

# 基于多 Agent 的特征提取模型研究\*

车 骏, 顾绍元

(同济大学 计算机科学与技术系, 上海 200092)

摘 要: 讨论了基于多 Agent 的特征提取模型的工作流程, 主要研究了该多 Agent 模型的构建框架和模型中单个 Agent 的通用结构, 借助 JAFMAS 这一面向 MAS 的基于 Java 的 Agent 框架实现了系统的建立, 并最终给出了所举例子的实际运行结果。

关键词: Agent; 多 Agent 模型; JAFMAS; 特征提取

中图法分类号: TP18 文献标识码: A 文章编号: 1001-3695(2005)05-0040-02

## A Model of Feature Extraction Based on Multi-Agent

CHE Jun, GU Shao-yuan

(Dept. of Computer Science & Technology, Tongji University, Shanghai 200092, China)

**Abstract:** On the basis of utilizing the way of tag sort as well as extracting commonness to distinguish the differences between the normal cell and the cancer cell, a model of distributed Agent-based feature extraction has been advanced. In addition, a variety of Agents constructions and the ways to implementation are also discussed.

**Key words:** Agent; Multi-Agent Model; JAFMAS; Feature Extraction

随着人工智能技术的不断发展, 特征提取作为人工智能中的一个重要部分, 被应用于更加广泛的领域。但是, 在使用各种特征提取算法处理实际问题时, 常常面临海量数据的采集和分析等诸多的问题。特别是针对大量待处理数据的动态性、分布性的特点, 传统的客户/服务器方法遇到很大的困难, 常常导致网络的严重负载和工作效率低下。随着计算机与通信的技术发展, 应用多 Agent 技术建立特征提取模型为解决这一问题提供了一个新的思路; 通过跨平台的信息系统集成、资源共享、互相协调优化特征提取的分析工作, 加快工作速度, 降低网络负载。Agent 是一个封装了某些状态和知识, 并使用消息进行通信, 在一定的环境中能够独立自主地完成特定功能的智能主体<sup>[1]</sup>。多 Agent 系统(MAS)是指一些 Agent 通过协作完成某些任务或达到某些目标的计算系统<sup>[2]</sup>。通过系统中多 Agent 之间的协调、通信以及外环境的交互通信, 多 Agent 技术被越来越广泛地应用于电子商务、网络管理等网络环境。因此, 本文以医学上对细胞类型的识别和分类为例, 提出一种网络环境下基于多 Agent 技术的特征提取模型, 这种系统具有传统分布、并发问题求解的优点, 同时具有复杂的交互模式, 并具有一定的通用性和可扩充性, 可以有效地完成分布式环境下海量数据的特征提取工作。

### 1 基于多 Agent 的特征提取模型构造和分析

#### 1.1 问题的提出

图 1 为四个细胞的组织结构图。

图中  $B_i$  表示细胞内部的组织结构编号, 每个组织旁边的数字代表该组织的重量(数字 1 代表一个重要单位, 其他类

推)。假设某项工作需要从全国各地的数百家单位的数据库系统中对类似的细胞结构图进行特征提取, 从而获得细胞在组织结构上的特征, 并由规则库中的正常细胞和癌变的判别规则自动分辨细胞种类。

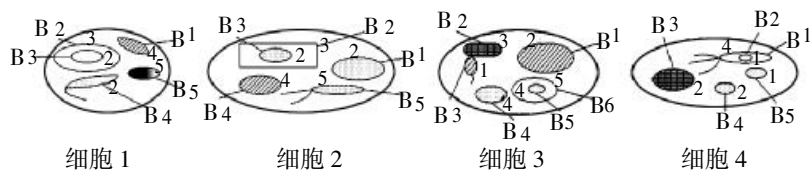


图 1 四个细胞的组织结构图

#### 1.2 Agent 内部结构

系统中所使用的 Agent 都可以通过修改 Agent 通用结构来实现其功能。KQML(Knowledge Query and Manipulation) 是 Agents 间交换信息和知识的语言及协议, 为消息的表达提供标准格式, 用于支持 Agent 间的实时知识共享<sup>[3]</sup>。Agent 间的所有消息都是基于 KQML 封装<sup>[4]</sup>。本文中所使用的 Agent 的组成结构图如图 2 所示。Agent 组成结构中各部分含义如下:

- (1) Agent 标志——是 Agent 之间相互区分的属性特征, 如 Agent 的名称、地址等。
- (2) 用户接口模块——用于和用户或系统的管理人员进行交互。
- (3) 任务求解模块——包括运行模块和分析模块。运行模块包括 Agent 的初始化和消息处理。分析模块根据 Agent 的目标、知识和最新信息进行分析。
- (4) 学习模块——从系统的运行中总结经验, 为知识库增加新的知识或规则。
- (5) 数据库——保存 Agent 运行时所需的数据、中间结果和由 Agent 采集处理并将要发送回用户的数据。
- (6) 知识库——保存 Agent 运行时所需的知识和规则, 并保存在任务求解过程中获取的知识和任务求解结果。

收稿日期: 2004-03-28; 修返日期: 2004-06-28

基金项目: 国家自然科学基金资助项目(70171061)

(7) 内部状态集——是 Agent 执行过程中的当前状态, 它影响 Agent 的任务求解过程, 同时 Agent 的任务求解又作用于内部状态。

(8) 约束条件集——Agent 创建者为保证 Agent 的行为和性能而设置的约束参数的集合, 如响应时间等。

(9) 通信模块——用于 Agent 之间的交互, 实现消息的收发、任务的传递等。

### 1.3 多 Agent 系统模型的建立的分析

如图 3 所示, 本文构建的基于多 Agent 的细胞特征提取系统分为两大类, 即管理协调类 Agent 和特征提取类 Agent 模块。

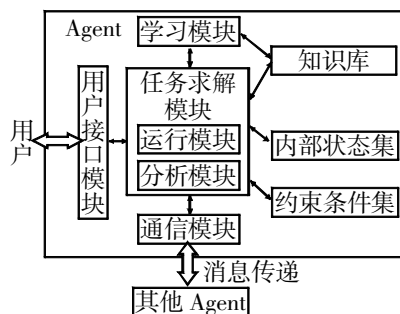


图 2 Agent 的组成结构

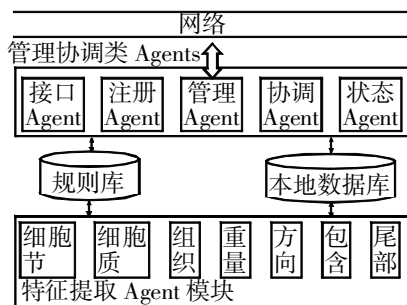


图 3 基于多 Agent 的细胞特征提取系统

管理协调类 Agents 由接口 Agent、注册 Agent、管理 Agent、协调 Agent 和状态 Agent 组成。接口 Agent 负责人机界面, 与本地系统上的管理人员进行通信。当网络上某一单位的本地管理人员通过接口 Agent 启动系统后, 系统产生一个种子 Agent (称为父 Agent), 这个种子 Agent 将从本地规则库中读取本次特征提取所需使用的算法或规则。在父 Agent 产生后, 注册 Agent 更改系统的特征标志为 1, 并且父 Agent 开始自我复制, 向网络上所有的活动着的系统发送它的拷贝 (称为子 Agent)。对于网络上其他单位的, 如果其系统特征标志为 0, 则接收子 Agent, 子 Agent 将本身所包含的本次特征提取所需使用的算法或规则存入当前系统的规则库中, 并通过注册 Agent 更改本地系统的特征标志为 1; 如果其系统的特征标志已经为 1, 则拒绝接收。当所有活动系统的特征标志都为 1 时, 父 Agent 停止复制。

管理 Agent 主要负责特征提取任务执行中的突发事件。它从状态 Agent 收集系统目前执行的状态信息并进行分析识别, 当突发情况或非正常情况出现时, 它及时报告协调 Agent, 并通过接口 Agent 反馈给管理人员。协调 Agent 收到管理 Agent 的报告后, 可以通过接口 Agent 接收管理人员的指令, 也可以根据其内部条件集来判断是否中止系统的运行或者改变任务的执行方式。状态 Agent 是一个移动 Agent, 当系统开始执行任务时, 它移动到特征提取类 Agents 模块中, 观察和收集相关系统信息, 并发送给管理 Agent, 从而保证系统在有效的监控当中。

## 2 系统的实现和运行

从图 1 中可以提取这四个细胞的组织特性, 其中细胞质形状有鱼形 (Fish)、圆形 (Circle)、椭圆形 (Ellipse)、方形 (Square)、多边形 (Heptagon)。组织特点有空白 (Blank)、阴影 (Shaded)、点 (Spot)、十字 (Crossed)、条纹 (Stripes) 等。其细胞特征的具体描述及定义如表 1 所示, 提取了细胞七个方面的特征, 并给出了相应的类型描述和定义域。

表 1 细胞特征描述及定义

名称	定义	类型	定义域
Cire	细胞节个数	线性的	{1, 2, ..., 10}
Shape	细胞质形状	结构化	{Fish, Circle, Ellipse, Square, Heptagon}
Texture	组织	外观的	{Blank, Shaded, Spot, Crossed, Stripes}
Weight	重量	线性的	{1, 2, ..., 8}
Orient	方向	线性环链	{N, NE, E, SE, S SW, W, NW}
Contains	包含	外观的	{T, F}
HasTails	尾部	外观的	{T, F}

在本系统中, 细胞的特征分为细胞节个数、细胞质形状、组织、重量、方向、包含和尾部共七种, 因此产生了七个特征提取 Agent, 每一个特征提取 Agent 负责处理细胞组织结构图的一个特征的提取工作。对于提取的结果存入数据库中, 并根据规则库中的细胞判别规则标记每一张细胞图。当出现异常, 或是无法用已知规则进行判别时, 特征提取模块将提取到的细胞特征描述存入规则库, 并通过接口 Agent 反馈给管理人员。

假设初始规则库中正常细胞和癌变细胞的判别规则为: 癌变细胞的节点个数都为偶数, 而正常细胞的节点个数为奇数; 癌变细胞的 Fish 形细胞质的尾数为 1, 而正常细胞的 Fish 形细胞质尾数都大于 1; 癌变细胞的 Fish 形细胞质的重量都很小, 而正常细胞的 Fish 形细胞质重量较大; 癌变细胞的 Fish 形细胞质的方向不确定, 而正常细胞的 Fish 形细胞质方向都为 E。

本文采用 JAFMAS 来实现所建立的多 Agent 模型<sup>[6]</sup>。JAFMAS 是一个面向 MAS 的基于 Java 的 Agent 框架。它提供了解决 MAS 系统中 Agent 之间通信行为的一类通用方法, 给出了一个 Agent 的抽象结构, 并使用 Java 编写的一系列类来支持和实现这些类。JAFMAS 总共含有 16 个类, 分别提供了 Agent 通信、协作等方面的支持。

表 2 每个细胞的特征描述

细胞编号	Cire	Shape	Texture	Weight	Orient	Contains	HasTails
细胞 1	6	B1 ellipse	stripes	4	NW	F	F
		B2 circle	blank	3	—	T	F
		B3 circle	blank	2	—	F	F
		B4 fish	spot	2	SW	F	T(1)
		B5 circle	shaded	5	—	F	F
细胞 2	8	B1 circle	stripes	2	—	F	F
		B2 heptagon	crossed	3	E	F	F
		B3 fish	stripes	1	N	F	T(1)
		B4 circle	spot	4	—	F	F
		B5 circle	spot	4	—	F	F
		B6 circle	blank	5	—	T	F
细胞 3	7	B1 circle	spot	2	—	F	F
		B2 square	blank	3	—	T	F
		B3 circle	spot	2	—	F	F
		B4 circle	stripes	4	—	F	F
		B5 fish	spot	5	E	F	T(2)
细胞 4	5	B1 fish	Blank	4	E	T	T(3)
		B2 circle	Blank	1	—	F	F
		B3 circle	crossed	2	—	F	F
		B4 circle	spot	2	—	F	F
		B5 circle	spot	1	—	F	F

系统中主要需要构建的类如下: 继承 JAFMAS 中的 CreateAgent 类, 生成 CreateSCMAgent 类。这个类负责系统中其他 Agent 的生成和初始化工作。通过扩展 JAFMAS 的 Conversation 类生成用户 Agent 中的用户通信类 (User Conversation Class), 目标是定义各 Agent 只见的通信机制。继承 JAFMAS 的 ConvRule 类生成本系统中的各个规则类。这些规则包含了本地规则库中的正常细胞和癌变细胞的判别规则, 定义了系统中各个 Agent 内部的知识库。系统构建完毕后, 可以使用以下的 Java 命令来编译和启动系统。 (下转第 44 页)