

多智体在 IDSSIM 结构设计中的应用^{*})游福成¹ 苏占东² 谢永红¹ 杨炳儒¹(北京科技大学信息工程学院 北京100083)¹ (华北水利水电学院 郑州450008)²

摘要 本文讨论了基于信息挖掘的智能决策支持系统 IDSSIM 的总体结构模型和各子系统功能描述。IDSSIM 完善并扩充了现有智能决策系统的功能,将领域专家的知识获取、推理机制、KDD^{*}挖掘和 Web 挖掘有效地融入智能决策支持系统中,从而形成了一类具有双网、六库、综合集成、多层递阶和模型驱动的新型智能决策支持系统。

关键词 信息挖掘,智能决策,KDD^{*},Web 挖掘,推理机制

The Application of Multi-agent in Structure Design of IDSSIM

YOU Fu-Cheng¹ SU Zhan-Dong² XIE Yong-Hong¹ YANG Bing-Ru¹(School of Information Engineering, Beijing University of Science & Technology, Beijing 100083)¹(North China Hydraulics & Hydroelectricity College, Zhengzhou 450008)²

Abstract This paper mainly discusses the general structure model of IDSSIM and describes the function of each of its sub-systems. IDSSIM has perfected and extended the function of conventional IDSS. Knowledge acquiring of domain experts, reasoning mechanism, KDD^{*} mining and Web mining are efficiently fused into IDSS. Consequently, IDSSIM is a new type of Intelligent Decision Support System that covers two networks and six bases. Furthermore, it is synthetically integrated, multi-tiered, and model-driven.

Keywords Information mining, Intelligent decision, KDD^{*}, Web mining, Reasoning mechanism

1 引言

自从美国麻省理工学院的 Michael S. S. Morton 和 Peter G. W. Keen 于20世纪70年代首次提出决策支持系统(DSS)以来,其发展迅速,并成为许多行业经营管理中一个不可缺少的现代化决策支持工具。现在正逐步形成新一代的 DSS:群决策支持系统(GDSS),分布式决策支持系统(DDSS),战略决策支持系统等。智能决策支持系统(IDSS)的出现,将人工智能的知识推理技术引入 DSS,使 DSS 的发展进入了一个新的阶段。

但是,这些决策支持系统尚存在着推理技术单调、自学习能力较差、知识库中的知识不够丰富的“瓶颈”现象。近年来 Internet 迅速发展,网上信息极大丰富而知识却相对缺乏。如何处理这些非结构化的信息并用于决策,是新一代决策支持系统的发展方向。

为此,我们提出了基于信息挖掘的智能决策支持系统 IDSSIM (Intelligent Decision Support System Based on Information Mining),这里的信息挖掘区别于通常的数据挖掘,是从结构化数据及复杂类型数据(文本、日志、音频、视频、图像、图形等)中提取新颖、潜在有用知识的非平凡抽取过程。IDSSIM 的理论基础是我们提出的基于双库协同机制的知识发现系统 KDD^{*}[1]和 Web 数据挖掘过程[7]。

2 基于信息挖掘的智能决策支持系统总体结构模型

2.1 智能体 (Agent) 及多智体 (Multi-agent)

Agent 是一个运行于动态环境的、接受另一个实体的委托并为之提供服务的、具有较高自治能力的实体,是一种模拟

人类智能行为并提供相应服务的计算机程序。Agent 能够通过感知器对外界环境中的条件作出反应,并对外界环境作出解释,产生推理并作出决定。Agent 具有以下特点^[9]:自主性、协同性、响应性、预动性及智能性。

Multi-agent 系统是指由多个 Agent 组成的一个较松散的多 Agent 联邦,这些 Agent 成员之间相互协同、相互服务、共同完成一个任务。当 Agent 之间出现依赖关系时,必须通过相互协商与合作来加以控制与管理。

2.2 基于信息挖掘的智能决策支持系统总体多智体结构模型

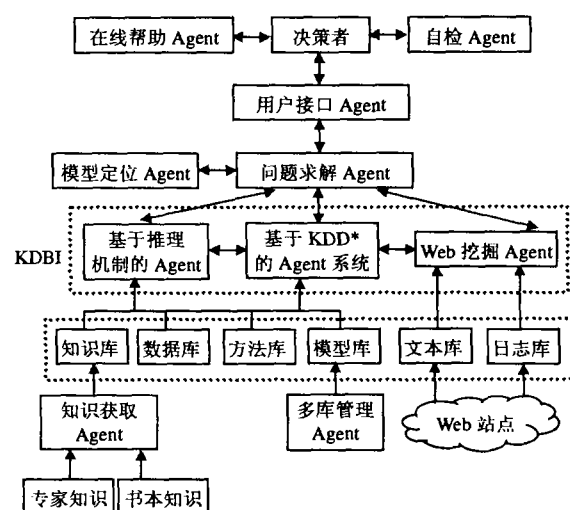


图1 系统多智体结构模型

^{*}) 本文得到国家自然科学基金重点项目(69835001)、教育部科技重点项目(教技司[2000]175)及北京市自然科学基金项目(4022008)支持。杨炳儒 教授,博士生导师,研究领域为数据挖掘与知识发现、柔性建模与集成技术;游福成、苏占东、谢永红 博士研究生,研究领域为知识发现。

基于信息挖掘的智能决策支持系统总体多智体结构模型如图1所示,它由多个 Agent 组成,是一个多库协同、双网(Internet 和 Intranet)融合、统一调度的复杂系统。

可以看出,系统的核心是“源于信息的知识发现 KDBI”(Knowledge Discovery Based on Information),它在模块实现上主要包含三个部分:基于双库协同机制的知识发现(KDD*)Agent^[2]、基于推理机制的知识发现(KDRM)Agent和基于 Web 挖掘的知识发现(KDWM)Agent。可粗略地表示为:KDBI $\triangleq$ KDD*+KDRM+KDWM。

IDSSIM 以多个知识源的知识融合、多抽象级与不同知识层次的结构,以及使数据库、知识库、方法库、模型库、文本库、日志库六库协同运作为特征,形成了丰富的动态知识库系统与相应的集成推理机制,为解决决策系统构造中的核心技术提供了一条有效的途径。

下面首先介绍 IDSSIM 的理论基础——基于语言场与语言值结构的知识表示方法,然后就各个子系统分别进行讨论。

2.3 理论基础——基于语言场与语言结构的知识表示方法

本系统是以语言场与语言值结构为知识表示方法,以属性为基础构建了知识素节点和数据子类结构中层之间的对应关系,从而在相应的论域内,以属性为基础将规则库化为若干规则子库,每一规则子库与挖掘数据库相对应,从而为定向数据挖掘奠定了基础。

定义1 在语言变量相应的基础变量论域中,各个被划分的交叉区间的中点连同 ϵ -邻域(ϵ 通常为允许误差值)内的点,称为标准样本(点),其取值邻域称为标准值;其余诸点称为非标准样本(点),其取值称为非标准值。它们分别构成标准样本空间与非标准样本空间,并统称为一般样本空间。

定义2 $C = \langle D, I, N, \leq_N \rangle$, 若满足下列条件: 则称 C 为语言场。

(1) D 为基础变量论域 R 上交叉闭区间的集合, D^+ 为其对应开集;

(2) $N \neq \Phi$ 为语言值的有限集;

(3) $\leq_N$ 为 N 上的全序关系;

(4) $I: N \rightarrow D$ 为标准值映射, 满足保序性, 即: $\forall n_1, n_2 \in N$
 $n_2 \wedge n_1 \leq_N n_2 \rightarrow I(n_1) \leq I(n_2)$ ($\leq$ 为偏序关系)

定义3 对于语言场 $C = \langle D, I, N, \leqslant_N \rangle$, 称 $F = \langle C, W, K \rangle$ 为 C 的语言值结构, 如果:

(1)C 满足定义2;

(2) K 为自然数;

(3) $W: N \rightarrow R^k$ 满足:

$$\forall n_1, n_2 \in N (n_1 \leq_N n_2 \rightarrow W(n_1) \leq_{dc} W(n_2), \forall n_1, n_2 \in N (n_1 \neq n_2 \rightarrow W(n_1) \neq W(n_2)))_0.$$

其中, $\leq_{\text{dic}}$ 为 $[0, 1]^k$ 上的字典序, 即 $(a^1, \dots, a^k) \leq_{\text{dic}} (b^1, \dots, b^k)$ 当且仅当存在 h , 使得 $0 \leq j < h$ 时 $a^j = b^j, a^h \leq b^h$ 。

3 各子系统功能

3.1 基于 KDD' 的 Agent 系统

KDD^{*}利用双库协同机制构建了数据库与基础知识库的内在联系“通道”,用基础知识库去制约与驱动 KDD 的发掘

过程,改变 KDD 固有的运行机制,提高了知识发现系统的认知自主性。粗略地讲:KDD' $\triangle$ KDD+双库协同机制。

KDD* 的结构模型由多个单 Agent 组成,如图2所示。

系统的运行机制由维护协调 Agent 和启发协调 Agent 完成。维护协调 Agent 的主要功能是:当从真实数据库的大量数据中经聚焦而产生假设规则(知识)后,使 KDD 进程产生“中断”,而去定向搜索知识库中有无此生成规则的重复、冗余和矛盾。若有重复与冗余,则取消该生成规则而返回 KDD 的“始端”;若有矛盾,则在领域专家的介入下,根据可信度与规则强度取消矛盾规则或用扩展前提等方法消除矛盾;若无重复和矛盾,则继续 KDD 进程,即评价后结果入库。启发协调 Agent 的主要功能是:在以属性为基础的知库建库原则下,通过搜索知识库中知识结点的不关联态,以发现知识短缺,产生“创思意向”,并根据关联强度排出定向发掘的优先级,去启发与激活真实数据库中相应的数据子类结构,以产生定向发掘进程。

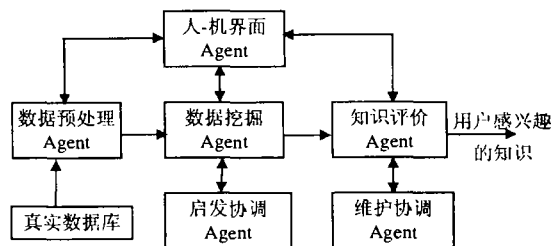


图2 基于 Multi-agent 的 KDD' 模型结构图

3.2 基于推理机制的 Agent

它包括用一种或多种知识表示方法描述的已知问题及其解法的描述集合的知识库和需要求解的问题集;通过 Fuzzy 推理、演绎推理、广义综合归纳推理和基于案例的推理等,构造规则集和新知识;并由决策树提供问题的求解过程。决策树用 SLIQ 算法^[6]来实现,在选择树的分叉时,同时考虑相关的多个属性,从而提高规则的产生效率。

另外,通过 T 协调 Agent 激发人机交互以“理解”知识并将先验知识应用于知识发现中,并且可以将发现的规则返回 KDD* 对其进行评估,使 KDD* 和基于推理机制的知识发现有机地结合起来。

3.3 Web 挖掘 Agent

Web 挖掘是指从大量 Web 文档的集合 C 中发现隐含的模式 p 。如果将 C 看作输入, 将 p 看作输出, 那么 Web 挖掘的过程就是从输入到输出的一个映射 $\xi: C \rightarrow p$ 。

Web 挖掘 Agent 主要用于处理 Web 页面中的大量非结构化或半结构化数据的知识发现问题。根据任务源的不同,又可细分为用户访问信息挖掘、文本挖掘和结构挖掘。

访问信息挖掘是指对用户访问 Web 时在服务器日志文件中留下的访问记录进行挖掘。通过对 Web 日志挖掘, 站点管理员可以发现用户的浏览模式, 以改进 Web 服务器的设计和提高 Web 服务器的性能, 增加个性化服务, 在电子商务中发现潜在的客户群。在 Web 日志挖掘中可以挖掘关联规则、序列模式、频繁遍历路径, 对用户的行为进行预测和聚类分析。

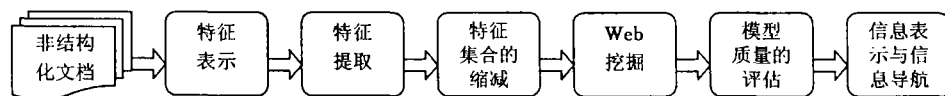


图3 Web 挖掘过程示意图

结构挖掘是指从 WWW 的组织结构和链接关系中推导知识。利用这些信息,可以对页面进行排序,发现重要的权威页面和高质量 Web 结构和资源。

文本挖掘是指从 Web 页面内容或其描述中进行挖掘,进而抽取知识的过程;Web 文本挖掘的基本过程如图3所示,主要包括特征提取、Web 挖掘、模型质量评价和信息表示与信息导航等几部分。

这里,我们给出 Web 文本特征抽取中的权值评价函数

$$W_{ik} = \frac{f_{ik} \log\left(\frac{N}{n_k} + 0.01\right)}{\sqrt{\sum_{k=1}^n (f_{ik})^2 \cdot \log^2\left(\frac{N}{n_k} + 0.01\right)}}$$

(其中 f_{ik} 表示词条 T_k 在文档 D_i 中出现频率数, N 表示全部样本文档总数, n_k 表示词条 T_k 的文档频率数)和信息增益评价函数

$$\text{InfGain}(F) = P(W) \sum_i p(C_i|W) \log \frac{P(C_i|W)}{P(C_i)} + P(\bar{W}) \sum_i p(C_i|\bar{W}) \log \frac{P(C_i|\bar{W})}{P(C_i)}$$

(其中 F 为对应于单字 W 的特征, $P(W)$ 为单字 W 出现的概率, $\bar{W}$ 意味着单字 W 并不出现, $P(C_i)$ 为第 i 类值的出现概率, $P(C_i|W)$ 为当单字 W 出现时属于第 i 类的条件概率),以指导文本挖掘过程和评价挖掘效果。

3.4 模型定位 Agent

主要用于问题模型的识别、自动查找、匹配和结构选择:制定面向问题和用户需求的灵活的解题规划,对解题、决策过程进行协调控制;调用综合知识库的知识、数据、模型和方法执行协同解题操作和控制及存取管理,是系统进行下一步操作的引导器。

3.5 问题求解 Agent

主要用于识别待解问题,理解用户需求,存储和管理决策过程的中间结果和最终结果,并向用户解释。

3.6 知识获取 Agent^[3,4]

该 Agent 通过良好的人机交互界面,对专业人员进行经验训练;通过实例、书本直接得到数据,从中发现知识,完成专业领域知识的学习功能;还可以通过对决策过程和决策结果的分析,进行学习,从而改进协调策略和决策方法,以适应不同用户的需求。

3.7 多库管理 Agent

多库管理 Agent 分为数据库管理模块、知识库管理模块、方法库管理模块、模型库管理模块、文本库管理模块和日志库管理模块,主要功能是进行编辑、删除、检索、查询、添加、备份等管理。

同时,多库管理 Agent 还有一个重要的功能,即协调六库,进行调度和通信;使各库既可以独立工作,又可以相互配合协同工作。

3.8 自检 Agent

为避免因测试硬件本身的故障引起错误,在进行用户使用之前,自检 Agent 将对测试硬件进行一次闭环的自我检测。

3.9 在线帮助 Agent

为使用户更加有效、方便地使用本系统,随时能够得到相关信息,在线帮助 Agent 可为用户提供实时在线帮助。

至此,我们得到了 IDSSIM 的总体过程,而此过程具有多个抽象级别,是不同知识层面的多层递阶、综合集成的决策支持系统。因为当按照上述构筑的进程可以完成第一个发现(认识)阶段,即第一抽象级。以第一抽象级的综合知识库作为第

二抽象级的基础知识库,按类似于第一抽象级的各个发现环节运行,完成第二抽象级。如此往复,在认识发展与时空环境变迁的不同阶段,不断使知识丰富与升级,不断使知识深化。

结论 IDSSIM 的构建属于原创性的工作,并在 Windows2000 与网络环境下,使用 VC6.0 完成了系统设计。在实例运行中,我们以世界某地区社会调查结果的部分数据资料为数据源,对系统进行了试运行,运行结果如图4所示,其结果良好。

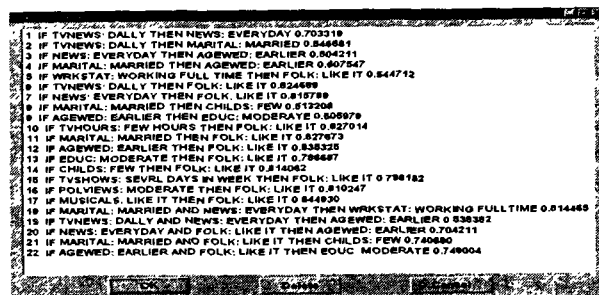


图4 系统决策结果

综上所述, IDSSIM 有如下特征:

(1) 知识源的丰富性:传统的决策支持系统仅从结构化信息中得到决策知识,而 IDSSIM 不仅可处理结构化信息,还可处理 Web 上的非结构化信息,同时还可以从领域专家那里获取知识,使知识源得到扩充,数量和质量得到丰富和提高。

(2) 双网、六库的协同性: IDSSIM 系统采用多库协同方法,对六库进行调度与通信,使各库既可以独立工作,又可以相互配合协同工作。另外,内部网(局域网)与外部网(互连网)彼此独立地提供信息源与进行信息挖掘;同时有交互作用,实现信息融合。

(3) 独立性:采用结构化系统分析的方法,将整个系统分成相对独立的可完成不同功能的多个子系统。各子系统既可以协同工作,又可以独立地被不同用户使用。

(4) 自学习和自适应性:通过协调器的作用、知识学习以及训练方式可提高系统自学习的功能,并把不断获得的新知识加入到动态知识库中;同时基于知识发现的动态知识库及数据库都在时间与空间的延伸中,随着抽象级的增加而再生出适合于环境变迁的新知识,使整个系统具有较强的自适应性。

参考文献

- Yang Bingru. KD (D&K) and Double-Base Cooperating Mechanism. Journal of System Engineering and Electronics, 1999, 10(1): 48~54
- KDD 中内在机理研究(Ⅰ). 中国工程科学, 2002, 4(5): 34~43, 78
- 杨炳儒. 专家知识的归纳获取. 计算机研究与发展, 1997, 34(7): 492~496
- 杨炳儒. 基于知识发现的广义诊断型专家系统构造方法. 模式识别与人工智能, 1999, 12(4): 373~380
- Fayyad U, Shapiro P G, Smyth P. Knowledge Discovery and Data Mining: Towards a Unifying Framework. In: proc. Of KDD-96, Menlo park, CA: AAAI, 82-88
- Anand S S, Bell D A, Hughs J G. EDM: A General Framework for Data Mining Based on Evidence Theory. Data & Knowledge Eng., 1996, 18: 189~223
- Adamic L A, Huberman B A. The Web's Hidden Order. Communication of the CAM, 2001, 44(9): 55~59
- Meghabghab G. Discovering authorities' and hubs in different topological Web graph structures. Information Processing and Management, 2002, 38(1): 111~140
- 李碧蓉, 肖德宝. 一种基于智能移动 Agent 的网络管理模型. 计算机科学, 2000, 27(8): 11~12