维普资讯 http://www.cqvip.com

Web服务器负荷状态检测方法研究*)

王 田[1,2] 杨士中[1]

(重庆大学通信工程学院 重庆400044)[1] (重庆工商大学计算机科学与信息工程学院 重庆400067)[2]

摘 要 本文对Web服务器的负荷状态检测方法进行了深入研究。在基于对系统负荷状态与内部资源关系的研究基础上,对Web服务器的负荷特性进行了分析,给出了负荷状态检测参数、检测模型及基于多参数数据融合的检测方法,并通过实验数据对其有效性进行了分析。

关键词 Web服务器,负荷检测,数据融合,负荷控制

A Study on the Approach of Load State Detection in Web Server

WANG Tian[1,2] YANG Shi-Zhong[1]

(Chongqing University, Chongqing 400044)[1] (Chongqing Technology and Business University, Chongqing 400067)[2]

Abstract Some thoroughly research on load state detection in Web server is introduced in this paper. Load sources and characteristics of Web server are analyzed in detail after investigating the relationship between inter resource and load state in the system. And load detection parameters and model are presented. Moreover, a novel strategy to detect system load state based on fusion of multi-detection data is presented. By some test, the theory and method are proved most feasible.

Keywords Web server, Load detection, Data fusion, Load control

由于Web应用的广泛性及其特有的B/S结构模式,给Web服务器带来了高负荷性。Web服务器是多数Internet应用和服务的核心,在下一代Internet中随着带宽瓶颈的突破,其在整个网络中的地位将更加突出,更多的流媒体、交互式游戏、智能服务、业务平台功能都将主要由Web服务器及其附属服务器来负责实现,要承受巨大的业务处理量。目前,随着用户量和业务量的迅速增长,服务器在某些特定时段内极易受到突发性的大业务流量冲击,使系统处于过负荷(overload)状态,若过负荷状态得不到及时控制和缓解将使其提供的应用QoS和系统性能迅速恶化,导致响应时延急剧增加、数据丢失甚至系统崩溃。有效的负荷状态检测是对Web服务器进行过负荷控制的基础[1,2]。

1. 负荷状态的形式化定义

负荷是系统瞬间运行状态的反映,与处理的业务量、负荷特性、任务调度策略、资源配置数量、资源分配策略有密切关系。

Web服务器中的服务资源主要包括两大类:信息资源和硬件资源。其中硬件资源是本文研究的主要对象,包括:处理机、内部存储器、外部磁盘阵列、内部总线、I/O资源、网络通道资源等几个方面。其中处理机是影响Web服务器业务性能的最主要因素。

为了分析的方便,下面先给出负荷的形式化定义和描述[3]。

定义1 若系统在有限的资源集R下处理对象$X(t)$的理论能力容限为$C_{max}|_{X(t)}$,在时刻t正在处理的业务量为$C(t)|_{X(t)}$,则系统在时刻t的负荷状态定义为:

$$L(t)=\frac{C(t)}{C_{max}}\Big|_{X(t)} \tag{1}$$

其中:$X(t)$称为负荷源,资源集R由配置的资源元素r_i构成,即:$R=[r_1,r_2,\cdots,r_n]$。

若系统在时刻t的资源消耗状态为$R_C(t)=[r_1(t),r_2(t),\cdots,r_m(t)]$,$m<n$,则系统在时刻$t$的负荷状态也可以表示为:

$$L(t)=f(\cdot)\left(\frac{r_1(t)}{r_1},\frac{r_2(t)}{r_2},\cdots,\frac{r_m(t)}{r_m}\right) \tag{2}$$

式中:$f(\cdot)$表示资源元素的消耗量与负荷状态之间存在的映射关系。

在系统设计时,一般根据处理能力的设计指标配置资源R,理论容限$C_{max}|_{X(t)}$为最大处理能力的设计指标X_{max}。按照可靠性理论设计时,最大实际处理能力$C'_{max}|_{X(t)}$一般大于理论容限。

定义2 给定系统的理论容限$C_{max}|_{X(t)}$对应的负荷阈值L_0,对于任意时刻t,若$L(t)>L_0$,则称当前状态为过负荷状态。在不同负荷控制阈值L_i下,系统负荷状态可以分为不同的级别。

当$L_i<L_0$时,可以用$X(t)|X_{max}$近似表示当前的负荷状态;当$L_i>L_0$时,系统性能处于不稳定状态,同时由于资源元素配置的差异以及资源元素消耗量的不均衡性,不能继续用$X(t)|X_{max}$近似表示当前的负荷状态。

系统过负荷与资源元素的消耗量有直接关系,并可能出现不同现象和后果。

(1)当业务量超过处理机的处理能力后,待处理的业务队列长度增加,平均等待时间增加,QoS下降,处理机出现性能不稳定;

(2)等待处理的业务数据超过缓存容量,则数据可能溢出,导致部分数据丢失;

(3)内部消息交换量过大导致总线阻塞,则平均处理时延增加;

*)高等院校博士点基金和中国四联仪器仪表集团博士后研究项目资助。王 田 博士后,研究方向:高性能网络技术,多媒体通信;杨士中 教授,博士生导师,中国工程院院士,研究方向:信息与通信系统。

维普资讯 http://www.cqvip.com

(4)频繁读写存放在数据库服务器中的数据导致I/O通道阻塞,则平均处理时延增加。

(5)大量连接会话得不到释放,新的连接请求建立的时延增加,拒绝率增加。

Web服务器中各类请求的特性存在着差异,导致系统过负荷的因素各不相同,各种因素之间是相互影响的,如总线阻塞或I/O通道阻塞必然导致平均处理时延增加,从而使待处理的业务数据积压而出现缓存溢出。系统的负荷状态与资源消耗量并不是线性关系,具有典型的非线性特征。

2. Web服务器的负荷特性

高精度的Internet业务流量监测数据表明[4,5]:Internet业务流量在较大时间尺度下呈现出统计自相似性,不能用传统流量理论中的泊松过程或马尔可夫模型描述,应该用Pareto等重尾分布来描述和分析。不论网络的拓扑结构、用户数量、请求类型如何变化,这种自相似性始终存在。业务流量的自相似性反映了用户行为在较大范围时间尺度上的统计相似性,流量的突发持续时间没有明确的长度,不能将它们平滑掉。当用户数量增加时,聚集业务的突发性变得更突出而不是减少。这种性质对Internet的流量控制及Web服务器中的排队分析有显著影响。

Internet上大量HTTP连接请求在Web服务器中主要产生两类负荷:静态负荷和动态负荷。静态负荷由对静态页面的处理产生,静态页面指那些不是由CGI脚本动态生成的文档,即后缀为HTML、HTM、SHTML、GIF、JPG、PS等的文档;动态负荷分为标准动态负荷和交互式动态负荷,标准动态负荷由对各种GET和CGI请求的处理产生,这里所指的CGI包括传统的CGI、ISAPI、ASP、NSAPI、JSP、Java Servlet等所有能动态产生HTML页面的脚本;交互式动态负荷涉及与数据库服务器的交互,增加了Web服务器对数据存取的额外处理负荷。

Crovella、Arlit、Williamson、Abdulla等学者通过对客户端Web请求踪迹、对不同层次服务器日志或代理服务器日志的分析研究,归纳出了一些共性的Web负荷特性[6~8]:

(1)文档类型分布。HTML页面和图像文件一起约占所有Web请求文件的90%,CGI(所有能产生HTML页面的脚本)约占2%并呈上升趋势。

(2)文档大小分布。Web服务器端文档大小变化很大,其分布具有重尾特性,基本符合参数$\alpha\approx1$的Pareto分布。

(3)文档引用行为。对同一文档的引用是独立的指数分布,极少数请求是对截然无关的文档的引用。文档引用次数的分布服从Zipf法则(即:如果将文档按引用次数p从高到低排序,则文档的排名r和引用次数p有如下关系:$p\approx kr^{-1}$,$k>0$)。

(4)传送文档大小分布。传送文档大小分布也具有重尾特性,符合参数$\alpha\approx1.06$的Pareto分布。文档大小的分布对传送文档大小的分布有很大的影响。

(5)用户"思考"时间。用户"思考"时间(OFF time)分为主动和非主动OFF time。主动OFF time为浏览器分析HTML页面及为该页面嵌入的对象(如图像)建立新的TCP连接所花费的时间,非主动OFF time为用户浏览页面时思考时间。

在标准性能评测协会(SPEC,Standard Performance Evaluation Corporation)开发的Web服务器性能基准测试软件SPECweb99中,对各种请求在模拟Web服务器工作负荷中所占比例作如下约定[9]:

表1 SPECweb99中各类请求比例

负荷类型	请求类型	所占比例
静态负荷	静态GET	70%
动态负荷	标准动态GET	12.45%
	标准动态GET(CGI)	0.15%
	Customized动态GET	12.6%
	动态POST	4.8%

可以看出静态负荷是Web服务器的主要负荷。随着客户和服务器间交互性的增强,CGI和Java技术使用的持续增长,动态负荷所占比重将增加。Java applet运行在浏览器上与Web服务器性能关系不大,而各种不同类型CGI应用和HTTP服务器+数据库服务器的模式趋于普遍(对数据库的访问也主要是通过CGI),Web服务器对CGI的处理负荷越来越大。

3. 负荷状态状态评价参数

3.1 负荷状态评价参数

从定义1中可知,系统的负荷与资源消耗及系统处理能力容限有关,资源的消耗量可以作为负荷评价的参数。曾经被研究及用作服务器负荷指标的主要包括:处理机资源占用率、内存使用率、用户数据区使用率、业务队列长度、某段时间内平均队列长度、内部总线占用率、I/O通道利用率等。

在Web服务器中,负荷状态可以用几个参数进行综合评价:CPU利用率、缓存使用率、请求平均等待时间、总线利用率、I/O通道利用率等。这些参数都是对系统的负荷状态的反映,每个参数与负荷状态存在着一定的映射关系[11]。这种映射关系并不具有精确性,而是一个区间映射。这种映射关系随着处理机调度策略、缓冲区管理策略、业务队列排队规则以及软硬件结构的差异而不同。

另外,在对Web服务器的性能评测中[9,10],通常使用每秒请求数RPS(requests per second),每秒字节数BPS(bytes per second)和往返时间RTT(round-trip time)3个指标进行性能评价。RPS定义为单位时间内完成的Web请求数,一个Web请求即对文档的一次请求,包含对文档嵌入式对象(如内嵌图像)的请求;BPS定义为单位时间内完成的请求传输的数据量(Byte),包含发送的请求、HTTP头及接收的所有数据;RTT定义为从客户开始建立与服务器的连接到客户收到关于本次请求的最后一个字节的时间差。因此,考虑负荷量与服务器性能的相应关系,可以把RPS和BPS作为负荷状态监测内容。用实时BPS与服务器最大输入输出字节数的比值作为评价参数;用实时连接数量与服务器最大允许接纳请求数的比值作为负荷状态评价参数。没有用RTT作为监测指标,主要因为RTT是基于客户端评价的,请求平均等待时间参数中已经考虑了服务器的响应时延。

3.2 参数灵敏度

作为负荷状态评价的参数(或指标),参数量的变化对负荷量的变化必须很敏感。为了描述系统负荷和评价参数之间的关系,下面引入参数灵敏度的概念。

定义3 对于任意时刻t,给定系统负荷量$C(t)$和负荷评价参数$k(t)$,则评价参数对系统负荷量变化的灵敏度(Sensitivity)S_i^C为:

$$S_i^C=\frac{\delta k(t)/k(t)}{\delta C(t,k)/C(t,k)}\approx\frac{\partial k(t)}{\partial C(t,k)}\cdot\frac{C(t,k)}{k(t)} \tag{3}$$

式中:$\delta C(t,k)$表示在时刻t,参数$k(t)$变化时$C(t)$的变化。

显然S_i^C越大,参数越能灵敏地反应负荷状态变化。作为负荷评价的参数应该选择S_i^C值较大,且受时间影响较小的参

维普资讯 http://www.cqvip.com

数。

4. 负荷状态检测模型和算法

对 Web 服务器负荷状态的准确评价和判断至关重要，误判断在过负荷控制中将导致严重的负效应。对 Web 服务器负荷状态的判断涉及多个参数，每个参数都是一个独立的信息源，根据单参数检测结果进行负荷状态评价有一定的局限性，具有较高的误报率和信息不完整性。可以根据多个参数进行信息融合综合判断服务器的负荷状态，获得更完整的信息，降低误报率。用数据融合的方法可以把来自多个检测参数的信息进行联合、相关、组合以获取精确的负荷状态估计和严重程度的完整评价，在可靠性和置信度上具有很大的优势。

我们采用了一个包含 n 个检测参数、n 个局域决策点和1个融合中心的分布式负荷状态检测模型。若各检测参数仅向融合中心传送独立的判决，该判决信息不能充分反映参数的信息量如过负荷严重程度、可信度等，信息损失较大，融合系统的性能有所下降。因而，可以采用冗余信息融合方法，即单个参数同时向融合中心传送其是否处于过负荷状态的判决、负荷状态级别及判决的可信度，融合中心根据各个参数的判决及其可信度形成最终的判决[3]。

若用 H_1 表示系统出现过负荷的概率，H_0 表示系统正常的概率；C_i 表示第 i 个参数的检测；u_i 表示其判决：$u_i=1$ 表示系统处于过负荷状态，$u_i=0$ 表示系统处于正常状态。K_i 表示负荷阈值。单参数的判决就是将其检验统计量 $L(C_i)$ 与负荷阈值 K_i 进行比较：①当 $L(C_i)>K_i$ 时，判决服务器处于过负荷状态；②当 $L(C_i)<K_i$ 时，判决服务器处于正常状态。

$L(C_i)$ 超过或低于负荷阈值 K_i 的程度实际上反映了负荷状态的级别和此参数判决的可信度。单参数判决的可信度可通过对 $L(C_i)$ 相对于 K_i 的偏差进行有限级别的量化来获得，量化级别由系统预定义的负荷级别决定。

当 $L(C_i)\geqslant K_i$ 时，为了反映 $L(C_i)$ 超过 K_i 的程度，需要对区间 $[K_i,\infty]$ 进行有限量化。若存在某一数值 A_i，$A_i\in[K_i,\infty]$，使得 $P(L(C_i)>A_i)\,|\,H_0)\approx 0$，则充分反映判决 $u_i=1$ 的可信度仅需对区间 $[K_i,A_i]$ 进行量化。

当 $L(C_i)<K_i$ 时，若存在某一数值 B_i，$B_i\in[-\infty,K_i]$，使得 $P(L(C_i)<B_i)\,|\,H_1)\approx 0$，则充分反映判决 $u_i=0$ 的可信度仅需对区间 $[B_i,K_i]$ 进行量化。

对于一个任意分布的检验统计量 L，根据契比雪夫不等式可以求出 A_i 和 B_i。

若单参数的判决可信度 T_i 用 N 个级别进行描述。当 $L(C_i)\geqslant K_i$ 时，令

$$K_{i,j}^{1}=K_i+j\frac{A_i-K_i}{N},\ j=0,1,2,\cdots,N-1 \tag{4}$$

记 $K_{i,j}^{1}=\infty$，则判决 $u_i=1$ 的可信度为

$$T_i=j,\ K_{i,j}^{1}\leqslant L(C_i)\leqslant K_{i,j+1}^{1} \tag{5}$$

当 $L(C_i)<K_i$ 时，令

$$K_{i,j}^{0}=K_i-j\frac{K_i-B_i}{N},\ j=0,1,2,\cdots,N-1 \tag{6}$$

记 $K_{i,j}^{0}=-\infty$，则判决 $u_i=0$ 的可信度为

$$T_i=j,\ K_{i,j+1}^{0}\leqslant L(C_i)\leqslant K_{i,j}^{0} \tag{7}$$

多参数负荷状态检测系统由 n 个参数构成，各参数的判决信息为 U_i：

$$U_i=\begin{bmatrix}u_i\\T_i\\O_i\end{bmatrix} \tag{8}$$

式中 u_i，T_i，O_i 分别表示负荷状态的判决、可信度和负荷级别，融合中心根据向量 $U_T=(U_1,U_2,\cdots,U_n)$ 进行假设检验形成最终的判决。

5. 测试分析

5.1 测试方法

我们对重庆网通信息港的 Web 服务器的业务流量和性能进行了长期监测（采集了从2001/11/24至2003/02/05的运行数据）。重点对服务器的 CPU 利用率、内存使用率、缓存使用率、总线利用率、网络通道利用率、每秒连接数、每秒字节数、请求平均等待时间等8个指标进行了监测。

监测数据的采集由三部分完成：利用服务器性能监测系统运行日志完成 CPU 利用率、内存使用率和缓存使用率3个参数的采集，按时间间隔0.5s 进行统计；网络监视器完成网络通道利用率和每秒字节数的采集，取0.2s 内平均值；另外，在任务调度模块上设计了监视进程驻留在服务器底层对总线利用率、每秒连接数和请求平均等待时间进行采集，取 Δt 内的平均值（Δt 控制在0.5s～1s，若 Δt 设置过小，虽然可以迅速地反映系统状态变化，但将增加服务器额外的负荷）。

Web 服务器负荷量用接纳的连接数量的加权平均予以表示，因为服务器对各类 HTTP 请求的处理时间及所消耗的资源量不相同，若以到达请求的绝对数量作为服务器负荷量予以分析是极不准确的。为此，我们在测试常见的各类 HTTP 请求平均处理时间后，采用针对不同的请求类型赋予不同的权值方式进行加权平均处理，然后再统计。

5.2 结果分析

图1给出根据采集到的数据综合处理后得到的 Web 服务器负荷量与资源占用率之间的关系。

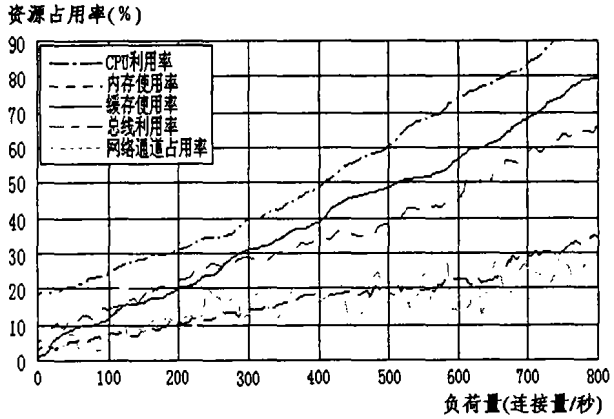

图1 服务器中负荷量与资源占用率的关系

从图中可以看出服务器的负荷状态可以用资源的占用率进行度量和评价。服务器资源消耗与接纳的请求量存在着直接的关系，负荷量增加时，资源占用率同步增长，反映了服务器负荷状态的实时变化。CPU 占用率、缓存使用率、总线利用率等参数具有很高的灵敏度，可以作为负荷状态实时检测参数，并能通过与系统内部拥塞程度之间的映射关系确定负荷的程度或级别。由于系统在初始配置和内存管理策略上的差异，内存利用率灵敏度较小，不宜作为负荷状态实时检测参数。由于 Web 服务器与各从属服务器间采用高速以太网络连接，内部消息通信所消耗的网络资源较少（占用率不到带宽的35%），服务器间通信不是性能的瓶颈，且动态变化范围极大，不宜直接作为负荷状态实时检测参数。

参 考 文 献

1 Rumsewicz M. Load Control and Load Sharing for Heterogeneous

维普资讯 http://www.cqvip.com

Distributed Systems [A]. In: Proc. of the Intl. Teletraffic Congress'99(ITC-16), Edinburgh, 1999
2 Maria K. Overload control strategies for distributed communication networks: [Ph. D. dissertation]. Sweden: Lund Institute of Technology, 1999
3 王田. SMSC 负荷检测方法研究. 电子科技大学学报[J], 2003, 2(4)
4 Paxson V. Growths trends in wide-area TCP connections [J]. IEEE Network, 1994, 8(4): 8～17
5 Crovella M, Bestavros A. Self-similarity in world wide web traffic evidence and possible causes[A]. In: SIGMETRICS'96, Marriott, PA, 1996
6 Cunha C, Bestavros A, Crovella M. Characteristics of WWW client-based traces [D]. Department of Computer Science of Boston University: [TechRep: TR-95-010]. 1995
7 Arlitt M, Williamson C. Web server workload characterization: The search for invariants (extended version) [A]. In: ACMSIGMETRICS'96, Marriott, PA, 1996
8 Abdulla G, Fox E, Arbams M. Shared user behavior on the world wide web. In: ProcWebNet97, Association for the Advancement of Computing in Education (AACE), 1997. http://www.aace.org/
9 Standard Performance Evaluation Corporation. SPECweb99 Benchmark User's Guide, 1999. http://www.spec.org/
10 欧灿辉, 李晓明. Web 服务器性能评测[J]. 计算机研究与发展, 2002, 39(5): 540～547
11 Ranganathan M, Acharya A, Saltz J. Distributed resource monitors for mobile objects[A]. In: Proc. of the Fifth Intl. Workshop on Object Orientation in Operating Systems, Seattle, WA, 1996: 19～23

(上接第12页)

输入：待加入常识概念 $c=\{w_1,w_2,\cdots,w_k\}$，心理常识概念库 $B=\{Attr,X,F,S\}$，其中，$Attr=\{A_1,A_2,\cdots,A_m\}$是心理常识属性集合，$X=\{x_1,x_2,\cdots,x_n\}$是心理常识基础概念集合，子类生成规则集合 $F=\{F_1,F_2,\cdots,F_p\}$，子类检查规则集合 $S=\{S_1,S_2,\cdots,S_q\}$；

输出：对 c 与 B 的冗余性和不一致性的判定；

Step1：读入 c 中的各个语义成分$\{w_1,w_2,\cdots,w_k\}$；在索引表中查找各语义成分对应的层次结构，设为$\{t_1,t_2,\cdots,t_k\}$；

Step2：查找 S 中子类检查规则 S_1是 S 的子集，使得 S_1包含所有的$\{t_1,t_2,\cdots,t_k\}$；

Step3：逐个检查 $c=\{w_1,w_2,\cdots,w_k\}$中各个语义成分是否严格遵守 S_1中子类检查规则；

Step4：若存在 $S_2\subseteq S_1$，使得 $c=\{w_1,w_2,\cdots,w_k\}$中各语义成分严格满足 S_2中任一子类检查规则，如果 S_2不为空，则 c 与心理常识概念库 B 一致；否则返回“不一致”；

Step5：检查 F 中的子类生成规则集合 F_1是 F 的子集，使得 F_1包含所有的$\{t_1,t_2,\cdots,t_k\}$，逐个检查 $c=\{w_1,w_2,\cdots,w_k\}$中各语义成分是否严格 F_1中子类生成规则；

Step6：若存在 $F_2\subseteq F_1$，使 $c=\{w_1,w_2,\cdots,w_k\}$中各语义成分严格满足 F_2中任一子类生成规则，如果 F_2为空，则根据语义成分的值，添加该概念到概念库合适位置，停止算法；否则，返回“有冗余”；

算法2的时间复杂性远低于算法1。由于在初始化之前建立了心理常识基础概念的索引，设所有心理常识基础概念的数量是 n，则算法4.1按折半查找的复杂性是 $n\log_2 n$。由于心理常识基础概念对应的结构或常识属性已在每一心理常识基础概念后给出了链接，因而查找的时间复杂性为常量级。找到 F_1后检查待检查概念的语义成分与 F_1中〈子类检查规则〉的匹配性的时间复杂性为语义成分的数量 m(通常 m 小于30)和〈子类生成规则〉数量 k(子类生成规则的数量也为有限值)的乘积。则算法2的复杂性为 $k\times m\times n\times\log_2 n$。由于 m 和 k 的数量都是常数级别的，所以，算法2的时间复杂性为 $|n\log_2 n|$。

结束语 本文讨论了与人的心理相关的常识概念的获取、组织和分析。目的在于克服现有概念模型中缺乏自动获取概念的方法，并且以心理学范畴为基础，在手工获取心理常识基础概念和心理常识属性的基础上，给出自动生成心理常识复合概念以及组织它们的算法，进一步，对于新加入概念，提供算法考察其与现有概念库的冗余问题和一致性问题，本文的方法为以心理常识概念为基础的心理常识库的建构提供了良好的概念基础。

同时，由于心理常识复合概念的生成依赖于〈子类生成规则〉，所以如何正确书写、组织和规范〈子类生成规则〉是心理常识概念获取的关键，本文提供了一些基准原则，但是，自然语言非常丰富，我们仍然需要对复合概念的语义成分的构词进一步分析，并在此基础上加强与语言学家的合作，构造出更为精确的构造复合概念的规则。

参考文献

1 Guha R V, Lenat D. Cyc: A midterm Report. AI magazine, 1990. 33～59
2 Lenat D. The Dimensions of Context-Space. http://www.cyc.com, Oct. 1998
3 Churchland P M. Folk Psychology and the Explanation of Human Behavior. In: Churchland PM, ed. A Neurocomputational Perspective, MIT Press, 1989. 111～127
4 Stich S, Ravencroft I. What is Folk Psychology?. in Cognition, 1994. 50: 447～468
5 Ravenscroft I. Folk Psychology as a Theory, in Stanford Encyclopaedia of Philosophy, 1997
6 CYCorp, Feature of CycL. http://www.cyc.com, 1998
7 陆汝钤, 姬广峰. 基于常识的研究, 世纪之交的知识工程与知识科学. 清华大学出版社, 2001. 469～505
8 Richardson S D, Dolan W B, Vanderwende L. MindNet: Acquiring and Structuring Semantic Information from Text. In: Proc. of COLING '98, 1998
9 Vanderwende L. Ambiguity in the acquisition of lexical information. In: Proceedings of the AAAI 1995 Spring Symposium Series, working notes of the symposium on representation and acquisition of lexical knowledge, 1995. 174～179
10 Fellbaum C. WordNet: an electronic lexical database. The MIT Press, 1998
11 Baker C F, Fillmore C J, Lowe J B. The Berkeley FrameNet Project. In: Proc. of the COLING-ACL, Montreal, Canada, 1998
12 梅家驹, 竺一鸣, 等. 同义词词林. 上海辞书出版社, 1983
13 董振东. 语义关系的表达和知识系统的建造. 语言文字应用, 1998(3): 76～82
14 董振东, 董强. 知网介绍. http://www.keenage.com, 1999
15 Osherson D N, Smith E E. On the Adequacy of Prototype Theory as a Theory of Concepts. in Cognition, 1981, 9: 35～58
16 Cohen B. Models of Concept. Cognitive Science, 1984, 8: 27～58
17 Palmer S E. Fundamental Aspects of Cognitive Representation. In: E. Rosch, B. L. Lloyd, ed. Cognition and categorization, 1978. 259～302
18 薄冰. 高级英语语法. 世界知识出版社, 2000
19 Correl S F, Porter B W. Iterative Refinement of Knowledge Bases with Consistency Guarantees, AAAI 1997 Spring Symposium on Ontological Engineering, 1997