

Spark 框架下基于对比散度的网络服务 TLRBM 推荐算法

那 勇

(吉林省远程教育技术科技创新中心 吉林 长春 130022)

(吉林广播电视大学远程教育技术中心 吉林 长春 130022)

摘 要 为提高 Web 服务推荐算法的可靠性和时效性,提出一种 Spark 框架下基于对比散度的网络服务推荐算法。基于用户之间的直接信任关系,构建信任网络无向图模型,提出一种基于两层受限玻尔兹曼机的 Web 服务质量预测模型 TLRBM(Two Layers Restricted Boltzmann Machine Model),并将该模型应用于 Web 服务质量预测。为提高算法处理 Web 服务大数据的能力,采用对比散度算法 CD(Contrastive Divergence)来提高收敛速度,并采用 Spark 框架实现 TLRBM 的并行化执行,大幅度提升了 Web 服务推荐算法的计算速度。通过在 Epinions 数据集上的仿真测试,验证了该算法在 Web 服务推荐算法的可靠性和时效性上的性能优势。

关键词 Spark 框架 并行化 Web 服务 玻尔兹曼机 推荐算法 云计算 大数据

中图分类号 TP391.9 文献标识码 A DOI:10.3969/j.issn.1000-386x.2019.08.049

TLRBM RECOMMENDATION ALGORITHM FOR NETWORK SERVICES BASED ON CONTRAST DIVERGENCE UNDER SPARK FRAMEWORK

Na Yong

(Jilin Distance Education Technology Innovation Center, Changchun 130022, Jilin, China)

(Department of Distance Education Technology, Jilin Radio and TV University, Changchun 130022, Jilin, China)

Abstract To improve the reliability and timeliness of Web service recommendation algorithm, I proposed a recommendation algorithm based on contrast divergence in Spark framework. Based on the direct trust relationship between users, I constructed the undirected graph model of trust network, and proposed a prediction model based on two-layer restricted Boltzmann machine (TLRBM). The model was applied to the prediction of Web service quality. The Contrastive Divergence (CD) algorithm was adopted to improve the convergence speed, and it improved the algorithm ability of processing Web big data. The Spark framework was used to implement the parallel execution of TLRBM, which greatly improved the computing speed of Web service recommendation algorithm. The simulation tests on Epinions dataset verified that the performance advantages of the proposed algorithm in the reliability and timeliness of Web service recommendation algorithm.

Keywords Spark framework Parallelization Web services Boltzmann machine Recommendation algorithms
Cloud computing Big data

0 引 言

Web 服务是可通过网络互通信的计算机应用程序,由服务提供商根据用户需求开发,随着互联网用户需求不断增加,Web 服务利润增长潜力巨大。许多服务提供商倾向于结合业务和营销计划进行 Web 服

务开发,这会导致多个提供商可提供功能相似的服务。同时,Web 服务推荐过程中,服务质量取决于多种因素,如网络传输带宽、实现机制和硬件等^[1-2]。因此,用户希望从服务提供商获得理想 QoS 的 Web 服务,例如延迟、可用性和可靠性等。然而,目前几乎所有的服务提供商都关注利润,每个服务的 QoS 属性都由服务提供商预先确定,但在其推广过程中,服务提供商常常

对 Web 服务 QoS 进行夸大以吸引更多的用户。因此, QoS 合理评价是面向应用的 Web 服务推荐的重要应用之一。

协同过滤推荐是 Web 服务推荐的主要方法,主要分为两类^[3]:模型协同过滤和内存协同过滤,其中内存协同过滤采用的是 Web 服务评分矩阵方式进行项目的推荐和预测,常用到聚类分析算法,但是在每次计算中都需要对这个评分矩阵进行计算,算法的通用性不强;而模型协同过滤算法则是根据 Web 服务评分矩阵进行评价模型的演化,在后续的 Web 服务推荐过程中采用的就是模型预测方式取代原有评分矩阵方式,有助于提高算法计算普适性和精度。评分矩阵的模型构建方式很多,例如贝叶斯方法、马尔可夫方法、受限玻尔兹曼机 RBM (Restricted Boltzmann Machine) 方法等^[4]。相对而言,近年来 RBM 方法得到了学者们更多的研究,其采用无向图方式进行模型构建,因此可得到鲁棒性更强的评价模型,其采用可见层和隐藏层两层模型形式。已有很多文献研究了 RBM 方法应用于推荐系统的实现方式,例如文献[5]提出一种基于 RBM 方法的推荐问题求解策略,利用隐藏层权重共享实现算法性能的提升。文献[6]同样基于 RBM 方法设计了一种推荐问题求解策略,其将实值推荐问题表示为多维 0-1 向量表示问题,对 RBM 方法进行了模型扩维,虽然提高了推荐算法的精度,但降低了算法的执行效率。通过分析已有文献,虽然都针对某些方面进行了改进和实现提高了算法的性能,但是在互联网大数据时代,传统的 Web 推荐方法在实现过程中存在两个方面的问题:(1) 算法的自学习更新问题。现有算法模型多是固定式模型,其对于特定的 Web 服务推荐形式有效,但是通用效果不佳,实际应用效果不理想。Web 服务推荐某些应该根据时间和需求等因素的变化具备自学习的能力,这样才能提高算法的适应性和应用价值。(2) 具有学习能力的算法,比如神经网络算法等,虽然具备一定的学习性能,但是算法的计算过程非常复杂,算法的时效性无法得到有效满足。对大数据的处理,如何采用并行计算方式进行模型效率提升问题,提高算法的执行效率,是提升算法应用价值的另一个重要因素。这两个问题是制约 RBM 方法在 Web 服务推荐过程中的应用成功与否的关键。

为了克服上述缺点,本文提出一种基于 Spark 框架下的基于两层受限玻尔兹曼机的 Web 服务质量预测模型,并基于用户之间的直接信任关系,提出了一种基于受限玻尔兹曼机的 Web 服务质量预测模型。同时为提高算法处理 Web 服务大数据的并行化执行能力,采用了对比散度算法来提高收敛速度,并考虑了大

数据情形采用 Spark 框架实现 TLRBM 模型的快速执行算法,大幅度提升了 Web 服务推荐算法的计算速度。

1 理论介绍

1.1 受限玻尔兹曼机

受限玻尔兹曼机模型结构的无向二部图如图 1 所示,具有 m 个可见单元 $V = (v_1, v_2, \dots, v_m)$,以及 n 个隐藏单元 $H = (h_1, h_2, \dots, h_n)$ 。其中: V 表示可观测数据, H 是观测变量之间的相关性。与传统玻尔兹曼机相比,RBM 模型结构中同层节点之间没有关联性,其可见层和隐藏层的联合构型 (V, H) 具有能量函数:

$$E(v, h) = - \sum_{i=1}^m \sum_{j=1}^n v_i w_{ij} h_j - \sum_{i=1}^m v_i b_i - \sum_{j=1}^n c_j b_j \quad (1)$$

式中: v_i 和 h_j 分别是可见单位 i 和隐单元 j 的二进制状态; b_i 表示第 i 个可见节点的偏差, c_j 表示第 j 个可见节点的偏差; w_{ij} 是单元 v_i 和 h_j 之间的边缘相关联的实值权重。

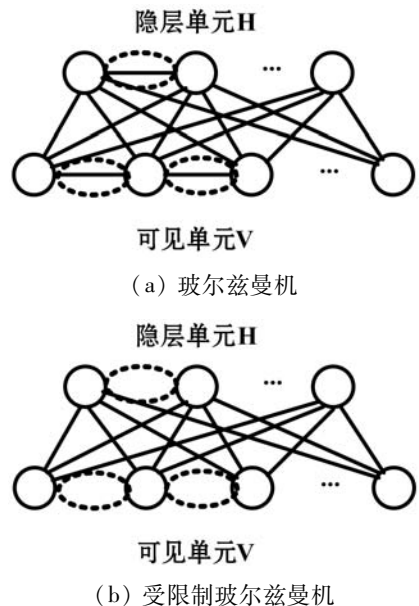


图 1 受限制玻尔兹曼机与传统玻尔兹曼机对比

该模型下的联合概率分布是通过 Gibbs 分布 $p(v, h) = 1/Z e^{-E(v, h)}$ 求出所有可能的可见和隐藏向量对,其中 $Z = \sum_{v, h} e^{-E(v, h)}$ 。因为隐藏单元之间没有直接联系,所以可简单的获得无偏样本数据 $\langle v_i, h_j \rangle_{data}$ 的条件概率如下:

$$p(h_j = 1 | v) = sig\left(\sum_{i=1}^m W_{ij} v_i + c_j\right) \quad (2)$$

类似地,可见单元之间没有直接联系,可获得无偏样本 $\langle v_i, h_j \rangle_{model}$ 的条件概率如下:

$$p(h_j = 1 | h) = sig\left(\sum_{i=1}^n W_{ij} v_i + c_j\right) \quad (3)$$

式中: $\text{sig}(x) = 1/(1 + e^{-x})$ 是 Logistic 函数。为了处理数据,可通过调整权重和偏差来提高节点的概率,以降低该节点的能量并提高其他节点的能量,特别是在低能量情况下。因此,它对分割函数有很大的贡献。具有权重的训练向量的对数概率的导数为:

$$\frac{\partial}{\partial W_{ij}} \log P(v) = \langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}} \quad (4)$$

式中: $\langle \dots \rangle$ 是 Gibbs 分布的期望。

式(3)可以反复计算直到样本向量均位于可见层上,隐藏层是条件独立的,并且来自 $\langle \dots \rangle$ 的无偏样本,使用对比散度(CD)学习会使结果快速收敛。

1.2 用户信任度计算

Web 服务推荐过程中需要使用到用户的评价问题,一般采用信任网络模型进行表示,该网络模型采用有向图 $G = (U, E)$ 表示,其中模型参数 U 表示信任网络模型节点集,Web 服务在网络中以节点形式进行表示; E 表示信任网络模型的边集,信任值在网络中以节点之间的边值 w_i 形式进行表达,具体如图 2 所示(以 6 个 Web 服务信任网络模型为例)。

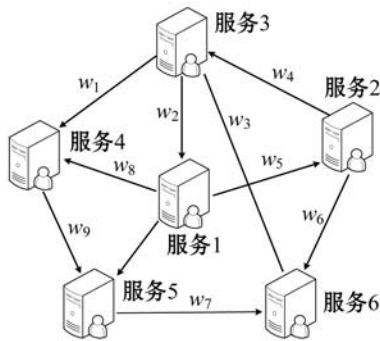


图 2 信任网络模型(6 个 Web 服务)

图 2 所示信任网络模型中节点之间的信任值一般采用 $[0, 1]$ 区间内的实数进行表示,其可表征节点之间的信任关系。其中 0 表示节点之间不存在信任关系,1 表示节点之间具有完全信任关系。实际的 Web 服务推荐过程中,节点表示的 Web 服务是真实存在的,但 Web 服务之间的关系属性值却不容易获得。

对此这里选取 Pearson 系数作为信任网络模型的 Web 服务之间的信任值,该值可对 Web 服务之间的关系属性值进行直接表达:

$$\text{sim}(u, v) = \frac{\sum_{c \in I_{u,v}} (R_{u,c} - \bar{R}_u)(R_{v,c} - \bar{R}_v)}{\sqrt{\sum_{c \in I_{u,v}} (R_{u,c} - \bar{R}_u)^2 \sum_{c \in I_{u,v}} (R_{v,c} - \bar{R}_v)^2}} \quad (5)$$

式中:参数项 $I_{u,v}$ 是 Web 服务 u 和 Web 服务 v 之间所具有的共同评分项; $R_{u,c}$ 是 Web 服务 u 对在项目 c 的评价值; \bar{R}_u 和 \bar{R}_v 分别是选取的训练集内 Web 服务 u 和 Web 服务 v 在项目上的评价均值。这种 Web 服务评

价方式的基本思想是:利用训练好的 RBM 模型进行 Web 服务评价,如果两个 Web 服务之间存在关联,则对这两个 Web 服务的所有项目进行模型的预测评价,并采用式(5)所定义的 Pearson 系数进行 Web 服务相似度的计算,进而可得到不同 Web 服务之间的信任值计算结果。

2 基于并行计算的 Web 服务推荐

2.1 基于两层受限玻尔兹曼机的 Web 服务质量预测模型

基于两层受限玻尔兹曼机的 Web 服务质量预测模型 TLRBM 被解释为具有对称连接的随机神经网络,其中图形模型中的节点由两层二进制变量组成。其中,隐藏层(h)表示对于不同服务用户具有不同 QoS 值的随机二进制特征,可见层(V)表示服务用户、Web 服务项和 QoS 值,但存在问题是有效地处理缺失的 QoS 值。可见层通过对称加权连接与隐藏层连接。用 Gibbs 分布给出在可见层和隐层上的联合概率分布。

令 V 表示 m 个可见单元 v_1, v_2, \dots, v_m 的向量,其由 m 服务用户和 n 个 Web 服务项目组成,称为用户项目矩阵。该矩阵 $r_{i,k}$ 中的每个条目表示服务用户 i 在 Web 服务项 k 上观察到的 QoS 值的向量(例如:响应时间、故障率等)。令 F 表示 h 个隐藏单元 $h = (h_1, h_2, \dots, h_n)$ 的数量,它可代表随机二进制特征,这些特征对于不同的用户具有不同的值。

每个服务用户被认为是 TLRBM 的一个单独的训练案例,它有一个对称地连接到一组二进制隐藏层的 softmax 可视层。然后,每个隐藏单元可以学习建模不同值之间的显著依赖关系。每个 TLRBM 都有一个单独的训练案例,但是所有相应的权重和偏差都被绑定在一起,因此如果两个用户具有相同的值,那么它们的两个 TLRBM 必须在该 Web 服务项和隐藏层的 softmax 可视层之间使用相同的权重。

根据图 3 所示模型,对每个观察可见层的列使用条件多项式分布 softmax,对隐藏层用户特征 h 使用条件多项式分布。因此,可将式(2)和式(3)推广如下:

$$p(v_i^k = 1 | h) = \frac{\exp(b_i^k + \sum_{j=1}^F h_j W_{ij}^k)}{\sum_{l=1}^k \exp(b_l^k + \sum_{j=1}^F h_j W_{lj}^k)} \quad (6)$$

$$p(h_j = 1 | V) = \text{sig}\left(\sum_{i=1}^m W_{ij}^k v_i^k + b_j\right) \quad (7)$$

由于在一个层的变量之间具有独立性,因此可简

单使用 Gibbs 采样进行数据采集,其具有并行采样特性。因此,Gibbs 采样可利用以下两个步骤完成:基于 $p(h|v)$ 对隐藏层的新状态 h 进行采样,以及基于 $p(v|h)$ 对可见层的状态 v 进行采样。边际分布可以通过求和可见向量 \mathbf{V} 的所有可能值进行计算:

$$p(\mathbf{V}, h) = \sum_h \frac{\exp(-E(\mathbf{V}, h))}{Z} \quad (8)$$

式中: $Z = \sum_{v^k} \exp(-E(\mathbf{V}^k, h))$ 是一个归一化常数, E 是能量函数,可定义为:

$$E(\mathbf{V}, h) = - \sum_{i=1}^m \sum_{j=1}^F \sum_{k=1}^K v_i^k W_{ij}^k h_j - \sum_{i=1}^m \sum_{k=1}^K v_i^k b_i^k - \sum_{j=1}^F h_j b_j \quad (9)$$

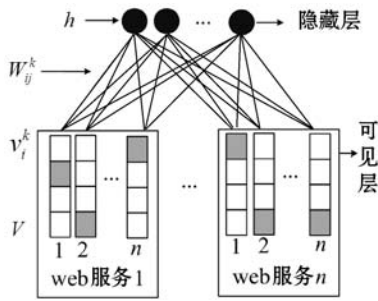


图3 基于两层受限玻尔兹曼机的 Web 服务质量预测模型 (TLRBM)

2.2 模型学习过程

首先,提出在无条件的 TLRBM 模型中的学习方法。尽管与图 1 中的模型相比,TLRBM 模型中的可见单元的激活功能已经改变,但是其对于式(4)的学习过程是一致的。唯一区别是 Gibbs 采样仅用于在非丢失的 QoS 值上重构分布。因此,可依据式(4)获得如下形式参数更新:

$$\Delta W_{ij}^k = \varepsilon \frac{\partial \log p(\mathbf{V})}{\partial W_{ij}^k} = \varepsilon (\langle v_i^k h_j \rangle_{\text{data}} - \langle v_i^k h_j \rangle_T) \quad (10)$$

$$\begin{cases} \Delta b_i^k = \varepsilon (\langle v_i^k \rangle_{\text{data}} - \langle v_i^k \rangle_T) \\ \Delta b_j = \varepsilon (\langle h_j \rangle_{\text{data}} - \langle h_j \rangle_T) \end{cases} \quad (11)$$

式中: ε 是缺失 QoS 值的学习预测值; $\langle v_i^k h_j \rangle_{\text{data}}$ 是关于数据分布的期望。它使用式(6)从训练数据集中定义具有值 k 和特征 j 的 Web 服务 i 的频率; $\langle v_i^k h_j \rangle_T$ 表示运行 Gibbs 采样器的样本分布。执行 Gibbs 采样的 T 个步骤通常在学习开始时设置为 1,并且随着学习收敛而增加。CD 学习相对有效,并且大大减少了用于学习的估计的方差。当样本向量被加持在可见层上时,隐藏层是条件独立的,来自 $\langle \dots \rangle_{\text{data}}$ 的采样数据可以计算方程(6),计算 $\langle \dots \rangle_T$ 将变得更简单,而不是为了达到平衡而随机地更新层,现在可以通过 Gibbs 采样来更新层,这包括更新相互平行运行的链中的可见层和隐藏层。所幸的是,CD 学习可以快速收敛,即使在一

次迭代中也能达到均衡分布。

其次,学习条件 TLRBM 模型与基本模型相似。但是在条件 TLRBM 模型中,有一个额外的学习参数 D :

$$\Delta D_{ij} = \varepsilon (\langle h_j \rangle_{\text{data}} - \langle h_j \rangle_T) r_i \quad (12)$$

参数 Δc_i 的更新公式为:

$$\Delta c_i = \varepsilon (\langle v_i \rangle_{\text{data}} - \langle v_i \rangle_{\text{recon}}) v_i \quad (13)$$

再次,为了进行推荐,通过在 softmax 单元上使用推理过程来预测缺失 QoS 值,然后对所有缺失 QoS 值执行单个 Gibbs 采样步骤。给定可见向量 \mathbf{V} 的值,可预测查询服务 q 的 QoS 值,其相对于隐藏层的数量具有时间线性关系:

$$p(v_q^k = 1 | \mathbf{V}) \propto \exp(-E(v_q^k, \mathbf{V}, h)) \propto \exp(v_q^k b_q^k) \prod_{j=1}^F \sum_{h_j \in \{0,1\}} \exp(\sum_i v_i^k h_j W_{ij}^k + h_j b_j) \quad (14)$$

最后,可对 K 值归一化以获得概率 $p(v_q = k | \mathbf{V})$,以及期望值 $E[v_q]$ 。进而可预测 n 个服务 q_1, q_2, \dots, q_n 的 QoS 值,并可计算 $p(v_{q_1}^{k_1} = 1, v_{q_2}^{k_2} = 1, \dots, v_{q_n}^{k_n} = 1 | \mathbf{V})$ 。

2.3 TLRBM 服务推荐模型

如上所述,CD 学习算法使用了运行吉布斯的样本分布,迭代(T 步骤)直到收敛为止。使用称为动量法的启发式方法,其思想是在计算迭代 t 上的更新时考虑在迭代 $t-1$ 处计算的更新,并且采取以下形式:

$$\Delta W_{ij}^k(t) = \varepsilon (\langle v_i^k h_j \rangle_{\text{data}} - \langle v_i^k h_j \rangle_T) + a \Delta W_{ij}^k(t-1) \quad (15)$$

式中: $a \in [0, 1]$ 。Web 服务推荐的 TLRBM 模型算法学习过程见算法 1 所示。

算法 1 Web 服务推荐的 TLRBM 模型学习过程

1. 初始化:利用标准正态分布得到 $\Delta W_{ij}^k, \Delta b_i^k, \Delta b_j, \Delta D_{ij}$;
2. 令 $n=0$,在时间 n 上计算预测误差 E_n ;
3. Repeat
4. for all S 集内样本 do
5. for all 当前样本中的 u do
6. 基于可见层计算隐藏概率贡献累积:
 $SumW \leftarrow \Delta W_{ij}^k, SumD \leftarrow \Delta D_{ij}$ (参见式(10) - 式(12));
7. 计算隐藏状态的概率:
 $p_j = p(h_j = 1 | \mathbf{V}, r)$ (参见式(14));
记录样本 $v_i^k b_j, v_i^k, b_j$;
8. 吉布斯抽样:令 $T=0$;
Repeat
 $p' = p(v_i^k = 1 | p_j)$ (参见式(6) - 式(7));
在开始另一次迭代之前计算误差;
Until(++ step $T < T$ steps)
9. endfor
10. endfor
11. 对比贡献积累差异:
计算第一个样本的平均值: $\langle v_i^k \rangle_{\text{data}}, \langle v_i^k \rangle_{\text{recon}}, \langle h_j \rangle_{\text{data}}$;

计算第二个样本的平均值： $\langle v_i^k p_j \rangle_T \langle v_i^k \rangle_T \langle h_j \rangle_T$;

12. 更新权重及参数:

利用式(10)更新 ΔW_{ij}^k ;

利用式(11)更新 Δb_i^k ;

利用式(12)更新 ΔD_{ij} ;

利用式(13)更新 Δc_i ;

13. endfor

14. $n++$; 在时间 n 上计算预测误差 E_n ;

15. Until($E_{n-1} - E_n > \varepsilon$;

2.4 基于 Spark 并行化实现

随着现代互联网科技的快速发展,Web 服务数量呈现出大幅度增长趋势,以面向 Web 的运行 nginx 系统的计算机数量为例,据相关统计,2010 年大约有不到 5 万台数量,2014 年的数据大约在 40 万台左右,而到了 2017 年,数量则大幅度增加到 140 万台左右,增长速度呈现出加速增长态势。但是,目前采用单线程处理方式进行 Web 服务推荐已经无法与现代 Web 服务推荐应用实际相适应,算法的执行效率受到很大的制约。Spark 并行化模型是一种采用内存计算方式的系统,其具有开源属性便于开发利用,目的是实现大型数据并行化处理。Spark 采用的是一种具有分布式结构的弹性数据结构,可将中间计算数据在内存中以缓存形式进行存储,从而省去了中间硬盘读取环节,便于计算速度的提升。

图 4 所示并行化模型是针对 TLRBM 模型设计的并行化实现方案。图 4 中的参数 $\theta = \{W, b, c, D\}$, 在 Spark 并行化实现过程中,首先将待评价 Web 服务样本进行切片,并将其分配到不同 Spark 并行分片上,在实施过程中,各分片通过并行化计算进行参数学习,同步获得不同分片上的模型参数,其中 $\Delta W_{ij}^k, \Delta c_i, \Delta b_i^k$ 和 ΔD_{ij} 的更新计算分别见式(10) - 式(14)。在 Spark 并行化模型的汇总阶段对每个模型分片计算得到的 TLRBM 模型参数进行汇总,采取求取均值的方式计算得到。

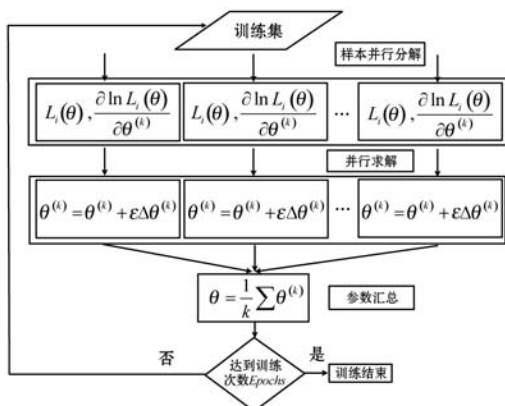


图 4 基于 Spark 并行化实现

3 实验分析

3.1 实验设置

为验证所提算法的性能,选取 Epinions 数据集作为实验对象,该数据集由 Massa 等通过爬取技术在 Epinions 网站上获得的 Web 服务数据,共含有 401 593 个 Web 服务在 139 528 个项目上的测评结果,参见文献[13]。设置数据集中前 80% 的 Web 服务数据构建算法模型的训练集,剩余的 20% Web 服务数据构建算法模型的测试集。对 Epinions 中的 Web 服务数据进行优化处理,对于不存在关联属性的 Web 服务数据进行删除,可得到具有 31 932 个 Web 服务在 78 893 个项目上的测评结果。

实验过程中,设定 RBM 模型中隐藏单元的节点数量为 160,模型训练过程的迭代次数上限是 $Epochs = 100$ 或者收敛精度设置为 $\zeta = 1e - 5$ 。实验硬件配置: CPU 为 i5-6400k 3.0 GHz,内存是金士顿 ddr3-1600K,系统为 Windows 10 旗舰版,仿真平台选取 MATLAB 2013a。

当前 Web 服务推荐系统中大多使用推荐精度作为算法推荐效果的评价指标,其定义为模型评价结果与用户真实评价结果之间的误差均值绝对值指标(MAE 指标),或是根均方差指标(RMSE 指标)。其中,MAE 指标主要评价计算推荐结果与用户真实评价结果的误差均值:

$$MAE = \frac{\sum_{(u,i) \in R_{test}} |R_{u,i} - \hat{R}_{u,i}|}{|R_{test}|} \quad (16)$$

另一评价指标 RMSE 指标的具体定义如下:

$$RMSE = \sqrt{\frac{\sum_{(u,i) \in R_{test}} |R_{u,i} - \hat{R}_{u,i}|^2}{|R_{test}|}} \quad (17)$$

式中: R_{test} 是 Web 服务推荐系统模型的测试集; $R_{u,i}$ 是 Web 服务 u 在项目 i 上实际使用评价价值; $\hat{R}_{u,i}$ 是 Web 服务 u 在项目 i 上的预测使用评价价值; $|R_{test}|$ 是测试集的规模参数,表征数据集中 Web 服务数据的数量。

3.2 结果分析

选取集群节点的数量作为评估参数,考察集群节点数量变化对于算法计算性能的影响。集群节点的数量变化区间是 $[1, 10]$,评价指标选取算法计算时间和加速比两项指标,实验结果如图 5 所示。

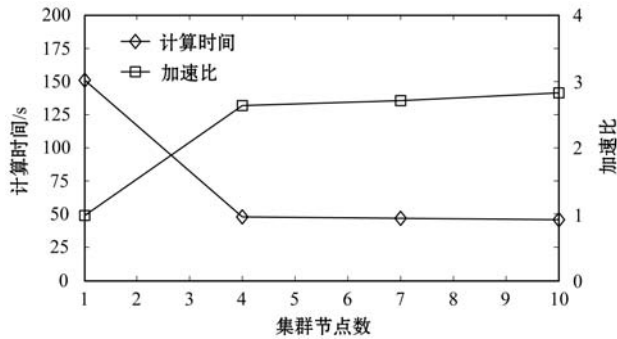


图5 集群节点数实验影响

根据图5所示实验结果,横坐标为实验选取的计算集群节点数量,纵坐标是算法运行时间或计算加速比。根据实验结果可知,随计算节点数量的增加,算法的计算时间呈现出单调下降趋势,而计算加速比则呈现出单调加速趋势。同时也可看出,随着集群节点数量的增加,算法的计算时间降低幅度和加速比增加幅度逐渐趋缓,主要原因是随着节点的增加,节点之间的通信开销逐渐增加,导致算法的计算性能增加幅度受到制约,因此对于本文选取的实验对象选取计算节点为4是最为合适的设置方式。

下面,对算法的推荐精度和计算效率性能进行对比实验,对比算法选取:文献[14]、文献[15]以及未采用Spark并行化实现的受限玻尔兹曼机算法(RBM)。表1给出选取不同收敛精度情况下算法的计算效率对比情况。

表1 计算效率对比 $\times 10^5$

收敛精度	文献[14]	文献[15]	RBM	本文算法
$\varepsilon = 1e-1$	4.3	5.9	6.1	1.8
$\varepsilon = 1e-2$	5.7	7.3	8.2	2.1
$\varepsilon = 1e-3$	7.3	9.4	11.6	2.9
$\varepsilon = 1e-4$	9.5	12.6	15.7	3.8
$\varepsilon = 1e-5$	14.6	18.9	22.4	4.9

根据图5实验结果,本文算法选取集群节点数为4,另外,这里选取的收敛精度指标并不是式(16) - 式(17)所定义的MAE和RMSE指标,而是式(9)所定义的能量函数指标。

根据表1结果可知,随着收敛精度设定参数的增加,集中算法的计算时间均呈现出增加趋势,因为收敛精度越高算法收敛过程所需的迭代步数越多,因此计算时间越大。同时,在几种算法对比中,本文算法的计算时间最少,这是因为采用了并行计算方式,实现了多个计算节点的并行执行,因此本文算法更加适用于互联网大量服务推荐过程,并且对于不同的数据流,可通

过扩展集群节点方式进行计算性能改善,具有更高的可扩展性和更高的实际应用价值。

设置迭代次数上限是Epochs = 100,文献[14]、文献[15]、RBM算法以及本文并行算法的MAE和RMSE指标的实验结果如图6和图7所示。

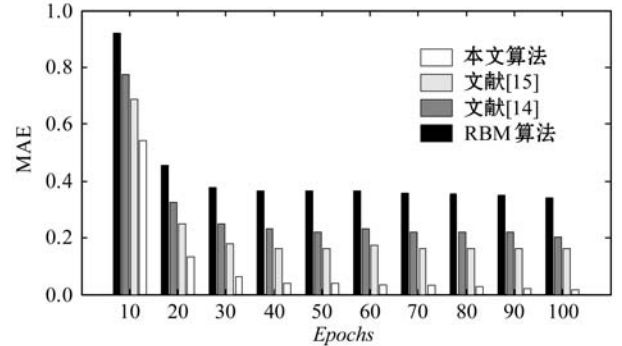


图6 MAE指标对比结果

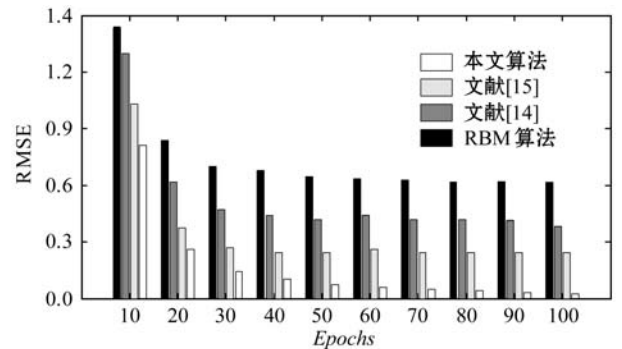


图7 RMSE指标对比结果

根据图6 - 图7实验结果可知,随着迭代次数增加,几种算法的MAE指标和RMSE指标均呈现出逐渐收敛趋势。从收敛精度和速度上看,本文算法的Web服务推荐性能要显著优于文献[14]、文献[15]以及RBM算法的推荐性能。另外,文献[15]优于文献[14]和RBM算法的推荐性能。其中RBM算法的推荐性能最差,收敛速度非常缓慢,这也从侧面印证了本文采用的Spark并行化实现方式的有效性。

4 结语

本文提出一种Spark框架实现的基于两层受限玻尔兹曼机Web服务受限玻尔兹曼机推荐模型,主要贡献如下:(1)引入一种新的模型,通过有效的学习和推理过程从服务提供者和客户端可实现Web服务QoS的实时监控。(2)为了有效地处理大数据集,采用了CD学习来提高收敛时间,特别是提出采用Spark并行化实现算法的有效拓展。下一步研究方向,主要是基于所提TLRBM模型,完成不同业务用户QoS值信息采集以及主动用户QoS预测系统的构建。

参 考 文 献

- [1] Senthil K S, Margret A S. QoS-Based Concurrent User-Service Grouping for Web Service Recommendation[J]. Automatic Control and Computer Sciences, 2018, 52(3):220-230.
- [2] Chen W H, Paik I, Yen N Y. Discovering internal social relationship for influence-aware service recommendation[J]. Multimedia Tools and Applications, 2017, 76(18):18193-18220.
- [3] Chen L, Wu J, Zheng Z B, et al. Modeling and exploiting tag relevance for Web service mining[J]. Knowledge and Information Systems, 2014, 39(1):153-173.
- [4] Huang Z Z, Li T X, Xiao S. Research on Library Recommendation Reading Service System Based on Adaptive Algorithm[J]. Wireless Personal Communications, 2018, 102(2):1963-1977.
- [5] Jayapriya K, Mary N A B, Rajesh R S. Cloud Service Recommendation Based on a Correlated QoS Ranking Prediction[J]. Journal of Network and Systems Management, 2016, 24(4):916-943.
- [6] Yu C Y, Huang L P. A Web service QoS prediction approach based on time-and location-aware collaborative filtering[J]. Service Oriented Computing and Applications, 2016, 10(2):135-149.
- [7] Karimian K, Ashtiani M, Azgomi M A. An approach based on the transferrable belief model for trust evaluation in web services[J]. Soft Computing, 2018, 22(21):7293-7311.
- [8] 唐明董, 张婷婷, 杨亚涛, 等. 基于因子分解机的质量感知 web 服务推荐方法[J]. 计算机学报, 2018, 41(6):1080-1092.
- [9] Kim J, Lee D, Chung K Y. Item recommendation based on context-aware model for personalized u-healthcare service[J]. Multimedia Tools and Applications, 2011, 71(2):855-872.
- [10] 胡堰, 彭启民, 胡晓惠. 一种基于隐语义概率模型的个性化 web 服务推荐方法[J]. 计算机研究与发展, 2014, 51(8):1781-1793.
- [11] 李鸿超, 刘建勋, 曹步清, 等. 融合多维信息的主题自适应 Web API 推荐方法[J]. 软件学报, 2018, 29(11):3374-3387.
- [12] Christou I T, Amolochitis E, Tan Z H. AMORE: design and implementation of a commercial-strength parallel hybrid movie recommendation engine[J]. Knowledge and Information Systems, 2016, 47(3):671-696.
- [13] Massa P, Avesani P. Trust-aware boots trapping of recommender system[C]//Proceedings of the ECAI work shop on recommender systems. RicadelGarda, Italy, 2006:29-33.
- [14] Wang P, Chao K M, Lo C C. Satisfaction-based Web service discovery and selection scheme utilizing vague sets theory[J]. Information Systems Frontiers, 2015, 17(4):827-844.
- [15] Yoo H, Chung K. Mining-based lifecare recommendation using peer-to-peer dataset and adaptive decision feedback[J]. Peer-to-Peer Networking and Applications, 2018, 11(6):1309-1320.
- ~~~~~
- (上接第 287 页)
- [9] Wiki-Pedia. ID3-algorithm[DB/OL]. 2018-7-6. https://en.wikipedia.org/wiki/ID3_algorithm#See_also.
- [10] 卢东标. 基于决策树的数据挖掘算法研究与应用[D]. 武汉:武汉理工大学, 2008.
- [11] 颜会娟, 秦杰. 基于非线性 SVM 模型的木马检测方法[J]. 计算机工程, 2011, 37(8):121-123.
- [12] Harrington P. Machine learning in Action[M]. 李锐, 李鹏, 曲亚东, 等译. 北京:人民邮电出版社, 2013.
- [13] Liu Bing. Web Data Mining[M]. 俞勇, 薛贵荣, 韩定一, 译. 北京:清华大学出版社, 2009.
- [14] Alpaydin E. Introduction to Machine Learning[M]. 范明, 咎红英, 牛常勇, 译. 北京:机械工业出版社, 2009, 6.
- [15] 咸云浩, 张恒德, 谢永华, 等. 多元逐步回归与卡尔曼滤波法在霾预报中应用[J]. 系统仿真学报, 2018, 30(4):1482-1489.
- [16] 刘丛山, 李祥宝, 杨煜普. 一种基于近邻元分析的文本分类算法[J]. 计算机工程, 2012, 38(15):139-141.
- [17] Zhou H T, Chen J, Dong G M, et al. Bearing fault recognition method based on neighbourhood component analysis and coupled hidden Markov model[J]. Mechanical Systems and Signal Processing, 2016, 66/67:568-581.
- [18] 邓帅. 基于改进贝叶斯优化算法的 CNN 超参数优化方法[J/OL]. 计算机应用研究, 2019, 36(7). [2018-08-02]. <http://kns.cnki.net/kcms/detail/51.1196.TP.20180412.0812.030.html>.
- [19] Rasmussen C E, Williams C K I. Gaussian Processes for Machine Learning[M]. MIT Press, 2005.
- [20] Shahriari B, Swersky K, Wang Z, et al. Taking the Human Out of the Loop: A Review of Bayesian Optimization[J]. Proceedings of the IEEE, 2016, 104(1):148-175.
- [21] 崔佳旭, 杨博. 贝叶斯优化方法和应用综述[J]. 软件学报, 2018, 29(10):176-198.
- [22] 柴慧敏, 赵昀瑶, 方敏. 利用先验正态分布的贝叶斯网络参数学习[J]. 系统工程与电子技术, 2018, 40(10):219-224.
- [23] 黄良斌. 高校贫困生认定标准与认定模型研究[J]. 职业教育研究, 2012(4):11-12.
- [24] Martinez-Cantin R. BayesOpt: A Bayesian Optimization Library for Nonlinear Optimization, Experimental Design and Bandits[J]. Journal of Machine Learning Research, 2014, 15:3735-3739.