doi:10.3772/j.issn.1002-0470.2020.03.006

一种基于动机感知的用户识别实时算法①

张梦菲②*** 邱 强* 肖茁建*** 姚 晓*** 方金云③*

(*中国科学院计算技术研究所 北京 100190)

(**中国科学院大学 北京 100190)

摘 要 用户识别是电商大数据行为挖掘的基础,本文提出了一种电商用户识别的新算法,该算法引入用户行为动机感知技术,采用初次匹配和精确识别二阶段模式来识别用户。初次匹配阶段算法利用启发式规则划分用户数据,在精确识别阶段通过实时分析用户的访问动机,依据用户行为相异数矩阵来识别用户。在 Spark 上的优化使算法在分布式场景中具备实时处理大规模数据的能力。实验结果表明该算法的准确率达 97.89%,并具有良好的识别效率。

关键词 用户识别;电子商务;Spark;用户动机;分布式计算

0 引 言

Web 数据挖掘一直是学术界和工业界研究的热点之一。随着大数据技术的发展,面向电子商务的 Web 数据挖掘在智能推荐、广告投放等领域发挥着越来越重要的作用。截至 2018 年 12 月,我国网络购物用户规模达到 6.10 亿,电子商务已成为大数据的重要来源[1]。用户识别技术作为 Web 日志挖掘的基础,是从大量无序的数据中分析出匿名用户的独立行为轨迹和特征,并最终识别出唯一的用户个体。其结果的准确性直接影响了后续数据挖掘和个性化服务的效果。因此,研究电商平台的用户识别具有重要的应用价值。

用户识别技术的研究主要集中在 Web 数据挖掘[2]、电子设备[3-5]、文本信息中匿名作者[6,7]以及共享账户[8]等领域。在 Web 挖掘领域中的用户识别方法主要有 2 种:(1)基于启发式规则的方法[9,10];(2)根据用户行为模式的方法。基于启发式规则的识别算法利用 IP、用户代理(userAgent)、cookie 技术识别用户,userAgent 是用户的操作系统及其版本信息和浏览器及其版本信息。Yen 等人[10]在启发式规则中证明了 cookie 技术比 IP + userAgent 方法具有更高的识别用户准确率,然而由于用户隐私问题,很难获得 cookie 数据的完整数据项。肖慧等人[11]提出了重写 URL 的 IASR(IP, agent, session and referrer)算法用户跟踪方法,在启发式规则中引入用户会话(session)来识别用户,该方法在服务器端支持 session 的情况下提高了准确率,但是实际情况中难以保证 session 的完整性和实效性。基于行为模式的用户识别算法根据用户兴趣和习惯的独特性[12]和稳定性[13],并利用数据挖掘和机器学习算法对点击流数据分类和预测来识别用户。Yang[14]提出了一种用户画像识别用户的算法,通过对时间 T 之前已知用户会话行为建立用户画像来预测未知用户,该方法对小规模数据取得了不错的效果,但对于规模较大的网站该算法不具备可行性。Naini 等人[12]将数据划分成匿名和已知用户数据,并用直方图展示用户 2 个星期内访问不同网站的习惯和行为信息,把问题转化为 2 个图的最小带权匹配问题,准确率达到 90.0%,但该算法主要

① 国家重点研发计划(2016YFB0502300,2016YFB0502302)资助项目。

② 女,1991 年生,博士生;研究方向:文本挖掘与推荐系统;E-mail: zhangmengfei @ ict.ac.cn

③ 通信作者,E-mail: fangjy@ ict.ac.cn

(收稿日期:2019-06-11)

针对的是用户在多个网站的访问行为,实际用户在一个电商网站的访问行为要更复杂。

在电商平台中,用户识别面临着以下的问题:(1)“多用户问题”和“单用户问题”,同一个用户在不同的时间内通过在地址栏输入 URL 或从收藏夹中进入网页会被识别为多个用户,即“多用户问题”;多个用户共享一个 IP 甚至使用同种设备和浏览器可能会被识别为一个用户,即“单用户问题”。(2)效率问题,关键绩效指标(key performance indicater, KPI)指数在百万级别以上的情况下,对用户识别算法的效率提出了更高的要求。Web 日志处理也越来越趋于实时化[15,16]。

针对以上问题,本文以义乌购小商品官网电商平台义乌购(www. yiwugo. com)的实际运营数据为研究对象,提出了基于 Spark[17] 的启发式和用户动机感知相结合的通用用户识别算法(Spark based user identification by Heuristic rules and user motivation perception,SHUMP)。SHUMP 算法采用初次匹配、精确识别的二阶段模式来实时识别用户。在初次匹配阶段,利用用户 IP、用户代理(userAgent)、cookie、用户会话(session)、引用(referer)等信息匹配用户。在精确识别阶段,提取每条点击流日志的 URL 特征,通过感知用户动机来精确识别用户身份。对于“多用户问题”和“单用户问题”,该算法在用户的 session 或者 cookie 缺失、用户设备相同、引用信息无法匹配的情况下,根据用户的行为动机来区分用户,将动机相似的识别为同一个用户,从而解决“多用户问题”;将动机不同的识别为多个用户,从而解决“单用户问题”;并对上述算法在 Spark 计算框架下进行了优化。

本文第 1 节介绍用户识别中的概念和问题定义。第 2、3 节分别介绍了 SHUMP 算法中 2 个阶段的具体技术细节。第 4 节介绍为解决实时化用户识别算法效率设计的分布式计算方法。第 5 节对本文算法进行实验验证和分析。第 6 节总结并展望本文的用户识别算法。

1 问题定义

定义 1　日志的用户访问记录集合 $I = < p_1, p_2, \cdots, p_m >$,其中 m 为记录的个数,$p_i(1 \leqslant i \leqslant m)$ 是每个用户的一次页面访问记录。

定义 2　网站每个用户的一次页面访问记录表示为:$p_i = <$time, URL, status, referrer, userAgent, sessionId, userId, cookie, IP$>$,p_i 的各个数据项记录了用户单次访问的行为信息,包括时间(time)、地点(IP 和 userAgent)、用户(userId)、来源(referer)、内容(URL)、会话 ID(sessionId)、cookie、状态(status)等信息。

定义 3　电商平台中用户类型集合如图 1 所示,其中 U_1 是已识别用户,对应定义 2 中 userId 数据项不为空的记录;U_2 是可追踪用户,对应定义 2 中 cookie 数据项不为空的记录,系统可以通过 cookie 数据项追踪用户;U_3 是未识别用户,对应定义 2 中 userId 和 cookie 数据项都为空的记录,网站只能获取这类用户的 IP 和 userAgent 信息。

定义 4　定义 3 中的 U_2、U_3 是本文的研究对象。用户识别是将定义 2 中 userId 数据项为空的记录与其他记录进行分析,判定哪些是同一个用户的行为记录,并为这些用户生成唯一的 ID 编码,即 userId。

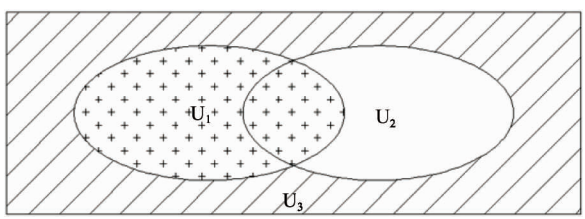

图 1　用户类型集合

2　基于启发式规则的初次匹配算法

本文对电商平台的所有日志数据分别根据 IP + userAgent、cookie 和 userId 划分为三级桶,其中在二、三级桶数据划分时,第 1 个阶段按照启发式规则进行初步匹配,第 2 个阶段用 URL 动机感知的算法来精确识别用户。

基于启发式规则的初次匹配算法是以 userId、cookie、session 和网站拓扑结构信息为用户初次识别的 4 个维度值,并根据相关规则进行匹配。具体步骤见算法 1。

算法 1　启发式规则的初次匹配算法

输入:2 个用户的行为特征 $feature_1$ 和 $feature_2$

输出:$feature_1$ 和 $feature_2$ 是否匹配

1. if(f = =0){key_1 = l_1. *cookie*,key_2 = l_2. *cookie*}
   else {key_1 = l_1. *userId*,key_2 = l_2. *userId*}
2. if(key_1 ! = null&key_1 = = key_2){return true}
3. if(l_1. *session* = = l_2. *session*){return true}
4. if(l_1. *referer* = = l_2. *url*){return true}
5. if((l_1. *referer* = = l_2. *referer*)&(l_1. *url* = = l_2. *url*)){returntrue}

算法输入为定义 1 中的 2 条日志 l_1 和 l_2 标志位 f,f 代表要匹配的是 cookie 还是 userId。如果非空的 userId、cookie 或者 session 相同,算法直接判定为同一个用户;否则,按照网站的拓扑结构,如果当前 URL 可以由上一条记录到达,则匹配。

3　基于动机感知的精确用户识别算法

3.1　动机感知模型

电商平台的用户动机主要反映在 URL 中,URL 中包括了用户浏览、搜索、点赞、购物车等行为信息,通过感知用户的动机,将用户访问行为反馈到用户识别。本文通过对每条 URL 提取出类型(type)、网站版块(block)、访问的商铺 ID(shopId)、访问的商品 ID(productId)、搜索关键词(word)5 个特征来构建用户的访问动机模型。具体步骤见算法 2。

算法 2　动机获取算法

输入:URL

输出:该 URL 所反映的用户动机

1. 对每条 URL 提取出特征 type
2. 如果 type = 4, 提取 block
3. 如果 type = 3,解析搜索词 word
4. 如果 type = 7,提取 shopId 和 productId

算法 2 构建了每条点击流日志 5 个维度的用户动机模型,其中特征 type 代表用户的不同行为类型,如表 1 所示,以电商网站为例,当 type = 4 时,表示用户在浏览网站的版块(block),如:论坛、采购、投诉等版块。shopId、productId、word 3 个特征包含了用户感兴趣的商铺、商品和搜索求购信息。表 2 展示了 3 种不同类型访问记录的动机解析结果,其中空白部分表示对应的信息不存在,shopId 和 productId 提供的用户动机信息还需根据 3.2 节内容来计算。

表 1　典型的电商网站 URL 类型解析

type	含义
1	外面的页面
2	首页
3	搜索页面
4	各个活动版块
5	购物车
6	订单页面
7	商铺或商品详情页面

表 2　义乌购 URL 动机解析示例

编码	URL	类型	网站版块	商铺 ID	商品 ID	关键词
1	/product/detail2.htm?productId=928685056	7	2	056034	928685056	
2	/search/s.html?q=%e7%ba%a2%e9%85%92	3	0			红酒
3	/sunbuy/list/1.html	4	9			

3.2　相似度计算

用户访问动机是指在一定时间阈值 T 范围内的用户访问轨迹,本文取阈值 T = 30 min。用户动机中如果不包含 shopId、productId、word 3 个特征,则匹配 URL 的 type 和 block;否则根据 shopId、productId、word 3 个特征对应的商铺主营范围、商品标题、搜索词来计算动机的相似度矩阵。

相似度矩阵是通过计算 2 个用户行为动机模型

的相异数来确定用户动机的距离。文献[18]提出了利用相异数计算 2 个词的距离,相异数越小,说明距离越小,2 个词越相似。算法提取了商品标题、商铺主营范围、搜索词等短文本的核心词集合 $< tag_1, tag_2, \cdots, tag_n >$,并把核心词映射到商品分类树的节点上,计算树的各个节点之间的距离,得到基础相异数;并利用式(1),从基础分析、文本分析、维度分析和扩展分析 4 个角度修正相异数,其中 tag_i 是核心词、id 是核心词在分类词库中的词码,dis_{ij} 是基础分析得到的相异数,f_1、f_2、f_3、f_4 分别是从词频分析、位置分析、附属维度和品牌维度得到的计算因子,表示该核心词在短文本中的重要程度,具体描述见算法 3。

$$Dis = \left\{ \begin{array}{l} tag_1 : id_1 = \dfrac{dis_{11}}{f_1 \times f_3 \times f_4}, \\ tag_1 : id_2 = \dfrac{dis_{12}}{f_1}, \\ tag_2 : id_3 = \dfrac{dis_{23}}{f_2}, \\ tag_3 : id_4 = dis_{34}, \\ tag_4 : id_5 = dis_{45}, \\ tag_5 : id_1 = \dfrac{dis_{11}}{f_1 \times f_3 \times f_4} \end{array} \right\} \quad (1)$$

算法 3 相似度计算算法

输入: 2 个用户行为的特征 $feature_1$ 和 $feature_2$

输出: 2 个行为是否匹配

1. 初始化阈值 threshold
2. if ($feature_1$ 和 $feature_2$ 的 word、shopId、productId 的任何一个特征同时存在){转至第 4 步}
   else {转至第 3 步}
3. if($feature_1$. match($feature_2$)) {return true}
   else{return false}
4. $T_1 < tag_{11}, tag_{12} \cdots tag_{1m} >$ = 提取 $feature_1$ 的核心词集合
   $T_2 < tag_{21}, tag_{22} \cdots tag_{2n} >$ = 提取 $feature_2$ 的核心词集合
5. 计算 T_1 和 T_2 的相异数 $value$;
6. if($value >$ threshold) {return false}
   else{return true}

表 3 ~ 表 5 是商铺主营、商品标题、搜索词 3 种用户行为动机模型的提取核心词示例,其中商品标题的核心词用相异数修饰其在标题中的重要程度。表 6 是用户行为动机相似度矩阵计算结果,P1、P2、S1、S2、W1、W2 代表表 3 ~ 表 5 提取的核心词,S1 指

表 3 商铺核心词提取示例

编号	商铺主营	核心词
S1	批发太阳镜,平光镜眼镜框,偏光镜,儿童眼镜	儿童款, 墨镜, 太阳镜, 镜,儿童镜, 眼镜
S2	包芯丝,天鹅绒,连裤袜,双层保暖裤,一体裤等	包芯丝,天鹅绒,连裤袜,双层保暖裤

表 4 商品相异数示例

编号	商品标题	相异数
P1	批发眼镜促销 儿童款太阳镜 墨镜椭圆形 儿童镜 015-43	墨镜:80240101000000 = 0; 太阳镜:80240101000000 = 0; 镜:80140363010000 = 16; 儿童镜:80240109000000 = 0; 眼镜:80240000000000 = 0; 眼镜:80240100000000 = 0;
P2	丝袜连裤防勾丝 打底袜女款 天鹅绒打底袜 连脚显瘦	打底袜:80090207041600 = 0; 丝袜:80090207050500 = 6; 丝袜:80090207040500 = 5; 服饰:80090200000000 = 8; 袜子:80090207000000 = 2; 连裤袜: 80090207041600 = 0; 袜:80090207000000 = 2;

表 5 搜索词核心词提取示例

编号	用户搜索词	核心词
W1	女士太阳镜	太阳镜
W2	蓝色打底裤加绒	打底裤

表 6 用户行为相似度矩阵计算结果

	P1	P2	S1	S2	W1	W2
P1	0					
P2	16	0				
S1	0	16	0			
S2	16	0	16	0		
W1	0	16	0	16	0	
W2	16	8	16	6	64	0

一条 URL 代表的用户行为是访问了一个商铺,该商铺的主营是 S1。表中的值代表用户行为相异数,S1 和P1的用户行为相异数是 0。当 type 和 block 相等或用户行为相异数小于阈值 *threshold* 时,则判定两条 URL 为同一个用户。

4 二阶段识别算法的实时化

二阶段识别算法的复杂度为 $O(n^2)$,如何满足在线访问用户的实时识别直接关系到电商平台推荐系统的用户体验。本文在 Spark 分布式计算框架下对算法进行了优化,算法流程如图 2 所示。

步骤 1 数据清洗,Spark Streaming 从 Kafka 中并行读取实时数据流,并调用 filter 算子,将日志爬虫和异常信息过滤,得到真实用户访问的数据流 normalDStream;

步骤 2 一级桶划分,将日志数据以 IP + userAgent 作为 key 值,对相同的 key 进行数据划分,产生一级桶数据 $b_{11}, b_{12}, \cdots, b_{1n}$, 数据集合为 IAD-Stream;

步骤 3 数据匹配,调用 groupBykey 算子和 mapValues 算子,在每个桶 b_{1i} 中进行数据匹配,数据匹配包括初次匹配和 URL 动机相似度计算;

步骤 4 二、三级桶划分,对一级桶 b_{1i} 进行桶划分的原则是:假设 $p_i \in b_{1i}$ 并且 $p_j \in b_{1j}$, 则在 b_{1i} 中总是存在记录 p_k ($i \leqslant k \leqslant j$), p_i 和 p_k 相互匹配并且 p_k 和 p_j 相互匹配。通过判断第 2 节中算法 1 (l_1, l_2, f) 的 f 标志位,并按照 cookie 和 userId 划分成二、三级桶数据 b_{2i}, 和 b_{3i};

步骤 5 userId 生成,为所有匿名用户生成 userId,登录用户的 userId 保持不变。

SHUMP 算法利用 Spark 将日志数据划分成大小均衡的 DStream 数据流,分配到指定多个节点的内存中,并利用转换算子对数据进行三级桶划分和用户识别,从而找到用户各自所属的类别。

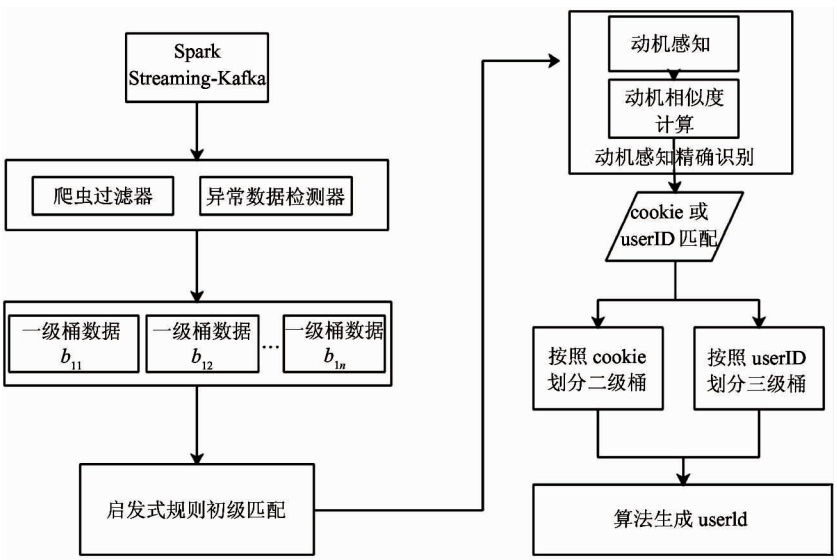

图 2 SHUMP 算法并行计算流程

5 实验结果与分析

5.1 实验数据集与环境

本文采用义乌购(www. yiwugo. com)网站的 nginx 日志数据,该网站依托全球最大的小商品批发市场——义乌小商品城。网站每天产生百万级别的真实点击流日志记录,其数据格式如表 8 所示,每一条日志对应定义 2 中的 p_i。

本文从用户识别的准确性和效率 2 个方面进行了实验验证。在由 7 台服务器搭建的集群环境中,每台节点都配置 jdk1. 8、Hadoop-2. 7. 0、Zookeeper-3. 4. 8和 Spark-2. 2. 0 相关环境,采用主从方式进行实验。表 7 是实验所用服务器的硬件配置信息。

表 7 服务器硬件配置信息

Operation system	CPU frenquence (GHz)	Memory size (GB)	Disk size (TB)
CentOS 7.2	24Core 2.67GHz	64	1

5.2 准确性验证

5.2.1 准确率验证

在准确率实验中,本文将日志按照天进行分割,对 2017 年 10 月 1 日至 10 月 31 日 31 天的日志进行预处理,抽取出登录用户的日志,并将日志中的 userId 数据项清除。式(2)是准确率计算公式,其中,*originalNum* 为清除 userId 数据项之前的原始用户数量,*SHUMPNum* 为经过 SHUMP 算法计算得到新的用户数量。算法准确率最高达 99.97%,月平均值为 97.89%(图 3)。

$$accuracy = \begin{cases} \dfrac{originalNum}{SHUMPNum}, & originalNum < SHUMPNum \\ \dfrac{SHUMPNum}{originalNum}, & originalNum \geqslant SHUMPNum \end{cases} \tag{2}$$

表 8 义乌购电商平台 nginx 日志格式

IP	日期	方法/URL/协议	userAgent	cookie	session	状态码	引用	域名	userId	请求时间	访问时间戳
203.116.188.204	14/Dec/2016:02:26:58	GET /index.html HTTP/1.1	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.1(KHTML, like Gecko) Chrome/21.0.1180.89 Safari/537.1	CgMDZFhPiSJWFxRSCz+lAg==	D3CD803F6E32682403FA3374EEA142B8	200	-	www.yiwugo.com	-	0.073	1481653620.682

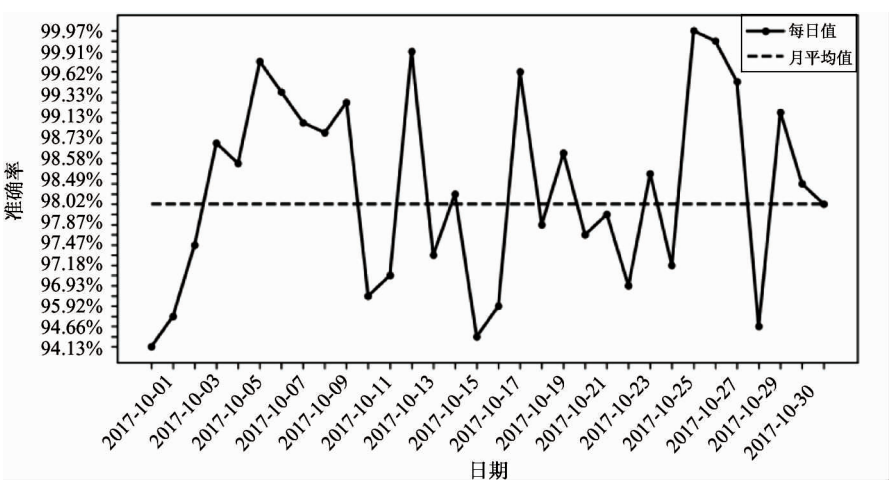

图 3 1 个月内 SHUMP 算法的准确率变化图

5.2.2 对比验证

文献[12]构建用户之间访问路线的二部图结构,然后计算二部图的最大匹配,但是实验数据将用户访问的 URL 加密,每个网站作为最小单位,计算匿名用户访问不同网站的统计信息,并且数据量小,极大简化了用户识别的问题,然而并不适用于内容繁多、页面链接复杂的的 Web 场景。文献[14]将用户识别问题转化监督问题,基于用户行为预先建立

用户画像,然后按照 session 进行匹配得到分数,并分类到已知的用户中,在小数据集比如 2 个用户的识别上准确率最高达到 99.30%,低于本文算法的准确率,然而该算法随着数据量增大,准确率下降,仅在 100 个用户的数据集上准确率就下降至 87.36%。以上方法不仅存在大数据量的可扩展性问题,效率也没有得到验证。与监督学习不同的是,本文的方法并非将测试数据分类到正确的类别上,而是将一组用户同时找到它们各自所属的:“类别”,它们可能为同一个用户,也可能属于独特的一个用户,本文方法能找出并赋予它们独特的 ID。在对比实验中,本节与复现的文献[11]进行实验对比,验证 SHUMP 算法的时间效率和用户数量,同时将与部署的百度统计结果进行对比,进一步验证算法的可靠性。

表 9 展示了原始记录数为 500 万条的日志经过 IASR 算法和 SHUMP 算法之后的数据,SHUMP 算法处理速度明显高于 IASR 算法,识别的用户数量多于 IASR 算法,由于 IASR 算法仅仅考虑传统的启发式匹配规则,对于 IP 地址和设备相同的用户,如果用户禁用 cookie 等信息,或者从网页收藏夹直接进入网站造成引用信息无法获取,该算法无法区分不同的用户,无法解决“单用户问题”和“多用户问题”,而 SHUMP 算法则会根据用户的动机进行相似度匹配计算,可以识别出更多的真实用户,提高识别准确率。

表 9　SHUMP 与 IASR 算法效率和准确率对比

算法	总记录数(万条)	耗时(ms)	用户数(个)
IASR	500	64 738	135 792
SHUMP	500	785	194 020

另外,本文在义乌购网站中部署百度统计对比识别出的网站用户个数,实验通过计算网站 14 天的用户识别数量与百度统计相同时间段内的用户数量对比如图 4 所示。可以看出指标的趋势基本一致,但本文数据比百度统计略高,这是因为百度统计依赖于浏览器设置 cookie、JavaScript、图片等,如果浏览器禁用,则无法获取访问数据,本文直接分析实际的日志文件,不受浏览器的限制,数据更加完整。因此本文的用户识别算法是可靠的。

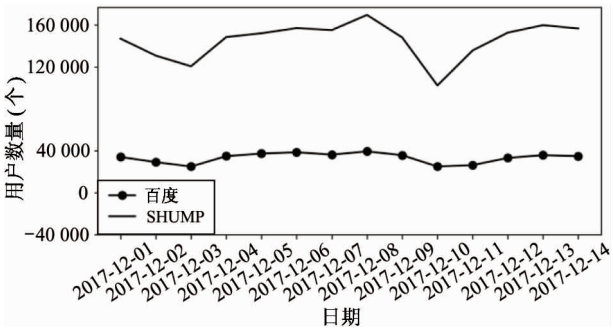

图 4　百度统计与 SHUMP 算法识别出的用户数量对比图

5.3　效率验证

实验将一段时间的日志数据预处理之后得到正常数据,并分成 20 份,从 50 万条日志增大到 1 000 万条。实验分别用单线程、多线程、Spark 框架实现二阶段用户识别算法,并按式(3)计算 3 个算法的处理时间。其中,T_i 代表各个数据量的算法处理时间,T_i 由统计 10 次处理时间求平均并 4 舍 5 入求整得到。

$$T_i(1 \leqslant i \leqslant 20) = round(\frac{\sum_{j=1}^{10} T_{ij}}{10}) \qquad (3)$$

图 5 为 3 种二阶段用户识别算法效率随着数据量大小的变化趋势,其中,HUMP with single thread 和 HUMP with multi-thread 为二阶段用户识别算法的单线程和多线程实现版本,统称为单机算法(HUMP),SHUMP 是二阶段用户识别算法在 Spark 上的优化。可以看出,多线程 HUMP 算法处理时间低于单线程 HUMP 算法,但二者皆在数据量达到 900 万条之后处理时间陡增,而 SHUMP 算法处理时间远远低于 HUMP 算法,集中在 0 ~ 1 s,表 10 详细记录了其执行时间。在百万级别的数据情况下 SHUMP 算法耗时始终保持毫秒级,可以满足用户实时识别的场景。

6　结　论

本文采用启发式规则初次匹配和用户动机精确识别的二阶段识别用户算法。对于 cookie 相同或者 session相同或者符合网站拓扑结构或者访问动机相

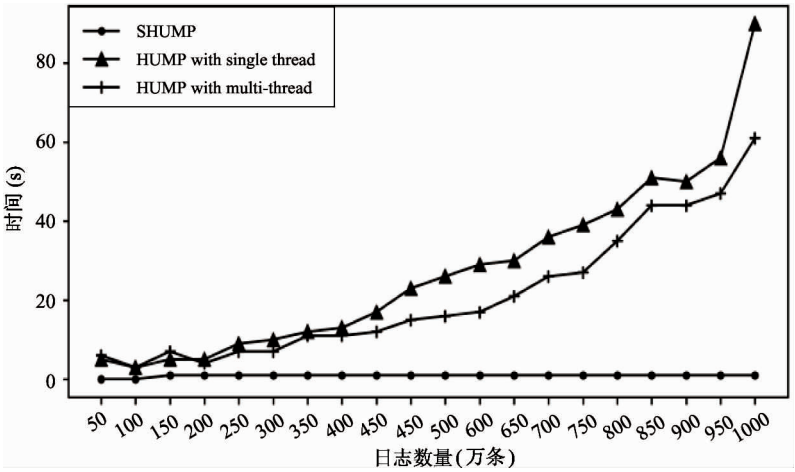

图 5 SHUMP 算法与 HUMP 单线程和多线程算法随着数据大小变化执行时间变化图

表 10 SHUMP 算法随数据量大小变化处理时间分布

日志大小(万条)	平均处理时间(ms)
50	452
100	458
150	502
200	541
250	517
300	717
350	636
400	553
450	577
500	566
550	604
600	708
650	847
700	714
750	764
800	646
850	765
900	664
950	722
1 000	609

同的用户,算法分配相同的 ID。用户动机分析有效解决了相同 IP 地址的多个用户使用同种操作系统和浏览器访问网站会被认为是单用户的问题以及当一个用户多次直接进入网页被认为是多用户的问题。实验表明 SHUMP 算法具有较高的准确率和处理效率,比百度统计结果更可靠。SHUMP 算法已经应用在义乌购电商平台。

本文提出的 SHUMP 算法为用户行为分析和商品推荐奠定了基础,算法对电商网站的用户识别具有一定的通用性。但受电商业务影响,用户动机模型不尽相同,用户的个性化访问习惯也增加了用户识别的难度,这将是下一步研究的内容。

参考文献

[1] 中国互联网络信息中心(CNNIC). 第 43 次中国互联网络发展状况统计报告[R]. 北京:CNNIC, 2019

[2] Srivastava J, Cooley R, Deshpande M, et al. Web usage mining: discovery and applications of usage patterns from Web data[C]//Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Boston, USA, 2000: 12-23

[3] Ahmad M, Khan A, Brown J, et al. Gait fingerprinting-based user identification on smartphones[C]//Proceedings of the 2016 International Joint Conference on Neural Networks, Vancouver, Canada, 2016: 3060-3067

[4] Cao W, Wu Z, Wang D, et al. Automatic user identification method across heterogeneous mobility data sources [C]//Proceedings of the IEEE 32nd International Conference on Data Engineering, Helsinki, Finland, 2016: 978-989

[5] Malatras A, Geneiatakis D, Vakalis L. On the efficiency of user identification: a system-based approach[J]. *International Journal of Information Security*, 2017, 16(6): 653-671

[6] Okuno S, Asai H, Yamana H. A challenge of authorship identification for ten-thousand-scale microblog users[C]

//Proceedings of 2014 IEEE International Conference on Big Data, Washington DC, USA, 2014: 52-54

[7] Narayanan A, Paskpv H, Gong N Z, et al. On the feasibility of internet-scale author identification[C]//Proceedings of the 2012 IEEE Symposium on Security and Privacy, San Francisco, USA, 2012: 300-314

[8] Lesaege C, Schnitzler F, Lambert A, et al. Time-Aware user identification with topic models[C]//Proceeding of the IEEE 16th International Conference on Data Mining, Barcelona, Spain, 2016: 997-1002

[9] Juels A, Jakobsson M, Jakobsson T. Cache cookies for browser authentication[C]//Proceedings of the 2006 IEEE Symposium on Security and Privacy, Oakland, USA, 2006: 301-305

[10] Yen T, Xie Y, Yu F, et al. Host fingerprinting and tracking on the web: privacy and security implications [C]//Proceedings of the 19th Annual Network and Distributed System Security Symposium, San Diego, USA, 2012: 1-16

[11] 肖慧, 王立华. Web 日志挖掘中的用户识别算法[J]. 计算机系统应用, 2011, 20(5): 223-226

[12] Naini F, Unnikrishnan J, Thiran P, et al. Where you are is who you are: user identification by matching statistics [J]. *IEEE Transactions on Information Forensics and Security*, 2016, 11(2): 358-372

[13] Olejnik L, Castelluccia C, Janc A. On the uniqueness of Web browsing history patterns[J]. *Annals of Telecommunications*, 2014, 69(1-2): 63-74

[14] Yang Y. Web user behavioral profiling for user identification[J]. *Decision Support Systems*, 2010, 49 (3): 261-271

[15] 孙大为, 张广艳, 郑纬民. 大数据流式计算: 关键技术及系统实例[J]. 软件学报, 2014, 25(4): 839-862

[16] 王元卓, 靳小龙, 程学旗. 网络大数据: 现状与展望[J]. 计算机学报, 2013, 36(6): 1125-1138

[17] Zaharia M, Chowdhury M, Franklin M J, et al. Spark: cluster computing with working sets[C]//Proceedings of the USENIX Workshop on Hot Topics in Cloud Computing, Beston, USA, 2010: 1-7

[18] 陈翠婷. 无分类小商品搜索引擎关键技术研究[D]. 北京: 中国科学院计算技术研究所, 2016: 27-30

A real-time motivation aware user identification algorithm

Zhang Mengfei[*,**], Qiu Qiang[*], Xiao Zhuojian[*,**], Yao Xiao[*,**], Fang Jinyun[*]

([*]Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190)

([**]University of Chinese Academy of Sciences, Beijing 100190)

Abstract

User identification is the basis of electronic commerce big data behavior mining. A new algorithm for electronic commerce user identification is proposed. This algorithm introduces the technology of user behavior motivation perception, and identifies the users by using the rough match and the accurate identification of two phases. User data is divided by heuristic rules in the stage of rough matching, and the user's motivation is analyzed in real time during the precise identification phase, and the user is identified according to the dissimilarity matrix of user behaviors. Finally, the Spark computing framework is used to deal with large-scale data in distributed scenarios. Experiment results show that the accuracy of the proposed algorithm reaches 97.89%, and it has good identification efficiency.

Key words: user identification, electronic commerce, Spark, user's intention, distributed computing