多 Agent 协作的强化学习模型和算法*)

刘 菲 曾广周 宋言伟
(山东大学计算机科学与技术学院 济南 250061)

摘 要 结合强化学习技术讨论了多 Agent 协作学习的过程,构造了一个新的多 Agent 协作学习模型。在这个模型的基础上,提出一个多 Agent 协作学习算法。算法充分考虑了多 Agent 共同学习的特点,使得 Agent 基于对动作长期利益的估计来预测其动作策略,并做出相应的决策,进而达成最优的联合动作策略。最后,通过对猎人-猎物追逐问题的仿真试验验证了该算法的收敛性,表明这种学习算法是一种高效、快速的学习方法。

关键词 协作学习,强化学习,多 Agent 学习,学习模型,学习算法

Reinforcement Learning Model and Algorithm Based on Multi-agent Cooperation

LIU Fei ZENG Guang-Zhou SONG Yan-Wei
(School of Computer Science and Technology, Shandong University, Jinan 250061)

Abstract The multi-agent cooperative learning process based on Reinforcement Learning is addressed and a new multi-agent cooperative learning model is proposed. Based on this model, a cooperative learning algorithm is introduced. This algorithm pays fully attention to multi-agent cooperative learning together simultaneity, so it can make each agent predict its action policy based on the estimation on its action's long-time reward. At last relevant decisions to be the best associated action policy is made. We conduct a series of empirical evaluation of the algorithm on the hunter-prey problem to validate its astringency. The result shows this algorithm is an efficient and fast method for multi-agent learning.

Keywords Cooperative learning, Reinforcement learning, Multi-agent learning, Learning model, Learning algorithm

1 引言

学习是人类智能的重要表现形式,赋予 Agent 学习能力是提高它的智能,增强它适应能力的必由之路。有了学习能力,Agent 才能对变化的环境作出及时的调整,对于未知的情形作出恰当的反应,同时也能够简化设计人员的设计工作。各种学习方法中,强化学习[1](Reinforcement Learning)和人类通过尝试获得知识的过程十分相似,获得了广泛的关注。通过强化学习的试错模式,Agent 可以逐渐获得某种有意义的行为,可以通过调节自己的行为参数提高性能,也可以在多 Agent 协作的层次上通过学习改善整个系统的性能。

Tan[2]提出在协作的多 Agent 环境下采用 3 种协作强化学习的方式,即共享感知、共享策略和共享经历时间。不过这 3 种协作强化学习方法的共同之处在于系统中每个时刻只有一个 Agent 在学习,其余 Agent 只是在辅助其学习。蔡庆生和张波以 Q 学习算法[3]为基础,提出了一种基于 Agent 团队的强化学习模型[4]。这个模型的最大特点是引入主导 Agent 的角色作为团体学习的主角,并通过 Agent 角色的变换实现整个团队的学习。但实际上在该学习模型中只有主导 Agent 承担学习任务,即每次同时最多只有一个 Agent 在学习。Kevin Irwig[5]将 TD(0)算法和 Q 学习算法应用到 Tileword 领域,提出了一种促使 Agent 学习协作动作的替代强化方法,也就是说 Agent 采取动作所得到的回报,不应该只考虑得到直接回报,还应该考虑其它 Agent 的回报,该动作的真正回报应该是自己的回报和其它 Agent 回报的加权和。

本文所研究的与以上方法不同,关注的是同时有多个 Agent 共同学习的问题。在此基础上,提出了一种基于强化学习的多 Agent 协作学习模型,通过利用其它 Agent 的经验和知识,一个学习 Agent 可以更快速地学习,并且尽可能地减少错误。Q 学习是一个有效的无模型的强化学习算法,对于单 Agent 学习,它是一个集中式的、有效的学习算法。但是,对于多 Agent 学习来说,由于状态空间呈指数增长和 Agent 的数量不断增多,Q 学习算法的代价过于庞大。本文提出了一个多 Agent 协作学习算法,在该算法中,利用一个协作学习方法——对长期得益的估计,每个 Agent 通过观察协作者的历史动作来预测其动作策略,并做出相应的决策,进而达成最优的联合动作策略。

本文第 2 节给出了强化学习有关的概念定义;第 3 节中首先提出一个多 Agent 协作学习模型,然后在此基础上,提出一个多 Agent 协作学习算法;第 4 节中通过模拟试验验证了该算法的有效性。最后给出结论以及未来研究的方向。

2 相关概念

强化学习的原理是 Agent 对环境执行某种动作,改变环境的状态并获得环境给予的回报信号来强化某一状态与最优动作策略之间的映射关系,反复执行这一过程,Agent 可获得在任意环境状态下给出最优动作策略的能力。

假设 Agent 和环境之间的交互可以看作是一个马尔可夫决策过程(Markov decision processes, MDP)[6]。MDP 模型意味着,Agent 感知到的目前的环境状态和 Agent 所选择的

*)国家自然科学基金项目资助(编号:60573169)。刘 菲 博士研究生,研究方向为移动计算、智能计算;曾广周 教授,博士生导师,主要研究方向:CSCW、智能计算、移动计算;宋言伟 博士研究生,研究方向为 CSCW、ad-hoc,工作流。

动作，将一起决定一个固定的概率分布，决定下一个状态及即时回报。这个模型是非记忆型的，Agent 在决定最优策略时，不需要记忆以前的状态和动作。

一个单 Agent MDP 被定义为一个四元组 (S,A,P,R)，其中 S 是一个有限的状态集，A 是一个有限的动作集，P 是转移函数，R 是奖励函数。状态转移函数 $P:S\times A\times S\to[0,1]$ 表示在状态 $s\in S$ 下执行动作 $a\in A$ 转移到状态 $s'\in S$ 的概率。奖励函数定义为一个实数值 $\mathcal{R}$ 限定函数 $R:S\times A\times S\to\mathcal{R}$。在这个 MDP 模型中，一个确定的策略被定义为一个函数，给 MDP 的每一个状态赋值。在一个策略 π 下某一状态的动作值 $Q^{\pi}(s,a)$ 就是在状态 s 下采取动作 a 并执行策略 π 所获得的总折扣报酬的期望。

Q 学习算法是无需环境模型的强化学习的一种形式，它提供 Agent 在 Markov 环境中利用已经历的动作序列执行最优动作的一种学习能力。在 Q 学习中，Agent 初始化一个任意的值 $Q(s,a)$，$s\in S$，$a\in A$。在每个时间 t，Agent 选择一个动作并观察它的奖励 r_t。Agent 基于下面的等式更新它的 Q 值：

$$Q_{t+1}(s,a)=(1-\alpha_1)Q_t(s,a)+\alpha_t\left[r_t+\gamma\max_b Q_t(s',b)\right]$$

其中，$\alpha_t\in[0,1]$ 是学习率，γ 称为折扣因子。

3 多 Agent 协作的强化学习模型和算法

学习 Agent 的特性主要有 3 点：1）能够接受外界实体的委托并为其提供帮助和服务；2）能够感知、调整自身以适应外界环境的变化；3）主动采取包括通讯、学习或推理等手段在内的各种智能行为。

3.1 多 Agent 协作学习模型

多 Agent 协作学习是指多个 Agent 在追求一个共同的目标过程中彼此相互通信、合作，由于获取信息而改变自身状态和周围环境，每个 Agent 在学习过程中受到其它 Agent 的知识、信念、意图等的影响。与单 Agent 学习相比，多 Agent 协作学习提出了一个新的问题：交互的 Agent 通过交换信息或者改变它们所处的环境，都可以显著地影响其它 Agent 的个体学习，特别当多个 Agent 试图以团队的形式去完成单个 Agent 不能完成的学习任务时。通过共享传感信息来克服隐藏状态的问题，通过强化信号的共享，让 Agent 进行合作行为的学习。

Bowling 等人[7]利用马尔可夫决策过程给出了群体 Agent 环境下的学习模型 $\langle n,S,A_{1,\cdots,n},T,R_{1,\cdots,n}\rangle$，其中 n 表示 Agent 个数，A_i 表示第 i 个 Agent(Ag_i)的可选动作集合，R_i 表示 Ag_i 的立即奖励。本文在其工作基础上，定义了多 Agent 协作学习模型，如图 1 所示。

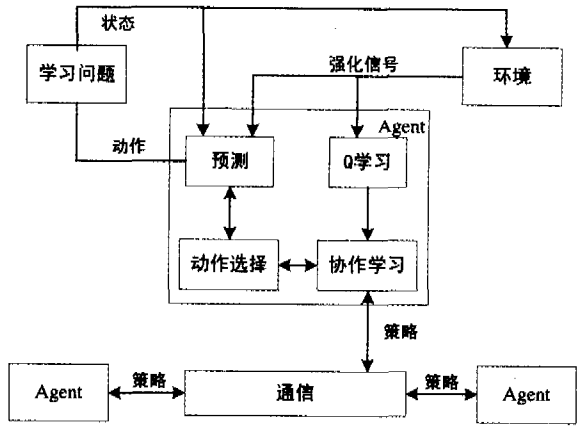

图 1　多 Agent 协作学习模型

预测模块：对其它 Agent 动作进行预测，通过观察其它 Agent 的动作执行情况，可按照不同预测策略对下一步其它 Agent 要执行的动作做出预测。

动作选择模块：根据对其他 Agent 的动作预测和相应 Q 值，根据学习算法，以较大概率选择目前看来最好的动作策略，以较小的概率选择目前看来不太好的动作策略。

Q 学习模块：根据环境状态 s，奖励 r 和采取的协作动作 a 通过标准的 Q 学习方法完成不断的学习，来调整协作策略。通过协作学习，可以对环境变化及其他 Agent 的策略做出快速的反映，直到最优联合解产生为止。

多 Agent 协作学习模型定义为一个六元组 $\mathrm{MMDP}=\langle M,\{A_i\}_{i\in n},S,T,R,Q\rangle$，其中 M 表示参与协作学习的 n 个 Agent 组成的有限集合；$\{A_i\}_{i\in n}$ 表示对于每个 $Ag_i\in M$，都有一个有限动作集合 A_i，n 个 Agent 采取的联合动作 $a=\{a_1,a_2,\cdots,a_n\}$，$a_i\in A_i$ 构成了联合动作空间 $A=\times A_i$ 中的元素；S 是系统的状态空间；R 是奖励函数：$S\times A\to\mathcal{R}$；T 表示在随机环境下的状态转移函数，$T:S\times A\times S\to\Delta$，$\Delta$ 为环境状态空间 S 上的概率分布，满足 $\sum_{s'\in S}T(s,a,s')=1$；$Q$ 表示多 Agent 的 Q 值，它是通过以下公式进行更新的：

$$Q_{t+1}(s,a)=\begin{cases}Q_t(s,a), s\neq s_t \text{ or } a\neq a_t\\ r(s,a)+\sum_{s'\in S}T(s,a,s')\max_{a'\in\{A_i\}}Q(s',a')\\ s=s_t\ a=a_t\end{cases}\quad(1)$$

3.2 多 Agent 协作学习算法

当有其它学习 Agents 存在的情况下，Agent 的学习算法需要满足以下的标准。

在线学习：一个在线学习的 Agent 需要在任何时间了解它自身的运行性能。一个 Agent 在每一步都要仔细选择动作来维持一个合理的性能。一个学习算法要求是在线的，因为 Agents 通常与动态变化的环境交互，并且没有无偿试错的机会。

合理性：如果其它 Agents 的策略收敛到静态策略，那么学习算法就会收敛到一个最佳的响应策略上。这需要学习算法考虑到其它 Agents 的实际策略，所以独立学习是不可能的。

我们提出一个多 Agent 协作学习算法，该算法通过对协作者历史动作的采样，计算动作的长期得益的估计值 LR，估计值最大的相应动作是当前最好动作。LR 计算公式为：$LR(s,x)=\sum_{\substack{A_i=x\\ A_i\approx\Phi(s,b)}}\frac{K_t(s,\bar{A}_i)}{k}l(s)$，该公式所表示的是状态 s 下所考察的 Agent 基于协作者的动作来选择动作的概率模型。采取某个动作 x 时所获得的长期得益的估计值，也就是协作者的所有该状态下的可能动作以概率为权值的加权和。$\Phi(s,b)$ 表示的是在状态 s 下的阶段中，所考察的 Agent 的协作者 b 的最优解的响应动作集合。式中用于动作采样的常数 m 和 k 分别称为记忆长度和样本长度。对于每个动作 x 需要维护一个长度为 m 的队列，按照时间顺序存放协作者的响应动作作为采样源。$\frac{K_t(s,\bar{A}_i)}{k}$ 表示的是当访问状态 s 的次数为 t 时 $(t\geqslant m\geqslant k)$ 在 k 个样本中协作者采取动作 $\bar{A}_i$ 的频率。$l(s)$ 是

联合动作$\langle A_i,\overline{A_i}\rangle$的长期得益值。

单 Agent 学习的动作选择策略，一般应用最多的选择方法是 Boltzman 公式：$P(a)=\frac{e^{Q(s,a)/T}}{\sum_{a\in A}e^{Q(s,a)/T}}$，$a\in A$，其中 $P(a)$表示选择执行动作 a 的概率；T 是温度参数，体现探索的程度，它随着时间的延续而降低。本文提出了一种适合于多 Agent 协作强化学习的动作选择策略——基于 LR 的选择策略，其计算公式为：$BR_i(s)=\{a_i\mid a_i=\arg\max_{a_i\in A_i} LR(s,x)\}$。动作选择过程是这样的：当对于某一状态 s 的访问次数小于记忆长度 m 时，随机选择动作作为 x；否则依据 Pr' 选择动作 x。

根据 Bayes 公式，可得：

$$p(\overline{A_i}\mid A_i,s')=\frac{p(s'\mid\overline{A_i},A_i)\cdot p(\overline{A_i})}{p(s'\mid A_i)} \tag{2}$$

式中，$p(s'\mid\overline{A_i},A_i)$为采取联合动作后到达状态 s'的状态转移概率；$p(s'\mid A_i)$为 Agent 采取单独动作后到达状态 s'的状态转移概率。由式(2)得到的概率估计的乘积表示相应的组合动作的概率，即

$$p((A_i,\overline{A_i})\mid A_i,s')=\prod_{j\neq i}p(A_j\mid A_i,s') \tag{3}$$

根据以上的算法思路，给出算法：

(1)初始化

初始化状态的访问次数 $n(s)=0$

$\forall s\in S,a_i\in A_i$，$T(s,a,s')=\frac{1}{|S|}$，$r_i(s,a)=0$，初始化该状态下的协作者动作队列为空。

给定记忆长度 m 和样本长度 k。

(2)循环执行

①当 $n(s)\leqslant m$ 时，随机选择动作 x，否则状态 s 下依据下式给出的动作选择策略及动作 x：

$$BR_i(s)=\{a_i\mid a_i=\arg\max_{a_i\in A_i} LR(s,x)\}$$

其中，$LR(s,x)=\sum_{\substack{A_i=x\\ \overline{A_i}=\phi(s,b)}}\frac{K_i(s,\overline{A_i})}{k}l(s)$

②观察协作者的局部联合动作 y，联合动作 $a=\langle x,y\rangle$，转移的下一个状态 s'，该步动作的奖励为 r。

若 $A_i=x$，$\overline{A_i}=y$，更新如下变量：

瞬时得益 $p(s):=r$

长期得益 $l(s):=(1-\alpha)l(s)+\alpha(p(s)+\gamma p((x,y)\mid x,s')Q(s,a))$

其中，γ 是折扣因子($0<\gamma<1$)；α 是学习率；$Q(s,a)$是协作者的 Q 值，见公式(1)；$p((x,y)\mid x,s')$是组合动作的转移概率，见公式(3)。

③更新该状态下的协作者动作队列。

④$n(s):=n(s)+1$

若长期奖励 $l(s)$收敛到稳定值，则循环结束。

该算法中环境给予的回报是基于联合动作的，每个 Agent 基于这一回报是从全局的角度考虑学习过程，所求得的是最优的联合动作。它由于考虑后继状态和动作得益，采用联合动作的长期得益作为学习的线索，更有利于寻找最优联合动作策略。但是其中每个 Agent 由于需要预测其它 Agent 的动作，其学习过程较为复杂。

4 模拟试验

猎人-猎物追逐问题是由 Benda 等[8]提出的，这是在多 Agent 系统中的一个基本问题。该问题是由 4 个蓝色的 Agents(猎人)从一个格子领域的 4 个方向尝试捕获 1 个红色的 Agent(猎物)。Agent 的移动被限制在水平方向或者垂直方向的每个单元格。猎人 Agent 的移动是随机的。两个 Agents 不能允许同时占领同一个位置。

猎人 Agents 组成了一个协作的团队，通过 4 个猎人 Agents 围绕在猎物地 4 个方向上来共同捕获它。对于猎人，每次有五种可能的动作，向上，向下，向左，向右和静止。当 4 个猎人同时占领了猎物的所有邻接格时，这次猎取就是成功的。每一个猎人 Agent 都有它有限的观察范围，因而它不能在每一个位置上都能看到猎物。所以，猎人 Agents 的成功依赖于它们之间的协作。

我们模拟一个关于该问题的试验。H 代表猎人，P 代表猎物，如图 2 所示。该系统由一个 7×7 的网格和 5 个 Agent (4 个猎人和 1 个猎物)组成。当猎人位于猎物旁边任一相邻位置时，猎物就要设法逃脱。通过这样的试验，检验算法是否能够提高系统的学习效率，以及算法是否能够在尽可能少的时间内达到收敛。

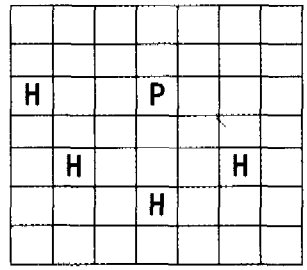

图 2 猎人-猎物问题

系统采用 5 台 PC 分别模拟独立的 Agent，并对上述问题进行了两个试验，两个试验的参数(记忆长度 m 和样本长度 k)设置为不同的值。每个试验进行 100 次，每次试验进行到 50 步时判定任务失败，重新开始新一次试验。以每 10 次试验为一组，共得到 10 组数据。试验对比结果如图 3 所示，图中横坐标表示试验的组数，纵坐标表示该组的成功概率。试验 1 中，记忆长度 $m=3$，样本长度 $k=2$，折扣因子 $\gamma=0.9$；试验 2 中，记忆长度 $m=4$，样本长度 $k=3$，折扣因子 $\gamma=0.9$。从这两个试验的结果可以发现在不同的参数设置下，该学习算法都能够使得成功概率收敛到区间[0.9,1]，而且参数的设置会影响收敛的速度和收敛的稳定值，m 是影响收敛稳定值大小的因素，k 是影响收敛速度的因素。

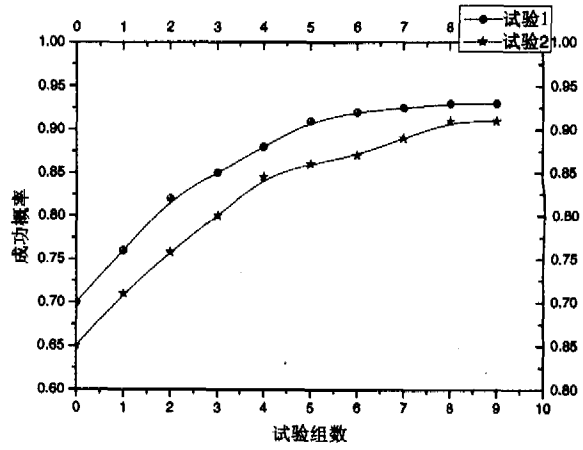

图 3 成功概率随学习过程的变化图

(下转第 186 页)

机学报,2003,26(3):310~322
4 樊兴华,仲昕,张勤,等. 因果图推理的一种新方法[J]. 计算机科学,2001,28(11):48~52
5 樊兴华,张勤,黄席樾. 可能性传播图模型的专家知识获取方法[J]. 计算机科学,2001, 28(1):53~56
6 张勤,樊兴华,黄席樾,等. 因果图用于复杂系统故障诊断研究[J]. 计算机工程与应用,2002, 38(4):43~47
7 Fan Xing-Hua, Sun Mao-Song. A reasoning algorithm of applying causality diagram to fault diagnosis of complex hybrid systems [A]. In: Proceedings of the World Congress on Intelligent Control and Automation [C]. Hangzhou, China, 2004, 2(5): 1741~1745
8 Shi Qingxi, Wang Hongchun, Zhang Qin. Intelligent fault diagnosis technique based on causality diagram [A]. Proceedings of the World Congress on Intelligent Control and Automation [C]. Hangzhou, China, 2004, 2(5): 1751~1755
9 王洪春,张勤. 基于因果图的一种近似推理算法[J]. 重庆大学学报(自然科学版),2004,27(8):96~99
10 沈文武,汪成亮,程克非,等. 因果图转换为信度网的算法[J]. 重庆大学学报(自然科学版),2004,27(10):33~36
11 Mitauo Y. The median for a L-R fuzzy number [J]. Microelectron Relia, 1995, 35 (2) : 269~271
12 杨伦标,高英仪. 模糊数学原理及应用[M]. 广州:华南理工大学出版社,1993
13 Zhang Qin, An Xuegao, Gu Jin, et al. Application of FBOLES-a prototype expert system for fault diagnosis in nuclear power plants [J]. Reliability Engineering and System Safety, 1991, 34 (2): 225~235

(上接第 158 页)

以上两个试验表明:在多 Agent 强化学习模型基础上使用学习算法,五个 Agent 都能够学习得到关于长期得益的知识,相互之间的预测能够逐渐准确,二者的联合动作更为连贯,且趋向最优联合动作,从而成功概率逐渐收敛为较高的稳定值,算法有效性得到了验证。另外,这种学习系统还可以推广应用到其它方面,如著名的 Taxi 问题、任务调度和机器人足球中。

结论与展望 多 Agent 学习技术使得系统能够适应不确定的环境,提高系统的问题求解能力。多 Agent 强化学习不必具备明确的环境模型,因此该方法在学习者对环境了解甚少的问题域中非常适用。本文的研究目标是针对多 Agent 协作团队这种协作系统中多 Agent 协作求解过程的特点,研究适合的学习方法,以提高协作求解的效率和系统的整体性能。提出了一个新的多 Agent 协作强化学习模型,这个学习模型能加快学习速率,并且降低状态空间和动作空间。基于这个模型,根据 Agent 动作的长期得益的估计,提出了一个新的动作选择策略,并实现了多 Agent 协作学习的学习算法。文中以一个猎人-猎物追逐问题为例,实现并应用了这一学习算法,试验结果表明多个 Agent 通过采用该学习算法最终找到最优联合动作策略。尽管该例讨论的是五个 Agent 的情况,这一结论可以推广用于多个 Agent 的情况。

本文的研究还可以根据具体应用领域进一步细化,同时进一步的研究将从两个方面展开:一是在相互依赖的竞争性目标下学习协作动作和多 Agent 协作团队在开放环境中进行协作求解时的学习方法;二是在 Agent 不完全知道其他 Agent 行为策略集的情况下,Agent 如何学习以获得最优策略。

参 考 文 献

1 Sutton R, Barto AG. Reinforcement Learning: An Introduction. MIT Press, 1998
2 Tan Ming. Multi-agent reinforcement learning: independent vs, cooperative Agents. In: Proceedings of the 10th International Conference on Machine Learning(ICML-93), 1993. 330~337
3 Watkins C J C H, Dayan P. Q-learning. Machine learning. 1992, 8:272~292
4 蔡庆生,张波. 一种基于 Agent 团队的强化学习模型与应用研究. 计算机研究与发展,2000,37(9)
5 Irwig K, Wobcke W. Multi-Agent Reinfoecement Learning with Vicarious Rewards. Electronic Transactions on Artificial Intelligence, 1999, 3(B):23~45
6 Mataric M J. Interaction and intelligent behavior: [Ph D Thesis]. Department of Electrical Engineering and Computer Science, MIT, USA, 1994
7 Bowling M. Convergence problems of general-sum multi-agent reinforcement learning [A]. In: Langley P, ed. Proceedings of the Seventeenth International Conference on Machine Learning [C]. San Francisco: Morgan Kaufmann Publishers, 2000. 89~94
8 Benda M, Jagannathan V, Dodhiawalla R. On optimal cooperation of knowledge sources: [Technical Report]. BCS-G2010-28. Boeing AI Center, Boeing Computer Services, Bellevue, WA, August 1985

(上接第 168 页)

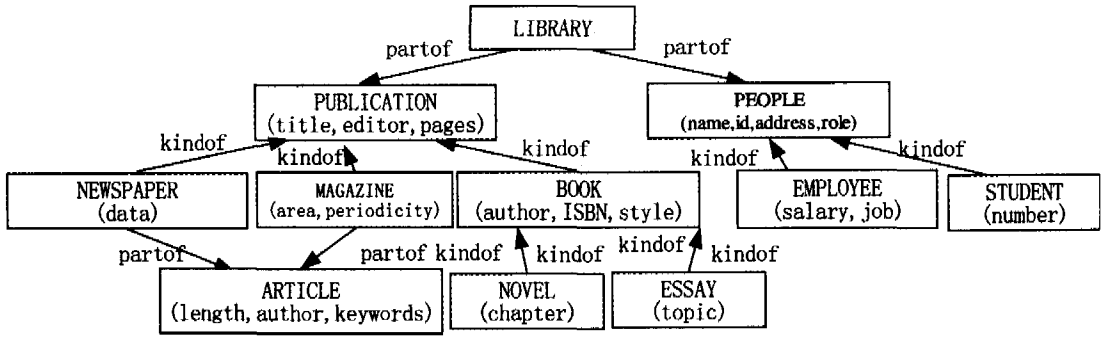

图 4 关系冲突解决后的领域本体

3 Kietz J, Maedche A, Volz R. A Method for Semi-automatic Ontology Acquisition from a Corporate Intranet. Workshop "Ontologies and text", co-located with EKAW, 2000
4 Suryanto H, Compton P. Discovery of Ontologies from Knowledge Bases. In: Proceedings of the First International Conference on Knowledge Capture, 2001. 171~178
5 Deitel A, Faron C, Dieng R. Learning ontologies from RDF annotations. Proceedings of the IJCAI Workshop in Ontology Learning, Seattle, 2001
6 Papatheodorou C, Vassiliou A, Simon B. Discovery of Ontologies for Learning Resources Using Word-based Clustering. ED-MEDIA, 2002