

云和声搜索算法及其在知识服务组合中的应用*

倪志伟^{1,2}, 吴昊^{1,2}, 尹道明^{1,2†}, 杨佳^{1,2}

(1. 合肥工业大学管理学院, 合肥 230009; 2. 教育部过程优化与智能决策重点实验室, 合肥 230009)

摘要: 云计算环境下的知识服务是知识与服务的融合, 为知识管理、知识创新提供了新的发展方向。针对基于服务质量(QoS)的知识服务组合优化问题, 在云计算平台下实现和声搜索算法的并行化, 提出了云和声搜索算法。将 Skyline 方法和理想点法融入到云和声搜索算法中, 对云和声搜索算法进行改进, 提高了算法的运行效率, 确保了解的有效性。实验结果表明, 改进云和声搜索算法在求解知识服务组合优化问题上取得了较好的结果, 在解的质量以及算法的性能方面均有较好的表现。

关键词: 云计算; 和声搜索算法; MapReduce; 知识服务

中图分类号: TP301.6

文献标志码: A

文章编号: 1001-3695(2013)03-0806-04

doi:10.3969/j.issn.1001-3695.2013.03.042

Cloud harmony search algorithms for optimization of composite knowledge as a service

NI Zhi-wei^{1,2}, WU Hao^{1,2}, YIN Dao-ming^{1,2†}, YANG Jia^{1,2}

(1. School of Management, Hefei University of Technology, Hefei 230009, China; 2. Key Laboratory of Process Optimization & Intelligent Decision-making, Ministry of Education, Hefei 230009, China)

Abstract: Knowledge as a service (KaaS) is the integration of knowledge and services in cloud computing, which provides a new area for knowledge management and knowledge innovation. This paper presented the harmony search (HS) algorithm to solve the composite KaaS based on quality of service (QoS), and also proposed cloud harmony search (CHS) algorithm, which parallelized HS. With introducing the method of Skyline and TOPSIS, it improved the efficiency and effectiveness. Results of experimental demonstrate the superiority of the improved cloud harmony search (ICHS) algorithm for optimization of composite KaaS in the quality of solution and algorithm performance.

Key words: cloud computing; harmony search algorithms; MapReduce; knowledge as a service (KaaS)

0 引言

随着并行计算、分布式计算以及虚拟化技术的不断进步, 云计算^[1]作为一种新兴的商业计算模型逐步发展起来。它将计算任务分布在大量计算机构成的资源池上, 使用户可以按需获取计算能力、存储空间以及各种应用服务^[2]。当前 IT 巨头 Google、亚马逊、IBM 和微软等都拥有自己的云计算平台。其中, Google 云基础设施由四个相互独立又紧密联系在一起的系统组成, 它们分别是提供海量数据存储和访问能力的 Google 文件系统 GFS、使海量信息的并行处理更简便易行的并行编程模型 MapReduce、保证分布式环境下并发操作同步问题的分布式锁服务 Chubby、使海量数据的管理和组织更方便的分布式结构化数据存储系统 Bigtable^[3]。

云计算的发展为智能算法的不断进步提供了新的思路。利用云计算系统强大的计算与存储能力, 将智能算法进行并行化, 可快速、高效地解决海量数据处理问题。和声搜索算法^[4]是一种启发式智能优化算法, 它最早是由 Geen 等人于 2001 年提出的。它模拟乐曲创作过程, 在音乐演奏中, 乐师们不断调

整各乐器音调, 以使其演奏的和声效果趋于完美。与此类似, 将任务处理过程中的所有个体类比为音乐演奏中的各乐器, 通过记忆思考、音调微调等过程创造最优和声, 即逐渐逼近最优解。和声搜索算法具有结构简单、容易实现、收敛速度快、鲁棒性好等优点; 其缺点是容易陷入局部最优, 收敛不够稳定。近年来, 和声搜索算法已成功应用于求解多种优化问题^[5,6], 如水资源优化配置、零空闲流水线调度、任务分配、水磨工序参数优化等。

本文在基于 Hadoop 技术的云计算基础平台上, 研究和声搜索算法的并行实现方法, 提出了云和声搜索算法 (cloud harmony search algorithm, CHS)。针对知识服务^[7]的组合优化问题, 构建基于服务质量 (quality of service, QoS)^[8]的动态组合优化模型, 定义云和声搜索算法的 Map 过程以及 Reduce 过程, 对该问题进行求解。并在此基础上, 对基本云和声搜索算法进行改进: 和声记忆库的初始化、相关参数的设置以及最优服务的选择等, 使其能够更好地解决知识服务组合优化问题, 提高服务组合效率及准确率。

收稿日期: 2012-07-08; **修回日期:** 2012-09-05 **基金项目:** 国家“863”计划基金资助项目 (2011AA040501); 国家自然科学基金资助项目 (71271071); 安徽省教育厅自然科学重点项目 (KJ2011A006)

作者简介: 倪志伟 (1963-), 男, 安徽桐城人, 教授, 博导, 博士, 主要研究方向为数据挖掘、知识管理、云计算等; 吴昊 (1973-), 男, 安徽合肥人, 博士研究生, 主要研究方向为智能计算; 尹道明 (1986-), 女 (通信作者), 天津人, 硕士研究生, 主要研究方向为数据挖掘、云计算等 (daomingy@126.com); 杨佳 (1987-), 女, 安徽肥东人, 硕士研究生, 主要研究方向为数据挖掘、云计算等。

1 相关工作

1.1 知识服务

面向服务的云计算为知识的管理与创新提供了新的发展方向。知识服务(knowledge as a service, KaaS)是指将知识以服务的方式提供给知识需求方,从而达到知识的交换、共享、创新以及基于知识进行任务协作和问题求解的目的,它是知识与服务的融合^[9]。近年来在实际应用中,对知识服务的需求不断趋于多样化和复杂化,功能相对单一的知识服务已无法满足知识需求方日趋复杂和多变的应用要求。因此,将多个知识服务组合成一个功能强大的、高效的服务来满足用户的应用需求已成为必然趋势。知识服务组合是将已有的多个小粒度的知识服务联合起来组成具有附加价值的大粒度的知识服务,从而满足用户在任务协作过程中各方面的需求^[10]。知识服务组合流程如图1所示。系统首先对用户提交的知识服务请求进行判断。若请求是单一任务,表明该请求服务可直接发现,不需要对其进行组合;若请求是非单一任务,则需要对任务进行分解,再分别发现与其相对应的服务,然后按照子任务序列对知识服务进行组合优化,形成一条知识服务链,提供给用户供其调用。

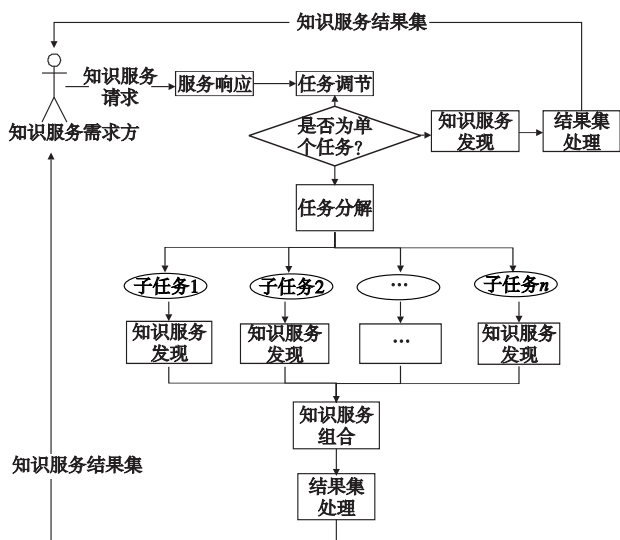


图1 知识服务组合流程

1.2 Hadoop 关键技术

Hadoop 是 Apache 软件基金会旗下的一个分布式计算开源框架,它模拟并实现了 Google 云计算的主要技术。Hadoop 框架的核心设计是分布式文件系统 HDFS(Hadoop distributed file system)^[11]和并行计算编程模型 MapReduce,它们分别是 Google 云计算核心技术 GFS 和 MapReduce^[12]的开源实现。

1.2.1 HDFS

Hadoop 分布式文件系统 HDFS 与 Google 分布式文件系统 GFS 思想类似,都是将分布在计算机集群中不同服务器上的共享文件夹组织在一起,构建成一棵目录树,从而对其进行各种操作。从用户角度看,所有共享的文件存储在一个地方,只需访问一个共享的 HDFS 根目录,就能够访问分布在网络上的文件或文件夹,而不必知道这些文件的实际物理位置。

HDFS 是一个主从结构体系,由一个 NameNode 和多个 DataNode 组成 HDFS 集群。NameNode 管理文件系统的元数据,DataNode 存储实际数据。NameNode 运行在一个专门的机器

上,集群中的其他机器各运行一个 DataNode;也可以在运行 NameNode 的机器上同时运行 DataNode,或者一台机器上运行多个 DataNode。当用户要访问一个文件时,客户端首先从 NameNode 获得组成文件的数据块的位置列表,找出数据块具体被存储在哪些 DataNode 上,然后客户端直接从 DataNode 上读取文件数据,NameNode 不参与文件的传输。

1.2.2 MapReduce 编程模型

MapReduce 是一种并行编程模型,它把运行在大规模集群上的并行计算过程抽象为 Map 和 Reduce 两个阶段。MapReduce 通过 Map 函数将数据切割成不相关的区块,分配给大量计算机进行并行处理,达到分布运算的效果,然后通过 Reduce 函数将结果汇总,输出需要的结果。简单地说,MapReduce 的思想就是任务的分解与结果的汇总。

图2描述了 MapReduce 的运行机制。接收用户输入的数据后,系统将数据分解为 n 个独立的数据块,并由 NameNode 管理分配到不同的 DataNode 上。在 Map 阶段,由多个服务器完成 Map 运算任务并生成中间结果;再对中间结果进行复制、分组、排序后进入 Reduce 阶段,由 Reduce 函数对中间结果进行归约;最后输出运算结果。此外,MapReduce 编程模型是一个递归的操作,在输出阶段得到的结果并不一定是最终结果,它也可能是下一轮 Map 运算的输入数据。

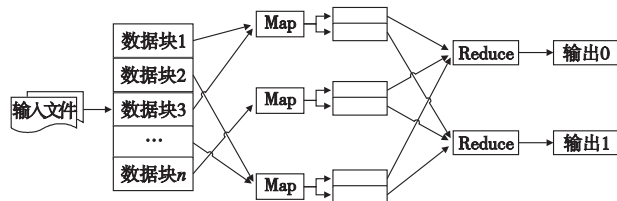


图2 MapReduce编程模型

2 云计算环境下知识服务组合优化建模

2.1 服务质量(QoS)

服务质量(QoS)作为评价知识服务满足消费者需求能力的重要指标,在知识服务的组合优化过程中占有重要的地位。知识服务的属性包括:a)功能性属性,主要描述服务所能完成的任务、作用等;b)非功能性属性,描述了该服务的基本属性,通用的 QoS 属性包括响应时间、执行成本、可用性、可靠性、吞吐量、执行的成功率、网络延迟等。

知识服务组合中所提及的服务质量(QoS)通常是指服务的非功能性属性,它包含通用的 QoS 属性以及知识服务组合中所涉及到的特定的知识服务属性。本文主要考虑知识服务的响应时间、可用性和吞吐量,其余 QoS 属性可同样利用本文方法进行处理。

2.2 知识服务组合优化建模

令 KT (knowledge task)表示业务流程, KT_i 表示该业务流程的 n 个子任务,则 $KT = \{KT_i | i = 1, 2, \dots, n\}$ 为知识服务组合需要完成的 n 个子任务的集合。设 $KC_i = \{KS_j | j = 1, 2, \dots, m_i\}$ 表示完成各子任务 KT_i 的某候选知识服务类, m_i 为该知识服务类的知识服务实例数量。为寻求最优服务,本文对 QoS 评价指标吞吐量作如下定义,响应时间及可用性的定义可参见文献[14]。

定义1 吞吐量。吞吐量指单位时间内系统处理用户的请求数。该指标反映的是服务器承受的压力,它能够说明系统

的负载能力。在没有达到性能瓶颈的情况下,吞吐率 $TP = KSU \times R/T$ 。其中, KSU 表示知识服务需求方的数量, R 表示每个服务需求方发出的请求数, T 表示性能测试所用的时间。

设满足用户要求的服务响应时间不超过 $\max RT$, 服务可用性不低于 $\min AA$, 吞吐率不小于 $\max TP$ 。基于上述定义, 云计算环境下知识服务组合优化问题可描述为带约束条件的多目标优化问题:

$$\begin{aligned} \min F(KS) &= w_1 \text{ total } RT + w_2 \text{ total } AA + w_3 TP \\ \text{s. t. } &\begin{cases} RT(KS_i) \leq \max RT \\ AA(KS_i) \geq \min AA \\ TP \geq \min TP \end{cases} \end{aligned}$$

其中: $\{w_1, w_2, w_3\}$ 为指标权重, 由服务需求方主观设定, 且满足 $\sum_{i=1}^3 w_i = 1$ 。

3 求解知识服务组合优化问题改进云和声搜索算法

云和声搜索算法利用 Map 函数来完成和声记忆库的初始化, 对和声记忆库进行思考, 随机选择音调以及音调微调, 以致新和声的产生。Reduce 负责解的寻优以及和声记忆库的更新任务。由 MapReduce 对和声搜索算法中的耗时部分进行并行化, 较大地提高了算法的运行效率。

3.1 云和声搜索算法的改进

在基本云和声搜索算法的基础上对其进行改进, 提出改进云和声搜索算法(improved cloud harmony search algorithm, ICHS)。在云计算环境下, ICHS 算法利用 Skyline^[13] 方法对和声记忆库进行初始化, 采用理想点法对最优知识服务组合进行选择, 在提高算法运行效率的同时, 确保了解的有效性。文献[14]描述了利用 Skyline 以及理想点法改进和声搜索算法的详细步骤。Skyline 方法的执行过程如图3所示。

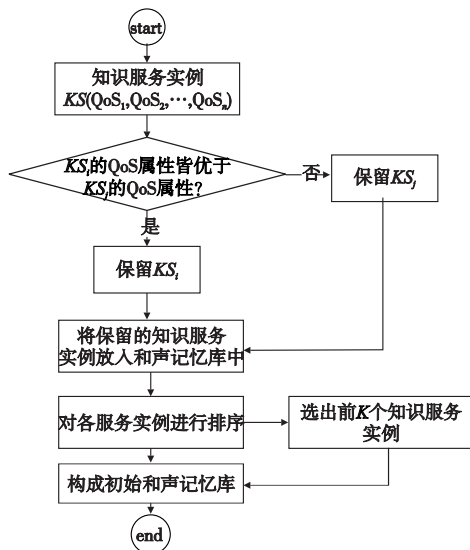


图3 Skyline方法执行过程

3.2 ICHS 的 Map 函数设计

Map 函数的主要功能是对和声记忆库进行初始化以及新的和声的产生。Map 的输入为请求的子任务相对应的知识服务类, 输入的数据 $\langle \text{key}, \text{value} \rangle$ 对是 MapReduce 框架的默认格式, 具体形式为 $\langle \text{知识服务实例 ID}, \text{QoS 属性值} \rangle$ 。每个 Map 函数都会读入一个知识服务类, 根据 QoS 评价指标及其权重, 利用改进云和声搜索算法(ICHS)选出该知识服务类中最优知识

服务实例, 并输出中间结果。输出数据记录 $\langle \text{key}, \text{value} \rangle$ 对的形式为 $\langle \text{效用函数值}, \text{list}(\text{知识服务实例 ID}) \rangle$ 。Map 函数的描述如函数1、函数2所示。

函数1 Map 函数

```

Function Map (key1, value1) {
    if (HM == null) { //初始化和声记忆库;
        Skyline();
    } else {
        for (i = 1; i <= MI; i++) {
            if (rand() >= HMCR) {
                //对各属性进行随机选择;
                x_i = x_i^min + (x_i^max - x_i^min) * rand();
            } else { //对该知识服务的各 QoS 属性进行记忆思考及音调调节;
                x_i = x_i^j (j = 1, 2, ..., HMS)
                if (rand() <= PAR) {
                    if (rand() > 0.5) {
                        x_i = x_i + FW(i) * rand();
                    } else {
                        x_i = x_i - FW(i) * rand();
                    }
                }
            }
            输出 <key2, value2>;
        }
    }
}
  
```

函数2 Skyline 函数

```

Function Skyline() {
    num = 0;
    for (i = 1; i < n; i++) {
        各 QoS 属性值;
        for (j = i; j <= n; j++) {
            if (KaaS[i][j] >= KaaS[i+1][j]) {
                //将服务存入和声记忆库
                HM[num] = KaaS[i]; num++;
            }
        }
    }
}
  
```

3.3 ICHS 的 Reduce 函数设计

Reduce 函数的主要任务是根据 Map 产生的新的和声更新记忆库, 并根据 QoS 评价指标及各知识服务类的权重计算各子任务的效用函数值, 选出最优知识服务组合, 利用理想点法找出最优知识服务实例, 输出当前最优解。Reduce 函数将 Map 函数的输出作为自己的输入, 输入的 $\langle \text{key}, \text{value} \rangle$ 对的形式为 $\langle \text{效用函数值}, \text{list}(\text{知识服务实例 ID}) \rangle$, 对当前效用函数值相同的知识服务进行归约操作, 更新和声记忆库。Reduce 函数的输出为 $\langle \text{最优效用函数值}, \text{list}(\text{知识服务实例 ID}) \rangle$ 。Reduce 函数的描述如函数3、函数4所示。

函数3 Reduce 函数

```

Function Reduce (key2,
    list (value2)) {
    minFun; HM;
    读取各子任务知识服务实例的最优解;
    if (fun < minFun) {
        minFun = fun;
        HM[最优解] = HM[i];
    }
    TOPSIS();
    更新和声记忆库;
    输出 <key3, value3>;
}
  
```

函数4 TOPSIS 函数

```

Function TOPSIS() {
    for (i = 1; i < num; i++) {
        根据各 QoS 属性值, 计算各服务与最优解之间的距离 F(s);
        if (F(s_i) < F(s_{i+1})) {
            f_best(s) = f(s_i);
        } else { f_best(s) = f(s_{i+1}); }
        输出最优服务 f_best(s);
    }
}
  
```

4 仿真实验及结果分析

本文实验在 Hadoop 平台上运行,共 2 个节点,每个节点的处理器为 Intel® Core™2 Duo CPU,2.83 GHz,内存为 2 GB,操作系统为 Windows XP。实验所需软件包括 VMware Workstation、Ubuntu, Hadoop 版本为 hadoop-0.20.0, JDK 版本为 1.6.0, Eclipse 版本为 3.3.2。实验中 2 台主机节点均作为从节点,并把其中一个节点同时作为主节点。和声搜索算法参数设置如表 1 所示。表中 HMS 为和声记忆库的大小, HMCr 为和声记忆库思考概率,最小音调调节概率为 PAR_{min} ,最大调节概率为 PAR_{max} , FW_{min} 、 FW_{max} 分别为最小频宽和最大频宽。

表 1 HS 算法参数设置

参数名称	HMS	HMCr	PAR_{min}	PAR_{max}	FW_{min}	FW_{max}
参数值	5	0.90	0.01	0.09	$(X_{max} - X_{min})/100$	$(X_{max} - X_{min})$

在云计算环境下引入模拟退火算法 (cloud simulated annealing, CSA) 进行比较。表 2 给出了 CSA、CHS 和 ICHS 在基于上文所述 QoS 评价指标上的实验结果对比。表中任务数为 T , 知识服务数为 S , MI 为迭代次数, Map 数目为 10。实验中分别选择 5 组不同的任务数和服务数对算法的性能进行测试,这 5 组 (任务数, 服务数) 对分别为 (6, 50) (12, 100) (18, 180) (24, 280) (36, 400), 其中, 每条服务有其相应的 QoS 属性值, 包括知识服务的响应时间、服务可用性以及吞吐率等。对每组数据进行 20 次实验, 并取其结果的平均值作为实验结果。Timeaver 为每组数据各进行 20 次实验所用的平均时间, 单位为 s。 $F(KS)_{best}$ 表示实验中得到最优效用函数值, Solution 表示每组实验所获得的可用解的数量。

表 2 CSA、CHS 和 ICHS 结果比较

T	S	MI	Algorithm	Time _{aver}	$F(KS)_{best}$	Solution
6	50	2000	CSA	3.9584E+01	8.6859E+02	20
			CHS	3.8285E+01	8.6859E+02	20
			ICHS	4.1674E+01	8.6859E+02	20
12	100	5000	CSA	7.2216E+01	1.6730E+03	20
			CHS	7.2586E+01	1.6730E+03	20
			ICHS	7.2194E+01	1.6730E+03	20
18	180	8000	CSA	1.1705E+02	3.0206E+03	20
			CHS	1.1625E+02	3.0291E+03	20
			ICHS	1.0243E+02	3.0156E+03	20
24	280	15000	CSA	1.8600E+02	4.2070E+03	20
			CHS	1.8627E+02	4.2321E+03	20
			ICHS	1.6296E+02	4.1459E+03	20
36	400	20000	CSA	2.5078E+02	5.4895E+03	20
			CHS	2.5846E+02	5.5438E+03	20
			ICHS	2.4228E+02	5.4021E+03	20

由表 2 可以看出, CSA、CHS 与 ICHS 在求解知识服务组合优化问题上都具有较高的成功率, 每组实验均得到了最优解。随着任务数与服务数的增加, ICHS 表现出了较 CHS 以及 CSA 较优的性能。在任务数为 6、知识服务数为 50、迭代 2 000 次的情况下, CHS、CSA 与 ICHS 取得了相同的最优效用函数值, 只是 CHS 的运行时间较少。但是在其他数据组的实验中, ICHS 的效率均优于 CHS 以及 CSA, 且在一些数据组中, ICHS 取得了较优的效用函数值。例如, 在任务数为 18、知识服务数为 180 的数据组中, ICHS 的平均运行时间为 10243E+02, 最优效用函数值为 3.0156E+03, 优于 CHS 和 CSA。实验结果表明, 本文改进云和声搜索算法 ICHS 在算法的性能和求解效率上均优于基本云和声搜索算法 CHS 以及云计算环境下的模拟退

火算法 CSA。

图 4、5 表明了寻找最优知识服务组合过程中, ICHS 的收敛性优于 CHS 和 CSA。由图 4 可以看出, 在任务数为 12、服务数为 100、迭代 5 000 次的过程中, CHS、CSA 与 ICHS 得到了相同的效用函数值 1.6730E+03, 但是 ICHS 较先达到收敛状态。图 5 中, 在算法寻优过程中, ICHS 的寻优结果始终优于 CHS 和 CSA, 并取得了更好的最优效用函数值 4.1459E+03。

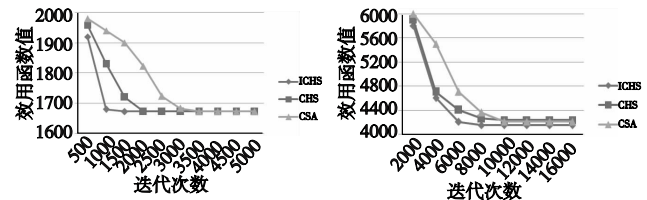


图 4 收敛过程比较(12,100,5000) 图 5 收敛过程比较(24,200,15000)

在云计算平台 Hadoop 下, 实验的时间开销除任务求解所必需的时间外, 还与所选 Map 的数目有关。另外, Hadoop 框架本身还需消耗一定的时间, 如任务部署、中间结果处理、网络传输等。在实验中, 选择任务数为 18、服务数为 180 的数据组, Map 数目分别取 10、20、30, 对算法运行时间与 Map 数目选取之间的关系进行分析。如表 3 所示, 其中 T_1 为任务部署时间; T_2 为 Map 阶段运行时间, 包括初始化和声记忆库、记忆思考、随机选择、音调调节等过程; T_3 表示 Map 到 Reduce 之间数据处理时间, 包括中间结果的复制、分组、排序等操作; T_4 是 Reduce 阶段的运行时间, 包括当前最优解的输出以及和声记忆库的更新; T_5 表示从 Reduce 阶段结束到任务完全结束的时间。由表 3 可以看出, T_1 、 T_3 、 T_4 以及 T_5 各阶段的运行时间基本不变, 且在总运行时间中所占比重较小, 主要的时间消耗在 T_2 Map 阶段。本文改进云和声搜索算法主要是对 Map 阶段进行优化, 减少其运行时间, 提高算法效率。

表 3 算法各阶段运行时间

Map	Algorithm	T_1	T_2	T_3	T_4	T_5
10	CSA	8.0298E+00	8.8017E+01	9.0547E+00	3.2748E+00	8.9975E+00
	CHS	8.3214E+00	8.6545E+01	9.5624E+00	3.5717E+00	9.8634E+00
	ICHS	8.2857E+00	7.1520E+01	9.2436E+00	3.7212E+00	9.5251E+00
20	CSA	8.9624E+00	6.8562E+01	9.2521E+00	4.7856E+00	9.0541E+00
	CHS	7.8962E+00	7.0500E+01	9.8577E+00	4.1002E+00	9.2164E+00
	ICHS	8.2900E+00	6.1925E+01	9.1578E+00	4.9982E+00	8.3547E+00
30	CSA	9.2541E+00	9.6502E+01	9.8546E+00	3.7187E+00	9.4256E+00
	CHS	9.6514E+00	9.2158E+01	1.0548E+01	4.2516E+00	9.0275E+00
	ICHS	8.5526E+00	8.9676E+01	9.5236E+00	3.9517E+00	8.6529E+00

5 结束语

本文在云计算平台 Hadoop 下实现了基于 QoS 的知识服务组合优化问题的求解。利用 HDFS 文件存储以及 MapReduce 编程模型实现和声搜索算法的并行化, 定义 Map、Reduce 过程, 提出云和声搜索算法。对基本云和声搜索算法进行改进, 利用 Skyline 方法对和声记忆库进行初始化, 提高算法效率; 采用理想点法寻找实际存在的最优知识服务组合, 确保了解的有效性。实验结果表明, 改进云和声搜索算法在知识服务组合优化问题上得到了成功应用。在本文基础上, 可进一步对和声搜索算法进行改进, 在大规模集群中优化算法性能; 同时可将云和声搜索算法应用在其他多目标优化问题的求解上, 扩展算法的实际应用价值。

(下转第 813 页)

Measure 值较高;支持向量机分类器的优势在链接特征中并不明显,使用 LPCCA 和 DCCA 方法后分类精度较高;朴素贝叶斯分类器在使用了典型相关分析方法之后,F1-Measure 值会大幅提高,其中 CCA 方法的分类效果最好。使用 KCCA 的决策树分类器也取得了最高的 F1-Measure 值。

这一部分的实验更加验证了把垃圾网页检测作为多视图分类问题的必要性,例如在内容特征中分类效果较好的支持向量机在链接特征中并没有很大优势,对链接特征使用典型相关分析方法之后支持向量机分类效果也有所提高,另外对于不同视图的特征所使用的典型相关分析方法也有所不同。由于内容特征的维数比链接特征多,在将两视图数据融合时受内容特征的影响较大,这也解释了支持向量机分类器在表 4 中 content + link 情况下 F1-Measure 值较高的原因。

4 结束语

将垃圾网页检测作为多视图问题来考虑,用典型相关分析及其推广方法对垃圾网页内容和链接特征进行特征提取,解决了如何处理垃圾网页特征的问题。对分类器选择的研究也为今后基于多视图的垃圾网页检测提供了依据。垃圾网页检测除了使用基于多视图的特征提取方法外,还可以从分类的角度运用多视图手段。考虑到现实中大多数的网页都是未被标记的,所以可以利用基于多视图的半监督协同训练来分类。

参考文献:

- [1] BECCHETTI L, CASTILLO C, DONATO D, *et al.* Web spam detection: link-based and content-based techniques [C]//Proc of European Final Workshop Integrated Project Dynamically Evolving, Large Scale Information Systems (DELIS). 2008;99-113.
- [2] CASTILLO C, DONATO D, GIONIS A, *et al.* Know your neighbors: Web spam detection using the Web topology [C]//Proc of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press, 2007;423-430.
- [3] BÍRÓ I, SIKLÓSI D, SZABÓ J, *et al.* Linked latent dirichlet allocation in Web spam filtering [C]//Proc of the 5th International Workshop on Adversarial Information Retrieval on the Web. New York: ACM Press, 2009;37-40.
- [4] LOURDES A, JUAN M R. Web spam detection: new classification features based on qualified link analysis and language models [J]. *IEEE Trans on Information Forensics and Security*, 2010, 5(3): 581-590.
- [5] ABERNETHY J, CHAPPELLE O, CASTILLO C. Graph regularization methods for Web spam detection [J]. *Machine Learning*, 2010, 81(2): 207-225.
- [6] HARDOON R D, SZEDMAK R S, SHAWE-TAYLOR R J. Canonical correlation analysis: an overview with application to learning method [J]. *Neural Computation*, 2004, 16(12): 2639-2664.
- [7] MELZER T, REITER M, BISCHOF H. Appearance models based on kernel canonical correlation analysis [J]. *Pattern Recognition*, 2003, 36(9): 1961-1971.
- [8] SUN Ting-kai, CHEN Song-can. Locality preserving CCA with applications to data visualization and pose estimation [J]. *Image and Vision Computing*, 2007, 25(5): 531-543.
- [9] SUN Ting-kai, CHEN Song-can. A novel method of combined feature extraction for recognition [C]//Proc of the 8th IEEE International Conference on Data Mining. Washington DC: IEEE Computer Society, 2008;1043-1048.
- [10] 孙廷凯. 增强型典型相关分析研究与应用 [D]. 南京: 南京航空航天大学, 2006.
- [11] WANG Wei, ZENG Guo-sun, TANG Dai-zhong. Using evidence based content trust model for spam detection [J]. *Expert Systems with Applications*, 2010, 37(8): 5599-5606.
- [1] BECCHETTI L, CASTILLO C, DONATO D, *et al.* A break in the clouds: towards a cloud definition [J]. *ACM SIGCOMM Computer Communication Review*, 2009, 39(1): 50-55.
- [2] SUBASHINI S, KAVITHA V. A survey on security issues in service delivery models of cloud computing [J]. *Journal of Network and Computer Applications*, 2011, 34(1): 1-11.
- [3] 刘鹏. 云计算 [M]. 北京: 电子工业出版社, 2010;10-34.
- [4] GEEM Z W. Effects of initial memory and identical harmony in global optimization using harmony search algorithm [J]. *Applied Mathematics and Computation*, 2012, 218(22): 11337-11343.
- [5] 韩红燕, 任立群. 求解批量流水线调度问题的和声算法 [J]. *计算机应用研究*, 2011, 28(1): 87-90.
- [6] PAN Quan-ke, SUGANTHAN P N, TASGETIRENC M F, *et al.* A self-adaptive global best harmony search algorithm for continuous optimization problems [J]. *Applied Mathematics and Computation*, 2010, 216(3): 830-848.
- [7] XU Shou-huai, ZHANG Wei-ning. Knowledge as a service and knowledge breaching [C]//Proc of IEEE International Conference on Services Computing. Washington DC: IEEE Computer Society, 2005: 87-94.
- [8] 王尚广, 孙其博, 杨放春. 基于全局 QoS 约束分解的 Web 服务动态选择 [J]. *软件学报*, 2011, 22(7): 1426-1439.
- [9] MAURER M, BRANDIC I, EMEAKAROKA V C, *et al.* Towards knowledge management in self-adaptable clouds [C]//Proc of the 4th IEEE International Workshop of Software Engineering for Adaptive Service-Oriented Systems. 2010.
- [10] VASSILIKI D, CHRISTOS M, YANNIS P, *et al.* Techniques to support Web service selection and consumption with QoS characteristics [J]. *Journal of Network and Computer Applications*, 2008, 31(2): 108-130.
- [11] SHVACHKO K, HUANG Hai-rong, RADIA S, *et al.* The Hadoop distributed file system [C]//Proc of the 26th IEEE Symposium on Massive Storage Systems and Technologies. Washington DC: IEEE Computer Society, 2010;1-10.
- [12] DEAN J, GHEMAWAT S. MapReduce: simplified data processing on large clusters [C]//Proc of the 6th Conference on Operating Systems Design & Implementation. New York: ACM Press, 2004;137-150.
- [13] 谢海军, 齐连永, 窦万春. 基于 Skyline 和局部选择的启发式服务组合方法 [J]. *东南大学学报: 自然科学版*, 2011, 41(3): 449-452.
- [14] 倪志伟, 尹道明, 王力, 等. 基于和声搜索算法的知识服务动态组合优化 [J]. *计算机工程与应用*, 2012, 48(32): 216-221, 247.

(上接第 809 页)

参考文献: