文章编号:1671-4598(2018)08-0227-06 DOI:10.16526/j.cnki.11—4762/tp.2018.08.051 中图分类号:TP391 文献标识码:A

基于 USDR 模型的云推荐方法研究

陆佳炜，卢成炳，王辰昊，肖 刚
（浙江工业大学 计算机科学与技术学院，杭州 310023）

摘要：在云环境中，对于云数据的统一建模一直是研究热点；尤其在推荐系统中，对于多源异构数据灵活性和安全性的要求更高；随着数据信息技术的不断发展，网络更新逐步加快，数据的更新也越来越快，从海量信息中如何快速帮助用户获取偏好的信息变得更加困难；针对多源异构数据的特征，综合移动互联网安全性和隐私性等特点，提出了一种 USDR 模型，并在该模型的基础上对云环境中的推送方法进行了研究，主动帮助用户发现自己偏好的信息，并将这些信息展现给可能需要的用户，并且实现了传统的数据推荐方法无法处理的多源异构数据的云推荐；根据云推送平台在实验环境中的运行情况及相关指标分析，说明该云推荐方法能适用于多源异构数据的推荐，是一种高效可行的推荐方法。

关键词：云推荐；云计算；多源异构数据；数据推送

Research on Cloud Recommendation Method Based on USDR Model

Lu Jiawei, Lu Chengbing, Wang Chenhao, Xiao Gang
(College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract: In the cloud environment, the unified modeling of cloud data has been a hot research topic. In particular, the requirement for flexibility and security of multi-source heterogeneous data is higher. With the continuous development of data and information technology, the gradual acceleration of network updates, data updates are also getting faster and faster, from the mass of information on how to quickly help users get the preference of information becomes more difficult. According to the characteristics of heterogeneous data, comprehensive mobile Internet Security and privacy features proposed a USDR model, and on the basis of the model of cloud environment in the push method were studied, take the initiative to help users find their preference information and will show users may need the information, and to achieve the recommended that cannot be handled by traditional methods of data recommended multi-source heterogeneous data cloud. According to the operation condition and related indexes of cloud push platform in experiment environment, it is suggested that the cloud recommendation method can be applied to multi-source heterogeneous data, and it is a highly efficient and feasible recommendation method.

Keywords: cloud push; cloud computing; multi-source heterogeneous data; data push

0 引言

Web 在科技的进步和信息的更新交替中进入了“2.0 时代”，同时由于各种信息更新速度的加快，互联网的数据资源也同步进入了大数据云时代，在某种程度上，网络垃圾和无效资源也越来越多，当普通用户想要寻找某种有用的资源时，如何在海量数据中筛选出特定的资源变成一个急需解决的问题。

搜索引擎作为人们获取信息的渠道和关键，始终是各大互联网公司的一个争夺的热点。当人们坐下来，打开电脑，面对庞大的互联网世界时，第一件事情往往就是打开搜索引擎，输入关键字，从而以最快的速度找到自己想要的信息。但是同样存在明显的缺陷，即对用户的文化水平有一定程度的门槛，有一部分人不知道如何联想到并精确的概括自己的目标信息，从而错过很多实时信息；还有一些用户并没有绝对明确的目标，只是想浏览一些自己感兴趣的话题，并不想要通过某些关键字使得信息狭隘化，因为有些关键字之间的共同信息领域很小；还有一些用户对感兴趣的话题并不能用几个关键字去概括，因而无法定位到自己想要的数据资料。然而对数据信息的制造者而言，由于现在互联网的竞争非常激烈，希望自己的信息被关注被采纳、用户量节节攀升也不是一件容易的事情。在这种情况背景下，数据推荐应需而生。对于用户而言，数据推荐系统可通过云计算，在使用界面里主动跳出或许对用户有价值的信息，从而使用户达到自己的目的，得到更好地使用体验；而对于制造者，数据推荐可以在一定程度上合理地把信息推销给潜在用户，从而增加自己的点击量，这对于双方而言是一个共赢的局面。

现如今，数据推荐引擎适用范围非常广泛，尤其值得关注的就是近几年发展迅猛的电子商务平台，以淘宝为例：当使用者搜索过某类商品以后，它就会储存这个点击数据同时进行某种用户偏好的计算统计，结合商家的综合排位和对淘宝平台的广告买位，在使用者平台上进行个人化的

收稿日期:2018-01-08； **修回日期**:2018-01-25。
基金项目:国家自然科学基金项目(61573316)；浙江省重大科技专项(2014C01048)；浙江省重点研发计划项目(2018C01064)。
作者简介:陆佳炜(1981-)，男，硕士，讲师，主要从事云计算、软件工程方向的研究。
肖 刚(1965-)，男，博士，教授，主要从事图像识别、云制造方向的研究。

反馈，使用者就会很容易的注意到自己感兴趣的信息，同时商家获取更多的点击量和利润，淘宝自身也获得巨额利润，这是一个“三赢”的结果。再如分享交互类的社交平台，以新浪微博为例，建立推荐的机制，向用户推荐好友的搜索热点和关注人分享的内容，使得每一个使用者的界面都是独特的个人化的，而且这都是使用者一手操办，所以这些信息对于使用者而言是感兴趣的有价值的。同时被关注者也可以利用这种关注量和影响力获得经济利益，平台作为秩序的维持者和信息资料的拥有者也可以获得巨大的利益。就目前而言，信息推荐系统在各大领域都产生了良好的效果和不可或缺的作用，用户也逐渐习惯和信赖信息推荐系统，可以说这是一个成功的机制。

1 相关工作

国内外学者和研究机构从不同的视角对多源异构数据和推荐方法进行了研究。

从 RSS 推荐技术方向出发的代表性工作主要有：Hao Han 等人[1]在 RSS 推送的基础上构造网络新闻文章内容自动提取系统，可以从新闻网页中提取对用户有价值的文章内容；陈锋等人[6]对信息服务资源进行聚合需求分析，提出了一种基于 RSS 推送技术的信息服务内容聚合服务方式。

其次，协同过滤推送是目前主要使用的推送方式之一，协同过滤推送不仅可以实现信息的推送，而且可以根据用户的兴趣实现个性化推送。目前对协同过滤推送技术研究中具有代表性的有：郭艳红等人[7]提出了一种基于稀疏矩阵的个性化改进策略，能够避免用户之间相似度不密切的关系，提高了矩阵在稀疏情况的预测准确度。李聪、梁昌勇等人[8]提出了基于领域最邻近的协同过滤推荐算法，使数据的稀疏性得到了降低，提高了推荐准确性。

从数据传输方向出发的代表性工作主要是 Menglan Hu 等人[2]设计了一种分阶段获取云端分享数据的算法，能够有效地控制数据的传输成本。国内的许富龙、刘明等人[9]进一步提出了一种基于相对距离感知的动态数据传输策略，采用传感器节点到汇聚点的相对距离来计算节点传输概率的大小，并以此作为消息传输时选择下一跳的依据。

在利用推送技术实现系统的研究中，中国科学院软件研究所的刘鑫、陈伟[10]提出了一种基于 AJAX 和 Server Push 的 web 树组件，为用户提供了类似于在 windows 资源管理器中对目录树操作的基本功能和用户体验。

但以上方法均只是通过修改推送方式而实现对单一数据源进行推荐，并没有过多考虑多源异构数据的个性化推荐问题，也没能实现云推荐。本文提出的 USDR 模型面向多源异构数据，通过将用户数据和系统数据分类来快速得到用户和系统的不同推荐度，以实现数据的高效推荐。

2 USDR 模型概念

在数据物流云推送平台中，各类云数据数量庞大，种类繁多，根据系统服务种类大致可以分为成绩查询服务、工资数据服务、排队服务、交通数据服务、购物信息服务、股票期货服务、多媒体数据推送服务等。

由于是基于云推送的数据物流服务平台，平台中许多系统会提供类似的服务，比如 3 种股票软件都通过本平台为客户提供金融数据推送，但是其中一款股票软件是收费软件，数据推送响应时间更快、推送的服务更多，但价格也是同类股票软件中最高的。除了相同类型的服务中出现的情况，用户数据信息之间也存在不同，用户将会根据自己的基础信息选择不同的服务。比如交通数据服务中，有些用户可能上班时间比较自由，那么他们可以选择上下班高峰期过后的道路数据推送服务，而有些用户需要准时到达单位，那么推送给他们当时的路况数据，可以使他们选择在上下班高峰期避开一些拥堵路段；同样，购物信息服务中，经济条件好的用户可能比较偏好奢侈品，而经济条件一般的用户则偏好于普通实用的商品，所以在推送数据时就会有一定的差异性，需要建立用户和系统的关系数据模型。

当用户请求获取一种类型的服务时，数据物流服务平台应该自动根据现平台中相同类型的系统和用户自身的数据，推送给用户最合适的服务，这样就既能满足用户的功能性需求，同时也满足了用户的个性化需求。

用户数据主要可以分为用户基础数据、时间数据、地点数据、用户偏好数据、历史数据等。

系统数据主要可以分为服务类型数据（如成绩查询服务、金融股票服务等）、服务介绍以及这些服务的范围（价格、位置）。这些系统中的数据结构多样，类型复杂，并且有些数据是动态变化的。为了能够有效的处理这些云数据，本文提出了 USDR 模型。

2.1 用户数据模型建模

根据上文中的分析可以看出，用户数据基本可以划分为五类：

用户基本数据（BasicData）：包括用户姓名、性别、身份证、电话、出身日期、职业、毕业学校、爱好、出生地等。

时间数据（TimeData）：记录用户使用系统的日期和时间，同时也记录用户所在的时区。

地点数据（LocationData）：用于记录用户所在的位置，包括城市，住所和工作地。

环境数据（EnvironmentData）：记录当日天气情况，温度等。

用户偏好数据（PerferenceData）：记录用户的偏好情况，如运动、电影、理财、旅游、读书等。

历史数据（HistoryData）：记录用户曾经使用的系统服务，常用的理财，消费记录以及日志数据等。

通过 UML 工具可以很清晰的看出用户各类数据之间的关系，并且通过设置主键显示出各条属性的重要程度，具体如图 1 所示。

系统数据服务有成绩查询服务、工资数据服务、银行排队服务、交通数据服务、酒店预订服务、股票期货服务、多媒体数据推送服务。这些系统都属于不同的领域，这些系统的数据类型复杂程度高，数量大，若不进行建模将很

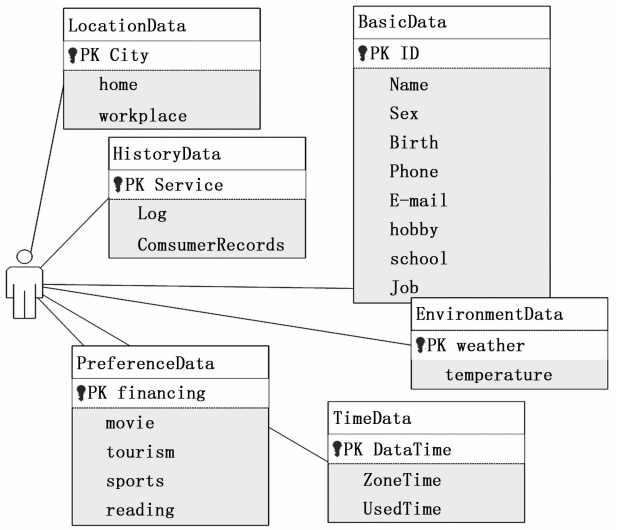

图 1　用户数据模型

难进行云推送，在对系统数据进行建模之后也更利于数据的个性化推荐，本章选择具有代表性的成绩查询服务系统和多媒体数据服务系统进行系统数据模型建模。

2.2　系统数据模型建模

2.2.1　成绩查询服务系统

成绩查询服务系统主要为在校学生提供每个学期结束之后的成绩查询服务，首先最高层应该为用户的类型，为本科生、硕士研究生还是博士研究生，确定了学生类型之后需要到各个学院中查询数据，由于很多学院中的必修课是相同的，所以为了避免重复的查询接下来模型中将分为必修课和选修课以及实践活动。最终得到各门功课的成绩。最后学生得到了该门课的成绩之后，还需要对老师进行评价。

系统数据模型再结合用户数据模型可以看出，在用户数据模型中的用户偏好，毕业院校就可以更加精确地给用户推送推荐数据，同时这种分层的结构能使云推送更加高效。

2.2.2　多媒体数据服务系统

多媒体数据服务相对于成绩查询服务将会复杂很多，多媒体数据服务各种系统中，可以将数据的类型分为文字数据、音频数据、视频数据、图文数据等。根据多媒体服务的不同类型和用户的偏好将分为新闻，体育，娱乐，游戏，电影等，然后再对具体需要推送的数据进行分类。

系统数据模型主要元素包括基本数据、功能数据和其他数据：

基础数据（SerBasicData）：主要是对系统服务的基本描述，包括服务提供商，服务类型，服务 ID，服务名称，服务简介等数据。

功能数据（SerFunctionData）：主要对服务中的功能性参数进行描述，即服务输入输出参数，服务的接口参数，最终服务执行结果等。

其他数据：主要有些系统需要定位数据，天气数据等其他因素。

3　基于 USDR 模型的云推荐算法

传统的推荐算法有皮尔逊相关系数法、向量余弦法、斯皮尔曼相关系数法等等，在不同的领域中，需要选取不同的相似度计算方法。由于云数据的特殊性，本文重新设计了基于 USDR 的个性化云推送推荐算法，根据用户、系统的相似值来计算推荐的系统数据。本章的模型中存在用户数据和系统数据两种数据类型，针对该模型设计了基于用户的云推荐算法和基于系统的云推荐算法。

3.1　基于用户的云推荐算法

基于用户的云推荐算法主要目的在于计算两个用户的相似度，本算法中主要使用用户行为相似度来计算用户的类似喜好。本算法由两部分组成：一部分采用用户基础属性来决定用户的相似程度，通过计算得出的基本属性差异越小，则相似程度越高；第二部分是偏好、位置和服务记录数据等，通过查看用户的地理位置和历史感兴趣的系统的数值，该数值越大，则用户之间的相似程度越高，最后计算总相似度。

3.1.1　基础属性相似度

基础属性一般都是数值类型，如性别，年龄，毕业院校等。对于数值型属性，只需要计算绝对值之差 $|D|=|Attr1-Attr2|$。对于名称型的基础数据，一般取值类型比较单一，就可以采用二进制编码的方式来表示，比如性别：男、女，分别对应 00、01。其他以此类推。最终将用户全部名称型数据编码串联起来，行成一个二进制串。

不同的数值型属性的绝对值最大与最小的差距为 $[\alpha_1, \alpha_n]$，然后把这个区间划分为 $n-1$ 个相等的区间 $\{[\alpha_1, \alpha_2], [\alpha_2, \alpha_3], \ldots, [\alpha_{n-1}, \alpha_n]\}$，对每个区间给予相应的数值 {0，1，2，3... n}，当用户的数值型属性绝对值落在某个区间时，即可得出属性间的距离 D_{bnum}。对于名称型属性，通过确定编码位数 n，然后将每个取值通过格雷编码，然后依次链接起来，最后通过计算海明距离，得到名称型属性距离 D_H。定义用户 A 和 B，每个基础属性的权重值为 w_i，则所有属性权重值满足：

$$\sum_{i=1}^{n} w = 1 \tag{1}$$

对于数值型的属性距离 D_{bnum}，根据上面的解释，定义不同的取值区间：

若 $\alpha \in [\alpha_1, \alpha_2]$，则 $d_{\text{bnum}}=0$；

若 $\alpha \in [\alpha_2, \alpha_3]$，则 $d_{\text{bnum}}=1$；

.....

若 $\alpha \in [\alpha_{n-1}, \alpha_n]$，则 $d_{\text{bnum}}=n-1$；

数值属性的距离计算为：

$$DNum = \sum_{i=1}^{n} widi \tag{2}$$

对于名称型的属性距离 D_{bnum}，则对不同的取值进行编码。将用户的全部名称属性编码串联起来，形成二进制串 At；采用 At 的海明距离来计算用户名称属性的距离。

$$DH = wDhm(D_{\text{bnum}}A, D_{\text{bnum}}B) \tag{3}$$

最终得到 2 个用户 A 与 B 的基础属性距离：

$$D_{A-B}=\sum_{i=1}^{n} widi+\overline{W_{num}}DHm \tag{4}$$

通过差值 D_{A-B}可以看出，D_{A-B}越小，相似度则越大，D_{A-B}越大，则相似度越小。

3.1.2 用户偏好相似度

若给定用户 A 和 B，N（A）表示用户 A 的偏好相似度集合，N（B）表示用户 B 的偏好相似度集合（如时间，位置，系统使用情况等），运用余弦公式相似度计算公式：

$$wnv=\frac{|N(u)\cap N(v)|}{\sqrt{||N(u)||N(v)||}} \tag{5}$$

表 1 用户偏好表

用户 A	成绩查询系统	金融服务系统	酒店预订系统
用户 B	成绩查询系统	工资系统	
用户 C	交通查询系统	金融服务系统	多媒体系统

从表 1 的用户偏好可以得出：用户 A 对｛成绩，金融，酒店｝方面的系统感兴趣，用户 B 对｛成绩，工资｝方面的系统感兴趣，所以可以计算出用户 A 和用户 B 的偏好相似度，如下所示：

$$wAB=\frac{|\{成绩，金融，酒店\}\cap\{成绩，工资\}|}{\sqrt{\{成绩，金融，酒店\}?\{成绩，工资\}}}$$

用余弦公式计算用户间两两的相似度之后，算法通过综合分析基础数据相似度和用户偏好数据相似度后，再进行推荐，推荐度公式如 6 所示：

$$Recommduser=\sqrt{D_{A-B}+\sum_{v\in Re(u,\ l)\cap N(i)} wabybi} \tag{6}$$

公式中，D_{A-B}为基础数据的差值，N（i）表示对项目 i 有偏好的用户组，Re（u，k）表示存在与用户 A 偏好类似的用户组。W_{ab}描述用户 A 与用户 B 的相似度，y_{bi}表示用户 B 对项目 i 的偏好程度。

3.2 基于系统的云推荐算法

基于系统的云推荐算法和基于用户偏好的推荐算法有些类似，主要通过以下两步完成：首先计算系统之间的相似程度，然后根据相似度生成系统推荐列表。

根据余弦公式可得系统的相似度：

$$wij=\frac{|Num(i)\cap Num(j)|}{\sqrt{|Num(i)|}} \tag{7}$$

从余弦公式中可以看出，Num（i）表示偏好系统 i 的用户数量，Num（j）表示偏好系统 j 的用户数量，与 $\sqrt{|Num(i)|}$ 的比值表示在偏好系统 i 的用户中同时也偏好系统 j 的比例。但是当系统 j 是一个所有人都偏好的系统时，如工资系统，任何其他系统通过公式（7）得出的结果都会很大，所以本文中将公式（7）进行修改，如公式（8）所示：

$$wij=\frac{|Num(i)\cap Num(j)|}{\sqrt{|Num(i)|}?\sqrt{|Num(j)|}} \tag{8}$$

公式（8）在分母中加入了 $\sqrt{|Num(j)|}$，相当于降低了系统 j 的权重。首先设定权重值 w 区间范围为［w_1，w_n］，将［w_1，w_n］分割为 n 个小区间｛［w_1，w_2］，［w_2，w_3］，...，［w_{n-1}，w_n］｝，每个区间赋值｛0，1，2....，n｝，然后对所有系统两两比较，如果用户的偏好落在区间范围内，那么认为这些系统属于同一个领域，相似度很大，值得推荐。分三步介绍采用基于系统的云推荐算法的简单例子。

假设有 a，b，c，d，e5 个系统，同时存在 A，B，C，D，E5 位用户，对每位用户偏好的项目用矩阵表示：

用户 A：偏好 a，b，c 系统，用矩阵表示为：

$$\begin{bmatrix}0,1,1,0,0\\1,0,1,0,0\\1,1,0,0,0\\0,0,0,0,0\\0,0,0,0,0\end{bmatrix}$$

用户 B：偏好 a，b，d 系统，用矩阵表示为：

$$\begin{bmatrix}0,1,0,1,0\\1,0,0,1,0\\0,0,0,0,0\\1,1,0,0,0\\0,0,0,0,0\end{bmatrix}$$

用户 C：偏好 a，d 系统，用矩阵表示为：

$$\begin{bmatrix}0,0,0,1,0\\0,0,0,0,0\\0,0,0,0,0\\1,0,0,0,0\\0,0,0,0,0\end{bmatrix}$$

用户 D：偏好 b，c，e 系统，用矩阵表示为：

$$\begin{bmatrix}0,0,0,0,0\\0,0,1,0,1\\0,1,0,0,0\\0,0,0,0,0\\0,1,0,0,0\end{bmatrix}$$

用户 E：偏好 a，e 系统，用矩阵表示为：

$$\begin{bmatrix}0,0,0,0,1\\0,0,0,0,0\\0,0,0,0,0\\0,0,0,0,0\\1,0,0,0,0\end{bmatrix}$$

将 A，B，C，D，E 矩阵全部相加之后可得矩阵 $\boldsymbol{S}$，$\boldsymbol{S}$［i］［j］则表示同时对系统 i 和系统 j 都偏好的用户数量。

$$S=\begin{bmatrix}0,2,1,2,1\\2,0,1,1,1\\1,2,0,0,0\\2,1,0,0,0\\1,1,0,0,0\end{bmatrix}$$

得到相似度矩阵之后，通过公式（7）计算用户 a 对系统 i 的推荐度：

$$Recommdsys(a,\ i) = \sum_{i \in \mathbf{N}(u) \cap S(i,\ k)} wij \qquad (9)$$

公式（9）中表示当前用户的偏好集合，S（i，k）表示与系统 i 比较相似的 K 个系统的集合，w_{ij} 是系统 i 与系统 j 的相似度。将该推荐度从大到小排列，采用 TOP－N 的方式取前 N 个系统推荐给用户。

3.3　基于 USDR 的云推荐算法运行过程

为了达到更好的用户体验，为用户提供个性化的推荐服务，基于 USDR 模型运行过程如图 2 所示，首先根据用户注册数据为用户建模，其次为平台中每个系统进行建模，当模型构建完成之后，分析用户注册数据中的基础属性数据，计算出基础属性相似度，再算出用户偏好属性相似度，最后同理算出基于系统的云推送推荐算法，最终为用户推送推荐数据。

通过分别计算用户和系统数据的推荐度会导致结果比较粗糙，为了使得云推荐算法更加精确，将用户数据推荐度加入到系统数据推荐度中，得出综合推荐度列表，将使推荐度的结果更加准确和方便，更加方便于下一步的云推送。如何使用基于 USDR 模型的云推荐算法得出用户推荐度列表的具体流程如图 3 所示。

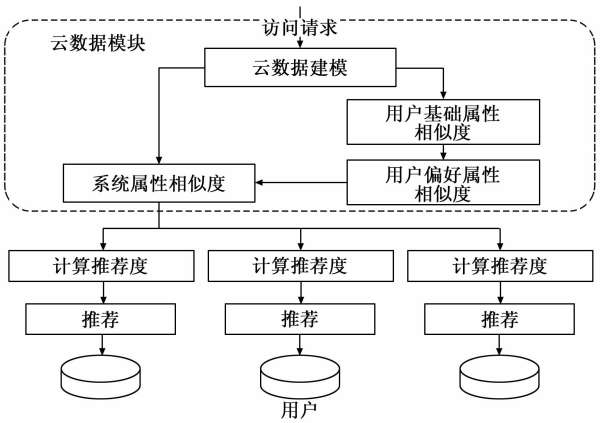

图 2　基于 USDR 模型运行过程

1）查看用户历史记录数据表，若用户的历史数据为空，则说明为新注册用户，那么就执行步骤 2），否则执行步骤 5）；

2）查看用户基础数据中的好友表，若有好友，则执行步骤 3），若无，则执行步骤 4）；

3）使该用户分别与每位好友分别用公式 D_{A-B} 进行计算，得出相似度，查看相似度在设定的权重值内的用户与该用户关系最密切的好友，执行步骤 4）；

4）使用公式（4）计算所有在权重值范围内的好友的偏好推荐度 $Recommend_{user}$，加入用户推荐列表中，执行步骤 5）；

5）使用公式（7）计算的历史数据表中每个系统的推荐度 $Recommend_{System}$，将这些系统放入推荐列表，执行步骤 6）；

6）将步骤 4）和步骤 5）中的 $Recommend_{user}$ 和 $Recommend_{System}$ 分别平方，再求和开根号得出综合推荐度：

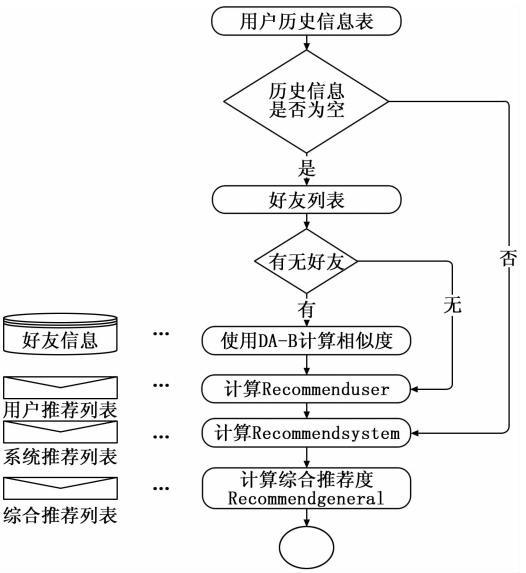

图 3　基于 USDR 模型的云推荐算法运行流程

$$Recommend\,\text{general} = \sqrt{Recommend\,\text{user}^2 + Recommend\,\text{system}^2} \qquad (10)$$

7）根据综合推荐度，加入到综合推荐度列表。

4　案例分析与实验

本文使用云推荐方法在安卓和 iOS 中进行了测试，云数据来源于成绩系统，工资系统和微影视系统，采集到数据后将数据的主要权重值分为：用户权限，用户登录时间，用户发布/订阅的模式（一对多/一对一），用户登录数量，传输数据量。权重 w 是经过综合考虑而确定的。目前数据物流云推送平台中存在的系统如图 4 所示。

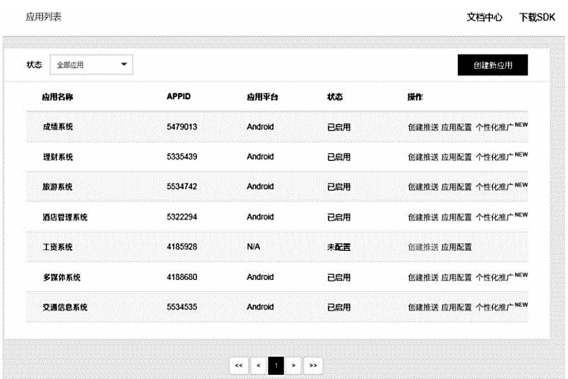

图 4　用户登录平台选择偏好的系统

用户在手机端进行注册，主要需要将用户基础数据和偏好信息填写完毕，方便接下来的云推荐系统给用户推荐个性化内容。当用户基础数据、偏好数据和系统数据都进行绑定之后，得到如图 5 所示的界面，此时平台已经根据云推荐算法将推荐数据放入推荐列表中，等待下一步的云推送。

5　算法有效性分析

为衡量本文提出的 USDR 云推荐算法的能力，从算法

图 5 用户功能页面

效率、系统数量、平均传输率，通信率，静置时流量等方面来对算法有效性进行评估。

首先测试单机中的普通推荐算法和云推送平台中的推荐算法进行比较，单机使用 Windows8 64 位操作系统，8GB 内存，10 台虚拟机同样使用 Windows8 64 位操作系统，8GB 内存，分别计算虚拟数据量为 10～100 万的数据量。

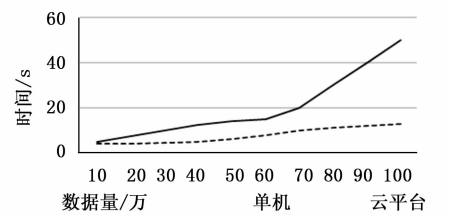

图 6 单机与云平台中推荐算法效率对比

为了得到当在数据量相同时运算速度与虚拟机数据的关系，实验中使用 50 万的数据量，分别测试虚拟机数量为 5～10 台时云推荐算法运行的效率。

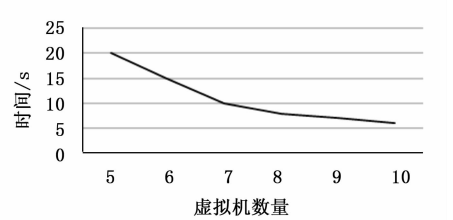

图 7 虚拟机数量不同对推荐算法的影响

平均传输率是指数据传输平均的“倍速”数。单倍数传输时，即可记为 1 倍速，普通推送的平均传输率为 10 倍速，在数据量相同时，结果如表 2 所示。

表 2 平均传输率数据表 %

	云推荐算法	普通推荐算法
平均传输率	100	10
最好值	0.01028	0.02489
最差值	0.01523	0.02489
平均值	0.01308	0.02489

通信率是指单位时间内用户与云推送平台的通信次数，测试用户是否愿意使用该平台进行数据推送，并同时测试了在通信次数高的时候会不会产生其他问题（结果如图 8 所示），普通的推荐算法通信率基本不变是由于在推送任务队列消息的整个过程中一直都会向服务器发送请求，而本文提出的云推荐方法处于信息收集阶段，随着系统的运行，任务数量增多，优势就逐渐显示出来，在任务数越多时，花费的通信量反而变少。

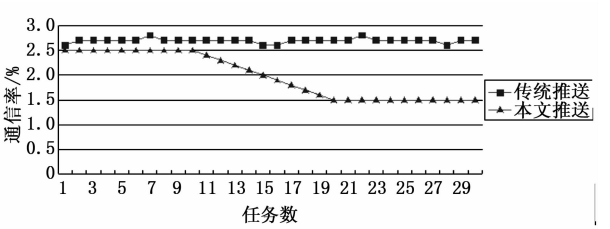

图 8 通信率变化图

静置时流量是指在手机静置时由于推送而产生的额外流量，测试云推送平台是否会因为通信率的改善而产生大量流量，分别使用云推送平台和传统推送平台进行测试（如表 3 所示），实验结果表明云推送平台在移动设备静置时间较长情况下流量消耗少于传统推送平台。

表 3 静置时流量对比

场景	云推送平台	传统推送平台
流量/KB	实测值	实测值
静置(8 小时)	6.8	8.46
静置(12 小时)	10.66	12.97
重连次数	0	0

6 结束语

本文针对传统的推送方式在推送多源异构数据时遇到的效率低，实时性差等问题，设计了面向多源异构数据的云推送平台来满足云推送环境，并通过 USDR 模型解决了多源异构数据推送问题，满足了用户需求。

然而，该平台能否满足所有的用户需求，能否供海量用户使用还需要进行验证，云推送平台本身的性能提升以及各种演化方式将是本文下一步的研究内容。相信随着这些关键问题的攻破，面向多源异构数据的云推送平台将为用户带来更好的推送体验。

参考文献：

[1] Han H，Tokuda T. A Layout－Independent Web News Article Contents Extraction Method Based on Relevance Analysis [A]. International Conference on Web Engineering [C]. Springer－Verlag，2009：453－460.

[2] Hu M，Luo J，Wang Y，et al. Practical resource provisioning and caching with dynamic resilience for cloud－based content distribution networks [J]. IEEE Transactions on Parallel & Distributed Systems，2014，25 (8)：2169－2179.

(下转第 237 页)