

# 一种基于双层环网络的 Web 服务发现方法

周玉姣<sup>1</sup>, 钟 诚<sup>1</sup>, 李 智<sup>1,2</sup>, 黄毅然<sup>1</sup>

(1. 广西大学 计算机与电子信息学院, 广西 南宁 530004;

2. 广西科技信息网络中心, 广西 南宁 530012)

**摘 要:**结合 One-hop 和 Chord 路由机制, 考虑实际网络中节点能力的差异, 构造一种双层环路由结构, 设计相应的区间查询定位和消息广播算法, 提出一种基于本体聚类的双层环 P2P 网络的 Web 服务发现方法, 实现了服务注册节点分类和查询请求迅速准确的定位。模拟实验结果表明, 该方法具有较高搜索效率和较短的响应时间, 能够显著地提高查询性能。

**关键词:**Web 服务; 服务发现; P2P; 路由

**中图分类号:**TP393

**文献标识码:**A

**文章编号:**1673-629X(2008)11-0109-04

## A Web Service Discovery Method Based on Ontology Clustering in Double-Loop P2P Networks

ZHOU Yu-jiao<sup>1</sup>, ZHONG Cheng<sup>1</sup>, LI Zhi<sup>1,2</sup>, HUANG Yi-ran<sup>1</sup>

(1. School of Computer and Electronics and Information, Guangxi University, Nanning 530004, China;

2. Guangxi Science and Technology Information Network Center, Nanning 530012, China)

**Abstract:** In order to achieve the classification of web services register nodes and exact location of query request, a double-loop routing structure is constructed, the interval query locating and message broadcasting algorithms are designed, and a web service discovery method based on ontology clustering in double-loop P2P networks is presented by combining the lookup protocols of One-hop with Chord and taking into account the ability differences of nodes in the network systems. The experimental results show that this presented method obtains good efficiency and short response time and improve the search performance.

**Key words:** web service; service discovery; P2P; routing

## 0 引 言

服务发现是 Web 服务的一项关键技术。采用集中式架构 UDDI 注册模型的传统 Web 服务, 难以满足 Web 服务数目的迅速增长和实时性的需求, 且其服务描述缺乏语义信息, 无法根据服务的功能进行查找和匹配, 只能局限于关键字匹配, 从而影响 Web 服务发现的效率及准确性。

P2P 网络模型具有对等性、分散性、独立性, 具有客户端、服务器的角色, 其与 Web 服务的结合有效地解决传统 Web 服务的瓶颈和单点失效的问题<sup>[1,2]</sup>。

P2P 系统性能的主要决定因素是系统的网络拓扑和路由协议。分布式哈希表(DHT)具有扩展性和容错性, 采用分布式哈希表技术进行资源组织、管理与服务发现成为结构化 P2P 网络(如 Chord、CAN、Pastry、Tapestry 系统)的主流方法。

文中在研究 P2P 环境和基于语义的 Web 服务发现基础上, 以获得高性能的 Web 服务发现为目标, 引入超级节点, 结合 One-hop 路由和 Chord 路由机制, 构造了一种双层环路由结构, 并设计相应的区间查询定位和消息广播算法, 提出一种基于本体聚类的双层环 P2P 网络 Web 服务发现方法。

该方法利用基于双层环路由结构的 P2P 网络组织 Web 服务的注册节点, 考虑实际网络中节点能力的差异, 充分利用 P2P 网络中搜索和定位节点的动态机制, 用服务领域本体概念映射区间对注册节点按其提供服务的类型进行聚类, 使特定领域的 Web 服务的注册与发现集中于双层环结构中外环少量的连续节点上, 以显著提高查询效率。

收稿日期: 2008-03-10

基金项目: 广西自然科学基金(桂科基 0575014); 广西教育厅科研项目(200708LX360); 广西研究生创新教育计划(2006105930812M29)

作者简介: 周玉姣(1979-), 女, 硕士研究生, CCF 学生会员, 主要研究网络与并行分布计算; 钟 诚, 博士, 教授, CCF 高级会员, 主要研究网络与并行分布计算; 李 智, 硕士, 研究员, 主要研究网络与分布计算。

## 1 双层环路由结构

在由  $N$  个节点组成的单环结构的 Chord 系统<sup>[3,4]</sup>中,每个节点只需维护其他  $O(\log N)$  个节点的信息,每次查找也只需  $O(\log N)$  条信息。也就是说,Chord 系统可以保证在  $O(\log N)$  跳数内定位对象,其跳数是逻辑跳数。

Chord 系统采用均匀散列函数为每个节点和数据对象分配  $m$  位的全局惟一标识符 ID,所有节点按节点 ID 从小到大顺时针排列在一个大小为  $2^m$  的环上,数据对象  $k$  ( $k$  是数据对象 ID) 被分配到环上顺时针方向紧随  $k$  (包括与  $k$  相等) 的第一个节点,该节点也称为对象  $k$  的后继,记做  $\text{successor}(k)$ 。每个 Chord 节点需维护一个前驱节点指针 (predecessor) 和一个后继节点 (successor) 指针及一个有  $m$  个表项的称为指针表 (finger 表) 的路由表。节点  $n$  的 finger 表中第  $i$  项指向节点  $s$ ,  $s = \text{successor}(n + 2^{i-1})$ ,  $1 \leq i \leq m$ , 因此,节点  $s$  是环上顺时针方向到节点  $n$  的距离至少为  $2^{i-1}$  的第一个节点,记为  $n.\text{finger}[i]$ , 而  $n.\text{finger}[1]$  则是节点  $n$  的后继节点  $n.\text{successor}$ 。finger 表中每一项既包含相关节点标识符,也包含该节点的 IP 地址和端口号。

Chord 系统对节点加入、退出 (失效) 频繁的网络能以相对较小的代价得到高的查询效率,但是对节点状态相对稳定的网络来说 Chord 系统的查询代价过大。通过引入超级节点,构造一种双层环路由算法来自动适应不同的网络波动以提高查询效率。此外,由于 Chord 系统没有考虑实际网络中节点传输带宽、存储容量和对等节点的计算能力及在线时间,导致网络中相对稳定且能力较强的节点 (称为超级节点) 未能有效发挥作用。另一方面,每个节点维护有限的成员关系信息,它虽然不必维护所有节点的路由就能有效地解决节点加入、退出频繁时节点路由表维护代价过大的问题,但它是以增加延时为代价的。考虑实际网络中节点能力的差异,超级节点具有更高的可靠性和更低的延时,如果让超级节点分担较多的负担,采用一定的策略,在节点加入、退出频繁程度不断变化的网络下,既能保持查找率又能使延时尽量少。

文中采取结合 One-hop 路由与 Chord 路由的方法,引入超级节点来提高路由效率,构造一种双层环路由结构:

(1) 双层环的拓扑结构:以  $2^m$  为模,采用均匀散列函数为每个节点分配  $m$  位的全局惟一标识符 ID,所有节点按节点 ID 从小到大顺时针排列在一个大小为  $2^m$  的环 (称为外环) 上,将外环上所有超级节点相连接形成一个内环,如图 1 所示。

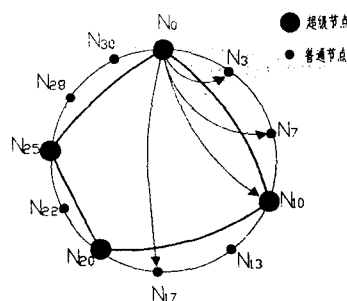


图 1 以  $2^5$  为模的双层环网络拓扑

(2) 双层环路由表的构造:将节点分为普通节点和超级节点。普通节点在路由信息的处理上,继承了 Chord 系统对路由信息的划分,保留了 Chord 路由表的全部信息,只对路由表增加一项 supernode 指针,用来保存所属超级节点的信息 (节点 ID, IP 地址)。超级节点除了要构建普通节点的路由表外,还要维护超级节点表 super 和成员表 member,超级节点表 super 用来保存所有超级节点的信息,成员表 member 用来保存当前超级节点与它的前一相邻超级节点间的所有普通节点信息,如表 1 所示。

表 1 节点的路由表示例

N <sub>0</sub> 的指针表		N <sub>4</sub> 的指针表		N <sub>8</sub> 的超级节点表		N <sub>8</sub> 的成员表	
start	node	start	node	ID	IP address	ID	IP address
N <sub>0</sub> +1	N <sub>1</sub>	N <sub>4</sub> +1	N <sub>5</sub>	N <sub>0</sub>	IP <sub>0</sub>	N <sub>1</sub>	IP <sub>1</sub>
N <sub>0</sub> +2	N <sub>3</sub>	N <sub>4</sub> +2	N <sub>7</sub>	N <sub>4</sub>	IP <sub>4</sub>	N <sub>3</sub>	IP <sub>3</sub>
N <sub>0</sub> +4	N <sub>7</sub>	N <sub>4</sub> +4	N <sub>11</sub>	N <sub>8</sub>	IP <sub>8</sub>	N <sub>7</sub>	IP <sub>7</sub>
N <sub>0</sub> +8	N <sub>15</sub>	N <sub>4</sub> +8	N <sub>15</sub>	N <sub>12</sub>	IP <sub>12</sub>	N <sub>11</sub>	IP <sub>11</sub>
N <sub>0</sub> +16	N <sub>1</sub>	N <sub>4</sub> +16	N <sub>5</sub>	N <sub>16</sub>	IP <sub>16</sub>	N <sub>15</sub>	IP <sub>15</sub>

(3) 双层环路由的查找策略:节点  $P$  得到一个查询请求  $Q(\text{key})$  ( $\text{key}$  是对象 ID) 时,若在它的 finger 表中未能找到相应节点,则将查询请求  $Q$  转发给它所在的超级节点  $N_s$ 。N<sub>s</sub> 将在其 finger 和 member 表中查询,若超级节点  $N_s$  找到相应目标节点,则将查询结果返回;否则,它将查询请求转发给 super 表中大于  $\text{key}$  且距  $\text{key}$  最近的那个超级节点  $N_m$ 。若超级节点  $N_m$  查询失败,则按照 Chord 系统基本路由协议进行查询。

## 2 基于本体聚类的双层环 P2P 网络 Web 服务发现方法

文中给出的 Web 服务发现方法的设计思想是:让双层环网络中每个注册节点仅负责某个服务类别的 Web 服务注册与发现,将提供同类型服务的注册节点进行聚类,使提供相同类型服务的注册节点邻接相连分布在双层环外环某段弧上,在进行 Web 服务发现时,根据所需服务的类型在相应弧上的注册节点中进行查询。

引入本体来描述服务领域<sup>[5]</sup>(本体是用于描述或表达某一领域知识的一组概念或术语,它给出了领域实体概念及相互关系、领域活动以及该领域所具有的特性和规律的一种形式化描述,为语义 Web 提供了相关领域的共同理解和统一的语义规范)。本体中每个概念描述一个服务领域,表示一类实例的集合,它的子类概念分别表示某个具体的服务概念,是它所涵盖实例集合的子集。将服务领域本体的所有概念映射到数轴上,每个概念被映射到一个整数区间,它的子类概念被映射到它的一个子区间。同一个概念的所有子类概念对应的区间是互不相交的且它们的并集等于这个概念对应的区间。注册节点的连接采用模为  $2^m$  的双层环结构实现,所有概念对应的区间的并集等于  $[0, 2^m - 1]$ 。某个概念  $C$  对应的区间  $T$  为  $[a, b]$ ,称  $T$  为概念  $C$  的目标节点区间。若节点  $n$  满足  $a \leq n \leq b$ ,则称节点  $n$  是概念  $C$  的一个注册节点,概念  $C$  的所有注册节点共同提供该概念对应领域的服务发布和检索。如图 2 所示,概念“Sell”映射到整数区间  $[19, 24]$ ,该分类区间的服务发布和检索由节点  $N_{20}$  和  $N_{22}$  提供。

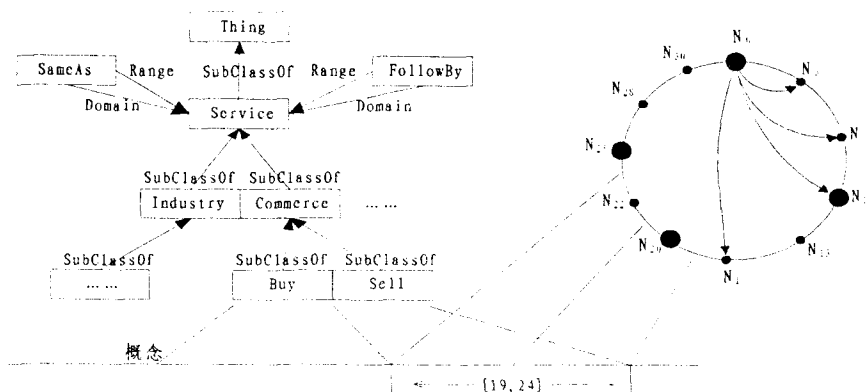


图 2 领域概念对应区间与双层环结构的外环映射示意图

## 2.1 查询定位算法

对任意节点提交一个查询定位请求  $Q(T)$ ,  $T$  是服务请求  $Q$  对应的目标节点区间,将  $Q(T)$  快速转发定位到节点 ID 属于  $T$  且节点 ID 最小的目标注册节点上。目标节点区间  $T$  为  $[a, b]$ ,任意节点  $P$  对服务请求查询定位操作为  $\text{Location}(P, Q(T))$ ,其转发规则如下:

(1) 若节点  $P$  是普通节点,则当节点  $P$  接收到查询定位操作时,分下面三种情况处理:

① 若  $P = a$ ,则将  $P$  作为结果返回,查询定位操作结束。

② 若  $P < a$ ,且  $P.\text{successor} \geq a$ ,则将  $P.\text{successor}$  作为结果返回,查询定位操作结束。

③ 若  $P > a$ ,向  $P.\text{supernode}$  转发查询定位操作

$\text{Location}(P, Q(T))$ ,如果超级节点查询失败,则  $P$  查询  $\text{finger}$  表,向  $\text{finger}$  表中小于  $a$  且距离  $a$  最近的节点转发查询定位操作  $\text{Location}(P, Q(T))$ 。

(2) 若  $S$  为超级节点,则当节点  $S$  接收到查询定位操作时,分下面两种情况处理:

① 若  $S = a$ ,则将  $S$  作为结果返回,查询定位操作结束。

② 若  $S \neq a$ ,则首先判断  $a$  是否是  $S$  所属成员,若是则查找  $\text{member}$  表,找出第一个大于等于  $a$  的节点作为结果返回,查询定位操作结束;若不是则查找  $\text{super}$  表,找出第一个大于等于  $a$  的超级节点  $S'$ ,向  $S'$  转发查询定位请求。如果超级节点  $S'$  查询失败,则查找  $\text{finger}$  表,向  $\text{finger}$  表中小于  $a$  且距离  $a$  最近的节点转发查询定位操作  $\text{Location}(S', Q(T))$ 。

## 2.2 区间消息广播算法

从注册节点所在目标节点区间中最小 ID 的节点出发沿双层环结构的外环顺时针方向给目标节点区间中其他所有节点转发消息的方法称为区间消息广播算法。设一个目标节点区间  $T$  为  $[a, b]$ ,  $P$  为区间内最小节点,区间消息广播算法的规则如下:

(1) 消息头部包含信息源节点  $S$  的节点 ID。

(2) 节点  $P$  接收到消息后,遍历  $\text{finger}$  表,若  $a \leq P.\text{finger}[i] \leq b$ ,则将  $P.\text{finger}[i]$  放入转发候选节点集  $M$  中,  $1 \leq i \leq m$ 。

(3)  $P$  向转发候选节点集  $M$  中的所有节点转发消息。设候选节点集  $M = \{M_1, M_2, \dots, M_k\}$ ,当  $1 \leq i \leq k-1$  时,  $P$  向  $M_i$  转发查询  $Q[M_i, \min(M_{i+1}, b)]$ ;当  $i = k$  时,  $P$  向  $M_k$  转发查询  $Q[M_k, b]$ 。

(4) 若区间  $T = [a, b]$  内某个节点  $R$  接收到某个查询  $Q[a_1, b_1]$  ( $a \leq a_1, b_1 \leq b$ ) 后遍历  $\text{finger}$  表所得候选节点集  $M$  为空,则不再继续往后继节点转发查询。

图 3 所示的是区间  $[0, 23]$  内消息广播的过程,整个消息在分类区间节点中的传播过程呈树状结构。

## 2.3 Web 服务发布与发现算法

节点进行服务查询时,必须提供所需服务的类型信息。当一个服务查询请求  $Q$  提交到任意注册节点  $P$  时,节点  $P$  首先将查询请求类型信息映射到对应的服务领域本体的概念,取得相应的目标节点区间  $T$ ,发出查询定位操作  $\text{Location}(P, Q(T))$ ,使用查询区间定

位算法将查询请求转发给  $T$  中节点最小的注册节点  $U$ , 节点  $U$  再通过区间消息广播算法, 将查询请求转发到  $T$  中的所有节点。所有接收到查询请求的节点, 在本地进行服务匹配并将结果沿查询请求转发路径返回给  $P$ 。

服务的发布过程与发现过程类似, 目标注册节点区间中的所有节点接到发布请求后将自身负载情况返回给请求节点, 请求节点根据返回的信息选择负载最小的节点进行服务发布。

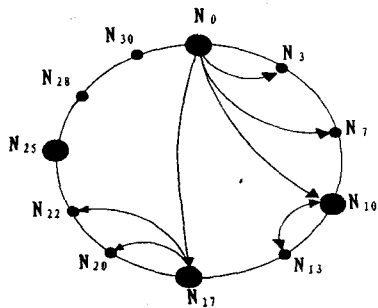


图3 区间[0,23]内消息广播的过程

### 3 模拟实验

采用 JAVA 语言编程模拟实现基于本体聚类的双层环 P2P 网络 Web 服务发现算法。从给出的查询定位算法可知, 在目标节点区间中查询转发的次数与区间的节点数相同且等于向分类区间中所有节点广播的最小转发次数。因此, 从发起节点定位到目标节点区间的查询转发次数成为衡量 Web 服务发现方法服务效率的一个重要方面。

为了评估将每个查询转发到目标节点区间的效率, 文中分别测试和统计了在不同的网络节点数目和目标节点区间数目下从任意节点发起对任意目标节点区间的查询定位操作的查询转发平均次数。图4所示的是模为  $2^{32}$ , 超级节点数为 64, 网络节点总数分别为 6000、8000 和 10000 的情形下, 当目标节点区间(服务领域本体的概念数)分别为 32、64、128、256 和 512 时, 每 10000 次随机查询定位操作的消息转发次数的平均值。实验结果显示, 当服务领域本体中的概念数相同时, 不同节点总数下每次查询定位转发的平均跳数基本相同。在网络节点总数固定时, 每次查询定位操作的查询转发平均次数只有很小幅度的增长, 这种增长速度远远小于目标节点区间数增长的速度且其随概念数总数的增加呈现减缓趋势。

另一个实验是对于节点总数为 4000、超级节点数为 64 的网络, 将失效节点全部设置分布在超级节点

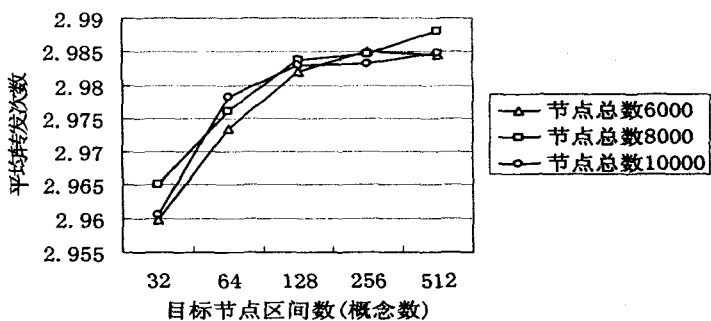


图4 节点数和概念数对查询定位操作的影响

中, 以测试查询定位操作受超级节点依赖性影响的情况。

图5的实验结果表明: 随着失效节点数的增大, 每次查询定位操作的查询转发平均次数明显增大, 超级节点对查询定位操作影响大。当超级节点失效时, 文中的服务发现机制将以增加延时来保证查询定位的效率。实验结果显示, 在最坏情形下, 当 64 个超级节点全部失效时, 每次查询定位操作的查询转发平均次数接近 7, 比全部超级节点有效时查询转发平均次数高出 1.3 倍。超级节点通常是比较稳定的, 因此, 应用超级节点能显著提高查询效率。

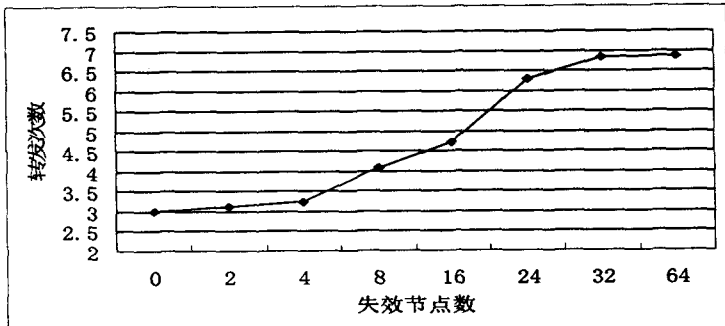


图5 超级节点对查询定位的影响

### 4 结束语

提出一种基于本体聚类的双层环 P2P 网络 Web 服务发现方法, 它利用基于双层环路由结构 P2P 网络组织 Web 服务的注册节点, 考虑实际网络中节点能力的差异, 充分利用了双层环 P2P 网络搜索和定位节点的动态机制, 采用服务领域本体对注册节点进行聚类, 使特定领域的 Web 服务的注册与发现集中于双层环结构外环少量的连续节点上, 从而减少了信息的转发, 显著提高查询效率。实验结果表明, 文中提出的方法可以应用于注册节点数目很大的情况, 并且在网络规模增大时仍能保持较好的性能。

(下转第 116 页)

度, Accuracy) 和学习能力(即无错误地识别任意样本的能力)之间寻求最佳折中, 以期获得最好的推广能力(Generalization Ability)。支持向量机方法的主要优点有: 可以解决小样本情况下的机器学习问题, 提高泛化性能, 解决高维问题, 解决非线性问题, 避免神经网络结构选择和局部极小点问题。

### 3.1 实验数据与结果

采用数据库中的 Ionosphere, Wine, Iris, Mushroom 这五组数据来做试验。其中 Ionosphere 数据库以其前 200 个样本作为训练样本, 后 150 个样本作为测试样本。其他三个数据都是采用 10-fold 交叉算法(即将样本平均分为 10 份, 其中 9 份作为训练样本, 1 份作为测试样本, 重复 10 次, 测试结果取 10 次的平均值)。所用数据信息见表 1。

表 1 实验数据表

数据库	提供者	维数	类数
Mushroom	Jeff Schlimmer	21	2
Ionosphere	Vince Sigillito	34	2
Wine	Stefan Aeberhard	13	3
Iris	R. A. Fisher, Michael Marshall	4	3

### 3.2 实验结果及分析

实验结果见表 2。

实验结果表明改进算法对 Ionosphere 和 Mushroom 的效果非常好, 提高了识别率: 平均识别率分别从原来的 93.264% 和 96.347% 提高到了 99.338% 和 99.176%; 而 SVM 的精度只有 87.778% 和 96.273%, 缩减了测试时间: 分别从原来的 0.0047s 和 0.043586s 减少到 0.003625s 和 0.0046875s, 而且减少了覆盖数: 从原来的 82.2 个和 69.406 个减少到 47 个和 26 个。对其他的几个数据也减少了覆盖个数, 识别率也有一定的提高, 并且结果非常稳定。由此可见, 学习顺序确实对实验结果有重大的影响。对原覆盖算法采用先聚类, 再求覆盖的修改方法取得非常好的效果。

## 4 结束语

覆盖算法已被广泛应用于模式识别、信号处理、模

糊控制、金融预测等方面, 并取得一定的成就。改进之后的基于聚类的覆盖算法以其独特的稳定性将在这些领域中得到更大更广泛的应用。

表 2 实验结果

		改进算法	一般算法	SVM	
Ionosphere	覆盖数	47	82.2	精度(%)	87.778
	拒识数	5.1	5.8		
	测试时间(s)	0.003625	0.0047		
	最大识别率(%)	99.338	95	测试时间	0.0010014
	平均识别率(%)	99.338	93.264		
Wine	覆盖数	18.4	23.80	精度(%)	97.778
	拒识数	1.8	1.8		
	测试时间(s)	0.003125	0.0016		
	最大识别率(%)	97.222	96.1111	测试时间	0.0010014
	平均识别率(%)	97.222	94.0444		
Iris	覆盖数	11.8	16.8	精度(%)	96.667
	拒识数	0.9	0.8		
	测试时间(s)	0.001563	0.0031		
	最大识别率(%)	98	96.6667	测试时间	0
	平均识别率(%)	96.02	94.56		
Mushroom	识别率(%)	92.4	91.9	精度(%)	96.273
	覆盖数	26	69.406		
	拒识数	24.7	41.831		
	测试时间(s)	0.0046875	0.043586	测试时间	0.2734
	识别率(%)	99.176	96.347		

### 参考文献:

- [1] 张燕平, 张 铃, 吴 涛. A Geometrical Representation of McCulloch - Pitts Neural Model and Its Applications[J]. IEEE Trans. on Neural Networks, 1999, 10(4): 925 - 929.
- [2] 张 铃, 张 钺. M - P 神经元模型的几何意义及其应用[J]. 软件学报, 1998, 9(5): 334 - 338.
- [3] 张 铃, 张 钺, 殷海风. 多层前向网络的交叉覆盖算法[J]. 软件学报, 1999, 10(7): 737 - 742.
- [4] Han Jiawei, Kamber M. 数据挖掘概念与技术[M]. 范 明等译. 北京: 机械工业出版社, 2001: 231 - 235.
- [5] Vapnik V N. 统计学习理论的本质[M]. 张学工译. 北京: 清华大学出版社, 2000.

(上接第 112 页)

### 参考文献:

- [1] 胡建强, 郭长国, 王 怀, 等. 一种基于 P2P 网络的服务发现方法[J]. 电子学报, 2005, 33(12A): 2503 - 2507.
- [2] Schmidt C, Parashar M. A peer - to - peer approach to Web service discovery[J]. World Wide Web, 2004, 7(2): 211 - 229.
- [3] Stoica I, Morris R, Liben - Nowell D, et al. Chord: A scalable peer - to - peer lookup service for internet applications [C]//

Proc. of ACM SIGCOMM. New York: ACM Press, 2001: 149 - 160.

- [4] El - Ansary S, Alima L O, Brand P, et al. Efficient Broadcast in structured P2P Networks[C]//Proc. of 2nd International Workshop on Peer - to - Peer Systems, vol. 2735. Berlin: Springer - Verlag, 2003: 304 - 314.
- [5] 刘志忠, 王怀民, 周 斌. 一种双层 P2P 结构的语义服务发现模型[J]. 软件学报, 2007, 18(8): 1922 - 1931.