

分类号 TP3

密级

UDC

编号

中国科学院研究生院

博士学位论文

面向人物追踪的知识挖掘研究

于满泉

指导教师 白硕 研究员

中国科学院计算技术研究所

申请学位级别 工学博士 学科专业名称 计算机软件与理论

论文提交日期 2006 年 4 月 论文答辩日期 2006 年 6 月

培养单位 中国科学院计算技术研究所

学位授予单位 中国科学院研究生院

答辩委员会主席

声 明

我声明本论文是我本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，本论文中不包含其他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

作者签名：

日期：

论文版权使用授权书

本人授权中国科学院计算技术研究所可以保留并向国家有关部门或机构送交本论文的复印件和电子文档，允许本论文被查阅和借阅，可以将本论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编本论文。

（保密论文在解密后适用本授权书。）

作者签名：

导师签名：

日期：

摘 要

近年来,关于文本信息处理的诸如检索、分类、聚类、抽取等技术有了很大的发展,目前的研究偏重于综合利用各种手段来更好地满足新的应用场景,一系列国际评测也推动了各项技术的发展。尽管在移动通信领域,跟踪人物行踪的技术已相对成熟;然而,在自由文本空间内,如何根据人物的报道自动整理出相关人物的信息并没有引起研究者的广泛注意,其过程和方法也没有统一的结论。

为此,本文提出了“人物追踪”的概念,首先对“人物追踪”进行了概括性的介绍,论述了“人物”和“人物追踪”的内涵和外延,提出了“人物模型”的文本表示方法,并对其进行了形式化的定义。随后,给出了人物追踪要解决的主要问题,相关的技术基础,处理文档资料的系统流程以及人物追踪在学术上和应用上的意义。

在对人物追踪涉及到的主要技术进行了理论和方法上的综述后,本文结合人物追踪的系统流程,介绍了人物追踪数据预处理的方法。处理的文档主要是汉语的新闻网页。重点研究了网页解析和人物属性抽取的方法。通过数据预处理,把文本内容转化成人物模型的信息片断。

接下来,本文研究了人物模型的同一性判断方法和数据融合方法。主要是处理人物模型片断的信息,不仅要把不同人物的信息区分开,还要把同一人物的信息进行整合,形成完整的人物生平履历。本文在这部分重点介绍了人物模型的特征表示和相似度计算方法。

在人物模型同一性判断基础上,针对同一人物的活动报道,本文研究了人物活动事件的组织方法。事件的组织分为“微观粒度的事件组织”和“宏观粒度的事件组织”。本文对宏观粒度事件组织进行了重点的研究,以标准的国际评测为平台,以文档或段落为基本的处理单位,不涉及具体的事件角色框架来对事件报道进行组织。本文不仅研究了宏观粒度事件的识别技术,还研究了事件之间层次关系的组织方法。

随后,本文探讨了人物追踪若干细粒度挖掘任务的处理方法,并以“流通度理论”和“动态流通语料库理论”为基础,重点研究了人物知名度的计算方法。通过跟踪人物不同时期的知名度变化,绘制出人物知名度变化的曲线图。

通过结合人物追踪的理论和技术研究,本文最后给出了人物信息搜索引擎的设计方案。不仅设计了体系结构,还给出了人物搜索引擎的索引结构和搜索结果显示界面。

本文的研究成果为自由文本空间内人物信息的大规模组织奠定了坚实的基础,所提出的“人物追踪”概念为文本挖掘领域提供了一个崭新的研究视角和方向,对于抽象对象的挖掘具有重要的启发意义和实用价值。

关键词: 人物, 人物追踪, 知识挖掘, 人物模型, 人物属性抽取, 同一性判断, 事件信息组织, 知名度计算, 人物搜索引擎

Research on Knowledge Mining in Person Tracking

Yu Manquan (Computer Software and Theory)

Directed By Professor Bai Shuo

There has been great improvement in the technology of information retrieval, text categorization, text clustering and information extraction. Now the research has focused on solving new problems using various methods. A series of evaluations has pushed forward the development of related research fields. In the field of mobile communication, the technology of tracking a person has been comparatively mature. But in the field of text mining, the method of tracking a person has not arisen enough attentions and has not drew a conclusion about it.

Therefore, this article proposed the notion of *Person Tracking*. First it carried on the concise introduction of *Person Tracking*, elaborated the connotation and the extension of *Person* and *Person Tracking*, then it proposed *Person Model* and carried on the formalized definition to it. Afterwards, it produced the main question, the technical foundation, the system flow as well as the application of *Person Tracking*.

After making the summery of *Person Tracking*, this article introduced the method of data preprocessing. The processed documents are mainly Chinese News WebPages. This article researched deeply of the method of HTML parsing and person attribute extracting. Through preprocessing, it turned the text content into fragment of *Person Model*.

Then this article studied the identical judgement of *Person Model*. The main problem is about processsing the fragments of *Person Model*, not only separating different person's information, but also integrating information of the same person. This article used the information to make biography for the person. This article deeply introduced the feature expressing and the similarity computing methods in this part.

In the foundation of identical judgement of *Person Model*, this article studied the methods of producing person's activity reports. This is so-called event-based information organization. The event organization divides into microscopic granularity event organization and macroscopic granularity event organization. This article researched the macroscopic ones. It taked the standard international evaluation as the platform, taked the documents or the paragraphs as the basic processing units and did not involve the concrete event role frame. In this way, this article not only studied the macroscopic granularity event detection technology, but also studied the method of event relation organization.

Afterwards, this article discussed the methods of several microscopic granularity tasks. It taked the circulation theory as the foundation, studied the method of the recognition degree computing. Through tracking a person's recognition degree in different periods, the person's recognition charts can be drew.

Through unifying the theory and the technical research of *Person Tracking*, this article

finally proposed a method of buiding person information search engine. It not only designed the system structure, but also produced the index structure and the interface of the search engine.

The research achievements of this article established solid foundation for the person information processing in free texts. The proposed *Person Tracking* provided a brand-new research angle for the field of text mining. And it had important practical value for the mining of abstract object.

Keywords: person, person tracking, knowledge mining, person model, person attribute extracting, identical judgement, event-based information organization, recognition computing, person search engine

目 录

摘 要.....	I
目 录.....	V
图目录.....	IX
表目录.....	XIII
第一章 引言.....	1
1.1 人物追踪产生的背景	1
1.1.1 信息增长的现实驱动	1
1.1.2 业务模式发展的内在要求	2
1.1.3 技术发展的内在逻辑	5
1.2 人物角度信息组织的研究现状	7
1.3 人物和人物追踪的内涵及外延	8
1.3.1 人物	8
1.3.2 人物追踪	9
1.3.3 人物追踪的主要问题	10
1.3.4 人物追踪的技术基础	11
1.3.5 人物追踪的系统流程	12
1.4 人物追踪的意义	12
1.5 本文的组织结构	13
第二章 人物追踪的相关技术综述	15
2.1 网页解析	15
2.1.1 基于 HTML 标记分布规律的解析方法	15
2.1.2 基于 HTML 标记间的关系的解析方法	16
2.1.3 基于视觉特征的解析方法	17
2.1.4 基于 TABLE 标记布局特性的解析方法	18
2.2 信息抽取	18
2.2.1 信息抽取的发展历史	18
2.2.2 信息抽取系统的基本流程	20
2.2.3 信息抽取的发展方向	21

2.3 文本聚类	23
2.3.1 文本的表示	23
2.3.2 距离的度量	24
2.3.3 聚类的算法体系	26
2.4 话题识别与跟踪	30
2.4.1 话题识别与跟踪的背景与历史	30
2.4.2 话题识别与跟踪的主要任务	31
2.4.2 话题识别的主要技术	32
2.5 本章小结	34
第三章 人物追踪中的数据预处理	35
3.1 数据预处理的主要流程	35
3.2 网页噪音过滤	36
3.2.1 网页解析算法 TVPS	37
3.2.2 实验结果与分析	39
3.3 人名的识别	40
3.4 人名指代的处理	41
3.5 人物对应语段的确定	42
3.6 时间和时序标签的确定	43
3.7 人物属性的抽取	43
3.7.1 人物属性抽取的总体框架	44
3.7.2 人物属性抽取语料集的标注	45
3.7.3 分类器的模型表示	46
3.7.4 分类器的训练	49
3.7.5 分类器的测试	49
3.8 本章小结	53
第四章 人物模型的同-性判别与动态填充	55
4.1 概述	55
4.2 相关研究工作	56
4.2.1 混合类型数据的聚类	56
4.2.2 IE 在 IR 中的应用	57
4.2.3 人名消歧的主要方法	58
4.3 人物模型 (Person Model) 的特征表示	59
4.3.1 属性特征的表示	60
4.3.2 数值特征的表示	60

4.3.3 各项特征的分布规律	62
4.4 人物模型的相似度计算方法.....	62
4.4.1 基本属性的相似度计算.....	62
4.4.2 介绍性属性的相似度计算.....	63
4.4.3 词场的相似度计算.....	64
4.4.4 人物模型总的相似度计算.....	65
4.5 人物模型的同源性判别与合并.....	67
4.6 实验结果与分析.....	68
4.6.1 数据集.....	68
4.6.2 评测方法	69
4.6.3 实验结果	70
4.7 本章小结	73
第五章 基于事件的信息组织.....	75
5.1 微观粒度的事件组织	75
5.2 宏观粒度的事件组织	76
5.2.1 宏观粒度事件的特征	78
5.2.2 针对事件特点的话题识别方法.....	80
5.2.3 基于多层聚类的话题层次化组织方法	83
5.2.4 实验结果与分析	87
5.3 本章小结	92
第六章 若干细粒度挖掘任务的处理.....	95
6.1 概述.....	95
6.2 人物知名度的追踪方法.....	97
6.2.1 词语流通度的计算.....	97
6.2.2 人物知名度的计算.....	98
6.2.3 流行人物的挖掘	101
6.3 本章小结	102
第七章 人物搜索引擎的设计.....	103
7.1 人物搜索引擎的体系结构	103
7.2 人物搜索引擎的索引结构	105
7.3 搜索界面及结果.....	107
7.4 本章小结	109
第八章 结束语	111

8.1 本文主要贡献与创新	111
8.2 下一步研究工作.....	112
8.3 前景与展望.....	113
参考文献	115
致 谢.....	i
作者简历	iii

图目录

图 1.1a “张静”的 Google 检索结果片断.....	3
图 1.1b “江华”的 Baidu 检索结果片断.....	3
图 1.2a “薄熙来”的 Baidu 新闻正文检索结果片断.....	4
图 1.2b “薄熙来”的 Baidu 新闻标题检索结果片断	4
图 1.3 同一性判断问题示意图	5
图 1.4 数据融合问题示意图	6
图 1.5 跟踪问题示意图	6
图 1.6 人物模型的示意图	9
图 1.7a “江泽民”的简历页面.....	10
图 1.7b “江泽民”的活动报道专集页面	10
图 1.8 人物追踪的系统流程	12
图 2.1 HTML 结构化显示标记关系图.....	16
图 2.2a 原始的标记关系树.....	16
图 2.2b 转化后得到的语义树	16
图 2.3 VIPS 算法流程图.....	17
图 2.4 网页的 TABLE 布局示例图.....	18
图 2.5 信息抽取系统的典型流程图	20
图 3.1 人物追踪中数据预处理的主要流程示意图	36
图 3.2 网页噪音过滤示意图	37
图 3.3 TVPS 算法分析 HTML 文档结构示意图.....	38
图 3.4 人物对应语段确定示意图	42
图 3.5 人物属性抽取方法流程图	45
图 3.6 使用概念义原前后人物属性抽取的效果对比图	52

图 4.1 人物模型片断示意图	60
图 4.2 单纯采用词场 R 对“薄熙来”进行同名判别的结果	65
图 4.3 同名人物判别方法示意图	66
图 4.4 人物简历模板结构	67
图 4.5 同名判别和人物简历的呈现结果示意图	67
图 4.6 典型的 10 个人名的同名情况分布图	69
图 4.7 人物模型的普通词场划分效果对比图	70
图 4.8 人物模型不同维度效果对比图	71
图 4.9 不进行“同名判别”时典型的 10 个人名的效果图	72
图 4.10 进行“同名判别”后典型的 10 个人名的效果图	73
图 5.1 事件发展规律示意图	78
图 5.2 不同事件的报道数和子事件数示意图	79
图 5.3 事件的不同时期内各大新闻媒体的报道数目示意图	80
图 5.4 话题层次结构示意图	84
图 5.5 MLCS 算法基本思想示意图	85
图 5.6 MLCS 算法描述	86
图 5.7 MLCS 算法示意图	86
图 5.8 不同分组策略效果对比图	89
图 5.9 不同分组间隔效果分布图	89
图 5.10 TDT2004 中 HTD 任务评测结果对比图	92
图 6.1 语段倾向性的张量模型示意图	96
图 6.2 人物知名度曲线示意图	101
图 7.1 人物搜索引擎 PerSou 体系结构示意图	104
图 7.2 人物搜索引擎 PerSou 索引结构示意图	106
图 7.3 人物搜索引擎 PerSou 的搜索界面图	108

图 7.4 人物搜索引擎 PerSou 的同名人物返回结果示意图	108
图 7.5 人物搜索引擎 PerSou 的具体人物信息示意图	109

表目录

表 3.1 TVPS 算法的实验结果.....	40
表 3.2 人物属性项对应的窗口长度	49
表 3.3 人物属性触发词库的建立方法	50
表 3.4 人物属性短语识别规则表	51
表 3.5 采用普通词场向量的人物属性抽取效果	52
表 3.6 采用概念义原向量的人物属性抽取效果	52
表 4.1 4315 个人物模型片断中各个属性的出现次数和稀疏度.....	62
表 4.2 人物模型各维度效果对比统计表	71
表 4.3 汉语语料人物同一性判别中不同系统的效果对比	72
表 5.1 针对事件特点的话题识别方法效果对比表	90
表 5.2 MLCS 算法生成的不同话题结构的效果对比表.....	91
表 5.3 TDT2004 中 HTD 任务所提交结果的参数与效果	91
表 6.1 2005 年 12 月 5 日 Alexa 发布的中文网站流量前 10 名基本情况	99

第一章 引言

随着信息传播手段的进步，尤其是互联网这一新媒体的出现，我们已经摆脱了信息贫乏的桎梏，迈入了一个信息极度丰富多彩的时代。然而，信息的极度膨胀在提供给人们丰富的信息内容的同时，也给人们带来了信息选择方面的困惑。在这种情况下，就造成了两种局面：一方面信息快速膨胀，极度过剩；另一方面人们却淹没于浩瀚的知识海洋中，找不到自己所需要的信息。因此，如何快捷准确的获取感兴趣的信息成为人们关注的主要问题。

1.1 人物追踪产生的背景

在这里，我们主要介绍本项研究产生的背景。主要从三个方面来阐述：信息增长的现实驱动；业务模式发展的内在要求；技术发展的内在逻辑。

1.1.1 信息增长的现实驱动

自九十年代初互联网开始迅速发展至今，虽然只有短短十几年的历史，却已经渗透到经济、社会、文化、教育以及娱乐等几乎各个方面，已成为我们工作和生活中不可缺少的一部分。与此同时，信息的规模呈现出爆炸式的增长趋势。

根据 2003 年美国加州大学伯克利分校发布的报告，2002 年全球产生了 5 万亿兆字节(Exabyte, 1 万亿兆= 10^{18}) 的新信息，其中 92% 的新信息以数字形式存储，这相当于 37,000 个美国国会图书馆（藏书 1700 万本）的信息量[Lyman 2003]。截止到 2004 年 12 月，Google 已经索引的网页数量已经超过 80 亿，而 Google 能检索到的页面数仅仅是整个 Web 的一部分。经过十年快速的发展，中国互联网已经形成规模，并呈现出空前的发展速度。据中国互联网络信息中心（CNNIC）在京发布的“第十五次中国互联网络发展状况统计报告”显示，到 2005 年 1 月止，我国上网用户总数为 9400 万，比半年前增长了 8%；上网计算机达到 4160 万台，比半年前增长了 14.6%；CN 下注册的域名数、网站数分别达到 43 万和 66.9 万，分别比半年前增长了 5 万和 4.2 万；网络国际出口带宽总数达到 74429M，IPv4 地址总数 59945728 个，分别比半年前增长 38%和 21%[CNNIC 2005]。而截至 2005 年 6 月 30 日，我国上网用户总数更是历史性地突破了一亿。

信息规模的极度膨胀，固然促进了信息的广泛共享，大大地丰富了人们的知识结构，为我们跨进信息时代奠定了坚实的基础，但同时也带来了许多问题和挑战。由于互联网最基本的功能就是任何信息通过它都能够自由在全世界广泛传播，信息传播的自由就使得互联网内部信息凌乱无序，有价值的信息湮没在大量冗余信息中，对其发现和管理变得越来越困难。在这种情况下，信息的规模越增长，人们就越难以找到有价值的信息。在浩瀚的知识海洋中，人们似乎只能停留在一个个“信息孤岛”上仰天长叹。因此，严

峻的现实就迫切需要人们把互联网上杂乱无序的信息梳理出脉络，按照内在的逻辑联系组织起来，才能应付“数字海啸”的到来。

1.1.2 业务模式发展的内在要求

如今，基于互联网的新的知识经济已经成为发达国家所追求的一个重要发展目标。显然，互联网经济的发展和壮大，已成为衡量一个国家现代化发展水平与质量的极为重要的指标之一。互联网应用走向多元化。人们在工作、学习和生活中越来越多地使用互联网，整个社会的运行都搭上了互联网的快车，并打上了互联网的烙印，互联网已经从单一的行业互联网发展成为深入我国各行各业的社会大众的互联网。

为了帮助广大互联网用户有效地发布和接受信息，以人工目录分类为基础的网站搜索，如：Yahoo[Yahoo!]、Excite 等，在互联网发展的早期不断涌现，它们因向广大互联网用户提供基本的信息搜索服务，而被称为互联网的门户网站。但进入九十年代后期，随着互联网开始步入正常发展阶段，先前那些以提供互联网信息目录服务的门户（综合）网站，已越来越难以满足人们对互联网信息服务的需求，因为它们所能覆盖的网页占整个互联网网页的总量的比例越来越小。

与此同时，各种采用新技术的互联网信息搜索引擎也在不断涌现。其中比较典型的是以超链分析技术[S.Brin 1998]为基础的大规模网页搜索，以 Google[Google]、百度[Baidu]为代表，其搜索结果的准确度从网站上升至网页。这类搜索引擎的特点是能够返回跟用户查询相匹配的文档列表，并把最可能符合用户需要的网页排在前边。经过几年的发展，这类搜索引擎返回网页的质量越来越高，得到了用户的广泛认可，所采用的竞价排名等业务模式，作为非常有效的广告手段也取得了巨大的成功，使搜索引擎服务成为目前互联网行业最热的市场，被称为继邮件、短信和网络游戏后众人争抢的互联网产业第四桶金，从而引发了风靡全球互联网行业的搜索服务大战。在国际搜索引擎市场，微软、YAHOO、GOOGLE 等国际巨头的争斗风生水起，上演了一幕幕悲喜剧；而以中国搜索[Zhongsou]、百度为首的一批中文搜索引擎厂商，也在中文搜索领域，与国际巨头分庭抗礼。

市场竞争的加剧使得市场格局变幻莫测。对于搜索引擎这类技术主导的产业来说，要想不被淘汰出局，必须加快技术的创新。从业务领域方面看，搜索平台逐步向桌面、移动等平台拓展，第三代搜索引擎已经在酝酿当中；从功能方面看，网页搜索依然是搜索引擎具备的基本功能，但目录搜索、新闻搜索、MP3、图片、游戏等娱乐性的搜索，也逐渐成为搜索引擎的必备功能；同时，行业搜索、书籍搜索、购物搜索、区域搜索以及影视搜索，成为近几年，特别是今年发展新的热点。

层出不穷的搜索功能极大地方便了人们对特定信息的查找。目前，人们不禁会感到，与搜索其它实体相比，搜索人物的信息极为不方便。由于目前市场上还没有完善实用的人物信息方面的搜索引擎，当人们查找一个人物信息的时候，往往不得不借助于普通的网页查找工具。我们以目前最成功的中文商用搜索引擎 Google 和 Baidu 为例，来说明目

前人物信息搜索效果的不足。

当用户想要查找有关人物“张静”的信息的时候，如果向 Google 提交查询请求，返回的前 20 个结果里面，有 6 个人物是叫“张静”的，还有一个人物是叫“张静初”的，如图 1.1a 所示。这样一来，用户不得不逐页地翻看，到底哪一个网页报道的是自己关注的那个特定人物。当用户关注人物“江华”的信息时，如果向 Baidu 提交查询，返回的结果中，大部分是包含“江华”字样的机构名，如图 1.1b 所示。这样，用户就很难从其中找到想要的信息。事实上，在查询人物的时候，这类歧义现象是十分普遍的。这些歧义可大致分为两类：一类是不同的人物及其它非人物实体可能具有相同的名字；另一类是同一个人物可能具有不同的名字或称谓[Taffet 2004]。这样的歧义就造成了人物搜索的结果十分不理想。

固然，搜索引擎提供这样的功能，就是可以在人名后边添加一个属性，通过组合查询来细化结果。但不幸的是，这样就违背了搜索引擎所追求的简单快捷的原则，而且用户往往不是精通搜索技巧的专家，他们往往不愿意提交额外的查询词。一项调查表明，当人们在 AlltheWeb.com 和 Alta Vista 上搜索人名的时候，愿意提交额外查询词的情况分别只占 11.4%和 17.5%[Spink, A. 2004]。



图 1.1a “张静”的 Google 检索结果片断



图 1.1b “江华”的 Baidu 检索结果片断

人们也许会发现，在一般搜索引擎上搜索名人的时候，遇到的“同名歧义”现象是比较少的，例如图 1.2a 和 1.2b 中所示的在 Baidu 的“新闻正文”和“新闻标题”中搜索“薄熙来”时的结果。



图 1.2a “薄熙来”的 Baidu 新闻正文
检索结果片断



图 1.2b “薄熙来”的 Baidu 新闻标题
检索结果片断

上述搜索结果之所以“同名歧义”现象较少,是因为一般的搜索引擎都会根据网页的重要程度对返回的结果进行排序。这样,在同名人物的网页中,名人的网页对非名人的网页具有“淹没”的趋势。如果我们正好关注的是名人,那么返回结果正好符合我们的需求。但是我们不禁要问:这样的结果就令人满意吗?如果了解一个人物都担任过什么职务,他现在多大年龄了,他经常跟哪些人交往最多,他都参加过哪些活动,各个活动之间有什么样的联系?返回的这些网页列表就很难让人找到答案。

从上面的分析可以看出,人们对于“人物搜索”的需求是普遍的和迫切的。一项调查表明,在 AllTheWeb 搜索引擎上,5-10%的查询词中包含人名[Guha 2004]。那么作为这么普遍的一类搜索需求,为什么“人物搜索”没有得到应有的关注,不像其它搜索功能那样流行呢?这是因为,跟其它搜索功能相比,“人物搜索”存在着一定的技术难点,主要是人物信息散落在自然语言文本片断中,还没有成型的技术能对自然语言文本中人物信息的挖掘进行有效处理,才造成商业模式的相对滞后。随着技术问题的解决,相信“人物搜索”一定会成为下一个互联网搜索的热点,推动搜索行业的进步。

另外,随着信息服务业的发展,个体和机构对信息的需求进入更加精细的阶段。信息提供者只有深化其服务内容,加强信息组织、加工的水平,提供更多经过加工、编辑、评价、筛选的信息,才能够吸引用户。互联网信息内容已经很丰富,相应的信息量足以对社会宏观特征和抽象对象的微观属性及关系进行表征,情报分析和社会调查工作也就有了新的信息源[白硕 2005]。这样,对特定人物信息的挖掘和追踪工作也就提到日程上来,例如,对政治人物活动报道追踪,对上市公司高管人员追踪等。

1.1.3 技术发展的内在逻辑

随着语料库语言学的兴起，尤其是以 Internet 为主的大规模真实文本的出现，自然语言处理和文本挖掘技术有了突发猛进的发展，实实在在的改变了人们的生活。

拿汉语来说，其信息处理技术已经有 20 余年的历史了，取得了许多实实在在的成就，其中一个重要方面就是浅层分析技术的发展。目前，汉语分词和命名实体识别的精度不断提高，已经进入实用阶段。与此同时，信息抽取、文本分类、聚类等文本挖掘技术也取得了巨大的进展，成为信息分析的有效手段。

在这种情况下，人们已不满足于对语言本身进行分析，而是希望利用语言分析的手段来挖掘一些感兴趣的抽象对象，例如事件、人物、地点、机构、音乐、软件等。这些对象的特点是可能存在歧义，分散在多数数据源中和需要持续跟踪关注等。拿“地点”来说，就存在着地名歧义，例如“新街口”，指的是北京的，还是南京的等；关于某一地点的信息可能出现在文本中，也可能出现在地图中，还可与 GPS 进行结合；地址的名称会发生沿革和变迁，例如“金陵”与“南京”等。人物信息也存在着类似的现象。总的来说，随着语言浅层分析特别是专名识别技术的成熟，目前的文本挖掘技术正在逐渐形成比文字对象高一个层次的抽象对象群[白硕 2005]。

与文字对象相比，对抽象对象的挖掘存在以下技术问题[白硕 2005]：

1. 有一个抽象的层次，与具体的名称并不形成一一对应，于是存在同一性判断问题。如图 1.3 所示。
2. 有若干属性，可能需要参照多个来源的数据才能补齐，于是存在数据融合问题。如图 1.4 所示。
3. 有持续的存在期，其间可能有种种动态变化，于是存在跟踪问题。如图 1.5 所示。

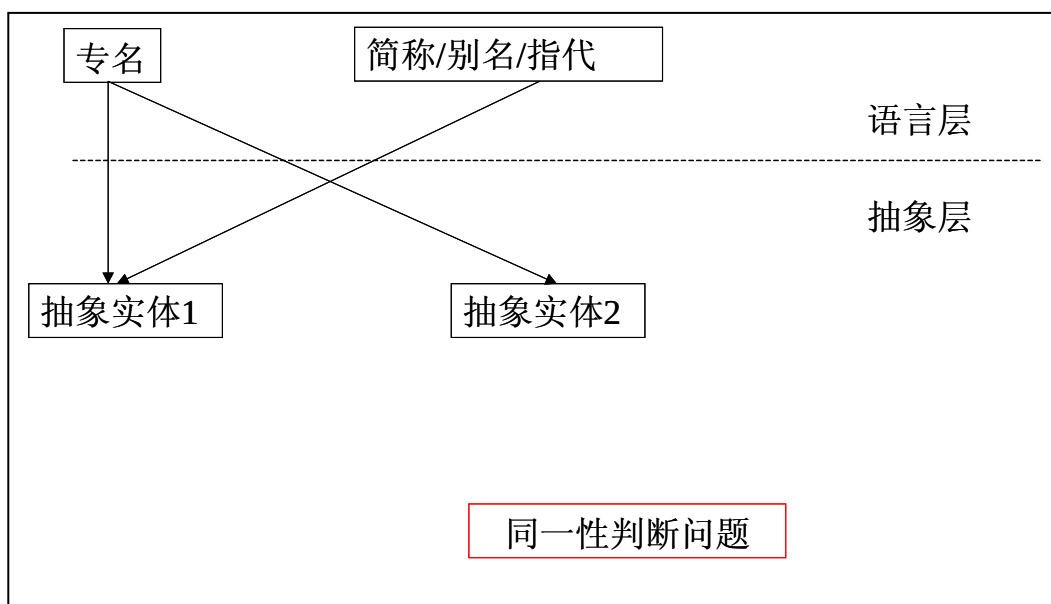


图 1.3 同一性判断问题示意图

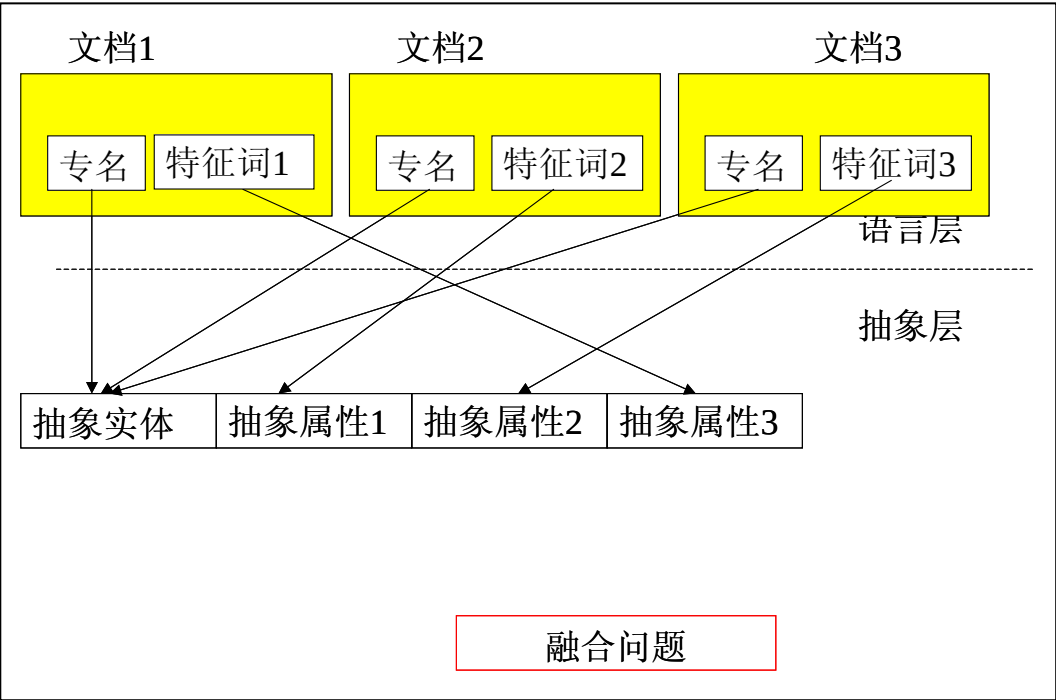


图 1.4 数据融合问题示意图

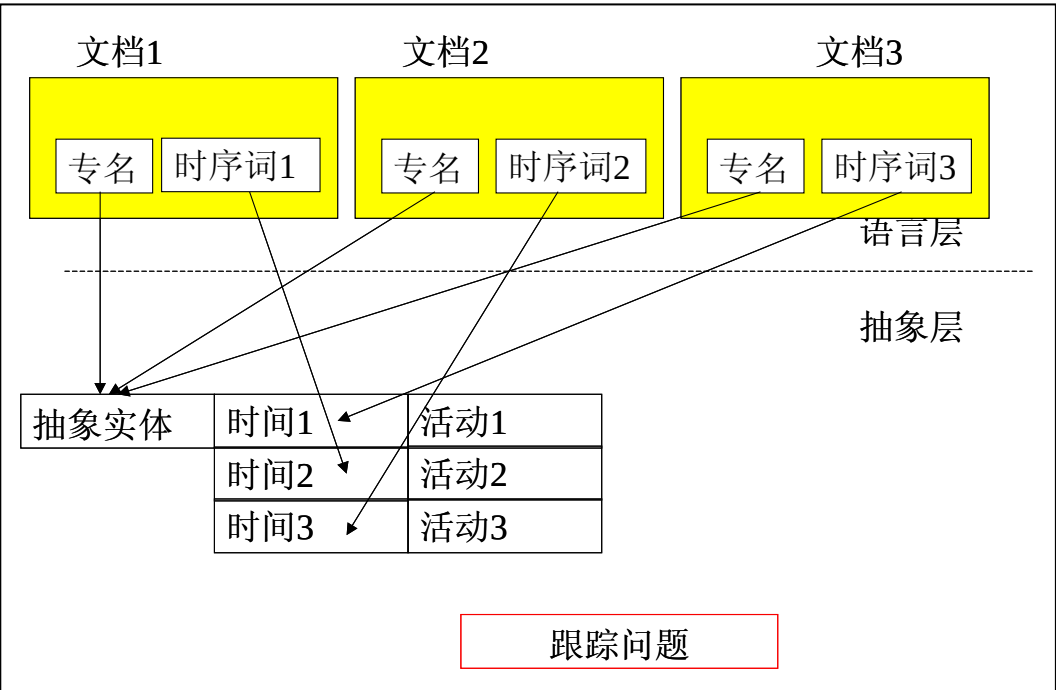


图 1.5 跟踪问题示意图

1.2 人物角度信息组织的研究现状

在信息世界里，人物角度的信息组织是一项很重要的信息需求，并随着互联网的发展，而受到越来越多的重视。

人物角度信息组织开始于自然语言处理方面的研究，主要集中在人名消歧方面。人名的消歧可以看作是自然语言里面“共指消歧”[Hobbs 1978]的一种类型，即识别出人名和它的指代词之间的对应关系。随着信息处理的不断加深，人名消歧也由单文档消歧[Wacholder 1997]扩展到跨文档消歧方面[Bagga 1998][Fleischman 2004][Niu 2004][Ravin 1999]。

近年来，信息检索技术快速发展，一些搜索引擎采用对搜索结果进行聚类的方式来减少结果中的歧义现象，例如 Vivisimo (www.vivisimo.com)、iBoogie (www.iBoogie.com)等。由于这类搜索技术意在对通用领域信息进行处理，对解决人名歧义的问题效果就不够理想。为此，一些研究者专门针对 Web 搜索结果中人名消歧的问题进行了研究[Al-Kamha 2004][Guha 2004][Xiaojun 2005]。与普通文本相比，Web 搜索结果中的语料有其独特的特点，可以利用的信息也更加丰富，例如，不仅可以利用网页中普通词汇的特征，还可利用在文字中隐含的人物属性的特征，以及利用网页之间的链接关系等。

与此同时，自然语言处理里面“多文档摘要”的技术也在不断进步，并成功应用在人物传记的生成方面[Schiffman 2001][Zoominfo]。其方法主要是结合语言学的知识和统计的信息，提选出最能代表人物特征的句子，组合起来，反映人物的生平情况。

跟人物信息组织相关的研究还包括人物社区方面的研究[Matsuo 2004]，论文引用中作者的消歧的研究[McCallum 2000][Han 2004]等。

人物信息组织的研究在搜索引擎产业中已经得到了应用，产生了若干相关的商业系统。例如，Askjeeves 的“名人搜索”功能，能够优先把检索到的名人的履历页面排在最前面；Zoominfo 的人物搜索引擎，依靠自动搜集再加用户参与的方式来提供人物的传记信息；Sogou 的人物搜索功能 (<http://people.sogou.com/>)，把名人和非名人的信息分开处理，对于非知名人物则返回数据库中用户自己注册的信息。

总的说来，人物角度信息组织在研究上和应用上都越来越受到重视，相关的技术资源也越来越完善，在某些方面已经取得了不错的进展。人物作为抽象对象的典型代表，对其进行挖掘的研究目前主要集中在“同一性判断”问题上，抽象对象挖掘的其它技术问题，尤其是“跟踪”问题的研究还有待进一步深入，也迫切需要相应理论框架的出台。另外，目前的研究主要集中在英文信息的处理上，汉语信息的需求虽然异常迫切，但相关研究却比较滞后。

本文在总结众多研究者已有工作的基础上，分析了人物角度信息组织存在的主要的问题，并提出了“人物追踪”的概念。然后，以汉语信息为处理对象，研究了相应的解决方法。下面，本文就对“人物”及“人物追踪”的概念和方法进行介绍。

1.3 人物和人物追踪的内涵及外延

为了给针对人物的信息组织的研究提供一个公共的概念平台，我们提出了“人物追踪”的概念，并分析了它要解决的主要问题以及相应的技术基础。

1.3.1 人物

在现实生活中，“人是一切社会关系的总和”。这是马克思在政治学里回答什么叫人的问题时给人下的一个定义。这是因为，每个人都生活在社会里，只有通过多方面的社会关系才能规定人的本质属性。

那么，在自由文本空间内，如何来定义“人物”？从计算机的角度来说，如果只看原始文本，只能看到字符串；有了命名实体识别，或许会看到人名，但是人名仍然不等于人物。作为真实世界直接映射的自由文本空间，“人物”的含义是通过多种信息来综合体现的。“人名”当然是其中非常重要的信息，但是还不完全，因为在文本空间内存在着大量的同名同姓却不同人物以及与姓名词形相同的非人物实体（例如“华盛顿”）的现象，即使是同一个人物实体，也可能包含着别名及多种称谓。

在文本中，人物实体是通过描述它的语段的特征来体现的，尤其是人名周边的词场，往往蕴涵着大量的相关人物的信息。词场中不同的词汇对于人物的刻画能力是不同的：有些代表着人物的属性信息，而且这些属性对于一个人物来说是终生不变的，例如“性别”、“民族”、“出生日期”、“家乡”、“父母”等，这些信息构成了人物的“基本属性信息”；还有一些词汇虽然刻画了人物的属性，但对于该人物来说可能发生变化，例如“职位”、“配偶”、“上级”、“下级”、“同事”、“工作单位”、“地址”、“电话”、“email”、

“传真”等，每一项都是一个可以变动的集合，这些信息构成了人物的“介绍性信息”；除此之外，剩下的词汇虽然不能刻画人物的属性，但往往也会跟该人物相关，可能描述了该人物参与的事件，也可能是对该人物的评论等，这些信息称之为“普通词场”。

在普通的文本表示模型中，往往把所有的词汇信息融合在一个向量里面来对事务进行描述。为了加强人物的刻画能力，我们采用“人物模型”（Person Model）的概念来表示人物的信息。考虑人物名称（姓名、别名、称谓等）的集合 $N = \{N_1, N_2, \dots, N_n\}$ ，人物基本属性的集合 $P = \{P_1, P_2, \dots, P_k\}$ ，人物介绍性信息的集合 $Q = \{Q_1, Q_2, \dots, Q_m\}$ 和普通词场的集合 $R = \{R_1, R_2, \dots, R_l\}$ ，则人物模型可以表示为 $Person = \{N, P, Q, R\}$ 。图 1.6 是人物模型的示意图。

在集合 N 、 P 、 Q 、 R 中，各个集合以及每个集合中的具体元素对人物 $Person$ 的刻画能力和方式是不同的，元素本身也可能是一个集合，它们的重要度大小也存在着差别。

如果各个集合对应的权重集依次为 $A = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$ ， $B = \{\beta_1, \beta_2, \dots, \beta_k\}$ ，

$C = \{\gamma_1, \gamma_2, \dots, \gamma_m\}$, $D = \{\delta_1, \delta_2, \dots, \delta_l\}$, 这样, 一个人物 *Person* 就对应着一个有着重要度排序的信息集合。

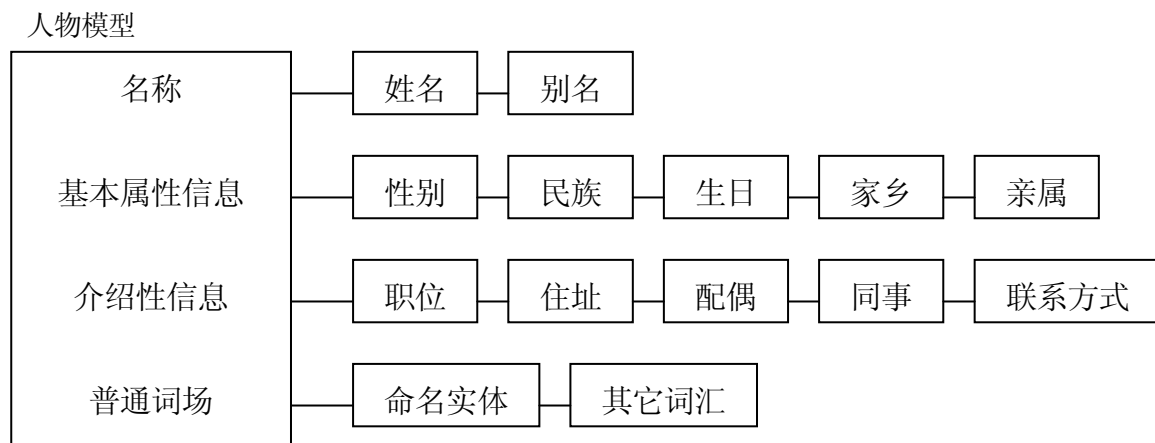


图 1.6 人物模型的示意图

1.3.2 人物追踪

在移动通信领域, 提到对人物进行追踪, 主要是跟踪某人物什么时间出现在什么地点, 说了什么话, 做了什么事情, 通过这些线索来发现该人做事的动机[Clarke 1999]。人物在真实世界中的行为会直接或间接地反映在相关的文档中, 在自由文本空间内, 对人物的跟踪也应该按照相关的线索来进行, 比如说人物的横向(空间)跟踪(人与人的联系, 人与单位的联系, 人与社会的联系等)和人物的纵向(时间)跟踪(人的职位变化, 人的处所变化, 人的态度立场变化, 人的生活状态变化等)等。

在单个的自然语言文本中, 人物相关的信息往往是零碎的、片面的, “人物追踪”就是要将不同文档中出现的人物信息片断有效地组织起来, 呈现人物信息的全貌, 反映该人物的生命历程。

在互联网上, 对于某些名人, 不乏这样的网页, 报道该人物的“简历”和“活动报道专集”, 例如图 1.7a 和图 1.7b 所示的人物“江泽民”相关的网页。通过这些网页, 人们可以清楚地了解该人物的相关信息。这些网页是利用手工整理出来的, 因而涉及到的人物范围有限。如果能够利用计算机为互联网上出现的每个人物都自动地整理出“简历”和“活动报道专集”等信息, 将使人物信息服务的水平大大提高。为此, 在自由文本处理领域, “人物追踪”在内容上可大体分为两大部分: 制作人物的履历表; 对人物的事件进行组织。

制作人物履历表就是从自由文本中自动提取出该人物的属性信息, 并把分散的各种属性信息进行整合, 形成完整的生平履历。对人物的事件进行组织, 就是把人物参与的相关事件的报道按照内在发展逻辑有效地组织起来, 形成清晰的逻辑结构。

除了大的方面外，对人物的追踪还可以沿着多条线索，按照多个角度来进行。如果按照内容进行细化，人物追踪可以有若干细粒度的深入挖掘任务。这些任务主要包括：

- 识别人物的社会关系网：识别社会上各个人物之间的横向联系。
- 建立人物的事件信息表：对人物参与的事件进行组织，提供符合事件特点的结果呈现方式。
- 识别人物职位的变化：识别人物职位的升迁和转移，挖掘人物的职业生涯。
- 识别人物处所的变化：识别人物所在地点的变化，确定人物的地点活动范围。
- 识别人物态度立场的变化：通过人物的相关报道，识别该人物对于某一事件的态度立场及其变化。
- 识别人物生活状况的变化：识别人物婚姻、家庭、亲友等的变化。
- 识别人物知名度的变化：通过相关报道判断人物知名度的变化，绘制人物知名度变化的统计图。



图 1.7a “江泽民”的简历页面



图 1.7b “江泽民”的活动报道专集页面

1.3.3 人物追踪的主要问题

人物追踪是一项旨在依据人物对语言文本信息进行组织利用的研究，也是为应对信息过载问题而提出的一项应用性研究。人物作为抽象对象的典型代表，对其信息进行组织必然会涉及到抽象对象挖掘的三大主要问题：同一性判断的问题，数据融合的问题和跟踪的问题。

人物追踪，说到底不是一个纯技术问题。所谓“人物”，不是一个技术概念，而是一个只有经过适当的建模后才能从技术层面加以处理的概念。在搜索引擎上，键入一个人名，会得到一些返回页面，其中可能不乏与同名同姓的不同人物相关的页面，也不乏与人物姓名词形相同的非人物实体（华盛顿[人名] vs. 华盛顿[城市名/州名]）。所以，第一

个技术问题就是人物的识别，总的思路是观察围绕这个人物姓名的“词场”，看能不能通过词场的某种特征反映不同的人物实体。还有就是指代的处理，指代识别正确了，潜在的词场才能转化为现实的词场，丰富人物实体的特征数据。这些都好了，“人物”这个抽象概念才在计算机可处理的世界里有了一个落脚点。

对人物的追踪，有两种方式。一种是“编年史”的方式，即以时间为主线制作人物的“大事年表”。一种是“叙事”的方式，即以事件为主线制作人物的“传记”。在有关人物的文件、活动记载具有明显时间和动态特性的情况下，把一个人物的资料抽取成类似“框架”那样的静态数据结构，是不够的。（当然不否认有些类似亲属关系那样的信息相对是静态的，但也不是绝对的。一个人有好几任妻子的情况，用一个静态的槽就难以描述。）所以，接下来的问题就是：时间（时间顺序）的识别和推理；事件同一性的识别和推理。

时间（时间顺序）的识别和推理，是指包含一个人物姓名的语段所叙述的内容在什么时间发生，是通过多种线索来体现的，需要把相关的时间或时序识别出来。在这些线索中，直接指出时间表达式（包括“1976年”这种绝对的方式和“第二天”这种相对的方式）是一种可能；通过生活在同时代的人物和其他命名实体的名称引用也是一种可能；更多的情形是，我们无法确定该人物所参与的一个活动的具体时间（比如，精确到年、月、日等等），但是可以确定这个活动与另外一个活动的相对时间顺序。这样，最后挖掘出来的，是一个带有时间指示和时间顺序标记的语段集合。特别值得指出的是，既然人物和其他命名实体在时间上有互相参照的可能，这里面就有一个以绝对的时间指示作为初始知识，按照所提及的相对时间顺序互相参照、相互进行时间顺序定位、逐步扩大关于相关人物的时间线索知识直至最后推理饱和的算法问题，它天然地要求把许多人物（也包括其他可能的命名实体）的追踪档案关联起来进行迭代推理。网上有好事者对“黄蓉到底是比郭靖小还是比郭靖大”进行推理[阎大为 2004]，用的其实就是这个方法。

事件同一性的识别和推理，指的是不同的文档资料从不同视角、不同阶段切入有关一个人物的活动的叙述，有可能说的是一个事件背景下的事情，应该将之归纳到一个线索当中去。不同的事件之间可能存在着千丝万缕的联系，事件本身也会发生演变，应当对其加以分析和辨别。

1.3.4 人物追踪的技术基础

作为一项面向应用的研究，人物追踪需要多方面的自然语言处理、信息检索以及模式推理技术的支撑。

从大的方面来看，比较相关的研究领域包括文本分类、聚类，信息抽取，模式推理等。从细处着眼，人物追踪依赖于一系列相关技术的解决，这些技术构成了人物追踪的基础。概括起来，人物追踪的技术基础主要包括：人名/非人名的区分、人物属性的抽取、同名人物的判别、时间和时序的确定、事件信息的组织以及地域的确定、背景事件的确定，甚至职务级别高低的确定等等。

无论哪一个问题解决好，都能对人物追踪的实现起到促进作用。

1.3.5 人物追踪的系统流程

按照对文档资料的处理流程，人物追踪可大致分为以下四个部分：数据预处理、人物模型的同一性判别与动态填充、人物事件信息的组织和若干细粒度挖掘任务的处理。各部分的关系如图 1.8 所示。本文在后面的章节会对各个部分的内容进行详细的介绍。

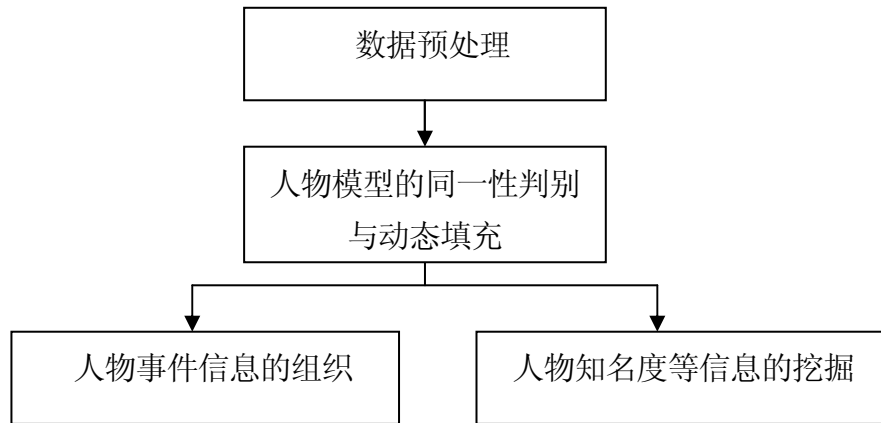


图 1.8 人物追踪的系统流程

1.4 人物追踪的意义

人物是抽象对象的典型代表，对其进行挖掘代表了文本挖掘技术的最新进展和新的增长点。人物追踪要解决的人物相关信息的“同一性判断问题”、“数据融合问题”和“跟踪问题”，将为与人物处于同一抽象层次的其它对象（例如事件、地点、机构、音乐等）的挖掘提供技术上的参考，也将为抽象对象挖掘本身凝练理论和方法。除了学术上的意义，人物追踪的研究还可以产生以下一些直接的应用：

1、人物搜索引擎：能够对互联网网页进行大规模的处理，推出一种全新的信息服务方式——人物搜索，该搜索引擎不仅能把同名人物的网页区分开，对每个人物还能够提供完整的信息，便于对人物进行全面的把握。

2、特定人物的跟踪：便于国家信息安全部门对某一敏感人物进行持续的跟踪。只要提供一小部分该人物的信息，就能够跟踪其参与的主要事件、言论及相关背景，并能将该人物以前的经历进行有效整理。

3、流行人物的发现：适用于信息内容提供商。通过对网上各个人物的分析、判别，能够发现一段时间内在互联网上比较流行的人物。把这些信息提供给用户，方便用户对网上新出现的名人的把握。

1.5 本文的组织结构

本文针对信息增长的大背景，结合业务模式和技术发展的内在逻辑，提出了“人物追踪”的概念。在第一章中给出了“人物”和“人物追踪”的内涵和外延，提出了“人物模型”的文本表示方法，并对其进行了形式化的定义。随后，给出了人物追踪要解决的主要问题，相关的技术基础，处理文档资料的系统流程。最后，阐述了人物追踪在学术上和应用上的意义。

第二章简单介绍了人物追踪涉及到的主要技术，对每种技术的理论和方法做了总结性的综述。

按照人物追踪的系统流程，第三章给出了人物追踪数据预处理的方法。处理的文档主要是汉语的新闻网页。重点研究了网页解析和人物属性抽取的方法。通过数据预处理，形成了人物模型的片断信息。

第四章研究了人物模型的同一性判断方法和人物模型的数据融合方法。主要是处理人物模型片断的信息，不仅要把不同人物的信息区分开，还要把同一人物的信息进行整合，形成完整的人物生平履历。

针对同一人物的活动报道，第五章研究了人物活动事件的组织方法。以标准的国际评测为平台，重点研究了大规模宏观粒度事件的组织，即以文档或段落为基本的处理单位，不涉及具体的事件角色框架。不仅研究了宏观粒度事件的识别技术，还研究了事件之间关系的组织方法。

第六章探讨了人物追踪若干细粒度挖掘任务的处理方法，重点探讨了人物知名度变化的判断方法。

结合前面几章的技术研究，第七章给出了人物搜索引擎的设计方法。设计了其体系结构，并给出了人物信息的索引结构和相关的界面图。

最后，对整个论文进行了总结，并给出了将来研究的方向。

第二章 人物追踪的相关技术综述

人物追踪是一项面向应用的研究，它的最终解决需要多方面文本处理技术的支撑，有些技术已经成熟，有些技术需要改进，还有的技术才刚刚起步。其中，跟人物追踪关系比较大的研究领域包括：网页解析，信息抽取，文本聚类，话题识别与跟踪等。

本章将对人物追踪的相关领域做一介绍，内容组织如下：首先介绍了网页解析相关的工作；然后介绍信息抽取方面的研究；在第三节介绍文本聚类的相关研究；随后，介绍话题识别与跟踪的研究；最后小结本章。

2.1 网页解析

对于人物追踪来说，网页解析的作用是把蕴含在半结构化信息中的有用内容提取出来，排除掉广告、菜单等无关信息的干扰。由于 Html 语言是一个标记性语言，很多标志都互相嵌套。为了更好的描述 Html 文件，一般都采用 DOM (Document Object Model) 树来表示 Html 文件的内部结构。然而，DOM 的目的在于揭示页面显示的结构，它往往对页面的内容结构揭示不足，因此仅仅依赖 DOM 树无法有效区分页面中不同的语义内容。基于 DOM 树，人们开展了一系列研究。

2.1.1 基于 HTML 标记分布规律的解析方法

网页中 HTML 标记有一定的使用习惯和规则，尤其是在各条记录的边界处。各条记录 and 记录边界处出现的标记往往呈现一定的规律，这些规律可帮助确定出记录的边界。

为此，人们利用这种信息，从格式较为简单的诸如“大学教员论文”类网页中抽取各个记录的条目。这种网页一般具有以下特征：(1) 网页中只有一个板块，含有多条记录；(2) 每个记录间至少有一个记录分隔符。针对网页的特点，可采用简单的启发式规则来做。

例如：[Embley 1999]采用了五个启发式规则：(1) 把出现次数最多的“候选”分隔符作为记录的边界；(2) 把从训练集中学习到的边界分隔符作为记录的边界；(3) 把所夹字符数标准偏差小的“候选”分隔符作为记录的边界；(4) 把同现次数多的两个或多个标记处作为记录的边界；(5) 根据所处理问题本体内的关键字段来确立记录的边界。[朱明 2000]在处理这类多记录网页时，仅采用一条启发知识：每条记录的格式是基本相同的。然后，利用这个启发知识从网页中分析出记录的格式，进而确定出记录的边界。

单纯利用 HTML 标记的分布规律对页面进行解析，在处理格式简单的诸如“论文网页”时比较有效；然而，由于基本启发知识的限制，在应对格式复杂或缺少共同表示特征的 Web 网页时就显得力不从心。

2.1.2 基于 HTML 标记间的关系的解析方法

HTML 语言中的标记不是独立的，它们之间存在着一定的层次地位关系；而且，每个标记对页面产生的作用也不同：有的只对视觉特征起作用；有的只对结构化显示起作用；还有的二者兼有之。

因为 HTML 语言主要是为了定义数据在页面中如何显示而设计的，大多数 Html 标志并不对 Html 文件的结构化显示产生帮助。因此，这种想法的初衷就是把对结构化显示没有帮助的标志去掉，然后精细分析剩下的标志，力图把剩下的标志构成的树转变为带有语义特征的 XML 树的形式。

一般地，根据标记是否对 HTML 文件的结构化显示产生作用，可以把标记分为两类。图 2.1 列举了对结构化显示产生作用的标记，并给出了标记间的优先级排序。

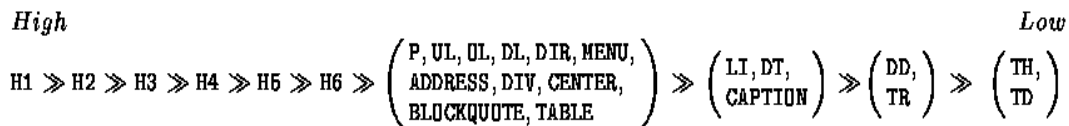


图 2.1 HTML 结构化显示标记关系图

图中表示，符号“ $\gg$ ”左边的标记在结构层次上的重要度比右边的标记大。[Lim 2000]就利用 HTML 标记间的这种关系，只分析图 2.1 中列举的这些标记，力图通过一系列具体的整合规则，把这些标记构成的结构树变换、整合为语义树的形式。例如，把图 2.2a 中的树转变为图 2.2b 所示的语义树。

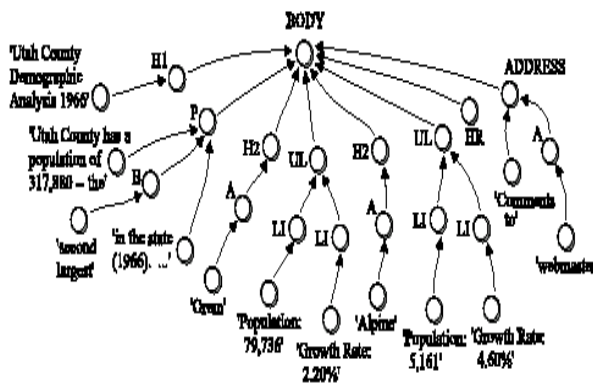


图 2.2a 原始的标记关系树

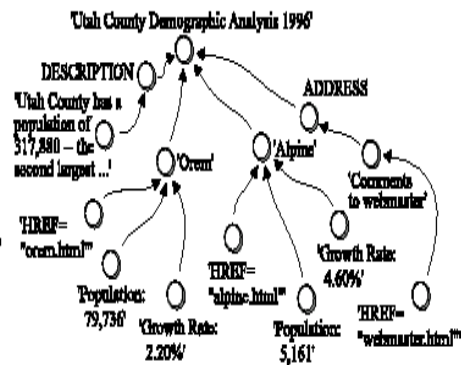


图 2.2b 转化后得到的语义树

如果单纯利用 HTML 标记间的关系对 HTML 文件进行解析，对于特定领域的结构简单规范的网页来说，会有比较好的效果；然而，现今流行的网页结构往往是复杂而不规则的，基于精细分析 HTML 标记的方法就显得很脆弱，无法正确整合文档的结构。因此，这种思想适合与其它方法结合起来使用。

2.1.3 基于视觉特征的解析方法

在 Html 文件中，很多标志的使用不仅是为了内容的组织，还可用来表示页面的外观，例如“Table”、“P”等。为了把页面分割成一个个的语义块，页面显示的外观提供了大量的线索。一般来说，网页设计者要有效地组织页面的布局使它容易阅读。因此，语义相联系的内容通常组织在一起，整个页面用一些视觉特征分隔成不同的区域。这些视觉特征包括不同的行、空白区域、不同的背景图象、字体不同的大小和颜色等。所以，充分利用页面的视觉特征会对页面的语义分块有很大的帮助。

基于页面视觉特征的分块解析方法以前研究的较少，只是近年来才有人提出。比较典型的是[Yu 2003]提出的基于视觉特征的网页分块方法 VIPS，并应用在了 TREC 2003 的评测中，取得了较好的效果[Wen 2003]。图 2.3 是 VIPS 算法的流程图。

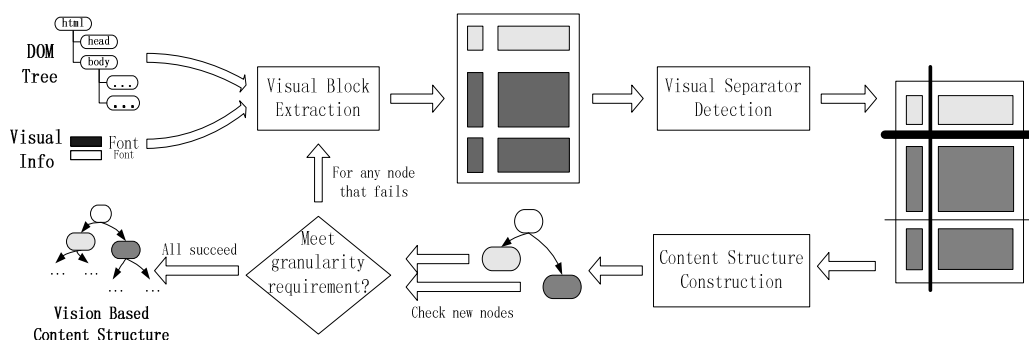


图 2.3 VIPS 算法流程图

VIPS 算法分块所用的视觉知识包括：

- 标志信息：有些标志例如“HR”经常用来分隔不同的主题，因此如果一个 DOM 结点含有这样的标志，可以考虑把这个结点分隔开。
- 颜色信息：如果一个 DOM 结点的背景颜色与它的一个孩子节点的背景颜色不同，就把这个结点分隔开。
- 文本信息：如果一个 DOM 结点的大部分孩子结点都是文本结点（例如，周围没有 Tag 标志），就不分隔这个结点。
- 大小信息：如果一个 DOM 结点的各孩子的尺寸标准差大于一个阈值，就倾向于把这个结点分隔开。

VIPS 确定各个视觉块的层次地位所用的知识包括：

- 距离模式：相邻两个块间的距离越大，这两个块的分隔就越大。
- 标志模式：如果两个块正好位于一些 Tag 标志例如“hr”的两侧，那么这两个视觉块的间隔就比较大。
- 字体模式：如果两个块间的字体大小、粗细、颜色等越明显，这两个块的间隔就越大。
- 颜色模式：如果两个块间的背景颜色不同，则两个块的间隔就比较大。

利用视觉特征对页面进行解析，能在一定程度上满足复杂页面对算法的要求；但由于视觉特征的复杂性，运用的启发知识往往较为模糊，需要人工来不断的总结调整规则，需要的规则往往非常多，一条规则的加入会对已经成功分析的网页产生影响，因此，如何保证规则集的一致性是一大难点。

2.1.4 基于 TABLE 标记布局特性的解析方法

随着互联网的深入发展，各个网站为了吸引浏览者的注意力，都竭力使设计出的页面具有特色，一系列网页开发工具也提供了这方面的便利，这就造成了互联网上的网页格式越来越复杂，共同特征越来越少，而且这种趋势将不断的发展下去。如今，每个页面几乎都包含网站分类栏、各个主题的导航栏、正文栏、相关链接栏、版权声明栏和广告信息栏。然而，由于标记“table”有良好的布局特性，格式复杂的页面往往都采用“table”标记来进行页面格式的布局。因此，可利用“table”标记对页面进行解析。

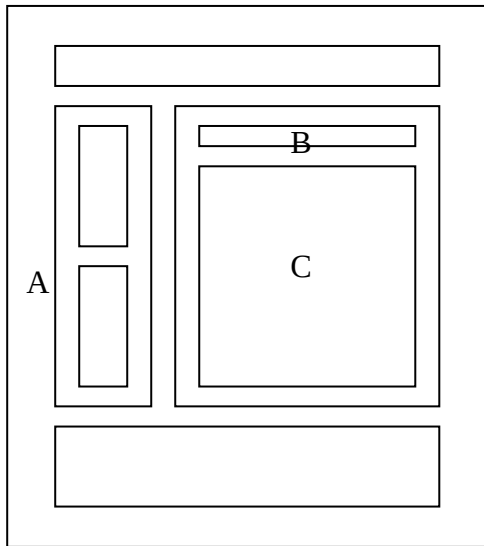


图 2.4 网页的 TABLE 布局示例图

如图 2.4 所示，整个页面用一个大的 table A 来布局，它的每一个表单项“td”里边又嵌套着不同的 table 等；这样不断嵌套，直至最底层的 table 里面包含着具体的记录项。[Lin 2002]就利用标记“table”来对页面进行解析，把页面分割成不同的块。

虽然思路简单，但用 TABLE 标记进行挖掘，对网页开发工具制作出来的格式特别复杂的页面来说，是比较有效的，它往往能胜任其它方法所应付不了的复杂网页。所以，在处理门户网站中的网页时，这种思想是比较可取的。

2.2 信息抽取¹

信息抽取(Information Extraction)是从一段文本中抽取指定的一类信息（事件、事实）并将其（形成结构化的数据）填入一个数据库中供用户查询使用的过程。例如，从一篇关于自然灾害的新闻报道中摘录出灾害的类型、时间、地点、人员伤亡、经济损失、救援情况等；或者从产品发布的新闻语料中提取某类产品的各种感兴趣的指标等。这些处理主要是以词和词组为对象来进行的。

2.2.1 信息抽取的发展历史

从自然语言文本中获取结构化信息的研究最早开始于 20 世纪 60 年代中期，这被看

¹ 这里只探讨针对自由文本中信息抽取的相关技术

作是信息抽取技术的初始研究，它以两个长期的、研究性的自然语言处理项目为代表[Gaizauskas 1997][李保利 2003]。

一个是美国纽约大学开展的 Linguistic String 项目[Sager 1981]，它开始于 60 年代中期并一直延续到 80 年代。该项目的主要研究内容是建立一个大规模的英语计算语法，与之相关的应用是从医疗领域的 X 光报告和医院出院记录中抽取信息格式。

另一个相关的长期项目是由耶鲁大学在 20 世纪 70 年代设计实现的 FRUMP 系统[Dejong 1982]。该系统从新闻报道中抽取信息，内容涉及地震、工人罢工等很多领域或场景。该系统采用了期望驱动（top-down，脚本）与数据驱动（bottom-up，输入文本）相结合的处理方法，被后来的许多信息抽取系统采用。

从 20 世纪 80 年代末开始，信息抽取研究蓬勃开展起来，这主要得益于消息理解系列会议（MUC，Message Understanding Conference）的召开。从 1987 年开始到 1998 年，MUC 会议共举行了七届，它由美国国防高级研究计划委员会（DARPA，the Defense Advanced Research Projects Agency）资助。

按照信息抽取领域顶级国际会议 MUC 的研讨结果，信息抽取任务分为如下五种类型[贾自艳 2004]：

(1) 命名实体识别（Named Entity Recognition）：确定人物、地点、组织、时间、金钱数量的所有名字。其性能可以达到人的水准，MUC 的内部数据表明，目前最好的效果：English: 95%，Spanish: 93.04%，Japanese: 92.12%，Chinese: 84.51%。

(2) 共指关系识别（Coreference Resolution）：识别文本实体（NE 识别确定的实体和对这些实体的指代）之间的关系。目前准确率为 50—60%，而普通人的准确率在 80% 左右。

(3) 模板元素构建（Template Element Construction）：定位和分类文档中的实体，并且将描述性的信息和实体相联系。MUC-7 最好的系统达到 80% 的准确率，而人类是 93%。

(4) 模板关系构建（Template Relation Construction）：模板文档实体之间确定一小部分可能有的关系，例如一个人和公司之间的雇员关系，两个人之间的家庭关系，两个公司之间的附属关系。

(5) 情节模板产生（Scenario Template Production）：将 TE 实体结合在一起，成为事件和关系描述。MUC 最好的系统达到 60% 的准确率左右，人类得分为 80+%，准确率和召回率之间可以进一步平衡。

从五种任务来看，“命名实体识别”和“共指关系识别”是信息抽取的必要步骤和技术基础，“模板元素构建”、“模板关系构建”和“情节模板产生”则是直接面向应用的几个信息抽取任务。

目前，除强烈的应用需求外，正在推动信息抽取研究进一步发展的动力主要来自美国国家标准技术研究所(NIST)组织的自动内容抽取(ACE, Automatic Content Extraction)评测会议[ACE]。这项评测的主要内容是自动抽取新闻语料中出现的实体、关系、事件

等内容，即对新闻语料中实体、关系、事件的识别与描述。它主要有两大任务：实体识别与跟踪(EDT, Entity Detection and Tracking)、关系识别与描述(RDC, Relation Detection and Characterization)。

中文信息抽取方面的研究起步较晚，主要的研究工作集中在对中文命名实体的识别方面，在设计实现完整的中文信息抽取系统方面还处在探索阶段[Chen 1998][Yu 1998][Zhang 2000]。

2.2.2 信息抽取系统的基本流程

一般来说，信息抽取系统主要由两大部分组成。第一步，系统先对文档进行局部分析，从文本中抽取出想要的独立的结果片断；第二步，系统通过篇章的分析和推理，把一个个独立的结果片断进行整合，得出最终的结果，转换成需要的输出格式[Grishman 1997]。图 2.5 是典型的信息抽取系统的流程图。

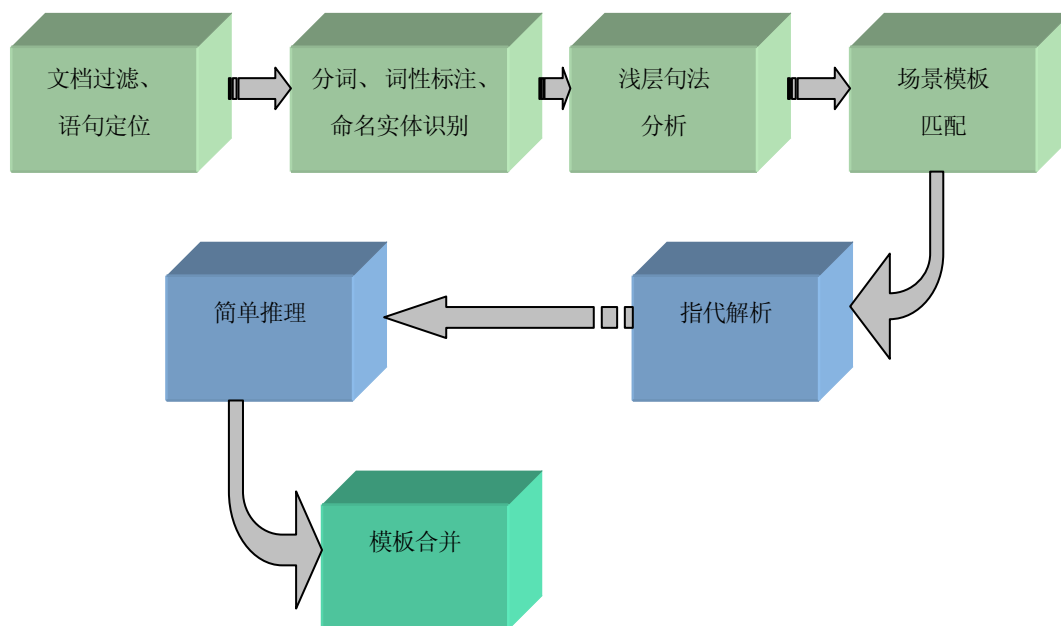


图 2.5 信息抽取系统的典型流程图

典型的信息抽取系统通常包括：

- ① 预处理过程，目的在于过滤掉与抽取目标不相干的文本，直接定位到相关语句上；
- ② 通过词法分析和标引，识别所有与抽取目标相关的词汇，这个过程通常包括识别特定的名词短语（人名、机构名、产品名、事件、地点等）和动词短语（事件描述、事实陈述），这需要使用合适的词典、构词规则库等知识库的支持；
- ③ 句法和语义分析，它只应用于所有包含了关键词的句子的集合，对每个句子的分析结果近似于该句子的语义框架表示（此框架的定义与抽取目标密切相关）；

④ 用一组信息模式描述感兴趣的信息。信息模式通常可表示为简单的一个句式，例如 <公司名> “推出” <产品名>。系统可以针对某一领域的信息特征预定义好一系列的信息模式，存放在模式库中供用户选用；使用模式匹配方法识别指定的信息（即找出信

息模式的各个部分)；

⑤ 进行上下文关联、指代、引用等分析；

⑥ 进行结果的简单推理，确定信息的最终形式。可能需要篇章分析和推理规则库的支持。

⑦ 输出结果（例如生成一个关系数据库或给出自然语句陈述等）。

在上述七个步骤中，第一到第四步属于文本局部分析范畴，第五到第六步属于文本篇章分析范畴。

2.2.3 信息抽取的发展方向

信息抽取经过二十多年尤其是最近十多年的发展，已经成为自然语言处理领域一个重要的分支，其应用不仅满足了人们对于特定的、更细粒度的信息的需求，还极大地推动了自然语言处理、信息检索、文本挖掘等相关技术的发展。但是，到目前为止，信息抽取并未象信息检索技术那样被广泛应用，其原因主要在于现有的信息抽取系统的性能不高，可移植性不好[Grishman 2000]。因此，今后信息抽取的研究将紧紧围绕如何克服和解决这两个问题展开，不断提高信息抽取的性能，增强可移植能力。

2.2.3.1 改善信息抽取的可移植性

由于不同的信息抽取任务需要不同的领域相关知识库的支持，因而信息抽取系统在本质上是领域相关的。为了改善信息抽取系统的可移植性，人们尝试了以下方法和策略[姜吉发 2004]。

① 把领域相关的知识库和模式匹配功能相分离

影响信息抽取可移植性的最大因素就是信息抽取依赖特定领域的知识。在信息抽取的场景模板匹配阶段，系统需要根据预定的知识来识别特定的信息。这些知识都是领域相关的，可以手工给定，也可以通过机器学习的方法训练出来。这样，系统就主要由两大功能模块组成：模式获取模块和模式匹配模块。当把知识库与模式匹配模块分离之后，当系统需要从一个新的领域中抽取信息时，只需修改相应的知识库，而不必修改系统的其它功能。这就大大地改善了系统的可移植性。

② 采用机器学习方法自动获取信息抽取模式

前面提到，信息抽取所需的领域知识可以通过手工给定，也可通过机器学习方法训练出来。手工给定知识费时费力，而且需要领域专家的参与，所以造成系统可移植性不高。为此，人们采用机器学习的方法来指导知识的获取，并不断改进模式学习的效果来减少用户的工作量，降低对用户的技能要求。

最早的模式学习方法需要用户手工标注规模较大的语料[Riloff 1993]；而标注大规模的语料费时费力，于是研究者就加以改进，只需要用户将训练语料分为相关和不相关的两类，然后就能自动的从中学习出相关的模式[Riloff 1998]；进一步的研究发现，即使要求用户将训练语料分为相关和不相关的两类，也并非易事。因而研究者采用“滚雪球”的方法，只要求用户提供几个可以轻易想到的有代表性的模式，相应的模式学习方法就

可以从一个未经分类的文档集合中学出更多的模式并同时完成对文档的相关性分类[Riloff 1999, Yangarber 2000]。

③ 设计跨领域的模式表达方式

针对从自由文本中进行英文事件的抽取，人们设计了各种各样的模式表达方式[Muslea 1999, Kim&Moldovan 1995]。无论这些模式表达方式如何不同，它们都充分利用了语法信息和语义信息的概括约束作用，而且当系统转到不同的领域时，这些模式的表达方式是固定不变的。

④ 设计图形用户界面（GUIs）

用户通过 GUIs 可以方便快捷的配置系统所需的领域知识，从而便于系统从一个领域转向另一个领域。如 Roman Yangarber 设计的 Proteus 系统的一个领域知识配置界面 PET[Yangarber 2001]。

⑤ 使用领域无关的概念层次知识库的支持

一般的信息抽取模式都利用语法和语义信息来进行概括约束。其中，语义信息的概括约束作用是通过将模式中的某些概念元素用它们的上位概念代替来完成的。研究者把语义约束部分采用领域无关的概念层次知识库如 WordNet 等来支持，这样就提高了领域的通用性[Chai 1997][Chai 1998]。

2.2.3.2 提高信息抽取的性能

信息抽取所获取知识的途径和所抽取出的信息多是来自于自然语言文本。然而，由于自然语言中存在着“长尾综合效应”，即绝大多数事实采用经常出现的、非常少量的表达方式来描述，而剩余的事实却需要大量的、不经常出现的表达方式才能覆盖，这就使得信息抽取系统达到中等水平的性能很容易，然而，要想进一步提高就非常困难。在 MUC 举行的多次测试中，很少有 F 值超过 0.60 的系统，且排名靠前的几个系统性能很接近[MUC-6 1995]。

要想使信息抽取性能进一步提高，就需要使信息抽取的每一个流程的效果都得到改善，如采用高效的句法分析模块，提高指代消解和篇章推理的效果，获取更加可靠的领域知识等。

出于现实的考虑，目前的信息抽取系统多采用“部分句法分析”，原因是部分句法分析能达到可用的效果，且可以分析大部分语言现象。然而，我们也要清醒地认识到，要想很好地分析全部的语言现象，我们必须依赖“完全句法分析”的全局约束规则来消除局部歧义。而完全句法分析的鲁棒性和时空开销目前还是阻碍其实际应用的头等问题。

用户关心的事件和关系往往散布于文本的不同位置，其中涉及到的实体通常可以有多种不同的表达方式，并且还有许多事实信息隐含于文本之中。因此，篇章分析、推理能力对信息抽取系统来说是必不可少的。大多数信息抽取系统只识别和保存与需求相关的文本片段，从中抽取零碎的信息。在这个过程中很可能把用来区分不同事件、不同实体的关键信息给遗漏了[李保利 2003]。因此，研究如何提高在有限信息条件下的篇章分析和推理的性能就十分必要。

不管是手工编写知识库还是系统自动学习知识，得到的领域知识往往很难达到比较高的语言覆盖面。因此，知识获取就成为信息抽取系统提高性能的一个瓶颈，且与提高移植性的方法形成了一定的矛盾。如何在保证能够快速获取知识的同时保证所得知识的质量，也是需要解决的一大问题。

2.3 文本聚类

聚类（Clustering）是将数据对象按照一定的标准分成若干个簇（Cluster），使得同一个簇中的对象之间相似度比较大，不同簇之间的对象相似度比较小。距离是经常采用的度量方式。聚类分析是一种无指导的机器学习方法，在机器学习、统计分析、模式识别、数据挖掘、生物学等许多领域得到了广泛的研究与应用。

近年来，在数据挖掘领域中提出了多种新的算法，这些算法主要处理的数据对象为空间数据库、多媒体数据库、图像等，这些算法充实了聚类分析算法的种类。与此同时，随着信息技术的应用与发展、万维网的普及，电子形式存在的文档急剧增多，对于我们的认知与决策带来了很大的压力，因此利用聚类分析技术对于大量的文档进行组织和分析是一项很有意义的工作。

文档的聚类分析同一般的聚类分析类似，往往包括如下 5 个步骤[Jain 1988]：① 模式表示，往往包括特征抽取和特征选择，把数据对象表示成适合于算法可计算的形式；② 根据领域知识定义模式之间的距离测度公式；③ 聚类或者分组；④ 数据抽象表达（如果需要）；⑤ 评价输出结果（如果需要）。本章主要对前三个步骤进行介绍。

2.3.1 文本的表示

文本聚类分析首先要考虑的是文本表示问题，该问题指如何从一个电子文本的符号和文字中抽取出特征，通过这些特征来表示相应的文本，利用这些特征数据进行聚类分析。在实际的文本聚类分析研究中，由于所处理的对象不同，文本的表示方法有所区别，但大致还是利用词（字、词组等）的词频信息建立文本向量，通过文本向量来进行聚类分析。

文本的表示是文本聚类的第一步，该步骤的变化很多，但是该步骤的选择正确与否、是否符合待解决问题的需要对于聚类的效果影响非常关键。不同的文本表示形式包括可以选择不同的表示模型，如向量空间模型、概率模型、语言模型等；可以选择不同的特征选择方法；可以利用 LSI 等不同的降维策略等。对于一个具体的应用而言，这个步骤的选择是比较消耗时间的。

最通用的数据表示方法是向量表示法。向量空间模型将文档表示成一个向量，向量的每一维表示一个特征，该特征可以是一个字、一个词、一个 n-gram 或某个复杂的结构。通常情况下，利用向量空间模型表示文档时，需要对于文档进行切分（中文分词、英文通过词的分界符识别单词）、停用词处理、英文词的词形还原或者提取词干(Stemming)，经过这些步骤的处理，基本上可以得到一系列词并将其作为文档的特征。所有的这些词

构成一个“空间”，每个词对应着空间中的一维。每个文档可以用文档中的词来表示，这些词及其权重构成一个向量。向量的每个维度描述对象的一个特征，每一个特征的重要度通常采用 TF*IDF 的计算方法来衡量[Salton 1975]，更多的工作是在参数调节上来获取更好的效果。在著名的信息检索系统 Smart 中提出过一套词权重计算的命名，该命名中综合了 TF*IDF 中的多种变化，把词的权重计算归结为三个组成部分：词频 TF、反向文档频率 IDF 和基于文档长度的规格化因子。每个部分都有不同的计算方法，这些不同的变化构成了文档的不同表示方式。常用的一个表示公式有(Smart 命名为 ltc) [Yang 1999]:

$$w(t,d) = \frac{1 + \log_2(tf(t,d)) * \log_2(N/n_t)}{|d|}$$

其中 $w(t,d)$ 表示词 t 在文档 d 中的权重； $tf(t,d)$ 表示词在文档 d 中的频率； n_t 表示在整个语料中出现词 t 的文档的数目； N 表示整个语料中的所有文档数目； $|d|$ 表示文档向量的长度。

文档聚类分析中的文档表示是一个关键的研究点，在以往的研究中，有关学者尝试过不同的做法来完成。比如考虑基于概念的聚类[Hotho 2002][卜东波 2002]，而不是基于简单的词的聚类；考虑利用文档中每个词的互信息向量[Pantel 2002]来聚类，而不是权重向量等等；也有从效率方面来考虑，用两次聚类，先用一字词聚类，再用二字词进一步聚类提高聚类精度[李振星 2003]。这些都是值得研究的方面，需要在具体的应用中去考虑不同的做法及其效果。

2.3.2 距离的度量

聚类分析是基于这样一个认识：每一类别内的对象之间较为相似，而两个类别的对象之间应该具有较大的差异。这两个特点越突出，聚类的效果越好。然而，必须采用一定的标准来衡量相似度或距离。

在文本聚类中，除了将文本进行表示外，还必须对于文本进行相似度度量。不同的距离度量方法会出现不同的聚类结果。文本聚类主要区分三个距离度量（或者相似度量）：

- (1) 文本与文本之间的相似度度量；
- (2) 文本与文本簇之间的相似度度量；
- (3) 文本簇与文本簇之间的相似度度量。

2.3.2.1 文本与文本相似度的度量

在向量空间模型与 OKAPI 模型中，文本与文本之间的相似度往往采用余弦距离，定义如下：

$$Sim(X,Y) = Cos(X,Y) = \frac{\sum_i x_i * y_i}{\sqrt{(\sum_i x_i^2) * (\sum_i y_i^2)}}$$

而在语言模型中，文档与文档之间的距离可以采用 KL 距离来计算：

$$Score(D, Q) = -KLD(\theta_Q \parallel \theta_D) = \sum_w p(w | \theta_Q) \log \frac{p(w | \theta_D)}{p(w | \theta_Q)}$$

对于以向量表示的文档，除了表示模型本身的距离度量方法之外，在传统的聚类分析，还存在多种其他距离度量方法，比如明考斯基距离，它是欧式距离和曼哈顿距离的概化，定义如下：

$$d_{ij} = \left[\sum_{k=1}^n |x_{ik} - x_{jk}|^r \right]^{1/r}$$

此外，还有马氏(Mahalanosis)距离与兰氏(Lance)距离表示[张尧庭 1982]等。

2.3.2.2 文本簇与文本簇之间的相似度度量

在聚类分析中，有时需要衡量文档簇和文档簇之间的相似度，从而选择簇与簇之间的合并或者拆分。文档簇与文档簇之间的相似度也存在多种定义，常见的有最小距离（Single link）、最大距离（Complete link）、平均距离（Average link）、质心法、离差平方和、代表点等。下面列出了常见的簇 C_1 , C_2 之间相似度的度量方法。

(1) 最小距离

使用两类间的最近两点的距离来描述两类间的相似程度。

$$D(C_1, C_2) = \min \{ D(X, Y) | X \in C_1, Y \in C_2 \}$$

(2) 最大距离

使用两类间的最远两点的距离来描述两类间的相似程度。

$$D(C_1, C_2) = \max \{ D(X, Y) | X \in C_1, Y \in C_2 \}$$

(3) 平均距离

使用两类间两两样本点距离的平均值来描述两类间的相似程度。

$$D(C_1, C_2) = \frac{1}{|C_1| * |C_2|} * \sum_{X \in C_1} \sum_{Y \in C_2} D(X, Y)$$

(4) 质心法

使用两类质心之间的距离来描述类间相似性。

$$D(C_1, C_2) = D(\bar{X}, \bar{Y})$$

(5) 离差平方和法

使用簇内的方差来度量。正确的聚类应当使得簇内方差尽量小，而两簇方差尽量大。

簇内方差定义为：

$$S(C) = \sum_{X \in C} (X - \bar{X})^T * (X - \bar{X})$$

因此，两个簇间的距离为：

$$D(C_1, C_2) = S(C_1 \cup C_2) - S(C_1) - S(C_2)$$

2.3.2.3 文本与文本簇相似度的度量

文本与文本簇相似度的度量一般仿照依据 2.3.2.1 与 2.3.2.2 中的度量公式来计算。假如把要计算距离的文本和文本簇中的文本都看成是一个只有一个文本的文本簇，那么可以采用 2.3.2.2 中的度量公式来度量；如果把两者中的文本簇都用其质心来表示，那么又可以当成两个文本的距离度量，采用 2.3.2.1 中公式度量。

2.3.3 聚类的算法体系

聚类分析算法体系的划分可以基于不同的标准，比如凝聚的还是分裂的、增量的还是非增量的、确定的还是随机的、Hard 还是 Fuzzy 等等，因此聚类算法体系的划分在文献中并不统一。在文献中可以找到大量的与聚类分析有关的文章，散布于各个研究领域之中。在[Berkhin 2002][Jain 1999][Han 2001]中我们可以看到关于聚类分析算法的综述性介绍。我们这里采用[Han 2001]的划分方法，把聚类算法分为层次聚类、划分聚类、基于网格的方法、基于密度的方法、基于模型的方法来介绍。

2.3.3.1 层次的方法

层次聚类将文本对象组织成为一个树，每个叶子节点是一个簇，该树称为一个聚类谱系图。根据树形成是自底向上还是自顶向下的可以划分为凝聚式和分裂式层次聚类。层次聚类算法复杂度高，但被认为效果比较好，可以根据需要获得不同粒度的类。该类别算法中一旦一个合并或者分裂被执行就不能再修改[Han 2001]。

(1) 凝聚式层次聚类

算法流程：

Step1: 将所有的点各自单独形成一个簇；

Step2: 从现有所有的簇中选择最近(或者最相似)的两个簇，进行合并；

Step3: 如果只剩下一个簇或者达到终止条件(比如达到需要的簇的数目)，聚类结束，否则返回 Step2。

在凝聚式层次聚类分析中，Step2 寻找最相近的两个簇需要用到簇与簇之间的距离度量公式，例如 Single-link、Average-link 和 Complete-link 等。这三类凝聚式层次聚类算法在文档聚类中应用比较多，但是时空效率并不高，制约了其在大规模聚类分析中的应用，往往应用于小规模聚类分析或者不考虑效率而效果要求高的聚类分析，也应用于复合聚类分析算法（聚类算法中采用多种聚类思想）中作为一个组成部分。

(2) 分裂式层次聚类

分裂式层次聚类分析与凝聚式层次聚类分析恰好是一个反向的过程。算法如下：

Step1: 将所有的点形成一个簇；

Step2: 通过某种策略从现有所有的簇中选择某个簇进行拆分，将该簇拆成 2 个簇；

Step3: 如果已经拆成 n 个簇(n 为所有文档数目)或者达到了终止条件则结束。

在分裂式层次聚类算法中，Step2 中用于选择分裂簇的策略会衍生出不同的算法以及不同的效果，一种最简单的方法就是选择最大的簇进行拆分，也可以根据一些聚类有效性分析的指标选择簇内部凝聚性最差的簇进行拆分。而根据拆分簇的方法的不同，也能够衍生出不同的算法。

(3) 其他层次聚类算法

目前在层次聚类研究中出现了一些结合其他聚类方法的多阶段聚类分析技术，比较有代表性的有：BIRCH (Balanced Iterative Reducing and Clustering using Hierarchy) [Zhang 1996]、CURE(Clustering Using REpresentatives) [Guha 1998]、Chameleon[Karypis 1999]等都是基于层次思想的组合算法。

BIRCH 是一个增量式动态聚类分析算法，通过数据结构 CF (Clustering Feature) 树对数据进行聚类。在 CF 树中，每个节点存储了该节点所代表的簇的特征表述 CF。算法顺序扫描数据库，中间结果保存在 CF 树中，第一次扫描得到聚类，第二次或者更多次扫描改进聚类的质量。

CURE 改变了簇的表示方法，不采用单个质心、单个代表对象或者全体所有对象来表示簇，而采用了固定数目的有代表性的数据点来代表簇。CURE 针对大型数据采用取样与划分的方法降低计算复杂度，特点是可以发现任意形状的簇和处理孤立点。

Chameleon 是一个两阶段的聚类分析算法。第一阶段：构建相似度矩阵，从矩阵中再构建 K 最近邻图，然后对于图进行划分得到小的簇；第二阶段：对小簇进行动态合并，合并时采用互连性与近似度来度量簇与簇之间的相似性。因此该算法是一个在层次聚类分析中采用动态模型的聚类算法，算法的复杂度并不低。计算 K 最近邻图时低维数据采用 K -d 树结构进行优化，平均复杂度为 $O(\log n)$ ，而对于高维空间至少需要 $O(n^2)$ 。

2.3.3.2 划分的方法

基于划分的聚类分析不需要建立起类之间的层次结构，只需要将数据集进行划分，形成一个平面的类结构。比较典型的方法如下所示。

(1) K-Means 聚类

K-Means 聚类分析算法是一种简单效率高的聚类分析算法，在文档聚类分析以及数据流聚类分析中都受到较大的关注。K-Means 将聚类对象划分成为 K 个簇， K 是事先必须确定的参数，K-Means 中每个簇用一个中心表示，该中心定义为簇中对象的平均值。算法流程如下所示。

Step1: 初始化 K 个簇中心；

Step2: 对于每个数据对象，分别计算它与 K 个类中心的距离，并将该对象分入距离最小（相似度最大）的簇；

Step3: 重新计算 K 个簇的中心，中心为该簇内所有点的算术平均；

Step4: 如果簇变化不大或者满足某种退出条件（例如达到最大迭代次数、满足某种目标函数等），那么结束聚类，否则返回 Step2。

K-Means 聚类的缺点是必须事先给出 K 、对数据输入顺序敏感、不适于发现大小很大的簇、对于噪声和孤立点敏感等。在文本聚类中如果不进行特征选择，特征维度过高的话容易扎堆成几个大类和众多小类。

K-Means 也是变种非常多的一个聚类分析算法，Step1 中初始中心的选择、Step2 中最相似簇的选择、Step3 中簇中心的定义都有很多种变化，可以演化出不同的算法。Step1 一般可以采用随机选择 K 个点作为簇中心，也可以通过某种策略来选择（比如特征数目最多的 K 个对象）。Step2 与 Step3 一般是等所有的点都分配给簇后才更新簇中心，而对于文档聚类，这种做法则可以变成每分配一个点就更新中心。

(2) K-Medoids 聚类

K-Medoids 聚类是与 K-Means 类似的一种聚类分析方法，其主要区别在于 K-Means 使用所有点的均值来代表簇，而 K-Medoids 则采用类中某个数据对象来代表簇。PAM (Partitioning around Medoid) [Han 2002]是最早提出的 K-Medoids 算法之一。与标准的 K-Means 相比，PAM 通过反复计算随机选择的点与原有 K 个中心的交换代价来决定如何形成新的中心点。PAM 算法复杂度高，对于大数据集合没有好的可伸缩性。因此由 PAM 演化出了 CLARA (Clustering Large Applications) 以及 CLARANS (Clustering Large Application based upon Randomized Search) [Han 2002]用于大规模空间数据的聚类分析。CLARA 从原始对象随机采样得到 K 个中心点，可以进行多次采样得到更优的 K 个中心；而 CLARANS 比 CLARA 更进一步，每一次应用 PAM 时不是一个固定的采样样本。如果对于 K-Medoids 的簇中心重新定义，比如采用类似 CURE 的代表点的方式来代表一个簇，可以演化出新的算法。

(3) 基于样本最大距离的聚类

该算法的聚类思路是取距离尽可能远的样本（相似度尽可能小）作为中心，算法与首先选择哪个样本有关。对于 N 个样本集合 $Ds=\{D_1, D_2, \dots, D_N\}$ ，算法流程如下：

Step1: 从 Ds 中任取一个样本，例如 D_1 ，以 D_1 作为簇中心新建一个簇；

Step2: 在 Ds 中找与 D_1 最远的样本并以之为中心新建一个簇，从而形成两个簇，记

录该最远距离为 max ;

Step3: 对于未处理的其他所有对象 D_i , 计算该对象与已有的所有簇的最小距离 d_i ;

Step4: $d = \max\{d_i\}$, 假如 d 大于给定阈值 (可以为 max 的 p 倍, $1/2 \leq p < 1$), 那么新建一个以该对象为中心的簇, 返回 Step2; 假如无法再新建簇, 那么执行 Step5;

Step5: 把剩余的样本分配到离它最近的那个中心所属的簇。

该算法思路比较简洁、效率高, 可以考虑在文本聚类分析中使用, 但算法变化余地比较小, 受初始样本影响比较大。

(4) 最近邻聚类

最近邻聚类可以说是一种最简单的聚类分析, 主要思想是如果某个待处理的对象与现有的所有簇的最小距离小于给定阈值, 那么将该对象归入该簇, 否则将该对象作为一个新的簇。算法流程为:

Step1: 选择一个样本, 以该样本为中心建立一个新簇;

Step2: 取下一个要分析的样本, 如果没有样本需要处理, 则聚类结束;

Step3: 计算当前样本与所有簇的距离, 得到距离最小的簇及其对应的距离值 d ; 如果 d 小于阈值 T , 则将该样本分配给选中的簇, 更新簇的中心; 否则以该样本为中心新建一个簇; 返回 Step2。

最近邻聚类不需要事先决定结果类数, 适合于类内距离小, 类间距离大的情况。结果与样本的输入顺序有关, 也与阈值 T 有关。随着样本集的增大, 产生的结果类数可能越来越多, 处理的速度会越来越慢。

2.3.3.3 密度的方法

基于密度的聚类算法将簇看成是数据空间中被低密度区域分割开的高密度区域[Han 2001]。密度定义为单位体积内的点数, 簇的内部密度大。由于密度是一个局部概念, 这类算法又称为局部聚类 (Local Clustering); 基于密度的聚类算法通常只扫描一遍数据库, 所以又称为单遍扫描聚类 (Single Scan Clustering) [陈宁 2001]。基于密度的聚类中比较典型的算法有 DBScan (Desity-based Clustering)、OPTICS (Ordering Points to Identify the Clustering Structure) 与 DENCLUE (Density-based Clustering)。

对于空间中的一个对象, 如果在给定半径 Eps 的邻域中的对象个数大于给定的数值 $Minpts$, 该对象称为核心对象 (core point), 否则称为边界对象。由一个核心对象可达的所有对象构成一个簇。DBSCAN 在研究中得到比较多的关注, GDBSCAN (Generalized DBSCAN), DBCLASD (Distribution Based Clustering of Large Spatial Database), FDC (Fast Density-based Clustering) 都是对 DBSCAN 的进一步扩展和优化。

2.3.3.4 网格的方法

基于网格的聚类首先把数据空间划分为一定数目的单元, 以后所有的操作都针对这些单元, 通过对单元的操作来产生结果簇。这种方法的优点是处理速度快, 处理时间独立于数据对象的数目, 仅仅依靠于网格空间的维度。基于网格的代表性例子有 STING,

它利用存储在网格中的统计信息；WaveCluster，它利用小波变换来聚类对象；CLIQUE，属于高维空间中基于网格和密度的方法[Han 2001]。

2.2.3.5 模型的方法

基于模型的方法为每一个簇假定了一个模型，寻找数据对给定模型的最佳拟合。基于模型的方法有统计学方法 COBWEB 和自组织映射（SOM）神经网络方法等。其中自组织特征映射方法首先对文档降维，把特征维度降低到足够小，然后进行 SOM 聚类。该算法的意图是把 SOM 算法扩展到处理高维数据对象的领域，适合于用户对文档内容所属领域不太了解的情况。

与普通的聚类场景相比，文本聚类属于高维空间上的聚类分析。高维空间的数据经常呈现奇怪的特征，例如噪声的比例增加、网格单元的个数指数级增加、数据空间的稀疏性大、聚类的密度低等，这在数据挖掘中称为“维数祸根”[Han 2001]。许多聚类分析的算法，例如 DBSCAN、STING、WaveCluster，随着维数的增加，算法的空间和时间的复杂性往往增长得很快，所以很难实用化。对于传统的聚类算法来说，当应用在文本聚类的场景中时，除了要调整好参数以外，还要尽量避免高维特征所带来的影响。

2.4 话题识别与跟踪

话题识别与跟踪（Topic Detection and Tracking，简称TDT）是一项旨在依据事件对语言文本信息流进行组织利用的研究，也是为应对信息过载问题而提出的一项应用性研究。与传统的信息检索技术不同，话题识别与跟踪所关注的问题领域被称之为“基于事件的信息组织”[Allan 2002]，目前主要是对热点新闻话题进行组织。

2.4.1 话题识别与跟踪的背景与历史

在当今时代信息爆炸的情况下，一方面网络信息的规模急剧膨胀，另一方面信息又凌乱无序，有价值的信息湮没在大量冗余信息中，对其发现和管理变得越来越困难。为此，以话题为主线对信息进行组织引起了越来越多人的兴趣，其研究已经成为国际上一个新的热点，其中的困难也是显而易见的。

同一个话题，往往存在着多个角度、多个侧面的描述与分析，这些信息通常孤立地分散在不同的角落；话题本身也会随着时间而不断地发展变化，描述它的信息会以不同的面目出现在不同的时间。如何对分散多变的话题相关的信息进行组织，传统的信息检索技术很难有效解决这个问题。话题识别与跟踪（Topic Detection and Tracking）就是在这种情况下应运而生的。通过话题识别与跟踪，人们可以将关于话题的分散信息有效地汇集并组织起来，从而帮助用户发现话题中各种因素之间的关系，从整体上了解一个话题的全部细节以及该话题与其它话题的联系。

话题识别与跟踪的概念最早产生于 1996 年，当时美国国防高级研究计划委员会（DARPA）提出要开发一种新技术，能在没有人工干预的情况下自动判断新闻数据流的

主题。在 1997 年，卡内基—梅隆大学、Dragon 系统公司、BBN 公司以及马萨诸塞大学的研究者对这一个新的问题领域进行了初始的研究，主要是定义话题识别与跟踪研究的内容和开发相关的初步技术。这些初始研究的目的是要确定来自信息检索领域的基于主题的技术在多大程度上能够用来解决基于事件的信息组织问题。

从 1998 年开始，话题识别与跟踪的研究得到了进一步的重视。在 DARPA 的支持下，美国标准技术研究所（NIST）每年都要举办话题识别与跟踪国际评测会议。到 2004 年为止，已经成功举办了 7 次大规模的公开评测，并在 2000 年正式归属为“跨语言信息识别、抽取及摘要计划”（TIDES）的一部分。多年来，话题识别与跟踪的评测得到了各大公司和研究机构的参与和重视，如 IBM Watson 研究中心、BBN 公司、卡耐基—梅隆大学、马萨诸塞大学、宾州大学、马里兰大学、Dragon 系统等。国内这方面的研究开展得要晚一些，国立台湾大学、香港中文大学参加了几次评测；北京大学、中科院计算所和东北大学也开展了相关的研究。

话题识别与跟踪评测采用的语料是由语言数据联盟（Linguistic Data Consortium，以下简称 LDC）提供的 TDT 系列语料，目前已公开的语料包括 TDT-Pilot、TDT2、TDT3、TDT4 和 TDT5，这些语料都人工标注了若干话题作为标准答案。通常，在每次评测中，先前开发的语料被用做训练集，而新标注的语料作为测试集。总的来看，TDT 系列评测会议呈现两大趋势：一是努力提高信息来源的广泛性，不仅包括互联网上的文本数据，还包括来自广播、电视的语音数据；二是强调多语言的特性，从 1999 年开始，TDT 会议引入了对汉语话题的评测，2002 年又计划增加阿拉伯语的测试集。

2.4.2 话题识别与跟踪的主要任务

为了组织话题相关的信息，必须明确定义什么才叫做话题。事实上，广义的话题（broad topic）有两种形式：基于主题的话题（subject-based topic）和基于事件的话题（event-based topic）。基于主题的话题描述的往往是一个大的领域（如美国的对华政策）或者某一类事件（如恐怖活动）；而基于事件的话题往往指的是某一个很具体的事件，例如中国首次载入航天飞行、伊拉克战争等。

一般地，基于主题的技术倾向于把内容相似度高的文本归为一体，这是通用的信息检索、分类、聚类等技术所擅长的领域；而基于事件的技术倾向于把具有事件内在发展逻辑的文本归为一体，需要识别出事件的发展变化：开端、发展和结局等。尽管描述同一个事件的新闻报道内容往往相似，但内容相似性与事件同一性没有绝对的对应关系。为此，TDT 评测会议对“话题”、“事件”、“报道”等进行了定义。

所谓话题（Topic），就是一个核心事件或活动以及与之直接相关的事件或活动。而一个事件（Event）通常由某些原因、条件引起，发生在特定时间、地点，涉及某些对象（人或物），并可能伴随某些必然结果。通常情况下，可以简单地认为话题就是若干对某事件相关报道的集合。

根据不同的应用需求，TDT 评测把话题识别与跟踪分成五个子任务：

(1) 报道切分 (Story Segmentation)

主要是找出所有的报道的边界, 把输入的源数据流分割成各个独立的报道。这项任务可以看作是其它任务的前提, 即将连续的语音流等切分成一个个报道, 供其它任务使用。

(2) 首次报道识别 (First Story Detection, TDT2002改称New Event Detection)

这项任务的目的是在数据流中发现某一个话题的第一篇报道。这项研究对于信息内容安全和股票分析等领域是尤其有用的, 即发现重要的新出现的事件。

(3) 话题识别 (Topic Detection)

这项任务要求识别出系统未知的话题, 并将相关报道也识别出来。这项研究等同于无指导的话题聚类, 并涉及到了“话题粒度”等核心问题。在传统的话题识别任务中, 参与评测的每个话题都是很小心地定义的, 并把话题按照统一的粒度标准安排在扁平结构上。系统把报道组织成一个个孤立的团, 不考虑团之间的关系。

事实上, 话题之间是有层次包含关系的, 例如“亚洲金融危机”的话题包含了“印尼股市暴动”和“日本银行破产”等子话题。为此, 在 2004 年的 TDT 评测中, 引入了“层次化话题识别”(Hierarchical Topic Detection)的任务, 不仅要求将话题识别出来, 还要将它们组织成有向图的结构[NIST 2004]。

(4) 话题跟踪 (Topic Tracking)

话题跟踪就是要识别出关于某个已知话题的新闻报道, 通常要事先给出一个或几个已知的、关于该话题的新闻报道。话题跟踪的任务很象信息检索技术里的信息过滤任务, 不同的难点是它的初始信息非常少, 且话题可能随着时间的发展, 特征发生变化。

经过几年的发展, TDT 相关的技术已经取得了巨大的进步。目前的成果表明报道切分的性能已经和人工相差无几, 话题跟踪技术也已基本实用, 但话题识别技术还有待改进, 其性能的提高还有一定的余地。另外, 对于人物追踪来说, 更多的是在事先不给定任何初始知识的情况下, 整理出相关人物的信息。因此, 话题识别任务与人物追踪最为接近。因此, 本文接下来着重介绍话题识别的主要技术。

2.4.2 话题识别的主要技术

话题识别是一个无监督的机器学习的过程, 文本聚类是天然的应用。在目前阶段, 话题识别也是围绕文本聚类的相关技术来展开的, 主要是调整算法使之更加适合话题的处理。

2.4.2.1 选择合适的聚类模型

在话题识别任务的定义中, 系统必须增量地处理话题的信息, 在某一时刻只能看到有限的时间窗口内的报道[Wayne 2000]。这样, 众多的研究者都采用了增量聚类的方法。

卡内基—梅隆大学采用了向量空间模型, 在时间窗口内, 利用组平均聚类 (GAC) 的方法先把报道聚成小类, 再利用最近邻聚类 (Single-pass) 将小类与已有的信息合并。组平均聚类属于层次凝聚式聚类的变种。实验证明, 这种按照时序进行组内优先聚类的方法是非常合适的, 因为它把时间相近的报道以更大的优先级聚到一起。并且, 实验发

现，组平均聚类更适合于“突发性的事件”，而最近邻聚类适合“持续性事件”，把二者混合起来能达到比较好的效果[Yang 1999]。与之相似的是 IBM 所采用的两层聚类方法，当时间窗口增大时，系统的效果提高得比较明显[Franz 1999]。

马萨诸塞大学采用了两种模型：向量空间模型和语言模型。向量空间模型中尝试了中心质点法和 K 最近邻法；语言模型中计算先前的哪个话题 M 最有可能产生新到达的文档 D ，即 $P(D|M)/P(D)$ ，使用 1 元模型， $P(D|M) = \prod_i P(d_i|M)$ ， d_i 表示 D 中的符号，并采用最大似然估计来计算 $P(d_i|M)$ 。实验发现，采用 K 最近邻聚类，K 取 1 时效果是最好的[Papka 1999]。

与此同时，Dragon 系统公司和 TNO 大学都采用的是语言模型[Allan et al., 1998][Spitters 2001]。其中，TNO 在最近邻聚类中引入了聚类结果调整的策略，即当一个时间窗口内的报道处理完毕时，重新比较每一篇文档和已有的类，把文档调整到最相近的类中。并且，如果一个类结果在一段时间内不发生变化，这个类就固定不动了。实验表明，结果调整的策略大大提高了系统的效果。

2.4.2.2 使用适当的距离公式

在衡量两个团之间相似度的问题上，众多的研究者也根据话题的特点，提出了不同的方法。比较典型的就是在相似度计算公式中引入了时间的因素。例如马萨诸塞大学采用了将时间因素与文档相似度线性相加的方法[Papka 1999]，公式如下：

$$\text{threshold}(q_i, d_i) = 0.4 + \theta \cdot (\text{belief}(q_i, d_i) - 0.4) + \beta \cdot (\text{date}_i - \text{date}_i)$$

卡内基—梅隆大学把时间因素与文档相似度相乘[Yang 1999]，公式如下：

$$\text{score} = \text{score} \cdot (1 - i/m)$$

而 IBM 则采用了对称 Okapi 的相似度计算公式，并提出了“话题半衰期”的概念：

$$\text{score}' = \text{score} \cdot (1 + 2^{-\text{age}/\text{half-time}})$$

2.4.2.3 挖掘合适的话题特征词

话题识别所处理的对象是热点话题的新闻报道，所用方法是否符合话题的特点，对系统性能的提高极为重要。因此，研究者采用了符合话题特点的文本表示方法，主要是挖掘合适的话题特征词和合理地计算特征词的权重。

在话题识别中，一个很明显的现象是事件所发生时间、地点、人物、机构等命名实体，对话题的区分作用比较大。因此，一般的研究者都对命名实体进行了加权处理[Yang 2002][Allan 2002]。

对于一个新闻报道的来源来说，可能有很多特有的惯用词汇。例如，对于 CNN 网站，词汇“CNN”就可能出现的很频繁，而恰恰就是这样的词汇，对该网站来说表达力比较弱。因此，研究者针对每一个新闻来源内的文档单独进行词频的统计[Franz 1999]。

马萨诸塞大学比较了不同的权重计算方法，比较的方法包括：Inquery TF_IDF、TF_{only}

和 TF_IDF。实验表明,效果比较好的是 Inquery TF_IDF, 公式如下:

$$tf_{comp} = \frac{tf}{tf + 0.5 + 1.5 \frac{len_d}{len_{average}}} \quad idf_{comp} = \frac{\log(N / df)}{\log(N + 1)}$$

在 Inquery TF_IDF 中,系统倾向于产生较高的准确率,而召回率略有降低。并且,对于一个词来说,在一篇文档中出现一次重要度会比较高;如果连续出现,每次的重要度都会降低。由于文档长度的平滑作用,短文档对 TF 值的贡献会比较大,而长文档则对一篇文档内出现频率低的词不太重视[Allan 2002]。

跟话题识别相关的特征策略还包括采用增量的文档倒排频率(IDF)计算模型[Franz 1999],过滤短文档[Yang 1999],考虑词在文档中的位置(即报道首段的词权重较大)[Lam 2000]等。

另外,尽管有些技术不是应用在话题识别的任务中,但其特征的调整策略对话题识别也完全适用。Carthy 等人利用 WordNet 构建词汇链,即文本中相关词汇的信息串,作为文本表示的主要特征[Carthy 2002]。Makkonen 等人提出了事件向量的概念,并采用基于本体的方法进行地名相似度的计算[Makkonen 2004]。Brants 等人采用基于倒排事件频率的特征权重计算方法,即一个词在各个事件中出现的越多,这个词就越倾向于不重要[Brants 2003]。Swan 等人挖掘一段时间内流行的词汇,用来指导事件信息的组织[Swan 2000]。Fukumoto 分别采用文档和段落粒度的特征权重计算方法来识别真正能代表事件的特征[Fukumoto 2000]。David 等人利用时间词与地名词的绑定来对事件进行识别[David 2003]。

2.5 本章小结

本章主要介绍了人物追踪的相关技术。人物追踪的最终解决需要多方面文本处理技术的支撑,其中有些技术已经成熟,有些技术需要改进,还有的技术才刚刚起步。

跟人物追踪关系最为密切的研究领域包括:网页解析、信息抽取、文本聚类 and 话题识别与跟踪。对于人物追踪来说,网页解析主要是用来从网页中抽取出比较规范报道,去除网页的噪音;信息抽取主要是从报道中提取出人物的属性等信息;文本聚类主要是让系统自动发现一些人物相关的隐含信息;话题识别与跟踪主要是把人物相关的报道按照内在的发展逻辑组织起来。

随着相关技术的进步和完善,人物追踪必然会取得更大的发展;人物追踪的发展也会带动相关研究领域的进步。

第三章 人物追踪中的数据预处理

人物的信息往往隐含在不同来源的数据中，数据的格式多种多样，每种数据的结构也可能不断发生变化。如何对异构多变的数据进行清理，把杂乱、充满噪音的数据转变为适合于人物追踪要求的统一数据格式，是数据预处理阶段的主要工作。数据预处理是人物追踪的首要环节，是人物追踪的前提和基础。数据预处理的质量制约着人物追踪的最终性能。

人物追踪可以利用的数据来源很多，例如广播、电视、短信、报纸、杂志、网页等等。各种形式的数据的质量参差不齐，可以达到的效果也各不相同。与其它的研究内容一样，在起步的阶段，研究者往往尽量排除非次要因素的干扰，力求在比较理想的环境下探索核心技术，再向其它应用环境推广。另外，鉴于目前电子文档的爆炸式增长和人们在互联网上寻找信息的迷茫，我们选取目前互联网上格式和内容都相对比较规整的新闻网页作为研究的对象。与其它数据源相比，新闻网页具有格式规范、权威性高、报道及时等特点，因此成为人物追踪研究的理想数据源。

在这一章里，我们主要介绍针对汉语的新闻网页，人物追踪数据预处理的主要流程，然后对每一流程的具体内容详细介绍。我们重点介绍了网页噪音过滤和人物属性抽取的方法。

3.1 数据预处理的主要流程

对于人物追踪来说，数据预处理主要解决的是把蕴涵在文本中的人物相关信息挖掘出来，建立起人物模型的片断，为后续的处理做准备。在 1.3.1 节中有所介绍，人物模型中的信息主要由以下几个部分构成：人物名称、基本属性信息、介绍性信息和普通词场。数据预处理应该能够从含有噪音的数据中挖掘出上述相关的信息。

如果数据来源于互联网的新闻网页，则这个阶段涉及的内容主要包括：网页噪音过滤、人名的识别、人名指代的处理、人物对应语段的确定、时间和时序标签的确定、人物属性的抽取。如图 3.1 所示。

网页噪音过滤主要解决的是从新闻网页中提取出新闻报道的正文、标题和报道时间等信息，过滤掉广告、链接等噪音；人名识别主要是对新闻内容进行汉语的词法分析，识别出人名标签，尤其需要进行人名/非人名的区分；在人名指代处理阶段，主要是识别出人名指代的代词，把潜在词场转化为现实词场；然后，确定在一篇报道内跟某一人物相关的语段范围，正式确定该人物对应的词场；接下来，确定这些语段所描述的事件所发生的具体时间，或其它事件相比的时间顺序；最后，识别人物的属性信息，包括基本属性信息和介绍性信息，填充进人物模型，形成独立的人物模型片断。

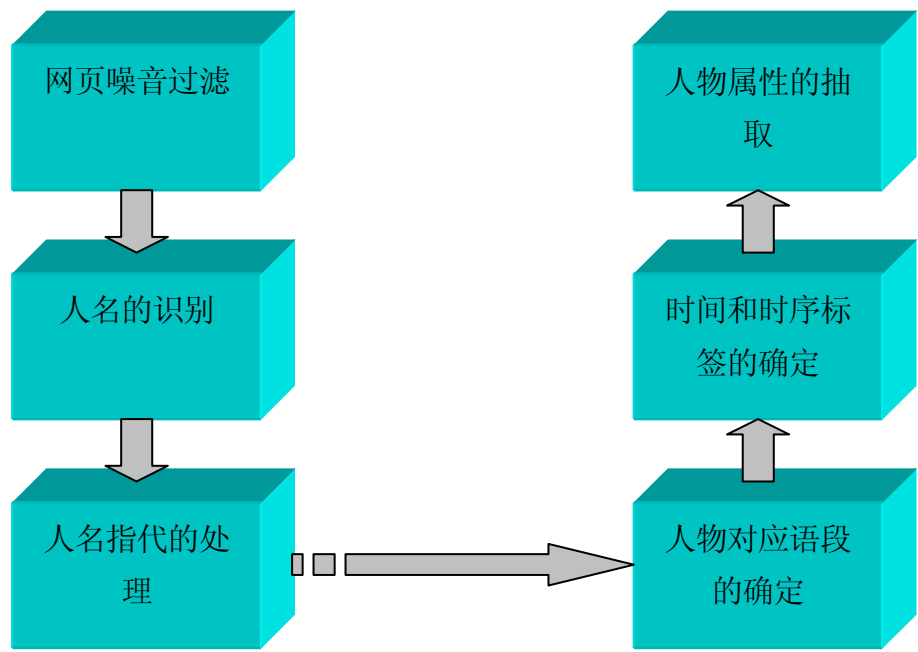


图 3.1 人物追踪中数据预处理的主要流程示意图

3.2 网页噪音过滤

在 WWW 中，存在着两类网页：一类以超链接文字为主，这类网页仅提供一个目录，给内容上有关联的零散网页提供一个公共的入口，这类网页称之为“索引页面”。还有一类网页本身含有实际的内容，一般是显示具体信息的末级页面，我们称之为“内容页面”。即使在内容页面内部，也往往存在着大量与主题无关的信息，例如：为了便于用户浏览而加入的导航链接，以及出于商业因素所加入的广告链接等。这样，一个 Web 页面中往往包含着多个语义无关的部分，根据各个部分性质的不同又可以划分为“索引块”和“内容块”。

对于人物追踪来说，除了报道正文和标题等信息以外，其它所有内容均可视为网页噪音。我们需要从新闻网页中提取出新闻的正文和标题，以及报道的时间。值得一提的是，新闻标题的提取并不是很简单的问题。在通常的搜索引擎上，返回的标题往往是 HTML 页面中“TITLE”标记所夹的标题，然而，由于这个标题往往很随意，不一定能够代表报道的内容。如图 3.2 所示的网页，“TITLE”标记所夹的标题“上海热线体育频道”就无法有效代表正文的信息。为了得到高质量的新闻标题，稳妥的方法只能是分析挖掘蕴涵在网页文字中的标题，例如图 3.2 中的“杰克逊不谈夺冠谈对手 活塞球星乃是湖人一手制造”。只有这个标题才能真正代表新闻报道的内容。

在这里，我们重点研究针对汉语的新闻网页，如何提取新闻正文、新闻标题和报道时间的方法。提取的结果如图 3.2 右侧结果所示。



图 3.2 网页噪音过滤示意图

3.2.1 网页解析算法 TVPS

网页大多是用 HTML 语言写成的，网页噪音过滤多通过对 HTML 页面的解析来完成。一般来说，HTML 页面解析可利用以下四个规律来进行：HTML 标记分布规律[Embley 1999]、HTML 标记间的关系[Lim 2000]、页面视觉特征[Yu 2003]、TABLE 标记的布局特性[Liu 2002]。

从 HTML 页面解析的基本技术手段来看，如果处理的对象是特定领域的格式较为简单、规整的网页，利用 HTML 标记分布规律和 HTML 标记间的关系，效果较好，更有针对性；而如果处理的是样式结构较为复杂、格式缺乏共性的网页，页面视觉特征和 TABLE 标记的布局特性比较有效。

在处理诸如新闻网页的复杂页面时，采用视觉特征或 TABLE 标记布局特性的方法各有利弊：视觉特征很好地反映了页面的外在样式与页面内各语义块间的内在联系；TABLE 标记的布局特性则从页面的结构组织上反映了页面的结构与各语义块间的联系。前者分析精细，但知识表示模糊；后者规则明确，但缺乏微观的把握。

为了能够有效处理现今新闻网站上普遍流行的网页，我们把这两种方法综合起来，利用各自的优点，克服不足，提出了综合利用 TABLE 标记和视觉特征对页面进行解析的算法 TVPS (Table and Vision based Page Segmentation)。TVPS 算法能够将网页内各个语义块进行划分，并对各个块进行分析，判别其是属于索引块还是内容块；针对内容块，还可进行精细的挖掘，抽取出标题和报道时间等信息。在综合分析页面中各个块的基础上，判断当前页面是索引页还是内容页；然后，把索引页和索引块中的信息视为噪声过滤掉。与完全基于视觉特征对网页进行分块解析的方法[Yu 2003]相比，该算法有效地简

化了规则集的规模,较好地解决了规则一致性的问题,取得了很好的效果。图 3.3 是 TVPS 算法对 HTML 文档结构分析的示意图。

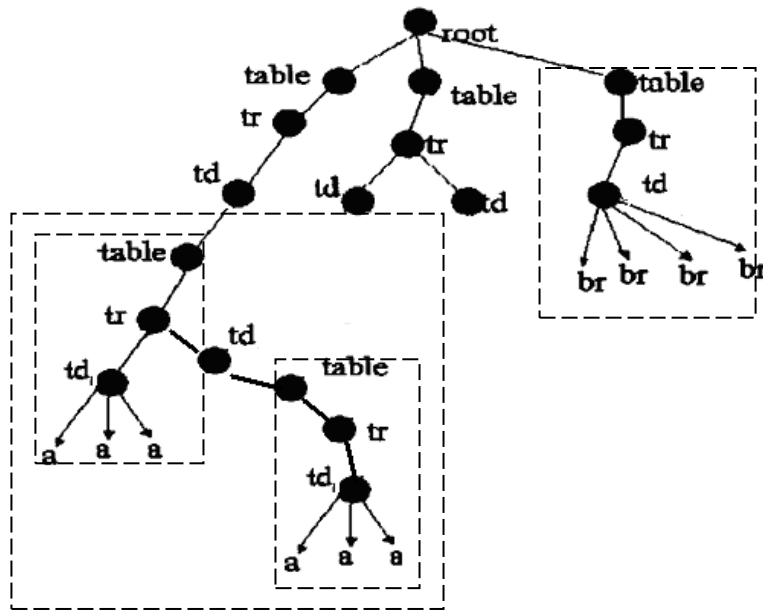


图 3.3 TVPS 算法分析 HTML 文档结构示意图

TVPS 算法主要步骤如下:

(1) 构造 HTML 标记树

首先对 HTML 代码进行预处理,例如补全“”、“<hr>”等的结束标志,再去除一些无用标记,如“form”、“img”、注释标记等;然后构造一棵兄弟-孩子结点树。树中每个结点保留字体大小、粗细、颜色、背景等信息。父结点的属性可以继承到子节点上。

构造 HTML 标记树可以通过编写递归算法来实现。需要注意的是对 HTML 文件中错误语法的处理。由于 HTML 语法书写的随意性很大,其中的错误类型往往会超出 HTML 语法分析程序设计者的想象,稍有不甚,就会造成内存错误,使系统崩溃。

为了能够有效地对 HTML 文件中的语法错误进行处理,我们采用 W3C 提供的 HTML 词法分析器[W3C]来进行 HTML 标志名和标志属性的提取。该词法分析器可以分为 5 个状态:1) 初始状态;2) 标记开始状态;3) 文本状态;4) 标记结束状态;5) 结束状态。每个状态分别调用相应的触发函数:InitLib、beginElement、addText、endElement、closeLib。其中在 beginElement 函数里边可以取得当前的标记名及其该标记的属性。

(2) 对无用子树进行剪枝

我们主要是对记录信息进行识别,而那些叶子结点不是文本或链接的子树,不含记录信息,被认为是无用子树,把这样的子树剪枝掉。

(3) 分块,并寻找各块的标题

首先把各个最底层的 Table 标记对应的子树合并成一个块,把该块中大多数记录的视觉属性作为该结点的视觉属性。如果发现最左端那条记录比其它记录醒目,则把该条记录定为该块的标题。其中,记录 A 是否比记录 B 醒目采用[李晓黎 2001]提出的五个

判断因素：A 的字是否比 B 的大，A、B 的颜色是否不同，A、B 的字体是否不同，A 是否是黑体以及 A 的周围是否存在空行等因素判断。当有二个或二个以上的因素为真时，A 比 B 醒目为真。

因为有的语义块标题和记录内容分别位于页面内不同的 Table 中，所以有必要对剪枝和合并后的子树进行检查。如果发现这样的两个子树 A 和 B，满足如下关系：

$Left(A, B) \wedge (Len(A) < 60) \wedge Outstanding(A, B)$ ，则把 A 和 B 合并，同时认为 A 是 B 的标题。

(4) 判断各语义块及页面的类型

在各块内部，统计链接文字跟非链接文字的比例，如果该比例大于 a ($a=2$)，则认为该块是“索引块”；否则为“内容块”。

如果一个页面中全部语义块都为索引块，则该页面为“索引页”；否则为“内容页”。

(5) 识别内容页的报道时间

为了有效识别出内容页新闻报道的时间，我们采用位置和词性双重约束的方式：考虑内容块标题和正文之间的文字，判断它们的词性，对词性为“数词(m)”或“时间词(t)”的文字串，把它挖掘出来，作为报道时间。

我们采用中科院计算所软件室研发的词法分析器 ICTCLAS[ICTCLAS]进行词性的判断，其对时间信息的分析结果如下所示：

“2004-06-15/m 08:/m 57:/m 45/m”、“2004 年/t 06 月/t 15 日/t 05:/m 37/m”。

3.2.2 实验结果与分析

目前，在网页解析方面，研究界还没有一个统一的数据集。因此，为了测试相关算法的性能，有必要单独建立一套数据集和评测方法。

3.2.2.1 数据集和评测方法

我们收集了 2000 余篇汉语的新闻网页，收集网页采用的是“元搜索”的方式，即我们制作了一个元搜索采集器，输入一些起始关键词，该采集器就能自动从“百度”新闻搜索引擎 (<http://news.baidu.com>) 中采集下来关键词对应的查询结果。在这里，我们输入的关键词是 5 个中文的人名。采集下来的初始网页一共 2632 篇，经过网页去重，剩下了 2110 篇。遗憾的是，通过新闻搜索引擎采集下来的网页都是内容页面。为了有效验证算法针对索引页与内容页划分的性能，我们又人为地加进了 40 篇索引页面。这样，我们的数据集合一共有 2150 篇。

需要测试的内容包括以下几项：索引页与内容页的划分、新闻正文的识别、网页噪音的过滤、报道时间识别、正文标题识别。其中，新闻正文识别和网页噪音过滤是否正确的判断标准比较模糊。我们规定，仅当新闻正文的首段和尾段的定位偏差都不超过两个自然段时，才认为新闻正文识别正确；仅当一篇网页上的噪音过滤掉 95% 以上时，才认为噪音过滤正确。这样，设某一项正确识别的结果数为 m ，需要识别的样本数为 n ，则我们只采用准确率 P 来衡量每一项的效果，公式如下：

$$P = m/n$$

3.2.2.2 实验结果

我们使用 TVPS 算法对数据集中的网页进行了处理，实验结果还是令人满意的。由于网页解析处于整个系统的前期阶段，它的性能将直接影响到后续的处理。从表 3.1 的结果来看，TVPS 算法基本可以达到实用效果。

表 3.1 TVPS 算法的实验结果

测试项	准确率
索引页过滤	99.0%
新闻正文识别	93.6%
网页噪音过滤	89.5%
报道时间识别	86.2%
正文标题识别	78.6%

就我们所知，目前网页解析的工作受到越来越多的重视，相关的研究层出不穷[Yu 2003][Hu 2005]，但还没有看见相关文献详细介绍从新闻网页中提取正文、标题、报道时间并过滤噪音的一体化算法。TVPS 算法的优点在于它充分利用了新闻网页采用 TABLE 标记布局的特点，同时结合视觉特征来辅助判断，因而能够达到比较好的性能。

3.3 人名的识别

经过网页的解析，得到了新闻正文和标题的原始文本。从计算机角度来看，原始文本只是一堆字符串，字符串中并没有对人物的信息加以特殊标记。为了有效地挖掘人物的信息，有必要识别人物信息的触发特征。人名作为人物的标识，自然成为触发特征的首选。

人名识别是指在信息处理中，把人名从无分隔符号标记的字串中准确地自动识别出来的过程。英语中人名的识别相对容易，对于中文人名识别来说，有其固有的困难，且往往跟中文的分词密切结合在一起。中文人名识别困难是因为中国人名数量众多，规律各异，且有很大的随意性，主要表现在：中国人名的构成具有多样性；人名内部往往相互成词；人名与其上下文经常组合成词；同源歧义的冲突等[张华平 2004]。另外，由于标准不够统一，外文人名翻译还常常产生差异和混乱。

针对中国人名的自动识别问题，人们已经做过深入的探索，并提出了多种解决方案。根据其使用的方法不同，这些方案大致可分为三种：规则方法，统计方法以及规则与统计相结合的方法。规则的方法简单且有针对性，当缺乏大规模语料的时候比较适用；而统计的方法需要大规模标注语料，所用方法一般比较复杂。因此，目前的研究一般倾向于将规则和统计的方法结合起来，根据调整二者的比重来满足具体情况的需求[张华平

2004]。

在人名识别方面，我们直接使用中国科学院计算技术研究所研制的汉语词法分析系统 *ICTCLAS*[*ICTCLAS*]。该系统是一个集分词、词性标注、未登录词识别、词性标注于一体的汉语词法分析系统，其中采用了基于角色标注的中国人名自动识别方法。该方法主要采用隐马尔可夫模型对分词结果进行人名构成角色的标注，然后根据各个不同的角色，进行简单的模式匹配并最终识别出人名。该系统人名识别的正确率和召回率分别达到 95.57% 和 95.23% [张华平 2004]，可以很好地满足实际需求。

值得一提的是，由于 *ICTCLAS* 采用了层次隐马模型（Hierarchical Hidden Markov Model; HHMM）[Shai 1998] 的结构，所以能够有效进行人名/非人名的区分。这是因为，人名经常嵌套在复杂的地名和机构名中，这样的人名并不能代表现实中的人物，例如“张自忠路”、“周恩来和邓颖超纪念馆”。在 *ICTCLAS* 中，首先在低层的 HMM 识别过程中识别出普通不嵌套的未登录词，然后在此基础上，利用相同的方法采取高层隐马模型，通过角色标注计算出最优的角色序列，从而进一步识别出嵌套的未登录词 [张华平 2004]。例如，切分片断“周/恩/来/和/邓/颖/超/纪念馆”，*ICTCLAS* 先识别出“周恩来”和“邓颖超”为人名 PER，得到新的词类序列“PER/和/PER/纪念馆”，最终识别出该片段为机构名，而不是人名。

我们采用 *ICTCLAS* 对新闻内容进行词法分析，并把词性标签为“nr”、“nrf”的词语视为人物的姓名，作为人物信息挖掘的触发词汇。

3.4 人名指代的处理

在文本中，对人物的标识词汇不一定是人物的姓名，还可能是人名的指代词。具有指代作用的词主要包括：姓、名、人称代词“他/她”等。只有把这些指代词跟人名的对应关系整理清楚，才能够找全人物信息的描述范围。指代识别正确了，潜在的词场才能转化为现实的词场，丰富人物实体的特征数据。

人名指代的处理是指代消解研究的一部分。目前，国内和国外有关指代消解的研究工作都取得了不少成绩 [Saliha 1998] [王厚峰 2001] [张威 2002]。人名指代处理中比较复杂的是人称代词的消解，即确定人称代词与先行词之间的关系，从而明确人称代词究竟指代什么对象。人称代词的消解是自然语言处理中十分重要的问题。针对汉语篇章，在必须解决人称指代问题时，一般都遵循“最近匹配原则” [张威 2002]。而指示词与先行词之间可以跨越多个句子；距离越大，先行词候选集规模可能越大，解决消解问题的复杂性也就越大。

在这里，我们的做法没有涉及人称代词的指代消解，而是只对名或姓的指代进行了处理。例如：“王建民”先行词与“王校长”、“建民同志”之间的指代关系。我们的做法主要基于这样的观察：除非指代消解模块的准确率非常高，否则它会损害人物追踪整体的效果。

我们实现了一个基于规则的人名指代消解系统，只有具有较高准确率的规则被应用进来，即我们只处理“姓”（词性标注为“nr1”）和“名”（词性标注为“nr2”）的指代词，且严格遵循先行词中“姓”和“名”部分的完全匹配。

3.5 人物对应语段的确定

在一篇新闻报道中，尤其是新闻报道比较长的时候，跟一个人物相关的语段范围是有限的，它往往不是全文，而只是报道的一部分。如果要精确地确定人物在报道中的作用语段，需要复杂的篇章分析。在人物追踪中，挖掘人物在报道中相关的语段，可以利用一些启发式知识，以人名及其指代词为中心来对文档进行分割。这些启发式知识主要是利用人名及其指代词在文档中的位置分布来确定人物的作用范围。在这里，我们确定的规则如下：

- ① 如果人名或其指代词出现在报道的标题和正文首段，则该人物的作用范围是报道全篇；
- ② 如果人名或其指代词出现在报道的其它段落，则该人物的作用范围是当前段落；
- ③ 如果某一段落（尾段除外）相邻的两个语段都是某人物的有效段落，则这个段落也是该人物的有效段落。

这样，如果一篇报道描述了 N 个人，我们就对该报道拆分重组成 N 个有效描述的集合 $\{P_1, P_2, \dots, P_N\}$ ，分别与这 N 个人相对应。我们采用上述规则来确定人物对应的语段，一个简单的例子如图 3.4 所示。

继续对日本进行正式友好访问 吴邦国委员长抵达大阪

本报大阪（日本）9月9日电 记者史宗星报道：全国人大常委会委员长吴邦国9日抵达日本第二大城市大阪，继续对日本进行正式友好访问。

下午，吴邦国考察了松下电器公司。松下电器公司董事长森下洋一、社长中村邦夫向吴邦国介绍了松下的产业发展和与中国合作的情况。松下公司是最早进入中国市场的日本公司之一，在中国开办了 53 家从事生产、研发或销售的独资或合资企业，直接投资达 70 亿元。中国已经成为松下海外事业发展的重要支柱。松下已经制定 2005 年中国市场年销售达 1 万亿日元（约 700 亿元人民币）的目标。

吴邦国积极评价了松下同中国的合作，表示很高兴地看到松下在同中国合作的过程中自己也得到了壮大，已经成为中国消费者家喻户晓的日本公司。

他表示，中国和日本现在又面临着双边关系发展的新机遇，希望有远见的日本公司能充分分享中国高速增长的市场，一起来创造更美好的明天。

傍晚，吴邦国会晤了大阪府知事太田房江、大阪市市长矶村隆文和日本关西经济界人士。

图 3.4 人物对应语段确定示意图

图 3.4 中的内容是一篇新闻报道。报道中出现了六个人名：“吴邦国”、“史宗星”、“森下洋一”、“中村邦夫”、“太田房江”、“矶村隆文”。其中，“吴邦国”在报道的标题中出现，则全篇报道都认定与他有关，所以“吴邦国”的作用范围是整个文档；“史宗星”是报道记者名，可以通过简单的规则过滤掉；“森下洋一”和“中村邦夫”只在文中第二段出现，所以把第二段分别作为这两人的作用范围；同理，“太田房江”和“矶村隆文”的作用范围是报道的最后一段。这样，我们可以把这篇报道重组成 3 个描述范围，分给不同的 5 个人。

3.6 时间和时序标签的确定

时间是事件描述的一个重要因素，是区分不同事件的重要依据。为了表明一篇新闻报道所发布的时间，一般报道本身会有一个时间戳记。根据描述的时间粒度不同，可以把时间表达式划分为日期表达式和时刻表达式[李保利 2003]。

从字面上看，同一个时间表达式可以有不同的表达方式，例如“2006 年 4 月 1 日”与“二零零六年四月一日”。虽然人工很容易判断是同一个日期，但在计算机看来，却是完全不同的两个字串。所以，有必要对时间表达式进行识别和规范化处理。

近年来，时间表达式的识别和处理得到许多研究者的关注[Mani 2000][Schilder 2001]。在汉语研究方面，时间表达式几十年来也一直受到汉语语言学界的注意[李向农 1997][陆俭明 1991]。但是，从中文信息处理角度对汉语时间表达式进行研究还没有得到应有的重视。

另一方面，由于新闻报道本身一般会有一个时间戳记，所以在新闻报道中经常使用相对时间表达式，例如“下午”、“3 日”、“明年”等。这样看来，新闻报道本身的时间戳记并不能代表人物对应描述 $P_i(1 \leq i \leq N)$ 所发生的时间。为了更好地确定人物对应描述所发生的时间，有必要对时间表达式识别并规范化处理之后，进行一定的时制运算。[马红妹 2002]从解决汉英机器翻译中英语动词时态的正确生成问题出发，提出了一种在汉语篇章语境中分析时间短语和计算时制的方法。[李保利 2003]采用对基准日期和相对日期进行规范化后再进行时制计算的方法，可以将文档中出现的绝对时间表达式和相对时间表达式转换成统一的数字格式。

在论文中，我们对时制运算不作进一步的研究，只是简单地把新闻报道本身的时间戳记作为人物对应描述发生的时间，并只对典型的几种时间描述格式进行了规范化处理。

3.7 人物属性的抽取

在前面介绍的几个数据预处理步骤中，经过词法分析和人名的识别，我们得到了人物的姓名；经过人名的指代处理和人物对应语段的确定，我们得到了描述人物的普通词场。此时，人物模型还有两大类数据没有获得：基本属性信息和介绍性信息。如 1.3.1 所介绍的，人物的基本属性信息主要包括：性别、民族、生日、家乡、亲属等；人物的

介绍性信息主要包括：职位、住址、配偶、同事、联系方式等。与普通词场相比，这些信息对人物的刻画更加深入和精确。如何从文本中得到这些信息，也需要更加精确有效的手段。其中，信息抽取（IE）尤其是实体关系抽取是解决这一问题的有效方法。

3.7.1 人物属性抽取的总体框架

人物属性抽取可以看作是实体关系抽取的具体应用。实体关系抽取一直是信息抽取研究领域的重要课题[Grishman 1997]，得到众多研究者的重视。MUC-7 和 ACE[ACE]评测会议都把实体关系抽取列为重要的评测内容。其中，ACE 评测提供的语料不仅是英文，还包括中文和阿拉伯文。

与自然语言处理领域的许多问题类似，实体关系抽取最初也是使用知识库的方法[Aone 2000]。这种方法需要经验丰富的领域专家手工构筑大规模的知识库（主要是编写抽取模板），需要大量的人工劳动；而且用这种方法实现的实体关系抽取系统很难从一个领域移植到另一个领域。

为了应对这种挑战，近年来研究者倾向于把基于知识库的方法改换成机器学习的方法。这种方法往往把实体关系抽取看作是一个分类问题，一般来说它分为两个阶段：训练阶段和测试阶段。在训练阶段，需要人工标注一定规模的语料库，主要是标注出存在着特定关系的实体对，然后系统从标注好的语料中学习并构造出分类器。在测试阶段，系统对具有潜在关系的实体对，使用分类器进行分类，并决定该实体对是否真正存在着特定的关系。

多种机器学习的方法都可以有效应用在实体关系抽取中。采用分类思想进行实体关系抽取属于监督式机器学习方法。为了减少语料的标注量，有的研究者引进了弱监督式机器学习[Zhang 2004][Agichtein 2000][Brin 1998]，基本用意是在减少语料标注量的同时仍能维持不错的抽取性能。

按照分类器的表示形式划分，实体关系抽取主要有两种：特征向量表示法和 Kernel 表示法。特征向量表示法主要是把对象实例数值化成向量形式，然后对于给定的一组训练数据 $(x_1, y_1)(x_2, y_2) \dots (x_n, y_n)$ ，其中 $y_i \in \{-1, +1\}$ ，学习一个分类函数 f ，使得对于给定的新的特征向量 x' ， f 能够将其正确地分类，即 $f(x') = y'$ 。与特征向量学习方法不同，Kernel 表示法不需要构造特征向量，而是直接使用实例的原始形式作为处理对象，需要计算的是任何两个对象之间的 Kernel 函数[Aron 2004][Zelenko 2003]。Kernel 表示法的优点是它可以搜索更广阔的特征空间，因此往往能够达到比特征向量法更佳的效果。当处理大规模数据集时，Kernel 表示法的速度要比特征向量法慢。

我们关注的是大规模互联网真实环境下人物属性的抽取。与其它抽取任务相比，我们所面临的问题有以下特点：语料规模巨大，对系统的效率要求很高；人名和各个属性之间表达方式灵活多样，简单的模板定制方法很难把各种情况概括全面；并且，抽取出

来的属性信息不仅用于结果呈现，还会参与后续的计算，所以对准确率要求很高。

基于以上三个特点，我们采用监督式机器学习的方法来抽取人物的属性，即手工标注一定规模的语料，然后训练出分类器用来判断新的实体对是否存在搭配关系。分类器采用特征向量的表示方法。图 3.5 是我们采用的抽取方法的主要流程。

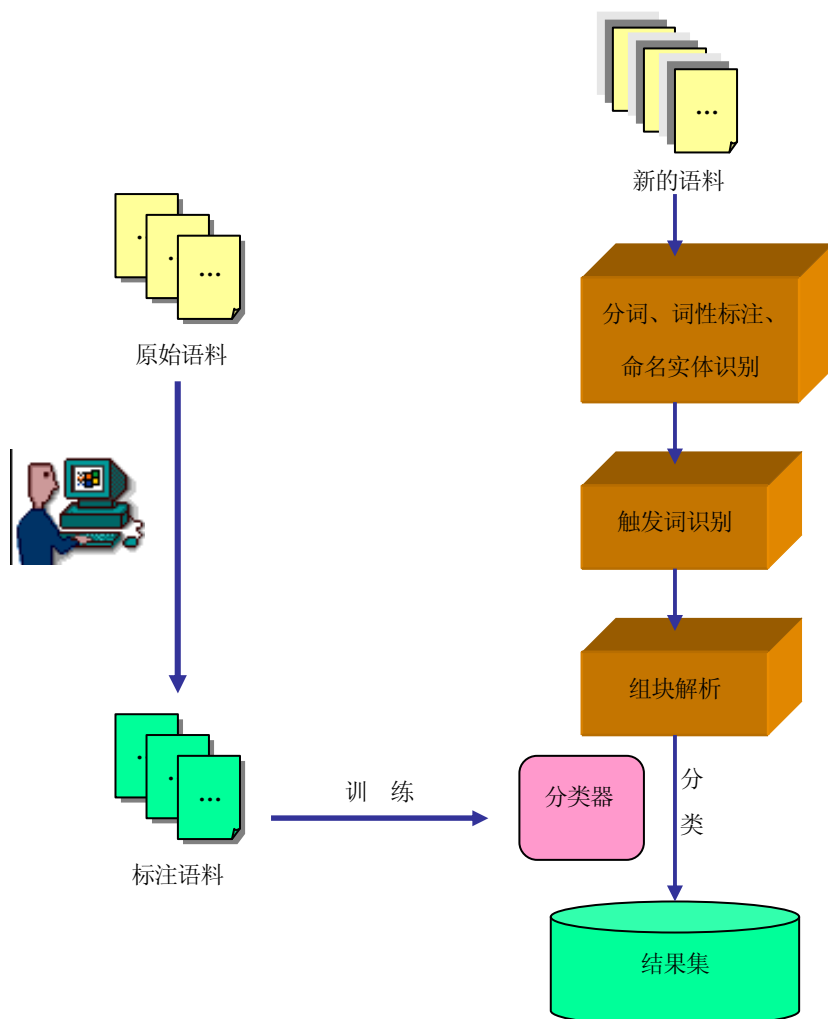


图 3.5 人物属性抽取方法流程图

3.7.2 人物属性抽取语料集的标注

我们关注的是汉语的新闻网页中人物属性的抽取问题。遗憾的是，尽管实体关系抽取的研究很多，但我们还没有找到符合我们要求的特定语料。我们拟自己收集新闻网页并进行标注，建立一个汉语的、针对新闻网页的人物属性抽取语料集。

网页的收集仍然采用“元搜索”的方式，即利用元搜索采集器从百度新闻搜索引擎上采集查询的返回结果，查询的输入信息是 17 个人名，所选人名力求在所属行业、知名度、同名程度上达到均衡，于 2004 年 11 月 27 日采集。这样，采集下来的原始网页一共 7731 篇，经过进一步的噪音过滤（把新闻正文中不含输入人名的网页都过滤掉）和网页去重，剩下 4315 篇。

在新闻网页中出现的人物属性，比较典型的有七类：性别、民族、生日、籍贯、职

位、住址、家庭关系。我们只对 4315 篇新闻正文中起始输入的 17 个人名的属性进行标注。为了使手工标注的词语边界与机器自动分析的相一致，我们的属性标注是在词法分析的结果上进行的。采用的词法分析软件是 3.2.1 所介绍的 *ICTCLAS*。语料标注的格式举例如下所示。

(1) 人物职位语料：

例一：[东安县职业中专/nt 校长/n] [李华/nr] 被/pbei 授予/v [永州市/ns 优秀/a 教育工作者/n] 荣誉/n 称号/n

例二：[李华/nr] 9/m 次/qv 被/pbei 评为/v [县/n 先进/a 教育工作者/n]

(2) 人物住址语料：

例一：[李华/nr] ， /wd 家/n 住/vi [北京市/ns 海淀区/ns]

例二：一个/mq “/wyz [濮阳县城关镇/nt 南环/ns 西路/ns 163/m 号/q] [李华/nr] ” /wyy 的/ude1 汇款/n 地址/n

(3) 人物亲属关系语料：

例一：姐妹/n 俩/m [李中/nr] 和/cc [李华/nr]

例二：那/rzv 妇女/n 名叫/v [赵桂花/nr] ， /wd 这天/rzt 打算/v 带/v 着/uzhe 儿子/n [李华/nr] 从/p 深圳/ns 乘/v 1506/m 次/qv 列车/n 回/v 江西/ns 老家/n

(4) 人物性别语料：

例一：[李华/nr] ， /wd [男/b] ， /wd 住/vi 广西/ns 全州县/ns 全州/n 镇/n 黄中路/ns 95/m 号/q

例二：[李华/nr] [先生/n] 被/pbei 任命/v 为/v 东安/nz 四中/n 校长/n

值得一提的是，关于人物性别，我们不对第三人称代词“他/她”体现的性别信息进行识别，我们认为其属于第三人称代词指代消解的研究范畴。我们进行语料标注的范围只限于汉语的长句，不关注跨句的属性抽取。这样，在 4315 篇语料的 17 个人名对应的属性信息里，“性别”信息出现 113 个，“民族”出现 15 个，“家庭关系”出现 23 个，“出生年月”出现 27 个，“家乡”出现 77 个，“住址”出现 25 个，“职位”出现 2301 个。

针对每种属性，我们把语料平均分成 3 份，且保证该属性在每份中所占的比重是相同的。我们分别拿其中 2 份做训练，1 份做测试。我们采用 3 次交叉验证的方式，最后的结果取 3 次结果的平均。

3.7.3 分类器的模型表示

在这里，我们主要介绍分类器的表示方法，前面一节主要介绍分类器的表示框架，然后介绍在此框架下引入的语义扩展方法。

3.7.3.1 普通词场的表示方法

我们采用特征向量来表示分类器，特征向量的内容来自于围绕人名和属性周边的若干词语和符号。也就是说，我们把人名和属性看作是一个二元关系<tag1,tag2>，围绕 tag1

和 tag2 的词汇按照位置可以划分为 Left、Middle 和 Right，这些信息构成了五元组 [Left, tag1, Middle, tag2, Right]。

在实际中，Left 取 tag1 左边窗口大小为 2 的词，如果遇到标点符号和句子边界则被截断；Right 取 tag2 右边窗口大小为 2 的词，如果遇到标点符号和句子边界则被截断；Middle 取 tag1 和 tag2 中间的词。

Left、Middle、Right 中有好多词汇起不到表征的作用，为此我们建立了停用词性表，把 Left、Middle、Right 中的“助词”、“人名”、“标点符号”去掉。经过停用词的去除，Left、Middle、Right 可能会出现值为空的情况。当 Middle 为空时，表示 tag1 和 tag2 紧邻，其关系可能会更加密切，我们对空值 Middle 采用特殊符号“&”加以标记。

我们采用向量空间模型来表示分类器，向量中每一维度的权重采用 TF 值来表示。为了简化，我们把每一属性的五元组中 Left、Middle、Right 的词汇混合成一个向量。

3.7.3.2 基于概念义原的语义计算

如果把互联网中的新闻网页看作是一个大的语料测试集，那么它的规模是巨大的。相比之下，所标注的训练集规模就异常的小，所以很难覆盖互联网上的所有语言情况；另外，汉语的表达方式灵活多变，描述同一概念的词汇多种多样，可能存在着字面上的差异。例如，描述职位相关的特征词汇就可能有“担任”、“上任”、“调任”、“评选”、“指定”等，字面描述的差异造成了关键性特征的分散，使特征向量无法有效起到标识作用。为了解决这个问题，一个较好的思路是引进语义计算的方法。

计算两个字或词之间的语义距离或相似度已经有多年的研究历史，方法可大致分为三类：基于语料库的方法，基于本体的方法和混合的方法[Zhang 2005]。基于语料库的方法往往通过计算两个词出现的语言环境来计算它们的语义距离[Church 1989][Grefenstette 1992]，这种方法不需要先验知识，然而却忽略了语言所固有的本质；基于本体的方法主要是利用词汇分类体系的先验知识来计算语义距离。英文中比较常用的是 WordNet [George 2003]，汉语里面比较流行的是知网（HowNet）和同义词词林[Li 2002][Liu 2002]。基于本体的方法简单、直观、易于理解，然而采用的方法易于主观。混合的方法是把前两种方法混合起来，综合利用二者的优点，克服不足[Jiang 1997]。

我们考虑如何在分类器中引入语义计算的方法。[Zhang 2005]采用两两词汇进行语义计算然后加权平均的方法来计算两个英语句子的语义相似度。类似地，我们也可以把该方法引入到两个汉语特征向量的语义相似度计算中来。然而，该方法倾向于引入大量语义噪音，稍不注意就会造成结果的下降；另外，英语和汉语的单个词汇的语义计算方法各有不同，在英语中效果好的方法不一定适用在汉语的环境。

为此，我们只在分类器训练时特征向量的形成阶段引入语义计算。基本方法是对特征向量普通词场中的词进行语义聚类，把聚类结果的每个词团作为统一的维度，并把该词团中每个词的权重相加作为该维度的权重。我们主要借用 HowNet 来进行语义计算。由于 HowNet 的基本元素叫义原，是指最基本的、不易于再分割的意义的最小单位；而

HowNet 总是通过用一系列的义原，利用某种知识描述语言来描述一个概念。因此，我们把基于义原体系得到的词语聚类结果称为“概念义原”。这样，我们的特征向量可以表示为：

$$\text{SEM}\{(\text{sem}_1(w_1, w_2), k_1), (\text{sem}_2(w_3, w_4, w_5), k_2), \dots, (\text{sem}_m(w_{n-2}, w_{n-1}, w_n), k_m)\}。$$

其中 $\text{sem}_1, \text{sem}_2, \dots, \text{sem}_m$ 是概念义原， w_1, w_2 是归为 sem_1 的两个词； w_3, w_4, w_5 是归为 sem_2 的三个词……。 k_1 是 sem_1 的权重，也就是 w_1, w_2 的权重之和。 k_2 是 sem_2 的权重，也就是 w_3, w_4, w_5 的权重之和……。

概念义原对应的是词在特定上下文中所表达的概念，因而字面上不同的词可能对应相同的概念义原。比如，“死”、“死亡”、“丧生”、“毙命”、“遇难”对应的概念义原均为“die/死”，因为它们在 HowNet 中的定义均为“DEF= die/死”。这样，特征向量中每个维度都以“概念义原”作为统计对象而不是以词作为统计对象，可以消除表达同一个概念的不同的词在字面上的差异，突出关键性的特征，解决语料稀疏的问题。

基于概念义原，我们关注如下相似度的计算：词语间相似度计算；概念义原间相似度计算；概念义原和普通词场相似度计算。

(1) 词语间相似度计算

两个词之间的语义相似度调用[Liu 2002]提供的“基于《知网》的词汇语义相似度计算”模块。该模块采用了“整体的相似度等于部分相似度加权平均”的做法。首先将一个整体分解成部分，再将两个整体的各个部分进行组合配对，通过计算每个组合对的相似度的加权平均得到整体的相似度。对于两个义原的相似度，该模块采用根据上下位关系得到语义距离并进行转换的方法。

对于两个汉语词语 W_1 和 W_2 ，如果 W_1 有 n 个义项（概念）： $S_{11}, S_{12}, \dots, S_{1n}$ ， W_2 有 m 个义项（概念）： $S_{21}, S_{22}, \dots, S_{2m}$ ，则 W_1 和 W_2 的相似度是各个概念的相似度之最大值，也就是说：

$$\text{Sim}(W_1, W_2) = \max_{i=1 \dots n, j=1 \dots m} \text{Sim}(S_{1i}, S_{2j})$$

(2) 概念义原间相似度计算

我们需要通过聚类的方法把语义上相同的词语聚在一起，采用的聚类算法是 2.3.3.1 介绍的凝聚式层次聚类。在凝聚式层次聚类过程中，需要衡量两个概念义原的相似度以便决定是否需要合并。我们采用常用的 Average Link 方法来计算两个概念义原的相似度，公式如下：

$$\text{Sim}(\text{Sem}_p, \text{Sem}_q) = \frac{\sum_{i=1}^m \sum_{j=1}^n \text{Sim}(W_{pi}, W_{qj})}{m \cdot n}$$

其中， Sem_p 和 Sem_q 分别是两个概念义原； W_{pi} 和 W_{qj} 分别是概念义原 Sem_p 和 Sem_q 中的词汇。

(3) 概念义原和普通词场相似度计算

概念义原和普通词场的相似度计算发生在分类器的测试阶段，即我们把分类器本身的特征向量采用概念义原来表示，但把测试文档采用普通词场向量来表示。基于标准的余弦公式，它们之间的相似度如下计算：

$$\text{Sim}(\text{Sem}_r, S_i) = \cos(\text{Sem}_r, S_i) = \frac{|\text{Sem}_r \cdot S_i|}{|\text{Sem}_r| \cdot |S_i|}$$

当计算 (Sem_r, S_i) 的时候，将 Sem_r 和 S_i 中出现相同词语的维度进行对应；但在计算二者权重乘积的时候，把 Sem_r 中这个词语的权重用它所属的概念义原的权重来代替，这就使得 Sem_r 和 S_i 之间的相似度从概念的意义得到了扩大。

3.7.4 分类器的训练

我们的分类器采用简单高效的中心分类法，它的训练与分类速度很快，并且分类精度还可以接受，因此在文本分类尤其是海量数据分类中得到了广泛的应用 [Han 2000][Shankar 2000]。

这样，我们就从训练语料中训练出性别、民族、出生年月、籍贯、职位、住址、家庭关系 7 个中心向量和各自最佳的阈值以及数据集中人名和属性所间隔的最大窗口。我们采用“和中心”来表示中心向量，即 $\vec{C}_i^S = \sum_{d \in C_i} \vec{d}$ ，决策规则为

$$C = \arg \max_{C_i} (\text{sim}(\vec{d}, \vec{C}_i)) = \arg \max_{C_i} (\vec{d} \cdot \vec{C}_i)$$

选取训练样本中离中心最远的距离作为评判测试样本的阈值；选取训练样本中人名与各个属性所间隔的最大词数作为该属性的窗口。表 3.2 是人物属性训练语料中 tag1、tag2 所间隔的最大窗口长度。

表 3.2 人物属性项对应的窗口长度

人物属性项	窗口长度
性别	4
民族	4
出生年月	6
籍贯	8
职位	45
住址	8
家庭关系	12

3.7.5 分类器的测试

测试阶段的步骤主要包括：词法分析、触发词识别、组块解析和分类四个步骤。词

法分析我们调用现有的词法分析软件 *ICTCLAS*[*ICTCLAS*]，下面我们主要介绍后三个步骤。

3.7.5.1 触发词库的建立

为了从未经标注的语料中发现人物的属性，关键是建立一部触发词库。表 3.3 表示出我们为每种属性建立触发词库的方法。

表 3.3 人物属性触发词库的建立方法

人物属性项	触发词库
性别	Hownet 里标注为 “human 人,male 男” 或 “human 人,female 女” 的词语，且去掉了一些特别宽泛易混淆的词，如 “生” 等。
民族	全国 56 个民族的名称
出生年月	词性为 “t”，且以数字与 “年”、“月” 组成的词语
籍贯	词性为 “ns” 的词语
职位	Hownet 里标注为 “human 人,#occupation 职位” 的词语，且去掉了一些特别宽泛易混淆的词，如 “人” 等。
住址	词性为 “ns” 的词语
家庭关系	词性为 “nr” 的词语

这样，对于一个新的句子，先对其进行分词、词性标注和命名实体识别，然后扫描该句子，如果发现满足某触发条件的词语，则开始判断该属性是否是特定人物的属性。

3.7.5.2 基于规则的组块解析方法

组块解析是自然语言处理里面一项重要的基础性研究，它对于句法分析、信息检索、信息抽取和机器翻译都有重要的实践意义。在这里，我们主要关注的是把人物属性触发词对应的最大名词短语（MNP）识别出来。与英文相比，汉语的 MNP 识别更加困难，主要工作有：[李文捷 1995]利用边界分布信息构造概率模型来进行 MNP 的自动识别；[Angel 1995]利用统计和规则相结合的方法，构造名词短语部分分析器，以发现句子中所有的 “的” 字名词短语；[周强 2000]对输入文本进行成分组块分析预处理的基础上，通过充分利用名词短语边界分布信息和内部结构组成知识来自动识别汉语 MNP。

本文不对最大名词短语识别进行深入的研究，只是在词性标注的结果上利用简单的规则来识别人物属性对应的最大短语。我们的方法也是基于自然语言中存在的 “长尾综合效应”，即绝大多数事例采用经常出现的、非常少量的表达方式描述。表 3.4 是我们识别人物属性短语所采用的规则。

另外，我们还对并列的职位短语和人名短语进行了识别，即如果两个职位短语或两个人名之间的间隔词为 “、” 或 “和”，则它们之间属于并列关系。并列的短语共享相同的特征向量。

表 3.4 人物属性短语识别规则表

人物属性项	规则方法
性别	仅触发词即可
民族	仅触发词即可
出生年月	从触发词往后顺序找词性为“t”且中心为“月”、“日”的词
籍贯	从触发词往后顺序找词性为“ns”、“f”、“a”、“n”、“b”、“s”的词，遇到“人”结束
职位	从触发词往前顺序找词性为“n”、“nt”、“ns”、“nz”、“nl”、“ng”、“b”、“f”、“vn”、“m”、“q”、“s”、“nsf”、“ag”、“a”的词
住址	从触发词往后顺序找词性为“ns”、“f”、“a”、“ng”、“m”、“q”、“n”、“b”、“s”的词，遇到“人”结束
家庭关系	仅触发词即可

3.7.5.3 分类器的结果产生

在找到满足触发条件的词语，并识别出最大的属性短语 Attribute_i 以及与该短语处于并列地位的其它短语后，我们在允许的窗口 Window_i 范围内，寻找出现的人名。对于任一属性短语 Attribute_i ，如果属性左边 Window_i 窗口内存在着人名，则可以挖掘出对应的特征向量 $\text{Vector}_{\text{Left}i}$ ；如果属性右边 Window_i 窗口内存在着人名，则可以挖掘出对应的特征向量 $\text{Vector}_{\text{Right}i}$ 。特征向量的构造方法同 3.7.3.1。然后与训练好的分类器中的语义向量进行相似度运算。

这样，就可以计算出该属性短语对应的左侧向量的相似度 LSim 和右侧向量的相似度 RSim ， LSim 与 RSim 哪个大，且超过阈值，该属性就属于哪侧的人名（包括并列的人名）。

3.7.6 实验结果及其分析

在测试过程中，我们更关注高的准确率，因为在 Internet 网页数量趋向于无限大的情况下，准确率比召回率更重要。计算 F 值时，我们认为准确率的重要性是召回率的 2 倍，

即 $F = \frac{3 \cdot P \cdot R}{P + 2 \cdot R}$ 。实验采用 3 次交叉验证的方法。

3.7.6.1 实验结果的对比

表 3.5 和表 3.6 分别是采用语义计算前后人物各属性取得最大 F 值时对应的各项结果值。图 3.6 是二者的效果对比图。

表 3.5 采用普通词场向量的人物属性抽取效果

人物属性	阈值	答案总数	结果总数	正确数	准确率	召回率	F 值
性别	0.05	113	113	108	0.955752	0.955752	0.955752
民族	0.10	15	15	15	1.000000	1.000000	1.000000
出生年月	0.20	27	27	27	1.000000	1.000000	1.000000
籍贯	0.40	77	74	65	0.878378	0.844156	0.866667
职位	0.30	2301	1998	1973	0.987487	0.857453	0.939971
住址	0.35	25	13	13	1.000000	0.520000	0.764706
家庭关系	0.28	38	24	14	0.583333	0.368421	0.488372

表 3.6 采用概念义原向量的人物属性抽取效果

人物属性	阈值	答案总数	结果总数	正确数	准确率	召回率	F 值
性别	0.16	113	110	108	0.981818	0.955752	0.972973
民族	0.10	15	14	14	1.000000	0.933333	0.976744
出生年月	0.20	27	27	27	1.000000	1.000000	1.000000
籍贯	0.46	77	65	62	0.953846	0.805195	0.898551
职位	0.35	2301	1998	1973	0.987487	0.857453	0.939971
住址	0.41	25	14	14	1.000000	0.560000	0.792453
家庭关系	0.47	38	26	25	0.961538	0.657895	0.833333

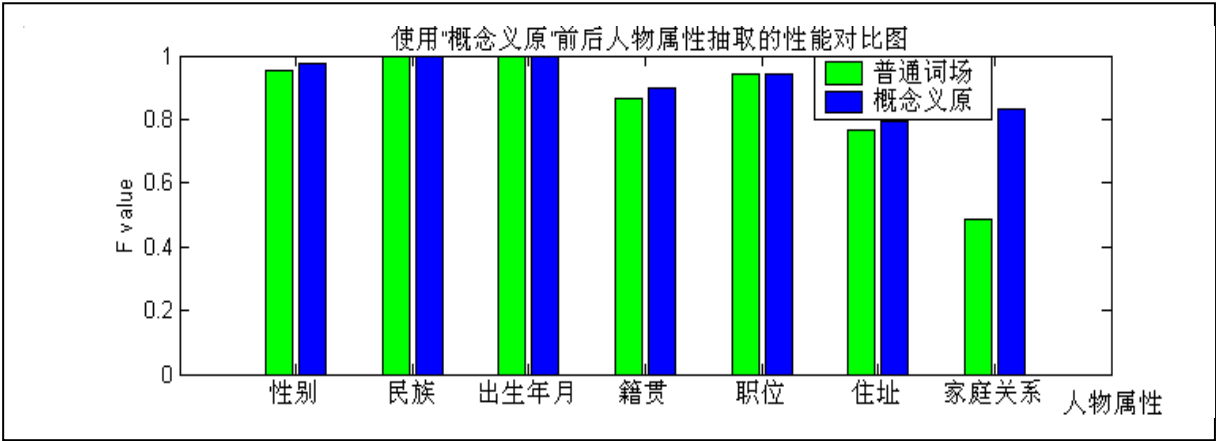


图 3.6 使用概念义原前后人物属性抽取的效果对比图

3.7.6.2 实验结果的分析

从上面结果可以看出，我们采用的人物属性抽取方法取得了比较好的效果。跟前人工工作比较起来，我们取得的实体关系抽取性能较高，原因是所使用的语料不同。我们的语料是严格按照互联网上新闻网页中各属性模式出现的频率来建立的，我们实验的结果

反映了采用该方法在实际互联网环境下，抽取人物属性所能取得的效果。

采用基于概念义原的特征向量后，各项属性抽取的结果都有所提高。其中，家庭关系抽取的性能提高的最为明显，因为家庭关系的特征词最为分散，例如有“弟弟”、“父亲”、“前夫”、“儿子”、“妻子”、“侄女”“夫妇”、“姐妹”等，只有通过概念义原才能把它们统一到一起，突出关键性的特征。

按照人物各个属性的结果逐个分析，可以看到：

(1) “性别”、“民族”、“出生年月”达到了很高的性能，是因为这两种属性的表述方式比较固定；

(2) “籍贯”、“职位”、“住址”、“家庭关系”可以达到较高的准确率，召回率略低，这是因为这四个属性的描述方式比较灵活，个别的例子系统没有识别出来。拿“职位”来说，因为在真实的新闻网页中，出现的职位短语绝大多数是与人名紧邻的，能够识别正确的职位基本上是与“人名”紧邻的“职位短语”；而不紧邻的情况识别得不够好，例如：

“[张林/nr] 今年/t 27/m 岁/qt， /wd 原/d 是/vshi 上海/ns 一/m 家/q [外企/n 白领/b]”，“聘任/v [张林/nr] 先生/n 为/p [公司/n 财务/n 总监/n]”

这些句子中“职位词”与“人名”离得太远，与职位密切相关的动词性特征如“是/vshi”、“聘任/v”等淹没在其它噪音词汇中，没有起到关键性的作用。

极少数识别错误的例子如：

“面对/v 百/m 位/q 民营/b 企业家/n， /wd [薄熙来/nr] 讲/v 了/rule 省委/n、 /wn 省政府/n 的/ude1 期待/vn”——把“民营/b 企业家/n”看作“薄熙来/nr”的职位；

“小/a 歌手/n [陆放/nr] 和/cc [张艺谋/nr] 申/vg 博/ag 片/ng”——把“小/a 歌手/n”识别成了“张艺谋/nr”的职位，因为“小歌手”是“陆放”的职位，“张艺谋”和“陆放”识别成了并列人名短语。

3.8 本章小结

在这一章里，我们介绍了针对汉语的新闻网页，人物追踪数据预处理的主要流程。数据预处理主要解决的是把蕴涵在文本中的人物相关信息挖掘出来，建立起人物模型的片断，为后续的处理做准备，其内容主要包括：网页噪音过滤、人名的识别、人名指代的处理、人物对应语段的确定、时间和时序标签的确定、人物属性的抽取。

在网页噪音过滤阶段，我们提出了综合利用 TABLE 标记和视觉特征对页面进行解析的算法 TVPS，并用该算法从新闻网页中提取出新闻报道的正文、标题和报道时间等信息，过滤掉广告、链接等噪音。

人名的识别我们直接使用中国科学院计算技术研究所研制的汉语词法分析系统 ICTCLAS，把词性标签为“nr”、“nrf”的词语视为人物的姓名，作为人物信息挖掘的触发词汇。

我们只是采用了若干简单有效的规则来进行人名指代的处理、人物对应语段的挖掘和时间标签的确定。

最后，我们采用监督式机器学习的方法来抽取人物的属性，即手工标注一定规模的语料，然后训练出分类器用来判断新的实体对是否存在搭配关系。抽取的人物属性有七类：性别、民族、生日、籍贯、职位、住址、家庭关系。分类器采用特征向量的表示方法。为了消除描述同一概念的词汇在字面上的差异，我们提出了基于“概念义原”的特征向量表示形式，实验证明，该方法能有效提高人物属性抽取的效果。

经过数据预处理阶段，我们就识别出了人物的若干名称信息、基本属性信息、介绍性信息和普通词场信息，把这些信息填充进人物模型，从而形成独立的人物模型片断。

第四章 人物模型的同一性判别与动态填充

在前面章节介绍的人物追踪数据预处理流程只是识别出若干人物相关的零散信息，为人物模型的计算做好了准备，还没有形成完整的人物模型。人物作为抽象对象的典型代表，必须有效解决同一性判断问题和数据融合问题，才能凸现出人物这个抽取对象实体，进而为跟踪问题打下良好的基础。

在这一章里，我们将对人物对象挖掘的同一性判断问题和数据融合问题进行详细的论述。首先介绍研究的背景，然后介绍前人的相关工作。接下来，我们介绍了人物模型的特征表示和相似度的计算方法，在此基础上进行同一性判别和人物模型数据的融合。最后给出相关实验的结果与分析。

4.1 概述

在文本空间内，人物是比文字对象高一个层次的抽象实体。作为处于抽象层次的实体，它与文字对象所描述的具体名称并不形成一一对应关系。在文本中，存在着大量的同名同姓却不同人物的现象；即使是同一人物，也往往有不同的称谓。数据预处理阶段形成的只是人物模型的片断，我们需要对这些片断性的信息进行同一性的判别，即把描述不同人物的模型片断区分开。

另外，数据预处理阶段形成的人物模型片断信息是分散的、孤立的，不能代表人物的全貌，无法凸现出人物这个抽象实体。因此，我们不仅要把代表不同人物的模型片断区分开，还要对同一人物的模型片断信息进行有机整合，形成完整的人物模型，清晰地呈现人物的履历等信息。对于一个人物实体来说，它可能具有多方面的属性特征，每个文本只描述了其中一部分。为了取得完整的信息，可能需要参照多个来源的数据才能补齐，因此需要把多个人物模型片断的信息融合在一起。

然而，在文本提供的人物相关信息有限的情况下，如何有效地对人物进行同一性判别，并不是一件容易的事。一般来说，需要对人名出现的上下文进行某种划分，使得在同一分区里的人名，在真实世界里“指称”同一个人。我们说，这样的上下文分区与一个“人物”建立起了一一对应的关系，或者说在这个分区模型下获得了“人物”的概念。在文本中，对人物的区分作用较大的特征包括人名、人物的属性信息以及人名周边的普通词场等。值得说明的是，在人名周边的语句中，与该人经常共现的其它人名，即该人物的交往人群是一个重要的区分特征。

人物模型的同一性判断和数据融合是互相作用，相辅相成的。同一性判断是数据融合的前提和基础，只有进行同一性判断，才能为一个人物的数据融合提供信息来源；数据融合是提高同一性判断性能的关键，因为关于一个人物的初始信息是少量和片面的，只有不断地通过数据融合来得到充实，才能取得代表该人物的关键性特征，从而为同一

性判断过程带来重要的影响。

因此，本文关于人物实体的同一性判断问题和数据融合问题是放在一起研究的，因为从计算的角度来看二者是密不可分的过程，我们称之为“人物模型的同一性判别与动态填充”。

4.2 相关研究工作

在本文中，我们所关注的人物模型的同一性判别是在事先没有先验知识的情况下进行的，属于非监督式机器学习的过程。这样，同一性判别大体就是一个聚类的过程，总的思路是观察围绕这个人物姓名的“词场”，通过词场的某种特征反映不同的人物实体。不过与普通的聚类场景相比，人物模型的同一性判别有其自身的特点。不同的人物对应的词场划分不尽相同，词场内部的各个特征所起的作用也不同，所以只能结合具体的场景对普通聚类算法进行调整，是聚类问题的特殊应用。

因为人物模型片断中有各种类型的数据，有的是通过信息抽取得到的，也有的是直接转化成向量空间产生的。因此，同一性判别必须有效地对这些混合类型的数据进行处理。在这一节，本文介绍前人的相关研究工作。

4.2.1 混合类型数据的聚类

人物同一性判别大体是一个聚类的过程。聚类分析已经被广泛地研究，它能把数据点分成若干个组，使同一组内的数据点相似度比较大，不同组内的点相似度较小。传统的聚类方法假定数据点都是数值化的特征[Duda 1973][Jain 1988]，然而当聚类的应用从科学和工程领域扩展到医学、商业和社会等领域时，就不得不处理属性特征，例如颜色、形状、性别等。尽管这些属性之间的相似度也可以转化为数值形式，然而任何类似的转换都是困难和主观的。为此，人们进行了专门的研究。对属性特征进行聚类的方法再结合特征描述发现的过程被称之为“概念聚类”(Concept Clustering) [Fisher 1986]。

在属性聚类方面，比较典型的算法有：K-modes、STIRR、CACTUS、ROCK 等。K-modes 可以看作是 K-means 算法向属性聚类方向的扩展[Huang 1997a] [Huang 1998]，它采用“模式”来代替“均值”以便处理类属性特征，并采用基于频率的方法在聚类过程中来更新模式。在 K-modes 算法基础上，[Jollois 2002]提出了一个适应性混合模型来对属性特征进行聚类，这样就给 K-modes 算法的评价标准提供了概率解释。[Huang 1999]提出了 fuzzy K-modes 算法，然后一种限制搜索策略被应用在[Ng 2002]来提高 fuzzy K-modes 的效果。

STIRR 是一个基于非线性动态系统的迭代算法[Gibson 1998]，属性聚类被看作是图分解的过程。它的运行时间线性于记录数目，并接近线性于属性数目。如果动态系统收敛，类属性特征就可以被聚类。

CACTUS 采用了比较新颖的数据摘要的方法来进行属性聚类[Ganti 1999]，它包括三

个阶段：建立摘要、聚类 and 确认。CACTUS 算法随着属性数目的增长，时间复杂度呈指数级增长，这也限制了其在属性数目很大的情况下的聚类应用。

ROCK 是一个改进的层次凝聚式聚类算法[Guha 1999]，它采用布尔向量对关系数据库中的记录进行描述，并把两条记录共同拥有的相似邻居数量作为它们之间的相似度的值。ROCK 的时间复杂度是数据记录数的立方，为了处理大的数据量，ROCK 需要采样数据，但这样可能妨碍发现一些含有重要特征的小类。

对于人物追踪来说，预处理阶段生成的人物模型片断中，属性信息往往是稀疏、不充分的，还无法完全脱离普通文本的数值相似度计算，应该采取普通词场的数值相似度计算与属性特征的布尔匹配计算相结合的方法。在聚类研究中，只有很少数算法关注过混合类型数据的聚类，大多数的解决方法集中于以下三类：

① 把属性特征转化为数值特征，然后采用数值特征的相似度（距离）计算公式来计算。然而，在大多数情况下这样的转换是很难找到实际意义的。另外，转换的过程也往往带来计算维度的增加，并造成聚类结果的下降[Hinneburg 1999][Hinneburg 2000]。

② 把数值特征离散化，然后应用属性聚类的方法，例如 K-modes[Huang 1997a]。然而，在离散化过程中经常伴随着重要信息的损失，因而造成聚类精度的下降。这种离散化的损失在监督式机器学习过程中已经得到证实[Ventura 1995]。

③ 概括评价函数来同时衡量数值特征和属性特征，主要的难点在于如何概括评价函数上。K-Prototypes 是这方面比较典型的一个算法[Huang 1997b]，它是一个基于划分的方法，主要是定义了一个混合距离函数来处理混合类型的数据。函数中定义了权重系数来衡量两种类型数据的重要程度，形式化表示如下：

$$d(X_i, Q_l) = \sum_{j=1}^{m_r} (x_{ij}^r - q_{lj}^r)^2 + r_l \sum_{j=1}^{m_c} \delta(x_{ij}^c, q_{lj}^c)$$

4.2.2 IE 在 IR 中的应用

信息检索（IR）的目的是根据用户的查询请求从文档库中找出相关的文档，用户必须从找到的文档中翻阅自己所要的信息。而信息抽取（IE）是从文档中取出相关的信息点。这两种技术是互补的，若结合起来可以为文本处理提供强大的工具[Gaizauskas 1998]。

在人们的日常生活和学习、工作、科研中，经常遇到一些关于简单事实或概念定义的问题，他们需要的是对这些问题的简短精确的答案，而不是一个通过搜索引擎可以得到的可能含有用户所需信息的文档集合。在这种情况下，信息抽取就被应用在了信息检索技术里并发挥了巨大的作用，从而形成了一个新的研究方向——知识问答（QA）。知识问答是指从一个有限大的文档集合 $DS = \{d_1, d_2, \dots, d_{|DS|}\}$ 中针对用户用一个自然语言语句所提出的问题 q ，找出简短精确的答案 a 的过程。一般来说，基于 WEB 的汉语知识问答系统可分为六个步骤：形成查询请求，判断问题所期待答案的类型，相关文档查询，候选答案片断（snippet）抽取，答案抽取，答案选择。

信息抽取不仅可以象知识问答那样应用在信息检索技术中来产生新的应用领域，还可以直接用来提高信息检索技术的性能。因为信息检索通常利用词的统计信息来识别文档，这样可以对文档进行大规模处理。然而，基于词的技术存在着好多缺陷，例如，语言里有好多同义词、多义词、固定搭配、表达方式变换等情况。尽管在信息检索领域有多种方法，例如词汇消歧，来解决这一问题，然而所起的作用是有限的。因此，在一些准确度要求很高的应用场景下，研究者就开始尝试采用信息抽取的手段来提高检索技术的效果。

比较典型的是[Riloff 1994]中的工作，研究者把信息抽取的结果应用在高精度文本分类中，所采用的方法基于这样的观察：对于某些特定领域的文档分类问题来说，一个相关的句子就足以认为该文档是相关的；而一旦这个相关句子被识别出来后，文档中其它的内容就可以忽略了。因此，研究者针对“谋杀”事件定义了若干模板，采用该模板进行相关实例的抽取，从而识别这篇文档是否含有恐怖主义谋杀事件的报道。在实验中，所采用的模板结构越来越灵活和复杂，那么该方法所能识别出的事件描述种类也就越来越多。实验结果表明，采用信息抽取与检索技术相结合的方法能够有效克服词统计方法的缺陷，对特定领域的文本分类效果有很大的提高。当更换新的领域知识时，该方法可以很轻易地移植到新的应用领域。

4.2.3 人名消歧的主要方法

人名的消歧是自然语言里面“共指消歧”[Hobbs 1978]的一种类型，即识别出人名和它的指代词之间的对应关系。随着信息处理的不断加深，人名消歧也由单文档消歧[Wacholder 1997]扩展到跨文档消歧方面[Bagga 1998][Fleischman 2004][Niu 2004][Ravin 1999]。其中，[Fleischman 2004]所采用的方法分为两个步骤：首先采用最大熵模型来产生两个同名关系对间的共指概率，然后用一个改进的层次凝聚式聚类方法来确定最终的结果。[Niu 2004]采用了概率模型来进行跨文档人名消歧，在消歧过程中利用了信息抽取的特征。

[Mann 2003]通过定义信息抽取模板来抽取出人物的若干属性特征，然后把各种特征应用在层次凝聚式聚类中。研究者比较了四种模型：全部词模型、相关词模型、属性特征模型和扩展属性特征模型。“全部词模型”指的是用所有的词汇来进行人名消歧；“相关词模型”是指对词汇先根据互信息进行特征选择，只把重要的词汇用来进行人名消歧；“属性特征模型”是指采用信息抽取出来的人物属性进行消歧；“扩展属性特征模型”是指采用信息抽取的候选特征来进行消歧。研究者人工制作了一个模拟数据集，实验发现“属性特征模型”与其它模型相结合，能够产生较好的结果。

[Xiaojun 2005]重点考查了在搜索引擎的返回结果中人名的消歧问题。他们同样采用层次凝聚式聚类的方法，把词汇特征、命名实体特征和属性特征的相似度线性融合在一起。类似地，[Al-Kamha 2004]同样考查搜索引擎返回结果中人名的消歧，然而他们却采用了不同的特征集合，包括：人物属性集合、页面链接集合和页面相似度集合。然后用

这些集合构建了一个确定度矩阵，并在矩阵内部进行语料对的合并。

与其它研究者不同的是，[Guha 2004]考查了不同的应用场景。他们不是采用聚类方法，而是基于重新排序的方法来消歧人名。该算法要求用户选择一个搜索引擎的返回页面作为初始点，然后系统重新排序搜索引擎的返回页面，使倾向于描述同一人的页面排在前边。在这个场景下，用户需要逐个浏览搜索引擎的返回页面来选择初始点，使用上可能会不太方便。

[Bekkerman 2005]提出了两种非监督式算法框架来发现那些指向同一个人物的同名页面：一个基于页面的链接关系，另一个采用一种多路分布式聚类方法（MDC）。然而，他们的应用场景局限于判别一个已知的社会网络中同名人物的页面，在这个网络中人们是需要有一定联系的。

在本文中，我们关注的同样是互联网真实环境下人物的同一性判别问题。我们的算法是采用非监督的方式来进行的。与传统的混合类型数据聚类方法相比，我们所使用的类别属性特征不是事先就有的，而是通过信息抽取的手段提取出来的；我们尝试把信息抽取的结果应用在文本聚类技术中，来提高聚类系统的效果；我们的算法可以看作是上述诸多人名消歧算法的扩展，例如传统的消歧方法虽然也着重考查了人物属性在人名消歧中的作用，然而往往把多种属性特征混合在一起进行计算，没有具体分析各个属性的分布特点及其对人物同一性判别的不同作用。另外，人名周边词场中的不同特征表征了人物不同方面的信息，例如，共现人名就反映出该人物交往密切的人群特征。事实上，人物的不同维度的特征构成了一个完整的模型。为此，我们提出了“人物模型”的概念，并详细研究了互联网环境中人物模型不同维度的分布规律，以及人物模型之间的相似度计算方法和数据融合方法。在此基础上，我们还详细研究了人物同一性判别的结果呈现问题，例如整合出人物的履历信息等。

本文只关注“同名人物判别”的问题，不对“相同人物不同称谓的现象”进行研究。下面的章节将详细介绍本文所采用的相关方法。

4.3 人物模型（Person Model）的特征表示

如前所述，人物模型主要由四部分组成，即 $Person = \{N, P, Q, R\}$ 。 N 为人物名称（姓名、别名、称谓等）集合， $N = \{N_1, N_2, \dots, N_n\}$ ； P 为人物基本属性集合， $P = \{P_1, P_2, \dots, P_k\}$ ； Q 为人物介绍性信息集合， $Q = \{Q_1, Q_2, \dots, Q_m\}$ ； R 为词场集合， $R = \{R_1, R_2, \dots, R_l\}$ 。

设 $X = \{X_1, X_2, \dots, X_N\}$ 为数据预处理阶段得到的人物模型片断集合。对于 $X_i (1 \leq i \leq N)$ ，就我们关注的人名和七个人物属性（性别、民族、出生年月、籍贯、家庭住址、职务、家庭关系）以及词场而言，姓名 $\in N$ ，{民族、出生年月、籍贯} $\in P$ ，{家庭住址、职务、家庭关系} $\in Q$ ，{人名所在的句子、段落、文本，跟该人名经常共现的其它人名}

$\in R$ 。图 4.1 是人物模型片断的示意图。

	性别	民族	生日	籍贯	住址	亲属	职务	共现人名	普通词场
Name1	男	汉	1949		北京		演员	...	
Name2	女			山东		王红	教师	...	
Name3	男	蒙			安徽		书记	...	
Name4	男	彝	1970	河北		李莉	经理	...	

图 4.1 人物模型片断示意图

4.3.1 属性特征表示

在集合 N 、 P 、 Q 、 R 中， N 、 P 、 Q 为属性特征集合，然而每个属性特征的维度是不同的。设 DOM 为属性特征的维度，则 $DOM(N_i)=1$ ($1 \leq i \leq n$)，因为人物的称谓往往是唯一的； $DOM(P_i) \geq 1$ ($1 \leq i \leq k$)，因为人物的基本属性信息，例如民族，会有多种情况； $DOM(Q_i) \geq 1$ ($1 \leq i \leq m$)，因为人物的每项介绍性信息，例如职业，同样变化多样。

对于同一个人物而言，集合 $P = \{P_1, P_2, \dots, P_k\}$ 中每个小项值是唯一的，对于采用信息抽取（IE）来获取人物模型数据的情况，每一项对应一个“单槽”的值；集合 $Q = \{Q_1, Q_2, \dots, Q_m\}$ 中每个小项都可是一个可变的集合，每一个小项都对应着信息抽取里面一个“多槽”的值。

4.3.2 数值特征表示

人物模型中，词场集合 $R = \{R_1, R_2, \dots, R_l\}$ 为数值特征集合。在文本处理中，常用的表示模型有：向量空间模型、概率模型和语言模型。我们采用常见的向量空间模型来表示 R_i ($1 \leq i \leq l$)。

向量空间模型（VSM: Vector Space Model）将文档视为向量。向量的每一维代表一个信息表达元素，它可以是汉语中单个的字、也可以是任何语言中的一个词或者短语，也可以是没有实际意义的 n 元组。这些元素通过对数据集中的每一个文档进行处理而得到[许洪波 2003]。在向量空间模型中，不同的特征的重要程度不同，用特征 t_k 附加权重 w_k 来进行量化，可表示为 $(t_1, w_1; t_2, w_2; \dots; t_H, w_H)$ ，简记为 $D(w_1, w_2, \dots, w_H)$ 。这时我们说特征 t_k 的权重为 w_k , $1 \leq k \leq H$ 。

我们需要为向量空间模型进行特征的选取。首先确定人名周边词场的作用范围。我们考察了三种词场的范围：人名所在的整篇报道(Text)，人名所在的段落(Para)，人名所在的句子(Sentence)。在每种词场中，我们都进行了汉语的分词、词性标注和命名实体识别，然后把人物的属性信息词汇排除掉。在此基础上，进行停用词（性）处理和特

征选择。

停用词（性）处理主要目的是去除分辨力很低的词语，主要做法是只保留词场中的名词 N、动词 V、形容词 J、副词 R 以及命名实体 NE。针对这些保留的词性，再定制一份停用词表，把个别满足词性要求但分辨力差的词过滤掉。然后，我们采用基于文档频率的特征选择方法，把出现频率特别高和特别低的词过滤掉，只保留 DF（Document Frequency）值满足 $2 \leq DF \leq N/3$ （N 为全部文档个数）的词汇。

对于人物追踪来说，词场中的人名 NR 具有特殊的意义，因为它们在一定程度上能够代表该人物与其他人物的密切程度。因此，我们把词场中的词表示成两个向量：普通词场向量和共现人名向量。针对每个向量，采用了不同的权重计算方法。

4.3.2.1 普通词场向量的权重计算

普通词场权重计算采用常用的 TF_IDF 计算方法。TF(Term Frequency)代表词频率，IDF(Inverse Document Frequency)代表文档倒排频率。TF_IDF 方法存在很多变种，我们采用的权重估算公式为：

$$w(t, \vec{d}) = \log(tf(t, \vec{d}) + 1) \times \log(N / n_t + 1.0)$$

其中， $w(t, \vec{d})$ 为特征词语 t 的权重，而 $tf(t, \vec{d})$ 为特征词语 t 在文档 $\vec{d}$ 中出现的频率， N 为文档总数， n_t 为训练文档集中出现 t 的文档数。

在公式中， $\log(tf(t, \vec{d}) + 1)$ 反映的是特征 t 在文档中的重要程度，反映的是局部影响，不同文档中的同一特征 t 的 $\log(tf(t, \vec{d}) + 1)$ 值不同；而 $idf(t) = \log(N / n_t + 1.0)$ 反映的是特征 t 在整个向量空间中相对于其他维的重要程度，它反映的是一种全局影响，每个特征的 idf 值是在样本训练时计算出来的，只要训练集相同，不同的文档同一个特征的 idf 值相同。

4.3.2.2 共现人名向量的权重计算

采用向量空间模型来描述一篇文档时，往往把该文档中的词汇作为特征，并为每个词汇赋予一定的权重。类似地，人名的共现标识了不同人物之间关系的密切程度，每一个人物都有一定的交往人群。对于两个同名的人物来说，在一定程度上可以利用他们交往人群的不同来区分。我们规定，当两个人名出现在文档中同一个段落中时，他们构成共现关系。

与词场向量类似，我们引入了“共现人名向量”的概念。共现人名向量描述的是人物间关系的信息，它把与当前人名共现过的其它人名作为特征。影响特征权重的因素有两个：NF（Name Frequency）代表“人名频率”，即人名的共现次数越大，该人名与当前人物的关系越倾向于密切；IPF（Inverse Person Frequency）代表“倒排人物频率”，即某一人名与其它人名共现得越多，他/她的交际面就越广，那么他/她与当前人物的关系越倾向于不密切。共现人名向量中每个维度的权重计算公式为：

$$w(name, \bar{P}) = \log(nf(name, \bar{P}) + 1) \times \log(N' / n_{name} + 1.0)$$

其中, $w(name, \bar{P})$ 为共现人名 $name$ 的权重, 而 $nf(name, \bar{P})$ 为人名 $name$ 与人物 $\bar{P}$ 的名字共现的频率, N' 为训练文档集合中的人名总数, n_{name} 为训练文档集合中与人名 $name$ 共现过的其它人名数。

4.3.3 各项特征的分布规律

在互联网的真实环境中, 人物模型的各项特征往往不是完全的, 稀疏的程度也不同。例如: “性别”、“民族”、“家庭关系”、“籍贯”、“出生年月”、“住址”就特别的稀疏。表 4.1 是数据预处理阶段我们提取出来的 4315 个人物模型片断中, 各项属性的出现次数和稀疏程度。稀疏程度直接会影响其在人物模型同一性判别中的作用。

表 4.1 4315 个人物模型片断中各个属性的出现次数和稀疏度

	性别	民族	籍贯	出生年月	家庭关系	住址	职务
次数	113	15	77	27	38	25	2301
稀疏度	0.026	0.003	0.018	0.006	0.009	0.006	0.533

4.4 人物模型的相似度计算方法

人物模型 $Person = \{N, P, Q, R\}$ 包含着多个维度, 每个维度的相似度计算方法不尽相同, 且对人物模型总的相似度计算所起的作用也不同。我们规定, 在人物模型的同一性判别中, 进行相似度计算的两个人物模型, 它们的名称集合 N 一定是相同的。在此基础上, 我们讨论其它维度数据的相似度计算方法。

4.4.1 基本属性的相似度计算

因为基本属性信息对于一个人物来说是终生不变的, 所以人物基本属性集合 $P = \{P_1, P_2, \dots, P_k\}$ 中, 每一项的值都是唯一的, 即 $P_j = x_j$ ($x_j \in \text{DOM}(P_j), (1 \leq j \leq k)$)。

为了简化和抽象, 我们把“生日”、“籍贯”等信息进行规格化, 例如“生日”最多保留到月份, “籍贯”只保留省份, 其它的抛弃掉。这样, 基本属性信息, 例如“性别”、“民族”、“籍贯”、“生日”的相似度可表示为:

$$\delta(x_j, y_j) = \begin{cases} 1 & (x_j = y_j) \\ 0 & (x_j \neq y_j) \end{cases} \quad x_j \in \text{DOM}(P_j) \wedge y_j \in \text{DOM}(P_j), (1 \leq j \leq k)$$

生日的相似度稍微复杂, 举例来说, 会出现“1949 年”和“1949 年 7 月”这两种情

况，在上述公式中，“ $x_j \neq y_j$ ” 针对生日来说是出生的年份不相匹配。

4.4.2 介绍性属性的相似度计算

介绍性属性信息 $Q = \{Q_1, Q_2, \dots, Q_m\}$ 对于一个人物来说是会发生变化的，例如“住址”、“亲属”、“职位”，每一项的取值都对应着一个集合，即 $[Q_j = x_{j1}] \vee [Q_j = x_{j2}] \vee \dots \vee [Q_j = x_{js}]$ ($x_{jk} \in Q_j, (1 \leq j \leq m, 1 \leq k \leq s)$)。

设两个人物模型中介绍性信息的某一维 Q_j 的取值集合分别为 $X_j = \{x_{j1}, x_{j2}, \dots, x_{js}\}$ ， $Y_j = \{y_{j1}, y_{j2}, \dots, y_{jl}\}$ ，则这两个维度的相似度计算方法为：

$$Sim_Q(X_j, Y_j) = \sum_{i=1}^s \delta(x_{ji}, Y_j) = \sum_{i=1}^s \max\{\delta'(x_{ji}, y_{jk}) | x_{ji} \in X_j, y_{jk} \in Y_j\}$$

相对于人物的基本属性信息来说，介绍性信息的描述形式灵活复杂。举例来说，职位的描述就有：“所长”、“中科院计算所所长”和“中国科学院计算技术研究所所长”等多种形式。这种局面是由于汉语名词短语结构的复杂性造成的。

汉语语言学界对名词短语结构尚未形成成熟而统一的意见。但汉语的名词短语一般是由中心词和修饰词构成。[朱德熙 1982]在分析汉语中的偏正结构时对名词短语进行了分析，指出不能光凭中心词的性质来判定名词短语，还要考虑修饰语的性质以及整个结构所处的语法位置。[张卫国 1996]从自然语言处理的视角对汉语名词短语进行槽位分析，认为名词性短语的中心词前边有三个槽位，分别充当区分性、描写性或限定性定语，即[区分性定语][描写性定语][限定性定语][中心词]。其中，“限定性定语+中心词”结构被称为汉语的基本名词短语[赵军 1998]。

中心词前边可以有多个限定性定语。从意义上，限定性定语是对中心词的限制和认定，在中心词所指的人或事物中限定一个小类；从集合论的观点看，中心词表示一个集合，加上一个限定性定语，就得到这个集合的一个真子集；从词语的语义关系上看，中心词表示一个上位概念，加上一个限定性定语，就确立了一个下位概念，限定性定语越多，整个短语表示的概念的位置就越低[朱德熙 1982]。

在本文中，我们不对名词短语的结构做深入的研究，只是将名词短语 x 看作是由中心词 $H(x)$ 和修饰词 $A(x)$ 构成，即 $x = A(x) + H(x)$ 。当 $A(x)$ 为空值时，我们把 x 看作是“短名词短语”，相反则为“长名词短语”。

在这样的短语类型划分基础上，我们按如下方法来计算两个名词短语的相似度：

- ① 两个“短名词短语”，只要中心词匹配，它们的相似度就是 1，否则是 0；
- ② 两个“长名词短语”，在中心词匹配的情况下，还要计算修饰词的匹配程度。我们以字为单位，计算二者修饰词的交集，再除以字的并集；如果比值大于 η ，

则为 1，否则为计算出的 η 值（ η 为预设参数）。

- ③ “短名词职位”和“长名词短语”，只要它们的中心词匹配，相似度就是 1，否则是 0。

两个介绍性属性 x_{ji} 、 y_{jk} 的相似度计算公式可表示为：

$$\delta'(x_{ji}, y_{jk}) = \begin{cases} 1 & (H(x_{ji}) = H(y_{jk}); \text{当 } x_{ji}, y_{jk} \text{ 都是长短语时,} \\ & \text{Overlap}(A(x_{ji}), A(y_{jk})) \geq \eta) \\ \text{Overlap}(A(x_{ji}), A(y_{jk})) & (H(x_{ji}) = H(y_{jk}); x_{ji}, y_{jk} \text{ 都是长短语,} \\ & \text{Overlap}(A(x_{ji}), A(y_{jk})) < \eta) \\ 0 & (H(x_{ji}) \neq H(y_{jk})) \end{cases}$$

其中， $\text{Overlap}(u, v) = \frac{|u \wedge v|}{|u \vee v|}$ ； η 为经验阈值，取 0.32 效果较好；在实际做法中，

我们对修饰词中一些出现频率特别高的字，例如职位短语中的“市省有限责任公司区县乡”进行了权重的减半，因为出现得太频繁的字对短语的区分力较弱。

按照上面的相似度计算方法，职位短语相似度举例如下：

$$\delta'(\text{中科院计算所所长, 所长}) = 1$$

$$\delta'(\text{中科院计算所所长, 中国科学院计算技术研究所所长}) = 1$$

$$\delta'(\text{中科院计算所所长, 中国科学院自动化技术研究所所长}) = 0.1$$

因为“住址”信息的表述方式过于灵活，很难客观地衡量它们的相似度，所以“住址”信息没有参与我们的人物模型同一性判别的计算。

4.4.3 词场的相似度计算

词场集合 $R = \{R_1, R_2, \dots, R_l\}$ 中，我们采用向量空间模型来表示 $R_i (1 \leq i \leq l)$ 。在向量空间模型中，计算两个向量之间的相似度有多种方法。如 2.3.2.1 所介绍的，最简单的是直接计算两个向量的内积；最常用的是计算向量的夹角余弦函数；除此之外，还可以采用距离计算公式，例如明考斯基距离、欧式距离、曼哈顿距离、马氏 (Mahalanois) 距离和兰氏 (Lance) 距离等。

我们采用最常用的夹角余弦的计算方法，即：

$$Sim(X,Y) = Cos(X,Y) = \frac{\sum_i x_i * y_i}{\sqrt{(\sum_i x_i^2) * (\sum_i y_i^2)}}$$

4.4.4 人物模型总的相似度计算

人物模型 $Person = \{N, P, Q, R\}$ 包含着多个维度，因此计算两个人物模型的相似度会有很多的方法。最简单的就是不考虑其它维度，只根据词场集合 R 采用夹角余弦来计算。然而，仅根据词场的特征就把人物模型的同一性判定做好是很难的。

同名的两个人物，描述他们的文本特征往往非常相似，例如：同是游泳运动员的两个人物“李杰”，以致于词场模型很难把同名的人区分开；同一个人物可能存在着方方面面的报道，可能涉及不同领域的内容，词场模型的特征又很难把同一个人物的信息有效联系在一起。因而，词场集合本身无法有效解决同名判别的问题，图 4.2 是单纯采用词场集合 R 来对“薄熙来”进行同名人物判别的结果。

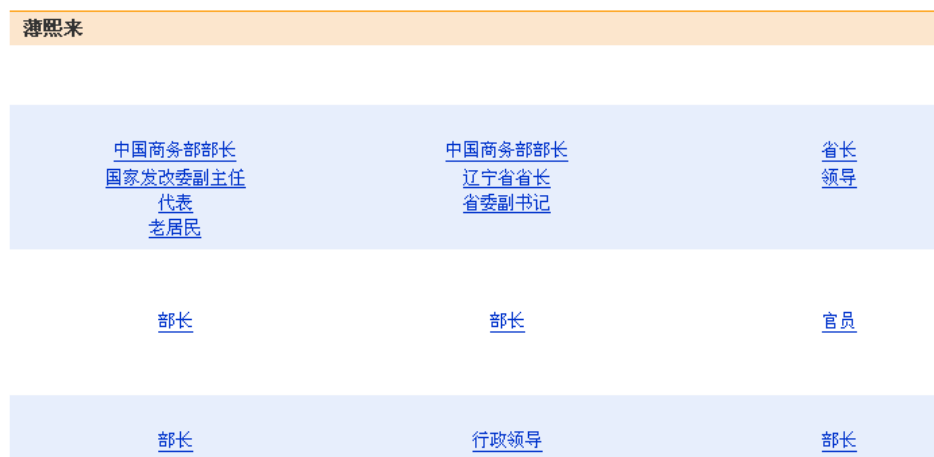


图 4.2 单纯采用词场 R 对“薄熙来”进行同名判别的结果

从图 4.2 可以看到的是，尽管很明显地，第一、二个结果实际指向的是同一个人，然而由于文本转化成向量空间模型时所固有的信息损失，使得系统无法有效进行同名人物的判别。

我们试图综合利用人物模型多个维度的特征来克服词场集合的缺陷。在人物模型 $Person = \{N, P, Q, R\}$ 中，属性集合 P 、 Q 中的信息对人物的标识作用是很强的，因为是通过信息抽取得到的结果。所以，我们可以通过信息抽取结果之间的匹配产生一些紧密的团；以这些紧密团为“枢纽”，再通过词场相似度的计算来产生最终的结果。图 4.3 是我们所使用方法的示意图。

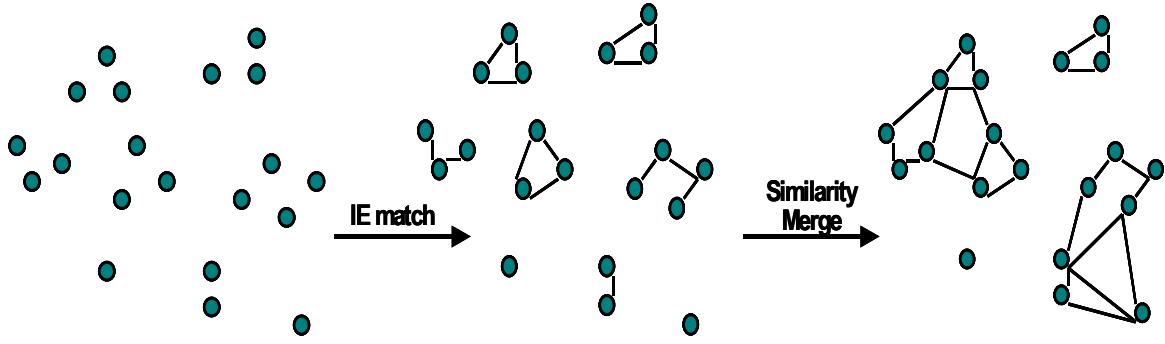


图 4.3 同名人物判别方法示意图

从我们关注的几个人物属性来看,“性别”、“民族”、“籍贯”、“出生年月”、“亲属关系”是非常稀疏的,稀疏度都在 0.03 以下;职位的稀疏度是 0.533。然而,由于属性特征对人物的标识力很强,确实可以形成如下强匹配的规则:

规则 1: 如果姓名相同的两个人物模型,生日相匹配(精确到月份),或亲属关系匹配其一,或长职位匹配其一,则这两个人物模型指向同一个人。

规则 2: 如果姓名相同的两个人物模型,性别、民族、籍贯、出生年有其一相抵触,则这两个人物模型指向不同的人。

在人物模型中,不稀疏的项包括:“职位”、“共现人名”、“普通词场”。对于两个同名的人物模型 X 、 Y ,我们先利用规则 1 和规则 2 进行判断,如果两条规则都不满足,则分别计算相似度 $Sim_{Position}(X,Y)$ 、 $\cos_{word}(X,Y)$ 、 $\cos_{name}(X,Y)$,然后考虑如何把三者融合起来。

我们假定“职位”、“共现人名”、“普通词场”三个因素是独立的,采用斯坦福确定性理论(*Stanford Certainty theory*) [Luger 1997]来进行三者的融合。斯坦福确定性理论定义了一种衡量确信度的方法,能够用一些简单的规则把几个独立的因素融合在一起。假定 E_1 、 E_2 、 E_3 是事务 B 的三个独立的因素,CF 代表确信度,则事务 B 的确信度 $CF(B) = CF(E_1) + CF(E_2) + CF(E_3) - CF(E_1) \times CF(E_2) - CF(E_1) \times CF(E_3) - CF(E_2) \times CF(E_3) + CF(E_1) \times CF(E_2) \times CF(E_3)$ 。

例如,如果三个因素对判断一件事务的可信程度分别是 88%、74%、66%,则它们混合起来后的可信度为 $88\% + 74\% + 66\% - 88\% \times 74\% - 88\% \times 66\% - 76\% \times 66\% + 88\% \times 74\% \times 66\% = 98.93\%$ 。

我们事先训练三项各自对应的阈值 μ_i ($0 < i < 4$),认为 E_i 的相似度只要达到该值,两个模型指向同一个人物的可信度就是 100%,即

$$CF(E_i) = \frac{Sim_{E_i}}{\mu_i}, \quad CF(E_i) \in [0,1]。$$

训练 μ_i 的方法是:按每项因素对训练集合单独聚类,在保持很高的准确率(98%以上)的前提下,选取使得召回率尽量高的那个阈值。经过训练,“职位”的阈值取 3;“共现人名”的阈值取 0.5;“普通词场”的阈值取 0.25。

4.5 人物模型的同一性判别与合并

同一性判别整个算法分为两步：

- (1) 利用规则 1、2 把同名的人物模型片断聚成若干紧密的团，称为“微类”；
- (2) 利用凝聚式层次聚类的方法对这些微类进一步聚类。

Step1: 将所有的微类各自单独形成一个簇；

Step2: 从现有所有的簇中选择最近(或者最相似)的两个簇，进行合并；

Step3: 如果只剩下一个簇或者达到终止条件(所有簇的相似度都低于阈值)，聚类结束，否则返回 Step2。

在人物同一性判别的同时，还要把代表相同人物的模型片断进行合并。其中，词场向量的合并比较简单，采用中心法表示，直接计算特征权重的和平均即可。属性信息的合并，涉及到信息抽取中结果模板合并的问题[Douglas 1999]。信息抽取结果模板的合并多采用基于规则的方法，即先确定模板槽类型，再对每一个类型比较两个候选者，看是否是共指、连续或包容关系。

我们采用 4.4.1 和 4.4.2 介绍的属性短语相似度计算公式来进行属性的去重，即如果两个属性的相似度为 1，则它们就是重复的，将其排除掉；否则保留下来。

合并过程结束后，我们要进行聚类结果的呈现。我们把同名人物的团并列排放在一起，这样便于比较和浏览。对于一个具体的人物，我们制定了“人物简历模板”，把识别出的各项信息填充进模板里，从而形成该人物的简历。模板的内容如图 4.4 所示。同名判别和人物简历的呈现结果如图 4.5 所示。

<姓名>，<性别>，<民族>，<出生年月>生，<籍贯>人，家住<家庭住址>。
主要亲属有{<亲属姓名>}。
现任<职务>。曾任{<职务>}。
相关人物：{<共现人名>}

图 4.4 人物简历模板结构



图 4.5 同名判别和人物简历的呈现结果示意图

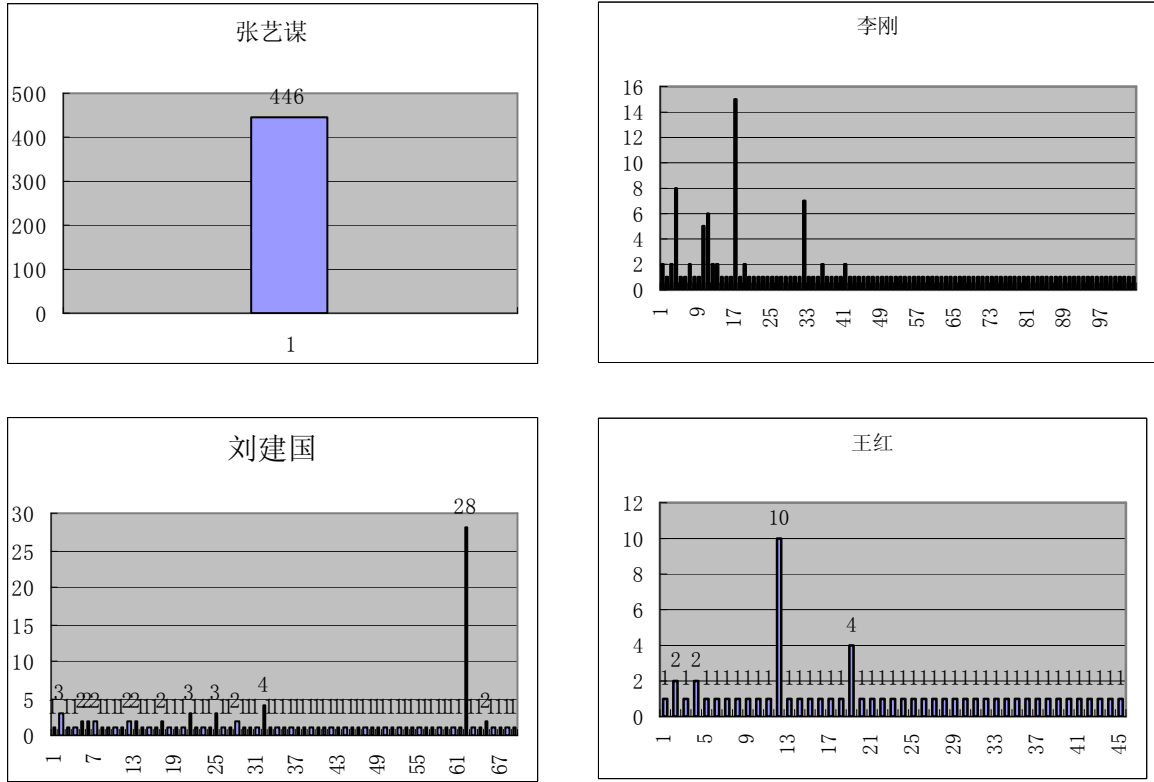


图 4.6 典型的 10 个人名的同名情况分布图

4.6.2 评测方法

我们采用基于人工标注类的评测方法。对于数据集中某个人工标注的主题 T ，假设在自动聚类形成的类层次结构中存在一个与之对应的簇 C 。为了发现 C ，遍历聚类谱系图，计算准确率、召回率和 F 值，以 F 值最大的簇作为与该主题 T 对应的簇。对于任何人工主题 T 和聚类簇 X ，计算公式如下：

$$\text{Precision}(X, T) = \frac{|X \cap T|}{|X|} \quad \text{Recall}(X, T) = \frac{|X \cap T|}{|T|}$$

$$F(X, T) = \frac{2P(X, T) * R(X, T)}{P(X, T) + R(X, T)}$$

对于每个人工主题 T

$$F(T) = \max_{X \in \text{CluResult}} \{F(X, T)\}, \text{ CluResult 表示所有聚类结果簇的集合。}$$

最终的 F 值为

$$F = \frac{\sum_{T \in \text{TopicHand}} (|T| * F(T))}{\sum_{T \in \text{TopicHand}} |T|}, \text{ TopicHand 表示所有人工标注主题}$$

4.6.3 实验结果

下面，我们通过实验来验证我们所提出的人物模型同一性判别的效果。

4.6.3.1 人物模型各维度比较结果

我们采用聚类的方法来进行人物模型的同一性判别。在普通的文本聚类中，常见的方法是将文本转化成向量空间模型，用来进行聚类。我们提出的人物模型中，普通词场的维度与此有类似之处。然而，我们需要确定人物所对应的普通词场的最佳范围。

我们着重考查四种人物对应的普通词场的范围：全篇报道（Text），人物对应语段（Para_ext），人名所在段落（Para），人名所在句子（Sentence）。其中，人物对应语段是指按照 3.5 节所介绍的方法来确定的人物在文本中的作用范围。

我们进行了两组实验。实验一是单独采用普通词场，考察四种词场范围对人物同一性判别的效果；实验二是先按照 4.4.4 节提出的规则 1 和规则 2 把语料形成若干紧密的团（这个过程标识为 IE），再用普通词场进行判别，考察四种词场范围的效果。图 4.7 是相关实验的结果。

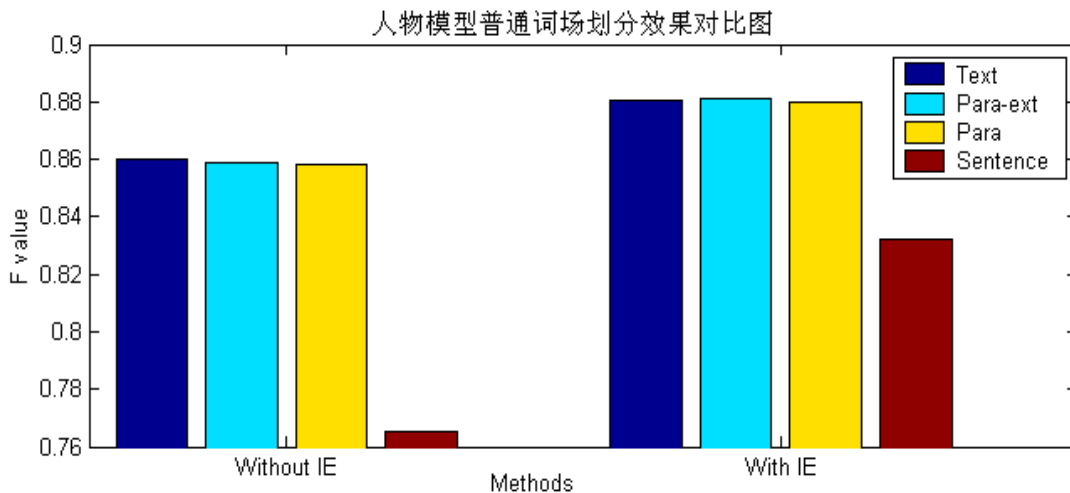


图 4.7 人物模型的普通词场划分效果对比图

通过图 4.7 的结果，我们可以发现：

- (1) 采用 IE 匹配的方法后，不管是采用何种范围的词场，效果都有显著的提高。
- (2) Para_ext 似乎是人物模型普通词场范围的最佳选择。这是因为，Para_ext 与 Text 相比，在引入 IE 匹配规则前后，效果各有胜负，然而 Para_ext 的计算代价明显比 Text 低；Para_ext 的效果始终略胜于 Para；尽管 Sentence 的计算代价最小，但它的效果比较差，可能是 Sentence 本身所含的信息量太小，无法对人物进行有效标识。
- (3) 尽管 Para_ext、Text、Para 的效果有所差别，但相差的并不明显。这说明单独调整普通词场这一维度对人物同一性判别效果的提高是有限的。

因此，在 Para_ext 基础上，我们逐渐引入人物模型其它维度的特征，比较了四个系统：

Para_ext: 单独采用普通词场向量；

Para_ext+Position: 普通词场向量和职位属性采用“斯坦福确定性理论”进行混合;

Para_ext+Position+Fellowship: 简称为 Combine, 即普通词场向量、职位属性和共现人名向量采用“斯坦福确定性理论”进行混合;

IE+Para_ext+Position+Fellowship: 简称为 All, 即先采用 IE 匹配规则形成紧密的团, 再采用 Combine 模型。

表 4.2 和图 4.8 展示了这四个系统的效果比较。

表 4.2 人物模型各维度效果对比统计表

Run	Precision	Recall	F_Measure
Para_ext	0.8841	0.9048	0.8586
Para_ext+Position	0.9066	0.9081	0.8790
Combine (Para_ext+Position +Fellowship)	0.9309	0.9042	0.8901
All (IE + Para_ext+Position +Fellowship)	0.9512	0.9243	0.9124

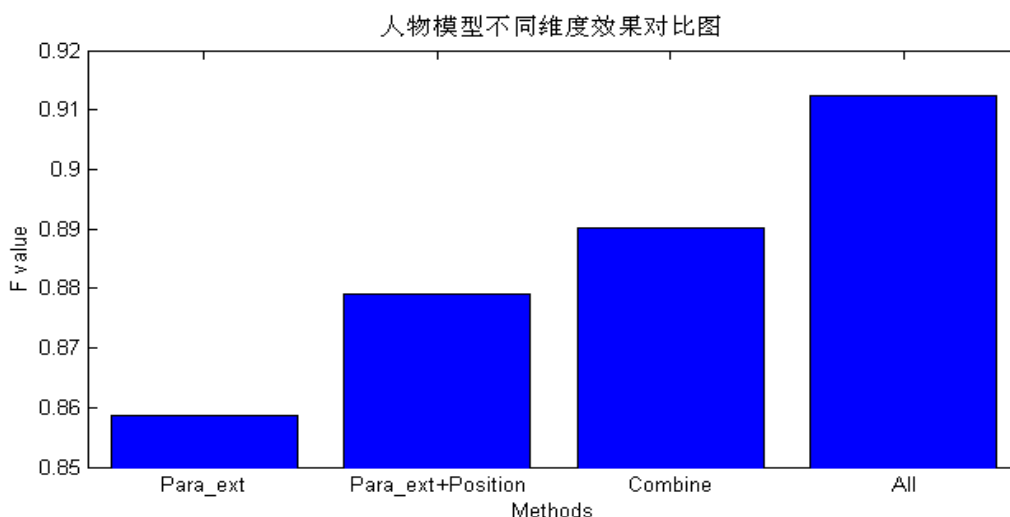


图 4.8 人物模型不同维度效果对比图

由此可见, 人物各维度的混合模型能够有效提高人物同一性判别的效果。因为描述一个人物的报道往往包括多方面的信息, 只有通过建立“人物模型”, 有效地利用人物的各种类型信息, 才能综合体现出人物的本质特征, 进而有效地对人物进行判别。

4.6.3.2 相关算法比较结果

到目前为止, 关于人物同一性判别的研究主要集中在对英文的处理, 据我们所知, 汉语方面还没有相关的研究。即使在英文处理方面, 研究者提出的方法要么不是针对互联网的语料情况[Bagga 1998][Fleischman 2004][Niu 2004][Ravin 1999][Mann 2003], 要么具体的应用场景与我们不同[Guha 2004][Bekkerman 2005]。为了与相关算法进行比较, 我们关注与我们应用场景最为接近的同名人物判别算法 WebHawk[Xiaojun 2005]。

WebHawk 是一个对互联网真实环境下的数据进行人物消歧的系统。它同样采用层次

凝聚式聚类的方法，主要利用了三方面的特征：词汇特征、命名实体特征和属性特征。词汇特征主要是文本中的普通词汇；命名实体特征是通过词法分析软件识别出的命名实体，不管它是不是对应于当前的人物；属性特征是通过信息抽取的方法识别出的人物属性信息。WebHawk 对这三类特征分别计算相似度，然后运用一种感知机方法（PAUM）将三者融合起来[Xiaojun 2005]。WebHawk 在英文网页的人物判别实验中取得了不错的效果，并通过用户的使用，验证了其性能超过了目前著名的搜索结果聚类引擎 Vivisimo (www.vivisimo.com)。

我们把 WebHawk 的同名判别方法应用在汉语语料上，来与我们提出的 Person Model 方法进行对比。为了比较两种算法的关键因素，我们把 WebHawk 的多因素融合步骤采用斯坦福确定性理论来进行，修正后的算法称之为 WebHawk_like。这样，我们在“百度”搜索引擎返回的汉语网页结果中进行了对比实验。表 4.3 是对比实验的结果。

表 4.3 汉语语料人物同一性判别中不同系统的效果对比

System	Precision	Recall	F_Measure
Baidu	0.2313	1.0000	0.2464
WebHawk_like	0.9201	0.9172	0.8904
Person Model	0.9512	0.9243	0.9124

从表 4.3 可以看出，通用的搜索引擎不区分同名人物的问题，即针对一个人名的查询，把所有的返回结果都认为属于同一个人物，这样的准确率只能达到 0.2313。WebHawk_like 和 Person Model 都能较好解决同名人物判别的问题，Person Model 效果略好。我们认为，主要原因在于 Person Model 能够更科学地反映人物的本质特征，能够把人物多方面的信息妥善地进行融合，并充分考虑到了人物特征的分布规律。图 4.9 和图 4.10 分别是对“百度”搜索引擎的搜索结果，采用 Person Model 方法处理前后，典型的 10 个人名的同一性判别效果。

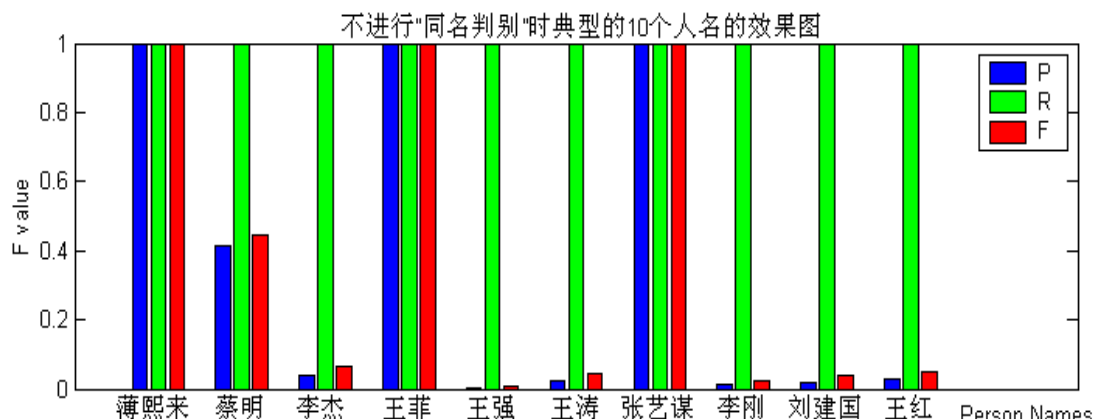


图 4.9 不进行“同名判别”时典型的 10 个人名的效果图

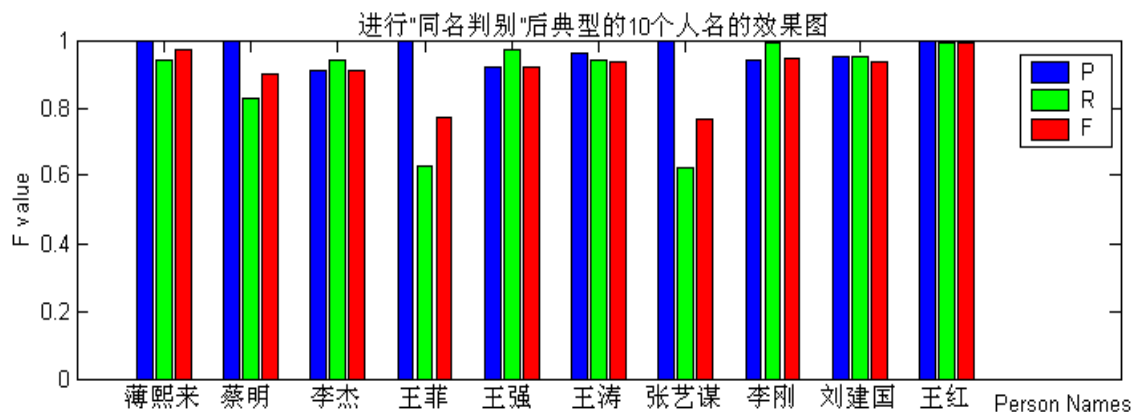


图 4.10 进行“同名判别”后典型的 10 个人名的效果图

4.6.3.3 实验结果说明

在我们的人物同一性判别实验中，即使采用最为简单的基于整个文档的向量空间模型聚类方法，也能够取得比较好的效果 ($P=0.8863$, $R=0.9041$, $F=0.8592$)，而采用了复杂的多维度混合方法后，性能只提高了 6%，我们认为这跟我们建立的语料有关，原因主要有三点：

(1) 新闻网页中出现的人物报道并不能体现人物的方方面面的生活状态，例如，“江泽民”，可能只会报道他的政治活动等，对其生活方面的描述可能很少，这就使得描述一个人的报道在内容上相似度比较高。

(2) 新闻网页并不是对社会上每个人物都报道，被报道的人物行业特征也比较明显，这就使得同名人物的报道内容相似度并不高。

(3) 我们建立的语料收集的是“百度”新闻搜索引擎的返回结果，这些结果的时间跨度比较短，一般不超过 1 年。这么短的时间内，人物的状态往往没有大的变化，对其报道的文字特征当然也就没有太大的变化。

我们相信，如果把我们提出的“人物模型”混合计算的方法应用在描述更加复杂的语料中，跟踪更长一段时期内人物的报道，将能够发挥更大的作用。

4.7 本章小结

在这一章里，我们对人物对象挖掘的同一性判断问题和数据融合问题进行了详细的论述。我们主要介绍了人物模型的特征表示方法和相似度的计算方法，在此基础上进行同一性判别和人物模型数据的融合。

人物模型包含多个维度：名称、基本属性信息、介绍性信息和词场信息。按照特征类型来划分，主要分为两类：属性特征和数值特征。“基本属性信息”和“介绍性信息”属于属性特征类型，数值特征主要包括“普通词场向量”和“共现人名向量”。本章对各种类型的信息在计算机中的表示进行了研究，并通过实验发现了各种特征在互联网真实语料中的分布规律。

接下来，本章详细探讨了人物模型各个维度的相似度计算问题，并结合各个维度的

数据稀疏程度，提出了基于混合类型数据的人物模型相似度计算方法；然后，我们分为两个阶段来进行人物模型的同一性判别。首先利用属性信息的强匹配，把语料分为若干紧密的团，然后以这些团为“枢纽”，再进行相似度的计算，得到最终的结果。

本章还探讨了人物模型不同维度数据的融合方法。通过数据融合，得到完整的人物模型信息。通过制定人物简历模板，我们把人物模型的信息很好地呈现出来，制作出人物的简历。

我们收集互联网上的真实网页，建立了语料集。在此基础上，比较了人物模型中各个维度的特征对人物同一性判别的作用，并跟目前效果比较好的系统进行了对比。实验表明，我们提出了人物模型同一性判别方法能够取得更好的效果。

第五章 基于事件的信息组织

经过数据预处理和人物的同一性判别阶段，我们能够把描述一个人物的信息有效地归结在一起。然而，描述一个人物的信息往往是杂乱无章的，我们需要对同一人物的信息进行整理，梳理出脉络，实现对该人物的追踪。

对人物的追踪，归根到底还是跟踪该人物参与的事件信息，具体来说有两种方式：一种是“编年史”的方式，即以时间为主线制作人物的“大事年表”；一种是“叙事”的方式，即以事件为主线制作人物的“传记”。从目前“事件信息组织”的技术水平来看，一种方法可以识别出具体的事件角色框架，但方法本身往往局限于特定的领域，我们称之为“微观粒度的事件组织”；另一种方法以文本或段落为粒度单位，可以对非限制领域事件信息进行大规模处理，但分析出的事件特征往往比较模糊，我们称之为“宏观粒度的事件组织”。

在这一章里，我们将对基于事件的信息组织技术进行详细的论述。因为“微观粒度事件组织”技术已经有多年的历史，技术上相对成熟，因此本章只对其进行简单的介绍。本章详细介绍“宏观粒度的事件组织”技术，目前该项研究国际上称之为“话题识别与跟踪”。我们首先介绍了事件的属性特征对话题识别的影响，然后介绍了话题的层次化组织方法。我们利用话题的层次化组织来挖掘话题之间的关系。最后，本章给出相关的实验结果及分析。

5.1 微观粒度的事件组织

自由文本中出现的事件，都有明确的当事者或者施事者，有构成事件的基本要素：时间、地点、人物等。每个事件都有一个相对完整的生命周期，有自身的起承转合。进一步细分，自然发生的事件，有它的起始、发展和完结；人为发生的事件，可能带有一定的目标，目标的设立是准备阶段，然后进入实施阶段，实施的结果可能是目标的达成、部分达成、转移、失败等等。

如果我们能够把人物事件的特征要素识别出来，填入事件列表，那么我们只要维护此列表即可。具体做法上，可以应用事件信息抽取的技术。事件信息抽取就是从文档中识别出某个事件发生的时间、地点、事件的参与者、造成的后果等信息，填入预先设定的模板槽中。一个事件信息抽取的例子是从恐怖活动新闻报道中识别出恐怖活动的时间、地点、类型、发起者、攻击的目标、对目标造成的损伤、所用的工具等。与实体属性和实体关系抽取相同，事件信息抽取也采取模式匹配法，而所需的模式需要手工或自动的方式来设定或获取。所以，事件抽取系统一般分为两个模块：模式获取模块和信息抽取模块[Douglas 1999]。

在应用事件信息抽取把人物相关的事件列表填满之后，可以对列表中的记录进行

组织。可以对记录按时间顺序进行排序（“编年史”方式），也可对各条记录按内容进行归类（“叙事”方式）。其中，前者涉及到人工智能领域的时态推理机制。时态推理可以看作是约束满足问题（CSP）的一个特例，其中变量表示时态对象，变量之间的约束对应于对象间的时态关系；时态问题还可以使用时态逻辑表达，时态推理对应着时态逻辑的归结证明过程[Pani 2001]。后者涉及到信息抽取模板之间的合一运算问题。这样，我们就可精确地识别出人物事件的特征角色，对事件进行细粒度的深入挖掘。

然而，由于事件抽取总是需要特定领域的知识库的支持，那种能够做到“无所不抽”的事件信息抽取系统离我们仍然遥远。尽管人们为了改善信息抽取的移植性已经做了大量的努力，例如：将领域相关的模式库和模式匹配功能相分离[Yangarber 2000]；采用机器学习的方法自动获取模式[Riloff 1998] [Riloff 1999]；设计各种跨领域的模式表达方式[Muslea 1999][Kim&Moldovan 1995]等，但仍然改变不了事件抽取领域相关的本质属性。另外，即使是在某一特定领域内，事件信息抽取也不是万能的。目前事件信息抽取所采用的模式是从含有事件相关信息的语句中泛化得到的。因此，事件的相关信息在句子(群)中越分散，模式的获取和匹配就越困难，也就代表着事件及其描述的复杂度越高，从中进行事件抽取难度就越大。目前，对这类复杂事件的信息抽取，效果还不理想。

因此，对于人物追踪来说，对一些特定领域、复杂程度较低的事件采用事件信息抽取的方法来进行处理，是比较适合的。具体地，对一个人物比较重要的事件类型，例如“职业变动”、“毕业升学”等，可以用事件抽取的方法，以“加入”、“当选”、“担任”、“考入”等词汇为触发特征来进行模式的获取和匹配。这样，我们就可用这些细粒度的事件信息来丰富人物的履历内容。

我们只是采取简单的方法来抽取人物的特定类型事件信息，即从 Hownet 中找出上位是“称作”、“担任”、“停做”、“当作”、“获胜”、“免除”、“是”、“使之是”、“选拔”等类型的动词。在抽取人物职务时，如果该职务被判断为属于该人物，且对应的特征向量中含有上述动词，则把该短句抽取出来，作为人物的特定事件描述。

对于互联网上好多名人，尤其是政治人物来说，会有关于该人物的活动报道专集，如图 1.7b 所示。目前，这些专集多是通过人工制作出来的。如果想要通过计算机自动地生成人物的“活动报道专集”，由于细粒度挖掘方法在通用性、高效性方面的局限，还不能满足实际要求。必须考虑以文本或段落为粒度单位，不涉及更加细致的事件结构，考虑宏观的事件特征，对事件抽取进行弱化。

5.2 宏观粒度的事件组织

宏观粒度的事件组织就是以文本或段落作为描述事件的基本单位，借用信息检索的技术把具有事件内在联系报道组织在一起。尽管这种利用宏观特征的处理方法不如细粒度事件挖掘那样精细和深入，却可以克服事件信息抽取方法的限定领域的弊端，对各种类型的事件进行大规模地高效处理。

在宏观粒度事件组织方面，国际上有“话题识别与跟踪”(Topic Detection and Tracking, 简称 TDT) 这样一个研究角度。话题识别与跟踪是一项旨在依据话题对语言文本信息流进行组织利用的研究，其目的是要克服传统的检索、过滤技术在新闻话题组织中的局限，发明适合于事件特点的技术方法[Allan 2002]。话题识别与跟踪能够把具有事件内在发展逻辑的报道以话题的形式组织在一起。例如，把所有关于“非典”话题的报道组织在一起，其中包括政治、经济、医学等方方面面的描述，它们的内容相似度也许并不高，但是有内在的逻辑联系。

话题识别与跟踪研究的目的是要克服传统的检索、过滤技术在新闻话题组织中的局限，发明专门适用于事件信息组织的技术。传统的检索、过滤技术是基于主题来组织信息的，而话题识别与跟踪所着眼的信息组织是基于事件的。既然是基于事件的信息，就一定有开端、发展和结局；而每一个阶段都会有构成具体事件的要素：时间、地点、人物；话题中包含着核心事件和与之相关的子事件。其中，时序问题是区别于基于主题的信息组织的一个显著方面，基于事件的话题总是随着时间发展而不断地变化，它所包含的事件尽管紧密相连，但却可能不再属于同一个主题了[Allan 2002]。

为了便于人们理解和认识，话题识别与跟踪对“话题”、“事件”、“主题”等相关概念做了定义。

定义 1：话题 (Topic) 是指“一个核心事件或活动以及所有与之直接相关的事件和活动”。

定义 2：事件 (Event) 定义为“由某些原因、条件引起，发生在特定时间、地点，涉及到某些对象 (人或物)，并可能伴随某些必然结果的一系列活动”。

定义 3：主题 (Subject) 是“事件所属的类别”，一个事件一定会属于某一个主题。

到目前为止，话题识别与跟踪领域的大部分研究主要体现在两个方面：一是选择调整传统的信息检索方法，使之更加适合于处理话题[Yang 1998][Papka 1999][Chen 2002][Brants 2003]；二是发明符合事件特点的方法和策略[Carthy 2002][Makkonen 2004][Swan 2000][Fukumoto 2000]。例如，考虑事件的发展阶段：开端、发展和结局；考虑构成事件的具体要素：时间、地点、人物；考虑事件的时序特性；考虑事件的发展模式：“突发性”和“持久性”；考虑事件的层次维度：时间维、地点维和语义维等。

从本质上讲，事件组织是“话题识别与跟踪”和“人物追踪”的共同基础。话题识别与跟踪在国际上研究的比较多，人物追踪这个方面则相对冷清。二者是组织事件的两个不同的角度——话题角度和人物角度。这是因为，在做话题识别与跟踪的时候，是把不同人物参与的围绕同一话题的事件组织在一起；在做人物追踪的时候，是把同一人物参与的围绕不同话题的事件组织在一起，它们之间涉及的基本技术是相通的。这样，我们就可以以人物在文本中对应的语段为基本的处理单位，借鉴话题识别与跟踪的技术，把人物事件信息在宏观粒度上组织起来。

话题识别与跟踪是一项评测驱动的研究任务。因为我们主要以非监督的方式来研究宏观粒度的事件组织问题，所以我们以“话题识别”(Topic Detection)任务为研究平台，

来开展对宏观粒度事件组织的研究。

5.2.1 宏观粒度事件¹的特征

为了对事件信息进行有效地处理，有必要了解清楚事件的特征。我们从事件自身的发展、事件之间的关系和事件的演变来概括事件的属性，主要有以下三方面特征：

(1) 事件是有生命周期的

一个事件从整体上之所以成为一个事件，是因为它有一个相对完整的生命周期，有自身的起承转合。不管是很宏观的事件还是很微观的事件，都是如此。

进一步细分，自然发生的事件，有它的起始、发展和完结。人为发生的事件，可能带有一定的目标，目标的设立是准备阶段，然后进入实施阶段，实施的结果可能是目标的达成、部分达成、转移、失败等等。目标还可以分解为子目标，子目标的达成情况对于总目标来说，可能意味着进展顺利、遭遇挫折、柳暗花明、接近成功、前功尽弃等等状态。

事件波及的地域，也可能随着事件的展开变大或变小。例如，不久前的反恐事件，一开始局限在阿富汗，后来又涉及到了伊拉克。

事件要有明确的当事者或者施事者。例如，通古斯大爆炸，是一个自然发生的事件，当事者就是通古斯这个地域；广岛原子弹爆炸，是美军发起的战争行为，是人为事件，施事者是美军，目标是造成大规模杀伤，迫使日本投降。从效果看，目标达成了。

事件的生命周期有长有短，在生命周期内的各个阶段往往会呈现不同的分布规律。按照出现规律，事件可分为两种：“突发性事件”和“持久性事件”。“突发性事件”的特点是持续的时间较短，但在出现时期内报道的数量很多；“持久性事件”的特点是持续时间长，但报道的数量在各个时间段内可能分布不均匀，在有的时间段内可能呈现突发性的趋势[Allan 2002]。图 5.1 是事件发展规律示意图，图中的每个点都代表一定数量的报道。

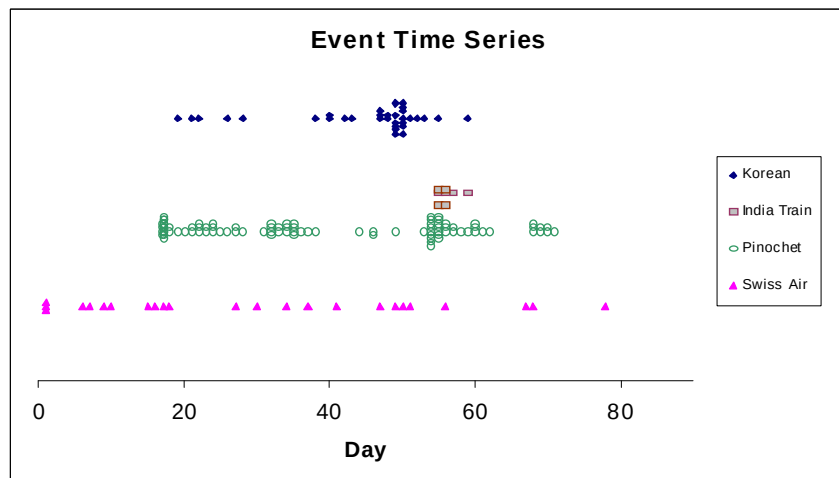


图 5.1 事件发展规律示意图

¹ 为了叙述的方便，本文在有些地方不区分“事件”和“话题”的概念

(2) 事件是有层次的

事件可由子事件构成。子事件构成的方式分为同构模式和非同构模式两种。例如，SARS 流行，是一种同构模式，也就是说，小的子事件从参数（角色）结构上看是相同类型的，其中的角色和母事件的角色有某种关联（比如中国的 SARS vs. 北京的 SARS；5 月份的 SARS vs. 六月份的 SARS 等）。再比如温家宝总理出访，可以分解成访问美国、加拿大、墨西哥、埃塞俄比亚四个子事件。子事件构成事件，除了同构的模式，还有非同构的模式（例如 SARS 流行中医务工作者参与的事件，普通大众参与的事件和患者参与的事件等）。

某个具体事件可以用三个维度来较为准确地刻画：时间维，空间维，语义维。从任一维度来看，事件及其子事件之间都能产生包含关系：

① 从时间维来看，发生时间范围广的事件包含发生在其间的子事件，例如，发生在 1966 年的文革事件包含在发生在 1966-1976 中的文革事件中。

② 从空间维来看，发生空间范围广的事件包含发生在其间的子事件，例如，发生在北京的文革事件包含在发生在中国的文革事件中。

③ 从语义维来看，语义层次高的事件包含语义层次底的事件，例如，“毛泽东 1966 年接见红卫兵”这个子事件发生整个文革的大事件中。

实际上，具体事件及其子事件之间的包含关系往往是同时从这三个维度出发来考察的。我们说一个事件包含一个或多个子事件，是指该事件比之于它的子事件来讲发生的时间范围宽、空间范围广、语义层次高。

(3) 事件是可变的

事件未必如预想的那样达成或完结，而是往往出现“变局”。

预想的达成或完结，可称之为“流变”。此外还可能出现两种情况：

一种是“裂变”，即一个事件分解成多个独立的事件分别展开。

一种是“蠕变”，即一个事件逐步“变味儿”，原有的目标或参与者逐步变更为新的目标和参与者，经过过渡变成了一个与原来事件有相当大区别的另一个事件。

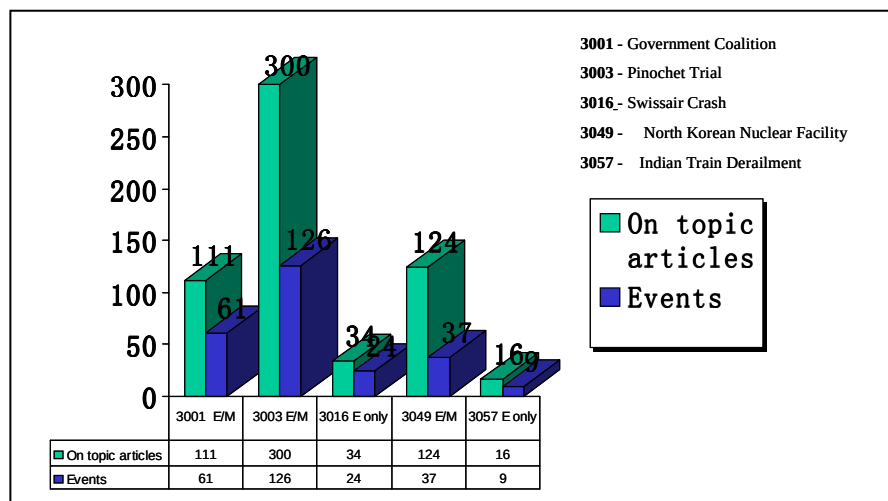


图 5.2 不同事件的报道数和子事件数示意图

当事件发生时，往往会出现多方面的报道。事件下面还包含着子事件。由于事件的知名程度不同和包含的子事件数目不同，对它的报道数量会有所不同；对于同一个事件的不同发展阶段，各大新闻媒体的报道数量也不尽相同。图 5.2 显示了话题识别与跟踪评测中，5 个典型事件所包含的子事件数和对应的报道数。图 5.3 表明了话题识别与跟踪评测中，某个典型事件的不同时期内各大新闻媒体的报道数目。

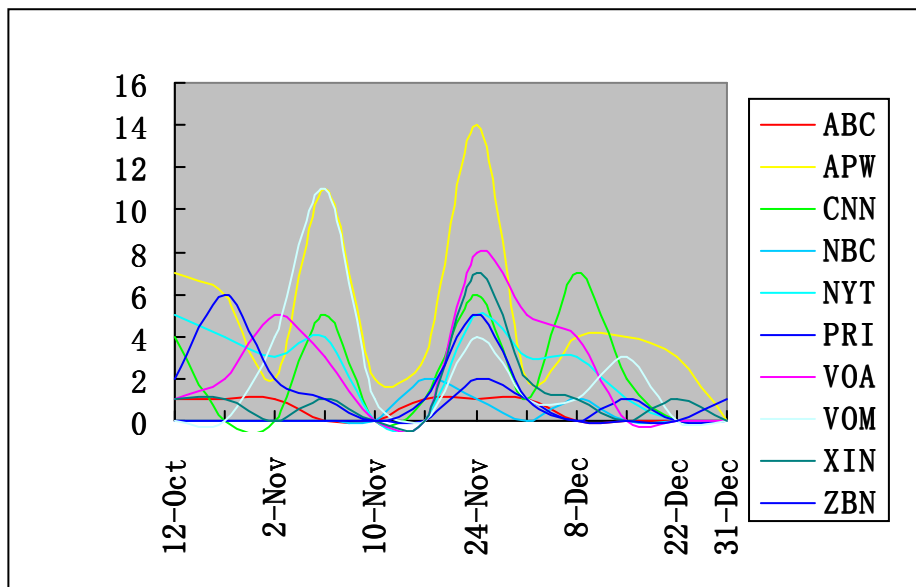


图 5.3 事件的不同时期内各大新闻媒体的报道数目示意图

5.2.2 针对事件特点的话题识别方法

本节我们主要研究如何更好地进行话题的识别。话题识别所处理的对象是热点话题的新闻报道，所用方法是否符合事件的特点，对系统性能的提高极为重要。对话题很好地识别，需要有效的事件组织技术，这是话题识别的基础。在这里，我们主要根据事件的特点来挖掘一些方法，着重关注于：话题的时序属性；事件角色特征词对话题识别的影响；自然语言处理技术在话题识别中的应用。

在自由文本处理领域，把两个属于同一类型的不同事件，例如“两次不同的火车事故”区分开是很难的[Allan 2002]。描述这两个事件的词汇如此相似，以致通常的文本计算策略很难把它们区分开；同一事件内部的报道相似度往往并不高，因为事件有不同的发展阶段，且往往会出现“变局”，例如“911 事件”，如何把事件的起承转合关系有效地识别并组织起来，是文本处理领域的一个严峻的挑战。由于事件信息处理有这样的特殊性和难点，所以话题识别不能完全遵循通常的文本聚类方法，必须针对事件的特点进行策略调整和方法创新。下面介绍我们所提出的核心方法和策略。

5.2.2.1 基于分组策略的识别方法

在话题识别与跟踪中，时序是一个重要的特征。随着事件的逐步展开，对事件的报道也是按照时间先后顺序逐步出现的。一般来说，讨论同一事件的报道在时间上往往是

相近的，时间间隔越远越倾向于描述不同的事件。按照出现的模式，事件可分为两种：

“突发性事件”和“持久性事件”。“突发性事件”的特点是持续的时间较短，但在出现的时期内报道的数量却很多；“持久性事件”的特点是持续时间长，但报道的数量在各个时间段内可能分布不均匀，在有的时间段内可能呈现突发性的趋势。

在传统的话题识别任务定义中，报道要求按照时序进行处理，并且允许有一定的时间延迟[Allan 2002]。我们的方法与[Yang 1999]类似，即将报道按照时序分组，组内优先聚类。然而，我们详细考察了时序信息在话题识别与跟踪系统中的作用。我们对文档按照固定间隔进行分组，组内优先聚类，考察了按照时序分组和不按时序分组，以及不同分组间隔的大小对系统的影响。在对比实验中，我们发现，对文档按照报道的时间顺序进行分组，组内优先聚类，再把聚类结果合并的策略能够极大地提高系统的效果。方法步骤如下所示：

- ① 提取报道的时间信息，把报道按照时间顺序排序；
- ② 按照一定的时间间隔将报道分成若干组；
- ③ 在每一组内，采用自底向上的层次凝聚式聚类方法进行聚类，利用阈值控制生成的“微类”数量；
- ④ 去掉各组之间的间隔，但仍然保持各“微类”结果的时间顺序；采用单遍聚类（Single-Pass）算法将各“微类”结果进行合并。

5.2.2.2 基于概念空间的命名实体权重计算方法

在这一节，我们考察事件特征对话题识别的影响。在事件的信息组织中，要有效识别出属于同一类型却不同事件的报道或属于同一事件却不同类型的报道，必须寻找出真正具有代表性的特征。[Allan 2002]猜测仅仅一小部分特殊的词汇才真正能对事件的区分产生作用。然而，问题的关键是，到底是什么样的特征词？我们把文本中的词汇划分为两种类型：命名实体 NE 和其它普通词 N。命名实体主要包括人名、地名、机构名等；普通词主要包括名词、形容词和一些特殊的符号。

从直觉上来看，当报道两个不同的火车事故时，事故所发生的时间、地点以及涉及到的人物会有比较大的区分能力。[Yang 1999]等提出了对命名实体加权的方法，[Papka 1999]发现当提高命名实体的权重时，话题识别的正确率会有所提高。然而，我们认为，对命名实体只进行加权是不够的，解决相关词之间的不匹配现象也极为重要。

在文本中，代表同一命名实体概念 C_i 的词通常不只一个，而往往是一个词的集合 $C_i = \{t_{i1}, t_{i2}, \dots, t_{ik}\}$ 。例如，代表“中科院计算所”这个概念的命名实体词可能有“计算所”、“中科院计算所”、“计算技术研究所”、“中国科学院计算技术研究所”、“ICT”等，这种现象在地区及其简称方面尤为常见。由于命名实体识别技术的限制，代表同一命名实体概念的词 $t_{ij}(1 \leq j \leq k)$ ，可能被标注为命名实体词 $NE_{t_{ij}}$ ，也可能被标注为普通词 $N_{t_{ij}}$ 。如果忽视这些现象，就会造成概念空间的不匹配，而命名实体的加权策略又会使

概念失配的影响加剧。为此，我们提出基于概念空间的命名实体权重计算方法，主要是把代表同一概念的词汇统一起来计算权重，具体如下：

$$IDF_{Ci} = \frac{N}{\sum_{j=1}^k n_{t_{ij}}} \quad n_{t_{ij}} \text{ 为出现 } t_{ij} \text{ 的文档数}$$

$$TF_{Ci} = \sum_{j=1}^k (TF_{NE_{t_{ij}}} + TF_{Nt_{ij}})$$

$$W_{Ci} = \alpha \log(TF_{Ci} + 1) \cdot \log(IDF_{Ci} + 1) \quad (\alpha > 1), \quad w \text{ 表示命名实体概念的权重。}$$

我们利用半自动的方式，对代表相同概念的命名实体建立了同义库，库中主要收录了国家、地区、城市、团体机构等的别称和简称；我们还建立了命名实体停用后缀表，后缀诸如：“省”、“市”、“县”、“镇”、“村”、“地区”、“特别行政区”等，出现在表中的命名实体后缀被过滤掉。我们利用这些资源把命名实体在概念空间上统一起来后，再对其进行加权。在 TDT4 中文语料 27,142 篇文档中，命名实体词出现了 477,277 次，利用概念空间统一起来的有 45,691 个，占命名实体词的 10%；被还原为命名实体的普通词有 33,410 个，占命名实体词的 7%。

5.2.2.3 指代消解在话题识别中的应用

到目前为止，话题识别与跟踪领域的大部分研究主要还是借鉴信息检索的方法，通过调整某些参数来使这些方法更适合于处理话题。但是，话题识别与跟踪研究的特殊性，如面向话题，基于时间等，也决定了利用信息检索方法对进一步提升系统的性能是有限的，要想突破必须借助于更多的自然语言理解技术。

我们尝试在话题识别与跟踪中引入指代消解技术。指代消解，就是确定指代词与先行词之间的关系，从而明确指代词指代什么对象。在新闻报道的叙述中，主要的事件角色特征词往往只出现一两次，更多的是利用指代词对该特征进行引用。考虑一篇新闻报道的事件角色特征集合 $T = \{T_{C1}, T_{C2}, \dots, T_{Cn}\}$ ，其中， $T_{Ci} = \{P_i, n_{i1}, n_{i2}, \dots, n_{ik}\} (1 \leq i \leq n)$ ， P_i 为角色 T_{Ci} 的先行词， $n_{ij} (1 \leq j \leq k)$ 为 P_i 的指代词。在以往的系统中，指代词要么与先行词完全独立进行处理，要么作为停用词性被过滤掉，即 $TF_{T_{Ci}} = TF_{P_i}$ (TF 为词频， $1 \leq i \leq n$)，这就大大削弱了相关角色在系统中的地位，造成事件角色识别的不均衡。为此，我们采用基于指代消解的事件角色词频计算方法，主要是把指示代词引入权重计算中：

$$TF_{T_{Ci}} = TF_{P_i} + \beta \sum_{j=1}^k Dis(P_i, n_{ij}) \quad (0 < \beta < 1)$$

$$Dis(P_i, n_{ij}) = \begin{cases} Count_{sentence}(P_i, n_{ij}) & \text{if } n_{ij} \text{ 是代词 and } \beta \cdot Count_{sentence}(P_i, n_{ij}) < 1 \\ 1/\beta & \text{else} \end{cases}$$

$Count_{sentence}(P_i, n_{ij})$ 为先行词 P_i 与指代词 n_{ij} ($1 \leq i \leq n, 1 \leq j \leq k$) 之间所夹的句子个数, β 为预设参数, 取为 0.3。在这种方法中, 指示代词与先行词的距离越大, 则表明它越能独立的代表一定的意思, 因而权重越高。

我们主要针对人物姓氏和第三人称代词进行了指代消解的处理, 所用的指代消解系统采用基于规则的方法, 它综合应用了性别、单复数和距离等属性信息。在 TDT4 汉语语料的 27,142 篇报道中, 人物全名出现了 142,601 次, 姓氏或名字单独出现的情况有 32,282 次, 第三人称代词出现了 56,653 次, 我们对这些词采用了基于指代消解的词频计算方法。

5.2.3 基于多层聚类的话题层次化组织方法

上一节, 我们研究了话题的识别方法, 着眼点是把属于不同话题的报道区分开。然而, 不同的话题往往不是孤立的, 它们之间存在着一定程度上的关系, 话题的粒度也是一个很难判断的概念。

为了解决这个问题, 一个思路是把话题按照层次化的关系组织起来。关于这项研究, 在话题识别与跟踪国际评测里有一项标准的评测任务——层次化话题识别 (Hierarchical Topic Detection, 简称 HTD) [NIST 2004]。层次化话题识别在 TDT 评测中是一项全新的任务, 它是传统的话题识别任务的扩展。本节主要介绍我们对层次化话题识别问题的研究, 首先介绍该项任务的优点及主要解决方法, 然后介绍我们的思路和方法。

5.2.3.1 层次化话题识别的优点

层次化话题识别是对原来的话题识别任务的扩展。传统的话题识别任务把所有的话题都折中在同一粒度层次上, 这样并不符合话题的特点。事实上, 话题是经常处在不同粒度层次上的, 例如, “亚洲金融危机”包含着“印尼股市暴动”、“日本银行破产”等等。因此, 与以往的话题识别任务相比, 层次化话题识别力图克服两个不符合实际情况的假设: 假设所有的话题都处在同一粒度层次上; 假设每一个文档至多属于一个话题 [NIST 2004]。

与传统的话题识别任务不同, 层次化话题识别系统把文档组织成有向无环图 (DAG) 而不是扁平式结构, 通过这种方式来更好地反映话题的层次包容关系。从应用角度来看, 层次化话题识别突破了传统的“一类一团”的结果呈现方式, 用户可以通过图形化的层次结构进行浏览, 有利于减少阅读的工作量。同样地, 层次化话题识别本质上是一个非监督式机器学习的过程, 文档聚类是其中很自然的应用。

在层次化话题识别中, 应用针对事件的信息处理技术把同一粒度层次的话题识别出

来后，还要把处于不同粒度层次的话题结点合理组织，形成科学的层次结构。下面介绍一下代表话题之间关系的有向无环图的拓扑结构。

5.2.3.2 有向无环图的拓扑结构

层次化话题识别把话题组织成有向无环图的结构，该图的具体含义如下：

① 每一个图结点都用一个唯一的字符串标识；② 每一个图结点都包含若干报道或其它图结点；③ 一个结点所包含的所有报道都自动包含在它所有的祖先结点中；④ 报道可存在于任何结点中，但不可冗余地直接存在于它的祖先结点中；⑤ 结点之间的边不可形成环；⑥ 有且仅有一个根结点，作为其它所有结点的总领；⑦ 内部结点对应一个置信度数值，指示其对应话题的聚合程度。有向无环图的示意结构如图 5.4 所示。

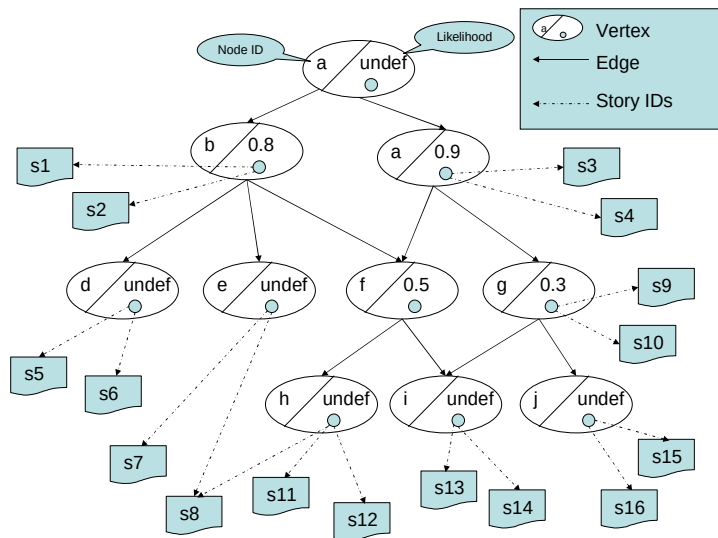


图 5.4 话题层次结构示意图

5.2.3.3 层次化话题识别的主要方法

层次化话题识别（HTD）任务在 2004 年 10 月举办的第七届话题识别与跟踪国际评测中首次出现。作为首次任务的参赛者，我们（ICT）与其它三个参赛队伍一起，开展了对这个问题的研究。另外三个参赛队伍分别是：TNO、UMASS、香港中文大学（CHUK）。每个参赛队伍所采用的方法都各有特色。

CHUK 采用分裂式 K-means 算法来产生话题的有向无环图。其中，K-means 算法用来把报道组织成一定粒度层次的话题结点，然后，采用分裂式聚类算法把粒度层次高的话题结点分割成粒度低的话题结点。

TNO 将层次化话题识别分为四个步骤来进行[Trieschnigg 2004]。他们首先采样 20,000 个报道来建立相似度矩阵，然后用自底向上的凝聚式聚类算法生成一棵二叉树。接下来，他们对二叉树进行适度的剪枝，然后，把剩余的报道分别插入到 10 个最相近的结点中。

UMASS 在聚类算法中引入一定的约束条件来减少时间复杂度 [Connell 2004]，即采

用带有约束条件的 1-NN 算法来形成话题结点，用凝聚式聚类算法建立层次结构。使用的约束条件主要是采用有限的窗口，同一新闻来源的报道优先聚类，然后是同一语言的报道，最后是所有语言的报道。

与其它参赛者相比，我们采用的方法也是自底向上的聚类方式。然而，我们是先形成每一粒度层次的话题结点，然后再把各个层次的结点组织起来。所以，我们提出的算法称之为基于多层聚类的层次化话题生成算法——MLCS (Multi-layered Clustering)。下面对该算法进行详细的介绍。

5.2.3.4 MLCS 算法的基本思想

从有向无环图的结构特点可以看出，根结点是所有话题的总领，它下面包含着若干子结点；每个结点即可以包含文档，也可以指向一个或多个更下层的结点；每个结点代表某一粒度的话题，结点的层次越往下，对应话题的粒度越小。

自底向上的文本聚类似乎是形成有向无环图的天然方法。一般地，可以利用不同大小的相似度阈值生成不同粒度层次的话题结点，相似度阈值越大，生成的话题结点粒度就越小。该阈值代表着对应话题结点的聚合程度。如果我们把通常的单层文本聚类算法组织成多层聚类，记录各层次间结点的合并过程，就会得到如图 5.4 所示的话题层次结构。特别地，如果一个结点与两个或多个上一层结点的相似度很接近，且高于其它的结点，则把这个结点同时合并到这几个上层结点中去。图 5.5 是 MLCS 算法基本思想的示意图。

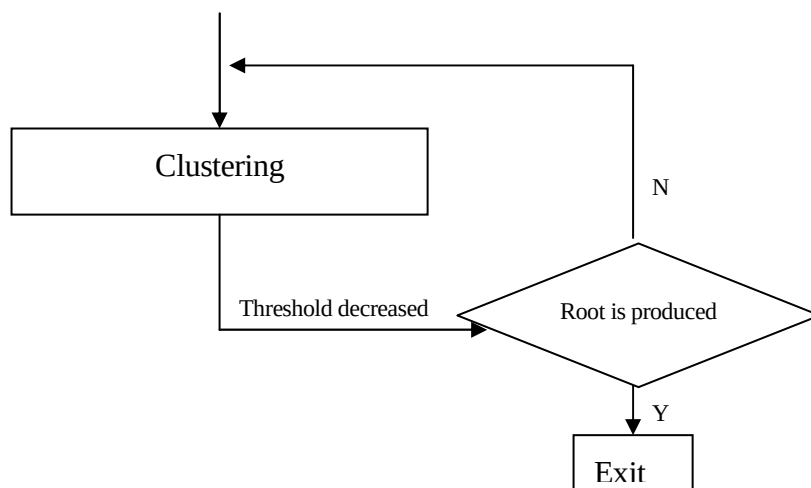


图 5.5 MLCS 算法基本思想示意图

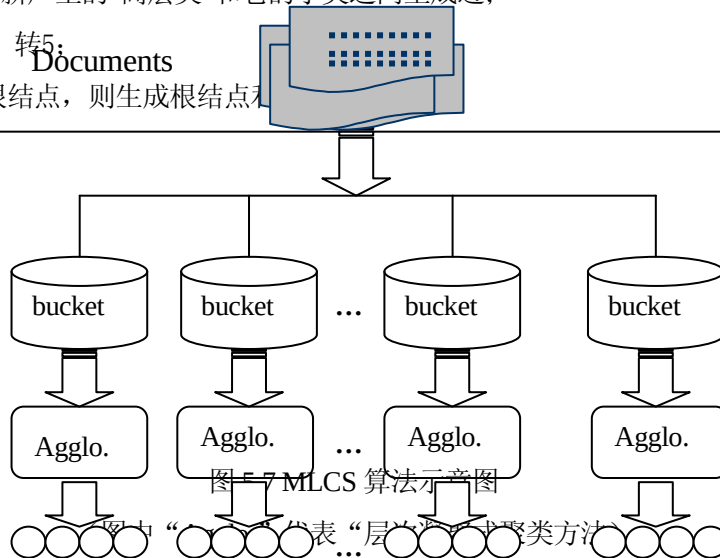
5.2.3.5 MLCS 算法的具体流程

从多层聚类算法对有向无环图结构的影响来看，聚类的层数决定了有向无环图的深度，每一层聚出的类数决定了有向无环图的宽度。算法 MLCS 采用自底向上的聚类方法，首先从一个较大的阈值开始从最底层对所有文档进行聚类，当该层聚类结束后，阈值减去一定的步长，对聚出的结果再一次进行聚类，以此循环，直至满足终止条件为止。在实际做法中，MLCS 算法利用阈值的个数控制聚类的层数，利用阈值的大小控制每层的

类数。

特别地, MLCS 算法采用了 5.2.2.1 提出的时序分组策略, 在最底层对文档按照报道时间间隔进行分组, 组内优先聚类。从话题层次化结构的角度来看待这种策略, 可以理解为: 事件是有层次的, 事件可由子事件构成; 事件可以用三个维度来刻画: 时间维、空间维、语义维, 各个维度综合反映了不同大小的事件粒度。利用这种方法, 把报道时间相近的文档优先聚合到一起, 突出了特征最为明显的时间维的特性。MLCS 的算法描述和示意图如图 5.6 和图 5.7 所示。

1. 初始化参数 θ 、 q 、 k 、 t ; 令 $i=1$;
2. 把报道按照时间顺序排序, 再按固定的时间间隔 t 将报道分成若干组 $\{P_1, P_2, \dots, P_n\}$;
3. 在 $P_j (1 \leq j \leq n)$ 内, 采用自底向上的层次凝聚式聚类方法进行聚类, 利用相似度阈值 θ 控制生成的“微类”数量; 在“微类”和对应的文档之间生成边;
4. 去掉各组的间隔, 但保持各“微类”结果的时间顺序;
5. 如果 $i > k$, 或者 $\theta < 0$, 转 8;
6. 采用单遍简单聚类 (Single-Pass) 算法将上一层“微类”结果进行合并, 利用相似度阈值 θ 控制结果的数量; 在新产生的“高层类”和它的子类之间生成边;
7. $\theta = \theta - q$, $i++$, 转 5;
8. 如果还没产生根结点, 则生成根结点和



在算法 MLCS 中, 影响算法的因素主要有相似度阈值 θ ($0 < \theta < 1$), 相似度递减步长 q ($0 < q \leq \theta$), 简单聚类层上循环次数 k ($k > 0$) 和分组时间间隔 t ($t > 0$)。通过调整具体的参数值, 可以生成不同结构的有向无环图。

5.2.3.6 MLCS 算法的增量模式

MLCS 算法本身的时间复杂度是输入文档数的立方, 在语料规模大时效率较低。为了提高算法的运行效率, 我们为 MLCS 算法引入增量模式。首先, 我们选取一定规模的文档, 并利用 MLCS 算法建立初始的有向图; 然后, 我们把剩余的文档一批一批地处理, 先将每一批文档形成若干微类, 对于每一个微类, 遍历有向图, 把该微类插入到跟它最相似的图结点中去。

我们把 MLCS 算法应用在 TDT2004 的 HTD 任务的跨语言场景中。在评测中，我们选取原始语言是英文的文本为初始文档，把汉语和阿拉伯语翻译成英语的文本作为增量处理的文档，运用 MLCS 增量算法进行处理。

5.2.4 实验结果与分析

我们利用基于以上技术的系统参加了 TDT2004 的国际评测。该系统以 TDT4 语料作为实验数据，然后在语言数据联盟(LDC)[LDC]新开发的 TDT5 语料上做了评测。下面，首先介绍我们所采用的数据集和评测指标，然后给出详细的实验结果、评测结果及相应分析。

5.2.4.1 数据集

TDT4 语料包含时间跨度为四个月(2000.10.1-2001.1.31)的新闻专线和广播语料。它一共包含 98,245 篇报道(英文 28,390 篇，汉语 27,142 篇，阿拉伯语 42,713 篇)。汉语和阿拉伯语语料提供本国语和翻译语料两种形式，翻译语料采用机器翻译软件翻译成英语。广播语料是通过语音识别软件自动转录而成的。整个语料一共 25,960 个文件，共计约 2.5GB。TDT4 语料主要来源于大型的新闻媒体机构，例如“纽约时代新闻”和“新加坡联合早报”等。语料里标注的与话题相关的报道一共 6,000 多篇，蕴涵在 100 个话题中。在实验中，我们利用了 TDT4 的汉语语料。

TDT5 语料同样包含着英语、汉语、阿拉伯语三种语言的语料，时间跨度为 6 个月(2003.4.1-2003.9.30)。与以往不同的是，TDT5 不包括语音广播类型的语料，所有语料都来自于新闻专线。并且，TDT5 语料的规模比以往任何 TDT 语料都大得多，其中包括 278,109 篇英语报道，56,486 篇汉语报道和 72,910 篇阿拉伯语报道，一共 407,505 篇。语料中一共标注了 250 个话题。

5.2.4.2 评测指标

层次化话题识别可以有多种评测的手段，其中一些在[NIST 2004]里有所介绍，目前主要采取基于最小代价的评测方法。该方法主要考虑两个因素：识别代价和遍历代价，评测公式如下：

$$\text{MinimumCost}(\text{topic}) = \text{WDET} * (C_{\text{Det}}(\text{topic}, \text{bestVtx}))_{\text{Norm}} + (1 - \text{WDET}) * (C_{\text{Travel}}(\text{topic}, \text{bestVtx}))_{\text{Norm}}$$

其中， bestVtx 是评测系统找到的使得对应话题代价最小的那个结点， WDET 是权重调节参数。

识别代价 $(C_{\text{Det}}(\text{topic}, \text{bestVtx}))_{\text{Norm}}$ 由误报率和漏报率加权求和并归一化而成，其计算公式如下：

$$(C_{\text{Det}})_{\text{Norm}} = \frac{(C_{\text{Miss}} \cdot P_{\text{Miss}} \cdot P_{\text{target}} + C_{\text{FA}} \cdot P_{\text{FA}} \cdot P_{\text{non-target}})}{\min\{C_{\text{Miss}} \cdot P_{\text{target}}, C_{\text{FA}} \cdot P_{\text{non-target}}\}}$$

其中， C_{Miss} 和 C_{FA} 分别是漏报和误报的代价； P_{Miss} 、 P_{FA} 和 P_{target} 是预设参数，用来调节漏报率和误报率在识别代价中所占的比重； $P_{\text{non-target}} = 1 - P_{\text{target}}$ 。

遍历代价($C_{\text{Travel}}(\text{topic}, \text{bestVtx})$)是从根结点遍历到 bestVtx 结点所花费的阅读代价, 其计算是一个递归累加的过程:

$$\begin{aligned} C_{\text{Travel}}(\text{topic}, \text{vertex}) &= C_{\text{Travel}}(\text{topic}, \text{parentOf}(\text{vertex})) \\ &+ \text{CBRANCH} \cdot \text{NumChildren}(\text{parentOf}(\text{vertex})) \\ &+ \text{CTITLE} \quad \text{其中, } C_{\text{Travel}}(\text{topic}, \text{root}) = 0 \end{aligned}$$

在该公式中, 根结点的遍历代价为 0; $C_{\text{Travel}}(\text{topic}, \text{parentOf}(\text{vertex}))$ 是当前结点的父结点的遍历代价; $\text{NumChildren}(\text{parentOf}(\text{vertex}))$ 是当前结点的兄弟结点的个数; CBRANCH 、 CTITLE 是预设参数, 分别表示从兄弟结点中找出最佳结点所花费的阅读代价和阅读当前结点的摘要所花费的代价。最后, 利用以下公式对计算得到的 ($C_{\text{Travel}}(\text{topic}, \text{bestVtx})$) 值进行归一化:

$$\frac{(C_{\text{Travel}}(\text{topic}, \text{bestVtx}))_{\text{Norm}}}{(C_{\text{Travel}}(\text{topic}, \text{bestVtx}))} = \frac{1}{(\text{CBRANCH} \cdot \text{MAXVTS} \cdot \text{NSTORIES} / \text{AVESPT}) + \text{CTITLE}}$$

归一化因子的含义是每个话题的平均阅读量。 MAXVTS 是每篇报道至多归属的话题数; NSTORIES 是语料中全部的报道数; AVESPT 是平均每个话题包含的报道数, 各个语料有所不同 (在 TDT4 汉语语料的实验中, AVESPT 的实际值为 16; 在 TDT2004 评测中, AVESPT 取值 88)。在以上公式中, 预设参数取值如下:

$P_{\text{target}} = 0.02$	$C_{\text{Miss}} = 1.0$	$C_{\text{FA}} = 0.1$	$\text{WDET} = 0.66$
$\text{CBRANCH} = 2.0$	$\text{CTITLE} = 1.0$	$\text{MAXVTS} = 3$	

5.2.4.3 话题识别中不同分组策略的实验结果

MLCS 算法中采用了基于时序的分组策略, 为了验证时序分组对话题识别代价的影响, 我们进行了相应的对比实验。实验中, 我们令 MLCS 算法中的参数 $k=1$, $q=0.00$, 令评测公式中的 $\text{WDET}=1$, 即不采用层次化的组织结构, 也不考虑遍历代价的评测影响因素。我们考察了不分组、按时序分组与不按时序分组对识别代价的影响, 结果如图 5.8 所示。

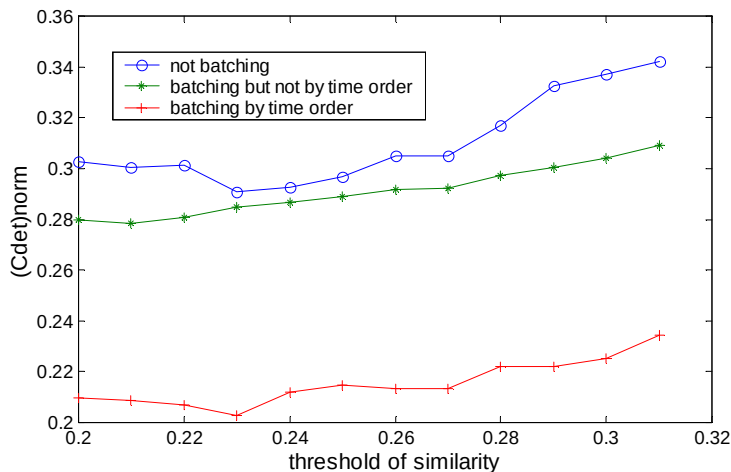


图 5.8 不同分组策略效果对比图

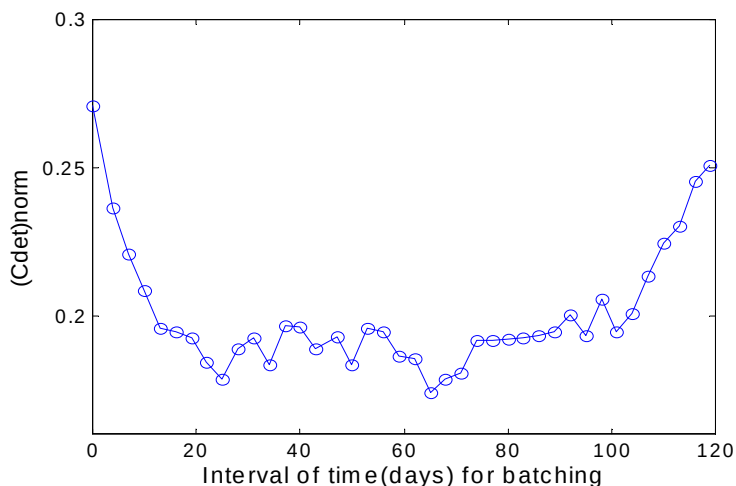


图 5.9 不同分组间隔效果分布图

图 5.8 中，曲线 1 表示不对文档进行分组，一开始就对全部语料进行单遍聚类的结果，横坐标是不同的聚类相似度阈值 θ 。曲线 2 是采用时序分组的结果，分组时间间隔为 7 天。在曲线 3 的实验中，我们仍旧把文档进行分组，组内优先聚类，与曲线 2 不同的是，我们把文档按时间均匀打散，也就是每组内的文档在时间上并不是相邻的，然后按固定的报道数进行分组。结果表明，基于时序分组的策略使识别代价降低了近 30%；当组内文档时间不相邻时，分组策略对效果提高不明显。我们又考察了在时序分组的条件下，不同的分组时间间隔对识别代价的影响，结果分布如图 5.9 所示。

通过实验可以看出，时序分组策略能有效降低系统的识别代价，主要原因是由于它把时间相近的文档优先聚合在一起，提高了属于同一话题的文档的合并机会。这说明新闻话题具有比较明显的时序聚合性。

5.2.4.4 针对事件的话题识别方法的实验结果

为了检验我们提出的话题识别方法对层次化话题识别系统性能的影响，我们构建了原型系统 BASE 和三个改进的系统：系统 1、系统 2、系统 3。

BASE 系统采用了向量空间模型，在语料预处理阶段，对于中文语料，系统进行分词、词性标注和命名实体识别；对英文语料，系统进行词性标注和词形还原。特征选择包括两层策略：一是通过常用的禁用词表来过滤禁用词，除此之外，还建立了禁用词性表，去掉动词，副词，助词等，只保留了名词、形容词、特殊符号和命名实体；二是使用文档频率信息，即把 $DF(t_i) \leq 3$ 或 $DF(t_i) \geq N/3$ 的特征词都过滤掉，因为符合这两个条件的特征分布太窄或太广，区分能力不强。权重计算采用常用的 $l_{tc} TF_IDF$ 计算公式：

$$Wt(ti) = \log(TF(ti, d) + 1) \cdot \log((N/DF(ti, d)) + 1)$$

其中词频 $TF(t_i, d)$ 为特征 t_i 在文档 d 中的出现频度, 文档频率 $DF(t_i, d)$ 为 t_i 在其中出现的文档数, N 为语料总文档数。相似度计算采用 *cosine* 公式。BASE 系统同样利用多层聚类使文档形成有向无环图结构, 然而, 与 MLCS 算法不同的是, 它没有对文档按照时序分组的策略, 而是直接利用单遍简单聚类算法进行多层聚类。

系统 1 (MLCS) 在 BASE 系统的基础上, 采用了我们提出的 MLCS 算法, 即首先对文档按照时序进行分组, 组内优先聚类, 再用单遍聚类算法进行多层聚类。实验中采用的分组时间间隔 t 为 7 天。

系统 2 (MLCS+NE) 在系统 1 的基础上, 又采用了基于概念空间的命名实体权重计算方法。

系统 3 (MLCS+NE+PR) 在应用系统 2 所有技术的同时, 又引进了基于指代消解的特征权重计算策略。

表 5.1 是这四个系统在 TDT4 汉语语料上得到的最好结果。

表 5.1 针对事件特点的话题识别方法效果对比表

系统	(Cost(detect)) _{Norm}	(Cost(travel)) _{Norm}	MinimumCost
BASE	0.2718	0.2653	0.2696
MLCS	0.1968 (↑ 27.59%)	0.2139(↑ 19.37%)	0.2026(↑ 24.85%)
MLCS+NE	0.1543(↑ 43.23%)	0.2267(↑ 14.55%)	0.1789(↑ 33.64%)
MLCS+NE+PR	0.1487(↑ 45.29%)	0.2105(↑ 20.66%)	0.1697(↑ 37.05%)

[注] 表 5.1 中“↑”标记指示当前结果与 BASE 系统相比上升的幅度

从表 5.1 可以看出, 应用针对事件的处理策略后, 系统的效果有了很大的提高。采用了 MLCS 算法的系统 1, 对系统效果提升得最为明显, 它不仅有效地减少了识别代价, 还大大降低了遍历代价; 既采用了 MLCS 算法又采用了基于概念空间的命名实体权重计算策略的系统 2, 进一步降低了识别代价和总的代价; 在采用系统 1、系统 2 所有技术的基础上, 又引进了指代消解技术的系统 3, 取得了最好的效果。从识别代价和遍历代价的提升幅度可以看出, 采取了有效的事件组织策略后, 最主要的是能够把真正属于同一话题的报道组织在一起, 从而大大地降低了系统的识别代价, 对系统总的代价也能够产生好的影响。

5.2.4.5 话题层次化结构组织的实验及评测结果

在话题层次化组织算法 MLCS 中, 通过调整相似度初始阈值 θ 、相似度递减步长 q 、简单聚类循环次数 k , 可得到不同的有向无环图结构。我们通过参数 θ 、 q 、 k 的不同组合, 来判断不同的结构对效果的影响。我们利用系统 3 在 TDT4 汉语语料上做了相关实验, 并用其参加了 TDT2004 层次化话题识别任务的评测。

在 TDT4 汉语语料中的实验结果如表 5.2 所示, 其中的时间分批间隔 t 取为 7。

表 5.2 MLCS 算法生成的不同话题结构的效果对比表

结构类型	θ	q	k	(Cost(detect)) _{Norm}	(Cost(travel)) _{Norm}	MinimumCost
平面形	0.35	0.00	1	1.0000	0.0000	0.6600
梯形	0.35	0.05	3	0.1959	0.5128	0.2410
锥形	0.35	0.05	$+\infty$	0.1501	0.2130	0.1715

表 5.3 TDT2004 中 HTD 任务所提交结果的参数与效果

提交结果	θ	q	k	(Cost(detect)) _{Norm}	(Cost(travel)) _{Norm}	MinimumCost
ICT3a	0.35	0.05	$+\infty$	0.0857	0.1049	0.0984
ICT3b	0.35	0.06	$+\infty$	0.1055	0.1030	0.1039
ICT3c	0.35	0.07	$+\infty$	0.1018	0.1016	0.1017
ICT3d	0.35	0.08	$+\infty$	0.0775	0.0931	0.0878
ICT3e	0.35	0.09	$+\infty$	0.1036	0.0936	0.0970

表 5.4 提交的 5 个结果所对应的有向无环图结构

深度	宽度				
	ICT3a	ICT3b	ICT3c	ICT3d	ICT3e
0	1	1	1	1	1
1	2	221	2	60	889
2	321	2,723	625	2,509	10,621
3	2,119	10,928	6,031	14,609	31,514
4	7,518	23,578	18,987	33,939	57,670
5	16,774	38,852	36,418	57,670	87,472
6	28,342	57,670	57,670	87,472	278,109
7	41,217	87,472	87,472	278,109	
8	57,670	278,109	278,109		
9	87,472				
10	278,109				

从实验结果可以看出，“平面形”的结构，也就是不分层，即传统的话题识别方法，当考虑进遍历代价的时候，效果最差，因为用户需要浏览的类数太多，以致无法利用有限的精力找到关注的话题（这一点从表 5.2 中识别代价和遍历代价的值可以看出）；“梯形”结构效果好一些，但遍历代价还是偏高，同时又影响了识别代价；“锥形”效果最好，即将文档从初始阈值开始逐步分层组织，直到自动生成根结点为止。

表 5.3 是我们在 TDT2004 层次化话题识别任务中提交的 5 个英文系统的结果，语料中一共包含 278,109 篇英文报道。表 5.4 是这 5 个结果的有向无环图结构。

从评测的结果可以看出，基于“锥形”的有向无环图结构，采取不同的参数 q 后，

虽然具体的宽度和深度有所不同，但各个系统的效果差别并不大，这说明算法对于参数 q 不敏感，因而 MLCS 算法是比较稳定的。

我们利用系统 3 参加了 TDT2004 的层次化话题检测任务的评测，在英语、跨语言子任务中都取得第二名的成绩（汉语系统只有我们一个队伍提交）。图 5.10 是各个队伍评测成绩的对比。

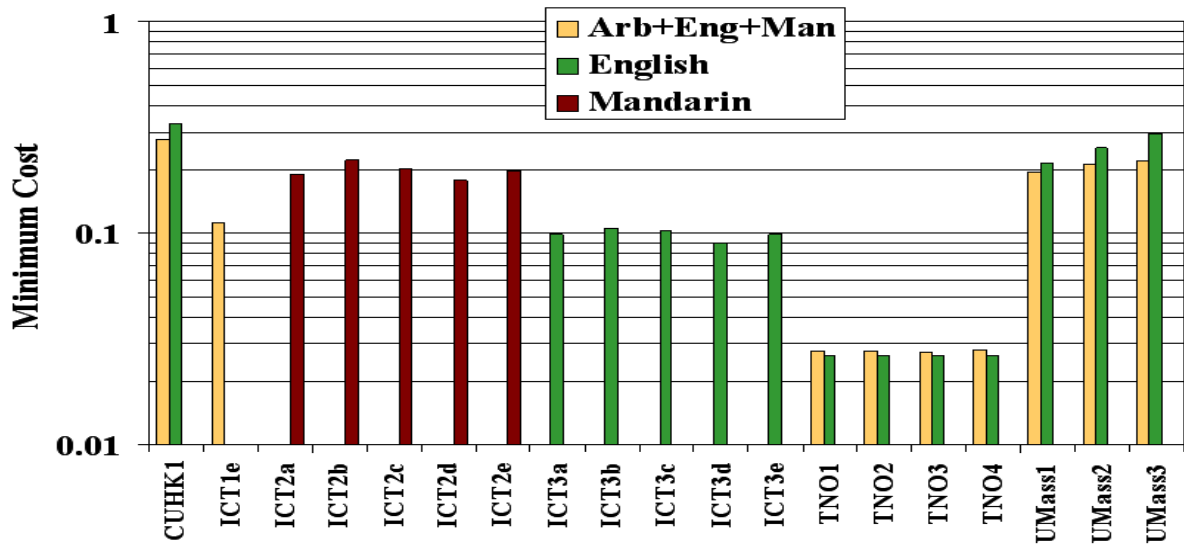


图 5.10 TDT2004 中 HTD 任务评测结果对比图

5.3 本章小结

本章主要研究了基于事件的信息组织技术。对于人物追踪来说，事件的组织主要分为微观粒度的事件组织和宏观粒度的事件组织。微观粒度的事件组织主要是借用信息抽取的技术，把一些特定领域、复杂程度较低的事件角色框架识别出来，丰富人物的履历内容；宏观粒度的事件组织是以文本或段落为基本单位，借用信息检索的技术把具有事件内在联系报道组织在一起。宏观粒度的事件组织由于不受事件领域的限制，可以对各种类型的事件进行大规模地高效处理，适合于建立人物的活动报道专集。

由于限定领域的事件信息抽取技术已经比较成熟，我们不对微观粒度事件组织进行过多的研究，只是在抽取人物的职务时，把特征向量中含有“称作”、“担任”、“免除”等动词的句子作为人物的特定事件描述。

宏观粒度的事件组织方法，国际上有“话题识别与跟踪”这样一个研究角度。因为话题识别与跟踪是把不同人物参与的围绕同一话题的事件组织在一起；而人物追踪是把同一人物参与的围绕不同话题的事件组织在一起，它们之间涉及的基本技术是相通的。所以，我们以“话题识别”(Topic Detection)任务为研究平台，来开展对宏观粒度事件组织的研究。

我们首先分析了宏观粒度事件的特点，并根据这些特点提出了三种话题特征调整策略。主要是考察了话题的时序属性、事件角色特征词对话题识别的影响、指代消解在话

题识别中的应用。

基于话题具有层次包含关系的特点，我们在话题识别的基础上，进一步研究了话题的层次化组织方法。为此，我们提出了多层聚类算法 **MLCS**，该算法主要通过相似度阈值来控制生成每一粒度层次的话题结点，然后再自底向上地把各个层次的结点组织起来。用户就可按照层次结构进行浏览。

我们在 **TDT4** 语料上开展了相关的实验，并在 **TDT5** 语料上参加了国际性的评测。通过实验对比，我们发现所提出的方法是有效的，并在 **TDT2004** 公开评测中取得了不俗的成绩。

第六章 若干细粒度挖掘任务的处理

本文前面所研究的数据预处理、人物模型的同一性判别和动态填充为人物追踪的研究打下了良好的基础；第五章所研究的基于事件的信息组织，把人物参与的事件信息有效组织，能够建立出人物的“重大事件信息表”和“活动报道专集”。

事实上，人物追踪还包含多个细粒度的深入挖掘任务，可以按照多种方式进行，例如：人物的横向（空间）跟踪（人与人的联系，人与单位的联系，人与社会的联系等）和人物的纵向（时间）跟踪（人的职位变化，人的处所变化，人的态度立场变化，人的生活状态变化，人的知名度变化等）等。

在本章中，我们对人物追踪若干细粒度挖掘任务进行研究，主要介绍了一些任务的解决思路和方法，并着重研究了人物知名度变化的计算方法。

6.1 概述

人物追踪可以有若干细粒度的深入挖掘任务，具体内容大致如下：

- 建立人物的事件信息表：对人物参与的事件进行组织，提供符合事件特点的结果呈现方式。
- 识别人物的社会关系网：识别社会上各个人物之间的横向联系。
- 识别人物职位的变化：识别人物职位的升迁和转移，挖掘人物的职业生涯。
- 识别人物处所的变化：识别人物所在地点的变化，确定人物的地点活动范围。
- 识别人物生活状况的变化：识别人物婚姻、家庭、亲友等的变化。
- 识别人物态度立场的变化：通过人物的相关报道，识别该人物对于某一事件的态度立场及其变化。
- 识别人物知名度的变化：通过相关报道判断人物知名度的变化，绘制人物知名度变化的统计图。

从这些细粒度挖掘任务来看，“人物的事件信息表”可以通过 5.1 介绍的微观粒度事件组织的方法，抽取对人物来说类型比较重要的事件角色，然后按照时间顺序进行排列。

“人物社会关系网”的信息在人物模型的“共现人名向量”中有所体现。这是因为，人名的共现关系可以体现人物之间关系的密切程度，在共现人名向量中，每一维的权重就是对这种关系密切程度的度量。我们可以对共现人名向量的各个维度按照权重由大到小进行排序，排在前面的若干人物就是该人物交往密切的人群。

“识别人物的职位、处所、生活状态的变化”主要依赖于人物模型的同一性判别和动态填充的基础。如果人物的属性及其关系抽取得足够好，经过人物同一性判别和人物模型的整合，可以呈现出清晰的结构。需要注意的是，既然是识别属性的变化，就需要把人物的属性信息跟时间进行绑定，每一个属性都可以看作是时间轴上的一段区域或一个坐标。

人物态度立场变化的识别需要较深入的自然语言处理技术，目前还没有有效的解决方法，比较相关的是“语气分类”（sentiment classification）方面的研究[Whitelaw 2005]。语气分类就是对文本的语气进行分类，比如按照肯定、中立、否定进行分类等。它在数据挖掘、市场调研与客户关系管理等方面有着广泛的应用。

目前，语气分类的方法可以分为两种：第一种是基于文本全部词汇的分类，根据词语的出现频率来训练出一个肯定与否定分类器，可能对个别词的权重进行调整；第二种方法把词语区分成“好”或“坏”两类，然后计算整篇文档的“好”或“坏”得分来对文档分类，这种方法应用得较为普遍。然而，要想对第二种方法解决得好，往往需要定义完整的语气分类体系。而且，态度表达的原子单位往往不是所谓的词语，而是一些关系紧密的表达一个语气的词语集合，称为“评价团”[Whitelaw 2005]。评价团由四个属性的词语构成：态度、语气、强度与极性。态度词主要是描述个人情感状态的词（例如“愉快”、“生气”等）；语气词表达肯定与否定；强度词表达态度的强烈程度（例如“非常”、“有点”等）；极性词表示一个词的使用极性，例如“愉快”（极性）与“不愉快”（非极性）等。

一般来说，人物态度立场变化的识别涉及到两个步骤：态度立场语段的抽取；语段倾向性的分析。前者可以依赖特定的对象词为触发特征进行语段的抽取；后者依赖语段倾向性的分析。判断倾向性，不是看语段中词的多寡，而是看语段中围绕着一一些词的一些词的多寡。这样，语段中的词可以形成“对象词”和“倾向词”两个属性空间，它们的笛卡尔积构成“张量空间”[白硕 2005]。图 6.1 是语段倾向性的张量模型的示意图。

对象词		
	倾向词	强度
	倾向词	强度
对象词		
	倾向词	强度
	...	...
...		

图 6.1 语段倾向性的张量模型示意图

人物知名度的计算结果可以用一个曲线图来呈现，曲线图表示出人物的知名度随着时间的变化。有了人物同一性判别的基础，知名度变化的识别变得相对简单。一般来说，可以利用一些启发式知识来进行识别。这些启发式知识包括：人名在新闻标题或正文开头出现的数量；在新闻正文中出现的数量；人名相关网页在搜索引擎中的排名情况；人名在各个媒体中的分布情况；报道人物信息的新闻媒体的级别程度等等。通过一段时间内相关信息的统计值来判断某一人物的知名度变化。下面，本文对人物知名度的计算方法进行详细的介绍。

6.2 人物知名度的追踪方法

6.2.1 词语流通度的计算

在现实社会里，一个语言现象能不能被认可，很大程度上决定于语感。无论是个人语感还是公众语感，类似于是一个“黑箱”，即看不见也摸不着。但是媒体的报道却相当于一个“白箱”。可以通过对“白箱”的测量来反映“黑箱”的情况。为此，张普教授提出了“流通度”概念，希望通过测量流通度来对语感加以数学界定、加以量化，使得“能不能说”、“被认可”、“被传播”变得可以通过流通度的计算进行判定[张普 1999]。

“流通度”(circulation)是一种语言现象在社会传播中的流行通用程度。流行高，人们的视觉、听觉已习惯于接受，就感觉能说，否则，就觉得陌生，不顺畅，不能说。

语言的流通度与社会传媒的流通度密切相关。流通度具有两个层次的含义：媒体的流通度和语言现象的流通度。首先是媒体具有流通度的属性和属性值，其次是刊载于这种媒体的文本具有流通度的属性和属性值，再次是依据这种文本所获取的语言知识也具有流通度的属性和属性值。其中，媒体的流通度分析，是整个流通度研究方法的最基本、最重要的步骤。[张普 2000]提出了书面流通度的最简单最基础的计算公式：

$$\text{流通度} = \text{流通量} \cdot \text{流通密度} \cdot \text{流通空间} \cdot \text{流通率} \cdot \dots$$

其中，流通量主要指媒体的发行量；流通密度主要指媒体的发行周期；流通空间主要指媒体的发行地区；流通率主要指媒体的阅读率。

在张普教授主持下，北京语言大学开发了动态流通语料库(DCC, Dynamic Circulation Corpus)，把语料库的建设和使用从静态推向动态，把文本的选择和抽样从分布原则推向流通原则。该语料库成为实现和检验流通度理论的物质基础，是动态语言知识更新理论的实验场地和研究平台。

流通度是在对频度、使用度进行考察的基础上提出来的。“频度”指一种语言的元素在文本中的出现次数与同类元素的总数之比。

“使用度”(usage)是综合频度以及该字符串在不同的语料类和不同的篇章中的分布三方面因素而计算出来的压缩了的频度。它的引入是从字符串分布的角度来修正单纯频度统计所带来的偏差。如果字符串在一定范围内的文本和语料类中散布得不均匀，即使它的频度很高，通过散布程度加权后所得的值就不一定会高。

“流通度”则从更大的媒体范围对“使用度”进行加权，考虑媒体的发行周期、发行数量、发行范围、阅读量等因素对字符串的影响，考虑了媒体对人群的影响，把人的“语感”同“流通度”联系起来综合考虑。以下是[王强军 2003]采用的三者的计算公式。

频度：字符串在任意一时间点上的频次/该点上字符串的总频次

使用度：频度 · (字符串出现在任一时间点上的文本数/该点上的文本总数) · (字符串出现在任一时间点上的媒体数/该点上文本总媒体数)

流通度：使用度 · 流通度种子 M · 媒体发行地域系数 · 媒体发行周期系数

“流通度”和“动态流通语料库”与传统语料库的本质区别就在于为语料加上了一个“时间”标记。无论是“频度”、“使用度”还是“流通度”，它们都已经不再是一个“点”，而是随时间推移而发展的一条曲线，一条可以观察到历时变化的曲线。在这种情况下，就可以绘制一个字符串的动态跟踪曲线图，可以观察到这个词语产生、发展、长期持续或突然消亡的时空变化轨迹。

6.2.2 人物知名度的计算

与词语流通度计算方法类似，人物知名度的判断也只能通过媒体中对人物的报道情况来度量。一个人物在媒体中报道得多，这个人物就容易被广大群众所熟悉，知名度就可能高。

在一个时间点上，人物知名程度的高低首先跟该人物的人名在媒体中出现的次数（“频次”）有关。人名出现次数越多，对应人物就可能越出名。当然，也要看该人物的名字出现在报道的什么位置。名字出现在新闻报道的标题和第一段，就会比出现在报道的其它位置显得重要。但是，在两个不同时间点上的同一个人物的知名度是涨还是跌，不能完全按照频次来判断。因为媒体的信息量在不断增长，媒体信息总量越大，对人物的报道可能就越多，虽然人名出现的次数增加了，但实际的知名度却可能下降了。因此，需要引入“人名总频次”因素进行修正。二者的比值相当于“流通度”计算中的“频度”。网络媒体与文本媒体相比，有一个很大的优势，即可以对每一篇报道页面进行打分，通过页面上的链接关系计算出页面的 PageRank 值。出现在 PageRank 值高的页面上的人名当然就重要。

当然，人名的频度只是一个因素，知名程度还跟人名“使用度”有关，即散布文本数和散布媒体数越多，人名分布就越均匀，使用度就越高，该人物越倾向于知名。例如，频度相同的两个人物，如果一个只在单一媒体中报道，另一个却在多个媒体上均有报道，那么，后者可能更被人们所熟悉，越可能知名。

人物知名程度还跟报道该人物的媒体有关，在著名媒体上报道的人物就比一般媒体上的人物知名。在传统的流通度理论中，考虑较多的是文本媒体的影响因素：发行量、发行周期、发行地区、阅读率等。但是网络媒体中的因素与之有所不同。对于网络媒体，地域因素有所弱化，因为电脑使用者通过互联网几乎可以访问世界上任何一个地区的网站。发行周期也变得不太重要，因为好多网站的内容都是增量式增加。跟网站最相关的因素可能是访问量、主页访问率、页面点击率等。然而，最困难的同样是有关数据的获取问题。

下面，我们探讨一下如何来衡量新闻媒体的重要度。

一直以来，人们关注的焦点是某一特定网站在各大搜索引擎中（如 Google、MSN 等）的排位问题，但由于各个搜索引擎的排位规则、更新周期和收录的 URL 不尽相同，导致缺乏一个公正的评价指标来对一个网站的世界排名进行综合评估，这时便需要参考一些专门发布网站世界排名的网站的统计数据。Alexa (<http://www.alexa.com/>) 就是一家

这样的网站。

Alexa 排名是目前常引用的用来评价某一网站访问量的一个指标。事实上, Alexa 排名是根据对用户下载并安装了 Alexa 工具条嵌入到 IE 等浏览器, 从而监控其访问的网站数据进行统计的, 因此, 其排名数据并不具有绝对的权威性。但由于其提供了包括综合排名、用户链接数 (Users Reach)、页面浏览数 (Page Views) 等多个评价指标信息, 且目前尚没有而且也很难有更科学、合理的评价参考, 大多数人还是把它当作当前较为权威的网站访问量评价指标。

Alexa 的网站排名是按照每个特定网站的被访问量进行排名的。访问量越大, 排名越靠前。访问量的计算主要取决于用户链接数 (Users Reach) 和页面浏览数 (Page Views) 两个因素。Alexa 系统每天对每个网站的用户链接数和页面浏览数进行统计, 通过这两个量的三个月累积值的几何平均得出当前名次。

用户链接数 (Users Reach) 指通过 Internet 访问某个特定网站的人数。用访问某个特定网站的人数占有所有 Internet 用户数的比例来表示。即: $\text{用户链接数} = (\text{访问人数} / \text{全部 Alexa 用户数}) * 100\%$, Alexa 以每百万人作为计数单位。以雅虎 (Yahoo) 为例, 如果它的用户链接数为 28% 的话, 就是说随意抽取一百万的 Internet 用户, 其中有 280,000 人访问 Yahoo。

页面浏览数 (Page Views) 是指用户访问了某个特定网站的多少个页面, 是所有访问该网站的用户浏览的页面数之和。每个用户浏览的页面数取平均值, 是所有访问该网站的用户每天每人浏览的独立页面数的平均。同一人、同一天、对同一页面的多次浏览只记一次。

当然, Alexa 的统计也受一些其它因素影响。例如, 受使用 Alexa 工具栏用户的语言、地域、文化等各方面的影响 (英文网站相对于其它语言的网站, 访问量数据更容易被充分地统计); 受网站对自己宣传的程度、打广告的多少、别的网站为其建立链接的多少的影响。但可以肯定的是, 任何网站排名的计算方法都是不完善的, Alexa 的统计方法还是相对公正和科学的。表 6.1 列出了 2005 年 12 月 5 日 Alexa 发布的中文网站流量前 10 名的具体结果。

表 6.1 2005 年 12 月 5 日 Alexa 发布的中文网站流量前 10 名基本情况

网站名	用户链接数	页面浏览数	访问量
Baidu.com	76,590	12.2	966.6
新浪新闻中心	63,990	11.1	842.8
搜狐	43,500	12.2	728.5
网易	49,120	8.2	634.7
腾讯网	43,160	7.9	583.9
淘宝网	15,385	15.4	486.8
3721 网络实名	35,550	2.7	309.8

TOM.COM	17,180	7.4	356.6
雅虎新闻	18,685	5.5	320.6
Yisou.com	16,885	4.2	266.3

基于以上分析，我们采用如下公式来计算某一时间点上的人物知名度。我们提出的方法在[王强军 2003]所介绍的流通度计算基础上，进一步考虑了人名在报道中的出现位置、页面的 PageRank 值以及网络媒体的访问量等因素。

$$Dg_{Famous} = \sum_{i=1}^{N_{page}} \left(\frac{\alpha \cdot n_{title_i} + \beta \cdot n_{first_para_i} + \gamma \cdot n_{content_i}}{\alpha \cdot N_{title_i} + \beta \cdot N_{first_para_i} + \gamma \cdot N_{content_i}} \cdot PageRank_i \right) \cdot \frac{n_{site}}{N_{site}} \cdot \sum_{j=1}^{N_{site}} \left(\frac{n_{page_j}}{N_{page_j}} \cdot \frac{\sqrt{UserReach_j \cdot PageView_j}}{M} \right)$$

其中， n 代表当前人物的人名的统计个数， N 代表全部人名的统计个数。例如， n_{title_i} 就指当前人名在新闻网页 i 的标题中的出现次数； N_{title_i} 指新闻网页 i 的标题中出现的全部人名的个数； n_{page_j} 指当前人名在网站 j 上出现的网页数； N_{page_j} 指网站 j 所含的全部网页数； n_{site} 指当前人名出现的网站数； N_{site} 指站点总数。

α 、 β 、 γ 是权重因子，代表人名在新闻网页的不同位置的重要程度。一般地， $1 \geq \alpha > \beta > \gamma > 0$ 。

类似地，我们为物知名度计算选定一个“种子”媒体网站值 M ，每个网站的加权系数，例如访问量，即用户链接数（Users Reach）和页面浏览数（Page Views）的几何平均，都是跟这个种子网站对应属性的比值。

我们以 Baidu.com 的访问量作为“种子 M ”。那么每一个网站的加权系数

$$\delta = \frac{\sqrt{UserReach_j \cdot PageView_j}}{M} \text{ 分别为:}$$

$$\text{新浪新闻中心_} \delta = 842.8/966.6 = 0.87$$

$$\text{搜狐_} \delta = 728.5/966.6 = 0.75$$

$$\text{网易_} \delta = 634.7/966.6 = 0.66$$

$$\text{腾讯网_} \delta = 583.9/966.6 = 0.60$$

$$\text{淘宝网_} \delta = 486.8/966.6 = 0.50$$

$$\text{3721 网络实名_} \delta = 309.8/966.6 = 0.32$$

$$\text{TOM.COM_} \delta = 356.6/966.6 = 0.37$$

$$\text{雅虎新闻_} \delta = 320.6/966.6 = 0.33$$

$$\text{Yisou.com_} \delta = 266.3/966.6 = 0.28$$

通过在每一个时间点上都为人物计算一个“知名度”的值，然后把各个点的值连起来，就可以得到该人物的知名度变化曲线。例如设置 12 个点，即以一年为一个大周期，以 12 个月为曲线变化的观察点。限于客观条件，我们无法动态地收集到网络上的报道，建立针对网络媒体的“动态流通语料库”，所以具体的实验作为将来的工作。图 6.2 是人物知名度曲线示意图。

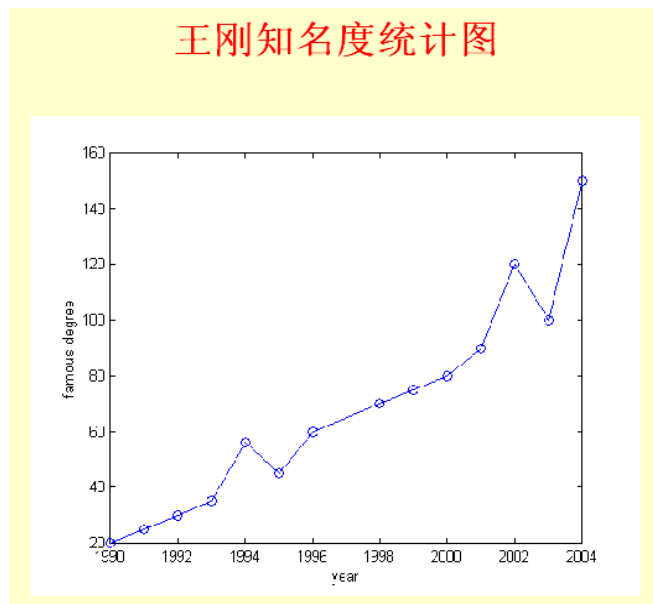


图 6.2 人物知名度曲线示意图

6.2.3 流行人物的挖掘

通过人物知名度的计算和跟踪，可以挖掘出当前的流行人物。流行人物一直是社会上公众关注的焦点，挖掘方法与“流行语”的挖掘有相通之处。

在“动态流通语料库 (DCC)”理论中，通过不断下载相关语料，对新时期的语言变化进行动态监测研究。在主流报纸流行语研究方面，该理论往往利用信息处理技术对词语的频度、散布度、流通度进行科学量化，然后计算全年平均流通度，并滤除全年中低频词和高频词，由计算机提取候选流行语，通过考察词语的曲线类型，最后确定出最终的流行语作为结果发布。

流行语的筛选往往考虑流行语的起点、峰值与谷值的落差、流行高峰持续期等因素 [史中琦 2004]。首先是“起点”，流行语的起点不允许太高，如果太高，就意味着这个词语以前也曾大量出现过，必然缺少新鲜感。然后是“峰值”，峰值是指流行语曲线所达到的最高点。一个词语若想成为流行语，必须在出现的频率上达到一定绝对高度。它往往是确定流行语的最主要因素。其实是“持续”，持续所需要达到的高度一般在最高峰值的 1/2。

所以，作为流行人物的挖掘，同样可以借鉴这些共性的规律：人名首先必须在一段时间内在数量上达到一定阈值；人名的起点不能太高；人名在达到最高峰之后一般要持续一段时间。

6.3 本章小结

文章主要介绍了人物追踪若干细粒度挖掘任务的解决方法。首先对“建立人物的事件信息表”和“识别人物的社会关系网”、“识别人物职位、处所、生活状况、态度立场的变化”等进行了方法上的探讨，然后重点介绍了“人物知名度”的追踪方法。

人物知名度的计算主要借鉴了“流通度理论”和“动态流通语料库理论”，并结合人名和网络媒体的特点，提出了人物知名度的计算方法。通过考察人物各时间点上的知名度情况，可以得到该人物的知名度变化曲线图，实现对人物知名度的跟踪。

在此基础上，我们还结合“流行语”的挖掘方法，探讨了“流行人物”的挖掘方法。

第七章 人物搜索引擎的设计

到本章为止，我们完成了人物追踪的基本研究工作，解决了人物追踪的技术难点。本章将要综合前面研究的技术来解决人物追踪的实际应用问题。

除了研究上的意义以外，人物追踪还可产生以下三种典型的应用：人物搜索引擎、特定人物的跟踪、流行人物的发现。其中，人物搜索引擎是最为典型的应用。由于目前的搜索引擎正在朝深入化、专业化方向发展，人物搜索作为一种常见的应用形式，定会得到广泛的发展空间。

为此，我们提出了一种人物搜索引擎——PerSou。下面本章介绍人物搜索引擎 PerSou 的体系结构、索引结构和搜索界面及结果。

7.1 人物搜索引擎的体系结构

在介绍 PerSou 的体系结构之前，我们先来介绍一下通用搜索引擎的典型结构。通用搜索引擎的结构一般由以下几个部分组成：采集器、索引器、页面库/知识库、搜索器、用户检索界面。

“采集器”又名“网络蜘蛛”，它从一些种子 URL 出发，沿着页面间的链接不断爬行，一方面采集下来经过的网页，另一方面提取出新的链接。

“索引器”往往是通用搜索引擎技术的集中体现，采用的向量空间模型 (Vector Space Model) 和 TF_IDF 标引策略直到今天仍作为一种行之有效的方法广泛采用。一般步骤是对文档进行分词、去除停用词、词根还原 (stemming) 等简单的预处理之后，对文档建立倒排表。主文档中的每一个词 (预处理之后保留的) 都在倒排表中有对应的入口词项。每个入口词都指向一个属性项链表，链表中每一项记录该入口词在主文档中的出现情况，如偏移量、(英文) 大小写等信息。

“页面库/知识库”是采集下来的网页和索引器生成的知识文件和记录的集中存放站，它是唯一与其它模块密切打交道的部分。由于要存放数以万个 G 的数据，因此它往往要占用令人望而却步的硬件资源，而且如此大量的数据也会给提高查找的效率带来比较大的困难。

“搜索器”根据用户的查询请求，到页面库/知识库中查找相关记录，再把查询结果返还给用户。在返还的结果中，要根据网页的相关度和重要性对结果进行排序，把质量最好的网页排在检索界面的前面。

“用户检索界面”接受用户查询，把查询提交给搜索器。它一般采用 cgi 程序形式调用搜索引擎提供的接口，用户可以自行设计界面。这样便于在用户网页上增加搜索功能，或实现元搜索。

人物搜索引擎 PerSou 所实现的功能与通用搜索引擎有很大差异。它对网页进行深加

工，挖掘出各个个人的信息，通过对信息的整合把结果返回给用户，因而也带来了体系结构的很大不同。图 7.1 是人物搜索引擎 PerSou 的体系结构示意图。

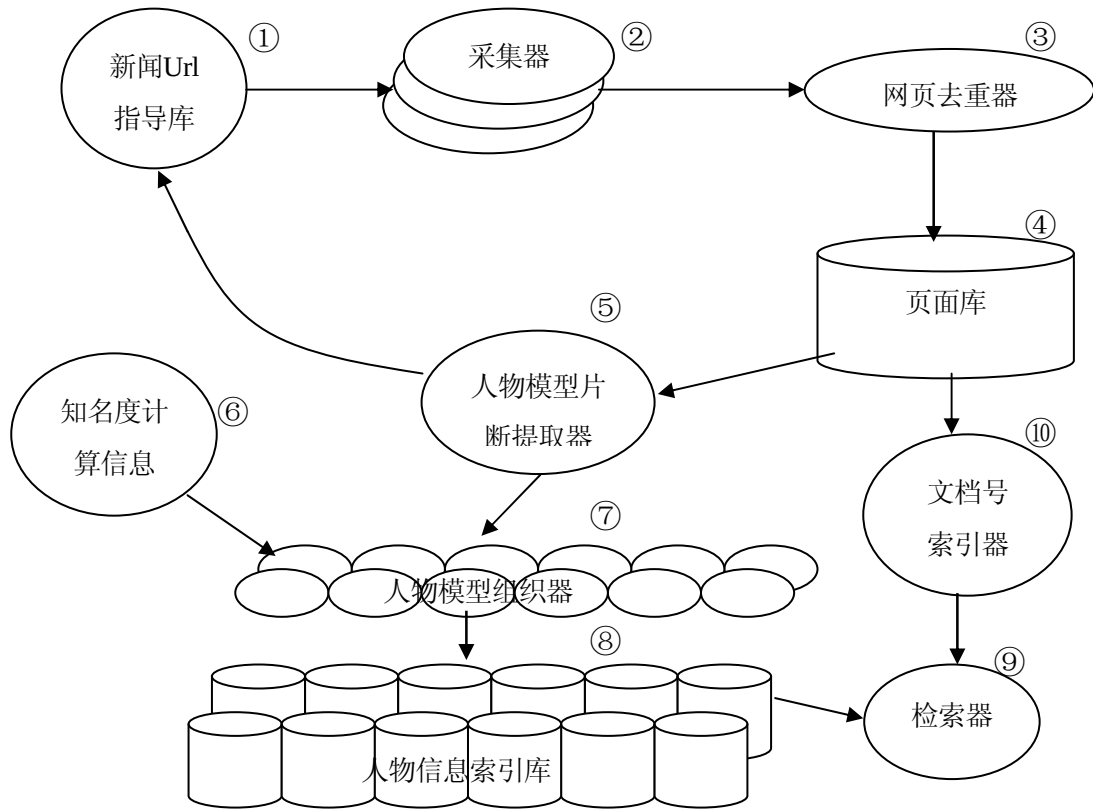


图 7.1 人物搜索引擎 PerSou 体系结构示意图

① 新闻Url指导库：指导PerSou只对新闻网页进行采集，因为新闻网页描述规范且报道及时。

② 采集器：PerSou使用分布式采集系统定时地遍历网页，将遍历到的网页送到网页去重器中。

③ 网页去重器：根据历史记录检测重复网页，只把未采集过的新网页存储到页面库中。

④ 页面库：存储采集下来的页面。由于数据量巨大，所以使用zlib等格式进行压缩。页面库获得每个网页的完全Html代码后，对网页及URL进行分析，记录下网页长度、URL、URL长度和网页内容，并赋予每个网页一个文档号(docID)，以便当系统出现故障的时候，可以及时完整地进行网页的数据恢复。

⑤ 人物模型片段提取器：主要是进行人物追踪的数据预处理工作，具体内容包括：网页噪音过滤、词法分析、人名识别、人名指代的处理、人物对应语段的划分、时间和时序标签的确定、人物属性的抽取、人物特定类型事件的抽取，以及人物模型片段信息的分发（按姓氏范围分发给不同的人物模型组织器）。

⑥ 知名度计算信息：用于进行人物知名度计算的必要信息，例如网页的 PageRank

值等。该信息由其它系统模块计算出来。

⑦ 人物模型组织器：采用分布式系统对人物模型片断信息进行处理，包括同名人物的判别、人物事件信息的组织、人物知名度判断等。

⑧ 人物信息索引库：存放人物信息的索引文件，主要有：姓名索引文件、同名人物文件、人物履历文件、人物活动报道文件。

⑨ 检索器：接受用户查询，把查询提交给搜索器。接受的查询词主要是人名，然后从人物信息索引库中查找出该人名对应的同名人物信息；也可接受人名和属性词的组合查询，那么就把最相关的人物信息返回给用户。检索器同时跟文档号索引器进行交互，通过文档号取得报道该人物的原始网页。

⑩ 文档号索引器：通过文档号，可以取得对应文档在页面库中的位置。

7.2 人物搜索引擎的索引结构

在通用搜索引擎中，索引的存放方式主要有倒排文件方式，后缀树（suffix tree）方式和签名文件（signature file）方式。其中，倒排文件被广泛地应用，它是一种十分快速的检索文件，任意给出一个查询词，从该检索词对应的入口词直接就可一次检中包含该词的文档。

从根本上说，倒排文件的基本结构是一个词表，每个表项是系统可以识别的一个词汇。从每个表项引出一个指针，指向一个数据结构，此结构记录了该词汇在所有文本中的出现位置和频率等信息。倒排文件的生成过程是这样的：取得一个文本之后，顺序扫描它，用一个分词词典对它进行分词，记录下每一个词的出现位置和频率，存入倒排表。合并所有文本产生的倒排表（当然，要记录下来词汇出现的文档），就得到倒排文件。

倒排文件的组织还需要在检索效率和更新效率上进行折衷。一般为了提高更新效率，倒排索引的索引项数据用链表方式分块存放有利，但会降低检索效率；反之，索引项数据连续存放有利于检索，而不利于更新。所以，面对大数据量情况下倒排文件生成的问题，一般的解决方案是分批阶段性的生成倒排文件，即把文件库划分成若干个部分，对于每一个划分分别建立倒排文件，再将各个倒排文件进行合并最终生成一个倒排文件。

同时，索引文件往往占用大量的存储空间，应采用适当的压缩算法，对索引文件进行压缩。对压缩算法的要求是，解压的效率要远高于对于压缩效率的要求，因为压缩是在建立索引时进行的，是离线的过程；而解压是在检索时实时进行的。

这样看来，对于通用搜索引擎来说，倒排文件方法有很突出的优点：

(1) 实现简单。因为原理很简单，所以几乎所有的工作都集中在数据结构的设计和编码上。一个好的数据结构是非常重要的，它会影响到系统的诸多方面，最重要的是：查找速度、存储空间和可维护性。只要数据结构设计合理，就会有效地缓解这些问题。

(2) 查找速度快。倒排表的结构是直接适应查询请求的。对于只含关键字的简单查询，系统要做的就是分析关键字，从倒排表中查找相应关键词，然后把它们对应的文档信息

提取出来进行合并输出。即使文本库的文本数量达到百万级，这个过程仍然是相当快的，一般几秒钟内会得到结果。如果含有逻辑操作符和其它操作符的查询，这个过程会复杂一些，但增加的复杂度仍然是线性的。

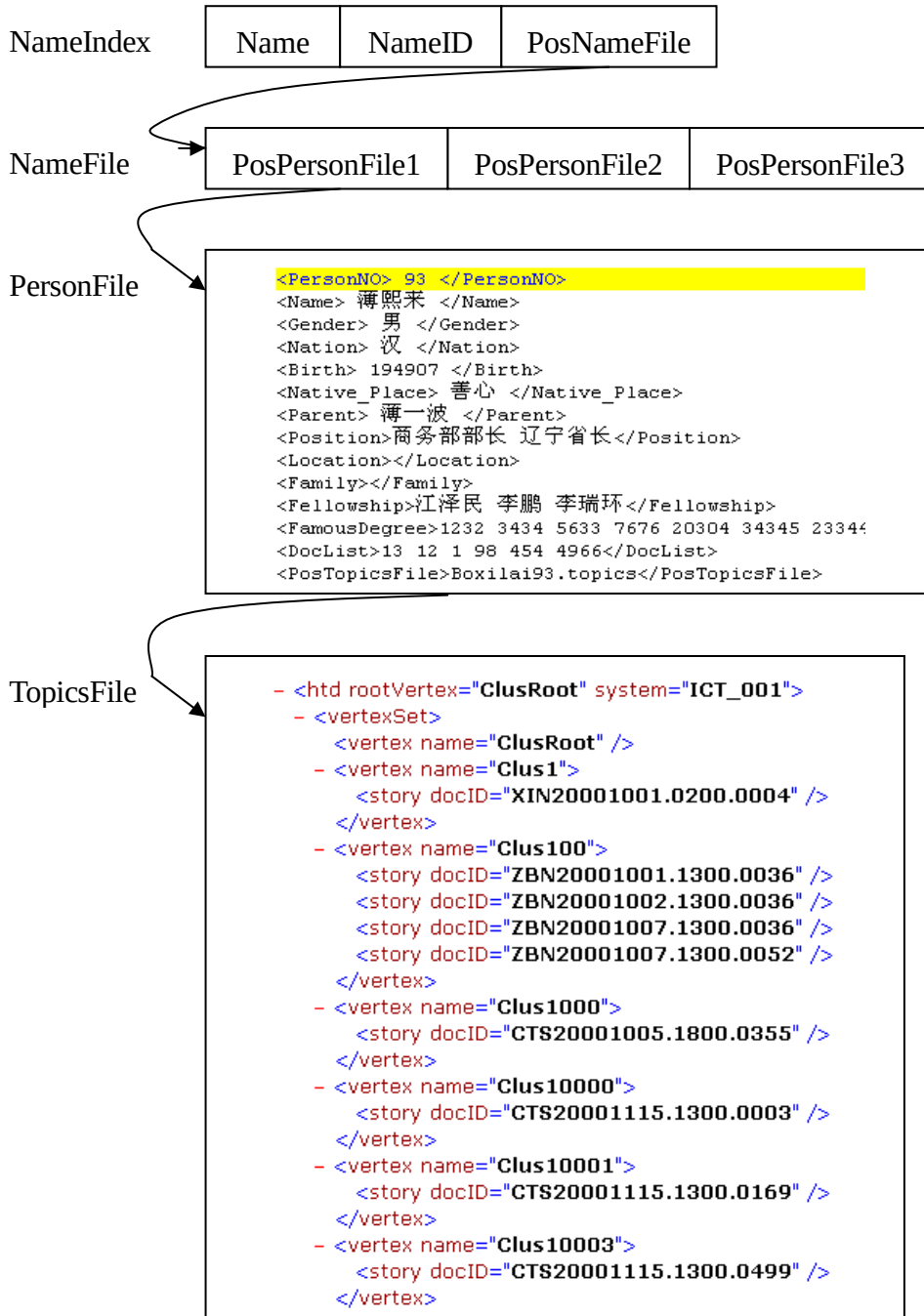


图 7.2 人物搜索引擎 PerSou 索引结构示意图

当然，倒排文件方法的缺点也是比较明显的：

(1) 要求海量存储器。由于倒排表中存储了大量词汇在文档中的位置、频度等数据，其数据量是非常庞大的。研究表明，倒排文件的大小有可能达到源文件的 300%；即使是经过压缩，还是要占到源文件的 40% 以上。

(2) 数据结构维护困难。这是倒排文件方法最主要的缺陷。一旦倒排文件被建立，就

很难进行插入和删除操作。然而，在目前巨大的 WEB 空间环境下，每天都有巨大数量的新网站和新网页上网，每天也有巨大数量的网页和网站被从网上撤除。这样一来，对索引文件的插入和删除操作不但是必不可少的，而且这种操作的数量会保持在很高的水平上。如果系统对这些需求没有考虑或者处理不当，会带来很多问题。例如：

大量的索引合并操作。在大规模应用情况下，因为数据量极大，合并操作很可能非常耗时，造成服务中断，响应迟缓。对于商业应用来说，这是非常致命的问题。

大量的无效链接。因为倒排文件的入口是词汇，所以要想从倒排表中删除无效链接是非常困难的，往往需要遍历整个数据结构。要想解决这个问题而不触及数据结构，要么是什么都不做，要么是在向用户返回链接时加入判断逻辑。前一种方法会因为空指针的问题白白占用大量磁盘空间，而且无效链接会给用户带来负面印象。

总的来说，倒排索引本身是一种静态数据结构，不能很好使用在需要频繁更新的文档数据库环境中。为此，我们结合人物搜索的具体应用，提出了人物搜索引擎 PerSou 的索引结构，如图 7.2 所示。

PerSou 的索引文件主要由四部分构成：姓名索引文件 NameIndex、同名人物文件 NameFile、人物履历文件 PersonFile、人物活动报道文件 TopicsFile。

其中，NameIndex 文件记录了人名 Name、人名对应的编号 NameID 和该人名对应的同名文件位置 PosNameFile；PosNameFile 文件则记录了记载该人名对应的各个人物信息的文件的位置 PosPersonFile1、PosPersonFile2、PosPersonFile3 等；PersonFile 文件记录了某一人物的具体信息，是人物模型信息的集中体现；TopicsFile 文件记录了该人物的活动报道专集的信息，包括各个话题的信息和话题对应的文档号列表。

因为每个人物的信息都是动态的、更新频繁的，所以对每个人物的信息单独建立一个索引文件进行存储，利于实现对人物的实时跟踪。

7.3 搜索界面及结果

前面探讨了人物信息的处理和存储问题，人物信息的呈现形式也同样非常重要。

PerSou 采用了两个阶段的返回结果来呈现人物的信息。首先是搜索接口界面，PerSou 接受用户输入的人名信息，或人名和属性词的组合信息。具体界面如图 7.3 所示。

一般地，如果输入的是人名，搜索界面上就会呈现出当前人名的同名人物信息。每个人物都用简短的属性信息词汇进行标识，便于用户一目了然地进行判断到底关注的是哪一个人物。例如，“王涛”的同名人物信息呈现结果，如图 7.4 所示。图 7.4 是前文介绍的“人物模型同一性判别”算法的真实结果体现。

在同名人物信息结果页中，如果用户发现了关注的人物，通过点击该人物的链接，PerSou 就会返回该人物的具体信息，包括简历、相关人物、活动报道专集和知名度变化曲线图等。真实结果如图 7.5 所示。这页就列出了“乒乓球运动员”“王涛”这个人物所对应的人物模型中的大部分信息以及对该人物的跟踪信息。用户通过点击“相关人物”

的链接，可以方便地浏览跟他关系密切的人物的详细信息。

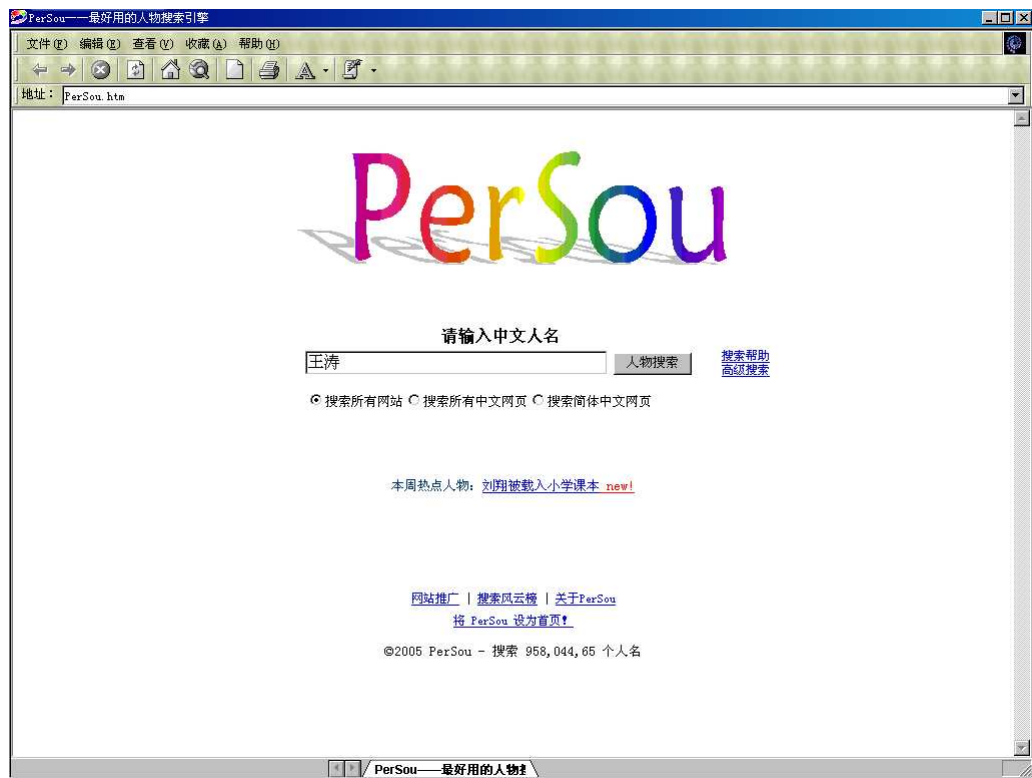


图 7.3 人物搜索引擎 PerSou 的搜索界面图



图 7.4 人物搜索引擎 PerSou 的同名人物返回结果示意图

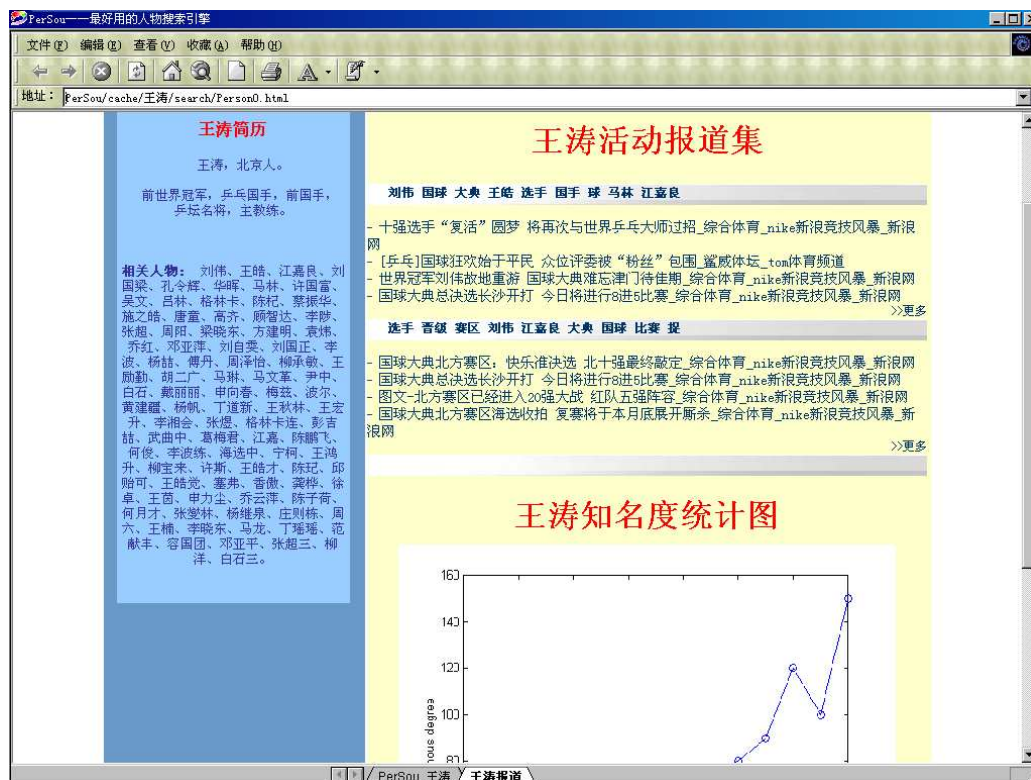


图 7.5 人物搜索引擎 PerSou 的具体人物信息示意图

7.4 本章小结

本章介绍了人物追踪的具体应用，主要是结合人物追踪的算法研究，介绍了人物搜索引擎的设计方法。具体内容包括：人物搜索引擎的体系结构、索引结构和搜索界面及结果。

在具体设计中，主要是吸收通用搜索引擎的经验，结合人物搜索的具体应用场景进行改进，以便于人物信息的计算、存储和更新。人物搜索引擎检索界面充分考虑到简单易用的因素，提供两个阶段的结果。先返回同名人物的信息，再返回某个人物的具体信息。

第八章 结束语

8.1 本文主要贡献与创新

本文介绍了自由文本空间内人物角度信息组织的技术及应用，提出了“人物追踪”的概念平台，并围绕人物追踪各个环节的一系列关键问题从理论上和实现方法上进行了深入分析和探讨，主要内容包括人物追踪的数据预处理，人物模型的同一性判断及动态填充，基于事件的信息组织，人物追踪若干细粒度挖掘任务的处理和人物搜索引擎的设计等。

本文的主要贡献和创新表现在以下几个方面：

- “人物追踪”概念平台的凝练和深化

人物追踪，说到底并不是一个纯技术问题。我们为事物概念建立“人物模型”，然后，从基础和派生任务两个方向展开研究。人物追踪的基础包括：事件组织、人名/非人名区分、同名人物判别、时间和时序的确定、背景事件的确定等等。派生任务主要包括：识别人物的社会关系网；建立人物的事件信息表；识别人物职位的变化；识别人物处所的变化；识别人物生活状况的变化；识别人物知名度的变化等等。

以上内容可大体整合为两大部分：制作人物的履历表；对人物的事件进行组织。“制作人物履历表”就是从自由文本中自动提取出该人物的属性信息，并把分散的各种属性信息进行有机整合，形成完整的生平履历。“对人物的事件进行组织”，就是把人物参与的相关事件的报道按照内在发展逻辑有效地组织起来，形成清晰的逻辑结构。

- 汉语信息抽取中基于“概念义原”的语义扩展方法

汉语信息抽取一直是有待深入研究的问题。我们采用监督式机器学习的方法来抽取人物的属性，即手工标注一定规模的语料，然后训练出分类器用来判断新的实体对是否存在搭配关系。分类器采用特征向量的表示方法。我们在特征向量中引入基于“概念义原”的语义计算，即对特征向量中的词汇进行语义聚类，把聚类结果的每个词团作为统一的维度，并把该词团中每个词的权重之和作为该维度的权重。词的语义聚类是基于HowNet来进行的。这样，特征向量中每个维度都以“概念义原”作为统计对象而不是以词作为统计对象，可以消除表达同一个概念的不同的词在字面上的差异，突出关键性的特征，解决语料稀疏的问题。

- 人物模型的同一性判别和数据融合算法

我们深入研究了人物模型中各个维度数据的特征表示和相似度计算问题，并结合各

个维度的数据稀疏程度，提出了基于混合类型数据的人物模型相似度计算方法。然后，我们分为两个阶段来进行人物模型的同一性判别。首先利用信息抽取结果的强匹配，把语料分为若干紧密的团，然后以这些团为“枢纽”，再进行混合类型数据聚类，得到最终的结果。通过对人物模型各个维度特点的深入分析，以及将信息抽取结果有效地运用在文本聚类中，我们不仅能够把不同人物的信息区分开，还能对同一人物的信息有效整合，形成完整的人物模型信息，为人物追踪打下坚实的基础。

- 事件信息层次化组织的有效方法和策略

我们以“话题识别”(Topic Detection)任务为研究平台，首先分析了宏观粒度事件的特点，并根据这些特点提出了三种话题特征调整策略。主要是考察了话题的时序属性、事件角色特征词对话题识别的影响、指代消解在话题识别中的应用。基于话题具有层次包含关系的特点，我们在话题识别的基础上，进一步研究了话题的层次化组织方法。我们提出了多层聚类算法 MLCS，该算法主要通过相似度阈值来控制生成每一粒度层次的话题结点，然后再自底向上地把各个层次的结点组织起来。这样，不仅能够有效地挖掘出话题团，还能组织好话题团之间的关系，用户就可按照层次结构进行浏览，有利于减少阅读工作量。

- 基于“流通度理论”的人物知名度计算方法

我们利用一些启发式知识来计算人物的知名度，包括：人名在报道中的频次和位置；人名所在网页在搜索引擎中的排名情况；人名在各个媒体中的分布情况；报道人物信息的新闻媒体的级别等等。计算方法借鉴了“流通度理论”和“动态流通语料库理论”，并根据人名和网络媒体的特点进行了调整。基于“流通度理论”的人物知名度计算方法能够把人物知名度的计算从静态推向动态，把文本的选择和抽样从分布原则推向流通原则，从而得到人物的知名度变化曲线图，有效挖掘流行人物。

- 互联网人物信息搜索引擎 PerSou

最终，我们将研究的理论与技术综合地集成到了 PerSou 系统中。在 PerSou 中，用户只要输入人名，该引擎就能够显示出同名人物的信息，并能够对每个人物提供履历信息和活动报道专集，以及该人物的知名度变化曲线图和相关人物信息。PerSou 是对传统搜索引擎在功能和技术上的突破，提供了建立大规模商业化人物信息搜索引擎的技术方案，具有良好的商业价值。

8.2 下一步研究工作

本文对自由文本中面向人物追踪的知识挖掘技术进行了卓有成效的研究，在人物追

踪数据预处理, 人物模型的同一性判别, 基于事件的信息组织和人物追踪细粒度的挖掘任务等方面取得了许多成果。作为一项探索性的初始研究, 本文的研究成果为人物追踪相关领域的研究打下了坚实的基础, 接下来的研究工作将围绕以下几个方面来进行:

- 联合多类型数据源的人物追踪

为了研究人物追踪的核心算法, 本文采用了描述相对规整的汉语新闻网页作为数据来源。在现实中, 人物信息会在方方面面的数据中得到体现, 例如数据库、报纸、电台、电视、邮件、短信、即时聊天等。因此, 有必要联合多个来源的异构数据对人物进行联合挖掘, 从而实现人物的全面追踪。

- 倾向性分析

倾向性分析是近年来出现的新的研究热点。通过判断倾向性, 把语段中的词分成“对象词”和“倾向词”两个属性空间, 从而产生新的文本表示模型。进一步挖掘整个社区对不同事物的不同态度的分布数据, 就可以生成“态度引擎”。输入代表一个事物的关键词, 就能输出网络舆论对这个事物的态度分布; 输入一组<关键词, 态度取向>二元组, 则返回针对这些关键词代表的事物具有相应态度取向的文档, 甚至可按态度取向的定量估值来排序。

- 热点事件预警

现有的事件组织技术往往倾向于事后分析, 而热点事件预警则倾向于事前分析。类似于天气预报, 热点发现需要一定的提前量, 即在该事件广为人知之前, 就要借助一些蛛丝马迹预测它是否将成为热点。这需要综合考虑事件的分布特点和内容特征, 并与用户的兴趣相结合, 从而缩小潜在热点事件的范围, 提供更准确、更有价值的结果。

- 其它抽象对象的挖掘

人物只是抽象对象的一个代表, 对其进行挖掘的研究可以借鉴到其它抽象对象的处理中, 例如: 地点、机构、音乐、软件等。拿“地点”来说, 就存在着地名歧义, 例如“新街口”, 指的是北京的, 还是南京的等; 关于某一地点的信息可能出现在文本中, 也可能出现在地图中, 还可与 GPS 进行结合; 地址的名称会发生沿革和变迁, 例如“金陵”与“南京”等。通过对各个抽象对象的挖掘, 使文本处理由文字层面发展为抽象对象层面。

8.3 前景与展望

人物是抽象对象的典型代表, 对其进行挖掘代表了文本挖掘技术的最新进展和新的

增长点。人物追踪要解决的人物相关信息的“同一性判断问题”、“数据融合问题”和“跟踪问题”，将为与人物处于同一抽象层次的其它对象（例如事件、地点、机构、音乐等）的挖掘提供技术上的参考，也将为抽象对象挖掘本身凝练理论和方法。除了学术上的意义，人物追踪的研究还可以产生一些重大的产业级应用，例如人物搜索引擎、特定人物的追踪、流行人物的发现等。

就拿人物搜索引擎来说，随着互联网的发展，人们对互联网信息服务的需求正呈现多样化发展的趋势，传统的搜索引擎并不能完全满足这种要求。尽管目前流行的信息服务形式很多，例如“图片搜索”、“歌曲/歌词搜索”、“热点新闻搜索”等，但从人物角度来组织信息的服务还不常见。目前已经公开的技术只能根据提交的人名关键词返回含有该人名的原始网页，处理的精细程度不够；或只能对部分名人的信息进行较精细的处理，返回相关名人的特征属性和特征人群，所能处理的人物范围有限。在这种情况下，人们要想了解一个在互联网上出现过的人物的信息，往往不得不借助于通用领域的搜索引擎，例如 Google 等，键入相关的人名，得到返回页面，其中不乏与同名同姓的不同人物相关的页面，也不乏与人物姓名词形相同的非人物实体；页面只是采取简单罗列的方式，未根据人物的特征进行整理，用户无法对所关注人物的信息进行清晰的把握。

这样，人物追踪技术的研究就可以克服现有技术的缺点，解决目前无法对人物信息进行大规模有效处理的问题，提供一种新颖实用的互联网人物搜索引擎的生成方法。用户只要输入人名，该引擎就能够显示出同名人物的信息，并能够对每个人物提供履历信息和活动报道专集，以及该人物的知名度变化曲线图。

随着互联网的进一步发展和业务模式的驱动，相信人物追踪的前景将会越来越广阔。

参考文献

- [ACE] <http://www.itl.nist.gov/iad/894.01/tests/ace/>, Site visited on August 30th, 2005
- [Agichtein 2000] E. Agichtein and L. Gravano. Snowball: Extracting relations from large plain-text collections. In Proceedings of the Fifth ACM International Conference on Digital Libraries, 2000
- [Allan 2002] J. Allan, Ed. Topic Detection and Tracking: Event-based Information Retrieval. Kluwer Academic Publishers, Norvell, MA, USA. 2002
- [Allan et al., 1998] Allan, J., Carbonell, J., Doddington, G., Yamron, J., and Yang, Y. (1998). Topic detection and tracking pilot study: Final report. In Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop, pages 194-218
- [Al-Kamha 2004] R. Al-Kamha and D. W. Embley. Grouping search-engine returned citations for person-name queries. In Proceedings of WIDM'04, Washington, DC, USA, pp. 96-103, 2004
- [Angel 1995] Angel S. Y. Tse, Kam-Fai Wong, & al. Effectiveness Analysis of Linguistics- and Corpus-based Noun Phrase Partial Parsers. In: Key-Sun Choi ed. Proceedings of Natural Language Processing Pacific Rim Symposium(NLPRS'95), Korea: Academic Press, 1995. 252-257
- [Aone 2000] C. Aone and M. Ramos-Santacruz. Rees: A large-scale relation and event extraction system. In Proceedings of the 6th Applied Natural Language Processing Conference, pages 76-83, 2000
- [Aron 2004] C. Aone and J. Sorensen. Dependency tree kernels for relation extraction. In Proceedings of ACL. 2004. Barcelona, Spain
- [AskJeeves] www.ask.com
- [Bagga 1998] A. Bagga and B. Baldwin. Entity-based cross-document co-referencing using the vector space model. In Proceedings of COLING-ACL'98, pp. 79-85, 1998
- [Baidu] www.baidu.com
- [Bekkerman 2005] R. Bekkerman and A. McCallum. Disambiguating web appearances of people in a social network. In Proceedings of WWW'05, pp. 463-470, 2005
- [Berkhin 2002] Berkhin, Pavel, Survey Of Clustering Data Mining Techniques ,2002, Technical Report of Accrue Software

- [Brants 2003] Brants, T., Chen, F. R., Farahat, A. O. A system for new event detection. In Proceedings of SIGIR 2003, the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2003 July 28-Aug 1; Toronto; Canada. NY: ACM; 2003; 330-337
- [Brin 1998] Sergey Brin, Lawrence Page. The Anatomy of a Large-Scale Hypertextual Web Search Engine. The Proceedings of the Seventh International World Wide Web Conference, Brisbane, Australia, April 14-18, 1998. Computer Networks 30(1-7): 107-117(1998)
- [Brin 1998] S. Brin. Extracting patterns and relations from the world wide web. In WebDB Workshop at 6th International Conference on Extending Database Technology, EDBT'98, 1998
- [Carthy 2002] J. Carthy. Lexical chains for topic tracking. PhD thesis, Department of Computer Science, National University of Dublin, 2002
- [Chai 1997] Joyce Yue Chai and Alan W. Biermann. The Use of Lexical Semantics in Information Extraction. In Proceedings of Workshop on Automatic IE and Building of Lexical Semantic Resources (ACL-97), 61-70. Madrid, July. 1997
- [Chai 1998] Joyce Yue Chai. Learning and Generalization in the Creation of Information Extraction Systems. Doctoral dissertation, Dept. of Computer Science, Graduate School of Duke University. 1998
- [Chen 1998] Chen H H, Ding Y W, Tsai S C, et al, Description of the NTU System Used for MET2, In Proceedings of the Seventh Message Understanding Conference, 1998
- [Chen 2002] Ying-Ju Chen, Hsin-Hsi. NLP and IR Approaches to Monolingual and Multilingual Link Detection. In Proceedings of the 19th International Conference on Computational Linguistics, Taipei, Taiwan, 2002
- [Church 1989] Church, K.W. and P. Hanks, 1989, Word Association Norms, Mutual Information, and Lexicography, Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics, ACL27'89, 76-83
- [Clarke 1999] R. Clarke. Person-location and person-tracking: Technologies, risks and policy implications. In 21st International Conference on Privacy and Personal Data Protection, pages 131--150, 1999
- [CNNIC 2004] 中国互联网络信息中心, 中国互联网络发展状况统计报告, 2005.1. <http://www.cnnic.net.cn/>
- [Connell 2004] Margaret Connell, Ao Feng, Giridhar Kumaran, Hema Raghavan, Chirag Shah,

- James Allan. UMass at TDT 2004. In The Seventh Topic Detection and Tracking Conference (TDT2004), December 2004
- [David 2003] David A. Smith. Detecting and browsing events in unstructured text. In Proceedings of the 25th Annual ACM SIGIR Conference, pages 73-80. 2003
- [Dejong 1982] Dejong G. An Overview of the FRUMP System. In: LEHNERT, W., & RINGLE, M.h. (eds), Strategies for Natural Language Processing. Lawrence Erlbaum, 1982, 149-176
- [Douglas 1999] Douglas E. Appelt and David J. Israel, Introduction to information extraction technology, A Tutorial Prepared for IJCAI-99
- [Duda 1973] R.O. Duda and P.E. Hart. Pattern Classification and Scene Analysis. John Wiley, New York, NY, 1973
- [Embley 1999] D.W.Embley, Y.S.Jiang, Y.-K. Ng. Record-Boundary Discovery in Web Documents, SIGMOD'99 Proceedings, 1999
- [Fisher 1986] D.H. Fisher and P.Langley. Methods of Conceptual Clustering and Their Relation to Numeric Taxonomy, in Artificial Intelligence and Statistics, W. Gale, ed., Addison Weseley, Ready, MA, 1986
- [Fleischman 2004] M. B. Fleischman and E. Hovy. Multi-document person name resolution. In Proceedings of the Workshop on Reference Resolution and its Applications: ACL'04, Barcelona, 2004
- [Franz 1999] Franz, M., McCarley, J.S., Roukos, S., Ward, T. and Zhu, W.-J. Segmentation and detection at IBM: Hybrid statistical models and two-tiered clustering. In Proceedings of the TDT 1999 Workshop. 2000
- [Fukumoto 2000] Fukumoto, F. and Suzuki, Y. Event tracking based on domain dependency. In Proceedings of ACM SIGIR, Research and Development in Information Retrieval, pages 57-64, 2000
- [Gaizauskas 1998] R. Gaizauskas, Y. Wilks. Information Extraction: Beyond Document Retrieval. Computational Linguistics and Chinese Language Processing, vol. 3, no. 2, pp. 17--60, August 1998
- [Ganti 1999] V. Ganti, J. Gehrke, R. Ramakrishnan. CACTUS-clustering categorical data using summaries. In: Proc of KDD'99, pp. 73-83, 1999
- [George 2003] George A. Miller, WordNet 2.0, <http://www.cogsci.princeton.edu/~wn/>, 2003
- [Gibson 1998] D. Gibson, J. Kleiberg, P. Raghavan. Clustering categorical data: an approach based on dynamic systems. In: Proc of VLDB'98, pp. 311-323, 1998

- [Google] www.google.com
- [Grefenstette 1992] Grefenstette, G., 1992, Use of Syntactic Context to Produce Term Association Lists for Text Retrieval, Proceedings of the 15th Annual International Conference on Research and Development in Information Retrieval, SIGIR'92
- [Grishman 1997] R.Grishman. Information extraction: Techniques and challenges. In MT Pazienza, editor, Information Extraction. Springer- Verlag, LNAI. 1997
- [Guha 2004] R. Guha and A. Garg. Disambiguating people in search. Stanford University, 2004
- [Guha 1998] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. CURE: An efficient clustering algorithm for large databases. In Proc. of 1998 ACM-SIGMOD Int. Conf. on Management of Data, 1998
- [Guha 1999] S. Guha, R. Rastogi, K. Shim. ROCK: a robust clustering algorithm for categorical attributes. In: Proc of ICDE'99, pp. 512-521, 1999
- [Han 2000] E. Han and G. Karypis. Centroid-Based Document Classification Analysis & Experimental Result. PKDD 2000
- [Han 2001] Jiawei Han, Micheline Kamber. Data Mining Concepts and Techniques, 2001. 范明、孟晓峰译. 《数据挖掘：概念与技术》. 2001, 机械工业出版社
- [Han 2002] CLARANS: a method for clustering objects for spatial data mining Ng, R.T.; Jiawei Han; Knowledge and Data Engineering, IEEE Transactions on , Volume: 14 Issue: 5 , Sept.-Oct. 2002 Page(s): 1003 –1016
- [Han 2004] H. Han, L. Giles, H. Y. Zha, et al. Two supervised learning approaches for name disambiguation in author citations. In Proceedings of JCDL' 04, Tucson, Arizona, USA, pp. 296-305, 2004
- [Hinneburg 1999] A. Hinneburg and D. A. Keim. Optimal Grid-Clustering: Towards Breaking the Curse of Dimensionality in High-Dimensional Clustering. In Proc. 25th Int. Conf. on Very Large Data Bases (VLDB'99), 1999, pp. 506–517
- [Hinneburg 2000] A. Hinneburg, C. C. Aggarwal, and D. A. Keim. What Is the Nearest Neighbor in High Dimensional Spaces? In Proc. 26th Int. Conf. on Very Large Data Bases (VLDB'00), 2000, pp. 506–515
- [Hotho 2002] A. Hotho and G. Stumme. Conceptual clustering of text clusters. In Proceedings of FGML Workshop, pages 37--45. Special Interest Group of German Informatics Society (FGML --- Fachgruppe Maschinelles Lernen der GI e.V.), 2002. <http://citeseer.ist.psu.edu/hotho02conceptual.html>

- [Hu 2005] Yunhua Hu, Guomao Xin, Ruihua Song, Guoping Hu, Shuming Shi, Yunbo Cao, and Hang Li, Title Extraction from Bodies of HTML Documents and Its Application to Web Page Retrieval, Proc. of ACM-SIGIR'05, 250-257
- [Huang 1997a] Z. Huang. A fast clustering algorithm to cluster very large categorical data sets in data mining. In: SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, 1997
- [Huang 1997b] Z. Huang. Clustering Large Data Sets with Mixed Numeric and Categorical Values. In Proc. First Pacific-Asia Conference on Knowledge Discovery and Data Mining, 1997
- [Huang 1998] Z. Huang. Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Mining and Knowledge Discovery, 1998, 2(3): 283-304
- [Huang 1999] Z. Huang, M. K. Ng. A fuzzy k-modes algorithm for clustering categorical data. IEEE Transaction on Fuzzy Systems, 1999, 7(4): 446-452
- [ICTCLAS] <http://mtgroup.ict.ac.cn/~zhp/ICTCLAS.htm>
- [Jain 1999] Jain, A.K., Murty M.N., and Flynn P.J. (1999): Data Clustering: A Review, ACM Computing Surveys, Vol 31, No. 3, 264-323.
<http://citeseer.ist.psu.edu/jain99data.html>
- [Jain 1988] A.K. Jain and R.C. Dubes. Algorithms for clustering data, Prentice Hall, Englewood Cliffs, 1988
- [Jiang 1997] Jay J. Jiang, David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy, proceeding of international conference research on computational linguistics (ROCLING X), 1997, Taiwan
- [Jollois 2002] F. Jollois, M. Nadif. Clustering large categorical data. In: Proc of PAKDD'02, pp. 257-263, 2002
- [Karypis 1999] G. Karypis, E.H. Han, and V. Kumar. Chameleon: A hierarchical clustering algorithm using dynamic modeling. Technical Report, Department of Computer Science, University of Minnesota, Minneapolis, 1999
- [Kim 2002] Kim, S., Alani, H., Hall, W., Lewis, P.H., Millard, D.E., Shadbolt, N., and Weal, M.J. Artequakt: Generating Tailored Biographies with Automatically Annotated Fragments from the Web. Workshop on Semantic Authoring, Annotation & Knowledge Markup, 15th European Conf. on Artificial Intelligence (ECAI), pages 1--6, Lyon, France, July 2002

- [Kim&Moldovan 1995] J. Kim and D. Moldovan. Acquisition of linguistic patterns for knowledge-based information extraction. *IEEE Transactions on Knowledge and Data Engineering*, 7(5):713-724, 1995
- [Lam 2000] Lam, W., Meng, H., and Hui, K., Multilingual Topic Detection using a Parallel Corpus, *Proceedings of the Topic Detection and Tracking (3) Workshop 2000*, Washington DC., Nov. 2000
- [LDC] www ldc upenn edu
- [Li 2002] Sujian Li, Jian Zhang, Xiong Huang and Shuo Bai. Semantic Computation in Chinese Question-Answering System. 2002, *Journal of Computer Science and Technology*
- [Lim 2000] Seung-Jin Lim, Yiu-Kai Ng. A Heuristic Approach for Converting HTML Documents to XML Documents. *Proceedings of the Sixth International Conference on Rules and Objects in Databases (DOOD 2000)*, pp. 1182-1196, London, England, July 2000
- [Lin 2002] Shian-Hua Lin, Jan-Ming Ho. Discovering Informative Content Blocks from Web Documents. *KDD 2002*:588-593
- [Liu 2002] Qun Liu, Sujian Li, Lexical semantic similarity computation based on HowNet, *Computational Lin-guistics and Chinese Language Processing*, Vol.7, No.2, August 2002, pp.59-76
- [Luger 1997] G. Luger and W. Stubblefield. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. Addison Wesley Longman, Reading, Massachusetts, USA, September 1997
- [Lyman 2003] Peter Lyman, Hal R. Varian, et. How Much Information? 2003. <http://www.sims.berkeley.edu/research/projects/how-much-info-2003/index.htm>, October, 2003
- [Makkonen 2004] Juha Makkonen, Helena Ahonen-Myka, and Marko Salmenkivi. Simple Semantics in Topic Detection and Tracking. *Information Retrieval*, 7 (3-4): 347--368, 2004
- [Mani 2000] Mani I. and Wilson G., 2000. Robust temporal processing of news. In: *Proceedings of the 38th Annual Meeting of the ACL*
- [Mann 2003] G. S. Mann and D. Yarowsky. Unsupervised personal name disambiguation, In *Proc. of CoNLL'03*, Edmonton, Canada, 2003
- [Matsuo 2004] Yutaka Matsuo, Hironori Tomobe, Kôiti Hasida, Mitsuru Ishizuka. Finding

- Social Network for Trust Calculation. ECAI 2004: 510-514
- [McCallum 2000] A. McCallum, K. Nigam, and L. H. Ungar. Efficient cing of high-dimensional datasets with application to reference matching. In Knowledge Discovery and Data Mining, pp. 169–178, 2000
- [MUC-6 1995] Proceedings of the Sixth Message Understanding Conference (MUC-6). Morgan Kaufmann Publishers. 1995
- [Muslea 1999] Ion Muslea. Extraction Patterns for Information Extraction Tasks: A Survey. 1999. <http://www.aaai.org>
- [Ng 2002] M. K. Ng, J. C. Wong. Clustering categorical data sets using tabu search techniques. Pattern Recognition, 2002, 35(12): 2783-2790
- [NIST 2004] NIST. The 2004 topic detection and tracking (TDT-2004) task definition and evaluation plan. Technical Report Version 1.2, National Institute of Standards and Technology, 2004
- [Niu 2004] C. Niu, W. Li and R. K. Srihari. Weakly supervised learning for cross-document person name disambiguation supported by information extraction. In Proceedings of ACL'2004
- [Pani 2001] Pani AK, Bhattacharjee GP. Temporal representation and reasoning in artificial intelligence: A review. Mathematical and Computer Modelling, 2001,34(1-2):55~80
- [Papka 1999] Papka, R., Allan, J., and Lavrenko, V. 1999. UMASS approaches to detection and tracking at TDT2. In Proceedings of the TDT-99 workshop. NIST
- [Papka 1999] R. Papka. On-line new event detection, clustering and tracking. PhD Thesis, Department of Computer Science, University of Massachusetts, 1999
- [Pantel 2002] Patrick Pantel and Dekang Lin. Document clustering with committees. In Proc. of SIGIR'02, Tampere, Finland, 2002
- [Ravin 1999] Y. Ravin and Z. Kazi. Is Hillary Rodham Clinton the President? Disambiguating names across documents. Proceings of the ACL '99 Workshop on Coreference and its Appli-cations, pp. 9-16, 1999
- [Riloff 1993] Ellen Riloff. Automatically constructing a dictionary for information extraction tasks. Proc. Eleventh National Conf. on Artificial Intelligence, 811-816
- [Riloff 1994] E. Riloff, W. Lehnert, Information Extraction as a Basis for High-Precision Text Classification, in ACM Transactions on Information Systems, Vol. 12, No 3, July 1994, pp. 296--333
- [Riloff 1998] Ellen Riloff, Information Extraction as a Stepping Stone toward Story

- Understanding, In *Understanding Language Understanding: Computational Models of Reading*, edited by Ashwin Ram and Kenneth Moorman, forthcoming from MIT Press
- [Riloff 1999] E. Riloff and R. Jones, Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping. Proc. Sixteenth National Conf. on Artificial Intelligence (AAAI-99), Orlando, FL, 1999
- [Sager 1981] Sager N, Natural Language Information Processing, Reading, Massachusetts: Addison Wesley, 1981
- [Saliha 1998] Azzam Saliha, Kevin Humphreys & Robert Gaizauskas. Coreference resolution in a multilingual information extraction. In the Proc. of the Workshop on Linguistic Coference. Granada, Spain.1998
- [Salton 1975] Gerard Salton, A. Wong, C.S Yang. A Vector Space Model for Automatic Indexing. Communications of the ACM, 1975,18(11)
- [Schiffman 2001] Barry Schiffman, Inderjeet Mani, Kristian J. Concepcion. Producing biographical summaries: Combining linguistic knowledge with corpus statistics. In Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics (ACL-EACL 2001), Toulouse, France, July 2001
- [Schilder 2001] Schilder F. and Habel C., 2001. From Temporal Expressions to Temporal Information: Semantic Taggings of News Messages. In: Proceedings of the ACL'01 WS on temporal and spatial information processing
- [Shai 1998] Shai Fine, Yoram Singer, and Naftali Tishby. The hierarchical Hidden Markov Model: Analysis and applications. Machine Learning, 32:41, 1998
- [Shankar 2000] Shrikanth Shankar and George Karypis. Weight adjustment schemes for a centroid-based classifier. TextMining Workshop, KDD, 2000
- [Shipeng 2003] Shipeng Yu, Deng Cai, Ji-Rong Wen, Wei-Ying Ma. Improving Pseudo-Relevance Feedback in Web Information Retrieval Using Web Page Segmentation. WWW 2003, 11-18
- [Sogou] www.sogou.com
- [Spink 2004] Spink, A., Jansen, B. J., & Pedersen, J. (2004). Searching for people on Web search engines. Journal of Documentation, 60(3), 266-278
- [Spitters 2001] Martijn Spitters and Wessel Kraaij. TNO at TDT2001: Language Model-Based Topic Detection. In Topic Detection and Tracking (TDT) Workshop 2001, November 2001. NIST

- [Swan 2000] Swan, R. and Allan, J. Automatic generation of overview timelines. In Proceedings of ACM SIGIR, Research and Development in Information Retrieval, pages 49-56, 2000
- [Taffet 2004] M. D. Taffet. Person resolution: resolving multireferent and multimorphic person names within and across full text documents. Dissertation Proposal, Syracuse University, 2004
- [Trieschnigg 2004] Dolf Trieschnigg, Wessel Kraaij, TNO Hierarchical topic detection report at TDT 2004. In The Seventh Topic Detection and Tracking Conference (TDT2004), December 2004
- [Ventura 1995] D. Ventura and T. R. Martinez. An Empirical Comparison of Discretization Methods. In Proc. Tenth Int. Symp. on Computer and Information Sciences, 1995, pp. 147–176
- [Wacholder 1997] N. Wacholder, Y. Ravin, and M. Choi. Disambiguation of proper names in text. In Proceedings of ANLP'1997
- [Wayne 2000] Wayne C., Multilingual Topic Detection and Tracking: Successful Research Enabled by Corpora and Evaluation, Language Resources and Evaluation Conference (LREC), pages 1487-1494. 2000
- [Wen 2003] Ji-Rong Wen, Ruihua Song, Deng Cai, Kaihua Zhu, Shipeng Yu, Shaozhi Ye and Wei-Ying Ma. Microsoft Research Asia at The Web Track of TREC 2003. The Twelfth Text Retrieval Conference (TREC'12), Nov. 2003
- [Whitelaw 2005] Casey Whitelaw, Navendu Garg, Shlomo Argamon. Using appraisal groups for sentiment analysis. CIKM 2005: 625-631
- [W3C] www.w3.org/Library/
- [Xiaojun 2005] Xiaojun Wan, Jianfeng Gao, et. Person resolution in person search results: WebHawk. ACM Fourteenth Conference on Information and Knowledge Management (CIKM2005)
- [Yahoo!] www.yahoo.com
- [Yang 1998] Y. Yang, T. Pierce, and J. Carbonell, A Study on Retrospective and Online Event Detection. In Proceedings of the 21th Ann Int ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98), pages 28-36, 1998
- [Yang 1999] Yiming Yang, Jaime Carbonell, Ralf Brown, Thomas Pierce, Brian T. Archibald, Xin Liu. Learning Approaches for Detecting and Tracking News Events, IEEE Intelligent Systems: Special Issue on Applications of Intelligent Information

- Retrieval, Vol. 14(4), pp 32-43, July/August 1999
- [Yang 2002] Y. Yang, J. Carbonell, and C. Jin. Topic-conditioned novelty detection. In Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM Press, pp.688-693. 2002
- [Yangarber 2000] Roman Yangarber, Ralph Grishman, Pasi Tapanainen, Silja Huttunen. Automatic Acquisition of Domain Knowledge for Information Extraction. In Proceedings of the 18th International Conference on Computational Linguistics(COLING 2000),Saarbrücken, Germany, August 2000
- [Yangarber 2001] Roman Yangarber. Scenario Customization for Information Extraction. Ph.D. Dissertation. New York University. January, 2001
- [Yu 1998] Yu S H, Bai S H, Wu P, Description of the Kent Ridge Digital Labs System Used for MUC-7, In Proceedings of the Seventh Message Understanding Conference, 1998
- [Zelenko 2003] D. Zelenko, C. Aone, and A. Richardella. Kernel methods for relation extraction. J. Mach. Learn. Res., 3:1083-1106,2003
- [Zhang 1996] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. BIRCH: An Efficient Data Clustering Method for Very Large Databases. In Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data, pages 103--114, Montreal, Canada, 1996. <http://citeseer.ist.psu.edu/zhang96birch.html>
- [Zhang 2000] Zhang Y M, Zhou J F, A Trainable Method for Extracting Chinese Entity Names and Their Relations, In Proceedings of the Second Chinese Language Processing Workshop, Hong Kong, Oct. 2000
- [Zhang 2004] Zhu Zhang. Weakly-Supervised Relation Classification for Information Extraction. ACM Thirteenth Conference on Information and Knowledge Management (CIKM). 2004
- [Zhang 2005] Hua-Ping ZHANG, Jian Sun, Bin WANG, Shuo BAI. Computation on Sentence Semantic Distance for Novelty Detection; Chinese Journal of Computer Science and Tech. vol.3, 2005
- [Zhongsou] www.zhongsou.com
- [Zoominfo] www.zoominfo.com
- [白硕 2005] 白硕. 文本挖掘的若干新进展. 第二届全国信息检索与内容安全学术会议主题报告. 北京, 2005.10
- [卜东波 2002] 卜东波,白硕,李国杰. 文本聚类中权重计算的对偶性策略. 软件学报

2002,13(11):2083-2089

- [陈宁 2001] 陈宁. 数据挖掘中的聚类算法研究. 中科院数学与系统科学研究院博士论文,2001
- [贾自艳 2004] 贾自艳. Web 信息智能获取若干关键问题研究. 中科院计算所博士论文. 2004
- [姜吉发 2004] 姜吉发. 自由文本的信息抽取模式获取的研究. 中科院计算所博士论文. 2004
- [李晓黎 2001] 李晓黎, WEB 信息检索与分类中的数据采掘研究, 中科院计算所博士学位论文, 2001
- [李向农 1997] 李向农. 现代汉语时点时段研究. 武汉: 华中师范大学出版社, 1997
- [李保利 2003] 李保利, 陈玉忠, 俞士汶. 信息抽取研究综述. 计算机工程与应用. 第 39 卷第 10 期第 1-5 页. 2003
- [李振星 2003] 李振星, 徐泽平, 唐卫清等. 网页多词元快速聚类算法, 计算机工程, 2003 Vol.29 No.2
- [李文捷 1995] 李文捷, 周明等. 基于语料库的中文最长名词短语的自动提取, 陈力为、袁琦主编, 计算语言学进展与应用, 北京: 清华大学出版社, 1995. 119-124.
- [陆俭明 1991] 陆俭明. 现代汉语时间词说略. 语言教学与研究. 1991.P.25-37
- [马红妹 2002] 马红妹, 王挺, 陈火旺. 汉语篇章时间短语的分析与时制验算. 计算机研究与发展. Vol.39, No.10, 2002
- [史中琦 2004] 史中琦, 张普. 基于 DCC 动态流通语料库的流行语类型分析. 北京语言大学. 中文语言文学网 www.china-language.gov.cn/webinfopub/list.asp?id=1351&columnid=163&columnlayer=01380163
- [王厚峰 2001] 王厚峰, 何婷婷. 汉语中人称代词消解的研究. 计算机学报, 2001, 24(2): 136-143
- [王强军 2003] 王强军. 基于动态流通语料库(DCC)的信息技术领域新术语自动提取研究. 北京语言大学博士论文. 2003
- [隋岩 2004] 隋岩. 基于“动态流通语料库”的“有效字符串”提取研究. 北京语言大学博士论文. 2003
- [许洪波 2003] 许洪波, 大规模信息过滤技术研究及其在 Web 问答系统中的应用, 博士学位论文, 中国科学院计算技术研究所, 2003 年 7 月
- [阎大为 2004] 阎大为. 黄蓉比郭靖年龄大. www.greenworm.com/kongfu/kongfu172.htm
- [赵军 1998] 赵军. 汉语基本名词短语识别及结构分析. 清华大学计算机系博士论文.

1998

- [张华平 2004] 张华平, 刘群, 基于角色标注的中国人名自动识别研究, 计算机学报, Vol.27, No.1, 2004.1
- [张华平 2004] 张华平, 语言浅层分析与句子级新信息检测研究, 中科院计算所博士学位论文, 2004
- [张普 1999] 张普. 关于语感和流通度的思考. 语言教学与研究. 1999 年第 2 期
- [张普 2000] 张普. 信息处理用动态语言知识更新的总体思考. 语言文字应用. 2000.2 期
- [张威 2002] 张威, 周昌乐. 汉语语篇机器理解中元指代消解初步. 软件学报, 2002, 4
- [张卫国 1996] 张卫国. 三种定语、三类意义和三个槽位. 中国人民大学学报, 1996.4
- [张尧庭 1982] 张尧庭等, 多元统计分析引论 科学出版社, 1982
- [周强 2000] 周强, 孙茂松, 黄昌宁. 汉语最长名词短语的自动识别. 软件学报, 2000, 11(2):195-201
- [朱德熙 1982] 朱德熙. 语法讲义. 北京: 商务印刷馆. 2002
- [朱明 2000] 朱明. 互联网信息智能搜索与获取方法研究, 中国科学技术大学博士学位论文, 2000

致 谢

2006 年的年初,在这辞旧迎新之际,我终于可以舒一口气,来结束我的博士论文了。五年多的硕博连读生活,在师长、亲友的多方大力支持下,我学习到了无比宝贵的知识财富。

在本课题研究结束之际,首先向自始至终给予我悉心指导,谆谆教诲的白硕研究员表示衷心感谢。白老师严谨的治学态度、渊博的专业知识、敏锐的学术洞察力,在学业上的悉心指点,对我的论文从开始选题,到全文的终结起到了至关重要的作用,论文的字里行间无不浸透了导师的心血。白老师在学术上带给我启迪,拓宽了我的思路,引导了我的学术思维,赋予了我自由、宽松的学习和研究环境,同时在生活上给予了我关切和爱护,才得以使论文顺利完成。可以说,正是白老师的言传身教,我才可能取得一点成绩。他不仅仅是我的博士论文指导老师,更是我终生的榜样。

我的论文工作是在中国科学院计算技术研究所的软件室进行的。在此我要衷心地感谢程学旗研究员,郭莉副研究员和王斌副研究员,是他们指引我迈入信息处理的大门,使我在信息处理方向有了现在的积累。三位老师悉心地关怀帮助、严谨求实的科学作风和平易近人的为人给我留下了深刻的印象。由衷地感谢各位老师对我的指导与帮助。

感谢软件室的同事张刚博士、郭岩博士、吴丽辉博士、孙健博士、刘悦博士、谭建龙博士、张凯、骆为华、刘萍以及我的师兄同学王映博士、俞晓明博士、谭松波博士、丁国栋博士、张华平博士、王树西博士、龚才春博士、潘文峰、杨哲、胡吉祥、李魁等,感谢他们在课题研究中给了我大量的启发与帮助。

尤其需要提到的是,课题研究过程中,我的课题小组长,软件室的许洪波博士对我的帮助很大。他思维敏锐,思路清晰,见解独到。每次与他探讨问题,我都会受到很大的启发。他刻苦钻研、锲而不舍的精神是我学习的榜样。不仅在学业上给予了我无私的指导,而且还在学习生活等诸多方面给予我关怀和指点。

感谢研究生处的宋老师、张老师、李老师、周老师、靳老师对我的帮助和教诲,以及软件室秘书处的宋姻姐姐和孙萍姐姐。他们待人总是那么亲切、平易近人,每每遇到生活上的困难,总能够跟他们进行深入的沟通和交流而化解心中的疑惑。

我要感谢给予我无限爱心的家人,特别是我的父母和姐姐。他们对我学业上的鼓励、经济上的支持和生活上的关怀使我有勇气和信心走过二十年的漫漫求学生涯。我从父亲身上学到的最多的就是做事的独立自主和坚持不懈,而母亲给予我的最可贵的品质就是诚实善良和勤劳简朴。多年来,父亲一直是家庭经济的顶梁柱,母亲则为我们营造了温馨的生活氛围。而当父亲几年前不幸患绝症突然去世后,我的母亲和姐姐毅然经受住了精神和经济上的双重打击,支撑起家庭的经济情况,使我安安心心地度过了最后几年的学习生涯。他们是我求学的坚强后盾,对我的亲情和付出是无私的,他们的殷切期望激

励着我奋发向上，希望我的些许成绩能让他们感觉到欣慰和快乐。

最后，我要感谢知书达理、对我真心相爱的女朋友赵庆芝女士。她对我的真情和关爱鼓励着我朝着理想的彼岸而前进，而她为我提供了心灵歇息的宁静港湾。我觉得这是前世的姻缘使我能够与她相识，是上天对我的恩赐。在我读博士期间，她为我付出了很多，我愿意把这篇文献给她！

最后，我要向所有培养、关心和帮助我的老师、亲人和朋友们表示衷心的感谢！

作者简介

姓名：于满泉 性别：男 出生日期：1978.10.20 籍贯：辽宁抚顺

2001.9—2006.6 中国科学院计算技术研究所软件研究室硕博连读生

1997.9—2001.7 南京理工大学计算机科学与工程系本科生

【攻读博士学位期间发表的论文】

- [1] Manquan Yu, Weihua Luo, Zhaotao Zhou, Shuo Bai. ICT's approaches to HTD and Tracking at TDT2004. In Proc. of the 7th Topic Detection and Tracking Conference (TDT2004), December 2004, Gaithersburg, Maryland, USA
- [2] Manquan Yu, Weihua Luo, Bin Wang, Shuo Bai. Multi-layered Clustering for Hierarchical Topic Detection. In the 2nd Indian International Conference on Artificial Intelligence (IICAI-05), December 2005, Pune, India
- [3] 于满泉, 骆卫华, 许洪波, 白硕. 话题识别与跟踪中的层次化话题识别技术研究, 计算机研究与发展, Vol.3, 2006
- [4] 于满泉, 许洪波, 赵章界, 余智华. 基于内容的个性化信息过滤系统的研究与设计, 计算机工程, Vol.31, No.12, 2005, pp.78-80
- [5] 于满泉, 陈铁睿, 许洪波. 基于分块的网页信息解析器的研究与设计, 计算机应用, Vol.25, No.4, 2005, P.974-976
- [6] 于满泉, 谭松波, 许洪波. 网页内部结构挖掘技术研究, 第一届全国信息检索与内容安全学术会议(NCIRCS), 2004.11, 上海
- [7] 王映, 于满泉, 李盛韬. JavaScript 引擎在动态网页采集技术中的应用, 计算机应用, Vol.24, No.2, 2004, P.33-36
- [8] 李盛韬, 吴丽辉, 于满泉, 潘文锋, 余智华, 王斌, 程学旗. 主题 Web 信息采集的研究与设计, 第七届全国计算语言学联合学术会议(JSCL2003), 2003.8, 哈尔滨
- [9] 潘文锋, 王斌, 于满泉, 谭松波. Winnow 算法在垃圾邮件过滤中的应用, 第一届全国信息检索与内容安全学术会议(NCIRCS), 2004.11, 上海
- [10] 郭岩, 白硕, 于满泉. Web 使用信息挖掘综述, 计算机科学, Vol.32, No.1, 2005, P.1-7
- [11] 吕铁强, 于满泉, 孟庆发. 基于网页分块的个性化信息采集的研究与设计. Vol.22, No.10, 2005

【攻读博士学位期间参加的科研项目】

- [1] 参加国家 973 重点基础研究项目 (编号 2004CB318109), 独立负责“互联网人物信息搜索引擎”的研究, 申请发明专利一项, 成功实现原型系统. 2004.8~2005.12
- [2] 国家安全某海量报文分析项目。参与了整体设计, 主要负责聚类子系统的实现. 2005.5~2005.11
- [3] 参加第 7 届“话题识别与跟踪”(Topic Detection and Tracking, TDT)国际评测。负责“层次化话题识别”任务, 在四家参赛队伍中取得第二名. 2004.6~2004.12
- [4] 参加“国家信息安全保障持续发展计划”支持下的“基于 Internet 的海量信息处理系统”, 负责信息过滤子系统的开发和后期维护, 重点开发了网页解析模块, 制作系统安装包, 已经验收. 2003.11~2004.5
- [5] 参加“国家信息安全保障持续发展计划”支持下的“特定信息发现系统”, 负责开发多媒体文件的采集模块, 已经验收. 2003.6~2003.10
- [6] 参加中科院计算所软件室“天罗 Web 采集系统”的三期开发和升级, 负责开发动态链接解析模块和修改采集器 bug. 2002.9~2003.8

【攻读博士学位期间的获奖情况】

- [1] 2005 年获得“中科院计算所与 Sony 公司联合奖学金”的论文优秀奖
- [2] 2004 年获得第 7 届“话题识别与跟踪”(TDT)国际评测中“层次化话题识别”任务第二名
- [3] 2003 年获得中科院计算所软件研究室“优秀学生奖”