MapReduce框架下的粒概念认知学习系统研究

米允龙[1,2],李金海[1],刘文奇[1],林 晶[2,3]
(1.昆明理工大学理学院,云南昆明 650500;2.怀化学院计算机科学与工程学院,湖南怀化 418000;
3.武陵山片区生态农业智能控制技术湖南省重点实验室,湖南怀化 418000)

摘 要: 针对经典的概念学习算法难以处理大规模数据集的问题,本文提出一种基于MapReduce框架的粒概念认知学习并行算法.该算法借鉴认知心理学的知觉和注意认知思想,并融合粒计算的粒转移原理.首先构建适应大数据环境的粒概念并行求解算法,并与经典粒概念构造算法做了对比,在此基础上分别从外延和内涵角度建立了粒概念认知计算系统,然后对给定对象集或属性集进行认知概念学习.实验结果表明,该并行算法是有效的,适合海量数据的粒概念认知学习.

关键词: 概念格;概念学习;认知计算;粒计算;MapReduce
中图分类号: TP18 **文献标识码:** A **文章编号:** 0372-2112 (2018)02-0289-09
电子学报URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2018.02.005

Research on Granular Concept Cognitive Learning System Under MapReduce Framework

MI Yun-long[1,2], LI Jin-hai[1], LIU Wen-qi[1], LIN Jing[2,3]
(1. *Faculty of Science, Kunming University of Science and Technology, Kunming, Yunnan* 650500, *China*;
2. *College of Computer Science and Engineering, Huaihua University, Huaihua, Hunan* 418000, *China*;
3. *Key Laboratory of Intelligent Control Technology for Wuling-Mountain Ecological Agriculture in Hunan Province, Huaihua, Hunan* 418000, *China*)

Abstract: Considering that the classical concept learning algorithms are difficult to deal with the massive data set, a MapReduce-based parallel algorithm for granular concept cognitive learning is proposed. The parallel algorithm is based on the cognitive thoughts of perception and attention in cognitive psychology, and it is combined with the granule transformation principle of granular computing. Specifically, a parallel algorithm is developed to compute granular concepts in big data environment, and a comparative analysis of the parallel algorithm and the classical granular concept construction algorithm is made. Granular concept cognitive computing systems are also constructed from the perspectives of extension and intension. Then, cognitive concept learning is performed by a given object set or attribute set. Experimental results show that the proposed parallel algorithm is effective and can be suitable for granular concept cognitive learning of massive data.

Key words: concept lattice; concept learning; cognitive computing; granular computing; MapReduce

1 引言

随着信息技术的飞速发展,数据已呈现爆炸式增长.面对海量数据处理中的众多问题,如何借鉴人脑的思维模式来分析大数据,从而获得有价值的信息,已成为当前研究的热点[1~3].众所周知,概念是通过识别其外延与内涵来完成的[4~6],即确定对象与属性之间的具体关系.近年来,为了适应各种数据分析,学者们已提出不同类型的概念[7,8].概念学习是从已知信息中运用特定学习方法获取未知概念.例如,通过问询方式[9],云模型[10,11],认知系统[12],近似逼近[13],概念迭代[6]等都是概念学习的具体表现.

在概念学习的基础上,通过模拟人脑的认知过程(包括知觉、注意等),将概念形成原理反映到人类的认

收稿日期:2016-09-26;修回日期:2016-11-17;责任编辑:马兰英
基金项目:国家自然科学基金(No.61305057, No.61562050, No.61573173)

知过程中,则称为认知概念学习(或概念认知学习).其主要思想由文献[14]提出,后经姚一豫[15]、李金海等[13,16,17]进一步完善后,现已形成认知概念学习的理论雏形.目前,认知概念学习是概念学习领域中的一个热点[18,19],且被认为是一种有效的概念学习方法.通常,认知概念学习由揭示概念的认知机理、建立认知计算系统和模拟认知学习过程三部分组成.

需要指出的是,面对海量数据集时,经典的认知概念学习算法的时间、空间代价均非常大.实际上,为了提高学习效率,可将复杂问题按特定规律进行粒化分解.粒计算[20]常用于降低问题求解的复杂性[21,22],已在概念学习领域取得较好的效果[6,12~17,19].原因是认知概念的存储完全可以通过基本信息粒完成[13,16,17].因此,从粒计算角度来提高认知概念学习的效率是可行的.

目前,已有学者从粒计算角度研究认知概念学习,并取得一定的成效[6,12~17,19].但是,主要是针对小规模的数据集进行测试或只给出应对海量数据的一些设想[16].针对数量巨大、结构复杂且形式多样的海量数据[3],本文给出一种高效的粒概念认知学习方法,它基于 MapReduce[23]并行框架,且融合粒转移原理.实验结果表明,该并行算法能够完成海量数据的粒概念认知学习任务.

2 预备知识

2.1 概念的认知机理

对任意对象集 U 和属性集 A,其幂集分别记为 2^U 和 2^A.设 $L:2^U\to 2^A$ 和 $H:2^A\to 2^U$ 是 U 和 A 之间的集值映射,并在文中简记为 L 和 H.

定义 1[13] 设 L 和 H 为集值映射.若任意 $X_1,X_2\subseteq U,B\subseteq A$ 满足:

$$X_1\subseteq X_2\Rightarrow L(X_2)\subseteq L(X_1),$$
$$L(X_1\cup X_2)\supseteq L(X_1)\cap L(X_2),$$
$$H(B)=\{x\in U\mid B\subseteq L(\{x\})\},$$

则称 L 和 H 为认知算子.

在无歧义下,文中均将 $L(\{x\})$ 记为 $L(x)$.

定义 2[13] 设 L 和 H 为认知算子.若 $X\subseteq U,B\subseteq A$,$L(X)=B$ 且 $H(B)=X$,则称序对 (X,B) 为认知概念,X 和 B 分别为 (X,B) 的外延与内涵.

定义 3[13] 设 L 和 H 为认知算子,$x\in U$ 且 $a\in A$,则称 $(HL(x),L(x))$ 和 $(H(a),LH(a))$ 为认知算子 L 和 H 的粒概念.

性质 1[13] 设 L 和 H 为认知算子,则任意认知概念 (X,B) 均可由粒概念合成:

$$(X,B)=\bigvee_{x\in X}(HL(x),L(x))=\bigwedge_{a\in B}(H(a),LH(a)),$$

其中,

$$\bigvee_{x\in X}(HL(x),L(x))=(HL(\bigcup_{x\in X}HL(x)),\bigcap_{x\in X}L(x)),$$
$$\bigwedge_{a\in B}(H(a),LH(a))=(\bigcap_{a\in B}H(a),LH(\bigcup_{a\in B}LH(a))).$$

为了方便,将认知算子 L 和 H 的所有粒概念记为:

$$G_{LH}=\{(HL(x),L(x))\mid x\in U\}\cup\{(H(a),LH(a))\mid a\in A\}.$$

2.2 认知计算系统

称满足 $U_1\subseteq U_2\subseteq\cdots\subseteq U_n$ 的 n 个对象集为非降对象集序列,记为 $\{U_t\}^{\uparrow}$;类似地,称满足 $A_1\subseteq A_2\subseteq\cdots\subseteq A_n$ 的 n 个属性集为非降属性集序列,记为 $\{A_t\}^{\uparrow}$.

定义 4[13] 设 U_{i-1},U_i 为 $\{U_t\}^{\uparrow}$ 的两个对象集,A_{i-1},A_i 为 $\{A_t\}^{\uparrow}$ 的两个属性集.记 $\Delta U_{i-1}=U_i-U_{i-1}$ 且 $\Delta A_{i-1}=A_i-A_{i-1}$.令

(1) $L_{i-1}:2^{U_{i-1}}\to 2^{A_{i-1}}$ $\quad H_{i-1}:2^{A_{i-1}}\to 2^{U_{i-1}}$,

(2) $L_{\Delta U_{i-1}}:2^{\Delta U_{i-1}}\to 2^{A_{i-1}}$ $\quad H_{\Delta U_{i-1}}:2^{A_{i-1}}\to 2^{\Delta U_{i-1}}$,

(3) $L_{\Delta A_{i-1}}:2^{U_i}\to 2^{\Delta A_{i-1}}$ $\quad H_{\Delta A_{i-1}}:2^{\Delta A_{i-1}}\to 2^{U_i}$,

(4) $L_i:2^{U_i}\to 2^{A_i}$ $\quad H_i:2^{A_i}\to 2^{U_i}$

为四对认知算子.若它们满足:

$$L_i(x)=\begin{cases}L_{i-1}(x)\cup L_{\Delta A_{i-1}}(x),\text{如果 } x\in U_{i-1},\\ L_{\Delta U_{i-1}}(x)\cup L_{\Delta A_{i-1}}(x),\text{其它},\end{cases}$$

$$H_i(a)=\begin{cases}H_{i-1}(a)\cup H_{\Delta U_{i-1}}(a),\text{如果 } a\in A_{i-1},\\ H_{\Delta A_{i-1}}(a),\text{其它},\end{cases}$$

并约定:当 $\Delta U_{i-1}=\varnothing$ 时,$L_{\Delta U_{i-1}}(x)$ 和 $H_{\Delta U_{i-1}}(a)$ 置为空;当 $\Delta A_{i-1}=\varnothing$ 时,$L_{\Delta A_{i-1}}(x)$ 和 $H_{\Delta A_{i-1}}(a)$ 置为空,则称 L_i,H_i 为 L_{i-1},H_{i-1} 通过更新对象和属性信息后得到的扩展认知算子.

定义 5[13] 设 L_i,H_i 为 L_{i-1},H_{i-1} 通过更新对象和属性信息后得到的扩展认知算子,则称 $S_{L_iH_i}=(G_{L_{i-1}H_{i-1}},L_{\Delta U_{i-1}},H_{\Delta U_{i-1}},L_{\Delta A_{i-1}},H_{\Delta A_{i-1}})$ 为一个认知计算状态.其中,$G_{L_{i-1}H_{i-1}}$ 表示认知算子 L_{i-1},H_{i-1} 的所有粒概念.进一步地,称若干认知计算状态组成的集合为认知计算系统.

注意,认知计算系统最终的粒概念 $G_{L_nH_n}$,可以通过一系列新输入的对象与属性信息对初始粒概念 $G_{L_1H_1}$ 反复迭代得到.

2.3 认知学习过程

定义 6[13] 设 X_0 为给定对象集,称 $(\underline{Apr}(X_0),L_n(\underline{Apr}(X_0)))$ 和 $(\overline{Apr}(X_0),L_n(\overline{Apr}(X_0)))$ 为学习 X_0 后得到的认知概念.其中,

$$\underline{Apr}(X_0)=H_nL_n(\bigcup_{(X,B)\in G^*_{L_nH_n},X\subseteq X_0}X),$$

$$\overline{Apr}(X_0)=\bigcap_{(X,B)\in G^*_{L_nH_n},X_0\subseteq X}X,$$

$$G^*_{L_nH_n}=\begin{cases}G_{L_nH_n}\cup\{(U_n,\varnothing)\},\text{如果}(U_n,\varnothing)\text{是概念},\\ G_{L_nH_n},\text{其它},\end{cases}$$

$$G^{\#}_{L_nH_n}=\begin{cases}G_{L_nH_n}\cup\{(\varnothing,A_n)\},\text{如果}(\varnothing,A_n)\text{是概念},\\ G_{L_nH_n},\text{其它}.\end{cases}$$

定义 7[13] 设 B_0 为给定属性集,称 $(H_n(\underline{Apr}(B_0)),\underline{Apr}(B_0))$ 和 $(H_n(\overline{Apr}(B_0)),\overline{Apr}(B_0))$ 为学习 B_0

后得到的认知概念. 其中,

$$\underline{Apr}(B_0) = \bigcap_{(X,B)\in G^*_{L_0L_0},B_0\subseteq B} B,$$

$$\overline{Apr}(B_0) = L_nH_n\Big(\bigcup_{(X,B)\in G^*_{L_0L_0},B\subseteq B_0} B\Big).$$

这里的认知学习过程的核心思想是通过上、下近似逼近认知概念.

3　认知概念学习的并行算法

本节将从并行模式角度阐述认知概念学习,即开发构建概念的认知机理、认知计算系统及认知学习过程的并行算法.

3.1　构建粒概念的并行算法

定义 8　设 L 和 H 为认知算子. 若 $x\in U, a\in A$ 满足 $HL(x)=\{x\}$ 且 $LH(a)=\{a\}$,则称 $(HL(x),L(x))$ 和 $(H(a),LH(a))$ 为原子粒概念.

定义 9　设 L 和 H 为认知算子. 若 $x\in U, a\in A$ 满足 $HL(x)\supset\{x\}$ 且 $LH(a)\supset\{a\}$,则称 $(HL(x),L(x))$ 和 $(H(a),LH(a))$ 为合成粒概念.

原子粒概念表示被选择的信息能唯一被属性集 $L(x)$ 分辨出来,而合成粒概念则表示优于属性集 $L(x)$ 的信息被选择.

性质 2　设 $x,y\in U$ 且 $a,b\in A$. 若 $(HL(x),L(x))$ 和 $(H(a),LH(a))$ 均为原子粒概念,则 $L(x)\subset L(y)$ 与 $H(a)\subset H(b)$ 都不成立.

证明　由定义 1 和定义 8 易证.

定理 1　(1)对任意 $x\in U$,若存在 $y_1,y_2,\cdots,y_n\in U$ 使得 $L(x)\subseteq L(y_i)$ $(i=1,2,\cdots,n)$,则 $HL(x)=\{x,y_1,y_2,\cdots,y_n\}$.

(2)对任意 $a\in A$,若存在 $b_1,b_2,\cdots,b_n\in A$ 使得 $H(a)\subseteq H(b_j)$ $(j=1,2,\cdots,m)$,则 $LH(a)=\{a,b_1,b_2,\cdots,b_m\}$.

证明　由定义 1,定义 9,性质 2 易证.

性质 2 与定理 1 表明,计算认知算子 L 和 H 的粒概念可通过一次扫描整个对象集(或属性集)完成. 以下算法 1 和算法 2 分别从对象和属性两方面计算粒概念.

算法 1　对象粒概念并行算法($H_0L_0(x)$,$L_0(x)$)

输入:(key:为首字母的首地址的偏移量;value:为原数据集 DS_0 中一条记录).

输出:(outKey:为对象集 $H_0L_0(x)$;outValue:为相对应属性集 $L_0(x)$).

步骤 1:对于任意对象,以值为"1"的属性来构造属性集,并将具有相同属性集的对象进行归并. 以属性集 attributeSet 为键,对象集 objectSet 为值进行输出.

步骤 2:以步骤 1 的输出作其输入,根据定理 1,对每一行的属性与数据集的属性进行比较,得到包含该属性集的对象集(即构造相同属性的对象集). 并以对象集 $H_0L_0(x)$ 为键,属性集 $L_0(x)$ 进行输出.

算法 2　属性粒概念并行算法($H_0(a)$,$L_0H_0(a)$)

输入:(key:为首字母的首地址的偏移量;value:为原数据集 DS_0 中一条记录).

输出:(outKey:为对象集 $H_0(a)$;outValue:为相对应属性集 $L_0H_0(a)$).

步骤 1:对于任意对象,先将属性值为"1"的属性与对应的对象输出,再以相同属性来构造对象集. 并以对象集 objectSet 为键,属性集 attributeSet 为值进行输出.

步骤 2:以步骤 1 的输出作其输入,把相同对象集的属性进行收集,进而构造属性集. 并以对象集 objectSet 为键,属性集 attributeSet 为值进行输出.

步骤 3:根据定理 1 将原数据集任意对象与步骤 2 输出的数据集中对象集进行比较,得到包含了该对象集的属性集(即根据相同对象集来构造属性集). 并以对象集 $H_0(a)$ 为键,属性集 $L_0H_0(a)$ 进行输出.

与算法 1 相比,算法 2 不仅多执行一个步骤,而且还是对象与对象之间的比较. 因此,当数据量一定时,属性多的数据集执行时间更大. 此处,可以认为数据集的属性敏感度比对象的敏感度要大.

3.2　认知计算系统的并行算法

图 1 至图 4 给出了认知计算系统的并行算法的基本框架,其中图 1 与图 2 分别从外延与内涵角度描述因对象增加而引起的粒概念变化,图 3 和图 4 描述的是因属性增加而引起的粒概念的外延与内涵的变化.

图 1 描述了面向对象的认知计算状态(从外延角度分析). 该算法由 5 个 MapReduce Jobs(图中以 MR 标记)组成. 其中,MR1 和 MR2 是从对象角度描述了因对象的增加而如何产生新的粒概念;MR3 和 MR4 是从对象角度根据定理 1 更新粒概念(其中,MR3 是指当对象 x 属于原对象集时,进行粒概念更新,MR4 指当对象 x 属于新增对象集时,更新粒概念);MR5 将 MR3 与 MR4 得到的粒概念按相同对象进行归并获得整个粒概念集合.

图 2 描述的是面向对象的认知计算状态(从内涵角度分析). 该算法由 4 个 MapReduce Jobs(图中以 MR 标记)组成. MR1 和 MR2 是从属性角度描述了因对象增加而如何产生新的粒概念;MR3 和 MR4 是从属性角度根据定理 1 进行粒概念的更新,从而得到整个粒概念集合.

图 3 描述了面向属性的认知计算状态(从外延角度分析). 该算法由 4 个 MapReduce Jobs(图中以 MR 标记)组成. MR1 和 MR2 是从对象角度描述因属性增加而如何产生新的粒概念;MR3 和 MR4 是从对象角度根据定理 1 进行粒概念更新,从而得到整个粒概念集合.

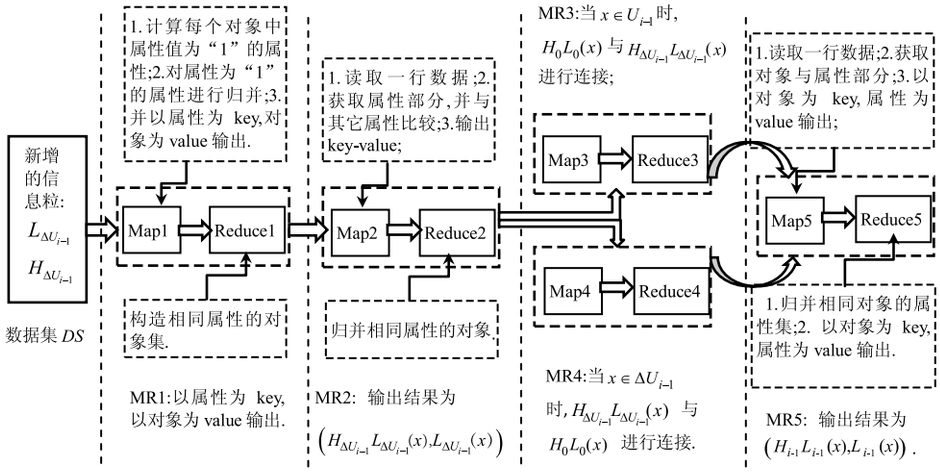

图1 面向对象的认知计算状态(外延角度分析)

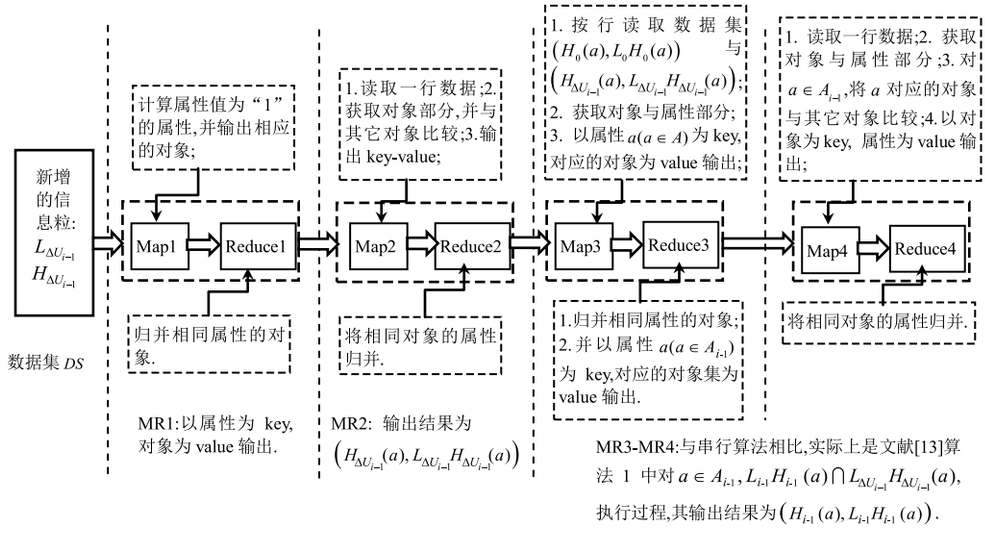

图2 面向对象的认知计算状态(内涵角度分析)

图4描述的是面向属性的认知计算状态(从内涵角度分析). 该算法由5个MapReduce Jobs(图中以MR标记)组成. MR1和MR2是从属性角度描述因属性增加而如何产生新的粒概念;MR3和MR4是从属性角度根据定理1进行粒概念更新(其中,MR3是指当属性a属于原属性集时,进行粒概念更新,MR4指当属性a属于新增属性集时,更新粒概念);MR5将MR3与MR4得到的粒概念按相同对象进行归并获得整个粒概念集合.

3.3 认知学习过程的并行算法

在认知计算系统的基础上,本小节从线索为对象集和属性集两个方面来模拟认知学习过程.

依据定义6,图5至图7分别展示了利用认知概念$(H_iL_i(x),L_i(x))$和$(H_i(a),L_iH_i(a))$获取线索为给定对象集X_0的上、下近似. 图5描述的是通过$(H_iL_i(x),L_i(x))$与X_0学习上、下近似(分别记为Ω_x与$\prod_x$). 图6描述的是通过$(H_i(a),L_iH_i(a))$与X_0学习上、下近似(分别记为Ω_a与$\prod_a$). 图7描述了合并从$(H_iL_i(x),L_i(x))$与$(H_i(a),L_iH_i(a))$学习所得的上、下近似Ω_x,$\prod_x$,Ω_a,$\prod_a$(分别记为$\overline{Apr}(X_0)$与$\underline{Apr}(X_0)$).

依据定义7,图8至图10分别展示了利用认知概念$(H_iL_i(x),L_i(x))$和$(H_i(a),L_iH_i(a))$获取线索为给定属性集B_0的上、下近似. 图8描述的是通过$(H_iL_i(x),L_i(x))$与B_0学习上、下近似(分别记为Ω_x与$\prod_x$). 图9描述的是通过$(H_i(a),L_iH_i(a))$与B_0学习上、下近似(分别记为Ω_a与$\prod_a$). 图10描述了合并从$(H_iL_i(x),L_i(x))$与$(H_i(a),L_iH_i(a))$学习所得的上、下近似Ω_x,$\prod_x$,Ω_a,$\prod_a$(记为$\overline{Apr}(B_0)$与$\underline{Apr}(B_0)$).

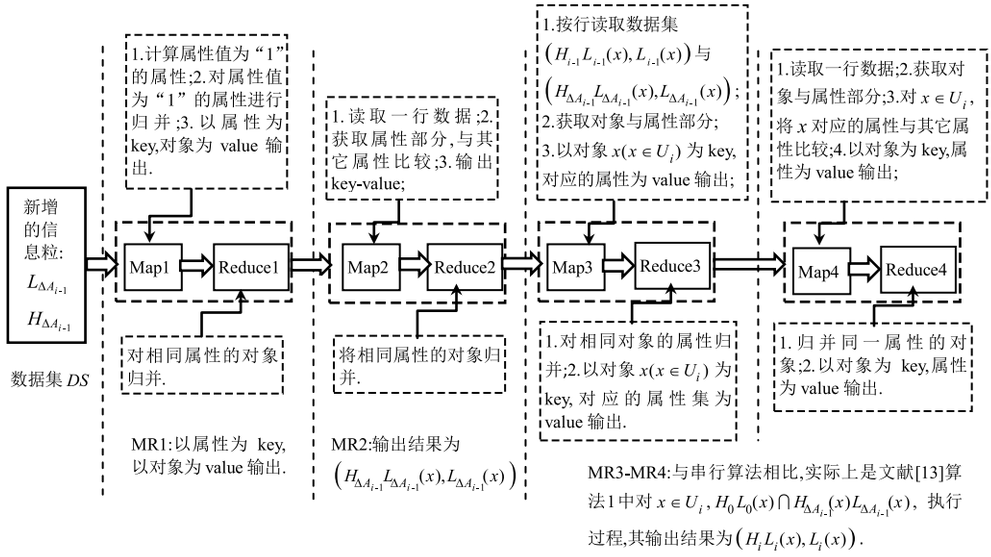

图3　面向属性的认知计算状态(外延角度分析)

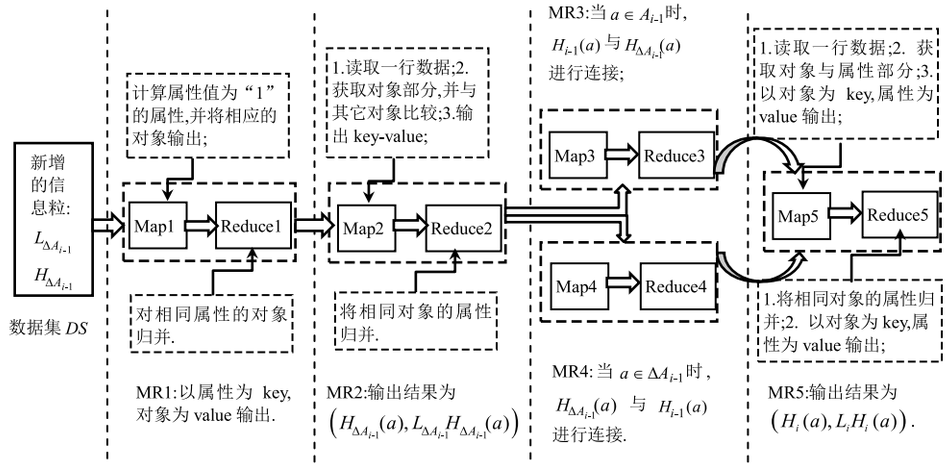

图4　面向属性的认知计算状态(内涵角度分析)

4　实验分析

4.1　实验环境

实验中采用的 MapReduce 集群由 1 个主节点和 8 个从节点构成. 其中,每个节点配置如下:CPU 为 Intel Core i3-2120 3.30 GHz,4.00GB 内存,500GB 硬盘;操作系统为 32 位的 Linux CentOS 6.6,JDK 为 Java 1.7.0_17,Eclipse 使用 32 位 Linux 版本的 eclipse-4.2,Hadoop 采用 hadoop-1.2.1. 其他参数为系统默认配置. 数据集均来源于机器学习数据库中的真实数据①,具体如表 1 所示.

4.2　实验结果

4.2.1　粒概念求解算法对比

将表 1 中的原始数据集通过标尺变换(Scaling)转化为标准数据集(形式背景),并分别记为 Data Sets 1,2,3,4,5 与 6. 表 2 给出了粒概念串行求解算法与并行求解算法的耗时对比. 其中,对象与属性采取 $\left(\frac{1}{2}U+\frac{1}{2}A\right)$的更新模式,串行算法运行在 1 台机器上,而并行算法运行在相同配置的 8 台机器上.

从表 2 可以看出,随着数据集的不断增大,本文并行算法与现有算法相比,有较大的优势,特别是数据集很大时,计算效率提高明显. 同时,可以看到,当数据集较小时,计算粒概念在很大程度上受属性规模影响(即 3.1 小节指出的数据属性敏感度要大于对象敏感度). 例如,虽然 Data set 2 比 Data set 1 规模大,但由于属性复杂性不同,导致本文并行算法的运行时间大小顺序倒置.

① http://archive.ics.uci.edu/ml/

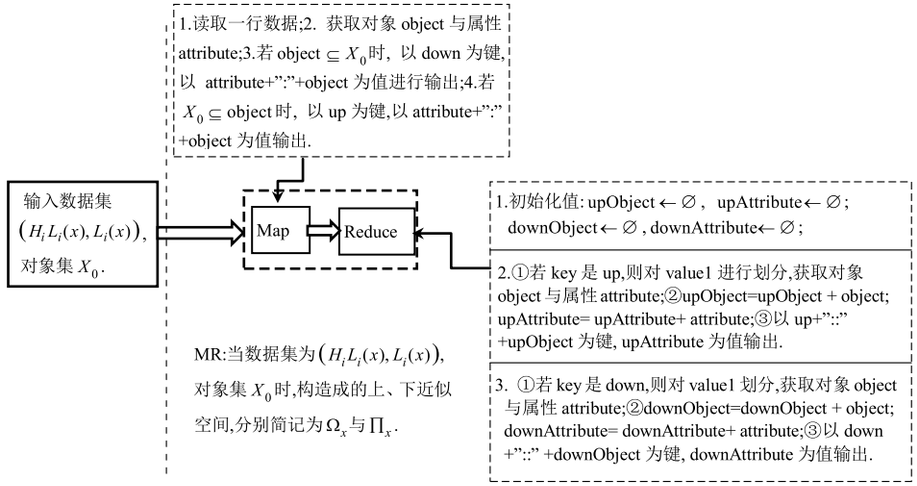

图5 从线索X_0学习上、下近似(认知概念(H_i, $L_i(x)$, $L_i(x)$)角度)

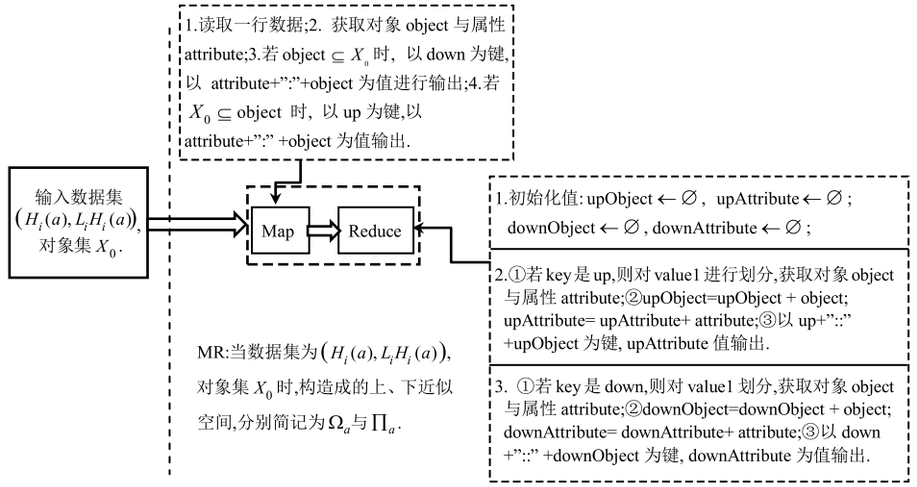

图6 从线索X_0学习上、下近似(认知概念($H_i(a)$, $L_iH_i(a)$)角度)

表 1 数据集的相关参数

数据集	对象数	属性类型(数量)	大小(单位:M)
Bank Marketing Data Set	45,211	离散(17)	4.83
kddcup. data_10_percent	494,021	离散(10),连续(32)	71.40
Buzz Prediction on Twitter	583,250	离散(77)	270.00
KDD Cup 1999 Data	4,898,431	离散(10),连续(32)	708.00
SUSY	5,000,000	离散(18)	2273.28
HIGGS	11,000,000	离散(28)	7659.52

表 2 粒概念串、并行求解算法耗时对比(单位:s)

数据集	对象数	属性数	文献[13]的串行算法	文献[16]的并行算法	本文并行算法
Data set 1	45,211	74	14,902.00	5285.59	733.11
Data set 2	494,021	163	8,282,570.82	2,964,126.00	528.70
Data set 3	583,250	142	10,720,690.47	8,165,500.12	6,476.00
Data set 4	4,898,431	163	very long	very long	39,536.76
Data set 5	5,000,000	38	very long	very long	64,932.12
Data set 6	11,000,000	58	very long	very long	325,017.18

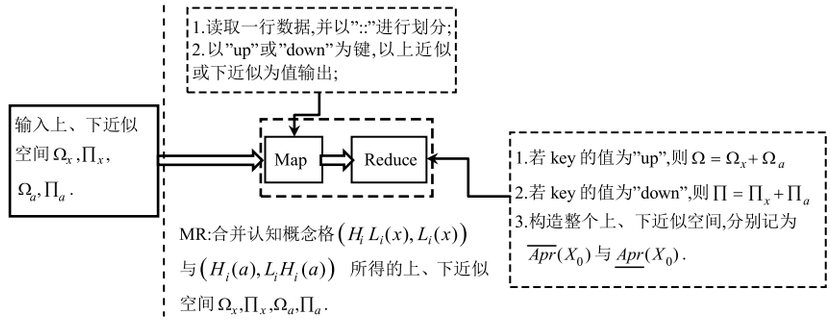

图7　从线索X_0学习上、下近似$\overline{Apr}(X_0)$与$\underline{Apr}(X_0)$

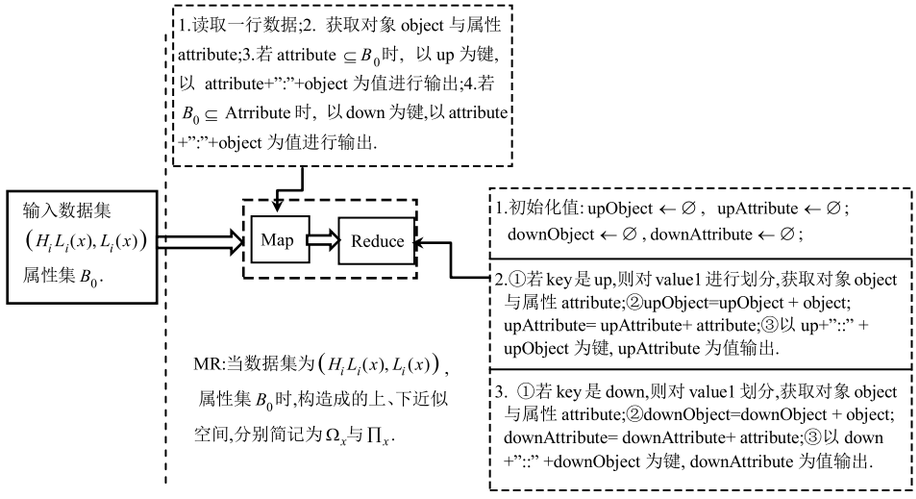

图8　从线索B_0学习上、下近似(认知概念$(H_i L_i(x), L_i(x))$角度)

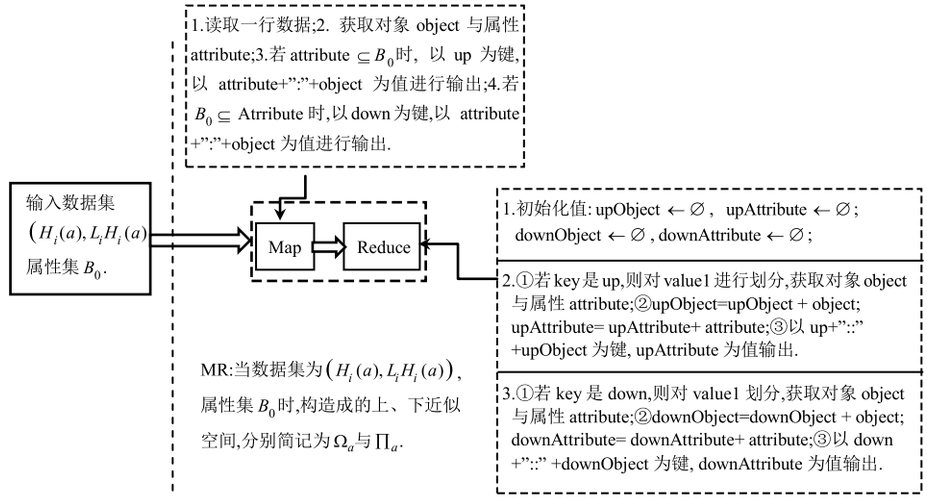

图9　从线索B_0学习上、下近似(认知概念$(H_i(a), L_i H_i(a))$角度)

4.2.2　数据集规模对并行算法耗时的影响

将粒概念并行求解算法、面向对象和面向属性的认知计算系统并行算法分别记为 BGCPA、CSPA1 与 CSPA2,而从给定对象集和属性集的认知学习过程并行算法分别记为 CLPA1 与 CLPA2.

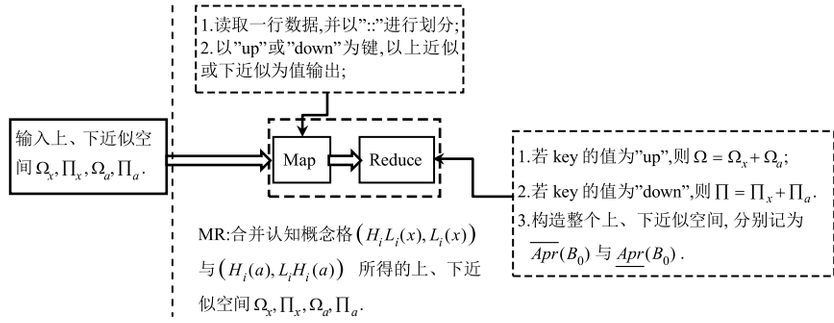

图10 从线索B_0学习上、下近似$\overline{Apr}(B_0)$与$\underline{Apr}(B_0)$

图 11 展示了粒概念并行算法与认知计算系统并行算法在不同数据集规模下的运行情况. 不难看出,计算粒概念的时间不仅随数据集规模增加而增加,而且与数据集本身的复杂性有一定的关系. 例如,Data set 4 的运行时间比 Data set 5 要高,原因是前者的属性多于后者.

图 12 描述了在认知计算系统基础上模拟认知学习过程的耗时情况. 其中,曲线 CLPA1 与 CLPA2 分别描述的是不同数据集上从给定对象集与属性集进行认知概念学习的并行算法的耗时. 由于这些数据集的粒概念的规模都不大,所以在此基础上进行认知概念学习的耗时均较小.

4.2.3 加速比

加速比描述的是同一数据集在不同节点上的运行时间,公式如下[24]:

$$\mathrm{Speedup}(m)=\frac{T_1}{T_m},$$

其中,T_1 是固定规模数据集在 1 个节点上的耗时,T_m 是在 m 个节点上的耗时. 通常,线性加速比是十分理想的,但是由于节点数增加,集群之间的通信时间也会不断增加. 因此,一般难以达到理想状态. 本文采用的数据集同样如此.

例如,图 13 展示了数据集 Data set 2 与 Data set 3 的加速比. 不难看出,数据集 Data set 3 的加速比较好,而数据集 Data set 2 的加速比较差. 原因可能是数据量较大时,集群各节点更能被充分运用.

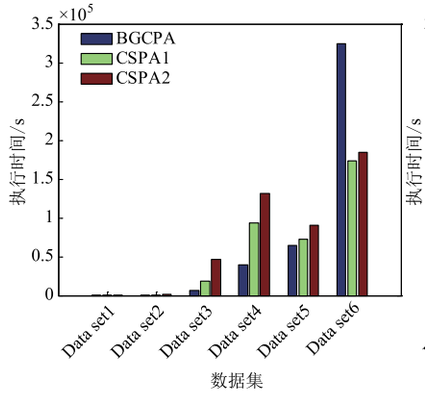

图11 计算粒概念与认知计算系统的并行算法耗时情况

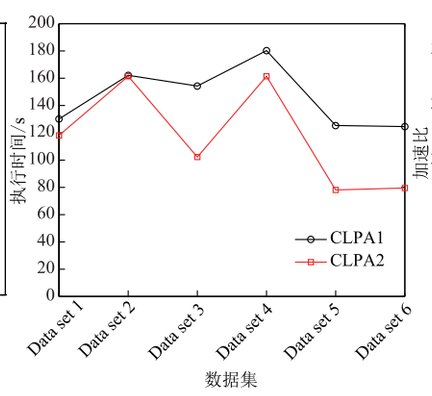

图12 认知学习过程的并行算法耗时情况

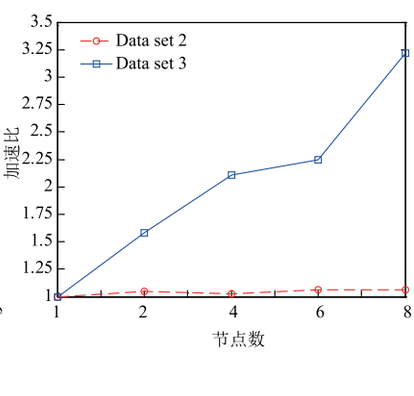

图13 加速比性能

5 结束语

本文基于粒计算理论和认知心理学基本原理,提出一种粒概念认知学习并行算法. 实验表明,本文提出的并行算法在处理海量数据时是有效的.

今后需要进一步开展的研究包括:如何充分运用集群的每一个节点,并考虑架 Spark 集群来处理循环部分,从而提高并行算法的执行效率;此外,考虑通过多次学习来降低算法耗时,特别是找到合理的学习次数及其次序.

参考文献

[1] Hurwitz J, Kaufman M, Bowles A. Cognitive Computing and Big Data Analytics[M]. Hoboken: John Wiley & Sons Inc, 2015.

[2] Wang G Y, Xu J. Granular computing with multiple granular layers for brain big data processing[J]. Brain Informatics, 2014, 1(1): 1 – 10.

[3] Zhong N, Yau S S, Ma J, et al. Brain informatics-based big data and the Wisdom web of things[J]. IEEE Intelligent Systems, 2015, 30(5): 2 – 7.

[4] 武秀波,苗霖,吴丽娟,张辉. 认知科学概论[M]. 北京:科学出版社,2007.
Wu X B, Miao L, Wu L J, Zhang H. Introduction to Cognitive Science[M]. Beijing: Science Press, 2007. (in Chinese)

[5] Wang Y. On cognitive computing[J]. International Journal of Software Science and Computational Intelligence, 2009, 1(3): 1 – 15.

[6] Qiu G F, Ma J M, Yang H Z, Zhang W X. A mathematical model for concept granular computing systems[J]. Science China: Information Sciences, 2010, 53(7): 1397 – 1408.

[7] Wang Y, Zadeh L A, Yao Y Y. On the system algebra foundations for granular computing[J]. International Journal of Software Science and Computational Intelligence, 2009, 1(1): 64 – 86.

[8] Ganter B, Wille R. Formal Concept Analysis: Mathematical Foundations[M]. New York: Springer, 1999.

[9] Angluin D. Queries and concept learning[J]. Machine Learning, 1988, 2(4): 319 – 342.

[10] Wang G Y, Xu C L, Li D Y. Generic normal cloud model[J]. Information Sciences, 2014, 280: 1 – 15.

[11] 李德毅,刘常昱,杜鹢,韩旭. 不确定性人工智能[J]. 软件学报,2004,15(11):1584 – 1594.
Li D Y, Liu C Y, Du Y, Han X. Artificial intelligence with uncertainty[J]. Journal of Software, 2004, 15(11): 1583 – 1594. (in Chinese)

[12] Xu W H, Pang J Z, Luo S Q. A novel cognitive system model and approach to transformation of information granules[J]. International Journal of Approximate Reasoning, 2014, 55(3): 853 – 866.

[13] Li J H, Mei C L, Xu W H, Qian Y H. Concept learning via granular computing: A cognitive viewpoint[J]. Information Sciences, 2015, 298: 447 – 467.

[14] 张文修,徐伟华. 基于粒计算的认知模型[J]. 工程数学学报,2007,24(6):957 – 971.
Zhang W X, Xu W H. Cognitive model based on granular computing[J]. Chinese Journal of Engineering Mathematics, 2007, 24(6): 957 – 971. (in Chinese)

[15] Yao Y Y. Interpreting concept learning in cognitive informatics and granular computing[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2009, 39(4): 855 – 866.

[16] Li J H, Huang C C, Xu W H, et al. Cognitive concept learning via granular computing for big data[A]. 2015 International Conference on Machine Learning and Cybernetics[C]. Washington: IEEE, 2015. 289 – 294.

[17] Li J H, Huang C C, Qi J J, et al. Three-way cognitive concept learning via multi-granularity[J]. Information Sciences, 2017, 378: 244 – 263.

[18] Kumar C A, Ishwarya M S, Loo C K. Formal concept analysis approach to cognitive functionalities of bidirectional associative memory[J]. Biologically Inspired Cognitive Architectures, 2015, 12: 20 – 33.

[19] Xu WH, Li WT. Granular computing approach to two-way learning based on formal concept analysis in fuzzy datasets[J]. IEEE Transactions on Cybernetics, 2016, 46(2): 366 – 379.

[20] Zadeh L A. Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic[J]. Fuzzy Sets and Systems, 1997, 90(2): 111 – 127.

[21] 梁吉业,等. 大数据挖掘的粒计算理论与方法[J]. 中国科学:信息科学,2015,45(11):1355 – 1369.
Liang J Y, et al. Theory and method of granular computing for big data mining[J]. Science China: Information Sciences, 2015, 45(11): 1355 – 1369. (in Chinese)

[22] 徐计,王国胤,于洪. 基于粒计算的大数据处理[J]. 计算机学报,2015,38(8):1497 – 1517.
Xu J, Wang G Y, Yu H. Review of big data processing based on granular computing[J]. Chinese Journal of Computers, 2015, 38(8): 1497 – 1517. (in Chinese)

[23] 李建江,崔健,王聃,等. MapReduce 并行编程模型研究综述[J]. 电子学报,2011,39(11):2635 – 2642.
Li J J, Cui J, Wang D, et al. Survey of MapReduce parallel programming model[J]. Acta Electronica Sinica, 2011, 39(11): 2635 – 2642. (in Chinese)

[24] Xu X W, Jager J, Kriegel H P. A fast parallel clustering algorithm for large spatial databases[J]. Data Mining and Knowledge Discovery, 1999, 3(3): 263 – 290.

作者简介

米允龙　男,1987 年生于湖南怀化. 硕士研究生,讲师,主要研究方向为数据挖掘与认知计算.
E-mail: yunlongmi@yeah.net

李金海(通信作者)　男,1984 年生于江西上饶. 博士,副教授,硕士生导师,主要研究方向为认知计算、粒计算与概念格.
E-mail: jhlixjtu@163.com