维普资讯 http://www.cqvip.com

基于实体分类的数据库模式匹配方法*)

于 波 唐世渭 张 鹏 童云海
(北京大学信息科学技术学院 北京100871)

摘 要 模式匹配在诸如数据集成、数据仓库、信息共享和计算机网络交换等许多应用领域起到关键作用。目前,自动模式匹配方法也不能解决复杂模式情况下的匹配问题。本文提出一种基于关系模式领域中实体分类的数据库模式匹配方法。该方法通过朴素贝叶斯学习将实体分为不同的类(子模式),然后以同样的类来匹配子模式之间的模式元素。本方法在复杂模式条件下可有效提高匹配效率,减少匹配工作量,节省人力资源。

关键词 模式匹配,实体,子模式,朴素贝叶斯学习,数据仓库

An Approach to Database Schema Matching Based on Entity Classification

YU Bo TANG Shi-Wei ZHANG Peng TONG Yun-Hai
(School of Electronics Engineering and Computer Science, Peking University, Beijing 100871)

Abstract Schema matching plays a key role in many application domains, such as data integration, data warehouse, and information share and exchange on computer network. Currently, approaches of automatic schema matching cannot solve matching issue under the circumstance of complex schema well. This paper introduces an approach based on entity classification in the domain of relation schema. It divides entities into different categories (sub-schema) using Na'ive Bayes Learning, and then matches schema elements between the sub-schemas with the same category. It can effectively improve matching results, reduce the number of element-to-element comparisons and save user efforts under the circumstance of complex schema.

Keywords Schema matching, Entity, Sub-schema, Na'ive bayes learning, Data warehouse

1 引言

模式是通过某些数据结构连接起来的数据成员集合,是数据成员的逻辑级视图。典型的模式有:关系模式、XML DTD、ontology、面向对象模型和 ER 模型等。模式匹配是指给定两个模式,根据可利用的信息,发现语义对应的模式成员之间正确的映射关系的过程[1~3],如"模式 T 中的成员 credit_limit_amount(信用额度)与模式 S 中的成员 credit_range 存在映射关系为 credit_limit_amount =credit_range"、"模式 T 中的成员 list_price 与模式 S 中的成员 price 和 tax_rate 存在映射关系 list_price =price * (1+ tax_rate)"。近些年,随着数据仓库、数据迁移、网络信息交换与共享以及语义 Web 等领域的信息集成需求的增长,模式匹配逐渐成为人们关注的焦点。

目前,模式匹配一般通过手工完成。手工匹配是一个枯燥、费时、容易出错且成本较高的过程。例如,国内某银行数据仓库建设中的模式匹配工作,共15个源系统数据库,30,000多个属性,在与数据仓库逻辑数据模型的1,000多个属性的匹配中,花费了360多个人日,且没有做深入的正确性检查。因此,设计一种自动、全面、高效的匹配方法是十分必要的。当前,已有不少自动模式匹配的研究成果,但多关注于模式比较简单的情况,如3~5个实体,每个实体有5个左右的属性。在这种情况下,匹配效率不是主要问题,匹配质量也不错。但实际应用中,模式都比较复杂,如前述案例中,任何一个数据源系统都有几十个实体和成百上千个属性,使用目前的方法,匹配效率较低,匹配质量也不高。本文以关系模式为例,介绍一种基于朴素贝叶斯学习进行实体分类,在同类子模式之间进行匹配的方法。在模式比较复杂的情况下,它能有效地提高匹配效率,减少匹配工作量,改善匹配质量。由于不同模式之间表示概念、语义的相似性,该方法稍加修改,也可用于其它模式。

本文的结构为:第2部分介绍模式匹配的相关研究情况,并进行问题分析;第3部分介绍基于实体分类的数据库模式匹配方法,包括基于朴素贝叶斯学习的实体分类方法和基于规则的模式匹配方法;第4部分给出有关实验结果和分析;最后进行总结和研究展望。

2 相关研究与问题分析

2.1 相关研究

自动模式匹配是用户参与下的自动(或半自动)地发现匹配关系的过程,一般包括匹配前的预处理、匹配操作、匹配维护等步骤。当前研究主要关注于匹配操作中的匹配方法和实现方式。

文[1]中对目前的匹配方法进行了综述,将其分为两大类:模式级(schema-based)匹配和实例级(instance/content-

*)本课题得到国家"973"重点基础研究发展规划项目(G1999032705)和国家"十五"科技攻关计划(2001BA102A01)资助。于 波 博士研究生,主要研究方向为数据仓库、数据挖掘、数据集成等。张 鹏 博士研究生,主要研究方向为数据仓库、数据挖掘、联机分析处理等。唐世渭 教授,主要研究方向数据仓库、数据挖掘、Web 信息集成、数字图书馆、嵌入式数据库等。童云海 博士,讲师,主要研究方向为数据仓库、数据挖掘、空间数据库等。

维普资讯 http://www.cqvip.com

based)匹配。模式级匹配主要利用名称、描述、数据类型、约束等模式级信息的相似性进行模式成员匹配。实例级匹配主要利用数据的统计特征(如 max,min,avg 等)、分布特征、文本特征(如字频、关键词等)、依赖关系(如互信息、相关性等)等实例级信息的相似性进行匹配。实例是对语义的深入描述,如果数据质量较好,在模式级信息语义不明确的情况下,它能起到很好的辅助匹配作用。但在实际应用领域中(如数据仓库),通常目标模式是一个全局视图,没有实例信息,因此只能依靠模式级匹配方法。在某些情况下,模式级和实例级信息的语义都不明确,文[4]中提出了一种基于信息论的无解释(un-interpreted)匹配方法较好地解决了这一问题。

为了利用以上匹配方法实现自动模式匹配,主要采用了两种方式:基于规则(rule-based)的实现方式和基于学习(learning-based)的实现方式。基于规则的实现方式使用预先定义的手工(handcrafted)规则进行匹配,通过考虑成员名称、数据类型、结构等模式信息的相似性来建立规则,指导匹配过程。例如,TranScm 系统[5]中使用规则"如果两个成员具有相同的名称(包括同义词)和子成员数,则匹配";CUPID 系统[6]使用了基于名称、数据类型和值域进行成员分类的规则。基于学习的实现方式利用机器学习方法进行匹配。例如,SemInt 系统[7]提出了一种基于神经网络的模式匹配原型;Automatch 系统[8]提出了一种基于贝叶斯学习和特征选择的模式匹配方法;LSD、COMAP 和 GLUE[9]等设计了一种三层结构的多策略学习(multi-strategy learning)框架。基于规则的方式实现简单、成本较低;基于学习的方式需要训练,但可以重用匹配结果,易于扩展。

2.2 存在的问题

匹配操作研究在一定程度上解决了简单模式中的匹配发现问题,但使其实用还有很大距离。除了当前的匹配方法尚不能较好地挖掘语义关系的原因之外,还有未充分研究匹配前的预处理、匹配结果的动态维护、自动匹配与用户参与的配合以及复杂模式中的匹配问题等因素。最近,人们开始重视这些方面的研究,例如文[6]和[7]介绍了一些匹配前的预处理方法,文[10]中介绍了一种用户有效参与的匹配方法,文[12]中介绍了一种模式映射结果的维护工具。但复杂模式中的匹配研究不多。因此,在应用中存在以下问题:

(1)匹配效率不高、质量较差。假设模式 S 有 X 个实体、M 个属性,模式 T 有 Y 个实体、N 个属性,如果仅考虑1:1的成对属性匹配,至少需要进行 $M\times N$ 次比对。如果属性较少,则工作量较小,质量较好;如果属性较多,则效率较低且质量较差。但实际应用中往往是实体和属性很多。此外,由此产生的匹配结果将是一个非常巨大的映射集合,由人工进行验证和筛选的工作量也会很大。实际上,直接成对匹配是没有意义的,因为绝大多数比较计算都发生在语义无关的模式成员之间。

(2)重复匹配。在数据库中,实体之间的联系通常用主外键表示,因此某些键码属性在多个实体中存在。另外,实际应用中为了方便查询和处理,某些常用信息可能存在冗余。如果直接匹配,会产生大量重复匹配。

(3)同名异义的语义冲突匹配。即名称(或描述)相同或相似,但实际语义不同。例如,在数据仓库建设中,数据源 S 中的实体 $S1$和实体 $S2$都用属性 *amount* 描述金额,但 $A1$表示交易类信息,$S2$表示申请类信息;假设数据仓库模式 T 中,属性 *amount* 仅用来代表交易金额,如果不考虑完整的语义和逻辑分类,直接进行匹配,很可能将 $S2$中的 *amount* 与数据仓库中的相应属性建立匹配,但它们分属于不同的语义背景,毫无关系。这种情况在实际应用中比较常见。

(4)复杂匹配关系的处理难度加大。复杂匹配是指模式成员之间的1:n,n:1或 n:m 匹配关系,它是模式匹配中的难点。在模式信息复杂的情况下,直接进行匹配,会产生很多无关和重复的对应关系,增加了复杂匹配的处理难度。例如,引言部分的复杂匹配示例中,如果没有实例信息,模式 S 中可能有多个 price 和 tax_rate 与模式 T 中的 list_price 有关,如何选择将比较困难。

解决这些问题实际上与匹配前的预处理工作有着直接关系。如通过规范化、拆分、合并、归约等方法来简化相似性判断难度,通过模式信息分类来提高匹配效率和简化工作量等。尤其是模式信息分类对复杂模式中的匹配问题更为重要。模式信息分类一般包括两个层次:一是模式级的分类,是根据数据成员集合所表示的完整语义来进行分类,如关系模式中的实体分类、XML 中的元素分类、面向对象数据库中的对象分类等;二是成员级分类,是根据某个成员的名称、类型、约束、数值特征等特性进行分类,如数据库和 XML 中的属性分类等。只有将两者有效结合,才能较好地解决以上四个问题。文[6]的基于规则的属性分类方法和文[7]的基于聚类的属性分类方法,在一定程度上提高了匹配效率,但不能消除重复匹配、同名异义的语义冲突问题,也不能降低复杂匹配关系的处理难度。本文提出的使用朴素贝叶斯学习进行实体分类,然后在同类子模式之间进行模式匹配的方法,有效地解决了上述四个问题。以下将进行详细说明。

3 基于实体分类的数据库模式匹配方法

3.1 方法概述

基于实体分类的数据库模式匹配方法的基本思想是:对给定的两个输入模式 S 和 T,按照已知的数据分类标准和分类标记集合 C,选择合适的分类学习方法 M,进行实体分类,分别形成关于两个模式的若干子模式;然后在具有相同分类的子模式之间进行匹配,生成匹配结果。

模式是数据的逻辑结构和特征的描述,是数据在逻辑层上的视图。根据数据库的设计思想和方法,实体在概念层上可以归为不同类别,且同一应用领域的数据库模式具有大致相同的分类标准。因此,可以对同一应用领域内的不同模式,采用同类分类标准进行划分。例如,可以按照银行数据仓库的逻辑数据模型中的主题划分标准,把数据源系统中的实体划分为客户、帐户、产品、交易、渠道等类别。

当然,根据分类标准,某些实体可能不止归为一类,如某系统中的交易帐户实体同属于帐户类和交易类。为了避免遗漏,需要选择合适的分类学习方法。本文中选用了朴素贝叶斯分类方法。原因在于:该方法是以概率形式表示某个样本属于特定类的可能性,且具有较好的查准率和查全率。对于某个分类,我们可以设定一个可能性阈值 ω,在此阈值之上的实体,视为同一类别,组成子模式,并用概率值表示实体对某一类别的隶属程度。然后在两个输入模式的同类子模式之间进行匹配。这样的话,既降低了匹配成本,又使某些具有多重身份的实体划分在多个子模式中,并参与多个子模式之间的匹配,兼顾了匹配的全面性。例如,经朴素贝叶斯分类和与对应分类阈值的比较,模式 S 中的实体 s' 分别以概率 P_i 存在于类 c_i 中和以概率 P_j 存在与类 c_j 中,那么它就可以参与两类子模式之

维普资讯 http://www.cqvip.com

间的模式匹配。

3.2 基于朴素贝叶斯学习的实体分类方法

通常，朴素贝叶斯分类方法用 n 维特征向量来表示数据样本。但实体信息包括实体名、属性名、属性描述、主外键、数据类型、值域、约束等内容，很难组织成 n 维特征向量的形式。因此，本文采用了文[11]中提出的基于朴素贝叶斯学习的文本分类算法，即把实体信息组织成文本，并对其进行了修改，以适合于前面所述的分类思想。基于朴素贝叶斯学习的文本分类算法选取数据样本中的单词表达主题特征，将样本表示成一组单词集合，单词在样本中出现的次数作为它的值，使用单词和分类的联合概率来估计样本的归属类别。这种方法的查全率和查准率都比较高。

但是，对于分类而言，并不是所有实体信息都是可用的，如数据类型、值域、约束等，不能直接作为分类依据，因此需要进行选择。一般来说，实体名代表了实体的语义，但限于长度和抽象难度，它往往不能完全反映语义内容。主键可以唯一地标识某个实体，但这种标识也不能完全反映出实体的语义和具体归属关系。而所有属性的集合(包括主外键)则在一定程度上比较完全地反映了实体的语义内容。因此，可以将属性集合作为数据样本。一般而言，同一模式中的命名遵守同一标准，如果属性命名不规范，为提高分类准确性，需进行解析和标准化等预处理，如将 custID 转换解析并转换成(cutomer，ID)。相应地，用于计算概率的词汇表由预处理后的实体名、属性名或属性描述构成。由此构成的训练数据集及其人工分类结果如表1所示。

表1 训练数据集示例

模式信息	分类
Customer, ID, name, age, address,…	*Customer*
Card, ID, type, state, balance …	*Product*
ATM, ID, model, location,…	*Channel*
……	……

算法分为两个阶段：一是学习阶段，分析所有训练样例，从中抽取所有出现的单词，然后在不同目标类 c_j 中计算某个单词 w_k 的频率以获得必要的概率估计 $P(w_k|c_j)$，同时也学习类别的先验概率 $P(c_j)$；二是分类阶段，输入新的待分类实体和分类阈值 ω(不同类别的阈值可以不同，为简便起见，在此我们选用了同一阈值)，使用前面计算出的概率估计，按照 $\{c_{NB}|c_{NB}=P(c_j)\prod_{i\in pos}P(a_i|c_j)\geqslant\omega\}$ 计算分类结果集 C_{NB}。算法的伪码如下：

算法1 基于朴素贝叶斯的实体分类方法

learn_naïve_bayes_entity(samples, C)
{ *//samples* 为一组实体信息文档和手工分类结果。*C* 为所有可能分类的集合。
(1)收集 *samples* 中所有出现的单词形成词汇表 *Vocabulary*
(2)计算所需要的概率项 $P(c_j)$ 和 $P(w_k|c_j)$
For $c_j\in C$ Do
{ $docs_j\leftarrow$*samples* 中分类为 c_j 的文档子集；
$P(c_j)=\frac{|docs_j|}{|samples|}$；
$Text_j\leftarrow$将 $docs_j$ 中所有成员连接起来建立的单个文档；
$n\leftarrow$在 $Text_j$ 中不同单词位置的总数；
For $w_k\in Vocabulary$ Do
{$n_k\leftarrow$单词 w_k 出现在 $Text_j$ 中的次数；
$P(w_k|c_j)=\frac{n_k+1}{n+|Vocabulary|}$}
}}
classify_naïve_bayes_entity(newentity, ω)
{//对 *newsentity* 返回其估计的分类结果集 $C=\{\langle c_j,P_j\rangle\}$。$a_i$ 代表在 *newentity* 中的第 i 个位置上出现的单词。
pos←在 *newentity* 中所有单词位置
返回分类结果 C：
$C=\{\langle c_j,P_j\rangle|P_j=P(c_j)\prod_{i\in pos}P(a_i|c_j)\geqslant\omega\}$

3.3 模式匹配方法

实体分类后，以同类实体作为输入，可以采用第2部分中介绍的各类方法进行匹配。鉴于本文的目的是为了介绍实体分类对数据库模式匹配过程的促进作用，因此选择了实现起来比较简单的基于规则的模式匹配方法。限于篇幅，在此仅介绍匹配的基本过程。

算法2 基于规则的模式匹配方法

Step1：schema preprocessing and transforming
将输入的同类(c_j)子模式中的已经预处理的实体 $s_i\in S$ 与 $t_j\in T$ 按名称、类型、约束等进行属性级分类(V)，并转换成模式树[7]形式。
Step2：linguistic matching
通过名称、描述相似性等语言学方法进行匹配，生成语言相似性：
$$lsim(m_1,m_2)=ns(m_1,m_2)\times\max_{v_1\in V_1,v_2\in V_2}ns(v_1,v_2)\times\min(P(s_i\in c_j),P(t_j\in c_j))$$
其中 $m_1\in s_i$，$m_2\in t'_j$，$ns(x,y)$ 为名称相似性估计，v_1 和 v_2 为 m_1 和 m_2 的属性级分类，$P(s_i\in c_j)$，$P(t_j\in c_j)$ 为 s_i 与 t_j 属于实体类别 c_j 的概率，在实体分类时已计算。
Step3：Structural matching
通过 TreeMatch 算法[7]计算成员在两个不同模式中的背景(*context*)相似性 *ssim*，并以加权相似性 *wsim* 表示最终的匹配程度
$wsim=w_{struct}\times ssim+(1-w_{struct})\times lsim$
Step4：Matching generation
根据匹配阈值，筛选匹配结果并生成映射关系。

4 实验结果

4.1 实验数据与方法

我们使用了国内某银行数据仓库建设中的实际案例，评估我们的方法。两个模式分别为：数据仓库逻辑数据模型，包括客户、帐户、产品、交易、渠道等11类主题，共340多个实体、1,000多个属性；银行核心业务系统，共15个子系统、700个实体、14,000多个属性。为了简化评估工作，我们选取了数据仓库逻辑数据模型中的客户、帐户和交易类主题的部分模式信息(共30个实体、152个属性)以及银行核心业务系统中的客户信息、公用和银行卡子系统的部分模式信息(共100个实体、983个属性)作为输入模式。

实验中以数据仓库的主题划分为分类标准，且数据仓库逻辑模型中的实体在设计时已做分类，因此只需对银行核心业务系统中的实体按照客户类、帐户类和交易类进行划分。实验中选取了40个实体(共378个属性)作为训练样例集并进行手工分类和构造词汇表，选取10个实体(共112个属性)进行验证，训练结果用于剩余50个实体(493个属性)的分类。经与手工分类结果比较，分类达到了85%的准确率。为了不影响匹配质量，在进行模式匹配之前，我们对错误的分类进行了更正。

4.2 评估与分析

为了评估匹配质量，我们事先已手工建立了所有100个实体与数据仓库30个实体之间的属性匹配关系，以此作为衡量匹配质量的基准。评估采用以下三个通用指标：

(1)查准率。匹配结果中的正确匹配结果的比率：$Precision=T/P=T/(T+F)$；

(2)查全率。匹配结果中的正确匹配结果占实际匹配结果的比率：$Recall=T/R$；

(3)全面性。评估后期匹配(post-match)工作量：$Overall=Recall*(2-1/Precision)$。

其中 T 为正确识别的匹配结果，P 为匹配方法返回的匹配结果，R 为手工匹配结果，F 为错误的匹配。评估结果如图1所示。

(下转封四)

维普资讯 http://www.cqvip.com

(上接第 159 页)

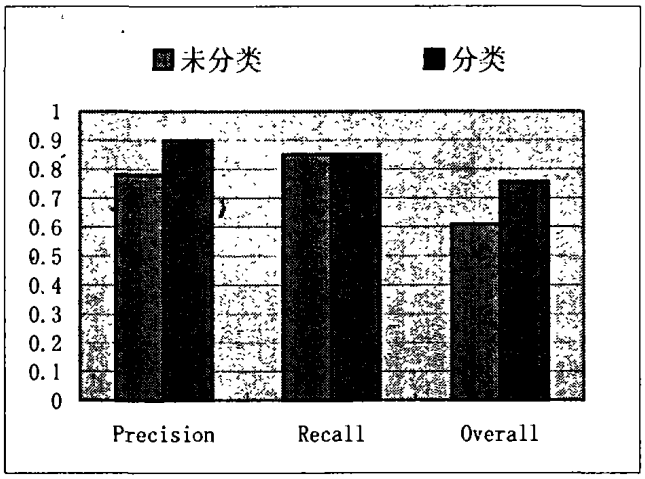

图 1 匹配结果评估

从图 1 中我们可以看出,实体分类后,Precision 和 Overall 都有较大提高。Recall 维持不变,其原因是使用了同样的匹配判断方法和实现方式。当然,在分类准确度不高的情况下,Recall 也有可能因为遗漏正确的匹配关系而下降,因此分类后需要借助人工来校正。Precision 和 Overall 的显著提高是因为实体分类有效地减少了同名异义的语义冲突匹配、重复匹配和匹配结果集的大小。如表 2 所示。

表 2 匹配结果集的比较

	未分类	分类
匹配结果集大小	353	297
同名异义的语义冲突匹配数	47	5
重复匹配数	18	5

匹配结果集缩小了约 16%,同名异义的语义冲突匹配数减少了约 89%,重复匹配数减少了约 67%。这三者的减少大大地减轻了用户筛选和验证匹配结果的工作量。此外,也降低了复杂匹配的处理难度。因为在没有实例信息的情况下,复杂匹配通常是简化为 1∶1 匹配后,由用户判定匹配成员的函数关系,所以 1∶1 匹配数量和语义冲突匹配数的减少,可以有效地降低复杂匹配的处理难度。更为重要的是,由于采用在相同类别的子模式之间进行匹配,因此可以大大减小匹配算法的搜索空间,提高效率。例如,假设实体分类前,两个模式匹配过程中的比较次数为 $M\times N$ 次,那么分类后,比较次数最多可以减少到 $(M\times N)/C$ 次(其中 C 为分类数),因此效率提高了约 C 倍。

总结与展望 本文介绍了一种基于实体分类的数据库模式匹配方法,在模式比较复杂的情况下,该方法有效地减小了匹配算法的搜索空间和匹配结果集,节省了用户筛选工作量,减少了重复匹配和同名异义的语义冲突匹配,降低了复杂匹配关系的处理难度,提高了匹配的总体质量。但是这种方法可能会因为分类方法的准确性问题,遗漏一些正确匹配关系,因此需要分类后的用户验证。未来可以针对模式信息特点,探讨更合理、更准确的分类方法的应用。此外,如何将模式级分类和成员级分类更为有效地结合,也值得进一步研究。其它方面,如用户的有效介入、复杂匹配关系的发现、匹配结果的维护等问题,以及与模式匹配相关的 ontology 映射和 Web 服务的相似行为匹配问题都值得深入研究。

参 考 文 献

1 Rahm E, Bernstein P A. A survey of approaches to automatic schema matching. The VLDB Journal, 2001, 10: 334~350
2 Berlin J, Motro A. Database Schema Matching Using Machine Learning with Feature Selection. In: Proc. of the 14th Intl. Conf. on Advanced Information Systems Engineering (CAiSE), 2002
3 Kang J, Naughton J F. On Schema Matching with Opaque Column Names and Data Values. SIGMOD Conf. 2003
4 Kang J, Naughton J F. On Schema Matching with Opaque Column Names and Data Values. SIGMOD Conf. 2003
5 Milo T, Zohar S. Using Schema Matching to Simplify Heterogeneous Data Translation. In Proc. of the Intl. Conf. on Very Large Databases (VLDB), 1998
6 Madhavan J, Bernstein P A, Rahm E. Generic Schema Matching with Cupid. In: Proc 27th Int Conf. on Very Large Data Bases (VLDB), 2001
7 Li W, Clifton C. SemInt: a Tool for Identifying attribute correspondences in heterogeneous databases using neural network. Data Knowledge Engineering, 2000, 33(1): 49~84
8 Berlin J, Motro A. Database Schema Matching Using Machine Learning with Feature Selection. In: Proc. of the 14th Intl. Conf. Advanced Information Systems Engineering (CAiSE), 2002
9 Doan A. Learning to Map between Structured Representations of Data: [A PhD Degree Dissertation]. http://anhai.cs.uiuc.edu/home/thesis/anhai-thesis.pdf, 2002
10 Wang G, Goguen J, Nam Y-K, Lin K. Interactive Schema Matching with Semantic Function. http://www.cs.utexas.edu/users/francois/Cam08.pdf, 2003
11 Joachims T A. Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization: [Computer Science Technical Report CMU-CS-96-118]. Carnegie Mellon University, 1996
12 Velegrakis Y, Miller R J, Popa L. Mapping Adaptation Under Evolving Schemas. In: VLDB, Berlin, Germany, 2003

9-2817
TP
2004.31(10)

计算机科学

(1974 年 1 月创刊)
第 31 卷第 10 期 (月刊)
2004 年 10 月 25 日出版

ISSN 1002—137X
CN50—1075/TP

定价: 20.00 元 国外定价: 5 美元
邮发代号: 78—68
发行范围: 国内外公开

主管单位: 国家科学技术部
主办单位: 国家科技部西南信息中心
编辑出版: 《计算机科学》杂志社
重庆市渝中区胜利路 132 号 邮政编码: 400013
电话: (023) 63500828 E-mail: jsjkx@swic.ac.cn
社 长: 牟炳林
主 编: 彭 丹
副 主 编: 朱宗元
印 刷 者: 重庆科情印务有限公司
总发行处: 重庆市邮政局
订 购 处: 全国各地邮政局
国外总发行: 中国国际图书贸易总公司 (北京399 信箱)
国外代号: 6210—MO