摘要

互联网的迅速发展导致 Web 信息飞速增长，Web 已经成为世界上最大的信息来源。由于 Web 资源的迅速膨胀以及 Web 信息的分散性与异构性，导致知识的难以查询。目前，互联网已经发展成为拥有亿页面的分布式信息空间，而在这些异质的亿页面的资源中，蕴含着大量的人们迫切需要的知识，如何对这些庞大的 Web 信息进行知识提取，成为近年来的一个研究方向，并产生了新的研究领域，即 Web 数据挖掘。Web 文本挖掘是 Web 数据挖掘的一个研究分支， Web 文本挖掘可以提高人们获取 Web 信息的效率，对 Web 资源进行分门别类，帮助用户快速的对目标知识进行定位，并且能够从中抽取有价值的知识。

本文研究并利用 Web 文本挖掘技术对 Web 信息进行挖掘，实现对 Web 文本的分类，详细设计并实现了具备完整功能结构的 Web 文本挖掘原型系统，重点讨论了当前流行的 Web 文本挖掘的关键技术，主要的研究工作包括：

1. 介绍 Web 文本挖掘的背景知识，分析 Web 文本挖掘的研究背景和现状，并探讨了 Web 文本挖掘的意义；

2. 详细分析和讨论了 Web 文本挖掘过程的关键技术，包括中文分词技术、权重计算方法、Web 文本特征的表示和提取方法；

3. 讨论了几种常用的 Web 文本分类和聚类的算法：KNN 近邻算法、朴素贝叶斯方法、SOM 自组织映射算法、K 均值算法，研究了 K 均值算法和遗传算法的理论和优缺点，在此基础上，提出基于 K 均值和遗传算法的聚类算法，并对其进行了实验，在挖掘评价方法的基础上，验证该算法的可行性。

基于以上的研究成果，本文描述了 Web 文本挖掘原型系统的设计和实现细节。

关键词： 数据挖掘 Web 文本挖掘 中文分词 特征提取

Abstract

With the fast development of the internet leading to the rapid growth of web information, web has been the largest of information source of all over the world. It is difficult to inquire out knowledge because of the fast expansive web resource as well as the separation and the heterogeneity of the web information. Now internet has been a distributed information space with billions of web pages, from which conceals a large number of knowledge that people urgently needed. It is one of research direction that how to extract knowledge from plentiful web information, bring a new research field named web data mining. Web text mining is one branch of web data mining. Web text mining can improve efficiency people obtain web information, divide web text into different classifications, help people to locate knowledge they want quickly and extract worthy knowledge from web resource.

This paper research and use web text mining technology to dig out web information, realize classifications to web text, design and achieve web text mining prototype system with total function and structure in detail, discuss current popular key technologies to web text mining primarily. The paper's chief research contains:

1. Introduce background knowledge of web text mining, analyze research backdrop and actuality of web text mining and discuss its significance.

2. Analyze and discuss the key technologies of web text mining process, which contains Chinese word segmentation, weight calculation method, feature representation and extraction approach to web text.

3. Discuss some common web text classification and clustering algorithms: KNN nearest neighbor algorithm, naive bayes method, self organizing maps algorithm and K-means algorithm, research the advantages and disadvantages of K-means algorithm and genetic algorithm. Based on the research, this paper puts forward a clustering algorithm named clustering based on K-means and genetic algorithm, carries out the experiment to verify the feasibility of the algorithm foundation on mining evaluation methods.

The paper describes design and realization details of web text mining prototype system based on research.

Keyword: Data Mining Web Text Mining Chinese Word Segmentation Feature Selection

目录

第一章 绪论

1.1 课题背景

本课题是国家科技支撑计划项目——社区电子服务关键技术研究及应用示范工程的一个组成部分。课题的主要研究任务是 Web 文本挖掘关键技术研究与实现。

随着 Internet 的飞速发展，庞大的 Web 资源已经成为人们获得知识与信息的来源[1]。Internet 上存储着各种文字、音频和视频等信息资源，这些资源分布在数以亿计的 Web 站点中。与传统的信息资源相比，Web 资源具有开放性、动态性和异构性等特点，这些特点导致了 Web 资源的迅速增长，Web 信息日趋海量化。Web 在给人们提供丰富的信息的同时，也带来了新的挑战，人们很难通过易用的方式快速准确地从海量信息中获取所需的信息资源[2]，如何快速有效地对 Web 信息进行分类和提取有用的信息成为一项重要的研究课题[3]。正是在这一背景下，产生了新的研究领域，即 Web 数据挖掘。

Web 数据挖掘是利用数据挖掘技术从 Web 文本中提取人们感兴趣的、隐藏的、有用的知识模式，是数据挖掘领域一个新的研究方向。Web 数据挖掘可以协助搜索引擎抓取高相关度的网页、分析 Web 文本结构等，使 Web 服务更加智能化。Web 数据挖掘的研究方向主要有三个：Web 内容挖掘、Web 结构挖掘和 Web 使用挖掘。Web 文本挖掘(Text Mining)属于 Web 内容挖掘的范畴，用于 Web 文本信息的知识发现[4]，通过统计理论和人工智能理论，如神经网络、支持向量机、可能性推理等，并结合文字处理技术，分析大量的非结构化文本源(如网页、文档、客户电子邮件、问题查询等)，抽取或标记关键字概念、文字间的关系，并按照内容对文档进行分类，获取有用的知识和信息[5]。文本挖掘通过分析大量非结构化文本，发现有效的、新颖的、有用的、可理解的信息，提高人们获取信息资源的效率和 Web 文档的使用价值。

1.2 研究现状

Web 数据挖掘日益得到众多研究者的重视，他们利用各种统计理论和智能理论研究 Web 数据挖掘，提出很多新的 Web 数据挖掘技术。目前，国际上对 Web 数据挖掘研究主要集中在自动文本分类技术、主题搜索引擎的设计、关键词的自动获取、文本特征提取、半结构化信息的提取以及 Web 上新型应用等领域。

国外对 Web 文本挖掘的研究开展较早，早期的信息抽取技术就是文本挖掘的

雏形。50 年代末，H.P Luhn 对文本挖掘技术进行了开创性的研究，将词频统计的思想用于文本自动分类。随后，众多学者在这一领域进行了卓有成效的研究工作[6]。到目前为止，在文本挖掘的关键词的自动获取、文本挖掘技术和半结构化信息获取等相关领域已经取得令人瞩目的研究成果。文本挖掘已经从最初的可行性基础研究经历了试验性研究进入到了实用化阶段，并在邮件分类、电子会议、信息过滤等方面得到了较为广泛的应用[7]。文本挖掘技术也已经进入了商业应用领域，例如 IBM 的文本智能挖掘机、Autonomy 公司的核心产品 Concept Agents 和 Megaputer 的 TextAnalyst 等。文本挖掘在军事、企业竞争情报领域也得到了大量的应用，很多政府机构、企业都开始利用文本挖掘软件对竞争伙伴的情报数据进行评估和分析。

相对于国外，国内对文本挖掘的研究起步较晚。1981 年，候汉清教授对于计算机在文本分类工作中的应用做了探讨[8]，并介绍了国外计算机管理分类表、计算机分类检索等方面的情况。1998 年，我国国家重点基础研究发展规划首批实施项目中，将文本挖掘的研究列为“图像、语音、自然语言理解与知识挖掘”中的重要内容。国内对文本挖掘技术的研究机构主要集中在高等院校、科研院所和信息公司，并且也取得了不错的成果，例如：

(1) 中科院计算机语言信息工程中心所研究的汉语分词、自然语言接口、句法分析、语义分析、音字转换等；

(2) 清华大学电子工程系研究的手写汉字识别、汉字识别多分类器集成；

(3) 上海交通大学计算机系研究的语句语义、自然语言模型、构造解释模型、范例推理等；

(4) 哈尔滨工业大学计算机系研究的自动文摘、手写汉字识别、自动分词等；

(5) 东北大学的词性标注、中文信息自动抽取、汉语文本自动分类模型等。

但是，国内在文本挖掘的研究，特别是文本挖掘工具的商业化应用方面仍明显落后于国外。因此，如何尽快提高我国的文本挖掘的研究水平以及应用能力，是计算机科学领域迫切需要研究的重要课题之一。

1.3 研究内容

本文研究的内容是 Web 文本挖掘关键技术的研究和实现，主要对 Web 文本挖掘的理论方法、技术和应用进行讨论和研究，包括中文分词技术、文本特征向量的提取、Web 文本的分类算法和聚类算法等。并在研究的基础上，整合各种技术，实现 Web 文本挖掘的原型系统。重点研究了以下几种 Web 文本挖掘的关键技术：

中文分词技术：介绍了中文语言的特点，分析了中文的几种分词方法，包括基于字符串匹配的方法、基于统计的分词方法和基于理解的方法，重点描述了正

向和逆向最大匹配方法；

特征向量的表示和提取：介绍了用于文本表示的向量空间模型和权值计算法，向量空间模型的特征维数是很高的，需要对其进行降维，本文分析了当前几种提取方法，包括χ^2统计量、互信息方法和信息增益方法等。

Web 文本聚类和分类：重点介绍了当前流行的几种聚类和分类方法，包括 KNN 算法、朴素贝叶斯方法、SOM 自组织映射方法、K 均值算法，并在分析 K 均值和遗传算法的优缺点的基础上，将 K 均值的局部搜索特点和遗传算法的全局搜索能力相结合，给出基于 K 均值和遗传算法的聚类算法。实验表明，该算法对文本聚类的准确性有所提高。

1.4 论文的组织结构

本文的组织结构如下：

第二章是该论文研究领域的基础性介绍。介绍了数据挖掘和 Web 数据挖掘的基础知识和研究进展，以及基于 Web 的文本挖掘的挖掘过程。

第三章分析了 Web 文件挖掘相关技术和理论，详细叙述了中文分词技术、特征表示相关知识、Web 文本分类和聚类算法，为实现 Web 文本挖掘原型系统提供理论支持。

第四章在分析 K 均值和遗传算法的基础上，给出基于 K 均值和遗传算法的聚类算法，并通过实验验证了可行性。

第五章详尽地描述了 Web 文本挖掘原型系统的设计和实现细节，通过大量的图例，展示了相关算法的流程和系统的模块划分。

第六章描述了本文研究工作的总结，并对 Web 文本挖掘相关知识进行了合理的预测和展望。

第二章 Web 文本挖掘

Web 数据挖掘是数据挖掘技术在 Web 上的应用，是数据挖掘领域一个新的重要研究方向。作为一门新的交叉学科，Web 数据挖掘涉及到机器学习、自然语言理解、人工智能、模式识别、概率统计等学科。随着因特网的迅速发展，针对海量的 Web 资源的数据挖掘技术成为研究的热点。

2.1 数据挖掘

2.1.1 概述

随着人们认识和管理水平的提高，对客观世界的描述愈来愈全面，数据量呈现出爆炸性的增长，然而，数据库中数据开发的应用层面上主要仍是检索查询，应用范围很小。此外，相当数量的数据具有很强的时效性，数据本身的价值随着时间的推移而迅速降低。简单的数据查询或统计应用虽然可以满足某些低层次的需要，但人们更为需要的是从大量数据资源中挖掘出对各类决策有指导意义的一般信息，这些信息是对大量数据的高度概括和抽象。许多有用的信息隐含在数据中而不能被发现和利用，造成数据资源的浪费。快速增长的海量数据收集存放在若干大型数据库中，如果没有强有力的工具来帮助，其结果是重要的决策不是基于数据库中丰富的信息，而是基于决策者的直觉。决策者迫切需要采用自动化程度更高、效率更好的数据处理方法来提高数据分析的效率，自动发现数据中隐藏的规律或模式，为决策提供支持。数据挖掘(DM)技术正是为满足上述要求而产生的。

数据挖掘是人工智能、机器学习与数据库技术相结合的产物，受数据库系统、统计学、机器学习、可视化和信息科学等多个学科影响，属于交叉学科领域。数据挖掘作为知识发现过程的一个特定步骤，是对数据及数据间关系进行考察和建模的方法集，并从大量数据中提取人们感兴趣的、隐含的、潜在有用的信息和知识，表示为概念(Concepts)、规则(Rules)、规律 (Regularities)、模式(Patterns)等形式。

1989年8月于美国底特律市召开的第十一届国际联合人工智能学术会议正式形成数据挖掘[9,10]的概念：数据挖掘是指从数据库的大量数据中揭示出隐含的、先前未知的、潜在有用的信息的非平凡过程。许多人把数据挖掘视为另一个常用的术语：数据库中的知识发现或KDD的同义词[11]。KDD是在1989年举行的第十一届国际联合人工智能学术会议上首先出现的[12]。

数据库中知识发现的过程一般由下列步骤组成：

1. 数据清理：主要是消除孤立点或不一致数据。

2. 数据集成：用于将不同数据库中的数据组合在一起。

3. 数据选择：从数据库中检索与分析任务相关的数据。

4. 数据变换：将数据变换成统一的适合挖掘的形式。

5. 数据挖掘：利用各种挖掘方法提取数据模式。

6. 模式评价：根据某种评价方法度量、识别表示知识的真正有趣的模式。

7. 知识表示：使用可视化和知识表示的技术，向用户提供挖掘的模式知识。

广义的数据挖掘不限于对数据库的数据进行挖掘，可以从存放在数据仓库或者其它信息库(文本数据库、多媒体数据库、WWW等)中的数据中挖掘有趣知识的过程。在这种观点下，数据挖掘系统具有以下主要成分：数据库、数据仓库或其他信息库、知识库、数据挖掘引擎、模式评价模块和用户图形界面。

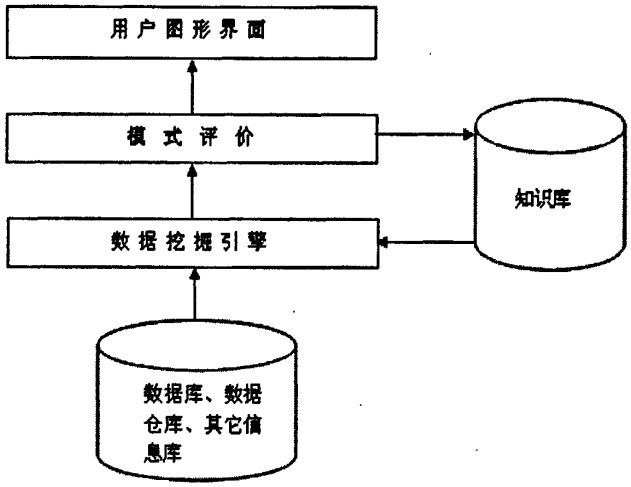

图 2.1 数据挖掘系统的主要成分

2.1.2 分类

作为一门交叉学科，数据挖掘融合了来自不同学科领域的技术和成果，使得目前的数据挖掘技术和系统表现出多样的形式。为了帮助潜在的用户区分不同的系统从而选择适合自己需求的种类，需要对数据挖掘技术进行分类。数据挖掘技术有几种分类标准：根据挖掘的数据库的种类分类、根据发现知识的种类分类和根据采用的技术分类[13]。

1. 根据挖掘的数据库的种类分类

数据挖掘的数据来源比较广泛，主要分为以下几种：基于数据库的数据挖掘系统、基于数据仓库的数据挖掘系统以及基于国际互联网（WWW）或Web的数据挖掘系统等。其中，基于数据库的类型也有多种：关系型、交易型、面向对象数

据库、空间数据库、时态数据库、文本数据源、多媒体数据库、异质数据库、遗留数据库等。

2. 根据发现知识的种类分类

根据发现的知识种类，数据挖掘可分为：关联规则发现、分类或预测模型知识发现、数据总结、数据聚类、序列模式发现、依赖关系或依赖模型发现、异常和趋势发现等。

3. 根据采用的技术分类

根据数据挖掘所采用的技术，数据挖掘大体上可分为：统计分析、机器学习方法、模式识别、面向数据库或数据仓库的技术、可视化技术和神经网络等。统计方法中，可细分为：回归分析、判别分析、聚类分析（系统聚类、动态聚类等）、探索性分析等。机器学习中，可细分为：归纳学习方法（决策树、规则归纳等）、基于范例学习、遗传算法等。数据库方法主要是多维数据分析或联机分析处理方法，另外还有面向属性的归纳方法。

当然，一个优秀的数据挖掘系统通常都是针对实际应用问题，结合各种挖掘技术的优点，设计出高效的、满足需求的系统。

2.1.3 挖掘过程

数据挖掘的过程如图2.2所示，一般由五个阶段构成：确定问题、数据的准备、模型的建立、模型的验证和评价、模型的实施。

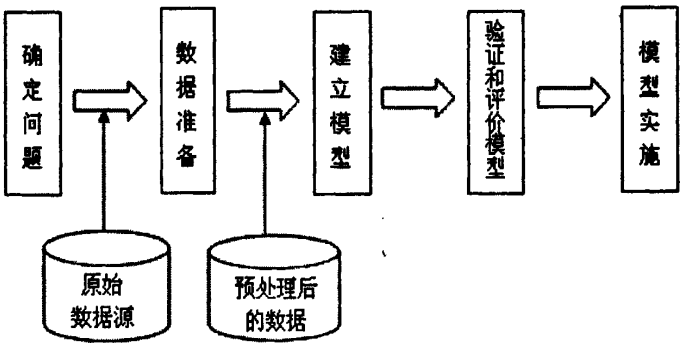

图2.2 数据挖掘过程图

1. 确定问题

数据挖掘的系统是针对实际应用的问题设计出来的，不同的需求需要不同的挖掘模型和技术。数据挖掘首先要明确表达出目标，然后才能提出解决方案，提高分析数据的效率，从而为决策者提供良好的支持。

2. 数据的准备

数据的准备一般包括数据收集、数据清理和集成、数据选择、数据转换等步

骤。

数据收集主要是根据实际问题，确定数据挖掘的数据，数据的来源可能是公共数据库或者购买的外部数据库等。

数据清理和集成主要是确定影响知识模型质量的数据特征。当数据来自不同的数据源时，需要对数据建立统一的数据模型，处理不一致的数据以及编码等问题，并检查数据的完整性，保证数据的统一性。

数据选择主要是分析挖掘任务需要的相关数据，并选择适当的数据。数据的选择必须合理，对于特定的挖掘任务起码不能损失相关信息。

数据转换主要是根据不同的挖掘方法，对数据进行一些转换工作，使得数据适合于挖掘的形式。

数据的准备要检查数据本身的质量，数据挖掘的目的是要探索数据的规律性的，如果原始数据有误，挖掘得出的知识模型很可能是错误的，再依此去指导工作，则很可能是在进行误导。

3. 建立模型

数据挖掘的核心工作环节就是模型的建立。模型的建立是一个反复的过程，需要从不同的角度考察不同的模型并判断哪个模型对于解决实际问题最为有效。同时，在反复的过程中，可能会导致对实际问题的重新思考，甚至改变其最初的定义。

根据实际问题确定好数据挖掘的类型之后，就需要选择模型的类型，模型的类型可能是一棵决策树、神经网络、甚至是传统的数学统计。选择什么样的模型决定了你需对数据做哪些预处理工作。如神经网络适合于绝对的数值，就需要做数据转换，有些数据挖掘工具可能对输入数据的格式有特定的限制等。数据准备好之后，就可以开始进行模型的训练了。

从目前的数据挖掘的技术水平来看，数理统计方法还是数据挖掘工作中最常用的主流技术手段。市场上很多的软件供应商和数据挖掘咨询公司一般都提供了很多的软件包，包含有很多实用的数理统计方法。而在你的数据挖掘模型中使用哪一种方法，具体用软件包中的什么方法来实现，主要取决于数据集的特征和要实现的商业目标，可以运用多种挖掘方法，从挖掘的效果评价来选择合适的挖掘方法。

4. 验证和评价模型

数据挖掘的模型建立完成后，可能存在一些对实际应用来说是冗余或者无意义的模式，需要对其进行验证和评估。为了便于识别挖掘生成的模型，还需要借助图形和图像等方式对挖掘的知识加以表示，这被称为挖掘结果可视化。可以使用建立模型的训练样本集或者新的测试集对模型进行测试，基于模型预测得到的

值和概率来评估模型的准确度，同时与商业分析人员讨论模型的意义。

一般来说，使用模型得到的如果是一个直接的结论，则当然很好，但是，实际上这种情况非常的少，更多的时候得出的是对目标问题多侧面的描述，这时就要能很好地总结它们的规律性，提供合理的决策支持信息。所谓合理，实际上往往是要在所付出的代价和达到预期目标的可靠性的平衡上做出选择。假如在数据挖掘过程中，就预见到最后要进行这样的选择的话，那么最好把这些平衡的指标尽可能地量化，以利于综合抉择。

在实际应用中，随着应用数据的不同，模型的准确率肯定会有所变化。更重要的是，准确度自身并不一定是选择最好模型的正确评价方法，需要进一步了解错误的类型和由此带来的相关费用的多少。在实际应用中，如果每种不同的预测错误所需付出的代价也不同的话，那么代价最小的模型就是我们所要选择的。

影响数据挖掘质量的因素最主要的有两个：一是数据挖掘方法的有效性，二是用于数据挖掘的数据质量和数量大小。经过模型的评估，可以验证模型的有效性。数据挖掘的过程是一个不断反馈的过程，对模型的评价较低，没有产生预期的效果，则需要重新选择挖掘的方法或者数据集。

5. 模型的实施

模型建立并且经过验证之后，可以将其嵌入到商业应用中或者应用于新的数据集上，实现智能化，提供预测给商业分析人员或者业务人员作为参考，进而提出切实有效的行动方案。当然，对于一个复杂的应用，数据挖掘可能只是整个产品的一小部分，虽然可能是最关键的一部分。例如，在欺诈检测系统中，我们常常把数据挖掘得到知识与领域专家的知识结合起来，然后应用于数据库中的数据。

对于实际的商业应用，每个挖掘模型都有其生命周期，因为数据可能是稳定的，也可能是频繁更新的，模型也就必须随之更新，这些都取决于问题的变更。最终，应该使用自动化的过程来确定模型的精确度和创建新的模型。

2.2 基于 web 的数据挖掘

2.2.1 概述

Internet 的迅猛发展导致了互联网上信息的急剧膨胀，如何对这些信息进行分类和复杂的应用成为研究的热点，目前正在蓬勃发展的数据挖掘技术正好为 Web 挖掘提供了技术基础。Web 数据挖掘是利用传统的数据挖掘技术，对 Web 信息进行挖掘分析，以发现有效的、隐含的、新颖的并最终是可理解的模式或者规则的过程[14]。相对于传统的基于数据库的数据挖掘，基于 Web 的数据挖掘显然要复杂得多。基于数据库的数据挖掘，其数据来源一般情况下都是结构化的，数据的定

义很清楚，而基于 Web 的数据挖掘，其数据来源则是庞杂的互联网信息，结构上是属于半结构化或者无结构化的，数据一般是计算机不可理解的，或者说数据是没有定义的，这也导致一些传统的数据挖掘技术无法应用于 Web 挖掘。

基于 Web 的数据挖掘是由传统的数据挖掘发展而来的，与数据挖掘的定义相类似，基于 Web 的数据挖掘是指从 Web 数据中提取隐含的、有用的、先前未知的知识模式的过程。但是 Web 挖掘是一项综合技术，涉及自然语言理解、人工智能、数据挖掘、计算机语言学、信息学等多个领域，不同研究者从自身的研究领域出发，对 Web 信息的含义也有着不同的理解，Web 挖掘的方法和知识模式的意义也各有其侧重点。例如，国外有人认为：Web 挖掘就是利用数据挖掘技术，自动地从网络文档以及服务中发现和抽取信息的过程。国内也是众说纷纭，有学者将网络环境下的数据挖掘归入网络信息检索与网络信息内容的开发。也有站在信息服务的角度上提出“Web 挖掘”，指出其有别于传统的信息检索，能够在异构数据组成的信息库中，从概念及相关因素的延伸比较上找出用户需要的深层次的信息，并提出 Web 挖掘将改革传统的信息服务方式而形成一个全新的适合网络时代要求的信息服务组合[15]。其实，Web 挖掘就是对 Web 文档的内容、可利用资源的使用以及资源之间的关系进行分析，以实现对 Web 存取模式、Web 结构和规则的分析，以及动态 Web 内容的查找。

虽然 Web 挖掘是从数据挖掘发展而来的，但相对于传统的数据挖掘，Web 数据挖掘有自身独特的地方。最主要的区别就是数据的来源不同，Web 挖掘的数据是分布式的、异构的 Web 文档，并且缺乏机器可理解的语义。传统的数据挖掘技术需要建立在对 Web 文档进行预处理、得出数学模型的基础之上，对 Web 文档进行特征表示是 Web 数据挖掘的研究重点。Web 在逻辑上是一个由文档节点和超链接构成的图，因此 Web 挖掘所得到的模式可能是关于 Web 内容的，也可能是关于 Web 结构的。对于 Web 挖掘来说，对 Web 服务器上的用户日志等数据进行的挖掘，属于传统的数据挖掘范畴。

2.2.2 分类

Web 上信息的多样性决定了挖掘任务的多样性[16]。Web 数据总体上可以分为以下三种：Web 页面数据，主要是 Web 文本和多媒体信息，比如视频、音频或者图像等；Web 页面的超链接关系数据，代表了 Web 文档之间的某种联系，同时为用户浏览 Web 页面提供路径；Web 服务器的日志数据，这些数据主要是用来记录用户访问站点的情况，对于日志数据来说，其语义是机器可理解的，属于传统的数据挖掘范畴。相应的，Web 挖掘也分为三类(分类图如图 2.3 所示)：内容挖掘(Web Content Mining)、结构挖掘(Web Structure Mining)和使用挖掘(Web Usage Mining)。

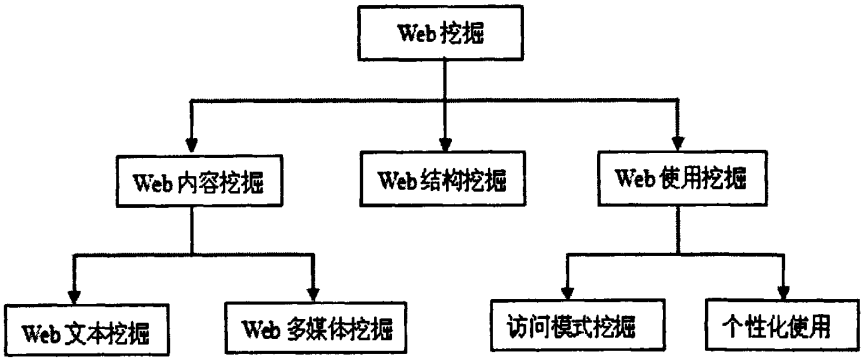

图 2.3 Web 挖掘的分类

(1) Web 内容挖掘

Web 内容挖掘是从 Web 文档本身的内容或者 Web 搜索的结果中抽取知识的过程，相应的，Web 内容挖掘也有两种策略：直接挖掘 Web 文档的内容，或在其它搜索工具的基础上进行改进。Web 内容挖掘可以对大量的 Web 文本集合进行分类、聚类、关联分析，对 Web 内容自动提取摘要，或者利用 Web 内容进行趋势预测。

Web 文档包含了各种不同种类的数据类型，例如：文本、图像、音频、视频、元数据（是指关于数据的数据，用以描述数据的属性）和超链接等，对 Web 上的图像、音频等的挖掘称之为 Web 多媒体挖掘。Web 内容数据是由非结构化数据（例如：自由文本）、半结构化数据（例如：HTML 文档）和结构化数据（例如：由 HTML 页面生成数据表）构成[17][18]。

Web 内容挖掘一般可以采用以下两种方法：传统的数据库方法：适用于 Web 内容挖掘的数据库方法是运用适当的技术把半结构化的 Web 数据组织成更加结构化的数据集合，并且使用标准的数据库查询机制和数据挖掘技术来分析这些数据集[19]。对 Web 页面内容进行挖掘，对页面中的文本进行文本挖掘，对页面中的多媒体信息进行多媒体信息挖掘，包括对页面内容摘要、分类、聚类以及关联规则发现。

(2) Web 使用挖掘

Web 使用挖掘的主要目标是从 Web 使用数据或者访问日志中提取感兴趣的模式。用户在访问 Web 服务器时，Web 服务器会产生记录关于用户访问和交互的信息，保存在日志文件或者数据库中。Web 使用挖掘就是从这些信息中进行挖掘，以发现用户的浏览习惯、相似用户群体、Web 页面的访问频率等知识模式。分析这些知识模式，可以更好地理解用户的行为，从而改进 Web 站点，例如设置访问量大的 Web 页面更加容易访问或者得到更多的链接、增加缓冲机制以提高 Web 服务器的响应时间或者合理地设置广告位等，为用户提供更加智能化的服务。

Web 使用挖掘主要包括了针对用户群的一般的访问模式追踪和针对单个或者某一类用户的个性化使用记录追踪。一般化的访问模式追踪是从 Web 日志中挖掘

用户的访问模式和预测用户的访问趋势。个性化的使用记录追踪是挖掘某一类或某几类用户（甚至某个用户）访问网站的行为规律，这使得网站能够动态地为用户提供个性化的服务以极大地满足用户的需求。

(3) Web 结构挖掘

Web 结构挖掘就是对 WWW 的组织结构、Web 页面的超链结构等进行挖掘并从中提取出隐藏的有价值的知识。Web 页面不是孤立存在的，相关的文档之间通常由超链接相互链接。Web 页面之间的超链反映了 Web 页面之间的某种联系，例如包含、从属等。超链中的标记文本(anchor)对链接页面起到了概括作用，并且这种概括在一定程度上比链接页面的标题要更为客观、准确。由于文档之间的互连，有用信息不仅包含在 Web 页面内容之中，而且也包含在页面的结构之中[20]。大量的 Web 链接信息提供了丰富的关于 Web 内容相关性、质量和结构方面的信息，对 Web 挖掘而言是可以利用的一种重要资源。目前，Web 结构挖掘的主要算法有 PageRank 算法和 HITS(Hyperlink-Induces Topic Search)算法等[21,22]。

PageRank 算法是斯坦福大学的博士研究生 Serger Brin 和 Lawrence Page 提出的，该算法基于网络的链接分析，建立在随机冲浪者模型之上，即用户点击链接的行为，是一种不关心内容的随机行为，而用户点击链接的概率完全由链接的数量来确定。PageRank 算法基于这样一个假设，获得链接越多的 Web 页面，其重要性越高。PageRank 算法的基本思想如下：对于一个 Web 页面 w，有指向其他 Web 页面的出链集合 $O(i)$，有指向 Web 页面 w 的入链集合 $I(i)$，入链集合 $I(i)$ 的数目越多，表示 Web 页面 w 的重要性也高，其重要程度由入链集合 $I(i)$ 的 Web 页面的出链集合数目所确定，出链集合数目越多，表示其给 Web 页面 w 的贡献越小。PageRank 方法是通过 Web 页面链接来反映页面的权威度，从而发现 Web 结构之间的相互关系。

HITS 算法是由康奈尔大学的 Jon Kleinberg 博士于 1998 年首先提出的。Kleinberg 认为用户的搜索行为从用户的检索开始，每个 Web 页面的重要性跟用户的检索关键字是密切相关的。用户的检索分为三种：主题检索提问，亦称窄主题提问；泛指主题提问，亦称宽主题提问；相似检索提问。HITS 算法主要致力于改善泛指主题的检索结果。HITS 算法引入了 Hub 网页的概念，Hub 网页是指那些指向权威(Authority)网页链接集合的 Web 网页，Hub 网页本身可能并不是重要的，并没有多少 Web 网页有指向 Hub 网页的链接，但 Hub 网页提供了指向某个主题的重要 Web 页面集合。一般来说，一个好的 Hub 网页指向许多好的 Authority 网页，一个好的 Authority 网页是有许多好的 Hub 网页指向的 Web 网页。可以利用 Hub 和 Authority 之间的相互关系来挖掘并发现 Web 结构和资源之间的关系。

2.3 Web 文本挖掘

Web 文本挖掘就是从 Web 文档和 Web 活动中发现、抽取感兴趣的潜在的有用模式和隐藏的信息的过程。Web 文本挖掘和通常的文本挖掘有类似之处。文本挖掘是指在大量文本集合或语料库上，发现其中隐含的、令人感兴趣的、有用的模式和知识。文本挖掘是数据挖掘的一个特殊的应用或方面。Ronen Feldman[23]在 KDD99 中给文本挖掘下了定义。他认为文本挖掘是一门新的研究领域，通过采用数据挖掘、机器学习、自然语言处理、信息检索和知识管理的技术以发现知识模式。但是，与通常的文档不同，Web 文档中的标记给文档提供了额外的信息，这些信息会对 Web 文档的特征向量空间产生影响。特殊的标记提供了一些 Web 文档的特征，我们可以利用这些标记更好地进行特征表示，借此提高 Web 文本挖掘的性能。

Web 文本挖掘属于 Web 内容挖掘的范畴,可以对上大量文档集合的内容进行概括、分类、聚类、关联分析。Web 文本挖掘是利用数据挖掘技术对 Web 文本进行挖掘，但是 Web 文本挖掘技术有别于基于数据库的数据挖掘技术。首先，与基于数据库的数据挖掘不同，Web 文本挖掘的对象是海量、异构、分布的 Web 文本，属于无结构或者半结构化数据，缺乏计算机可理解的语义，直接将基于数据库的挖掘技术应用到 Web 文本挖掘上，效果可能不理想。其次，Web 文本挖掘所要处理的是 Internet 上的庞大信息数据，其规模远非已有的数据库中的数据可以相比，因而使得许多对数据库中数据挖掘很有效的挖掘算法，特别是基于统计理论和机器学习的挖掘算法，在实际应用中，对于 Web 文本的挖掘变得不可行，时间效果很差。因为不可能将数据（哪怕只是抽样）一次调入内存处理，所以规模的增大，要求算法必须能够增量的执行。另外这对算法的效率也提出更高的要求。Web 的数据是不断增长的，新的数据类型不断出现。算法必须有能力在不完全重新分析已有数据的情况下，增量处理新的数据，更新挖掘结果。目前，基于统计理论和机器学习的 Web 文本挖掘算法，基本上都还处于研究阶段。

2.3.1 Web 文本挖掘过程

Web 文本挖掘的具体过程一般分为特征表示、特征提取、提取知识模式、评价模型质量。

1. 特征表示

Web 文档是通常是一组 HTML 格式的文档集合，属于无机构或者半结构的数据，缺乏像关系数据库中数据的组织完整性。必须采用某种模型对 Web 文档进行特征表示，将其转化成计算机可以处理的形式，并且，这些特征应该能够在一定

程度上反映 Web 文档的语义。特征表示的过程就是特征化文本信息的过程。

2. 特征提取

特征表示处理后的特征空间通常具有很高的维数，但其中有益于挖掘的特征通常只有很少的一部分，而由于过高维的特征不利于对 Web 文档的挖掘。因此，必须通过适当的方法来降低模型的维数，在不影响分类准确度的情况下，提取少量的特征，这些特征能很好地表示 Web 文档，提高挖掘的效率和准确度。

3. 提取知识模式

应用分类、聚类、关联分析等 Web 文本挖掘处理方法来提取面向特定应用的知识模式。Web 文本挖掘处理的数据就是经过提取的特征表示。

4. 评价模型质量

对获取的知识模式进行质量评价，若满足预先设定的要求，则存储或输出知识模式，否则返回到某个最可能影响模式质量的环节，进行新一轮挖掘工作。可以采用一些常用的评价方法，也可以选择建立面向特定应用的评价模型，这取决于 Web 文本挖掘的目的。

2.4 本章小结

本章主要描述了数据挖掘、Web 数据挖掘的基本概念与知识，介绍了 Web 文本挖掘的方法和过程。本章是后续工作的研究基础。

第三章 相关技术和理论

3.1 中文分词技术

Web 文本挖掘处理的是自然语言文本，对 Web 文本进行分词并建立数学模型，成为文本挖掘分析处理的基础。

中文和西方语言在语言学上的差异，导致在分词处理技术上有很大的区别。西方语言中的词在书写时是用空格分开的，因而词间界限非常清楚；而汉语在书写时，词与词之间没有空格，词的界限没有明确标注，一个汉语句子就是一大串前后相续的汉字的字符串。但在汉语中，词是最小的、能独立活动的、有意义的语言部分[24]，因此对词的切分是必要的。通过对文档的扫描，将文档分解成为词的集合，是实现中文文本结构化表示的前提。

中文分词技术属于自然语言处理技术范畴，是语义理解过程中最初的一个环节，它将组成语句的核心词提炼出来供语义分析模块使用。在分词的过程中，如何能够恰当地提供足够的词来供分析程序处理，并且过滤掉冗余的信息，这是后期语义分析的质量和速度的重要前提。中文分词是信息处理的难题之一。随着技术的发展，有很多用于中文分词的算法和软件被设计出来。因为技术路线的不同，不同的算法模型的分词结果和适用场合都有区别。目前，中文分词方法主要分为三大类：基于字符串匹配的分词方法、基于理解的分词方法和基于统计的分词方法。

3.1.1 基于字符串匹配的分词方法

基于字符串匹配的方法，也称机械分词方法，其基本思想是事先建立一个“充分大”的机器词库，词库中包含了所有可能出现的词，根据特定的策略切分待处理的中文字符串成子串，并与词库中的词条进行匹配，若在词库中找到某个词条与子串相匹配，则匹配成功[25](识别出一个词)。

基于字符串匹配的方法关键在于词库的建立。建立词库必须解决一个问题：哪些词应该收录到词库中？哪些词不需要收录到词库？词库的词量太少，分词效果可能不好，词量过大，又造成分词算法的时间复杂度和空间复杂度太高，并且会造成大量的歧义切分问题。目前，一般采用基本词库加专业词库的做法，其中，基本词库中收集那些与语料无关的常用词汇，专业词库则根据语料所属专业来选取。即使这样，也不能保证词库中确实含有特定语料中的所有词。为了对付这种情况，一般的中文分词系统都实现了动态维护(包括扩充)词库的功能。为了提高查

找匹配效率，词库又可细分为单字词库、双字词库、三字词库、四字词库和多字词库等。词库的数据结构比较简单，只需记录词的内部表示，而不必附带其他信息。词库可根据内部表示的大小组织成一个有序的表结构，这样可以利用二分查找法进行匹配查找，提高时间效率。词库的设计应以既省空间又能快速执行匹配查找为目标。

根据不同的扫描方向，基于字符串匹配的方法可以分为正向匹配和逆向匹配；根据长度优先匹配的情况，可以分为最大(最长)匹配和最小(最短)匹配；根据是否与词性标注过程相结合，又可以分为单纯分词方法和分词与标注相结合的一体化方法。

1. 正向最大匹配法：正向最大匹配法的目的是把最长的词给切分出来，其基本思想是：假设分词词库中的最大词长为 n，则取待处理的字符串序列中的前 n 个字作为匹配字段，查找分词词库，若词库中存在，则匹配成功，匹配字段作为一个词被切分出来；如果词库中找不到，则匹配失败，去掉匹配字段的最后一个字，重新进行查找，直到匹配成功或匹配字段为空。然后再按上面的步骤进行下去，直到切分出语料中的所有词为止。

2. 逆向最大匹配法：逆向最大匹配法的思路基本与正向最大匹配法相同，只是切分方向变成从右到左，如果匹配不成功，去掉匹配字段的首个字符。

由于中文本身的语言特点，一般很少采用正向最小匹配法和逆向最小匹配法进行分词。统计结果表明，正向最大匹配法的错误率为 1/169，逆向最大匹配法的错误率为 1/245。

3.1.2 基于理解的分词方法

基于理解的方法是利用句式信息、句法信息和语义信息来处理中文的歧义现象，其基本思想就是在分词阶段进行中文字符串的句式、句法、语义分析。一般的分词系统都是在分词的同时消除大部分的歧义切分现象。而有些系统则是把分词过程当成整个中文语言理解过程的一个阶段，通常是在后续分词过程中再处理歧义切分问题。基于理解的方法包括分词、句式语义分析两个处理过程，同时由控制系统进行调度，模拟了人对句子的理解过程，在控制系统的协调下，分词过程利用有关词语、句子等的句式、句法和语义信息来判断词语的歧义性。当前，基于理解的方法还处于理论阶段，因为中文语言系统具有很高的复杂性，这种方法所依赖的句式、句法和语义信息需要庞大的语言知识，而这些知识很难转换成计算机可以理解形式。

3.1.3 基于统计的分词方法

基于统计的方法不需要维护庞大的词典，这种方法只是对文档中的词句进行频率统计。两个相连的字同时出现的频率越高，说明这两个字是一个词的可能性更大，其代表字之间的紧密程度。当出现的频率超过指定阈值时，可以认为其构成了一个词。当然，同时出现频率高的字不一定就是词，比如“这一”。实际应用时，一般将基于统计的方法跟基于字符串匹配的方法一起交替使用，将串频统计和串匹配结合起来，既发挥字符串匹配分词切分速度快、效率高的特点，又利用了统计方法的结合上下文识别生词、自动消除歧义的无词典优点。

3.2 特征表示方法

Web 文本属于无结构或半结构的数据，没有关系数据库中数据的结构化特性，Web 文本挖掘系统必须将无结构或半结构的 Web 文本转化为数学模型，并对其进行编码。

3.2.1 向量空间模型

向量空间模型(VSM)是 Salton 等人于上个世纪 60 年代末提出的[26]，该模型采用项的选择、权重赋值策略，被广泛应用在自动索引和信息检索等领域，是高效的文本表示模型。

词、词组和短语是组成文档的基本元素，在不同内容的文档中各词条出现频率有一定的规律性，不同的特征词条可以区分不同内容的文本。向量空间模型是以特征项作为文档表示的基本单位，特征项由字词或短语组成，对每一个特征项赋予一定的权值，文档可表示为一组特征项及其权值组成的向量空间。

令文档集合 $D=\{d_1,...,d_i,...,d_n\}$，则文档 d_i 的向量空间可以表示为

$$V(d_i)=(t_1,w_1(d_i);...;t_j,w_j(d_i);...;t_m,w_m(d_i)) \qquad \text{式(3-1)}$$

其中，t_j 表示文档 d_i 的一个特征项，$w_j(d_i)$ 表示特征项 t_j 的权值。

$$w_j(d_i)=tf_{ij}/df_j \qquad \text{式(3-2)}$$

其中，tf_{ij} 表示特征项 t_j 在文档 d_i 中出现的频率，称为项频；df_j 表示文档集合中含有特征项 t_j 的文档数量。权值公式表示如果特征项 t_j 在文档 d_i 中有较高的项频

和相对文档集合较低的文档频率，则其在文档d_i中的权重就较大[27][28]。

3.2.2 改进的权值计算方法

与一般的文本不同，Web 文本中包含大量的 HTML 标记，有些标记之间的词比较具有代表性。比如 TITLE 标记表示 Web 文本的标题信息，而 H1、H2 等则代表了 Web 文本中某个段落的标题。META 标记包含了更多的信息，为了更好地被搜索引擎搜索到，META 标记常常包含了该 Web 文本的主题。我们可以利用 Web 文本的这些结构信息来确定一些更具有代表性的词组，给这些词组赋予更好的权值。利用词组出现的位置不同，与词组出现的频率相结合，对权值计算方法进行改进，能够帮助我们更有效的表示 Web 文本，建立数学模型。公式如下：

$$W_j(d_i)=\frac{1}{2}CW(t_j,\bar{d})+\frac{1}{2}SW(t_j,d_i) \qquad \text{式(3-3)}$$

公式(3-3)中的$CW(t_j,\bar{d})$是针对特征项t_j在HTML文档中出现的词频的权重计算方法，$SW(t_j,d_i)$则是针对该词在该 HTML 文档中出现的位置来计算权重。公式如下：

$$SW(t_j,d_i)=\sum_{e_k}(w(e_k)\times TF(t_j,e_k,d_i)) \qquad \text{式(3-4)}$$

这里e_k是一种 HTML 标记，它指的 TITLE 和 META 标签。$w(e_k)$表示分给e_k的权重值。$TF(t_j,e_k,d_i)$表示了特征项t_j出现在 Web 文档d_i的标记e_k中的次数。$w(e)$的定义如下：

$$w(e)=\begin{cases}\alpha, e\in\{META,TITLE\}\\ 1, \text{其它}\end{cases} \qquad \text{式(3-5)}$$

3.2.3 文本相似度

文本相似度是指文档之间的相关程度。在向量空间模型中，我们可以采用特征向量的欧式距离表征文本的相似度。Web 文本表示为特征向量空间$V(d_i)$，我们可以定义 Web 文本的相似度为：

$$Sim(d_i,d_j)=\sqrt{\sum_{k=1}^{n}\left|w_{ik}-w_{jk}\right|^2} \qquad \text{式(3-6)}$$

其中，w_{ik}表示 Web 文本d_i的第k维特征向量，n表示向量空间的维数，Web $Sim(d_i,d_j)$的值越小，说明 Web 文本d_i和d_j的相似度越高。

除了欧式距离之外，我们还可以采用特征向量的夹角余弦或者内积来度量文本相似度。夹角余弦忽略了特征向量的绝对权值，只从形状上考虑 Web 文本的关系。夹角余弦的 Web 文本相似度公式如下：

$$Sim(d_i,d_j)=\frac{\sum_{k=1}^{n} w_{ik}\cdot w_{jk}}{\left(\sum_{k=1}^{n} w_{ik}^{2}\right)\left(\sum_{k=1}^{n} w_{jk}^{2}\right)} \qquad \text{式(3-7)}$$

Web 文本d_i和d_j的夹角余弦值越大，表示它们的相似度越高。

内积的 Web 文本相似度公式如下：

$$Sim(d_i,d_j)=\sum_{i=1}^{n} w_{ik}\cdot w_{jk} \qquad \text{式(3-8)}$$

Web 文本特征向量的内积代数乘积越大，表示 Web 文本之间的相似度越高。

3.3 特征提取方法

使用向量空间模型表示法时，表示文档的特征向量的维数会达到成百上千那么大。同时，具有代表性的特征以及词汇特征也会很大，并且多数是冗余的。这种未经处理的文本矢量会给后继的处理工作带来巨大的计算开销，因此减少维数至关重要，这往往决定了文本挖掘的效率。特征提取主要用于选择出一部分最为有效的特征[29]。

3.3.1 特征词的文档频率

含有特征词的文本数目在文档集合中出现的概率称为该特征词的文档频率，简称 DF (Document Frequency)。其理论假设为稀有词条或者对分类作用不大，或者是噪声，可以被删除。基于特征词的文档频率的特征提取方法设定某个阈值，若特征词的文档频率高于该阈值，则表示其对文本特征表示的贡献大，若低于该阈值，则表示该特征词不含有或只含有较少的特征信息。基于特征词的文档频率的特征提取方法将低于阈值的特征词从原始的特征空间中去除，有效降低了特征空间的维数，提高了文本表示的时间效率，并且有可能提高特征表示的精度。但对于高频的特征词，若其均匀分布在文档集合中，则其特征表示的作用较弱；低

频词也有可能带有很大信息量，直接去掉低频词会影响分类效果。特征词的统计分布决定了特征表示的精度。

3.3.2 互信息方法

信息论最早引入互信息的概念，用于表示一个消息中两个信号之间的相互依赖程度。后来引入特征提取领域，用于表征特征 w 与类别 C 之间的依赖程度[30]。特征 t 与类别 C 的互信息公式如下：

$$MI(w,c_i)=\log\frac{P(w,c_i)}{P(c_i)\times P(w)} \qquad \text{式(3-9)}$$

其中，$P(w,c_i)$ 表示特征 w 出现在类别 c_i 中的概率，$P(c_i)$ 表示类别 c_i 在文档集合中的概率密度，$P(w)$ 表示文档集合中出现特征 w 的概率。当 $MI(w,c_i)>>0$ 时，表示特征 w 与类别 c_i 之间的关联程度较强；当 $MI(w,c_i)\approx 0$ 时，表示特征 t 与类别 c_i 之间的关联程度较弱；当 $MI(w,c_i)<<0$ 时，表示特征 w 与类别 c_i 存在互补的统计分布关系，不存在关联关系。对于低频的特征，其对互信息的影响较大，容易导致过学习现象。公式(3-9)的近似计算公式如下：

$$MI(w,c_i)=\log\frac{A\times N}{(A+C)\times(A+B)} \qquad \text{式(3-10)}$$

其中，A 是属于类别 c_i 并且包含特征 w 的文档数目，B 表示不属于类别 c_i 并且包含特征 w 的文档数目，C 表示属于类别 c_i 但并不包含特征 w 的文档数目，N 是整个文档集合的数目。

互信息方法的优点在于考虑低频特征带有信息量的情况，低频特征有时比高频特征的互信息要高。但互信息方法过于倾向低频特征，忽略高频特征。Web 文档中含有过多的低频特征将影响文本特征的提取。若一个文档集合中特征向量具有均匀分布的统计特征，互信息方法则不能精确表征特征向量的重要程度。

3.3.3 χ^2 统计量

基于 χ^2 统计量的特征提取方法亦称开方拟合检验方法(CHI)，表示一个特征 t 与一个类别 C 之间的相关程度[31]。基于 χ^2 统计量的特征提取方法假定在某个类别

C和其他类别中出现的高频特征对判断一个文档是否属于类别C有帮助，其计算公式如下：

$$\chi^2(w,c)=\frac{N\times(P(w,c)\times P(\bar{w},\bar{c})-P(w,\bar{c})\times P(\bar{w},c))}{P(w)\times P(\bar{w})\times P(c)\times P(\bar{c})} \quad \text{式(3-11)}$$

其中，w表示任意一个特征，c表示任意一个类别，N代表文档集合中的文档数目，$P(w,c)$为特征t和类别c同时出现的概率(即表示一个文本的特征向量空间包含特征w并属于类别c的概率)，$P(\bar{w},\bar{c})$表示文本的特征向量空间不包含特征w并且不属于类别c的概率，$P(w,\bar{c})$表示文本特征向量空间包含特征w但不属于类别c的概率，$P(\bar{w},c)$表示文本特征向量空间不包含特征w但属于类别c的概率。$P(w)$表示特征w出现在某一文本中的频率，$P(c)$为文本属于类别c的概率。若$(P(w,c)\times P(\bar{w},\bar{c})-P(w,\bar{c})\times P(\bar{w},c))>0$，表示特征$w$和类别$c$存在正相关的联系；若$(P(w,c)\times P(\bar{w},\bar{c})-P(w,\bar{c})\times P(\bar{w},c))<0$，表示特征$w$和类别$c$存在负相关的关系。若特征$w$和类别$c$不相关，则$\chi^2(w,c)$值为0。

式(3-11)表示一个特征相对于一个类别的CHI值，对于整个文档集合，特征的CHI值将取平均加权的综合方式，其计算公式如下：

$$\chi^2=\sum_{i=1}^{m}P(c_i)\chi^2(w,c_i) \quad \text{式(3-12)}$$

其中，m表示文档集合中的文档数目，$\chi^2(w,c_i)$表示特征w对于类别c_i的χ^2统计量，$P(c_i)$为文本属于类别c_i的概率。

3.3.4 信息增益方法

信息增益方法是基于熵的评价方法，涉及众多数学理论和复杂的熵理论，广泛应用与机器学习领域。信息增益方法定义为某个特征在文档出现前后的信息熵之差，其计算公式如下：

$$IG(w_j)=-\sum_{k=1}^{|C|}P(c_k)\log P(c_k)+P(w_j)\sum_{k=1}^{|C|}P(c_k\mid w_j)\log P(c_k\mid w_j)+$$

$$P(\bar{w}_j)\sum_{k=1}^{|C|}P(c|\bar{w}_j)\log P(c_k|\bar{w}_j) \qquad \text{式(3-13)}$$

其中，$IG(w_j)$反映了特征w_j为整个分类提供的信息量，$P(c_k)$表示类别c_k在整个文档集合中出现的概率，$P(w_j)$表示文档集合中包含特征w_j的文档的概率，$P(c_k|w_j)$表示包含特征w_j的文档属于类别c_k的概率，$P(\bar{w}_j)$表示文档集合中不包含特征w_j的文档的概率，$P(c_k|\bar{w}_j)$表示文档在不包含特征w_j的情况下属于类别c_k的概率，$|C|$表示文档集合中的文档数目。

特征提取时，设定信息增益阈值，计算特征的信息增益值，过滤掉增益值较低的特征，保留高增益值特征作为文档的特征。信息增益方法考虑到未出现特征对于文档的影响，故提取效果好。

3.4 Web 文本分类算法

Web 文本分类是按照预先定义的分类体系，将待分类的 Web 样本划归到一个或者多个类别中[32]。从数学角度看，Web 文本分类实际上是一个映射的过程，将待分类的 Web 样本映射到已存在的 Web 文本类别。Web 文本分类的映射规则是系统根据已经掌握的每类若干样本的数据信息，总结出分类的规律性而建立的判别公式和判别规则。然后在遇到新文本时，根据总结出的判别规则，确定文本相关的类别。Web 文本能有效处理海量的 Web 文本，提供分门别类的组织结构，提高 Web 文本的检索效率。经过文本分类处理，用户不但能够方便浏览文本，而且可以通过限制搜索范围来使文本的查找更为容易。

Web 文本分类是有指导的机器学习问题，一般分为训练和分类两个阶段，具体过程如下：

训练阶段：

首先要定义 Web 文本的类别集合$C=\{c_1,...,c_i,...,c_m\}$，这些类别可以是层次式的，也可以是并列式的；给出训练文档集合$S=\{s_1,...,s_j,...,s_n\}$，每个训练文档$s_j$被标上所属的类别标识$c_i$，文档可以有多个标识，实际上，很多类别都不是相互排斥的。统计S中所有文档的特征矢量$V(s_j)$，确定代表训练文档集合的特征矢量$V(c_i)$。

分类阶段：

对于测试文档集合$T=\{d_1,...,d_k,...,d_r\}$中的每个待分类文档$d_k$，计算其特征矢量$V(d_k)$与每个$V(c_i)$之间的文本相似度$sim(d_k,c_i)$，选取相似度最大的一个类别作为$d_k$的类别。也可以预先设定一个阈值，超过这个阈值，就将d_k划归该类别。实际上，很难确定阈值的初始值，理论上没有很好的解决方法，一般是根据经验来确定，不同的训练集或者训练集的大小不同，阈值都不一样，需要反复调整。

3.4.1 KNN 算法

近邻(Nearest Neighbor)算法是在模式识别中广泛应用的一种分类算法，是模式识别非参数算法中最重要的方法之一。K 近邻 (K-Nearest Neighbor) 算法[33]是近邻算法的推广。KNN 算法是基于机械学习的分类算法，是分类算法中较为简单的算法，算法不需要复杂的推理过程。

KNN 算法的基本思路：对于给定的文本样本，获取其特征向量空间，在训练集中搜索特征向量最近的 K 个近邻，统计 K 个最近邻的类别分布，将文本样本划归到统计值最高的一个类别。KNN 算法采用特征向量权值作为近邻的依据，权值相近的特征向量相似度较高。

目前并没有很好的方法确定 K 值，一般先定一个初始的 K 值，然后根据实际情况进行调整。

KNN 算法比较简单，容易实现，具有良好的时间复杂度和空间复杂度。

3.4.2 朴素贝叶斯算法

朴素贝叶斯算法(Naïve Bayes)也称简单贝叶斯算法，是一种简单而有效的传统分类算法[34][35]。概率统计理论中，Bayes 公式表述如下：对于样本空间S，$B_1,B_2,...B_n$是样本空间S的一个划分，即

$$\begin{cases} B_1 \cup B_2 \cup ... \cup B_n = S \\ B_i B_j = 0 \quad i \neq j, i, j = 1,2,...n \end{cases} \qquad \text{式(3-14)}$$

并且$P(B_i)>0(i=1,2,...n)$，A是样本空间S的一个事件，则存在

$$P(B_i \mid A) = \frac{P(A \mid B_i)P(B_i)}{P(A)} = \frac{P(A \mid B_i)P(B_i)}{\sum_{j=1}^{n} P(A \mid B_j)P(B_j)} \qquad \text{式(3-15)}$$

式(3-15)即为贝叶斯公式，贝叶斯公式是建立在样本空间划分是独立的基础

上。对于文本分类，朴素贝叶斯算法基于一个重要的假定，即特征向量空间的每一维向量都是互相独立的。文档d_i属于类别c_j采用朴素贝叶斯算法的计算方法如下：

$$P(c_j \mid d_i) = \frac{P(d_i \mid c_j)P(c_j)}{P(d_i)} \qquad \text{式(3-16)}$$

其中，$P(c_j)$表示文档集合中，文档属于类别c_j的概率。由于$P(d_i)$对于所有类别来说均为常数，所以有

$$P(c_j \mid d_i) \propto P(d_i \mid c_j)P(c_j)$$

由于我们假定样本的特征向量空间的各维特征向量是相互独立的，有

$$P(d_i \mid c_j) \propto P(c_j)\prod_{w_k \subset d_i} P(w_k \mid c_j)$$

其中，$P(w_k \mid c_j)$表示特征向量w_k在类别c_j中出现的概率。为了避免$P(w_k \mid c_j)$的值等于 0，我们可以采用 laplace 概率估计。

朴素贝叶斯算法中，文本被看作是词按照一定的方式“产生”的集合，根据产生的方式，朴素贝叶斯算法分为两种模型：多变量贝努里事件模型(Multi-variate Bernoulli event Model)和多项式模型(Multinomial event Model)。这两种模型的区别在于对$P(d_i \mid c_j)$的估计方法不同。

对于多变量贝努里事件模型，文本向量被看作布尔权重，如果特征词在文本中出现，则权重为 1，否则，权重为 0，不考虑特征词的出现顺序，忽略特征在文本中的出现次数，即某个特征出现或者不出现。多变量贝努里事件模型是贝叶斯分类算法的一个传统模型。

设B_{xt}表示特征在文本中的出现情况，则：

$$P(d_i \mid c_j) \propto P(c_j)\prod_{w_k \subset d_i}(B_{xt}P(w_k \mid c_j) + (1 - B_{xt})(1 - P(w_k \mid c_j)))$$

对于类别c_j中的某个样本，有

$$P(w_i \mid d_t) = \begin{cases} 1 & w_i \in d_t \\ 0 & w_i \notin d_k \end{cases} \qquad \text{式(3-17)}$$

$$P(w_k \mid c_j) = \frac{\sum_{d_i \in c_j} P(w_i \mid d_t)}{|c_j|} \qquad \text{式(3-18)}$$

其中，$P(w_i|d_t)$表示特征向量w_t在属于类别c_j的文档样本d_k的概率，$|c_j|$表示类别c_j中样本的数目。

可以对式(3-18)进行平滑处理：

$$P(w_k|c_j)=\frac{1+\sum_{d_k\in c_j}P(w_i|d_t)}{2+|c_j|} \qquad \text{式(3-19)}$$

多项式模型把文档看作一系列有序排列的词的组合，假定文档长度对于类别没有影响，并且词与其在文档中的位置无关。则词在文档中出现n_i次的概率为$P(w_k|c_j)^{n_i}$，有

$$P(d_i|c_j)\propto P(c_j)\prod_{w_k\subset d_i}\frac{P(w_k|c_j)^{n_i}}{n_i!}$$

$$P(w_k|c_j)=\frac{|c_{jk}|}{\sum_k|c_{jk}|} \qquad \text{式(3-20)}$$

其中，$|c_{jk}|$表示特征词w_k在类别c_k中出现的次数，$\sum_k|c_{jk}|$表示所有特征词在类别c_j中出现的次数。

为了避免$P(w_k|c_j)$出现零值，采用 laplace 平滑技术：

$$P(w_k|c_j)=\frac{|c_{jk}|+\delta}{\sum_k|c_{jk}|+\delta|c_j|} \qquad \text{式(3-21)}$$

δ可以取任意的非零正整数，一般取 1，$|c_j|$表示类别c_j的总特征词数。

对文档集合中所有的类别，计算其贝叶斯概率$P(c_k|d_i)$，将样本分类到贝叶斯概率最高的类别中。

3.5 Web 文本聚类算法

Web 文本聚类是将 Web 文本集合分组成为由类似文本对象组成的多类主题的过程。每个组里的文本在一定方面互相接近。如果把 Web 文本内容作为聚类的基础，不同的组则与 Web 文本集不同的主题相对应。Web 文本聚类是一种典型的无教师的机器学习问题。目前的文本聚类方法可以分为两大类：层次凝聚法[36]和平面划分法。

层次凝聚法：一般层次凝聚法最终构造出一棵生成树，树的一个结点表示一个簇，树根是包含了所有文本的簇，树叶是仅包含一篇文档的簇。每一个非叶结点是两个子结点(文档或簇)合并而成或者是父结点分裂而来。存在许多方法确定两个簇的合并，如 single-link，group average， ROCK 和 complete-link。层次凝聚法的特点是能够生成层次化的嵌套簇，并且准确度高。但是，在每次合并时，需要全局地比较簇之间的相似度，并选择出最佳的两个簇，因此速度较慢，不适合大量文档的集合，并且不能产生相交簇。

平面划分法：平面划分法是将文档集合水平地分割为若干个簇，不像层次凝聚法是生成层次化的嵌套簇。K 均值[37]，K 中心点，Autoclass，Graph-partitioning-based，Single-pass 和 Spectral-partitioning-based 等都属于平面划分法。平面划分法的特点是聚类速度较快，比较适合对 Web 文档集聚类，也适合联机聚类；并且可以产生相交簇，但也有缺点，比如 K 均值算法的主要缺点是：必须事先确定 K 的取值，且种子选取的好坏对聚类结果有较大影响；只有当所需簇关于使用的相似度近似于球形时，它的效果才是最优的，但实际情况中文档很可能不是落在球形簇内。Single-pass 方法也有这种缺点，另外它还依赖于处理文档的循序，而且容易产生大型集簇。

相对于层次凝聚法，平面划分法的计算量相对较小。对于大规模的文档来说，平面划分法比层次凝聚法更适合进行聚类。平面划分法中的 K 均值算法的时间复杂度为 $O(nkT)$，而层次凝聚法中的 Single-link 和 group-average 方法的时间复杂度则有为 $O(n^2)$，其中。n 表示文档数目，T 表示迭代次数，k 表示最终聚类数目。

除此之外，也出现了结合层次凝聚法和平面划分法的方法。

3.5.1 SOM 算法

SOM(Self Organizing Maps)网络是芬兰学者 T.Kohonen 在 1981 年提出的具有自组织、自学习功能以及侧向联想能力的人工神经网络。与人脑的神经细胞分工的差异性相似，T.Kohonen 认为一个神经网络在接受外界输入时，不同的神经区域对输入模式具有不同的响应特征，并且，响应过程是自动完成的。SOM 算法是基于神经网络模型的文本聚类算法。SOM 算法通过模拟人脑对信号处理的特点来实现文本聚类，SOM 算法由输入层和竞争层(输出层)构成两层神经网络，n 个神经元组成输入层，而竞争层则是一个 $m \times m$ 神经元组成的二维平面阵列，输入层和竞争层的各个神经元实现双向完全连接。并用 w 表示连接权重。SOM 算法的学习过程就是对所有的 n 个样本向量进行迭代学习，直到权值 w 的变化小于某一个设定的阈值或迭代达到一定的次数。对于每一个样本的向量，计算其获胜神经元，输出单

元相同的样本向量属于同一类。虽然 SOM 算法有学习过程，但是这种学习是从所有的样本中自动获取特征，没有教师的参与，因此称为无监督的学习方法。

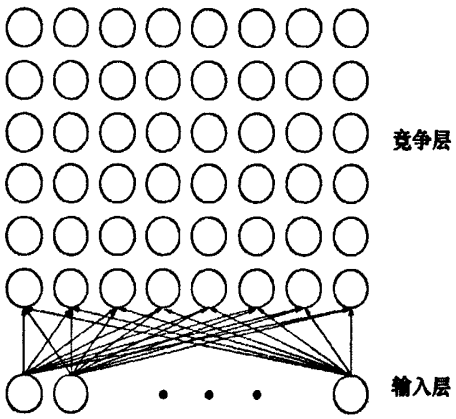

图 3.1 SOM 算法神经网络图

SOM 算法在开始学习时，首先赋予竞争层节点一个很小的随机权值。输入训练样本后，使输出节点进行竞争，并调整获胜神经节点及其邻域神经节点的权值。学习完毕后，竞争层节点的分布能够很好的保留数据在原来样本空间的拓扑结构。

SOM 算法的学习过程采用自组织竞争的原则。输入层神经单元对应一个 d 维的输入向量，竞争层的每个神经单元则分别对应一个 d 维权值向量。SOM 算法的学习过程如下：

1. 初始化竞争层每个神经节点的权值，并定义最大训练长度。SOM 算法的收敛条件很难找到，一般要定义一个最大长度作为训练结束的条件；

2. 从训练样本集中选择某个样本，计算样本与竞争层节点之间的距离，选出与样本距离最近的输出节点，即最匹配节点 (Best Match Unit, BMU)；

3. 根据预定义的邻域函数确定处于 BMU 邻域内的节点，调整 BMU 以及邻域内神经节点的权值；

4. 循环训练，直到收敛。

SOM 聚类算法克服了局部极值问题，能够反映训练样本的相似性关系，，有利于研究样本向量的统计分布，并且对噪声稳定。但训练集数目较少时，输入样本的次序会影响到 SOM 算法的聚类结果。SOM 算法在学习开始之前，必须确定一个竞争层的拓扑结构，要预先知道一个合适的拓扑结构是很困难的，这可能带来边缘效应、神经元欠利用等问题。Growing SOM 算法提出了一些改进，将 n 维的竞争层作为整体的构建模块，随着学习的不断深入，动态生成了 SOM 网络，提高了 SOM 算法的自适应能力。

3.5.2 K 均值算法

基于平面划分的聚类算法中，K 均值算法使用最为广泛。K 均值算法将文档集合分成 K 个聚类，使得每个聚类内具有较高的文本相似度，并且聚类间具有较低的文本相似度。K 均值算法可以采用文档间的欧式距离作为文本相似度的评判方法。

K 均值算法采用启发式的搜索，但需要预先指定聚类数目，并且需要设定相似度标准，因此对高维数据的适应性较差。算法的基本思路如下：

1、对于给定的文档集合 $D=\{d_1,...,d_i,...,d_n\}$，指定聚类的目标数目K，产生K个初始聚类中心。初始K个聚类中心的确定可以采用随机的方式，也可以采用其它的遴选原则。

2、根据文本相似度 $sim(d_i,c_k)$，将文本样本划分到相似度最高的类别中，并重新计算新的聚类中心。

3、重复 2 直到得到稳定的聚类结果，或变化小于指定阈值。

K 均值算法对于大规模数据集，具有相对的可扩展性，能有效的处理大数据集，并且保证聚类结果是一个超凸多面体。K 均值算法的收敛速度较快，具有较好的时间复杂度。由于通常采用欧氏距离作为文本相似度的度量标准，对高维空间的 Web 文本处理有一定的局限性，并且算法的效果与样本输入的次序相关。K 均值算法对孤立点或者“噪声”是敏感的，少量的孤立点将会对聚类中心产生几大的影响。K 均值作为一种启发式的算法，获得的聚类结果是一种局部最优解，并且容易陷入局部极值。

3.6 本章小结

本章分析了 Web 文本挖掘中的关键技术和理论，包括中文分词技术、特征表示和提取技术、Web 文本的分类挖掘和聚类挖掘算法，为 Web 文本挖掘系统提供理论支持。

第四章 基于K均值和遗传算法的聚类算法

K-均值聚类得到的是局部最优解，而遗传算法则具有较好的全部寻优能力，本文提出基于K均值和遗传算法的聚类算法，算法结合了K-均值聚类的局部搜索的优点和遗传算法的全局优化的特点，避免了K-均值算法陷入局部极值的现象。

4.1 遗传算法基本原理

Bagley J.D 在 1967 年提出遗传算法(Genetic Algorithm)的概念，1975 年 J.H Holland 开始对遗传算法的理论方法进行系统性的研究。遗传算法属于一种概率搜索算法，是自适应的迭代寻优过程。遗传算法从一个随机或者特定产生的初始种群开始，通过自然选择(selection)、交叉(crossover)、变异(mutation)等操作，不断进行迭代计算，并根据种群中每个个体的适应度值，过滤适应度低的个体，保留适应度高的个体，模拟自然进化过程来寻找所求问题的答案。遗传算法的主要优点是直接对结构对象进行操作，不存在求导和函数连续性的限定；具有内在的隐并行性和较好的全局寻优能力；采用概率化的寻优方法，能自动获取搜索过程中的有关知识并用于指导优化，自适应地调整搜索方向，不需要确定的规则。遗传算法具有良好的鲁棒性[38]。

遗传算法的基本求解过程如下：

1. 编码：编码是将问题的结构转化成位串形式表示的过程，这种位串形式的编码也叫做染色体，或者叫做个体。遗传算法常用的编码方法是二进制编码。

2. 初始化种群：随机选择 N 个个体，组成遗传算法的第一代种群，开始迭代过程。

3. 适应度函数：适应度函数是用来评估一个个体相对于整个种群的优劣程度。适应度的评估是模拟自然选择的过程，体现了染色体的适应能力。对于不同的求解问题，适应度函数的定义方式也不同，适应度函数的取值大小与求解问题对象有很大的关系。

4. 遗传操作：遗传操作包括选择、交叉和变异三种。遗传操作通过模拟自然进化的遗传原则来求解问题。

(1) 选择：也称复制操作，以适应度函数为评估标准，从当前种群中选出优良的个体，作为新的种群。适应度强的个体被选中的概率较大，而适应度弱的个体被选择的概率较小，遗传算法的选择过程体现了达尔文的适者生存原则。

(2) 交叉：交叉是从种群中抽取两个个体，按照某种交叉策略，使两个个体之

间相互交换部分染色体信息，得到新一代个体。

(3) 变异：变异是在种群中随机选择某个个体，并以一定的概率改变染色体信息。发生变异的概率是很低的，这也符合自然进化的规律。变异将产生相对于种群的新个体。

4.2 基于 K-均值和遗传算法的聚类算法

基于 K-均值和遗传算法的聚类算法的基本思路如下：

1. 首先对文档集合 $D=\{d_1,...,d_i,...,d_n\}$ 进行 K 均值聚类；

2. 随机产生一组特征向量的染色体，初始化种群；

3. 计算种群上每个个体的适应度函数；

4. 按照设定的概率对每个个体进行选择、交叉和变异操作，得到下一代群体；

5. 若完成了预先给定的进化代数或者达到概率收敛，输出种群中适应度最优的一组染色体作为结果，转到 7；否则，转到 6 进行操作；

6. 对新个体进行 K-均值优化；继续进行遗传操作；

7. 结合遗传算法的结果和 K 均值聚类的结果，得到聚类的划分。

算法首先对聚类样本进行 K 均值分析。K 均值的聚类基于文档特征向量空间与聚类中心的文本相似度来划分。对 K 取不同的值，可以控制解的质量和算法性能之间的平衡。K 均值算法具有很强的局部搜索能力，遗传算法建立在 K 均值聚类得出的良好解之上，可以弥补算法的局部搜索能力，并加快算法的收敛速度。

遗传算法容易产生早熟现象，并且无法避免多次搜索同一个可行解。目前也有出现很多混合遗传算法来解决此类问题。本算法采用 K 均值进行优化，若出现早熟现象，算法将结合遗传算法和 K 均值聚类的结果，重新进行 K 均值运算，并将运算结果作为新的个体继续进行遗传操作。

4.2.1 编码方法

遗传算法首先要解决的问题是编码。不同的编码方法，交叉算子和变异算法等遗传算子的运算方法也不同，编码方法在很大程度上决定了遗传算法的运算和进化效率，一个好的编码方法可以简化遗传操作[39]。目前，遗传算法的编码方法主要有以下几种：

1. 二进制编码方法

二进制编码方法是遗传算法中最常用的一种编码方法，其编码长度与问题求解的精度有关。二进制编码有以下优点：编解码的操作简单；易于实现交叉和变

异操作；符合遗传算法最小字符集的编码原则；便于利用模式定理对算法进行理论分析。但二进制编码反映不出问题的结构特征，对于连续函数的优化问题，由于遗传算法的随机特性而导致局部搜索的能力较差。

2. 实数编码：为了克服二进制编码的缺点，人们提出实数编码方法，即个体的每个基因值用实数表示。实数编码方法的优点如下：适合于遗传算法中表示范围较大的数；便于较大空间的遗传搜索；提高了遗传算法的精度要求；改善了遗传算法的计算复杂性，提高了运算效率；便于算法与经典优化方法的混合作用；便于设计专门问题的遗传算子。

在 Web 文本挖掘中，特征向量空间的维数可能达到成千上万维，若采用二进制编码，将导致遗传算法的搜索空间过大，在其中寻优使得遗传算法的运行性能相当差，所以基于 K 均值和遗传算法的聚类算法采用实数的编码方式。

经过分词、特征表示和提取后，Web 文本就表示为 n 维特征向量的组合(n 为特征向量空间的维数)，每个特征向量都有其权重。我们将 Web 文本表示为染色体，染色体的表示形式为：$w_1 w_2 ... w_k ... w_n$，w_k 表示 Web 文本的第 k 维特征向量。

4.2.2 适应度函数

适应度函数是对问题中的每一个染色体进行度量，体现了染色体的适应能力。适应度函数要有效地反映每一个染色体与最优解染色体之间的差距。若一个染色体与问题的最优解染色体之间的差距较小，则对应的适应度函数值之差就较小，否则就较大。本算法采用基于欧式距离的文本相似度作为评价来设计遗传算法的适应度函数，即：

$$f(s) = sim(c_i, r_j) = \sum_{k=1}^{n} (w_{ik} - w_{jk})^2 \qquad \text{式(4-1)}$$

其中，c_i 表示某个遗传个体，r_j 表示某个聚类中心，w_k 表示遗传个体 w_k 或者聚类中心 r_j 的第 k 维特征向量，n 表示特征向量空间的维数。

4.2.3 选择算子

遗传算法的选择操作的主要目的是为了避免基因缺失，提高全局收敛性和计算效率。选择操作是建立在对个体的适应度进行评价的基础上的，适应度高的个体被遗传到下一代群体中的概率大，适应度低的个体被遗传到下一代群体中的概率小。

最常用的选择算子是比例选择算子，比例算子是指个体被选中并遗传到下一代群体中的概率与该个体的适应度大小成正比。排序选择与个体的适应值的绝对值无直接关系，仅仅与个体之间的适应值相对大小有关。排序选择不利用个体适应值绝对值的信息，可以避免群体进化过程的适应值标度变换。由于排序选择概率比较容易控制，所以在实际计算过程中经常采用，特别是适用于动态调整选择概率，根据进化效果适时改变群体选择压力。最常用的排序选择方法是采用线性函数将队列序号映射为期望的选择概率，即线性排序选择。

基于K均值和遗传算法的聚类算法建立在K均值聚类结果之上，遗传算法的选择操作应该要提高算法的全局收敛性，避免选择、交叉、变异等遗传操作的随机性而破坏掉当前群体中适应度最好的个体。所以，算法采用最优保存策略进化模型作为选择算子来进行优胜劣汰操作，即当前群体中适应度最高的个体不参与交叉运算和变异运算，而是用它来替换掉下一代群体中适应度最低的个体。

最优保存策略进化模型的具体操作过程是：

1. 找出当前群体中适应度最高的个体和适应度最低的个体。

2. 若当前群体中最佳个体的适应度比总的迄今为止的最好个体的适应度还要高，则以当前群体中的最佳个体作为新的迄今为止的最好个体。

3. 用迄今为止的最好个体替换掉当前群体中的最差个体。

未成熟收敛是遗传算法中不可忽视的现象，主要表现在以下两个方面：群体中所有的个体大批趋于同一极值而停止进化；接近最优解的个体总是被淘汰，进化过程不收敛。为了抑制未成熟收敛的现象，需要在编码、适应度函数和遗传操作中考虑对策，包括提高变异概率、对适应度函数定标以及维持群体中个体的多样性。马尔可夫链方法证明，在概率收敛的情况下，经典遗传算法不会收敛到最优解。必须保留每一代的最佳个体，才能收敛到最优解。最优保存策略可以保证迄今为止所得到的最优个体不会被交叉、变异等遗传运算所破坏，这样不仅保留进化过程中出现的最优个体，而且是全局收敛的。

4.2.4 交叉算子

交叉运算是指对两个相互配对的染色体按某种方式相互交换其部分基因，从而形成两个新的个体。交叉运算是遗传算法区别于其它进化算法的重要特征，它在遗传算法中起关键作用，是产生新个体的主要方法。遗传算法中，在交叉运算之前还必须对群体中的个体进行配对，常用的配对策略是随机配对。目前，交叉算子主要有以下几种：

1. 单点交叉：也称简单交叉，它是指在个体编码串中随机设置一个交叉点，然后在该点相互交换两个配对个体的部分基因。

2. 双点交叉：是指在相互配对的两个个体编码串中随机设置两个交叉点，然后交换两个交叉点之间的部分基因。

3. 均匀交叉：它是指两个配对个体的每一位基因都以相同的概率进行交换，从而形成两个新个体。

4. 算术交叉：是指由两个个体的线性组合而产生的两个新的个体。为了能够进行线性组合运算，算术交叉的操作对象一般由浮点数编码所表示的个体。

本文采用算术交叉作为交叉算子，假设在遗传运算的第t代有两条染色体，为$(w_1^1...w_i^1w_{i+1}^1...w_n^1)$和$(w_1^2...w_i^2w_{i+1}^2...w_n^2)$，经过算术交叉运算后，为$(w'^1_1...w'^1_iw'^1_{i+1}...w'^1_n)$和$(w'^2_1...w'^2_iw^2_{i+1}...w^2_n)$，个体的线性组合为(其中，$\alpha$为一个常数)：

$$\begin{cases} w'^1_i = \alpha w_i^1 + (1-\alpha)w_i^2 \\ w'^2_i = \alpha w_i^2 + (1-\alpha)w_i^1 \end{cases} \quad \text{式(4-2)}$$

4.2.5 变异算子

所谓变异运算，是指将个体编码串中的某些基因值用其它基因值来替换，从而形成一个新的个体。遗传算法中的变异运算是产生新个体的辅助方法，但它是必不可少的一个运算步骤，因为它决定了遗传算法的局部搜索能力。交叉运算和变异运算的相互配合，共同完成对搜索空间的全局搜索和局部搜索。变异操作还可以维持种群的多样性，防止出现早熟的现象。目前，常用的变异操作方法有以下几种：

1. 基本位变异：是指对个体编码串以变异概率p_m，随机指定某一位或某几位基因作变异运算。

2. 均匀变异：是指分别用符合某一范围内均匀分布的随机数，以某一较小的概率来替换个体中每个基因。

3. 非均匀变异：是指对原有基因做随机扰动，以扰动后的基因值作为变异后的新基因值[40]。

4. 高斯变异：是指进行变异操作时，用均值为μ，方差为σ^2的正态分布的一个随机数来替换原有基因值。具体操作过程与均匀变异类似。

非均匀变异重点在原个体附近的某个局部区域进行搜索。非均匀变异在算法前期运行阶段进行均匀搜索，而后期阶段则进行局部搜索，随着遗传算法的运行，非均匀变异使得最优解的搜索局限在希望的重点区域中。基于K均值和遗传算法的聚类算法在K均值聚类的结果上，我们希望遗传过程中，重点在K均值的聚类

结果附近进行搜索，但又要避免象 K 均值算法容易陷入局部极值的现象。所以算法采用非均匀变异作为变异算子。具体的变异操作如下：

设有一条染色体 $w_1w_2...w_k...w_n$，进行非均匀操作之后，结果为 $w_1w_2...w'_k...w_n$，新基因值 w'_k 有下式确定：

$$w'_k = w_k \pm (MAX - MIN)\cdot(1 - r^{1-\frac{t}{T}}) \qquad \text{式(4-3)}$$

其中，MAX 和 MIN 分别表示 w_k 取值的最大值和最小值，r 表示[0,1]范围内符合均匀概率分布的一个随机数，T 表示最大进化代数。设定一个随机布尔数，当随机数为 0 时，采用加法，反之，采用减法。可以看出，随着进化代数 t 的增加，非均匀变异的搜索范围越来越局限在原个体附近，这正是算法所需要的。

4.3 实验结果

为了测试算法的聚类效果，必须对算法的聚类结果进行评价。目前，文本领域的常用评价标准主要有准确率(Precision)、召回率(Recall)、F-Measure 等。

$$\text{准确率 } p = \frac{tt}{tt + tf} \qquad \text{式(4-4)}$$

$$\text{召回率 } r = \frac{tt}{tt + ft} \qquad \text{式(4-5)}$$

其中，tt 表示属于某个类别并且正确聚类到该类别的文档数，tf 表示聚类到某个类别但属于其它类别的文档数，ft 表示属于某个类别但聚类到其他类别的文档数。

F-Measure 评价方法是 C.J.Van Rijsbergen 于 1979 年提出的，也称综合分类率，这种方法结合了准确率和召回率两种评价方法。F-Measure 的计算方法如下：

$$f = \frac{(\beta^2 + 1) * p * r}{\beta^2 * p + r} \qquad \text{式(4-6)}$$

其中，β 是用于调节准确率和召回率权重的参数，一般取 1，式(4-6)转化为：

$$f = \frac{2 * p * r}{p + r} \qquad \text{式(4-7)}$$

基于 K 均值和遗传算法的聚类算法还需要选择一些运行参数，包括编码长度、

种群规模，交叉概率P_c、变异概率P_m、终止进化代数T等。

1. 编码长度：算法采用了实数编码的方法来表示个体，所以编码的长度跟特征向量空间维数相等。

2. 种群规模：种群规模的大小影响算法的运算速度，较大时，会导致遗传算法的运行效率降低，而较小时，虽然可以提高算法的运算速度，但降低了群体多样性，并且可能会引起算法的早熟现象。一般建议在20~100之间。

3. 交叉概率P_c和变异概率P_m：交叉运算决定算法的全局搜索能力，而变异运算则决定了算法的局部搜索能力。算法的遗传操作要在K均值聚类结果附近搜索到满意解，需要提高算法的局部搜索能力，所以算法采取了较低的交叉概率和较高的变异概率，较高的变异概率同时可以抑制早熟现象。交叉概率建议选择在0.4~0.75，变异概率建议选择在0.01~0.1。

4. 终止进化代数T：终止进化代数T是算法运行结束的条件之一，一般建议的取值范围在100~1000之间。

5. K值和最优个体数：这两个参数的确定需要根据Web文本集的规模以及实际的需求进行设置。

实验中，我们使用爬虫系统从三大门户网站上抓取涉及科技、财经、军事、教育等八种类别的Web网页来作为测试数据，设定K=200，遗传算法的初始种群规模数为30，终止进化代数T=100，交叉概率P_c=0.6，变异概率P_m=0.05，输出八个最优个体。K均值算法中K取值为8。

比较K均值聚类算法和基于K均值和遗传算法的聚类算法的F-Measure值，如表4.1(算法1表示基于K均值和遗传算法的聚类算法，算法2表示K均值聚类算法)所示：

表4.1 两种算法的F-Measure值

算法	c_1科技	c_2财经	c_3军事	c_4教育	c_5汽车	c_6房产	c_7天气	c_8游戏
算法1	0.635	0.613	0.731	0.682	0.768	0.725	0.804	0.732
算法2	0.524	0.537	0.641	0.576	0.621	0.606	0.737	0.624

实验结果显示，基于K均值和遗传算法的聚类算法的F-Measure评价数据比K均值算法的要高，基于K均值和遗传算法的聚类算法确实能提高聚类的性能。分析其原因，基于K均值和遗传算法的聚类算法继承了遗传算法的全局优化能力，避免像K均值算法那样陷入局部极值而降低聚类效果。同时可以看出，天气类别的F-Measure值比较高，因为天气类别跟其它几种类别的相关度较少，聚类的效果也比较好。这也反映了聚类的效果依赖于聚类样本之间的关联性，聚类样本的选

择应该满足类别内的高相关度和类别间的低相关度，这样能得到更好的聚类结果。

4.4 本章小结

本章主要论述了基于 K 均值和遗传算法的聚类算法，分析了算法的基本实现技术，并通过实验进行了验证了算法的可行性。

第五章 系统设计与实现

基于以上的讨论，我们构建了 Web 文本挖掘的原型系统。下面我们将详细介绍系统的结构、设计和功能的实现。

5.1 系统框架

系统包括了中文分词模块、特征表示和提取模块、文本挖掘模块。中文分词模块主要是实现对 Web 文本进行分词，并对过滤掉一些对描述文本特征无关的常用停顿词等，收录词典中不存在的词；特征表示和提取模块基于 Web 文本的分词结果对 Web 文本进行特征的提取，将 Web 文本转化为数学模型，为挖掘模块提供挖掘的原始数据；文本挖掘模块应用各种挖掘方法对 Web 文本进行挖掘，进行 Web 文本的分类，提取感兴趣的模式。原型系统采用.NET 框架作为开发平台，开发语言采用 C#。

下面是系统的结构图：

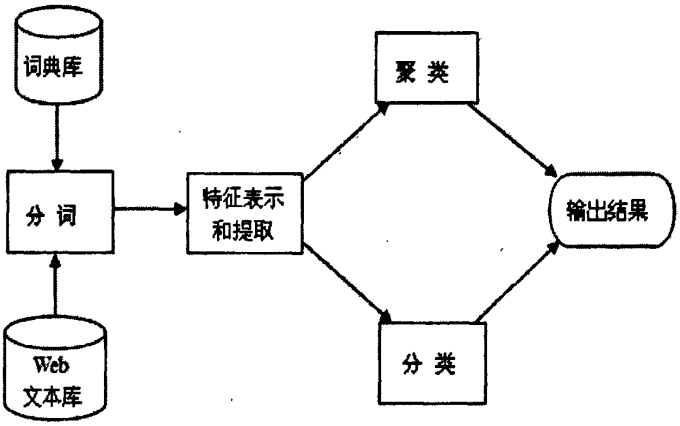

图 5.1 系统结构图

5.2 中文分词模块

中文分词模块的功能是将 Web 文本字符串切分成中文词串。中文的词是最小的、能独立活动的、有意义的语言成分。特征表示是以词作为特征向量的，就需要以词为基本单位。中文不同于英文，英语文本是小字符集上的已充分地分隔开的词串，而汉语文本是大字符集上的连续字串，它是一种无明显词间间隔的语言。我们已经讨论了中文分词的几种方法，主要有基于字符串匹配的分词、基于理解的分词和基于统计的分词等。本文实现了基于字符串匹配的分词和基于统计的分词这两种方法，基于理解的分词方法目前还处于研究阶段。

中文分词模块的主要类结构如图 5.2 所示：

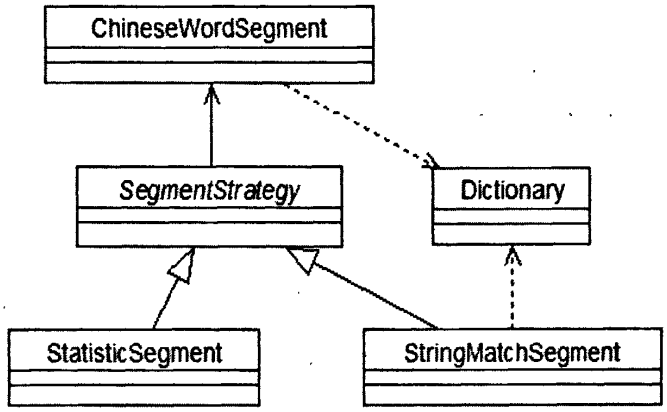

图 5.2 分词模块主要类结构图

图 5.2 中，ChineseWordSegment 类是负责对 Web 文本进行分词处理的主要类，包括数字、日期、对未收录词的判断，添加新词等，并去除一些常用助词等。系统采用策略模式对分词的方法进行处理，易于添加新的分词方法，提高了系统分词的灵活性。SegmentStrategy 类是一个抽象类，主要是对外提供分词方法的接口。StatisticSegment 类和 StringMatchSegment 类分别实现基于统计的分词方法和基于字符串匹配的方法。基于字符串匹配的方法需要维护一个词典，分词处理时以词典为基础，进行分词操作。中文分词模块中有一个词典类 Dictionary，负责加载和维护字典，在系统初始化时，可以从文件系统中加载词典，长期驻留在内存中，提高分词响应的速度。

分词方法的处理流程如下：

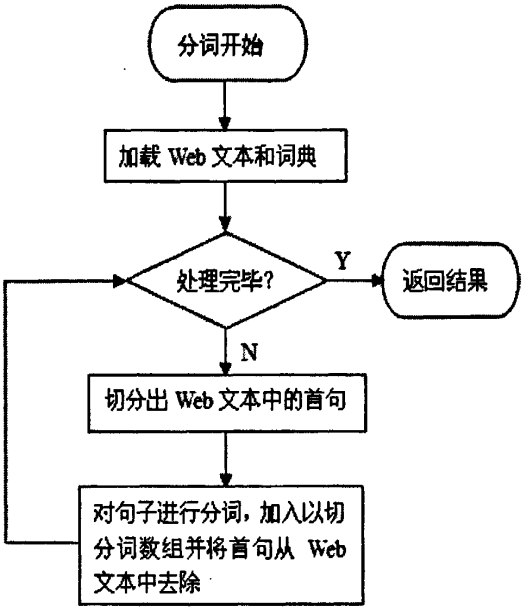

图 5.3 分词流程图

Web 文本加载的方式视其存储结构而定，一些大规模的 Web 存储都有自己自定义的文件系统，比如 GOOGLE 的文件系统。由于是原型系统，我们采用操作系统的文件存储管理系统来管理 Web 文本。系统提供了加载接口，只要符合接口的类型文件就可以对其进行分词处理，系统可以处理常用的类型文件如 Word、PPT 等。分词处理视加载不同的分词策略，其处理过程也不一样，系统提供了两种分词方法供选择。添加新的分词方法只要符合分词的抽象接口，系统就可以进行调用。常用停顿词和助词等对于表示 Web 文本基本没有贡献，但在 Web 文本中出现的频率是很高的，必须将其过滤，保证提取的特征能更好地表示 Web 文本。词典包括了常用词语、常用停顿词和助词。常用停顿词包括了中文中的一些单字，比如：的、一、不、在、人、有、是、以、上、之等，还包括了中文中的数字和阿拉伯数字和大写数字，如：零，0，1，2，3，4，一、二、三、四、十、百、千、万、亿、壹、贰、叁、肆等。

基于字符串匹配的分词方法需要维护一个“充分大”的词典，若在词典中找到某个字符串，则匹配成功。StringMatchSegment 类实现的是逆向最大匹配的方法，其它的匹配方法还有正向最小匹配，逆向最大匹配和逆向最小匹配方法。由于是原型系统，并且重在文本的挖掘，我们只实现了一种字符串匹配方法。逆向最大匹配算法的流程如图 5.4 所示：

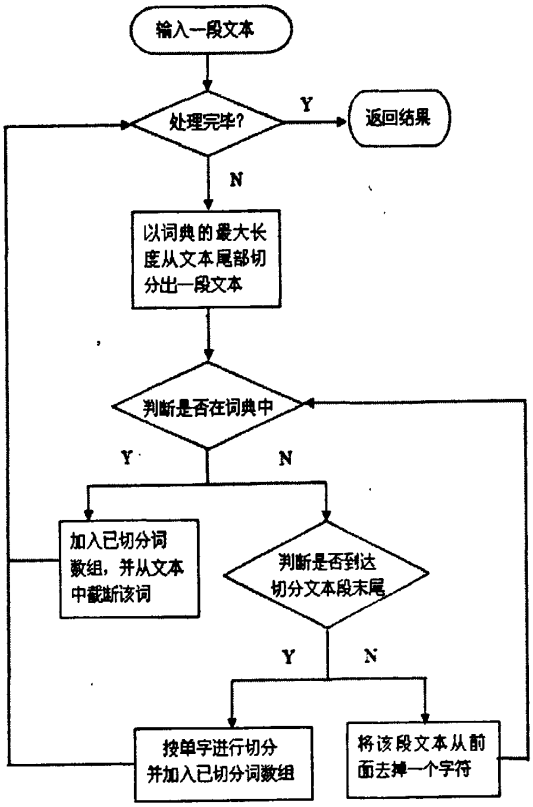

图 5.4 逆向最大匹配流程图

系统将词典加载到内存中，对于词典的查找过程，系统是基于哈希算法来实现的，哈希算法的时间复杂度接近于 O(1)，所以系统响应查找的效率是很高的。考虑到在整个系统中，词典只需要一个，所以系统只允许产生一个词典对象。添加新词的时候，不仅要更新内存中的词典，并且要更新词典文件，保持内存中词典和词典文件的同步性。

系统采用 XML 文件来存储词典数据，采用这种存储词典数据的方式主要基于以下的考虑：

1. 相比把词典数据存储在文本文件中，C#语言提供了丰富的管理 XML 数据的 API 来帮助系统管理词典数据；而与存储在数据库中比较，系统并不需要强大的并发控制和和存储引擎。系统只是需要在初始化时将词典数据加载内存中，并只是在添加新词的时候才更新词典数据，强大的存储引擎和并发控制能力对于系统来说是多余的；

2. 系统属于原型系统，词典的数据量不会太大，采用 XML 文件来存储完全能满足我们的实际需求。

采取 XML 作为词典数据存储格式的优点是很多的。XML 是 W3C 的推荐文件，作为 ISO 标准，得到几乎所有的操作系统和程序语言，具有良好的跨平台能力，可以在不同的操作系统间进行数据通信。C#和 JAVA 等高级程序语言都提供了大量的管理 XML 文件 API 对 XML 进行操作，可以采用文档对象模式(DOM-Document Object Model)标准和 SAX(Simple API for XML)这两个 XML API 来与 XML 数据进行交互，方便地访问 XML 文档的元素、属性。数据可以粒状地更新，每当一部分数据变化后，不需要重发整个结构化的数据，添加，更新和删除 XML 数据都很容易，由于平台独立性和语言的中立性。系统采用 C#语言，可以方便地对词典数据进行操作。

XML 具有良好的自描述性和可扩展性，提供对数据内容更准确的描述，可以自定义自己的标记集合来描述数据。XML 表示数据的方式是独立于系统的，并且数据能够重用。XML 文档被看作文档的数据库化和数据的文档化。我们可以随意定义词典的数据类型和结构，而不用担心不被系统支持。

XML 支持多种字符编码，包括 Unicode 双字节字符集(UTF-16)和压缩版本(UTF-8)，对中文字符编码(GB2312)的支持也很好。在 XML 文件中，我们可以随意存储词典中的中文字符，而不用担心不被应用程序支持。

XML 支持高级搜索。XML 路径语言（XPath—XML Path Language），提供了一种公用的语法和语义机制，用于 XML 内部结构寻址。查找 XML 文档内的内容更容易，因为文档内容的结构和含义是在它的语法中定义的。XML 文档的一个要求是它们必须具有严谨的格式。严谨的格式的一个特点是每个起始标记都必须具

有结束标记。另一个要求是标记必须以特定的次序嵌套。这些要求是为了便于对文件进行语法分析和操作。

词典数据的读取基于 SAX 的事件处理。因为系统在初始化时，需要将所有的词典数据读入内存，数据量相对来说还是比较大的。对于大型文件，基于 SAX 的事件处理比 DOM 基于树的处理的效果要好，因为在内存中构建一种 DOM 树的过程是比较缓慢的，SAX 则比较快。添加新词时，采用 XPath 对词典数据进行位置搜索，将新词插入到正确的位置上。

5.3 特征表示和提取模块

特征表示和提取模块对经过分词的 Web 文本进行特征表示和提取，将 Web 文本的信息描述成计算机可处理的结构化数据，对特征进行适当加权和变换，提取重要的特征，以符合 Web 挖掘算法处理的数据。

特征表示和提取的流程如下：

1. 计算经过分词的 Web 文本的词频。权值的计算采用 TF-IDF 的权值计算方法，对 Web 文本特殊标记中的词进行加权，能更好地反映 Web 文本的语义特征。

2. 将高维的向量空间进行缩减，提取最能反映 Web 文本的特征向量。提取方法可以采用文档频率、χ^2 统计量、互信息等方法，原型系统中提供了第三章介绍的几种特征提取方法。

3. 对提取的特征进行存储，原型系统采用 XML 文件来存储特征的提取结果。

特征表示和提取模块的流程图如图 5.5 所示：

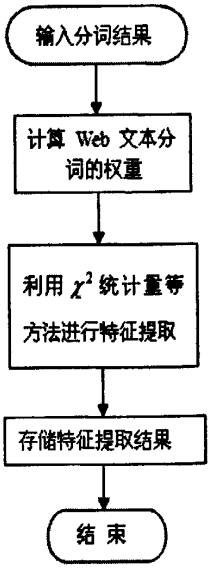

图 5.5 特征表示和提取模块流程图

特征表示和提取的主要类结构如图 5.6 所示。Extract 类是进行特征提取的主要类， 接受分词得到的结果，对分词的结果进行统计。Util 类是工具类，提供了特征权值和特征提取的计算方法。特征提取完成后，由 Extract 类负责将提取结果存储到 XML 文件中。系统提供了几种特征提取的方法，若是采用文档频率来做为提取的方法，要设定阈值。特征的提取需要预先设置特征向量空间的维数，具体的阈值和维数要根据不同的情况来确定。

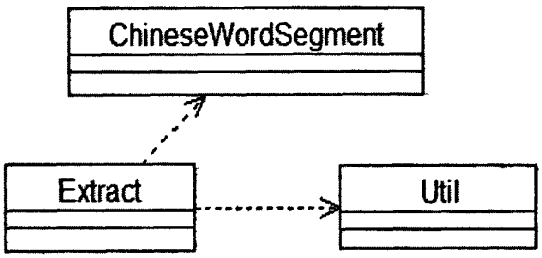

图 5.6 特征表示和提取主要类结构图

系统提供了互信息、χ^2 统计量和信息增益的几种特征提取的方法，可以根据不同的情况进行选择。这里主要介绍 χ^2 统计量方法流程：

1. 对于每个特征，计算它与一个类别的 $P(w,c)$、$P(\bar{w},\bar{c})$、$P(w,\bar{c})$、$P(\bar{w},c)$，并根据该类别的 $P(c)$ 和 $P(\bar{c})$，以及文本集合的总数，得到特征对于该类别的 χ^2 统计量值；

2. 对该特征计算它与所有类别的 χ^2 统计量值，并根据类别的 $P(c)$，计算出该特征对于文本集合的 χ^2 统计量；

3. 计算所有特征对于文本集合的 χ^2 统计量，并根据预先设定的文本集合的特征向量空间维数，得到文本集合的特征向量集。

提取出来的特征向量需要进行存储，包括特征词的存储和特征矩阵的存储。所以我们采用了 XML 文件来存储特征提取的结果。特征矩阵数据的查询非常方便，XPath 提供了 XML 内部结构寻址的方法。比如，进行文本相似度比较的时候，查询某个类别 c_i 中一个 Web 文本 d_j 的特征向量的寻址语法为：/Category [@id="i"]/ WebText[@id="j"] 。

特征矩阵在 XML 文件中的数据结构如下所示：

```
<Categories>
    <Category id="1" name="科技">
        <WebText id="0513">
```

```
        <Feature id="4286" Value="0.107636" />
        ...
    </ WebText >
    ...
</ Category>
...
</ Categories>
```

特征词在 XML 文件中的数据结构如下所示：

```
<Features>
    <Feature id="1" name="平台" Value="3.451723" />
    ...
<Features>
```

5.4 文本挖掘模块

文本挖掘模块主要是应用分类和聚类算法对 Web 文本进行分类。系统实现了 K 均值、基于 K 均值和遗传算法的两种聚类算法、KNN 算法和朴素贝叶斯算法两种分类算法。文本挖掘模块的主要类结构图如图 5.7 所示：

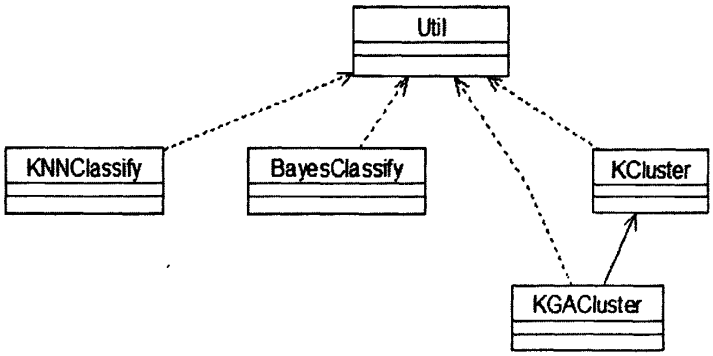

图 5.7 文本挖掘模块主要类结构图

KNNClassify 类主要实现 KNN 分类算法，需要输入一个 K 值来确定测试文本的近邻文本，利用 Util 类提供的文本相似度方法来搜索 K 个近邻。BayesClassify 类实现了朴素贝叶斯方法，包括了训练方法和分类方法。KGACluster 实现了基于 K 均值和遗传算法的聚类方法，调用 KCluster 类实现 K 均值聚类，然后进行遗传的操作，得到聚类的结果。

Web 文本的分类需要有已知样本集的分类，从已知样本中训练出分类模型，然后就可以用分类模型对文本进行分类了。已知样本集的分类要尽可能保证类内的高文本相似度而类间的低文本相似度，才能保证分类的效果。

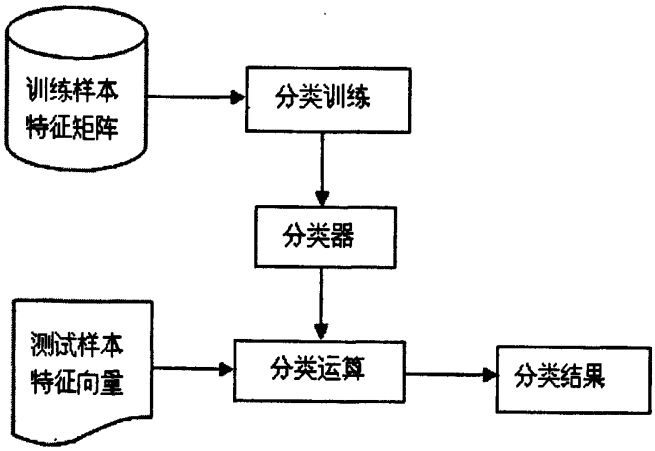

图 5.8 Web 文本分类过程图

系统采用基于多项式模型的朴素贝叶斯算法。朴素贝叶斯算法的训练流程如下：

1. 对某个特征词w_k，计算它对于训练样本集中每个类别的概率$P(w_k|c_j)$；

2. 对概率$P(w_k|c_j)$进行平滑处理；

3. 对所有特征词。进行前两步的操作，得到所有特征词对于类别的概率$P(w_k|c_j)$；

4. 计算每个类别的$P(c_j)$；

5. 将特征词对于类别的概率$P(w_k|c_j)$和类别的$P(c_j)$存储到 XML 文件中。

贝叶斯分类器训练完成后，就可以对文本进行分类了。朴素贝叶斯算法的分类流程如下：

1. 加载要进行分类的 Web 文本特征向量数据；

2. 读取样本库中某个类别c_j的先验概率$P(c_j)$；

3. 对 Web 文本的每个特征向量，读取其条件概率$P(w_k|c_j)$；

4. 根据特征向量在 Web 文本中出现的次数和该 Web 文本的总词数，计算该 Web 文本属于类别c_j的概率；

5. 对样本库中所有的类别进行前三步的操作，得到 Web 文本属于各个类别的概率；

6. 比较各概率的大小，将 Web 文本样本划分到概率最大的类别中。

Web 文本聚类功能是将 Web 文档集合分成若干个簇，使得同一簇内文档的文本相似度尽可能的大，而不同簇之间的文本相似度尽可能的小，从而实现对 Web

文档的分类。与 Web 文本分类不同，聚类没有预先定义好的类别。系统集成了两种聚类算法，前一章已经描述了基于 K 均值和遗传算法的聚类算法流程，这里主要介绍 K 均值方法的流程：

1. 输入 K 值，并随机产生 K 个聚类中心；

2. 对每个 Web 文本，计算其与 K 个聚类中心的文本相似度，将其分配到相似度最高的聚类中；

3. 对于 K 个聚类划分，计算其中 Web 文本的平均特征向量，作为新的 K 个聚类中心；

4. 重复 2、3 步，直到没有新的分配产生。

5.5 系统运行实现

Web 文本挖掘原型系统主界面如图 5.9 所示。在系统第一次启动的时候，由于没有已经分好类别的样本库，所以左侧的类别树是空的。右上方可以显示 Web 文本的内容和样本集的特征词。右下方可以显示 Web 文本的 ID、标题、所属类别和 URL。

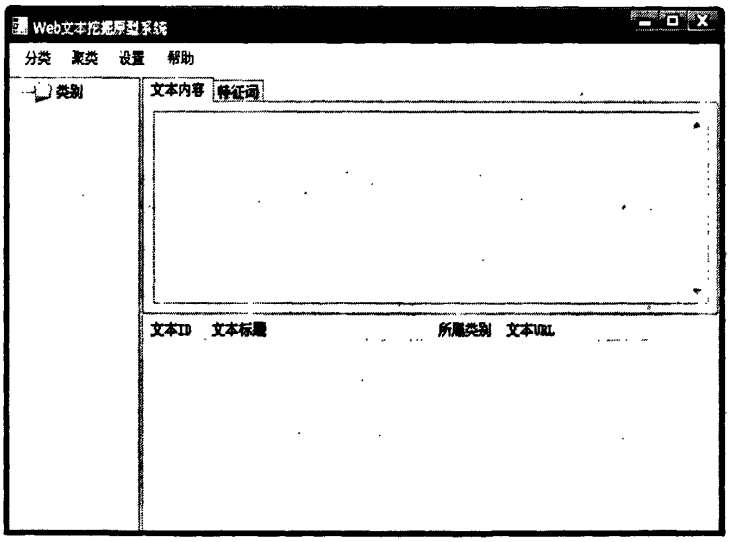

图 5.9 Web 文本挖掘系统主界面

我们将从聚类开始，展示系统的运行。首先要准备好 Web 文本集，我们从三大门户上集中下载了八种类别的网页。需要设置系统的聚类参数，系统的聚类参数界面如图 5.10 所示。可以选择 K 均值聚类方法或者基于 K 均值和遗传算法的聚类方法，如果选择基于 K 均值和遗传算法的聚类方法，必须设置 K 值、初始种群规模数等参数，指定聚类目录。

特征提取参数设置如图 5.11 所示，可以设置特征空间的维数和特征提取方法，系统提供了互信息、统计量、信息增益和文档频率四种特征选提取方法。

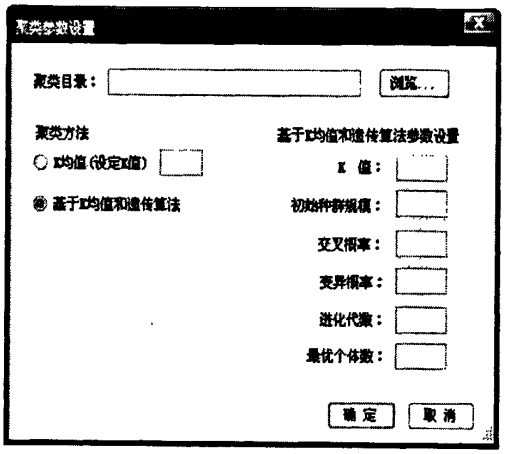

图 5.10 聚类参数图

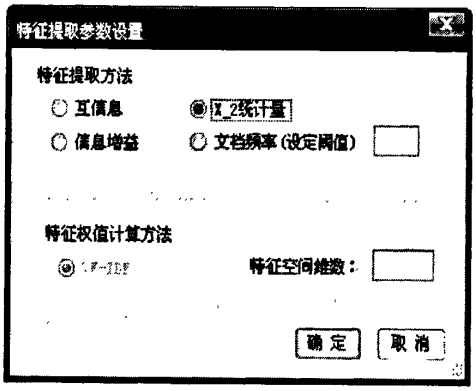

图 5.11 特征提取参数图

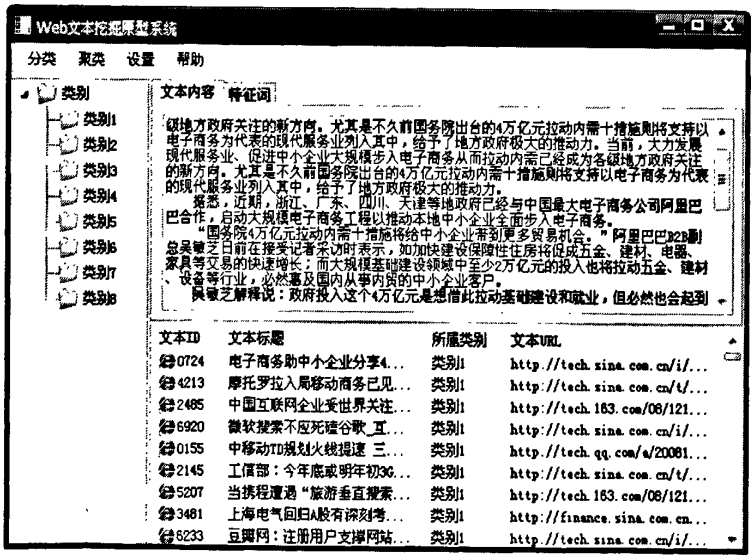

图 5.12 聚类结果图

参数设置好之后，点击聚类菜单，系统开始进行聚类操作。操作结果如图 5.12 所示。由于我们设定了聚类数目是八个，Web 文本被分成八个类别，双击可以修改类别的名称。右上角显示了类别 1 文本 ID 为 0724 的文本内容，点击右下角的

URL，系统将调用浏览器显示相应 URL 的网页。我们可以预先设置答案，系统会计算出聚类的评价结果。

进行分类训练之前必须要给出样本集。训练的参数设置如图 5.13 所示。选择训练的分类方法，并给出训练样本的目录。

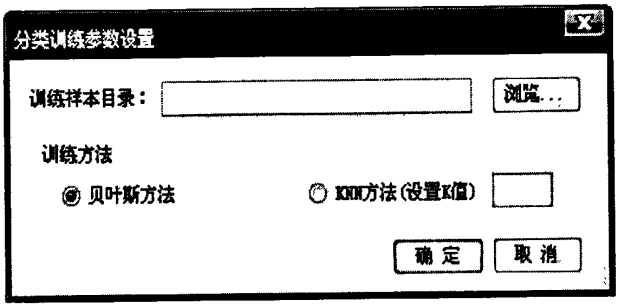

图 5.13 设置分类训练参数图

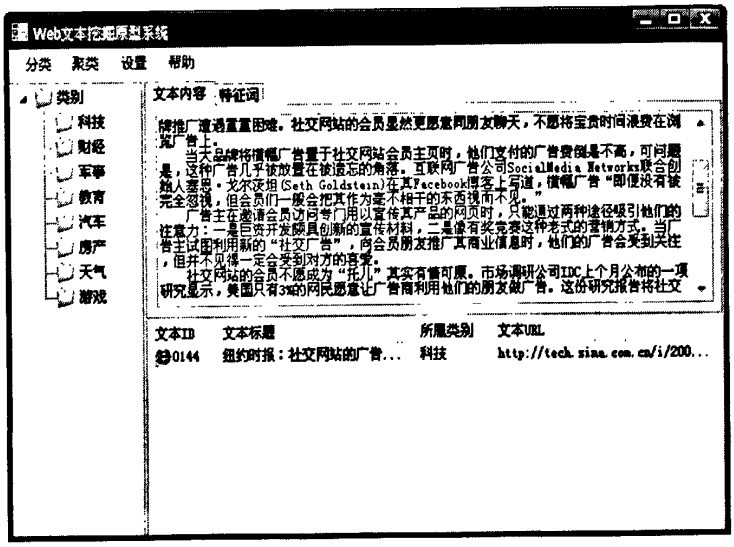

图 5.14 分类结果图

训练完成后，系统将训练结果并存储到自动创建的 XML 文件中。有了训练好的样本集，我们选择某个文本进行分类，分类的结果如图 5.14 所示。为了分析分类结果，可以对其进行评价，分类之前需要给出答案和测试样本集。系统进行分类操作，并将返回准确率、召回率和 F-Measure 值。

5.6 本章小结

本章详细描述了 Web 文本挖掘原型系统的设计和实现，通过大量的图例，展示了系统的功能原理和运行效果。以前一章介绍的理论作为基础，系统实现了中文分词、特征表示和提取、文本聚类和分类等挖掘技术。

第六章 总结与展望

6.1 研究工作总结

Internet 是一个分布广泛的全球信息服务中心，其中包含了许多有用的知识，如何利用数据挖掘能把这些有用的知识充分提取出来为用户所使用，是近几年研究的一个热点。Web 文本挖掘是一个较新的研究领域，吸引了越来越多研究人员从事这一领域的研究。目前这方面研究已取得很多成果。本文研究了 Web 文本挖掘的相关理论知识，并利用 Web 文本挖掘技术对 Web 信息进行挖掘，实现对 Web 文本的分类，详细设计并实现了具备完整功能结构的 Web 文本挖掘原型系统，重点讨论了当前流行的 Web 文本挖掘的关键技术，主要的研究工作包括：

1. 介绍 Web 文本挖掘的背景知识，分析 Web 文本挖掘的研究背景和现状，并探讨了 Web 文本挖掘的意义；

2. 详细分析和讨论了 Web 文本挖掘过程的关键技术，包括中文分词技术、权重计算方法、Web 文本特征的表示和提取方法；

3. 讨论了几种常用的 Web 文本分类和聚类的算法：KNN 近邻算法、朴素贝叶斯方法、SOM 自组织映射算法、K 均值算法，研究了 K 均值算法的理论和优缺点，在此基础上，提出基于 K 均值和遗传算法的聚类算法，并对其进行了实验，在挖掘评价方法的基础上，验证该算法的可行性。

基于以上的研究成果，本文设计并实现了 Web 文本挖掘原型系统。

6.2 未来展望

随着互联网在各个领域的发展，Web 文本内容也呈现出多样性的趋势，Web 挖掘的范围会随着互联网技术的发展而有新的拓展和延伸，对 Web 文本挖掘技术提出更高的挑战性。未来的互联网是多媒体数据的时代，越来越多的图像、声音等非文本文档将会成为主导用户使用的主要资源，目前的 Web 挖掘技术主要是针对文本数据进行知识的提取，基于音频、视频的内容挖掘是未来挖掘的方向。

迄今为止， Web 文本挖掘算法，都没有标准的训练集和测试集，导致专家和学者们提出的挖掘算法都是基于不同的文本数据集，不具备可比性也不利于相互学习和交流。因此，研究建立标准的文本训练集和测试集，这样显然可以加快整个 Web 挖掘领域的发展。

基于自然语言理解的方法，如句法分析，还没有深入应用到 Web 文本挖掘上来。将自然语言理解、语义概念和 Agent 技术等应用到分词技术、分类挖掘等领

域中，提高文本挖掘的智能化水平是需要我们进一步研究的问题。

致谢

本文的研究工作得到了导师刘志镜教授的悉心指导和关怀。刘老师渊博的知识、严谨的治学态度、开阔的视野，极大地启发了我的研究工作。刘老师在理论知识和科研实践上给予了我很多帮助，在研究方向和研究思路方法上的指导，尤其使我受益匪浅。从论文的选题、研究到撰写，倾注了刘老师极大的心血。从论文选题开始，刘老师就在研究方向上给予了有效的指导；论文研究阶段，刘老师的言传身教更起到了不可估量的作用；论文撰写阶段，刘老师认真负责，对论文从始至终严格把关，对于论文的顺利完成给予了极大的帮助。刘老师忘我的工作精神和雷厉风行的作风将使我终身受益。研究生学习阶段，刘老师在生活方面也给予了极大的关心。在硕士研究生生活即将结束之际，谨向刘老师致以深深的敬意。

还要感谢姚勇博士，两年多来给予我的研究很多帮助和支持。还有实验室所有和我朝夕相处的同学，他们在学习上和生活上给我提供了许多帮助和支持，他们的深厚情谊我将铭记在心。

尤其要感谢我的父母和家人，为我的成长花费了巨大的心血和精力。在我多年的求学生涯，给我默默的支持和鼓励。这份亲情是我毕生都难以回报的。

参考文献

[1] Pranam Kolari, Anupam Joshi, Web-Mining: Research and Practice, Computing in Science & Engineering, July/August 2004

[2] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In Proceedings of the Seventh International World Wide Web Conference, 1998.

[3] Clarke, I. A distributed decentralized information storage and retrieval system. Master's thesis, University of Edinburgh, 1999.

[4] XindongWu, Data Mining: Artificial Intelligence in Data Analysis, Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'04)

[5] Gerard Salton. Automatic Text Processing. Addison-Wesley, Reading, Mass., 1989.

[6] W. Cheung, V. T. Ng, A. W. Fu, and Y. Fu. Efficient Mining of Association Rules in Distributed Databases. IEEE Transactions On Knowledge And Data Engineering, 8:911-922, 1996.

[7] Google Press Release: http://www.google.com/press/overview.html

[8] 侯汉清. 分类法的发展趋势简论，北京: 中国人民大学出版社，1981.

[9] 姚毓才,王本年. 数据挖掘工具的分类与挖掘. 计算机技术与发展. 16(8), 2005:6-9.

[10] Han J, Kamber M. Data Mining: Concepts and Techniques. Sam Francisco, CA: Morgan Kaufmann, 2001.

[11] Gregory Pistetsky Shapiro etc. From Data Mining to Knowledge Discovery: An Overview. Advances in Knowledge Discovery and Data Mining. Menlo Park: AAAI/MIT Press, 1996:1-35.

[12] 范明, 孟小峰. 数据挖掘: 概念与技术及其应用. 背景: 国防工业出版社, 2001.6.

[13] 刘同明. 数据挖掘技术及其应用. 北京: 国防工业出版社，2001.

[14] 周梦麟, 张森. 一种基于自然语言理解的 Web 挖掘模型, 浙江工业大学学报, 第 32 卷 第 1 期

[15] 汪自强, 余以胜. 一种基于知识的检索方法研究, 情报杂志, 2004 年第 1 期

[16] S. Chakrabarti, M. van den Berg, and B. Dom. Focused crawling: A new approach to topic-specific web resource discovery. In The 8th International World Wide Web Conference, 1999.

[17] D. Eichmann. The RBSE spider: Balancing effective search against web load. In

Proceedings of the First World Wide Web Conference, 1994.

[18] J. Cho, H. Garcia-Molina, and L. Page. Efficient crawling through URL ordering. Computers networks and ISDN systems, 30:161–172, 1998.

[19] Caragea, A. Silvescu, and V. Honavar. A Framework for Learning from Distributed Data Using Sufficient Statistics and its Application to Learning Decision Trees. International Journal of Hybrid Intelligent Systems., 2003.

[20] 高飞, 谢维信. 互联网上的数据挖掘. 计算机科学. 2001, 28(5) . 81～84.

[21] 金玮, 张克君, 曲文龙等. 分布式 Web 用户兴趣迁移模式挖掘研究. 计算机工程. 2006, 12, 32(24): 44～47.

[22] 王实, 高文. 李锦涛. Web 数据挖掘. 计算机科学. 2000, 27(4). 28～31.

[23] Ronen Feldman and Ido Dagan. KDT-Knowledge Discovery in textual Databases. In Proceedings of the 1st Annual Conference on Knowledge Discovery and Mining, 1995. 112-117.

[24] 傅赛香, 袁鼎荣, 黄柏雄, 钟智. 基于统计的无词典分词方法. 广西科学院学报 2002, 4, 252-256.

[25] Jian-YunNie, FujlRen. Chinese information Retrieval: using characters or words. Information Processing and Management 35, 1999:443-462.

[26] 陈治平, 林亚平. 基于N层向量空间模型的信息检索运算, 计算机研究与发展, 2002, 10-12.

[27] Robertson and S.Walker. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. ACM SIGIR'94, Dublin, Ireland, 1994.

[28] Murata, Q.Ma, K.Uchimoto, H.Ozaku, M.Utiyama, and H.Isahara. Japanese Probabilistic Information Retrieval Using Location and Category Information. IRAL'2000, pp.81-88, Hong Kong, 2000.

[29] Yang Yiming and Pederson J.O. A comparative study on feature selection in text categorization. In Proc.of ICML'97,412-420.

[30] Kenneth, Ward Church and Patrick Hanks. Word Association norms, mutual information and lexicography[C]. In Proceedings of ACL27, pages 76.83, Vancouver, Canada,1989.

[31] H.Schutze, D.A.Hull, J.O.Pedersen. A comparison of classifiers and document representations for the routing problem. In Proceedings of SIGIR.95, 18'h ACM International Conference on Research and Development in Information Retrieval,229-237, 1995.

[32] 王继成, 潘金贵. Web 文本挖掘技术研究[J]. 计算机研究与发展, 2000.

Vol.37(5):513-520.

[33] F.Sebastiani, A.Sperduti, N.Valdambrini. An Improved Boosting Algorithm and Application to Text Categorization, Machine Learning, 2001.

[34] Soumen Chakrabarti, Byron Dom, Rakesh Agrawal, Prabbakar Raghavan. Using Taxonomy, discriminants, andsignature for navigating in text databases. In Proceeding of the 23rd VLDB Conference, 1997.446-455.

[35] Andrew McCallum and Kamal Nigam. A comparison of event models for naïve bayes text categorization, AAAI-98 Workshop on Learning for Text Categorization, 1998, 129-138.

[36] P. Willet. Recent trends in hierarchical document clustering: A critical review Information Processing & Management. 24(5),1998.577-597.

[37] Wang Ke and Liu Huiqing. Schema discovery from semi-structured data. In Proceeding of the 1st International Conference on Knowledge Discovery and Data Mining. Newport Beach, 1997.

[38] 蔡子兴, 徐光祐. 人工智能及其应用. 3. 北京: 清华大学出版社. 2004. 152-165.

[39] 周明, 孙树栋. 遗传算法原理及应用. 北京: 国防工业出版社. 1999.6. 32

[40] Michalewicz Z, et. al. A Modified Genetic Algorithm for Optimal Control Problems. Computers Math. Application, 1992,23(12)83-94

在读期间发表的学术论文

[1] Jun Zhou and Zhijing Liu. Distributed Clustering Based on K-means and CPGA. IEEE FSKD'08. **(Indexed by EI)**. Published.