了改进,从测试结果可以看出改进后的 K-means 准确率得到了较大提升。

**关键词** 用户行为分析 基于 Hadoop 的大数据分析平台 用户画像系统 用户价值模型 K-means

**中图分类号** TP311 **文献标识码** A DOI: 10.3969/j.issn.1000-386x.2018.03.002

## DESIGN AND IMPLEMENTATION OF USER PROFILE SYSTEM

Wang Yang[1,2] Ding Zhigang[2,3,4] Zheng Shuquan[2,3,4] Qi Wenxiu[1,2]

[1](*Shanghai Institute of Computing Technology*, *Shanghai 200040*, *China*)

[2](*Shanghai Industrial Technology Institute*, *Shanghai 201206*, *China*)

[3](*Shanghai Development Center of Computer Software Technology*, *Shanghai 201112*, *China*)

[4](*Shanghai Embedded System Engineering Research Center*, *Shanghai 201112*, *China*)

**Abstract** The user profile system constructed complete data set of users browsing behaviour by combing user browsing log and crawling data. The data set is cleaned, normalized, analysed, and processed by data analysis algorithm and a big data platform based on Hadoop distributed cluster to get interests of users and mark labels of different weights for users, which enables enterprises to understand users and paves the way of precise recommendation for users. In addition, the defect of K-means algorithm relying on the initialization centre is improved, and the improved K-means algorithm has higher accuracy.

**Keywords** User behaviour analysis Big data analysis platform based on Hadoop User profile system User business values model K-means

## 0 引 言

截至 2016 年 6 月,中国网民规模达 7.10 亿[1]。随着互联网的蓬勃发展,越来越多的人加入到互联网用户的队伍中来,这为互联网企业带来了诸多的机遇,同时,也带来了诸多的挑战。这将意味着谁更了解自己用户的上网习惯、浏览偏好等,谁就能够在激烈的竞争中脱颖而出。

现如今用户行为日志随着互联网的快速发展呈现激增的趋势,数据规模已经开始由 GB 向 TB 乃至 PB 级别迈进。为了解决数据量过大带来的种种问题,本文提出了一种基于 Hadoop 大数据平台的离线数据处理分析系统。

收稿日期:2017-06-08。上海大数据科技成果转化平台(16DZ1110101)。王洋,硕士生,主研领域:分布式系统,大数据。丁志刚,研究员。郑树泉,高工。齐文秀,硕士生。

用户画像就是根据用户的人口属性、偏好习惯和行为信息而抽象出来的标签化画像[2]。目前,国内淘宝和京东都推出了自己的用户画像功能,通过对用户的个体消费能力、消费内容等长时间多频次的建模,为每个客户构建一个精准的消费画像[2]。国外对于用户画像研究,基于复杂网络理论对用户行为探索始于 2005 年,Barabási 在 Nature 发表的一篇论文[3],该文通过对用户普通邮件和电子邮件的发送和回复时间间隔统计特性研究,发现相邻两个时间间隔的分布服从反比成幂率的长尾效应。此外,Barabási 在他最近一本名为《Bursts》的书中就大胆地提出,93% 的人类活动是可预测的[4]。

本文是用户画像在大数据环境下的一种实践。传统的分析方式基于少量精确的结构化数据,但是,面对数据量大的情况,会出现速度慢甚至程序崩溃的风险。由此,引入基于 Hadoop 分布式集群的大数据处理平台,在数据量较大的情况下提供更可靠的分析与处理服务。

## 1　需求分析与相关技术

### 1.1　需求分析

用户画像系统的建立需要依赖于具体的应用场景以及所拥有的数据。本系统采用了某公司推出的一款互联网 WiFi 产品中采集的用户行为日志以及其他相关的用户信息作为源数据。

该日志中包含了用户浏览部分核心页面的历史记录,包括: 用户 MAC 地址、访问时间、接入设备 MAC 地址、访问页面类型、页面 URL、客户端类型等。由于用户行为日志中提取出的电影和电视数据不足以支撑后续的分析与处理任务,需要通过添加辅助数据采集模块,采集相关的电影和电视节目表单数据作为用户行为日志的补充。

依据用户行为日志中现有的数据信息,补充日志中残缺的部分,所构成的完整数据集合提交给大数据处理分析平台进行处理分析。然后通过可视化模块进行展示达到用户画像助力企业为用户进行推荐的目的。

### 1.2　相关技术

网络爬虫是一种自动提取网页的程序,它为搜索引擎从万维网上下载网页,是搜索引擎的重要组成[5]。数据爬取模块通过网络爬虫获取网络电视和电影数据,为源数据作补充。

大数据处理平台主要使用了 Hadoop 以及 Hive 等框架。Hadoop 是 Apache 软件基金会下的一个开源分布式计算平台。HDFS 作为 Hadoop 生态系统中主要存储系统,在实时性要求不高的情况下,已经成为很多公司首选的存储方案[6]。

Hive 是构建在 Hadoop 上的数据仓库框架。将结构化的文件映射为一张数据库表,并提供查询功能,可以 SQL 语句转化为 MapReduce 任务运行。

Sqoop 是一个用来将 Hadoop 和关系型数据库中的数据相互转移的工具,可以将一个关系型的数据库中的数据转移到 Hadoop 的 HDFS 中,也可以将 HDFS 中的数据转移到数据库中。

可视化模块主要用了 Spring、SpringMVC、Mybatis 作为 Web 端开发框架。

Spring 是一款开源框架,为了解决企业应用开发的复杂而创建[7]。Spring 框架的 IOC 容器设计降低了业务对象替换的复杂性,对组件之间解耦起到了重要作用。SpringMVC 框架提供了构建 Web 应用程序的全功能 MVC 模块,分离了控制器、模型对象、分派器以及处理程序对象的角色。MyBatis 是支持定制化 SQL、存储过程和高级映射的优秀的持久层框架。

## 2　用户画像系统总体设计

用户画像系统的整体架构分为四层: 数据源层、数据采集层、基于 Hadoop 的大数据分析平台层、数据可视化层。基本流程为: 数据采集层采集系统所需数据并将数据存入数据源层; 大数据平台层由数据源层导入数据并且对数据进行分析与处理,将处理完成的结果导出到数据源层; 数据可视化层从数据源层读取数据并将数据呈现在 Web 端页面供管理者参考。用户画像系统架构如图 1 所示。

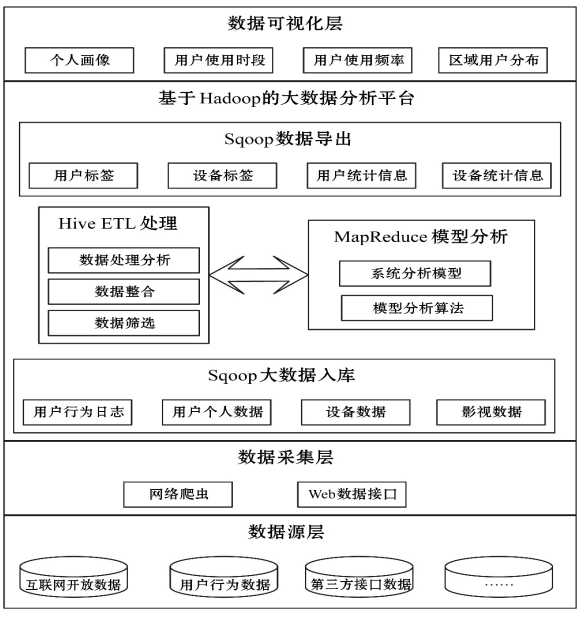



图 1　用户画像系统架构图

## 2.1 功能模块划分

用户画像系统分为三大模块: 数据采集模块、基于 Hadoop 集群的大数据分析平台、数据可视化模块。宏观上讲,数据采集模块主要用于补充用户行为日志中缺乏的电影数据、电视节目的相关数据以及源数据对接,使得数据集更加完备,为之后的分析与处理获得全面且合理的数据集作准备。基于 Hadoop 集群的大数据分析平台对用户行为日志经过清洗、规范化、分析与处理等步骤为用户标识相应权重的标签,实现为用户"画像"的目的。数据可视化模块将大数据平台中分析完成的结果进行展示,直观地看到用户的人画像,为决策起到辅助作用。

## 2.2 数据采集模块

数据采集模块包括三个部分: 电影数据爬取模块、电视数据爬取模块和源数据对接模块。其中,电视数据爬取模块和源数据对接模块主要采用调用第三方 API 接口获取数据的方式,定时抓取数据。

电影数据爬虫子模块通过爬取豆瓣电影网站中的相关数据来获取电影信息,主要包括电影名称、电影评分、电影导演、电影演员等。电影数据爬虫子模块爬取流程如图 2 所示。

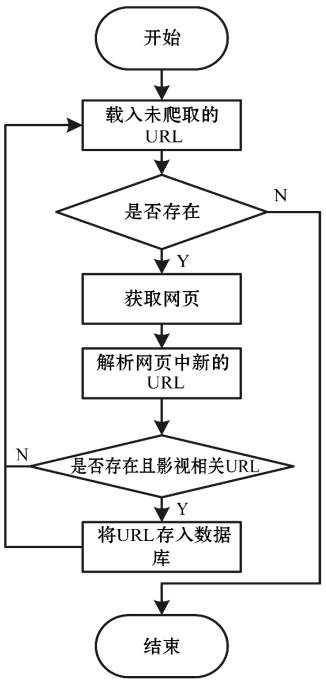



图 2 电影数据爬虫子模块爬取流程图

电影数据爬虫子模块流程如下:

1) 将初始待爬取 URL 存入数据库。并初始化其状态为未爬取状态。

2) 从数据库中获取未爬取状态的且 id 最小的 URL,使用基于 HttpClient 实现的 Http 访问工具包对该 URL 进行访问,获取到该 URL 的 HTML 页面数据。

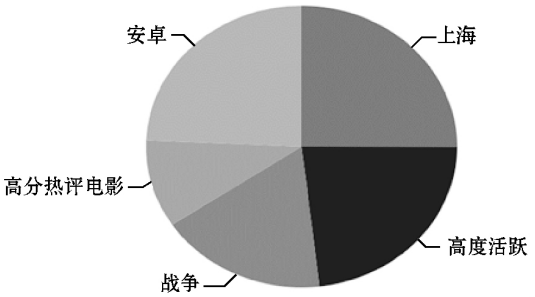
3) 对获取到的 HTML 页面利用 HTML 解析器 Jsoup 进行处理,分析并获取页面中的电影名称、演员表、导演表以及电影评分等,将解析成功的数据存入数据库中。

4) 获取该页面中固定模块中存在的 URL 并判断其是否电影相关的 URL,将符合要求的存入数据库中等待爬取。

## 2.3 数据可视化模块

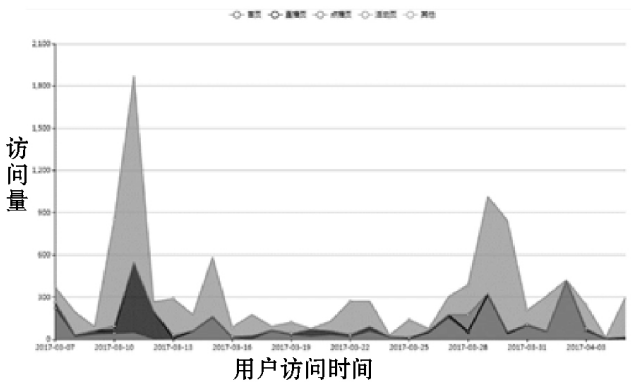
数据可视化模块中,采用基于 MVC 模式的三层架构。使用 Spring、MyBatis 和 SpringMVC 框架作为支撑,Echarts 商业级图标框架进行展示。

整个模块在展示用户个人画像的基础上,以接入的设备作为区域,展示区域内用户整体的统计分析量。数据可视化部分效果如图 3 所示。

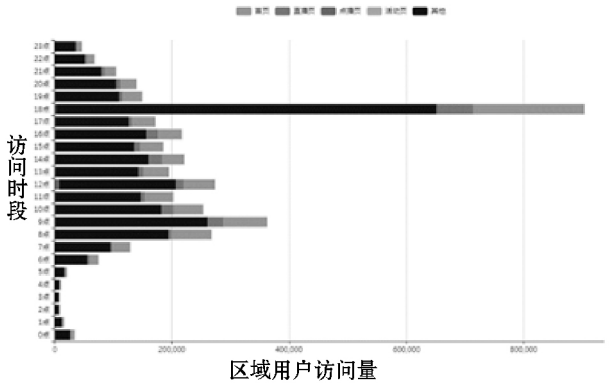



(a) 用户标签图

(b) 区域用户分布图




(c) 区域用户访问频率




(d) 区域用户访问时段图

图 3　数据可视化模块效果图

## 3　Hadoop 大数据分析平台

大数据指的是无法在规定的时间内用现有的常规软件工具对其内容进行抓取、管理和处理的数据集合。

### 3.1　集群拓扑结构

集群拓扑结构如图 4 所示。

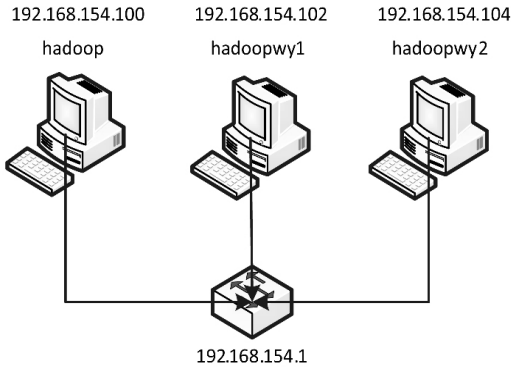



图 4　Hadoop 集群拓扑结构

Hadoop 节点结合 ZooKeeper 服务组成高可用 Hadoop 集群。取名为 Hadoop 的节点担任集群中 NameNode 角色。Hadoopwy1 作为 NameNode 节点的候选节点处于 StandBy 状态且担任集群中负责任务分配和资源管理的 ResourceManager 角色。Hadoopwy2 主要作用在于存储 HDFS 数据、处理 MapReduce 任务等。

### 3.2　大数据分析平台架构

大数据分析平台首先由 Hive 提取部分用户日志进行清洗与数据的整合交换，生成用于用户行为分析的建模数据，利用算法库算法对建模数据进行分析，生成用于用户行为分析的模型。然后，对所有用户行为日志进行数据清洗与数据的整合变换，得到预处理后的业务数据集，将得到的模型应用于该数据集得到最终的应用结果。最后，通过应用结果进行评测的方式评价模型优劣并对模型进行优化。

基于 Hadoop 集群的大数据分析平台架构如图 5 所示。

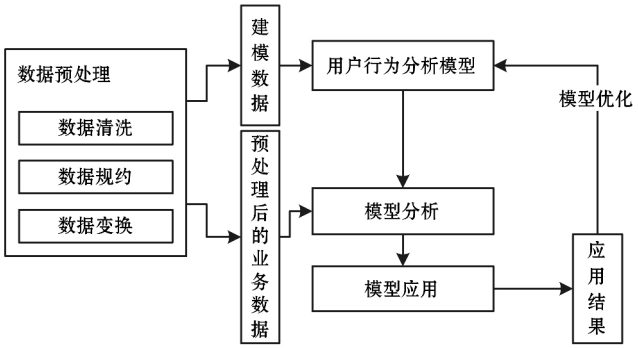



图 5　基于 Hadoop 集群的大数据分析平台架构图

### 3.3　用户画像标签体系

根据用户的相关数据将标签体系进行了划分，总体划分为两大类，基础信息标签、动态信息标签。基础信息标签是根据用户的注册信息中填写的个人信息得到; 而动态信息标签结合聚类等数据挖掘算法对用户行为进行分析挖掘，达到利用用户行为数据进行用户分类的目的。

平台部分标签体系如图 6 所示。其中，基础信息标签主要是从用户地址、用户的手机使用情况两方面进行分析。动态信息标签以用户价值标签和用户电影倾向标签为例进行分析与说明。

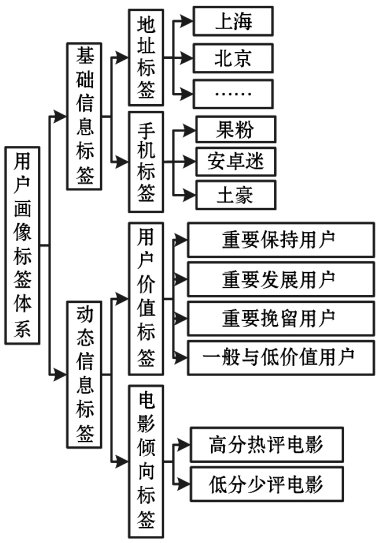



图 6　用户画像标签体系图

### 3.4　用户行为分析模型

基础信息标签主要采用统计的方式，按照一定的规则为用户标记对应的标签。

动态信息标签中，部分标签采用统计方法，另一部分标签采用数据挖掘算法对数据进行分析的方法，经过抽取数据分析所需要的数据，通过聚类等算法分析后区分各用户。

#### 3.4.1　基础信息标签模型

地址标签是按照用户注册时填写的用户地址直接

为用户标记该标签。手机标签模型是根据用户使用手机类型与数量进行标记。如表 1 所示。

表 1 用户手机标签模型

| 用户手机数目 | 手机类型 | 标签 |
|---|---|---|
| =1 | 苹果 | 果粉 |
| | 安卓 | 安卓迷 |
| | WindowsPhone | WP 粉 |
| >1 | 苹果、安卓、WindowsPhone | 土豪 |

### 3.4.2 动态标签模型及 K-means 改进算法

1) 用户价值模型 该模型将用户使用次数(UT)、初次使用到最近一次使用天数(DC)、最近一次使用距今天数(DF)、用户在线总时长(OT)四个值作为用户价值模型的参考指标,使用 K-means 聚类算法对数据进行聚类,通过对用户的行为数据聚类实现用户分类的目的。

K-means 算法是基于距离的聚类算法,在最小化误差函数的基础上将数据划分为预定的类数 $K$,采用距离作为相似性的评价指标,即认为两个对象的距离越近,其相似度就越大。K-means 算法[8] 对于给定样本集 $D=\{x_1,x_2,\cdots,x_m\}$ 聚类所得簇划分 $C=\{C_1,C_2,\cdots,C_k\}$ 最小化其平方误差:

$$E=\sum_{i=1}^{k}\sum_{x\in C_i}\|x-\mu_i\|^2 \tag{1}$$

式中: $\mu_i=\frac{1}{|C_i|}\sum_{x\in C_i}x$ 是簇 $C_i$ 的均值向量。

但是 K-means 存在着依赖初始化中心的缺陷,针对该问题对 K-means 进行改进。本文采用各个节点之间投票的方式选出第一个中心点,得到票数最多者胜出。这里假设每个节点为其余每个节点都准备一票,但是实际的投票值为除以两点之间的距离而得到的值。越是聚集的簇中心其所得票数相对较多,由此选出一个节点,然后求与已经得到的节点相距最远的节点,依次求出需要的节点。投票的值可以表示为:

$$E=\sum_{i=0}^{n}\frac{1}{d(x,y)} \tag{2}$$

式中: $d$ 是欧式空间中连个点之间的距离且 $x\neq y$。

初次聚类得到类簇以后,对每个类簇再分别随机选取其中的一个节点作为新的中心点。然后再一次进行聚类,循环多次选取方差较小者为最终聚类中心,从而降低 K-means 聚类时因为初始节点的选取而造成的影响。

实验表明,改进以后的 K-means 算法更加稳定,准确率更高。

通过 hive 进行统计与合并构成上述用户价值模型的数据集,运用 PCA 将数据降为 2 维数据如图 7 所示。

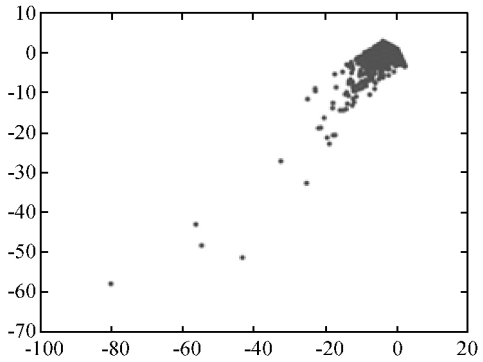



图 7 用户价值模型数据分布图

本文采用了 SSE(误差平方和)的方式对聚类效果进行度量。随着聚类簇数 $K$ 值的增大,总 SSE 值将逐渐减小。当 $K$ 的值小于其实际簇数的时候,随着 $K$ 值的增大,总 SSE 迅速下降。当 $K$ 值大于实际的簇数时,随着 $K$ 的增大,总 SSE 的值将呈现缓慢下降的趋势[9]。本文从多次聚类结果中选择总 SSE 递减趋势明显变缓的 $K$ 值作为聚类簇数。多次聚类后 SSE 值递减趋势如图 8 所示。

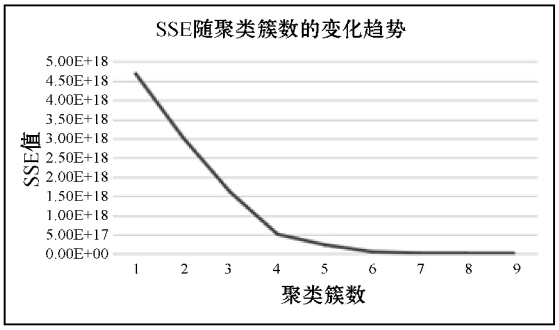



图 8 误差平方和 SSE 递减趋势图

从图 8 中可以看出,当 $K=4$ 的时候误差平方和的值已经递减十分缓慢,使用手肘法并结合业务需求以及图 7 中展示的 PCA 降维图确定 $K$ 值为 4,即本文中将用户行为数据划分为 4 个簇。

聚类分群后的结果如表 2 所示。根据表中分类结果可以看出,类 1 用户使用次数最少,初次使用到最近一次使用天数最少,最近一次使用至今天数最大,在线时长最少,基于这些特征将类 1 标记为一般与低价值用户。类 2 用户使用次数较多,初次使用至最近一次使用天数较长,最近一次使用至今天数较多,在线时长较多,基于这些特征将类 2 标记为重要发展客户。类 3 用户使用次数最多,初次使用至最近一次使用天数最大,最近一次使用至今天数最小,在线时长最长,基于这些特征可以将类 3 标记为重要保持客户。类 4 用户使用次数稍多,初次使用至最近一次使用天数稍多,

最近一次使用至今天数稍长,在线时长稍长,基于这些特征可以将类 4 标记为重要挽留客户。

表 2　用户行为聚类结果

| 类别 | 聚类个数 | 聚类中心 | | | |
|---|---|---|---|---|---|
| | | UT | DC | DF | OT |
| 类 1 | 49 320 | 18.2 | 10.6 | 35.9 | 4.3 |
| 类 2 | 41 | 4 101.9 | 43.6 | 20.4 | 67.1 |
| 类 3 | 5 | 13 599.6 | 47.2 | 12.4 | 392.8 |
| 类 4 | 402 | 785.7 | 43.4 | 20.9 | 46.3 |

2) 用户驱动力标签　用户驱动力标签是面向影视数据而言的,主要用于说明用户喜欢看电影对评分和星级的倾向。运用 K-means 算法对以电影评分、电影星级为数据集的电影数据进行聚类,方法与用户价值模型中类似。误差平方和的递减趋势如图 9 所示。

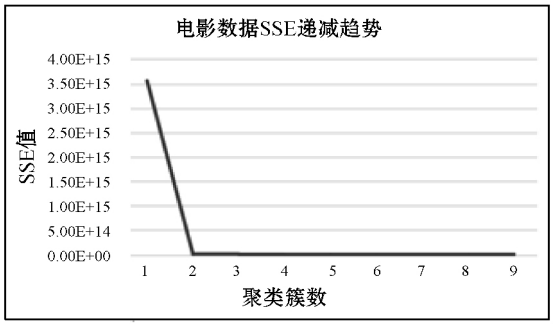



图 9　电影数据 SSE 递减趋势图

通过该图选取聚类簇数为 2,对电影数据进行聚类分析,聚类结果如表 3 所示。

表 3　电影数据聚类结果表

| 类别 | 聚类个数 | 聚类中心 | |
|---|---|---|---|
| | | 评分 | 星级 |
| 类 1 | 68 390 | 7.09 | 37.01 |
| 类 2 | 97 248 | 0.02 | 0.11 |

根据表 3 中的聚类结果进行分析,类 1 中电影评分较高,星级数目较多,基于这些特征将类 1 的电影定位为高分热评电影。类 2 中电影评分较低,星级数目较少,根据该特征将其定位为低分少评电影。

生成模型时通过计算用户所观看过的电影的评分与星级数目的平均值,与聚类质心计算距离,选取距离较近的质心所代表的类作为该用户的实际分类,将电影分类各自代表的标签赋予用户,表征用户的电影倾向标签。

## 4　系统测试

采用 UCI 机器学习库上面提供的开放数据集 Wine、Iris 以及本系统经过人工标记后的部分数据作为测试数据,验证文中提出的对 K-means 算法的改进。

Wine 数据使用 K-means 算法采用随机初始化中心时测试数据如表 4 所示。

表 4　K-means 算法对 Wine 测试数据表

| | 初始中心(行号) | | | 准确率 |
|---|---|---|---|---|
| K-means | 65 | 132 | 124 | 70.78% |
| | 79 | 41 | 154 | 47.75% |
| | 23 | 19 | 137 | 41.57% |
| | 161 | 140 | 9 | 70.78% |
| | 136 | 118 | 164 | 70.78% |
| | 97 | 18 | 17 | 41.01% |
| | 49 | 144 | 174 | 70.22% |
| | 74 | 105 | 62 | 71.35% |
| | 平均准确率 | | | 60.53% |

采用本文中提出的改进方案,多次测试结果趋于稳定,这不再一一列出。其初始化中心为:(3,13.73,4.36,2.26,22.5,88,1.28,0.47,0.52,1.15,6.62,0.78,1.75,520)、(1,14.19,1.59,2.48,16.5,108,3.3,3.93,0.32,1.86,8.7,1.23,2.82,1 680)、(2,12,0.92,2,19,86,2.42,2.26,0.3,1.43,2.5,1.38,3.12,278)。其中,第一列分类号不包含在计算之内。通过该聚类中心可以看出,本文提出改进起到了较好的初始中心的选择效果。针对该方案同样进行多次测试结果基本稳定,准确率的平均值为 70.78%。

Iris 数据集使用 K-means 算法采用随机初始化中心点的测试数据如表 5 所示。

表 5　K-means 算法对 Iris 测试数据表

| | 初始中心(行号) | | | 准确率 |
|---|---|---|---|---|
| K-means | 130 | 128 | 77 | 88.67% |
| | 20 | 48 | 139 | 56.67% |
| | 42 | 17 | 139 | 57.33% |
| | 48 | 128 | 8 | 52.67% |
| | 10 | 89 | 42 | 59.33% |
| | 67 | 19 | 18 | 56.67% |
| | 130 | 128 | 77 | 88.67% |
| | 130 | 128 | 77 | 88.67% |
| | 平均准确率 | | | 68.59% |

运用本文提出的改进算法,多次测试结果趋于稳定,在此不再一一列出。初始中心点为:(1,5.1,3.5,1.4,0.2)、(3.0,7.7,2.6,6.9,2.3)、(1.0,4.3,

3，1.1，0.1)。其中，第一列分类号不包含在计算之内。最终生成的聚类中心为：(5.01，3.42，1.46，0.24)、(6.87，3.09，5.75，2.09)、(5.9，2.75，4.41，1.43)。对比本文改进算法确定的第一个初始中心点和最终生成的聚类中心点可以看出二者相差很小。多次测试后的准确率平均值为：88.67%。

针对系统用户日志的标记数据进行聚类并统计其准确率同样得到了较高的提升，其中运用 K-means 随机确定初始中心的方式平均准确率为：64.50%，运用本文提出的改的 K-means 算法平均正确率为 83.43%。可以看出准确率在采用改进后的算法后有了较大的提升。

此外，为了验证本文中提出的基于 Hadoop 集群的大数据处理平台集群环境相对于单机环境效率的提升。在同等大小的数据集用户行为日志的基础上针对不同 MAC 地址各自操作次数的统计功能编写了在单机环境下和 Hadoop 集群环境下不同的实现方式，并测试了五组不同数据集的情况下的运行时间。

由实验数据得知，单机处理情况下，分别处理 250 MB、500 MB、1 GB、2 GB、4 GB 数据的时间随着数据量的突增呈指数级增长的趋势，用时分别为：5.45 s、7.5 s、13.94 s、42.18 s、158.75 s。而使用 Hadoop 分布式集群环境处理任务的情况下，分别用时为 17.01 s、20.99 s、21.30 s、22.42 s、30.04 s。单机与集群环境下运行时间对比如图 10 所示。从图 10 中可以看出单机情况下随着数据量的剧增处理时间也随之剧增，在 1 GB范围内的数据量表现较好，但是当数据量继续增长，执行时间随之集合呈现指数增长的趋势。而集群环境下，从 250 MB 数据剧增到 4 GB 的数据量的过程中，运行时间变化不大，变化趋势较为平稳。可见，基于 Hadoop 的分布式集群环境在大数据处理方面的优势很显著，而且随着数据量的不断增大其优势将越明显。

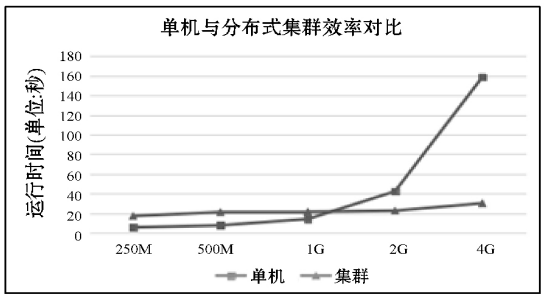



图 10 单机与集群执行时间对比图

## 5 结 语

本文针对用户喜好与互联网企业产品之间的矛盾，设计并实现了基于 Hadoop 集群的用户画像系统。首先，通过接口调用、网络爬虫等数据采集方式，获取到完善的数据集。然后，用户的行为日志经过统计分析和数据挖掘等方式的分析与处理，把用户抽象成标签的集合，从而达到为用户“画像”的目的。最后，可视化系统将用户画像系统的分析结果展示出来，为企业决策人员提供决策依据以及为后续向用户进行精准推送铺平道路。此外，本文对 K-means 算法依赖初始中心的缺陷进行了改进，改进后的算法更趋近于最终聚类中心，且准确率得到了极大的提升。

## 参 考 文 献

[1] 中国互联网络信息中心(CNNIC). 第 38 次中国互联网络发展状况统计报告[R]. 2016:1-2.

[2] 曾鸿，吴苏倪. 基于微博的大数据用户画像与精准营销[J]. 现代经济信息，2016(16):306-308.

[3] Barabási A. The origin of bursts and heavy tails in human dynamics[J]. Nature，2005，435(7039):207.

[4] Barabási Albert-László. Bursts: The Hidden Patterns Behind Everything We Do[M]. Dutton Adult，2010.

[5] 周德懋，李舟军. 高性能网络爬虫：研究综述[J]. 计算机科学，2009，36(8):26-29.

[6] 马建红，霍振奇. 基于 HDFS 的创新知识云平台存储架构的研究与设计[J]. 计算机应用与软件，2016，33(3):63-65.

[7] 李洋. SSM 框架在 Web 应用开发中的设计与实现[J]. 计算机技术与发展，2016，26(12):190-194.

[8] 潘巍，周晓英，吴立锋，等. 基于半监督 K-Means 的属性加权聚类算法[J]. 计算机应用与软件，2017，34(3):190-191.

[9] 成卫青，卢艳红. 一种基于最大最小距离和 SSE 的自适应聚类算法[J]. 南京邮电大学学报(自然科学版)，2015，35(2):103-107.

(上接第 7 页)

[8] Ong S. Beginning Windows Mixed Reality Programming: For HoloLens and Mixed Reality Headsets[M]. Apress，2017.

[9] Wang Xiangyu. Mixed Reality，in Architecture，Design & Construction[M]. Springer，2009.

[10] Penichet V M R，Peñalver A，Gallud J A. New Trends in Interaction，Virtual Reality and Modeling[M]. Springer London，2013.

[11] Kim G J. Designing Virtual Reality Systems: The Structured Approach[M]. Springer，2005.

[12] 顾君忠. 情景感知计算[J]. 华东师范大学学报(自然科学版)，2009(5):1-20.

[13] Ng L X，Ong S K，Nee A Y C. ARCADE: A Simple and Fast Augmented Reality Computer-Aided Design Environment Using Everyday Objects[C]//Proceedings of IADIS Interfaces and Human Computer Interaction 2010 Conference (IHCI2010)，2010:227-234.