密级_____

博士学位论文

中文自动文摘关键技术研究及应用

蒋效宇

导师姓名(职称) 樊孝忠教授 答辩委员会主席 林守勋教授

申请学科门类 工 学 论文答辩日期 2009-6-10

申请学位专业 计算机应用技术

2009 年 06 月 08 日

研究成果声明

本人郑重声明：所提交的学位论文是我本人在指导教师的指导下进行的研究工作获得的研究成果。尽我所知，文中除特别标注和致谢的地方外，学位论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京理工大学或其它教育机构的学位或证书所使用过的材料。与我一同工作的合作者对此研究工作所做的任何贡献均已在学位论文中作了明确的说明并表示了谢意。

特此申明。

签名：　　日期：

关于学位论文使用权的说明

本人完全了解北京理工大学有关保管、使用学位论文的规定，其中包括：①学校有权保管、并向有关部门送交学位论文的原件与复印件；②学校可以采用影印、缩印或其它复制手段复制并保存学位论文；③学校可允许学位论文被查阅或借阅；④学校可以学术交流为目的，复制赠送和交换学位论文；⑤学校可以公布学位论文的全部或部分内容（保密学位论文在解密后遵守此规定）。

签　　名：　　日期：

导师签名：　　日期：

摘 要

随着网络的日益普及，在线信息急剧增加，如何有效地获取和描述这些文本信息显得越来越重要。尽管用户通过搜索引擎可以快速获得丰富的文档，但要获取其中内容则需要耗费大量时间去阅读每一篇文档。单文档自动文摘能够为用户提供一个原文档的压缩版本，旨在减轻用户的阅读压力；而多文档自动文摘是从多篇文档中提取主要的或用户需要的信息，其在信息检索中的重要地位使其成为自然语言处理领域的一个研究热点。

本论文针对中文文档的特点，围绕自动文摘系统实现和应用过程中的关键词抽取、局部主题的识别、文摘句润色处理、文摘句排序策略和基于用户查询的文摘生成等关键技术开展了一系列研究，主要取得了以下几个方面的成果：

⑴提出了基于词汇链构建的关键词抽取的算法。在利用相邻词共现进行未登录词识别的基础上，给出了利用《知网》为知识库构建词汇链的算法。通过计算词义相似度构建词汇链，然后结合词汇所在词汇链的强度、信息熵和出现位置等属性进行关键词抽取并生成文摘。该方法考虑了术语识别和词汇之间的语义信息，能够明显改善关键词抽取的效果。实验结果表明，该算法在精确匹配测试和近似匹配测试中均取得比词频区域法更好的抽取效果。

⑵提出了基于主题聚类和语义分析的多文档文摘系统(CSA-MDSS)的体系结构，通过对局部主题个数的自动探测，采用 K-均值聚类算法对句子进行聚类，形成多文档集合的局部主题，并在局部主题确定的基础上，给出了一个重要性抽取和平均抽取相结合的文摘句抽取的方法。在文摘句抽取的基础上，提出了改进的 MMR-SS 句子冗余消除算法和基于语义分析的文摘平滑处理和连贯性处理的方法。实验结果表明该方法在信息覆盖方面要优于 MEAD 和 TOP-N，从而说明 CSA-MDSS 多文档文摘系统在文摘内容的平衡性及信息覆盖上具有很大的优势。

⑶提出了一种将局部主题间的内聚度融入多文档文摘句排序的改进 MO 算法。在分析了 MO 和 CO 两种常用的多文档文摘句排序算法的算法思想以及它们的不足之处的基础上，首先建立局部主题之间的关系有向图并计算其相互间的内聚度；然后在排序过程中每从有向图中输出一个顶点时，便从剩余顶点中查找与其具有最大内聚度的顶点，若该内聚度大于阈值，则将这两个顶点所代表的局部主题文摘句置于摘要中相邻的位置。自动评价和人工评价的实验结果表明该算法排序生成的文摘更具有连贯性

和可读性。

(4)提出了一种多特征融合的句子权重计算和查询文摘生成的方法。为了弥补搜索引擎返回网页信息量不足的缺陷，进一步提高网页相关性判断的速度和准确率，在对查询输入进行短语识别和网页结构分析的基础上，将关键词短语、网页结构等启发式规则信息融入到句子重要度计算，并分析句子与查询输入的关联特征，使文摘内容能与用户需求一致，实现文摘信息的内聚性和全面性。对比实验结果表明，此方法生成的查询摘要在网页相关性判断的速度和准确率等方面均优于现有方法。

最后，基于文档摘要在一定程度上覆盖了原文档中所有的重要信息和可以将摘要作为原文档的某种替代的假设，本文提出了采用自动文档摘要进行特征选择和分类器训练的文本分类方法。实验结果表明，文档摘要能够保留原文档中相对重要的特征，去除了大量不利于分类的噪音，大大降低了特征选择和分类的计算量，提高了分类的速度和性能。

关键词：自然语言处理 自动文摘 未登录词识别 关键词抽取 句子抽取 文摘句排序 相关性判断 特征选择 文本分类

Abstract

With the rapid growth of online information, it becomes more and more important to find and describe textual information effectively. Although it is convenient for users to obtain a great deal of documents with a search engine, users have to take the tedious burden of reading all those text documents. Automatic text summarization can alleviate users' browsing burden by providing users with a condensed version of the original text; while multi-document summarization, aiming at extracting major or user-interested information from the given multiple documents, which plays a vital role in Information Retrieval (IR), has become a hot topic in Natural Language Processing (NLP).

According to the characteristic of Chinese word segmentation, this paper does a series research which revolves keyword extraction, local topic recognition, sentences' polish processing, sentence ordering and the generation of query-biased summary etc. , in the implement process of automatic summarization system and its application. The main innovative achievements are as follows:

(1) A method to extract keywords based on lexical chain is brought forward. Based on the unknown words recognition using co-occurrence of neighbor words, an algorithm for constructing lexical chains based on HOWNET knowledge database is given in this method, lexical chains are firstly constructing by calculating the semantic similarity between terms, and then keywords are extracted according to the lexical chain's intensity, the terms' entropy and position. Unknown word recognition and semantic information between terms are considered in this method, which can significantly improve the effectiveness of keywords extraction. Experimental results show that the improved method gets better performance than other methods both in accurate-matching test and approximate-matching test.

(2) Architecture of CSA-MDSS (Multi-Document Summarization System Based on Clustering and Semantic Analysis) is given. Through the automatic detection

of the number of themes, K-means clustering algorithm is adopted to cluster all sentences from for multiple documents and to form a collection of local themes, and a candidate sentences extraction method is proposed which combine the most sentences from the head and tail themes and the average sentences from other themes. An improved MMR-SS algorithm is proposed to reduce the redundancy and to smooth the summarization based on semantic analysis. Experimental results show the improved algorithm performs better than MEAN and TOP-N algorithms, and shows that CSA-MDSS has certain advantages in the balance of content and the coverage of information.

(3) A new ordering algorithm is proposed which combines the mutual cohesion among themes and the Majority Ordering method. Based on the theory and shortcomings of Chronological Ordering and Majority Ordering are analyzed , a directed graph of the themes is built and the corresponding mutual cohesion is computed based on the statistical data about the relative position in each pair of themes. In the ordering process, when a vertex is output from the directed graph, the vertex possessing the greatest cohesion with the vertex is searched from the remaining vertexes. If the cohesion is bigger than the threshold value, the sentences from the two themes corresponding to the two above-mentioned vertexes are placed on adjacent locations in the summarization. Experimental results show that summarization generated by the proposed ordering algorithm is more coherent and more readable.

(4) A new sentence scoring method based on multi-feature integration is proposed. In order to overcome the shortcomings of insufficient information return by traditional search engines and improve the speed and accuracy of relevance judgment, query phrases are recognized from query using Mutual Information (MI) between keywords and the structure of web pages is first analyzed; and then query phrases and the structure information of web pages and so on heuristic rules are incorporated to the weight of sentences; finally, in order to let the query-biased summary be consistent with the needs of users, the associated characteristics between input query and each sentence are

analyzed. Experimental results show that the improved algorithm performs better in the speed and accuracy of relevance judgment than the existing methods.

Finally, since summary may cover all important information and be an alternative of original document, based on the above assumptions, using summarization to select features and classifier training is proposed in this paper. Experimental results show that summarization can retain almost all important features of original document, and rule out a large number of noise which is not conducive to classify, thus, the improved method can not only reduce the calculation of feature selection and categorization, but also increase the speed and performance of text categorization.

Key Words: natural language processing; automatic summarization; unknown words recognition; keyword extraction; sentence extraction; sentence ordering; relevance judgment; feature selection; text classification

目　录

图索引

表索引

第1章 绪论

互联网(Internet)为人们提供了一个便捷的信息获取渠道的同时，也留下了一个难题，那就是如何从不断涌现的海量信息中快速准确地获取自己感兴趣的信息。如果能够阅读到代表全文中心思想的摘要，就能够极大地提高获取信息的速度。自动文摘(Auto-Summarization)是从一篇或者多篇文档中提取主要的或用户需要的信息，其在信息检索(IR，Information Retrieval）中的重要地位使其成为自然语言处理(NLP，Natural Language Processing)领域的一个研究热点。

1.1 研究背景

自然语言处理是指利用计算机通过可计算的方法对自然语言的各级语言单位进行转换、传输、存储和分析等加工处理[1-2]，是与语言学、计算机科学、信息论和心理学等相关联的交叉性学科，并且与自然科学和社会科学的许多主要学科都有着广泛的联系[3-4]。具体地讲，与自然语言处理技术密切相关的学科有智能化人机接口、自然语言理解(Natural Language Understanding) 、计算语言学(Computational Linguistics)等。其中，智能化人机接口侧重于语言信息处理的应用研究，即应用语言处理技术改善人机交互的方式、手段和途径；自然语言理解是人工智能领域的一个重要分支，其研究重点侧重于对经过深度加工的语言信息的理解，相当于语言处理技术在高级语言单位上的应用基础研究；计算语言学是现代语言学的一个分支，它是利用计算机理解、处理和生成自然语言，即它的研究范围不仅包括语言信息的处理，还涵盖语言信息的理解和生成。

摘要作为从众多信息源中摘录出重要内容的一门艺术，已经成为人们日常生活中一个必不可少的组成部分，它可以帮助人们花更少的时间获得更多的有用信息。文档摘要是以提供文档梗概为主要目的，不做任何评论和补充说明，简洁准确地描述文档主要内容的短文[5]。手工摘要已经被广泛应用于科技文献、小说和新闻等领域，这些文摘的生成通常都要通过原文作者或专业的文摘人员来完成，然而，随着科技的发展和社会的进步，各种期刊会议和文献资料的增长速度远远快于手工摘要的处理速度，互联网的日益兴起使得人工摘要更是无以应对。

除了文摘生成速度方面的局限性之外，手工文摘还存在以下两点不足：(1) 手工文摘的过程在很大程度上受到文摘人员个人兴趣和知识背景的影响；(2) 手工文摘很

难满足基于用户请求或者面向特定任务的文摘任务。例如：在 IR 过程中不同的用户会用不同的关键字查询他所关心的信息，这就需要在撰写摘要时必须充分考虑到用户的查询请求，但是用户的请求往往是很难事先预知的，而且文摘人员感兴趣的话题不一定与读者感兴趣的话题一致，从而降低了手工摘要的实用性。

自动文摘技术是自然语言处理 NLP 领域的一个难点，目前所取得的研究成果还不像其它自然语言处理技术那样成熟并得到广泛应用，这主要是因为，为一篇文章或多篇文章撰写文摘，首先必须真正理解原文的内容，这一任务对人来说是很容易完成的，但是对计算机却显得异常困难。因为计算机要处理自然语言，必须具备与人一样的理解语言的有关知识和能力，其中句子分析与理解是最为关键的一个环节，尽管近些年来句法理论和语义理论有了较大的进步，但是还远远不足以让计算机完全理解自然语言，从而导致了自动文摘的整体质量还很难与人工摘要相比。

本论文就是基于以上背景，围绕自动文摘系统实现和应用过程中的关键词抽取、局部主题识别、文摘句润色处理、文摘句排序策略和基于用户查询的文摘生成等关键技术进行了深入研究和探讨。

1.2 国内外研究现状

自从 1952 年 H.P.Luhn[6]提出利用计算机进行文献压缩的思想以来，国内外众多研究人员从理论到方法上对自动文摘进行了大量研究，并取得了一系列研究成果。从单文档文摘到多文档文摘、从基于句子抽取的机械式文摘到基于语义分析的理解式文摘、从平面文档文摘到 WEB 文档文摘、从不分主题的文摘到基于主题划分的文摘、从英文文摘到多语言文摘，自动文摘技术结合了多个学科的研究成果，被广泛应用于搜索引擎(Search Engine)、自动问答(Question-Answering)等多个领域。

1.2.1 国外自动文摘的研究现状

考察国外自动文摘研究发展的历程，归纳起来可以分成3个主要的发展阶段[7]。第一阶段是从20世纪50年代末至60年代末。在这一阶段，单纯的基于文档浅层特征的统计学方法占据了研究的主导地位。

1958年，H.P.Luhn开创性地发表了世界上第一篇关于计算机自动编制摘要的论文“The Automatic Creation of Literary Abstracts”，该文中他将一篇文章的词汇分为两大类：内容词和通用词。通用词包括连接词、代词、冠词、介词、助动词，以及某些副

词和形容词，除此以外的词汇都为内容词。通用词的重要性被指定为0，词频超过预先设定阈值的内容词被认为是可以代表文章主题的关键词。通过统计文章中的关键词的词频来计算它们的权重，并利用句子中包含的所有关键词的权重来给各个句子打分，从中挑选出得分最高的若干句子构成摘要。

1969年，H.P.Edmundson[8]在Luhn提出的基于关键词频率统计的自动文摘方法的基础上，进一步提出了一个的改进设想，即将文档的标题、位置、关键词以及提示词这4种浅层特征联合起来考虑，并通过对它们的综合统计来为每个句子计算权重，这个权重就作为句子重要性的度量值。实验结果表明位置、标题和提示词综合加权策略取得了最好的摘要效果，而单纯使用关键词加权的方法生成的文摘效果最差。

第二阶段是从20世纪70年代初至80年代末。在这一阶段，以人工智能、知识工程以及自然语言理解为代表的方法逐渐成为主流。

1979年，耶鲁大学的Dejong[9]研制出了著名的FRUMP自动文摘系统。该系统利用语法知识来判定某个预期词在句子中的位置，并通过句法分析来遍历整个文档以寻找标记为已知脚本的短语，从而建立起各种故事的梗概。

1982年，J.I.Tait[10]对原有的FRUMP系统进行了改进，他提出首先将所有的资料先转换成概念依存结构；然后在此基础上通过分析、推测各种信息间的关系来构成摘要。

1982年，意大利Udine大学的Daniio FUM[11]成功开发出了自动文摘系统SUSY。该系统以一阶谓词逻辑作为文档的表达形式，利用概要产生器和分析缩写器来生成出满足特定需求的文摘。

1988年，德国康斯坦大学的Hahn[12]研制出TOPIC自动文摘系统，该系统针对的是微处理器领域的科技文档，它采用框架作为知识的载体，并通过联合语法、语义分析来生成各种长度的文摘。

1989年，美国GE研发中心研究员L.F.Rau博士[13]研发出了自动文摘系统SCISOR。该系统利用篇章主题分析以及复杂的句法结构分析等技术生成与文摘有关的框架，并采用某种预期驱动分析器从所有框架中提取出预期内容构成文摘。

第三阶段是从20世纪90年代初至今。在这一阶段，各种新颖的自动文摘研究思想、研究成果层出不穷。总的来说，占主导地位的研究方法又逐渐回归到以统计学方法为主、以深层次自然语言处理、信息抽取（Information Extraction）以及基于本体（Ontology）的知识工程方法为辅的混和型方法上[14]。

1994年，G.Salton[15]通过统计文档段落之间的共享单词的数目来计算段落之间的

语义相似度，构造文档的篇章结构图来辅助文档话语结构的自动分析，从而提出了基于篇章话语结构分析的自动文摘方法。

1995年，Kupiec[16]和Radev[17]将机器学习技术用于自动文摘领域。他们采用基于朴素贝叶斯理论(NB，Naïve Bayes)的机器学习方法，从以科技文献和文献摘要构成的语料库中提取出对抽取重要句子有贡献的联合特征，并在此基础上充分利用已获得的联合特征来从科技文献中抽取指定数量的句子构成文摘。

1998年，德国Endres-Niggemeyer从认知学角度开发了SimSum[18]文摘系统，美国哥伦比亚大学Radev D.R教授开发的SUMMONS[19]系统是从多个在线资源中提取相关报道进行比较，指出这些报道的一致性和矛盾点等特征并生成摘要。

2001年，Lin 和Hovy[20]尝试了用机器学习(Machine Learning)方法验证句子位置这一自然语言处理领域惯用的启发式规则对文摘句选取质量的影响。

2004年，Yi-hong Gong[21]和Xin Liu[22]两位研究人员提出了2种基于抽取的自动文摘方法：基于潜在语义分析(LSA，Latent Semantic Analysis)算法和基于相关性度量策略。基于LSA的文摘方法首先对“句子—词语”矩阵做SVD分解(Singular Value Decomposition)；然后将分解结果矩阵的对角线上若干最大特征值所对应的句子入选最终的摘要。基于相关性度量的方法挑选文摘句的策略是：首先计算每个句子和文档之间的语义相似度，从中挑选出相似度最大的那个句子放入摘要；然后从剩余的句子集合中依次去掉已包含在入选摘要的句子中的所有单词，再通过重新计算剩余句子和文档之间的相似度来选择出下一个具有最大相似度的句子入选最终的摘要。

2006年，Conroy和Oleary[23-24]尝试将隐马尔可夫模型(HMM，Hidden Markov Models)引入自动抽取型摘要的研究当中。

1.2.2 国内自动文摘的研究现状

我国对中文自动文摘的研究起步较晚，随着计算机在我国的普及，以及网络时代对信息处理的需求，中文自动文摘的研究是在20世纪80年代末才如火如荼地发展起来。取得一定科研成果的单位主要有中国科学院计算所、上海交通大学、哈尔滨工业大学、复旦大学、北京大学、北京邮电大学、山西大学和北京理工大学等。

1988年，上海交通大学王永成教授成功研制了汉语文献自动编制文摘实验系统SJTUCAA，该系统对从1983年第1期的《情报学报》上随机抽取的30多篇论文自动编制文摘，大多数文摘达到了比较满意的效果。随后，王永成教授又领导开发了“中文

文献自动摘要系统CASES”和“OA中文文献自动摘要系统”，这两个系统均集成了指示短语法、标题法、关键词法等多种方法，是一个实用的系统[25]。

2000年，王永成教授的学生史磊博士设计实现了中英文自动文摘系统AAS-CE，该系统研究了大规模语料库技术和汉语词库组织方法，并对主语与汉语话题的关系进行了研究[26]。

1990年，北京大学马希文教授对英文自动文摘进行了研究，并研制了一套实验系统EAAS（English Automatic Abstract System）。系统首先通过与用户交互获得需求信息集，然后对文档进行语法和语义分析，按照需求信息集从框架中推理出有关信息，最后生成具有一定逻辑性的文摘。

1992年，哈尔滨工业大学王开铸教授[27]研制出了基于理解的自动文摘系统MATAS，1994年研制出自动摘录式的非受限领域的自动文摘系统HIT-863，1997年提出了基于信息抽取和文本生成的自动文摘系统。

1998年，王开铸教授的学生刘挺博士进行了基于篇章多级依存结构的自动文摘研究[28]，提出了一种篇章结构的表示法——篇章多级依存结构TMDS（Text Multilevel Dependency Structure），并给出了获取TMDS、化简TMDS以及依据化简后的TMDS生成文摘的算法，但是没有给出一种很好的方法来解决如何提高文摘的简洁性，并且篇章微观结构分析的准确性还需要进一步提高。

1997年，复旦大学吴立德教授[29-30]研制的FDASCT系统首先对文档进行分词和词性标注，提取文档特征信息，然后进行词性与语义标注，随后进行词、句子、段落加权处理，最后根据权值选取句子构成文摘。

1998年，山西大学郭炳炎教授[31-32]提出了利用向量空间模型（VSM，Vector Space Model）对文档中的每个段落进行特征抽取统计，得到段落向量，计算各个段落向量间的相似度，通过确定文档段落之间内容的相关性来实现文档主题的自动分析，找出构成文章主题的各个局部主题，从这些局部主题入手来实现自动文摘。

1998年，北京邮电大学钟义信教授[33]成功研制了非受限领域复合式自动摘要系统，该系统根据词频统计、自由词标引的结果计算句子的重要性，然后运用依存关系树和语义框架法进行文摘候选句子的加工。

1999年，北京邮电大学李蕾博士为了解决领域过于受限的问题，研制出了面向特定领域的理解型中文自动文摘系统LADIES[34]。

2001年，胡舜耕博士进行了基于多Agent技术的自动文摘系统研究[35]，提出了面

向自动文摘的多Agent系统(MAS/ABS)方案。

2002年，清华大学罗振声教授[36-37]提出了基于主题概念的自动文摘方法。以概念统计和层次分析为基础，利用WORDNET作为知识库，以概念统计替代传统的词频统计，基于主题概念构建向量空间模型，计算句子重要度，并且根据主题概念在概念层次树上的分布进行文档结构分析和意义块划分，并以意义块为单元抽取文摘句。

2005年，华中师范大学何婷婷教授[38]提出了利用《知网》作为知识库，以概念统计代替特征词频统计，通过计算概念重要度抽取文摘的句子，一定程度上改善了摘要的质量。

2007年，哈尔滨工业大学王晓龙教授[39]提出了面向多文档自动文摘的多文档框架，通过系统地描述不同层面的文档单元之间的相互关系以及文档集合蕴含的事件在时间上的发生和演变，将多篇文档在不损失文档集合原有信息的前提下实现信息融合。

综上所述，国内自动文摘的研究与实现方法可以分为4类[40]：基于摘录的自动文摘、基于理解的自动文摘、基于信息抽取的自动文摘和基于话语结构的自动文摘。但这4类自动文摘的实现方法都有不尽如人意的地方。例如，基于摘录的自动文摘将原文档抽取出来的文摘句不做任何润色处理，导致生成的摘要往往缺乏连贯性，且容易出现句子冗余或重要文摘句的遗漏；基于理解的自动文摘系统的领域严格受限；基于信息抽取的自动文摘仍然受到领域限制，且生成的摘要千篇一律，十分呆板；基于话语结构的自动文摘需要分析文档的篇章结构，实现起来非常复杂。

1.3 单文档文摘的技术路线

1.3.1 基于抽取的方法

基于抽取的方法[6,8,41-47]主要借助于统计方法对文档进行浅层分析，只用到了一些浅层的语言学知识，而没有考虑更为复杂的语义、语法和语用知识。这种方法通常通过计算文档基本内容单元(如短语、词汇等)的统计相关性来找出关键单元(如句子、段落等)，并将其综合生成摘要。该方法分析阶段的核心任务是计算每个文档基本内容单元各项特征的权重，如内容单元在原文档中的出现频率、位置、是否有线索词以及其统计信息量等。这些权重的线性加权和就构成了一个内容单元的总权重。分析阶段包含一系列的频度计算操作和串匹配操作，并将结果进行加权求和，从而得到每个关键单元的权重。然后选择前 M 个最好的单元组成最终文摘(M 是由预设的文档压缩

率来确定的)。基于抽取的单文档自动文摘的体系结构如图 1.1 所示。

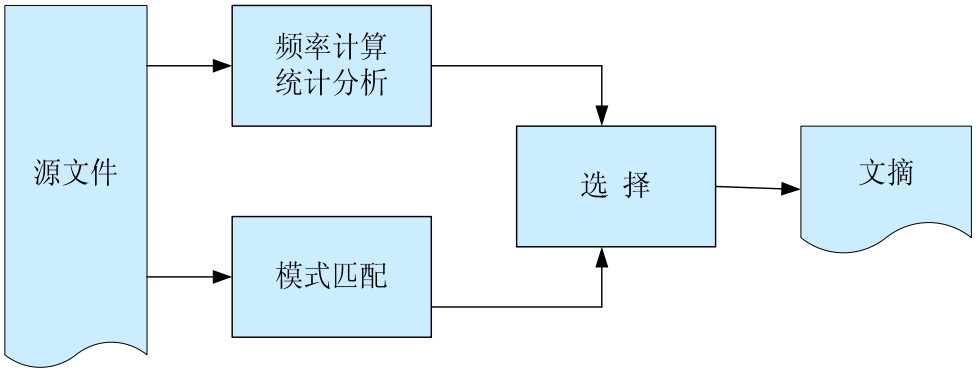

图 1.1 基于抽取的单文档自动文摘体系结构

在这种方法中，各个特征所起的作用与文档的体裁有很大的关系，因此，为区分各个特征的重要性需要确定各个特征的权重[4]。许多系统中参数的调节是采用手工调节的方法，但由于文档体裁和风格不同，导致手工调节参数的方法只能用于一些专门的领域或特定风格的文档。如果移植到其它领域或其它风格的文档上，需要重新手工调节参数。为了自动的获得这些参数，研究人员利用机器学习技术，从人工生成的理想文摘中学习所需要的参数[16]，但是这种方法依赖的两种语料(原始文档及原始文档的标准摘要)需要付出大量的劳动，具体体系结构如图 1.2 所示。

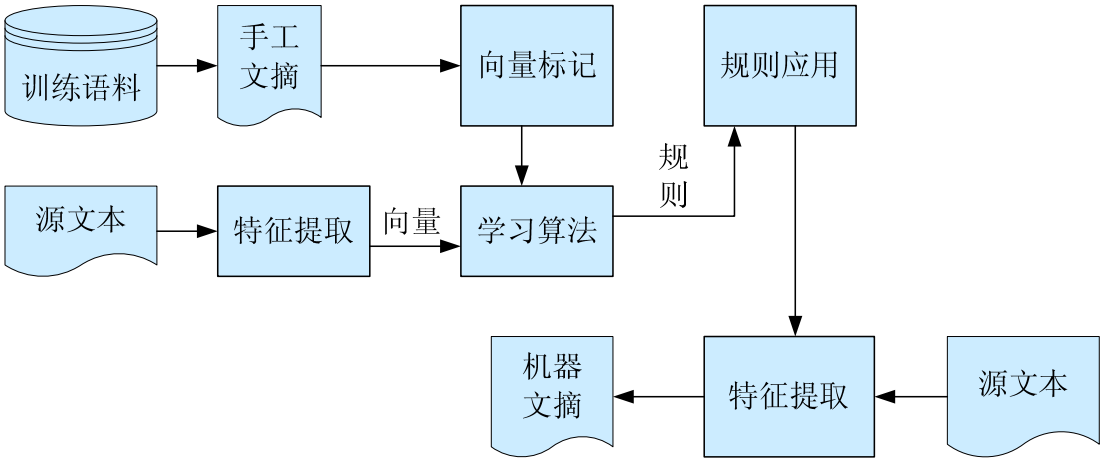

图 1.2 基于机器学习设定模型参数的抽取方法体系结构

总的来说，基于抽取方法的自动文摘技术魅力在于简单、实现容易、高效快捷。但是以句子为基本抽取单元的抽取方法没有考虑句子间的关系[48]，致使生成的文摘可读性差，甚至前后矛盾。为了解决这个问题，研究人员通常采用方法是通过开设一个滑动窗口，将被选择单元的前几个单元包括在窗口内，用于考察并解决可能存在的指代或首语重复问题。为了提高可读性，有些研究甚至直接将存在首语重复的单元从文摘中排除，或者将存在首语重复的单元的前几个单元也纳入到文摘中。如此一来，可

读性的问题看似解决了，但又带来了压缩率的问题。

1.3.2 基于理解的方法

基于抽取的方法生成的文摘只是将文档中相对重要的句子抽取出来作为文摘，这样的文摘并不能明确表述出文章作者的真实意图。如果自动文摘系统能够从理解文章内容角度抽取出作者的观点、意图或情感，例如从某人对减免汽车购置税的评论中总结出她对该政策的看法是“她支持这个政策”，给人的直觉是这样的文摘才算得上是真正意义上的文摘。

为了在摘要中体现出文章作者的真实情感，基于理解的方法需要更深层次的自然语言处理(NLP，Natural Language Processing)机制，如用于句子分析与生成的语法和词法、及分析过程中的领域知识和领域本体等。图 1.3 是基于理解的自动文摘系统的体系结构。

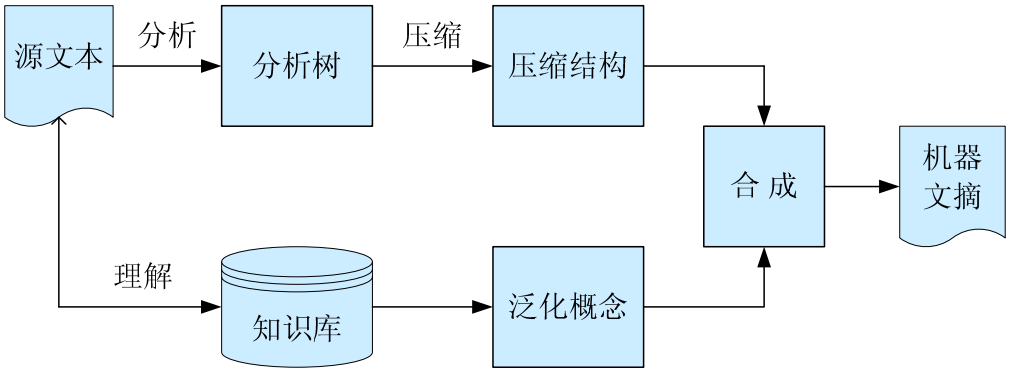

图 1.3 基于理解的自动文摘系统体系结构

基于理解的方法通常有两种途径：概念结构的压缩和形式结构的压缩(句法树或话语树的裁剪)。

形式结构的压缩是用传统的语言学方法对句子进行句法分析和句法树的语义标注，紧跟着的压缩过程是对句法树进行裁剪和重组，如根据一定的结构准则裁剪句法树中的子树如插入语、从句等。裁剪后的句法树就只保留了一些必须的结构，从而达到简化原文的目的。

在撰写文章时，作者一般会赋予文章一定的结构，如果文章结构能够搞清楚，那中心思想的位置就很容易找到。与句法树相比，话语树具有更大的粒度，与通过句法分析可得到句法树一样，通过话语分析可得到每篇文档的话语树。通过关联词的匹配规则和一些启以式规则，可以分析出一篇文档中各文本单元间的关系，如并列、递进、解释、选择、对立、补充、充分、必要和转折等，在此基础上构建话语树，并计算树

中节点间的关联程度，保留核心节点及与核心节点关系最密切的一些节点，即可得到文摘话语树。

概念结构的压缩则是以人工智能为基础，将重点放在了自然语言的理解上[49]。语法分析依然是分析过程中必不可少的部分，只是其结果不再是一个句法树，而是原文档的概念结构，用于构造文档的知识库[50]。这种结构可以是语义网络[51]、谓词公式[52]和一系列的模板或框架[53]等。转化阶段通过将冗余的信息丢弃或者对概念子图的裁剪，达到简化原文档的目的。另外，还可以通过对多段或多个模板进行信息融合，或者通过信息的泛化来进行更进一步的压缩或抽象。转化阶段的结果是产生文摘的概念表示结构，因此，这种途径的本质就是对概念的压缩。

基于理解方法的合成阶段是将概念结构或语法结构表示的文档转化成流畅的文档摘要。有些系统没有生成步骤，它将原文档中与压缩有联系的单元提供给用户，让用户通过一些点击操作直接参与压缩的过程。

与基于抽取的方法相比，基于理解的方法可以生成更为简练的摘要。由于这样的摘要是以原文档内容的形式化表达为基础的，所以更适合于个人数码助理PDA(Personal Digital Assistants)等这种对压缩率要求较高的场合，但由于基于理解的方法对知识的需求过多，从而影响其广泛应用[54]。

1.4 多文档文摘的技术路线

多文档自动文摘与单文档自动文摘有着不同的特点：首先，处理的对象不同。由于多文档自动文摘要处理的对象是多篇文档，这些文档或者在描述同一话题，只是侧重点不同，或者是对同一话题的跟踪等等。总的来说，多篇文档在描述相同或相近的内容时，不同文档在内容上有很大的重复；其次，对多文档进行理解的难度更大。对于多文档文摘，不同的文档来源不同，其体裁、结构或撰写风格也不尽相同，因此，对不同文档的内容进行分析、理解和融合的难度较单文档文摘要大[4]；第三，对文摘内容进行重新组织的策略不同。对于单文档文摘，文摘内容来自同一篇文档，对文摘内容进行重新组织时，有很好的参照物(原文档)。然而，在多文档自动文摘中，各个文档都可能存在非常重要的内容，需要将这些内容作为文摘的一部分。当对这些来自不同文档的内容进行重新组织时，将某一原文档作为参照系是不现实的，因此，需要有新的更复杂的策略来重新组织文摘内容。此外，来自不同文档的内容可能是不一致的，它们之间或许存在矛盾，这也是单文档文摘很少遇到的问题。

上述特点涉及到内容选择、文摘生成等自动文摘的几个关键问题，因此决定了单文档自动文摘的很多技术不再适宜于多文档自动文摘，如：内容抽取、文摘句排序等。典型的多文档自动文摘体系结构如图 1.4 所示。

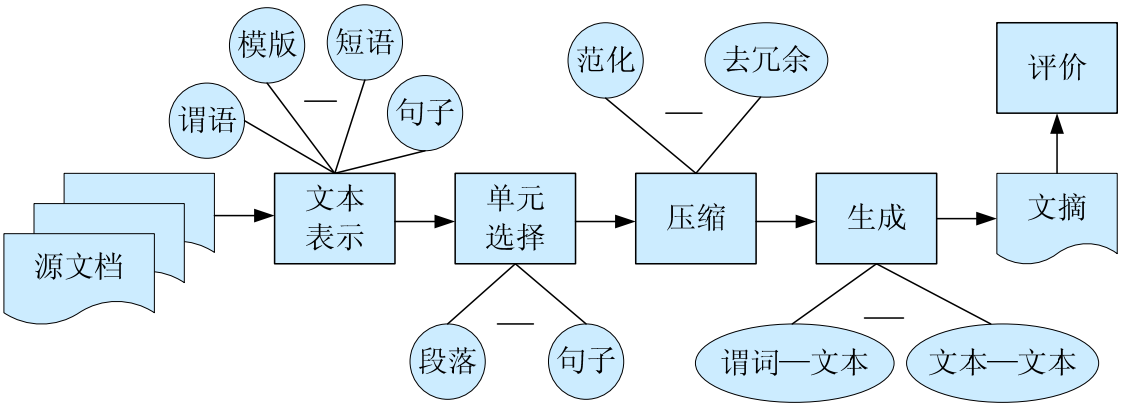

图 1.4 多文档自动文摘系统体系结构

多文档自动文摘一般分为 3 个阶段：文本表示阶段的主要任务是获得同一主题的文档并将文档拆分成基本信息单元；化简阶段包括单元选择和压缩两个部分，其主要任务是查找文档的共同主题、消除文档间的冗余信息、去掉相对无关的信息，以简化原文档而又保留其关键内容；生成阶段是将简化后的内容重新组织成一篇流畅易读的摘要。3 个阶段中，化简阶段是核心，也是目前各种多文档文摘技术的主要区别之处。通常与采用化简方法不同，其特征表示方式也不相同。

衡量原文档集中文档内容重要性的技术是文摘的核心技术。文摘作为原文档集的浓缩，不可避免的要从原文档集中过滤掉大量无用信息。文摘内容的选择决定文摘是否能有效把握原文档集的中心思想、最大程度返回用户需要信息，因而成为文摘系统性能的最关键指标。对于多文档文摘，因为内容来源于多篇内容相关的文档，因此在文摘中还要考虑将重复的、冗余的内容去掉。

从原文档集中选择内容的方法有多种：句子压缩法、抽取句子法、信息融合法等。尽管基于句子抽取的方法生成的文摘在可读性、一致性方面还存在诸多有待改进之处，但这种方法简单易行，需要的语言资源相对较少，而且在可读性、一致性方面可以利用其它 NLP 技术加以改善，因此，这些方法仍是目前多文档文摘最常用的方法。

1.4.1 基于质心的方法

基于质心(Centroid)的多文档自动文摘方法[55-56]的主要思想是：若一个句子的内容包含在其它句子中，则该句子在文摘过程中应该去除；可以选择一个具有代表性句子来替代具有同一话题的句子集的主要内容。因此，如何定位到这个代表性的句子是

该方法的关键。

基于质心的方法用向量空间模型[57]来表示文档质心和基本抽取单元。首先将具有同一话题的文档集表示成向量的形式，用以代表文档集的质心，其中的每个元素用文档集中单词的权值来表示。

$$Centroid(D)=(W(t_1),\cdots, W(t_i), \cdots, W(t_n)) \tag{1.1}$$

其中 $W(t_i)=TF(t_i)\times IDF(t_i)$，*TF* 和 *IDF* 分别表示特征 t_i 在文档集中的频率和反文档频率。为避免一些信息量不大的特征的干扰。只有当 $W(t_i)$大于预先设定的阈值时，特征 t_i 才会保留在向量中。

句子的重要性通过计算每个句子的权重得到，而句子权重采用句子特征的线性加权和，这些特征包括：句子所处位置、与文档集质心的距离、与所在文档首句的信息重叠程度等。由于每个句子是用向量表示的，所以句子与文档集质心的距离及与文档首句的信息重复程度都可以用两向量的夹角余弦来表示。当然，在计算句子权重时可以采用更多的特征，至于哪些特征或特征的组合更好，需要在实际系统中检验，并不是越多越好。

文摘句选择的准则是选择那些权重较高，而且与已选择句子的相似度较小的句子，类似于信息抽取(Information Extraction)技术中的 MMR(Maximal Marginal Relevance)方法[58]。选择句子的个数以不超过压缩率或用户要求的长度为准。最后将所选择的句子按照某种策略进行排列得到文摘。

1.4.2 基于聚类的方法

基于聚类方法[59-60]的基本思路是：首先将文档集中的所有文档拆分成句子，然后将每个句子表示成向量空间中的一个向量，对向量进行聚类后，从每个类中挑选一个具有代表性的句子作为文摘句，最后将所选择的文摘句整合成一篇摘要。

在将句子表示成向量时，通常是将每个单词的 *TFIDF* 值[61]代表向量中的一个元素。*TF* 是指单词在句子中的特征频率；*IDF* 表示单词的反句子频率。为了提高聚类的精度，有些研究人员利用短语[62]、概念[63]或基本要素(Basic Element)[64-65]来代替单词。

一个句子集的代表句一般是句子集的中心点或距离质心最近的样本点。根据每个句子集中样本点的数量由高到低对句子集排序，然后从高到低从每个句子集中选择具有代表性的句子组成文摘。

基于聚类的方法面临的一个难题是如何确定类的个数，这其实也是所有聚类算法

共同面临的一个难题。目前武汉大学刘德喜博士[65]和哈尔滨工业大学秦兵教授在自动探测原文档集中类的个数方面进行了一些研究，并取得了较为满意的实验效果。

1.4.3 基于信息融合的方法

基于信息融合的方法[66-67]同样需要通过聚类将表达同一主题的句子聚集在一起，但与基于聚类的方法不同的是：前者不是从每个类中选择代表句，而是对整个类中表达的信息进行融合处理，最后借助自然语言生成工具重新生成句子组成摘要。

在实现信息融合时，首先需要将类中的每个句子表达形式转换成一种依存树结构（Dependency Tree Structure），通过发现不同依存树之间的最大交集来获取多个句子之间描述的共同内容；然后按照集合中各元素出现的频率进行排序，并删除频率小于预设阈值的元素。包含交集中元素最多的那个依存树被挑选出来，并裁剪掉不在交集中的子树。其实，真正的信息融合是要将交集中的各元素重新组合生成一个全面的、新的、符合语法规范的依存树，但这一技术目前仍有很大的挑战性。为了简化算法，通常只是从中选择一个依存树。所以从这个意义上讲，基于信息融合的方法并没有实现真正意义上的“融合”。

除上述 3 种方法之外，还有一些多文档自动文摘的方法，例如：给出文档各单元关系的图形化描述[68]；通过遗传算法试图选择最优的句子组合[69]；先采用单文档自动文摘技术生成各文档的文摘，再对这些文摘优化组合[70]；给定受限领域的模板，用信息抽取技术对模板进行填充，将填充后的内容视为摘要[71]等等。

1.5 自动文摘的评测方法

对于自然语言理解系统而言，如何科学的、客观的进行评测是其研究的重点之一，同时也是最容易引起争议的地方，如机器翻译（Machine Translation）及对机器翻译结果的评测、文本分类及对文本分类结果的评测等等。评测方法难以解决已经成为制约自然语言理解发展的主要原因之一。人们对文档主题的认识是非常复杂的，不同读者对同一文档主题的认识可能有很大的差异，即使是同一个人在不同的时间对同一文档进行文摘，其结果也不一定完全相同[72]。因此，文摘系统的评测一直被研究人员认为是一项很艰难的研究课题[73]，它涉及到信息论和语言学等多个学科的领域知识。直到目前为止，自动文摘领域还没有一种统一的、方便的、可重复的评测方法[74]。目前，国内外研究人员在自动文摘研究工作中常用的评测方法有：标准摘要与机器摘要进行比

较、几个文摘系统之间相互比较、机器摘要与原文内容进行比较、机器摘要对某一特定任务的完成所提供的支持等多种方法。

自动文摘的评测方法可以分为外部评测方法[75]和内部评测方法[76-77]两类。外部评测方法是将机器摘要应用于某一特殊任务中，用机器摘要对提高该项任务的效果来评价自动文摘系统的优劣；内部评测方法是直接通过分析机器摘要的质量来评测自动文摘系统的，主要评测机器摘要的可读性和内容的完整性。可读性是指机器摘要在文字上的流畅程度，主要包括句子是否通顺、是否出现主语悬挂等等。内容的完整性是指机器摘要中包含标准文摘中的信息量的多少，其中标准文摘的来源主要有两个途径：(1)将经过加工和标注的原文作为标准文摘进行参考，为评测提供判定依据；(2)将多个专家根据原文生成的摘要的并集作为标准文摘。

1.5.1 内部评测方法

内部评测方法是指通过对机器摘要的直接分析来评价文摘的质量，例如分析机器摘要是否流畅、对原文关键内容的覆盖程度、与标准文摘的近似程度等。事实上，这些方法都不尽如人意，因为即便是人工产生多篇标准文摘，也很难说出哪篇更好。另外，正如描述事物方式的多种多样，用户对文摘的需求也是多种多样的，因此能够让读者接受的摘要也是多种多样的。实验结果表明，就连哪些段落或句子是否应该出现在摘要中也很难达到共识[43]。

对于内部评测方法而言，最可靠的方法还是采用人工评测。它是由语言学家在浏览自动文摘系统生成的一定数量摘要后，根据其概括性、连贯性和信息覆盖率等指标对系统做出主观的评价。由于人工评测是由语言学家根据自己对文档内容的理解来评判的，所以这种评测方法的主观性太强。另一种人工评测方法是采用 Turing 测试[78]，即将人工摘要与机器摘要混在一起，由第三方评测人员对每篇文档的所有文摘进行对比排序，然后统计机器摘要在所有摘要中的排序结果，并根据这一结果判断自动文摘系统的性能。总的来说，人工评测方法虽然可以全面衡量摘要的质量，但受到繁重工作量和主观因素的影响，使其不能被广泛应用。

1.5.2 外部评测方法

为了弥补内部评测方法的不足，研究人员提出了采用外部评测的方法对自动文摘系统进行评价。外部评测方法通常是在一个特定任务中来评价自动文摘系统，因而相

对于内部评测方法具有较少的主观性，易于对多个文摘系统进行比较。Wang[79-80]利用原文档和机器摘要分别对分类器进行训练，并根据分类器对测试用的机器摘要和原文档的分类效果来评测摘要质量的优劣。Brandow[45]尝试在信息检索(Information Retrieval)任务中评测自动文摘系统，将机器摘要进行检索与采用原文档进行检索的速度和准确率进行比较，来确定是否可以在信息检索中利用摘要来替代原文。在新闻分析任务中，Miike[81]根据利用机器摘要进行新闻分析的效果来评价摘要的质量。

在特定应用中评测自动文摘系统也有利于文摘系统在其它领域中的应用[82-83]。但是这类评测方法也具有一个缺点，即每次评测只能针对一个具体的任务，不利于系统性能的全面提升[84]。

1.5.3 自动评测方法

受机器翻译评测方法BLEU(Bilingual Evaluation Understudy)[85]的启示，微软研究院Chin-Yew Lin研究员[86]开发了一个文摘的自动评测软件ROUGE(Recall-Oriented Understudy for Gisting Evaluation)，并将其用于了第四届文本理解会议的评测。ROUGE将对等文摘和标准文摘中所共有的*n-gram*的多少作为评判对等文摘优劣的依据。具体来说，ROUGE提出了3类标准：

(1) ROUGE-N 是*n-gram*召回率，即在标准文摘和对等文摘里均出现的*n-gram*占标准文摘中所有*n-gram*的比例。ROUGE的计算方法如公式(1.2)所示：

$$ROUGE-N=\frac{\sum_{S\in\{Models\}}\sum_{n-gram\in S}Count_{match}(n-gram)}{\sum_{S\in\{Models\}}\sum_{n-gram\in S}Count(n-gram)} \tag{1.2}$$

其中，n为*n-gram*的长度，$Count_{match}$(*n-gram*)为*n-gram* 同时在标准文摘S和对等文摘中出现的次数，而*Count*(*n-gram*)为*n-gram* 在标准文摘*S*中出现的次数。在往年的评测中，n 取值为1、2、3、4。

Chin-Yew Lin和Eduard Hovy 在2001年DUC的测试语料上做实验，发现ROUGE-1和ROUGE-2与人工评价的一致性较高，尤其是ROUGE-1相当接近人工评价，而ROUGE-3 和 ROUGE-4 与人工评价的结果一致性较低[86]。

(2) ROUGE-L是基于最大公共子序列(LCS，Longest Common Subsequence)的方法，它考虑了单词的顺序。首先给出子序列(Subsequence)的定义[87-88]：

已知序列$Z=\{z_1,z_2,\ldots,z_n\}$，$X=\{x_1,x_2,\ldots,x_m\}$，如果存在严格递增的序列$\{i_1,i_2,\ldots,i_k\}$

使得$x_{ij}=z_j$ 成立($j=1, 2,..., k$)，那么Z 是X 的子序列。

如果序列Z 既是序列X 的子序列，也是序列Y 的子序列，那么Z 就是X 与Y 的公共子序列。所谓最大公共子序列就是两个序列的所有公共子序列中最长的序列。可以在$O(mn)$时间内计算出两个序列的最大公共子序列。

假定有两个文摘X 和Y，直观上可以认为，X 和Y 的最大公共子序列越长，它们之间就越相似。

ROUGE-L与ROUGE-N的主要区别在于：ROUGE-L考虑了单词的先后顺序。

(3) ROUGE-W也是基于最大公共子序列的方法，但是这种方法更倾向于连续的最大公共子序列。

1.5.4 中文自动文摘的评测

国家863计划中文信息处理与智能人机接口技术评测分别在1995年、1998年、2003年和2004年进行过自动文摘的评测。1995年的评测采用的是机器文摘与标准文摘之间重合率的方法，之后的评测都是采用专家评测的方法。2004年的863文摘质量采用人工打分的方法，评价标准[88]如示：

- 5分：句子通顺，句子间意义连贯，句子间在恰当处有关联词连接，原文内容概括全面，包含原文所有信息，文摘逻辑结构合理，真实体现原文主题，通过文摘就能了解全文主要内容；
- 4分：整个文摘总体上文字流畅，仅仅有个别句子不通顺或者有极小的不影响读者阅读的错误，文摘内容比较全面，全部围绕原文主题，仅仅遗漏个别要点，逻辑结构也比较合理；
- 3分：句子基本流畅，但句子间意义不够连贯，文摘内容大致体现了原文的主题，概括内容包含了大部分要点，逻辑结构有一定的主题，通过文摘可以使人了解大部分原文的内容；
- 2分：句子间意义不够连贯，原文内容概括不全面，逻辑结构混乱，但文摘的内容对人了解全文还能有所帮助；
- 1分：句子不够通顺，句子间的连贯性也不够好，逻辑结构混乱，文摘内容以偏概全，原文要点遗漏现象严重，不能体现原文主题，但仍能了解到部分内容；
- 0 分：句子不通顺，句子间意义不连贯，完全歪曲或者严重偏离原文主题，文摘内容不能为人所接受。

1.6 论文研究内容

本文针对汉语的特殊性，围绕着自动文摘过程中涉及的技术和理论展开研究和探索，具体的研究内容包括：基于关键词抽取的单文档文摘，基于主题聚类和语义分析的多文档文摘，多文档文摘句排序，面向用户查询的自动文摘以及这些研究在提升中文文本分类性能的应用。

1.6.1 基于关键词抽取的单文档文摘

关键词是单文档文摘的一个特例，由于用户通过关键词可以基本了解文章的主题，这充分说明了关键词蕴含了文档的重要信息。如果在事先知道文档关键词的情况下进行自动文摘，这势必能够进一步提高摘要的质量，所以本文将研究如何通过词汇链构建来抽取关键词，并在其基础上进行单文档自动文摘。

1.6.2 基于主题聚类和语义分析的多文档文摘

多文档集合是由一系列主题相关的文档组合而成的，尽管这些文档都是围绕着一个共同的主题的描述与说明，但是每篇文档描述的信息各有侧重，即使在一篇文档中也会出现不同的局部主题的描述。因此，多文档集合的主题是由不同的局部主题描述出来的，为了使文摘能够尽量多地包含文档集中的信息，本文将研究如何利用句子聚类的方法对局部主题进行识别，并在其基础上抽取文摘句，并研究如何对文摘句进行润色处理(文摘句冗余消除和平滑处理)。

1.6.3 多文档文摘句排序

从原文档集中抽取文摘句后，不同的排列方法可能导致不同的文摘质量。本论文研究一种将局部主题间的内聚度与 MO 算法相结合的文摘句排序模型，在统计局部主题间相对位置的基础上，建立它们之间的关系有向图并计算其内聚度，将排序问题转化为对图的搜索过程，每从有向图中输出一个顶点时，从剩余顶点中查找与其具有最大内聚度的顶点，若内聚度大于阈值，则将这两个顶点所代表的局部主题文摘句置于摘要中相邻的位置，从而实现对文摘句的排序。

1.6.4 面向用户查询的自动文摘

搜索引擎为人们从网络中快速获取信息提供便利的同时，也存在着诸多缺陷，即

自动检索到的结果是一个数量庞大的网页集合，其中包含有用信息和无用信息，而选取有用信息的过程则完全取决于用户个人，从而造成了用户在获取信息过程中的巨大负担，很大程度影响了信息检索的效率。显然，如果能够通过网页自动文摘技术将网页文档压缩成一两句主旨，将会大大提高用户在判断信息是否为有用信息的速度，本文在考虑网页文档的半结构化特征以及主题与用户查询的关联等信息的基础上，研究一种用于面向查询文摘的多特征融合的句子重要度计算的策略。

1.6.5 应用自动文摘技术的文本分类

摘要在一定程度上覆盖了原文档中所有的重要信息，用户完全可以通过阅读摘要了解原文的意思表达，所以摘要可以作为原文档的某种替代。既然通过阅读原文10%~30%的摘要，用户可以了解原文的基本信息，那么就一定能够依据摘要确定文章所属的类别，又因为传统的文本分类的特征空间的维数非常高，导致分类的计算过于复杂，鉴于上述考虑，本文研究是否可以直接用摘要参与传统的文本分类的特征选择或者是在摘要的基础上挑选特征参与分类器训练，以降低特征选择和训练的运算量和提高文本分类的性能。

1.7 论文组织结构

本论文的组织结构如下：

第 1 章首先阐述课题提出的背景、研究意义，以及本领域国内外研究现状；然后介绍了单文档文摘和多文档文摘的技术路线以及采用的评测方法；最后给出本文的主要研究内容和介绍论文的组织结构。

第 2 章主要研究基于词汇链构建的关键词抽取及基于关键词抽取的单文档文摘。首先介绍了目前关键词抽取的相关研究，提出了一种基于词汇链构建的关键词抽取算法；接着给出了利用文档关键词信息进行单文档文摘的具体实现步骤；最后给出了关键词抽取和单文档文摘的对比实验结果与分析。

第 3 章首先给出了基于局部主题聚类与语义分析的多文档文摘系统（CSA-MDSS）系统结构图，介绍了基于聚类分析的局部主题确定的方法和候选文摘句选择的方法；然后在候选文摘句抽取的基础上，提出了改进的 MMR-SS 句子冗余消除算法和基于语义分析的文摘平滑处理和连贯性处理；最后从信息覆盖率和信息冗余度两个方面对三个多文档文摘算法（MEAN、TOP 和 CSA-MDSS）进行了评测。

第 4 章首先介绍了多文档文摘句排序的必要性和研究现状，分析了两种常用文摘句排序(Majority Ordering 和 Chronological Ordering)的算法思想和不足之处；然后给出了一个将局部主题间的内聚度融入多文档文摘句排序的改进 MO 算法；最后从人工评价、ROUGE 自动评价和流利度自动评价三个角度对三种排序算法(MO、CO 和改进 MO 算法)进行了对比实验。

第 5 章讨论一种多特征融合的句子重要度计算的策略，首先对查询输入进行短语识别和网页结构分析；然后将关键词短语、网页结构等启发式规则信息融入到句子重要度计算，并分析句子与查询输入的关联特征，使文摘内容能与用户需求保持一致；最后利用句子相似度计算的方法减少文摘句的冗余度，并给出了对比实验结果与分析。

第 6 章提出应用自动文摘提升中文文本分类性能的方法，充分利用了文摘是原文档重要信息的表述，通过阅读文摘完全可以确定文档所属类别的假设，让训练集文档的摘要直接参与特征选择和分类器训练，并通过实验验证了该方法的有效性。

结论部分首先对本文的研究工作进行了总结，声明本研究的主要工作和创新性成果；最后还对本课题进一步的研究工作给出了设想。

第2章 基于词汇链构建的关键词抽取与单文档文摘

2.1 引言

关键词同文档标题和摘要一样，也是一种提供快速了解全文信息的重要途径，它相当于一种压缩率更高的摘要，而且不需要考虑文摘的连贯性问题。如果一篇文档有关键词描述，那么用户可以很容易地通过关键词判断该文档是不是自己所想要的[89]。目前学术论文一般都要求作者定义 3~5 个关键词，但是其它文献却很少提供关键词，如果手工直接从文献中抽取关键词不仅主观性强，而且费时费力，抽取不当还会对后续的应用研究造成不利影响。

关键词的生成方法主要归为两类：关键词指定(Keyword Assignment)[90]和关键词抽取(Keyword Extraction)[91-92]。所谓关键词指定是从系统预先定制的控制词汇中找出最适合的关键词，来指定给资料库中的每一篇文章，因此指定给每一篇文章的关键词不一定出现在该篇文章中；而关键词抽取则是依靠计算机从文档本身中寻找最能代表该文档的词汇，因此抽取的关键词一定是文章中出现的词汇，其在自动文摘、文本分类、信息检索等方面有着重要的应用。

关键词抽取的定义为：给定文档或文档集 D，$DS=\{W_1,W_2,\cdots,W_n\}$，其中 $W_i(i=1,2,\cdots,n)$是 D 中的词或者词组。关键词抽取的结果 R 满足 $R\in DS$，且 R 能较好地反映 D 的内容。

从定义可以看出，关键词抽取的基本假设是所抽取的关键词都必须存在原文档中。判断一个词是否是关键词是一项比较困难的工作，为此本章在对文档进行分词和未登录词识别的基础上，提出了一种通过词汇链的构建进行中文关键词抽取的新方法，给出了利用《知网》为知识库构建词汇链的算法，首先通过计算词义相似度构建词汇链，然后结合词汇所在词汇链的强度、信息熵和出现位置等属性进行关键词抽取，并利用抽取出来的关键词进行单文档自动文摘。

2.2 相关研究

由于关键词指定往往局限于某个领域，移植性和拓展型不强，所以近年来国内外研究人员热衷于统计和语义相结合的关键词抽取技术的研究。

在国外，Turney[90]将遗传算法和决策树机器学习方法相融合研制出了关键短语的

抽取系统 GenEx；Witten[91]采用朴素贝叶斯技术对短语离散的特征值进行训练，获取模型的权值，然后从文档中抽取关键短语；Hulth[92]采用 Rule Induction 学习算法，提出了在学术论文摘要中抽取关键词的方法，使得关键词抽取的准确率达到了 29.7%。

针对中文关键词抽取，Yang Wen-feng[93]提出了采用互信息等统计方法进行关键词抽取的算法，但该算法实现非常复杂，而且需要大量的存储空间；李素建[94]提出了利用最大熵模型(Maximum Entropy Model)进行关键词抽取的方法，由于特征的选择以及估计特征参数时不够准确，最大熵模型在关键词抽取中的应用效果不佳；王军[95]提出了一个用于自动标引的文档主题关键词抽取算法，但是该算法只能从已标引的结构化语料库中元数据的标题中抽取关键词。

2.3 基于《知网》的词汇链构建

2.3.1《知网》简介

《知网》(HOWNET)是一个以英语和汉语的词语所代表的概念为描述对象，以揭示概念与概念之间以及概念所具有的属性之间的关系为基本内容的常识知识库。《知网》系统的哲学是：“世界上一切事物都在特定的时间和空间内不停地运动和变化。它们通常是从一种状态变化到另一种状态，并通常由其属性值的改变来体现。”因此，《知网》的运算和描述的基本单位是：万物、属性、时间、空间、属性值以及事件。

《知网》作为一个网状的知识系统，它着力要反应的是概念的共性和个性以及概念之间和概念的属性之间的各种关系，构造一种易于计算机学习得知识网络体系，从而使知识对于计算机而言是可操作的。近年来国内学者利用《知网》进行了 NLP 领域中相关问题的研究，清华大学利用《知网》中的动态角色与属性进行基于语义依存关系的汉语语料库的构建，香港科技大学利用《知网》信息进行了汉语语料库的语义标注研究。《知网》作为比较完备的知识库已经被研究人员广泛利用起来。

2.3.2 词汇链的提出

McKeown[96-97]在研究如何根据描述相同时间的多个新闻报道形成一个一致的、连贯的摘要的过程中，发现相关性和一致性概念为人们获取文档主题提供了一种非常有效的途径。一致性表示一个包含多个句子的文档中根据子句或句子之间的宏观关系呈现出来的全局结构[98]。相关性表示一种将文档整合成一个文法单元的特性，它建立在文档中各种元素的相互关系基础之上。

为了表示词汇之间的相关信息，Hasan[98]提出了“聚合”的概念。“聚合”是指文档中不同部分的词汇“粘连在一起”，它可以从语义的角度，通过对词汇的替换、参考、省略、连接等成分的分析实现。词汇聚合[99] 是建立在对相关词汇进行语义分析的基础上，用来表示这样的词汇之间相关性的方式。

词汇聚合不仅出现在两个词汇之间，也出现在一系列相关的词汇之间，这种情况称为“词汇链”。词汇链为文档结构提供了一种语义相关性的表示形式，是一系列相关联的词汇组成的集合。词汇链被很多研究人员用在文档结构分析[100]、信息检索[101]以及检查文章的用词不当[102]等方面。

在本章中我们采用词汇链分析方法对于关键词抽取和单文档自动文摘进行了研究，首先在对文档进行分词和未登录词识别的基础上，通过计算词汇间的相似度构建词汇链；然后通过计算每个词汇链的强度及每个词汇的权重，从每个词汇链中抽取出能够代表文档主题的关键词；最后通过这些抽取的关键词计算句子的重要度并摘录出句子形成最终的文摘。

2.3.3 基于《知网》的词义相似度计算

词语相似度计算是其它语言单位(如语义块[103]、句子[104-105]以及篇章[106]等)相似度计算的基础，也是目前研究较多，应用较广的语言处理技术。词语相似度可以从多个方面(词性、语法、语义和语用)进行评判，对于不同的应用有不同的侧重。本章采用《知网》作为系统的语义知识库，参照中科院计算所刘群[107]提出的语义计算方法进行词语相似度计算，充分考虑了词汇与词汇之间的语义关系。由于《知网》中每个概念都是通过一组义原来描述的语义表达式，而且《知网》提供了义原分类树，树中子节点与父节点的义原具有上下位关系，因此两个词汇的语义相似度可以利用义原分类树来计算。

2.3.3.1 词义相似度计算原理

两个中文词汇W_1和W_2，如果W_1有 n 个概念：$S_{11},S_{12},\cdots,S_{1n}$，$W_2$有 m 个概念：$S_{21},S_{22},\cdots,S_{2n}$，规定$W_1$和$W_2$的相似度为各个概念之间相似度的最大值，其计算公式如下：

$$Sim(W_1,W_2)=\max_{i=1..n,j=1...m} Sim(S_{1i},S_{2j}) \tag{2.1}$$

从而两个词汇间的相似度计算问题归结为两个概念之间的相似度计算问题。

2.3.3.2 义原相似度计算

因为所有的概念最终都归结于用义原来表示，所以概念相似度计算就转化为义原的相似度计算。由于《知网》中存在 Entity 和 Attribute 等 11 棵义原树，但有些义原树上的义原没有父子关系，并不能体现义原之间的上下位特征，因此无法使用。而 Entity、Event、Attribute、Attribute Value、Quantity、Quantity Value 六棵义原树具有上下位关系，因此可以通过义原在分类树上的距离来计算义原相似度。两个义原相似度的计算公式如表(2.2)所示。

$$Sim(p_1,p_2)=\frac{\alpha}{Dis(p_1,p_2)+\alpha} \tag{2.2}$$

其中 p_1 和 p_2 表示两个义原，$Dis(p_1,p_2)$ 是 p_1 和 p_2 在分类树上的路径距离；α 是一个调节参数，实验中将α设为 1.6。

2.3.3.3 实词概念的相似度计算

《知网》中的实词概念由第一基本义原、其它基本义原、关系义原描述和符号义原描述组成。在计算实词概念相似度时，认为只有它对应的 4 个部分都相似，才会认为整体相似，因此应该在分别计算实词概念 4 部分相似度的基础上计算实词概念的相似度，这 4 部分相似度是：

(1) 第一基本义原相似度：即两个义原间的相似度 $Sim_1(S_1,S_2)$；

(2) 其他基本义原相似度：由于其它基本义原描述式有多个，因此必须分别对这些独立义原描述式计算相似度，然后加权平均得整体相似度 $Sim_2(S_1,S_2)$；

(3) 关系义原相似度：将相同的关系义原描述式分为一组，并计算其相似度 $Sim_3(S_1,S_2)$；

(4) 符号义原相似度：将相同的关系符号描述式分为一组，并计算其相似度 $Sim_4(S_1,S_2)$。

通过以上规则，两个实词概念语义表达式的整体相似度为：

$$Sim(S_1,S_2)=\sum_{i=1}^{4}\beta_i\prod_{j=1}^{i}Sim_j(S_1,S_2) \tag{2.3}$$

其中，$\beta_1(1\leqslant i\leqslant 4)$为调节参数，且：$\beta_1+\beta_2+\beta_3+\beta_4=1$，$\beta_1\geq\beta_2\geq\beta_3\geq\beta_4$。在实验中，把参数设为 β_1=0.5，β_2=0.2，β_3=0.17，β_4=0.13。

2.3.4 分词及未登录词识别算法

本章采用的分词算法是中科院计算所的 ICTCLAS 分词系统，但在实验过程中，我们发现分词后的文本中有很多个连续的单字独立成词，但经过观察和分析发现，这些连续的单字往往是一些未登录词(领域术语等等)，而这些未登录词对于关键词抽取有着重要影响，如果不能正确的识别，则将会大大降低了关键词抽取的准确率。针对上述的不足，本文利用了相邻词[108]的共现频率进行未登录词的识别。

词***t***的相邻词是指对句子进行分词后, 词***t***之前的一个词和之后的一个词。例如：对“自动问答领域”分词后得到“自动/问答/领域”， 那么“问答”的相邻词即为:“自动”(称为“前邻”)和“领域”(称为“后邻”)。显然，由于词可能出现在句首或者句尾，因此前邻和后邻可能为空。对文档中每个词***t***的相邻词的频繁程度进行考察，从而判断词***t***及其相邻词是否需要合并以成为语义完整的未登录词。未登录词识别的具体算法为：

(1) 利用 ICTCLAS 对文档 d 进行分词和词性标注，去除“的”、“得”、“在”、“个”和“是”等单字虚词后，将剩余的所有词都加入集合 W 中；

(2) I=0；

(3) I=I+1，若 I 大于阈值 γ，则转至(9)；否则，对词集合 W 中每个词 ***t*** 统计出它在文档 d 中的前邻和后邻的分布，并根据某种策略判断是否具备频繁前邻和频繁后邻。例如可以根据某个前邻 PR(前邻和后邻均不包含单字虚词)出现概率是否大于预定阈值 ζ 来认定它是一个频繁前邻。γ 通常取 4 或者 5，ζ 取 0.6。

(4) 若 ***t*** 具有频繁前邻 PR 和频繁后邻 BE，则将 PR+***t***+BE 拼成一个词加入未登录词候选集合 W^* 中；

(5) 若 ***t*** 仅具有频繁前邻 PR，则将 PR+ ***t*** 拼成一个新词加入词集合 W_{PR} 中；

(6) 若 ***t*** 仅具有频繁后邻 BE，则将 ***t***+ BE 拼成一个新词加入词集合 W_{BE} 中；

(7) 将 W_{PR} 和 W_{BE} 中共同出现的词加入 W^*，清空 W_{PR} 和 W_{BE}；

(8) 令 $W = W^*$，清空 W^*，转至(3)；

(9) 利用未登录词候选集合 W^* 进行二次分词。

通过上述算法能够提高未登录词的识别率，提高分词的准确率，更利于统计词频，更利于文档关键词的抽取。实验证明这种基于相邻词共现进行未登录词识别的方法是很有效的。

2.3.5 词汇链构建算法

Morris[100]根据 WORDNET 中单词间的关系，提出了基于 WORDNET 的词汇链构建算法。我们利用《同义词词林》和 HOWNET 来确定中文词汇间的关系。Morris 仅仅选择出现在 WORDNET 中的名词作为关键词的候选词，为了进一步提升关键词抽取的精度，我们选择的候选词是 HOWNET 中收录的动词、名词、形容词(词频大于预定阈值)以及识别的未登录新词。

要构建词汇链首先要对中文文本分词和未登录词的识别，并对词性进行必要标注，然后将正文中名词、动词、未登录词以及词频大于指定阈值的形容词标注为候选词，计算它们与初始词汇链的相似度，并加入相应的词汇链。具体算法如下：

(1) 对文档集进行分词、词性标注和未登录词识别，并统计每个词在文档集中的特征频率 *TF* 和文档频率 *DF*；

(2) 因为有些领域词汇并未被《知网》收录，而这些词汇相对比较重要，所以 *TF* 大于指定阈值 δ(一般 δ 取值为 3)的未登录词将单独作为一个词汇链 *M*；

(3) 选择文档集中的所有名词、*TF* 大于指定阈值 δ 的动词 $W_1, W_2, \cdots, W_n$ 作为候选词汇集，并取 W_1 构建初始词汇链 L_1；

(4) 依次从候选词汇集中选择词 W_i ($i \in [2,n]$)，按照公式(2.4)计算它与每个词汇链的词义相似度 $Sim(W_i,L_j)$，即与该词汇链中每一个单词的语义相似度的平均值；

$$Sim(W_i, L_j) = \frac{1}{N}\sum_{k=1}^{N} Sim(W_i, W_{kj}) = \frac{1}{N}\sum_{k=1}^{N} \max_{i=1..p, j=1...q} Sim(S_{1i}, S_{2j}) \tag{2.4}$$

(5) 如果最大词义相似度 $Sim(W_i,L_k)$大于预设的相似度阈值 ζ，就把 W_i 插入词汇链 L_k；

(6) 如果最大词义相似度 $Sim(W_i,L_k)$小于预设的相似度阈值 ζ，就重建一个词汇链，并把 W_i 插入新的链中；

(7) 重复步骤(3)~(6)，直至全部候选词汇计算完毕。

从上述算法中，容易看出阈值 ζ 的选择是与构建词汇链的数目成正比关系的，即 ζ 越大，词汇链数目越多。

2.4 词汇链权重计算

在计算词汇链重要性进行的时候，考虑了以下五个因素[109]：

(1) 构成词汇链的每个词汇的初始权值；

(2) 词汇链的长度(包含的词汇的数目)：对于一篇文档而言，作者选用的词汇大多是

为描述文档主题服务的，并且具有相似的语义，因此表述文档主题的词汇链相对比较长，关键词应该优先从长度大的词汇链中抽取；

(3) 词汇链覆盖文档的范围：词汇链覆盖的文档范围越大，则包含主题的内容就越多；

(4) 词汇链内部的拓扑结构：考虑词汇间的关联程度，加强核心节点的重要性；

(5) 词汇链中词汇的分布密度：词汇分布越集中，整体的重要性就越高。

至此完成了对文档的词汇链构建，并对词汇链进行了评价，赋给相应权重。每个文档表示成 $T=\{T_1,T_2,\ldots,T_n\}$，其中 T_i 表示各个词汇链的权重。词汇链的权值越大，表达文档主题的能力就越强；反之，权值越小，离文档主题就越远。我们预设了一个阈值，从中取出最强的几个词汇链(肯定包含未登录词所在的词汇链)来共同表示文档，当然关键词将从这几个词汇链包含的词汇中抽取。

2.5 基于词汇链的关键词抽取

2.5.1 关键词抽取算法

依据上述算法构建的词汇链是若干个语义相近的词汇的集合，抽取其中的哪些词汇作为关键词应该考虑下列4个因素：

(1) 首次出现位置：这个属性表示在词汇所在的文档中该单词首次出现位置之前的词汇数量占文档中所有词汇数量的比率，这一属性的取值在0到1；

(2) 所处文档区域：这个属性总共包括三个属性，分别表示当前词汇是否在文档标题、文档摘要和章节标题中。采用这个属性是基于如下的假设：出现在文档标题、文档摘要和章节标题中的词汇是文档关键词的可能性比其他词汇要大；

(3) 所处词汇链的强度：即词汇所处词汇链的权值，权值越大，表明该词汇链表达文档主题的能力越强；反之，权值越小，表达文档主题的能力越小；

(4) 词汇的信息熵：这个属性代表了词汇承担文档内容的程度，是衡量当前词汇的平均信息熵，计算方法如公式(2.5)所示；如果词汇几乎在所有的文档中出现，则信息熵将会很小，如果词汇只出现在个别文档中，则信息熵将会很大，这种思路与*TFIDF*是一致的；

$$E_i=\frac{1}{log_2(M)}\sum_{j=1}^{M}\left[\frac{f_{ij}}{df_i}log_2\left(\frac{f_{ij}}{df_i}\right)\right] \tag{2.5}$$

其中，E_i表示词汇i的信息熵；M表示单文档中的句子总数或多文档集中的文档总数；f_{ij}表示词汇i在句子j或者文档j中出现的次数；df_i表示出现词汇i的句子数或文档数。

综合考虑词汇首次在文档中的出现位置、所处文档区域、所处词汇链的强度和 *TFIDF* 等 4 个属性，提出了如公式(2.6)所示的词汇权重计算方法：

$$Weight_i = \alpha * log_2(f_i + 1.0) * (1 + E_i) + \beta * T_i + \gamma * \frac{Length_i}{Length} + \eta * Position_i \quad (2.6)$$

其中，f_i表示词汇 *i* 出现的次数；$Weight_i$表示词汇 *i* 的权值；T_i表示词汇 *i* 所在词汇链的权重；$Length_i$表示词汇 *i* 所在文档中该词汇首次出现之前的词汇数；*Length* 表示文档中所有词汇数量；$Positon_i$表示词汇 *i* 所处文档区域的权重，如果词汇 *i* 出现在文档标题中时，$Positon_i$=5；出现在文档摘要中时，$Positon_i$=4；出现在章节标题时，$Positon_i$=2；否则，$Positon_i$=0.5；α，β，γ 和η是词汇权重计算考虑四个属性之间的调节因子，一般情况下均取值 1 即可。

至此，对词汇链 T_i 中包含的所有词汇进行权重计算后，可以表示为 T_i= {t_{i1}, t_{i2} , …, t_{ij}, …, t_{im}}，t_{ij} 表示构成词汇链 L_i 的关键词的权值信息。对 2.4 节抽取出来的词汇链中的所有词汇进行权值计算后，按照每个词汇的权值进行降序排列，依次从词汇链中挑出预定个数的词汇作为关键词。

2.5.2 关键词抽取实验

2.5.2.1 实验数据

考虑到所有学术论文都有作者拟定的关键词，因此我们从中国期刊网(http://www.cnki.net)中下载了 100 篇以“自动文摘”为主题的学术论文作为实验数据。由于本文仅考虑抽取出那些在文档中出现过的关键词，而学术论文中有些作者定义的关键词并不一定在文章中出现，为了使实验结果更加客观，于是我们手工将这些关键词进行了替换(文章中出现与该关键词语义相近的词)或者删除(出现与该关键词语义相近的词)，实验过程中，我们发现 94%以上的类似这种情况的关键词都可以从文章直接找到与其语义相近的词汇。

2.5.2.2 评价标准

从文档中自动抽取关键词后，需要评价抽取效果的好坏，常用的方法是通过将自动抽取的关键词与人工抽取的关键词进行匹配来评价，本文从准确度(Precision)和召回率(Recall)两个不同的角度去比较抽取效果。

$$准确率P=\frac{|自动抽取的关键词集合\cap作者拟定的关键词集合|}{|自动抽取的关键词集合|} \tag{2.7}$$

$$召回率R=\frac{|自动抽取的关键词集合\cap作者拟定的关键词集合|}{|作者拟定的关键词集合|} \tag{2.8}$$

2.5.2.3 实验结果

为了验证本文提出算法的可行性，我们将基于词汇链的关键词抽取算法与基于词频区域的关键词抽取算法进行了对比实验。所谓词频区域法是根据词汇在文档中的词频和出现位置等统计因素对每个词汇进行权重计算，然后依次抽取若干个词汇作为关键词。考虑到关键词抽取的数目对评价指标中的准确率和召回率有重要影响，所以在不同关键词数目下，对上述两种算法进行了精确匹配的对比测试，具体测试结果如图 2.1 和图 2.2 所示：

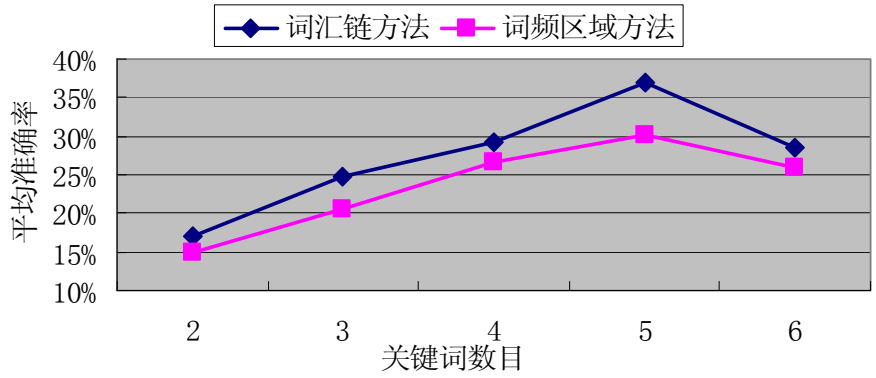

图 2.1 精确匹配的准确率对比测试结果

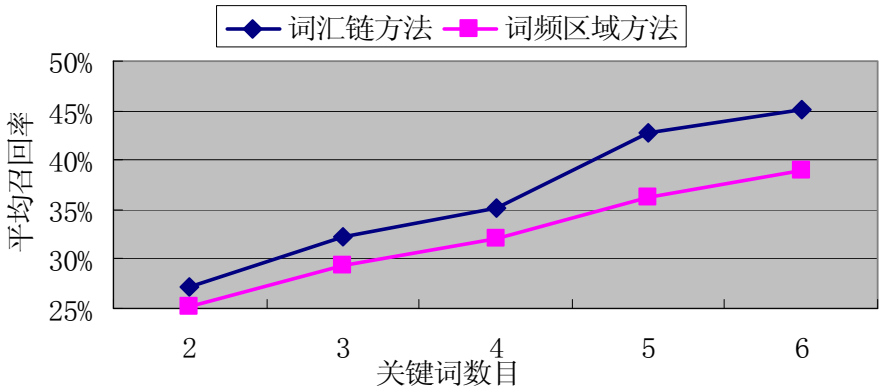

图 2.2 精确匹配的召回率对比测试结果

2.5.2.4 实验结果分析

从图 2.1 和图 2.2 的对比实验结果可以看出，词汇链方法的抽取效果要稍好于词

频区域方法，但是精确率和召回率比较低，均小于 50%。实验中，我们发现精确匹配有一定的弊端，例如文章作者拟定的关键词“分词系统”与自动抽取的“中文分词系统”在精确匹配时认为这两个关键词不匹配，显然不合理，所以我们采用了近似匹配的方法，即如果两个关键词之间存在包含关系或者两者的语义相似度大于指定阈值(本章采用 0.75)时，就认为这两个关键词是匹配的。近似匹配的测试结果如图 2.3 和图 2.4 所示：

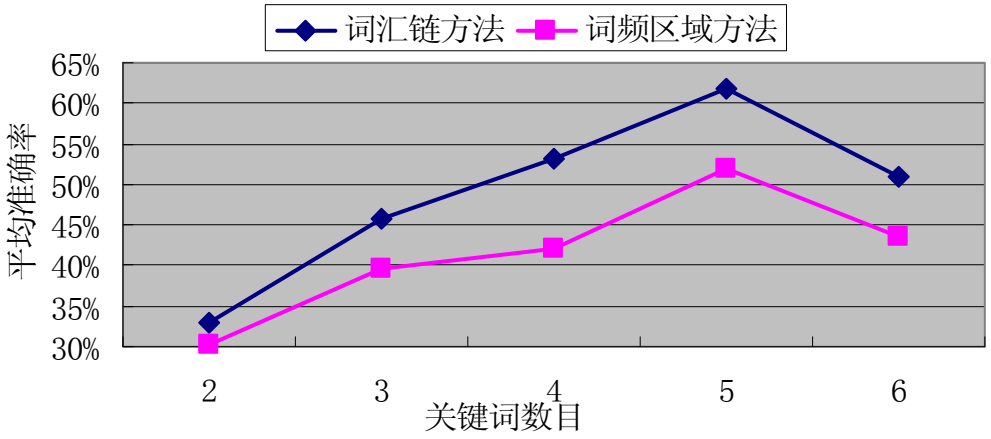

图 2.3 近似匹配的准确率对比测试结果

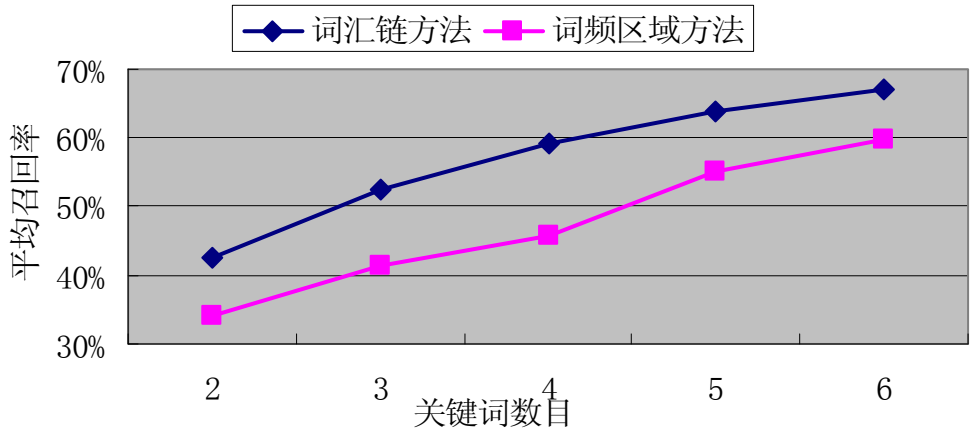

图 2.4 近似匹配的召回率对比测试结果

从图 2.3 和图 2.4 可以发现，近似匹配在比精确匹配获得更好的实验效果的同时，也凸现了词汇链方法比词频区域法的优势。分析其原因主要有两点：

(1) 词频区域法比词汇链法更依赖于文档中章节和标题的识别；

(2) 作者拟定的关键词不一定出现在标题和摘要中。词汇链法是对全文的词汇进行权重计算，并且文档的章节和标题只是权重计算的一小部分，它们识别的精度不高，对抽取效果的影响有限，所以词汇链法比词频区域法要更加合理有效，这一点也与实验结果一致。

2.6 基于关键词抽取的单文档自动文摘

2.6.1 单文档自动文摘算法

关键词是自动文摘的一个特例，由于用户通过关键词可以基本了解文章的主题，这充分说明了关键词蕴含了文档的重要信息。如果在事先知道文档关键词的情况下进行自动文摘，这势必能够进一步提高摘要的质量，所以本章在基于词汇链构建的关键词抽取的基础上，对单文档进行自动文摘的过程可以分为如下五个步骤：

(1) 将每篇文档拆分为若干句子的集合；

(2) 将抽取的关键词作为特征项，对每个句子进行向量表示；

(3) 句子重要度计算。根据句子中包含的关键词的数目和权重，以及句子所处位置等信息对句子的重要度进行计算；

(4) 依据重要度，将文档中的所有句子进行递减排序；

(5) 按照预先设定的长度或压缩比，选择权重最大的若干句子作为文摘句，并以它们在原文中出现的顺序输出生成文摘。

2.6.1.1 句子表示

向量空间模型 VSM(Vector Space Model)是 Salton 于 20 世纪 60 年代提出的，并成功地应用文本分类和信息检索等领域。向量空间模型以其简单、有效的文本信息表示模型被广大研究人员采用，本章以 2.5.1 节抽取的关键词集合作为文档表示的基本单位，将文档中每个句子表示为一个公式(2.9)所示的 VSM 向量：

$$V(S) = (t_1, W(t_1); \cdots; t_i, W(t_i); \cdots; t_n, W(t_n)) \tag{2.9}$$

其中，t_i 表示第 i 个关键词，$W(t_i)$ 表示特征 t_i 的权重，因为每个句子都是用同样的关键词集合来表示，所以句子的向量可以简化为：

$$V(S) = (W(t_1), \cdots, W(t_i), \cdots, W(t_n)) \tag{2.10}$$

2.6.1.2 句子重要度计算

为了衡量句子的重要性，需要给文档中的每个句子 S_k 赋予权重 $W(S_k)$，$W(S_k)$主要由以下几个因素决定：

（1） 句子中包含的关键词的重要性[110]：句子关键词权重之和越大则说明句子的重要度越大，为了消除句子长度的影响，应该将关键词权重之和除以句子所含的

关键词总数，得到句子的平均权重。

（2） 句子在文档中的出现位置：处于篇首、篇尾、段首和段尾等位置的句子通常比其他位置的重要度要高。

（3） 句子中是否包含提示语：例如：“综上所述”、“总而言之”等，如果包含，那么句子往往是对文档的主题内容进行了概括，因此该句子重要性相对较高。

（4） 句子是否为标题句：标题通常是对下文的一个概括，无论在信息量还是重要性都比较高。

（5） 句子是否以“例如”、“比如”等细节性词语开头，这些词语的出现意味着句子包含举例成分，并非概要性语句，因此重要性相对较低。

综合上述五个因素，句子的重要度计算 $W(S_k)$ $(1\leqslant k\leqslant m)$ 定义如下：

$$W(S_k)=\lambda_1\times\sum_{i=1}^{n}W(t_i)/Len+\lambda_2\times W_{pos}+\lambda_3\times W_{hint}+\lambda_4\times W_{title}+\lambda_5\times W_{ex} \tag{2.11}$$

其中，$\sum_{i=1}^{n}w(t_i)$ 是句子 S_k 中关键词的权值和；Len 是 S_k 中包含关键词总数；W_{pos} 表示句子 S_k 的位置权值；W_{hint} 表示提示语权值；W_{title} 表示标题句权值；W_{ex} 表示细节性词语权值；λ 是加权系数，$\lambda_1\geqslant 0.5$，$\lambda_1\geqslant\lambda_2\geqslant\lambda_3\geqslant\lambda_4\geqslant 0$，$\lambda_5\leqslant 0$，$\sum_{i=1}^{5}\lambda_i=1$。

2.6.1.3 粗文摘生成

文档中的每个句子的重要度计算出来后，依据其重要度将各句降序排列。摘要构造方法是依次将重要度最大的句子抽取出来，直到摘要达到指定长度，摘要长度一般由用户确定，通常是原文的 5%~25%，接着将这些从原文抽取的文摘句按其在原文中的顺序排列，这样文档的粗文摘就生成了。

2.6.1.4 文摘冗余消除

由于粗文摘中的每个文摘句的重要度都相对较高，而文档中经常会在不同位置对某些重要内容进行重复描述，而这些重复的、重要的描述可能多次出现在粗文摘中，从而导致粗文摘冗余度大的问题。为了提高文摘信息的覆盖率和全面性，本章采用通过句子相似度计算来减小文摘的冗余度。用 $Sim(S_i, S_j)$ 来表示句子之间的相似度，又记向量空间的原点为 O，则利用向量间的夹角余弦公式表示为：

$$Sim(S_i,S_j)=Cos\angle S_iOS_j=\sum_{k=1}^{n}(W_{ik}\times W_{jk})/\sqrt{(\sum_{k=1}^{n}W_{ik}^2)\times(\sum_{k=1}^{n}W_{jk}^2)} \tag{2.12}$$

其中，S_i 和 S_j 是 VSM 表示的句子向量，n 为向量维数，W_{ik} 和 W_{jk} 分别为第 k 个

关键词在两个句子中的权值。预设一个阈值，相似度高于该阈值的两个句子认为是意义重复的，只保留重要度高的一句，丢弃另一句。

2.6.2 单文档自动文摘实验

2.6.2.1 评价标准

目前国内外的自动文摘的评价方式主要分为人工评价和自动评价两种。人工评价通过一定的评价标准请多名专家对文摘进行打分评测，此方式的工作量较大且具有较强的主观性，从而延缓了系统的开发进程。本章我们采用基于句子命中率的自动评价方法，该方法通过考察机器生成的自动文摘与专家文摘在句子一级上的重合率对文摘进行评价，从准确率(Precision)、召回率(Recall)和调和值 *F_measure* 三个方面体现文摘的优劣，其中调和值 *F* 是准确率和召回率综合评价指标。

$$\text{准确率}P=\frac{|S_{\mathrm{m}}\bigcap S_{\mathrm{t}}|}{|S_{\mathrm{m}}|} \tag{2.13}$$

$$\text{召回率}R=\frac{|S_{\mathrm{m}}\bigcap S_{\mathrm{c}}|}{|S_{\mathrm{c}}|} \tag{2.14}$$

$$\text{调和值}F\_measure=\frac{(\beta^2+1)P\times R}{\beta^2\times P+R} \tag{2.15}$$

其中，S_{m} 是计算机自动摘录的文摘句，S_{t} 是由多位专家文摘人员手工摘录的文摘句集合的并集，S_{c} 是它们的交集，在本章的对比实验中令 β=1。

2.6.2.2 实验方法

为了对基于关键词抽取的单文档自动文摘方法进行评价，我们从 1998 年《人民日报》中选择了 100 篇文章进行测试，其中包括教育、财经、体育、军事 4 种类型的文章。将文摘压缩率分为 5%、10%、15%、20%、25%、30%共 6 种情况。首先请 4 位专业文摘人员独立地按照压缩比，手工从每篇文档中摘录出相应数目的句子，作为“理想文摘”，然后使用本章的基于关键词抽取的方法和 Edmundson 的方法生成各种压缩率的文摘，并用上述的 3 个评价指标对两个方法生成的机器文摘与理想文摘的重合率进行评价。

2.6.2.3 实验结果与分析

为了客观评价本章提出的基于关键词抽取的单文档自动文摘的效果，以及关键词

的数目对文摘效果的影响，我们对 4 类测试文档采用不同数目的关键词和六种不同的压缩率进行文摘，并与“理想文摘”进行了比较，*F_measure* 实验结果如表 2.1 所示：

表 2.1 基于关键词抽取的单文档自动文摘 *F_measure* 实验结果

分类	关键词数目	文摘压缩率					
		5%	10%	15%	20%	25%	30%
教育	3	0.4045	0.4825	0.4729	0.5325	0.5514	0.4847
	4	0.4219	0.5117	0.5135	0.5919	0.5676	0.5648
	5	0.3800	0.4940	0.4831	0.5598	0.5483	0.5375
财经	3	0.4155	0.4849	0.5150	0.5449	0.5294	0.5350
	4	0.4477	0.5647	0.5637	0.6119	0.6002	0.6238
	5	0.4095	0.4896	0.5167	0.5680	0.5553	0.5335
体育	3	0.4306	0.4860	0.5353	0.5293	0.5893	0.5131
	4	0.4538	0.5230	0.5684	0.6312	0.6184	0.6732
	5	0.4373	0.4818	0.5172	0.5842	0.5520	0.6068
军事	3	0.4274	0.5043	0.5039	0.5338	0.4857	0.4944
	4	0.4436	0.5342	0.6048	0.6109	0.6044	0.6301
	5	0.4258	0.4382	0.5489	0.5757	0.5525	0.5384

从表 2.1 的对比实验结果来看，对于 4 种类型的文档在不同关键词数目和不同文摘压缩比下的文摘的 *F_measure* 实验结果的整体评价还比较理想，其中当抽取的关键词数目设置为 4 的时候，在各种压缩率下均得到了比较好的抽取效果，尤其是体育类文章，分析其原因，主要是因为本章提出的关键词抽取的方法和文摘句摘录的方法在这两类文章的关键词及文摘句的位置等因素方面得到了更多的体现，而在教育类文章上没有更多的吻合，故此我们应该在关键词抽取和文摘句摘录方面需要考虑更多的因素。

为了客观地考察本章提出的自动文摘方法的实际效果，在六种不同的文摘压缩率下，将基于抽取不同数目的关键词生成的文摘与 Edmundson 方法生成的文摘进行了比较，具体实验结果如图 2.5 和图 2.6 所示。

从实验结果看，在压缩率较小(5%和 10%)的情况下，两种方法的文摘平均召回率和平均准确率是比较接近的，但当压缩率增大以后，基于关键词抽取生成的文摘平均召回率和准确率均有明显的提高，这是因为基于词汇链的关键词抽取将词汇按照它们的语义进行了“聚合”，能够从语义分析的角度对于文档词汇及其相关特征进行归纳，是一种基于理解的、深层分析的方法，将该方法抽取的关键词用于自动文摘更有利于把握文档的主题。

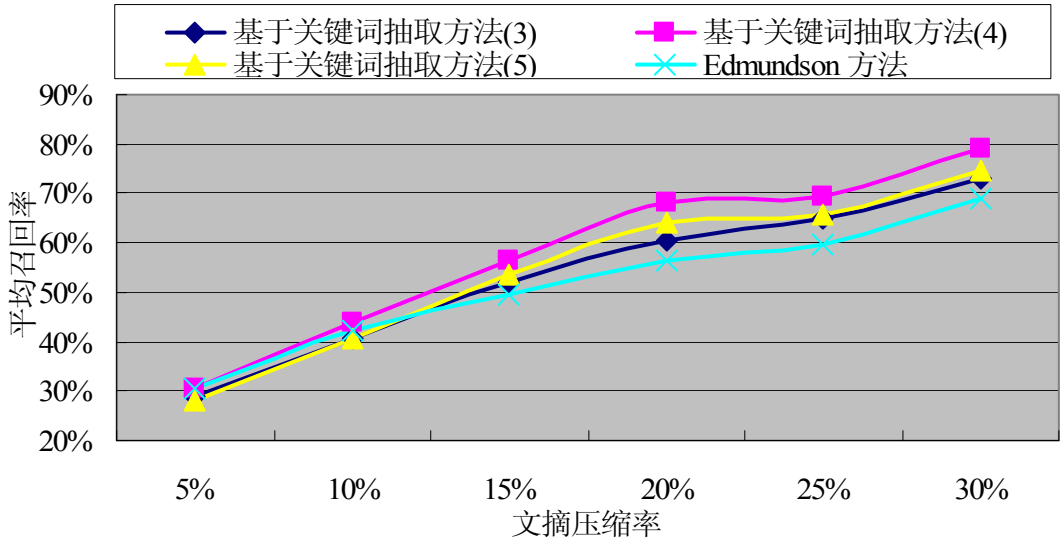

图 2.5 基于关键词抽取方法和 Edmundson 方法文摘的平均召回率比较

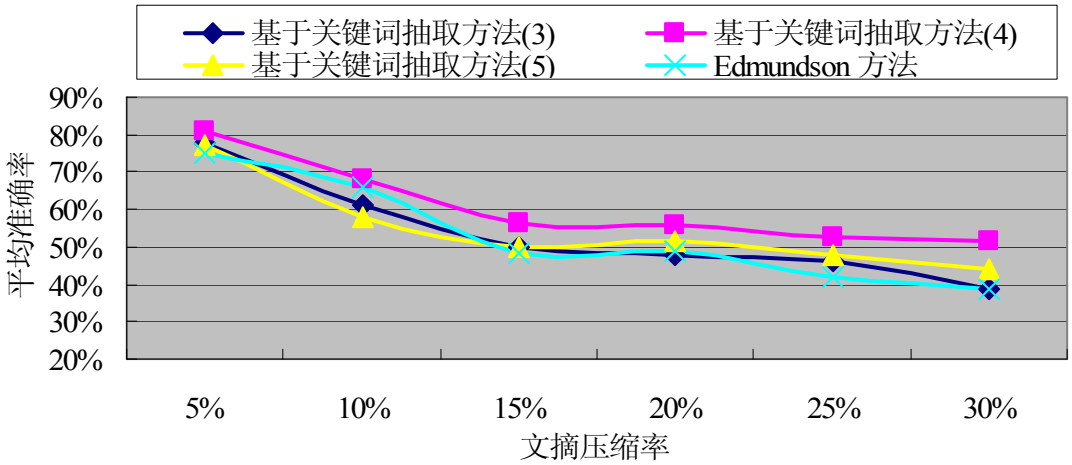

图 2.6 基于关键词抽取方法和 Edmundson 方法文摘的平均准确率比较

如果单纯从准确率来看，当压缩率较小时，文摘的准确率要明显高于压缩率较大时的准确率，这表明随着文摘长度的增长，其差异也在扩大。其实，人工摘录的情况也是如此，当文摘句很少时，摘录目标相对比较集中(一般是将文档的标题和开头段落的语句抽取出来作为摘要)；当文摘句较多时，摘录目标就变得比较分散了。

2.7 本章小结

本章介绍了《知网》和词汇链的基本概念，在利用相邻词汇共现进行未登录词识别的基础上，给出了利用《知网》为知识库构建词汇链的算法，提出了一种通过词汇链的构建进行中文关键词抽取的新方法，并给出了利用抽取出来的关键词进行当文档自动文摘的方法。

由于词汇链中的一系列词具有语义相关性，因此将文章中的词汇首先组织成词汇链，再结合词汇所在词汇链的强度、信息熵和出现位置等属性抽取关键词的方法有助于从语义的角度提高关键词抽取的准确性，从而更深层次上提高自动文摘的质量。实验结果表明，基于词汇链构建的关键词抽取方法在精确匹配测试和近似匹配测试均比词频区域法取得较好的抽取效果；在关键词抽取的基础上，将采用不同数目的关键词生成的文摘与 Edmundson 方法生成的文摘进行了比较，在召回率和准确率方面均有所提高。

但是当单文档中包含的文字信息很少，大多为图片或者广告信息时，本章提出的文摘方法就不太适用，是否可以利用网页中的链接信息来补充单个网页信息过少的问题还需要进一步研究。此外，正如聚类的类别数目的选择一样，如何自动地确定关键词抽取的个数仍然是一个难题。

第3章 基于主题聚类与语义分析的多文档文摘

3.1 引言

多文档文摘的数学模型为：

给定多文档集合 D：由 M 个文档组成的集合，每个文档 D_i 由 N 个句子，$S_i=\{S_{i1}, S_{i2},\cdots, S_{iN}, 0<i<M\}$，$S=\{S_1, S_2,\cdots, S_M\}=\{S_{ij}| S_{ij}\in D_i\}$，压缩率 p，选择子集 $S'\subseteq S$ 使 $|S'|\leqslant p\times|S|$，S' 即为多文档文摘。

从数学模型的定义可以看出，多文档集合是多个具有相同主题的文档的集合，各文档之间具有许多的共同信息，也包含与主题相关的不同信息。多文档文摘是将多文档集合中的多次出现的信息以一次出现在文摘中，其它与主题相关的信息根据重要性依次抽取的文档压缩技术[111]。

根据用户的不同需求，多文档文摘可分为问题无关的多文档文摘和问题相关的多文档文摘。问题无关的多文档文摘是对具有共同主题的多个文档的汇总，重点是消除冗余信息，以简洁全面的信息将多文档内容呈现给用户；问题相关的多文档文摘不仅汇总文档集合中的主要信息，消除冗余信息，还需要在选择文摘单元时考虑与问题相关的程度。

目前多文档文摘的技术路线主要有：基于信息融合、基于句子压缩和基于句子抽取等，其中基于句子抽取的方法因其简单、高效而倍受青睐。本章采用基于句子抽取的技术路线，并着重研究了问题无关的多文档文摘，研制了基于局部主题聚类与语义分析相结合进行文摘句抽取和润色的多文档自动文摘系统(CSA-MDSS，Multi-Document Summarization System Based on Clustering and Semantic Analysis)，该系统主要由 4 个模块组成：

(1) 局部主题的确定，将文档集拆分为句子集合，将句子表示为 *VSM* 向量，并运用 *K*-均值聚类算法进行聚类；

(2) 候选文摘句抽取，从每个类中选择分值最高的句子；

(3) 文摘句润色处理(句子冗余消除和平滑处理)；

(4) 文摘生成，将所选择的文摘句按改进的 MO 算法进行排序，组成文摘。

基于局部主题聚类与语义分析的多文档文摘系统(CSA-MDSS)的体系结构如图 3.1 所示：

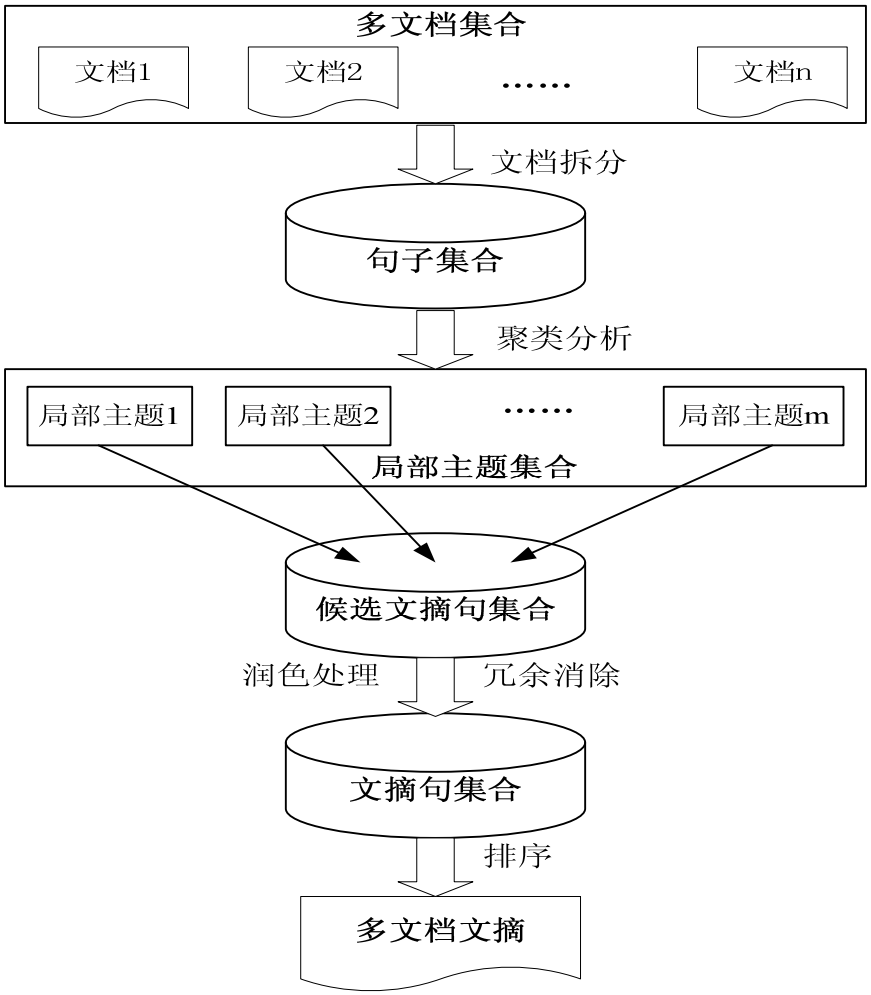

图 3.1 基于局部主题聚类与语义分析的多文档文摘系统结构

3.2 基于聚类分析的局部主题确定

3.2.1 局部主题的定义

尽管多文档集合都是围绕着一个共同主题的描述与说明，但是每篇文档描述的信息各有侧重，即使在一篇文档中也会出现不同的局部主题的描述。因此，多文档集合的主题是由不同的局部主题描述出来的，如果对这些局部主题进行识别，将原始的多文档集合转变为局部主题的集合，可以帮助文摘去除冗余信息，更好的获取原始文档中平衡信息，从而使生成的文摘更加简洁和全面。

从物理结构来看，多文档集合可以理解为文档的集合即 $D=\{d_i \mid i=1,2,\cdots,n\}$，每一篇文档可以表示为文档单元的集合，本章中采用的文档单元是句子，文档可以表示为 $d_i=\{S_{ik} \mid k=1,2,\cdots,m\}$，因此多文档集合可以看作为句子的集合，即 $D=\{S_{ik} \mid S_{ik} \in \forall d_i\}$，

其中 i 表示集合中文档号，k 表示 S_{ik} 所在文档 d_i 中的位置。

从逻辑结构来看，一个主题是由不同侧面的信息组合而成的，每一个侧面信息称之为一个局部主题（T_i）。因此，多文档集合又可以理解为由多个局部主题构成的[111]，$D=\{T_i|i=1,2,\ldots,l\}$，T_i 是一个句子集合。实际上就是将多篇相同主题的文档界限打破，按照句子表达意思的相近程度进行重新组合，表示成不同的局部主题的集合形式。将多文档集合以局部主题的形式表示出来，为突出该主题的主要信息做出了贡献，增加主题信息的鲁棒性，技术上便于对冗余信息进行处理，同时以局部主题结构表示多文档集合，可以保证原始文档信息在文摘中的平衡性，使个性化信息以同等层次表示出来，使其进入文摘中的可能性提高，从而满足多文档文摘对覆盖信息的全面性和内容的简洁性的要求。

3.2.2 句子表示

由于向量空间模型（VSM，Vector Space Model）可以将无结构的文本映射到可进行数学计算的向量空间中[57]，所以我们将文档集 D 中的所有词汇作为向量的特征空间。为了避免受到不重要词汇的消极影响，权重大于平均词汇权重的词汇才会保留在特征空间中。这样，文档集 D 拆分的句子集合中的每个句子 S_i 就可以表示成为一个向量 V_i。$S_i=\left(t_{i1},t_{i2},\cdots,t_{i\Pi}\right)$，$V_i=\left(t_{i1},v_{i1};t_{i2},v_{i2};\cdots;t_{i\Pi},v_{i\Pi}\right)$，$i=1,2,\cdots,N$。其中 N 是句子集合中句子的总数，$\prod$ 是向量空间的特征 t_i 的数目，v_{ij} 为句子集合中第 i 个句子的第 j 个特征的权重。因为整个文档集中所有句子都采用相同的特征来表示，所以句子 S_i 的向量 V_i 可以简化为：$V_i=\left(v_{i1},v_{i2},\cdots,v_{i\Pi}\right)$，本章用 TF×IDF 计算特征的权重：

$$v_{ij}^{'}=-log_2\left(1+tf\left(t_{ij}\right)\right)\times log_2\left(\left(N_j/N+0.01\right)\right) \tag{3.1}$$

其中，$tf(t_{ij})$表示特征 t_j 在句子 S_i 中的出现频率；N_j/N 是特征 t_j 的反句子频率，N_j 表示文档集 D 有多少个句子中包含特征 t_j。对 $v_{ij}^{'}$ 进行归一化得到：

$$v_{ij}=v_{ij}^{'}/\max_{i,j}\left(v_{ij}^{'}\right) \tag{3.2}$$

3.2.3 K-均值聚类

K-均值聚类[112]因其高效而受到研究人员的欢迎，其计算复杂度为 O(nkt)，其中 n 为特征空间的大小，k 为类的个数，t 为循环次数。原文档集中句子的聚类实际上就是对 N 个$\prod$维向量的聚类过程，其中 N 是从文档集中拆分出来的句子总数，$\prod$为所有词汇组成的特征空间的大小。应用 *K*-均值聚类，需要事先定义两个句子间的距离，而句

子间的距离恰好与句子间的相似度成反比，通常句子间相似性可以用句子间的向量夹角余弦来表示。句子间的相似度和距离计算如公式(3.3)和(3.4)所示：

$$Sim\left(V_i,V_j\right)=cos\left(V_i,V_j\right)=\frac{\sum_{t=1}^{\Pi} v_{it}\times v_{jt}}{\sqrt{\sum_{t=1}^{\Pi} v_{it}^2}\sqrt{\sum_{t=1}^{\Pi} v_{jt}^2}} \tag{3.3}$$

$$Dis\left(V_i,V_j\right)=1-Sim\left(V_i,V_j\right) \tag{3.4}$$

基于向量空间模型的 *K*-均值聚类算法具体实步骤如下：

(1) 选取聚类个数 *k*;

(2) 随机选择 *k* 个初始聚类中心：$Z_1(1)$, $Z_2(1)$,…,$Z_k(1)$，这里括号里的“1”表示是第一次迭代；

(3) 计算所有样本与聚类中心的距离，并按最小距离原则进行聚类：即

如果 $D_j(m)=min\{Dis(x,Z_i(m)),i=1,2,\cdots,k\}$，则 $x\in\omega_j(m)$；

(4) 利用公式(3.5)重新计算各个类的新的聚类中心向量：

$$Z_j\left(m+1\right)=\frac{1}{N_j}\sum_{x\in\omega_j(m)} x, j=1,2,\cdots,k \tag{3.5}$$

其中，N_j 为第 *j* 个聚类域ω_j中包含的样本个数。每一类以均值向量作为新的聚类中心，为使聚类准则函数 J_j 最小，利用公式(3.6)继续迭代运行；

$$J_j=\sum_{x\in\omega_j^{(k)}} Dis\left(x,Z_j\left(m+1\right)\right), j=1,2,\cdots,k \tag{3.6}$$

(5) 如果步骤(4)中的聚类中心不再变化，即 $Z_j(m+1)= Z_j(m)$，则算法收敛；否则转步骤(3)。

K-均值聚类算法的目标是能使聚类域ω中所有样本 *x* 到聚类中心 *z* 的距离最小，即使得 $J_j=\sum_{x\in\omega_j^{(k)}} Dis\left(x,Z_j\right)$ 的值最小。

K-均值聚类算法的优点是不需要知道聚类最后的目标类数，可以通过阈值确定聚类是否合适，适应了多文档集合局部主题数量不确定的特点。但是，阈值如何选择关系到最终类数 *K* 的多少，如果阈值选择的较小，最终类数就较少，否则最终类数 *K* 较大，目前，阈值的选择相对比较主观，导致聚类算法不适合移植。

3.2.4 类个数 k 的自动探测

K-均值聚类算法与其它聚类算法一样，其难以解决的问题是如何自动地确定类的

个数 k。在传统方法中，k 都要事先确定。然而，用户事先不知道多少个类对于通过聚类进行多文档文摘来说是比较合适的。武汉大学刘德喜博士[65]采用一种利用评价函数自动探测类个数的算法，该算法认为如果类的个数是正确的，那么类内句子的相似性最大，而类间句子的相似性最小，为此，我们定义了两个概念：类间离散度和类内聚合度，具体计算方法如公式(3.7)和(3.8)所示。

$$SCT\left(c_i,c_j\right)=\frac{1}{\left|c_i\right|\times\left|c_j\right|}\sum_{V_p\in c_i}\sum_{V_q\in c_j}Dis\left(V_p,V_q\right) \tag{3.7}$$

$$CHN\left(c_i\right)=\frac{2}{\left|c_i\right|\left(\left|c_i\right|-1\right)}\sum_{\substack{V_p,V_q\in c_i\\ V_p\neq V_q}}Sim\left(V_p,V_q\right) \tag{3.8}$$

其中，c_i 为聚类算法产生的第 i 个类的句子集合，$|c_i|$为句子集合 c_i 中包含的句子个数。聚类效果的评价函数为：

$$F\left(C\right)=\frac{1}{k}\sum_{c_i\in C}CHN\left(c_i\right)+\frac{2}{k\left(k-1\right)}\sum_{\substack{c_i,c_j\in C\\ c_i\neq c_j}}SCT\left(c_i,c_j\right) \tag{3.9}$$

其中，C 是类个数为 k 时的聚类结果。实际类的个数就是使 $F(C)$最大的聚类个数：

$$k^*=\underset{k\in\{2,\ldots,N-1\}}{arg\,max}F\left(C\right) \tag{3.10}$$

3.3 基于重要性抽取和平均抽取相结合的候选文摘句抽取策略

在 k 个局部主题确定后，需要解决的问题是如何从这些局部主题中抽取句子，抽取多少句子。对于多文档文摘而言，要尽量的减少冗余，并要最大程度的体现不同信息，这就要求我们从各局部主题中抽取最能代表本类的句子，但是在给定文摘的压缩比 p 后，从每个局部主题中抽取的句子数取决于具体的抽取方法。一般来说可以分 3 种抽取方法：(1)平均抽取，即从每个局部主题中抽取数目相同的句子数；(2)按照局部主题的重要性抽取，即从比较重要的局部主题中抽取较多数目的句子，反之，抽取较少数目的句子，甚至从一些不重要的局部主题中只抽取一句最有代表性的句子；(3)根据局部主题中包含的句子占整个文档集的句子总数的比例抽取句子。

在实验过程中，我们发现采用平均抽取方法的生成文摘中，存在着比较重要的内容没有充分说明，而不重要的内容占比较大的比重；采用重要性抽取会导致文摘中原来认为较重要的内容冗余，而认为不重要的内容描述地不够清楚；采用比例抽取句子的文摘会同时存在以上两个方面的不足。在对局部主题确定和多文档文摘生成进行手工分析时，我们发现以下特点：(1)有些局部主题包含的句子很多，有些局部主题包

含的句子很少；(2)总有一个局部主题中包含的句子主要在文档的开头出现，这些句子主要是对事件起因的描述；(3)总有一个局部主题中包含的句子主要在文档的结尾出现，这些句子一般是对事件的总结或评论。

基于上述分析，本章采用一个重要性抽取和平均抽取相结合的方法进行文摘句抽取，即从句子主要处在文档开头的局部主题抽取长度 16%**p***Length*(*D*)的句子，使得用户可以从多文档文摘中比较清楚地了解事件发生的起因；从句子主要处在文档末尾的局部主题抽取长度 12%**p***Length*(*D*)的句子，从而让用户知道事件发生的结果；其它局部主题一般是对事件发生过程的介绍和评论，重要性相当，所以采用平均抽取的方法从这些局部主题中抽取文摘句，具体比例的计算如公式(3.11)所示：

$$Len=(1-0.16-0.12)\times p\times Length(D)/(k-2) \tag{3.11}$$

其中，*p* 是预先设定的多文档文摘的压缩率；*Length*(*D*)是文档集的长度或者文档集包含的句子总数；*k* 是局部主题的个数；*Len* 是从局部主题中抽取候选文摘句的长度或者句子数。

3.4 基于语义分析的文摘润色处理

利用聚类分析进行局部主题识别编制的粗文摘并不能够成为一段合格的摘要，主要是以下 4 个原因：(1)前后句子之间可能不连贯；(2)句子之间的内容可能有重复，句子内部也有冗余的部分；(3)句子中可能有一些指示代词不匹配；(4)有一些从原文中摘出的句子的语气不适合放于文摘中。因此，对抽取出来的文摘句进行润色处理非常必要，文摘句的润色处理的主要目的是为了让生成的文摘能够连贯、简明、流畅，其中连贯、流畅主要指句子之间的关系，简明是指以较少的分句表达尽可能多的信息。本章涉及的文摘句润色处理主要包括两个方面：冗余消除和平滑处理。

3.4.1 改进的 MMR-SS 句子冗余消除算法

由于同一主题的文档之间存在着很多的冗余信息，因为对于同一事件，不同文档都会有与主题相关的描述，因此，如果按照传统的与主题或用户查询的相似度大小来选取文摘句，势必会使生成的文摘中信息冗余度较高。例如对于热点新闻，多家媒体会在不同时间对同一事件做不同的报道，按上述方法来选择文摘句，一方面会有大量的冗余信息，同时各家报道中的不同点也被舍弃了，使得文摘覆盖的信息过少，不能达到“文摘”的真正意义和要求。因此，对于多文档自动文摘系统来说，一个很重要

的方面就是要降低冗余，同时使信息的覆盖面更为广泛。

最大边缘相关技术(MMR，Maximal Marginal Relevance)是 Golestein 最先提出的，其最初目的是为了改善搜索引擎系统的性能[21]，主要是为了降低搜索引擎返回的文档间的冗余度。在搜索引擎中，一个文档具有边缘相关性当且仅当它与用户的查询输入具有较高相似度的同时，与已被选文档间的冗余信息最少。因为是要从整个文档集中寻找一个最合适的，所以命名为最大边缘相关。即：使所选文档在保持和主题或用户查询相关度较高的情况下，尽量降低和已选文档之间的冗余。该方法在选择下一个返回给用户的文档时，要根据下面公式计算得出最适合的文档：

$$MMR \equiv Arg \max_{D_i \in R \setminus S} \left[\lambda \left(Sim_1 \left(D_i, Q \right) \right) - \left(1 - \lambda \right) \max_{D_j \in S} Sim_2 \left(D_i, D_j \right) \right] \tag{3.12}$$

其中，Q 是用户的查询输入，$R = IR(C, Q, \theta)$是被 IR 系统选择出来的文档集合；C 是给定的一个文档集合；θ是相似度阈值(低于该值的文档将不被选中)；S 是已选出的文档集，为 C 的子集；$R \setminus S$ 是候选的文档集合，Sim_1 是候选文档与用户查询 Q 之间的相似度计算准则；Sim_2 是候选文档与已选文档之间的相似度，参数 λ 用来调节 Sim_1 和 Sim_2 的侧重方向，以选择比较合适的文档。

Goldstein 在发现利用 MMR 在搜索引擎中可以取得较好的效果后，将 MMR 技术应用到了多文档文摘中，称为 MMR-MD(Maximal Marginal Relevance Multi-Document)，它是一种纯粹的抽取式摘要方法，其基本思想与应用在 IR 系统中的 MMR 一样，旨在使生成的文摘与多文档主题的相关度较高，同时文摘的冗余度尽可能的低。

该方法是基于统计且与领域无关的，实验数据显示该方法对关于同一主题的新闻类文档进行自动文摘的效果较好，因为新闻类文档中含有更多的冗余信息，和普通的多文档文摘系统相比，MMR-MD 在降低冗余度方面有很明显的效果。

MMR 方法在多文档自动文摘中的应用体现了其可以有效降低文摘中信息冗余的优势的同时，但是也存在着一些不足之处：应用 MMR-MD 时，由于没有考虑自然语言理解和信息提取技术，得到的文摘缺乏指代的一致性，选取的段落之间缺少连贯性，甚至有语义上的模糊性。为了解决 MMR-MD 算法的上述不足，哈尔滨工业大学的刘寒磊[113]提出了基于语句级语义相似度的最大边缘相关方法 MMR-SS(Semantic Similarity based Maximal Marginal Relevance)，具体如公式(3.13)所示。该方法利用 MMR 基本理论选择文摘句时，引入语句级语义相似度计算方法来计算候选文摘句和主题以及候选文摘句与已选文摘句之间的相似度，提高相似度计算准则，同时改进原有方法，并结合其它统计信息和文章篇章结构分析知识，以达到选择最佳文摘句，提

高文摘质量的目的。

$$MMR-SS \equiv Arg\max_{P_i \in R \setminus S}\left[\lambda\left(Sim_1\left(P_i,T,D_i,D\right)\right)-\left(1-\lambda\right)\max_{P_j \in S} Sim_2\left(P_i,P_j,D_i,S\right)\right] \tag{3.13}$$

其中，Sim_1 是候选句和主题词集合(即主题)的相似度；Sim_2 是候选句和已选文摘句间的信息冗余度；D 是输入文档集合；P 是文档中任一句子，P_i 是候选文摘句，P_j 是已选文摘句；T 是主题词集合；D_i 是 P_i 所在文档；R=IR(D,P,T,L)；即：给定输入文档集合 D 和文摘长度 L，依据主题词集合，根据公式(3.13)，符合条件的可进入文摘的句子集合；S 是已选文摘句集合；R\S 是候选句子集合。

$Sim_1(P_i, T, D_i, D)$的计算如公式(3.14)所示：

$$Sim_1\left(P_i,T,D_i,D\right)=\alpha\times Sim\left(P_i,T\right)+\beta\times Score\left(P_i\right)+\gamma\times time-sequence\left(D_i,D\right) \tag{3.14}$$

其中，$Sim(P_i,T)$是候选句和主题的相似度，$Score(P_i)$是通过计算得到的候选句权值，$time\text{-}sequence(D_i,D)$是句子所在文档的发布时间和总时间间隔的时间比例，具体计算方法如公式(3.15)所示；α,β,γ 分别是权重参数。

$$time-sequence\left(D_i,D\right)=\frac{timestamp\left(D_i\right)-timestamp\left(D_{mintime}\right)}{timestamp\left(D_{maxtime}\right)-timestamp\left(D_{mintime}\right)} \tag{3.15}$$

$timestamp(D_i)$是文档 D_i 的发布时间，$timestamp(D_{mintime})$是该主题中最早发布文档的时间，$timestamp(D_{maxtime})$是该主题中最晚发布文档的时间。刘寒磊认为发布较晚的文档包含更多的新信息，所以用时间因素来对一篇文档进行加权。

由于新闻类文档中会包含大量的时间信息，MMR-SS 算法也利用了新闻文档的特征进行相似度计算和权重计算，从而实现文摘句的选择。但是，我们在研究过程中发现，即使在某些新闻文档中也不包含时间信息，而且，目前时间实体识别的准确率和召回率并不是很高。为此，面向更宽泛的通用多文档文摘，我们在局部主题划分和候选文摘句抽取的基础上，通过用改进的 MMR-SS 算法来实现文摘句的冗余消除，具体如公式(3.16)所示：

$$MMR-SS' \equiv Arg\max_{S_i \in R \setminus S}\left[\lambda\left(Sim_1\left(S_i,R,D_i\right)\right)-\left(1-\lambda\right)\max_{S_j \in S} Sim_2\left(S_i,S_j,D_i,S\right)\right] \tag{3.16}$$

其中，R 是 3.3 节中抽取的句子集合；S 是已选文摘句集合；$R\setminus S$ 是候选文摘句集合；S_i 是候选文摘句；D_i 是 S_i 所在文档；S_j 是已选文摘句；Sim_1 是候选文摘句与文档集描述主题的相似度；Sim_2 是候选文摘句与已选文摘句间的相似度；λ 是用来调节句子和主题的相关度以及句子间冗余度的侧重参数。

$Sim_1(S_i, R, D_i)$的计算如公式(3.17)所示：

$$Sim_1(S_i,R,D_i)=\alpha\times Sim(S_i,R)+\beta\times W(S_i)+\gamma\times W(D_i) \tag{3.17}$$

其中，$Sim(S_i,R)$是候选文摘句 S_i 和文档集描述主题的相似度；$W(S_i)$是候选文摘句 S_i 的权重；$W(D_i)$是 S_i 所在文档 D_i 的重要度；α，β，γ 分别是权重参数且 $\alpha+\beta+\gamma=1$。

$Sim_2(S_i,S_j,D_i,S)$的计算如公式(3.18)所示：

$$Sim_2(S_i,S_j,D_i,S)=\xi\times Sim(S_i,S_j)+\eta\times Include(S,D_i) \tag{3.18}$$

其中，$Sim(S_i,S_j)$是候选句 S_i 和已选文摘句 S_j 之间的信息冗余度；$Include(S,D_i)$是候选文摘句 S_i 所在文档 D_i 中的句子被选入文摘与文档长度的比例，添加该因子的原理是：对于一篇文档，如果该文档中已有句子被选入文摘中，那么该文档中其它句子被选入文摘的可能性就降低，这样是为了保证文摘的内容来自多篇不同的文档，使信息的覆盖面更广，而不是大部分内容都来自一篇文档，比较单调。ζ 和 η 分别是权重参数且 $\zeta+\eta=1$。

通过实验结果分析，上述公式的参数取值为：$\alpha=0.5$，$\beta=0.3$，$\gamma=0.2$，$\zeta=0.75$，$\eta=0.25$。同时实验结果表明，取 $\lambda=0.65$ 时能获得较好的文摘效果；既保证了文摘句与主题的相关性，又保障了内容上的全面性。

3.4.2 文摘平滑处理的实现

3.4.2.1 文摘句平滑处理

自动文摘中的平滑处理主要是对相邻的两个句子或多个句子进行合并、缩合、指代消解等处理。为此，首先需要对文摘句进行句法分析，生成文摘句的句法树。本章我们采用的句法分析算法是北京理工大学自然语言实验室郭庆琳[1]在TCAAS自动文摘系统中提出的基于转移网络的句法分析方法。

文摘句的平滑处理需要通过进一步调整输入句法树的结构，使文摘句的内容合理、结构清晰。在平滑处理的过程中，需要根据句子的谓词是否相同而采用不同的策略，为此，需要为每一种特定的平滑处理操作分别制定相应的规则。规则定义包括两部分：判断条件和操作体[114]。判断条件主要是语义条件，即判断两个句子的相应语义成分的数据是否相同；操作体定义了当两个句子满足判断条件时应进行的句法树调整操作。考虑到句子优化的特点，规则必须具备如下特性：

(1) 认为句子越短则优化的效果越好；

(2) 规则不可以与具体的词汇或短语直接相关；

(3) 规则在规则库中必须有序地存放；

(4) 规则库必须可扩展。

平滑处理的实现是通过对句法树中的节点进行替代、增加、合并、删除等操作，为了使句法树的结构更为合理，这些操作必须依据规则来进行。

下面是两个句子合并的实例：

(a) 他们生产塑料盆。(b) 他们生成轮胎。

两句的句法树如图 3.2 所示。

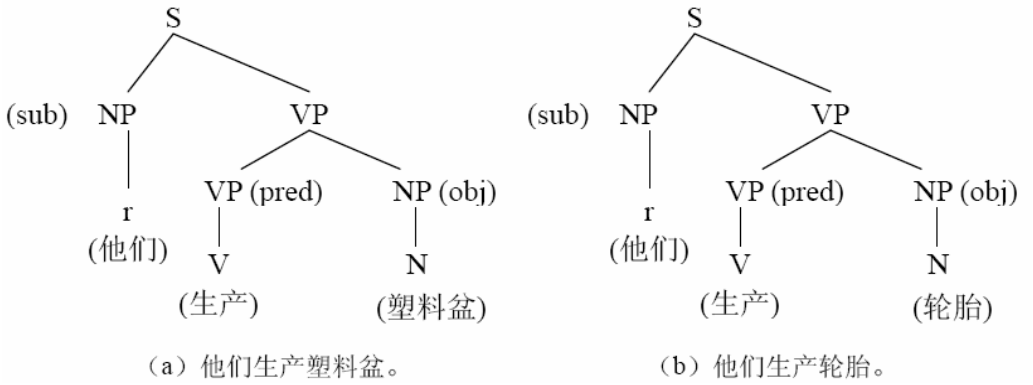

图 3.2 两个例句的句法树

由图 3.2 的句法树可以看出，两个句子只有受事不同，可以将其合并为一个具有复合受事成分的句子。合并后的句法树如图 3.3 所示。

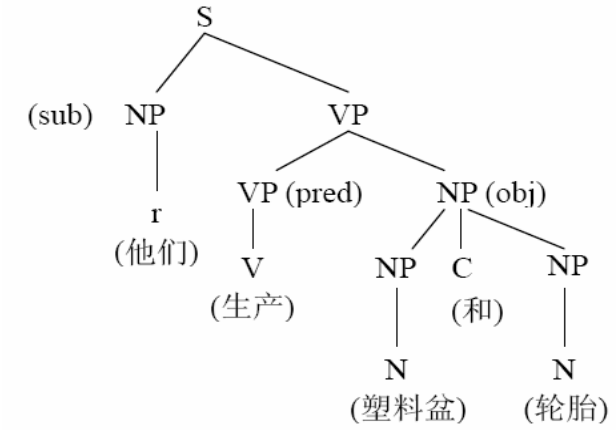

图 3.3 合并后的句法树

由合并后的句法树可以得到两句合并后的句子为“他们生产塑料盆和轮胎”。显然两个并列的句子“他们生产塑料盆”和“他们生产轮胎”不如合并后的句子“他们生产塑料盆和轮胎”在语言上简明流畅。

由于多条规则可能在两棵句法树的合并过程中同时适用，所以如何解决规则间的冲突是一个极为重要的问题。为此，我们通过对规则库中的规则进行有序组织来解决这一问题：(1)规则在规则库按照不同的优化类型区别存放，同一优化类型的规则排

列在一起；(2)将最重要的类型放在规则库的最前面，而将次要的类型放在后面，由于系统运行是按照由前至后的顺序使用规则，这便保证了最重要的规则发挥最大的作用，同时也解决了不同类型的规则间的冲突问题；(3)对每一类型尽可能细分，定义尽可能多的规则[115]。

文摘句平滑处理算法的基本思想是对相邻两个句子的句法树递归地匹配每一条规则，最终把句法树修改为更合理的文档结构树。具体算法步骤如下：

输入：句法树

输出：调整的句法树

算法：

(1) 选择一条规则；

(2) 在句法树中寻找两相邻的句法树；

(3) 若该规则与此两棵树匹配则应用之，然后重新选择句法树中的两棵相邻树；如果不匹配，选择下两棵相邻树，直至结构中不再存在与规则匹配的相邻句法树；

(4) 对所有规则执行以上步骤。

在文摘的平滑处理中，规则的制定是一个难点，需要全面考察中文的语法和语义特点，并做出精心的规划。具有可连接性的两个核心句能合并成一个扩展句，这里我们重点论述一下语句的合并规则。

如果两个句子能够合并成一个扩展句，通常需要同时满足三个原则：(1)句法共价原则[116]；(2)语义连贯原则；(3)语篇衔接原则[117]。句法共价原则是指两个句子拥有一个相同的配价成分；语义连贯原则是指两个句子表述的内容必然具有某种语义相关性；语篇衔接原则是指合并后的句子要衔接地自然顺畅，保证合并句在语篇中的可接受性。

当然在合并的过程中，句子不可避免地要进行多种变化，如配价成分的增删、提升、降格、移位等等，从而达到衔接自然的目的。在自然语言中，合并显然并不局限于单重合并，还有双重、三重甚至更多重的合并，但多重合并仍要按照单重合并的基本原则来进行。在 CSA-MDSS 系统中，我们将仅仅解决两个句子的单重合并。

3.4.2.2 文摘句连贯性处理

由于生成的文摘是直接从文档中抽取的句子组成，所以可能会拆散原文档句子间的内在逻辑关系(如并列、转折、承接和因果等)，从而造成文摘语义的不连贯。针对

出现的缺乏可读性的一些规律，我们制定了相应的规则，针对性地解决逻辑不连贯的问题。以因果关系为例，规则如下：

Clause (i)：以“因为”为起始的句子
Clause (i+1)：以“所以”为起始的句子
IF (Clause(i)在文摘句中)&&(Clause(i+l)在文摘句中)
    Continue；
IF (Clause(i)在文摘句中)&&(Clause(i+l)不在文摘句中)
    直接删除Clause (i)；
IF (Clause(i+l)在文摘句中)&&(Clause(i)不在候选输出中)
    补充Clause(i)为文摘句；

对于转折关系，转折关系要强调的内容通常在复合句的后半部分，所以如果只出现前半句的话，应视其重要程度，补充后半句、删除转折词或删除前半句；如果出现包含转折词的后半句，那么仅仅删除转折词即可。

如果原文包含了像“首先”、“其次”、“第一”、“第二”等序列词，很可能根据重要程度不能全部抽取出来，此时应该删除这些序列词；还有出现的“举例来说”通常是事例性的表述，可以直接删除；“一方面”、“另一方面”通常是并列、递进或不明显的转折，如果只含其一，应该补充另外一句。“此外”通常是补充性的说明，可直接删掉。

3.5 实验与评价

3.5.1 自动评测标准

相对于单文档自动文摘来说，多文档自动文摘的评测更具难度，因为每篇文档的侧重点都不尽相同，要把握多篇文档的主要内容，并判断哪些应该进入文摘中，组织成简短的文摘，人工做的摘要也不能达到很好的效果。为了比较客观的对多文档文摘进行评测，我们希望对自动文摘系统进行自动评测。

根据863评测中对文摘的流畅度和信息覆盖面的要求，同时由于同主题的一组文档之间存在很大的信息冗余性，因此，我们采用了信息覆盖率、文摘流利度和信息冗余度[113]三个评测参数对本章提出的基于主题聚类和语义分析生成的多文档文摘进行评价。

$$信息覆盖率=\frac{机器摘要与专家摘要匹配的句子数}{专家摘要的句子总数} \tag{3.19}$$

由于信息冗余度是指摘要中相同或相似信息出现的程度，因此，我们在计算文摘的信息冗余度时不考虑专家文摘，而是根据机器摘要中的句子信息进行计算。

$$信息冗余度=\frac{机器摘要中相互匹配的句子数}{机器摘要的句子总数} \tag{3.20}$$

$$文摘流利度=\frac{机器摘要与专家文摘匹配且顺序一致的句子数}{专家摘要的句子总数} \tag{3.21}$$

需要指出的是，上述3个公式中提到的专家摘要，不是一篇标准文摘，而是多个专家的标准文摘的并集。公式中“匹配”的含义是机器摘要中的句子和专家文摘的句子相同或相似度大于预设的阈值。由于文摘流利度主要评测的是文摘是否流畅和是否具有可读性，而文摘句(属于多个文档)的排列顺序如何确定将在下一章详细介绍，所以本章不对文摘流利度进行评测。

3.5.2 实验方法

为了验证本章提出的基于主题聚类和语义分析的多文档文摘系统(CSA-MDSS)生成文摘的性能，实验使用的测试数据是从搜索引擎和新闻网站的专题中下载的新闻文档，包括10个新闻专题集220余篇文档。4个相关语言学专业人员被要求从每个专题集中各抽取一个约300字的摘要，并将4个摘要中句子的并集作为专家摘要参与评测。

10个新闻专题集的机器摘要的压缩率被设定为5%，为了正确评价CSA-MDSS生成文摘在信息覆盖率上的表现，选择了两种常用的多文档文摘的方法与本章所提方法进行对比：TOP-N方法是对文档集合中每篇文档的第一句组合起来作为多文档文摘；MEAD方法是一种基于质心的多文档文摘。

为了评价本章提出的改进的MMR-SS进行冗余消除的有效性，实验中我们从上述10个新闻专题集中选择了前5个专题，分别在5%和10%的文摘压缩率的条件下进行机器文摘，同时采用两种冗余消除算法(基于相似度进行冗余处理和刘寒磊提出的基于语义的MMR-SS算法)与改进的MMR-SS算法进行了对比实验。

3.5.3 实验结果与分析

TOP-N，MEAD和CSA-MDSS三种多文档文摘方法在10个新闻主题集上进行5%压

缩率生成的摘要与专家摘要进行信息覆盖率评测的实验结果如表3.1所示。

表 3.1 多文档文摘信息覆盖率自动评测结果

主题编号	文摘方法	信息覆盖率	主题编号	文摘方法	信息覆盖率
	TOP-N	0.598		TOP-N	0.445
1	MEAD	0.634	6	MEAD	0.770
	CSA-MDSS	0.701		CSA-MDSS	0.808
	TOP-N	0.518		TOP-N	0.506
2	MEAD	0.628	7	MEAD	0.658
	CSA-MDSS	0.656		CSA-MDSS	0.674
	TOP-N	0.553		TOP-N	0.572
3	MEAD	0.783	8	MEAD	0.689
	CSA-MDSS	0.868		CSA-MDSS	0.721
	TOP-N	0.547		TOP-N	0.602
4	MEAD	0.644	9	MEAD	0.754
	CSA-MDSS	0.658		CSA-MDSS	0.827
	TOP-N	0.607		TOP-N	0.294
5	MEAD	0.676	10	MEAD	0.661
	CSA-MDSS	0.736		CSA-MDSS	0.541

从表3.1的对比实验结果可以看出，对于10个新闻主题，由CSA-MDSS多文档自动文摘系统生成的文摘在信息覆盖上要优于TOP-N和MEAD方法，信息覆盖率达70%以上的共有6篇，占整个文摘数量的60%，分析其原因，主要是因为首先利用聚类技术对整个文档集进行局部主题的确定，然后根据其位置和重要性等特征从每个局部主题中抽取文摘句，保证了文摘尽量多地包含文档集的每个细节，当然这也跟我们引入MMR技术有关，消除文摘句中表达相同或相近的语句，让更多的句子进入文摘，从而进一步提高了文摘的信息覆盖率。但是，在第10个主题中，CSA-MDSS的信息覆盖率表现不理想，甚至低于MEAD方法12个百分点，我们通过人工分析发现，主要因为是这类网页文档比较特殊，文档的开头和结尾并不是对事件的描述和评论，而是一些导航和广告链接信息，而我们的文摘将这些信息大量的放入了摘要中，从而导致摘要的信息覆盖率很低。

表3.2多文档文摘信息冗余度自动评测结果

主题编号	冗余消除算法	信息冗余度	
		5%	10%
1	句子相似度	0.359	0.461
	MMR-SS	0.252	0.406
	改进的MMR-SS	0.218	0. 359
2	句子相似度	0.339	0.424
	MMR-SS	0.227	0.409
	改进的MMR-SS	0.189	0.321
3	句子相似度	0.361	0.432
	MMR-SS	0.269	0.358
	改进的MMR-SS	0.245	0.342
4	句子相似度	0.385	0.478
	MMR-SS	0.297	0.395
	改进的MMR-SS	0.261	0.316
5	句子相似度	0.281	0.337
	MMR-SS	0.153	0.282
	改进的MMR-SS	0.122	0.236

从表3.2的实验结果可以看出，不管压缩率是5%还是10%，改进的MMR-SS算法进行冗余消除的效果都要优于句子相似度和MMR-SS算法，分析其原因，三种冗余消除算法中都会用到句子相似度算法，但是MMR-SS和改进的MMR-SS算法不仅仅计算句子与主题的相似度，还考虑了已选文摘句与候选文摘句的关系，以及文摘句和文摘句所处文档的重要性等等一些特性，所以MMR-SS和改进的MMR-SS算法进行冗余处理的性能要好于仅仅使用相似度计算的算法；MMR-SS算法中融入了文档的时间信息，又由于目前对时间表达式的识别率还不高，如果将没有正确识别的时间信息用于相似度计算反而会降低冗余消除的效率，所以本章采用的MMR-SS算法在计算相似度时，在没有考虑时间信息的基础上增加了文摘句所处文档的权重，从对比实验结果来看，改进的MMR-SS算法用于冗余处理的性能要略好于MMR-SS算法。

用改进MMR-SS算法进行冗余消除的文摘具有更强的概括能力，能够抓住文章的主要内容，包含的信息较多，且每篇文档中的特有信息也基本都能够包含到文摘中，

达到了信息的过滤和融合；内容相似或相同的句子确实只出现了一次，降低了冗余度；得到的文摘比较流畅，可读性较好，能使读者清楚了解事件的整个发展过程。

3.6 本章小结

本章首先给出了基于主题聚类和语义分析的多文档文摘系统(CSA-MDSS)的体系结构，然后给出了局部主题的定义和多文摘集合句子的表示，通过对局部主题个数的自动探测，采用K-均值聚类算法对句子进行聚类，从而形成多文档集合的局部主题。在局部主题确定的基础上，提出了一个重要性抽取和平均抽取相结合的方法进行文摘句抽取的方法。在候选文摘句抽取的基础上，提出了改进的MMR-SS句子冗余消除算法和基于语义分析的文摘平滑处理和连贯性处理。实验结果表明本文的方法优于MEAD和TOP-N的覆盖算法，从而说明基于局部主题的多文档文摘方法在内容的平衡性及内容覆盖上具有一定的优势。

本章还通过信息覆盖率和信息冗余度两个评测标准对CSA-MDSS多文档自动文摘系统进行自动评测的实现，总结了影响多文档文摘自动评测的3个因素：

(1) 标准摘要的制定。虽然标准摘要被认为是“理想文摘”，但真正要达到“理想”的目标是非常难的。即使是专家制定的文摘，不同专家制定的文摘也不尽完全相同，这对自动文摘的评测结果有一定的波动影响。

(2) 摘要的长度。一般来说，在生成文摘时，或者是按照文档集长度的百分比或者是按预先设定的长度抽取文摘句，这样可能使得生成的机器文摘会有长度偏差。

(3) 相似度计算。由于自动评价中，有很多地方是根据两个句子之间的相似度值来判断两个句子是否匹配，所以相似度计算结果是否准确，直接影响到评测结果。

通过聚类分析确定的局部主题不仅方便对多文档集合的结构进行分析，还可以应用于其它的自然语言处理领域。在下一章中，我们将介绍如何将局部主题间的内聚信息应用于文摘句排序过程。

第4章 基于内聚度的多文档文摘句排序

上一章主要介绍了如何抽取构成文摘内容的文摘句。在获得文摘句后，还需要考虑其在文摘中的先后顺序。文摘句之间存在多种排列，假如有 m 个文摘句，其排列共有 $m!$种之多，如何选择文摘句的排列顺序将影响到文摘的质量(特别是流畅性、一致性和逻辑性等等)，直接关系到用户是否可以正确理解原文的内容。

(A) The UN found evidence of rights violations by Hun Sen prompting the US House to call for an investigation. (*APW19981027.0491, 2*)

(B) The three-month governmental deadlock ended with Han Sen and his chief rival, Prince Norodom Ranariddh sharing power. (*APW19981113.0251, 1*)

(C) Han Sen refused. (*APW19981016.0240, 1*)

(D) Opposition leaders fearing arrest, or worse, fled and asked for talks outside the country. (*APW19981016.0240, 1*)

(E) Cambodian elections, fraudulent according to opposition parties, gave the CPP of Hun Sen a scant majority but not enough to form its own government. (*APW19981022.0269, 7*)

(F) Han Sen guaranteed safe return to Cambodia for all opponents but his strongest critic, Sam Rainsy, remained wary. (*APW19981118.0276, 1*)

(G) Chief of State King Norodom Sihanouk praised the agreement. (*APW19981124.0267, 1*)

图 4.1 DUC2004 数据集中的一篇文摘

图 4.1 中，如果按照由 A~G 的顺序排列句子，所构成的文摘前后逻辑性就非常差，有碍读者对文摘内容快速正确的理解[4]。

在单文档自动文摘中，文摘句的排序相对比较容易，最为常用的方法就是根据其在原文档中的位置来排列。对于用抽取方法得到的文摘句，可以在原文档中找到其对应的句子，文摘句在原文档中如果排在前面，则在文摘中也排在前面，否则排在后面。但是，单文档文摘中的句子排序方法并不适用于多文档文摘，因为多文档文摘中的句子并非来自一个文档，不同的句子可能来自不同的文档，此时不仅要考察来自同一文档的文摘句在文档中的内部关系，还要兼顾来自不同原文档的文摘句间的关系[118]。

以往的多文档文摘的研究工作大多都集中在文摘句的选择上，对文摘句的排序研究得较少。在本章中，首先对文摘句的排序的必要性进行了验证，分析了 MO 算法和 CO 算法的工作原理以及它们的不足之处，然后提出了一种将局部主题间的内聚度与 MO 方法相结合进行文摘句排序的改进算法，本章最后通过人工评价、ROUGE 评价和流利度评价三个评测方法对三个算法进行了对比评测。

4.1 文摘句排序的必要性

一个完整的文摘系统包括两大部分：一部分是使原始文档中的重要信息被抽取出来的文摘句的抽取模型，另一部分是使读者读起来更流畅的文摘句的排序模型。虽然多文档文摘的排序问题还没有引起人们足够的重视，但是我们清楚认识到了文摘句的顺序对读者正确理解原文起到至关重要的作用。哈尔滨工业大学的秦兵教授[111]通过一个简单的实验对此进行了验证。

实验的语料是由抽取的文摘句随机的排序形成的文本，作为该多文档集合的文摘提供给用户。测试人员是由没有看过原始文档的语料的人员组成的。选择 10 组多文档集合的文摘，经过文摘句的抽取，形成无序的文本，让测试人员对其打分。为了定性的说明问题，将文摘的质量粗略划分为 3 个等级：差(Poor)、一般(Fair)、好(Good)。然后分别对这 10 组文摘进行人工排序，使其按照原始文本表达的意图输出，经过测试人员的重新打分，结果如表 4.1 所示。

表 4.1 10 组文摘结果排序前后打分情况对照表

文档编号	文摘打分	人工排序后文摘打分
文档1	一般	好
文档2	差	差
文档3	一般	好
文档4	差	差
文档5	一般	好
文档6	差	一般
文档7	好	好
文档8	差	一般
文档9	差	一般
文档10	一般	好

由表 4.2 可以看出，经过排序质量好的文摘数从 1 个达到了 5 个；质量差的文摘数经过排序从 5 个降为 2 个。当然由于文摘是对原始文档集合的信息的抽取，所抽取的句子不一定完全概括原始文档的信息，造成经过人工排序后仍然会有少量文摘不能很好的理解。从表 4.2 中可以定性地看到，文摘句的排序对用户理解原文的意图起到很重要的作用。

表 4.2 10 组文摘排序前后人工评价汇总

	好	一般	差
排序前	1	4	5
排序后	5	3	2

既然文摘的排序对正确理解原文档的意思表达起到了非常大的作用，同时也看到不同的人对排序问题的理解也不尽相同，从中可以观察到文摘句排序的个性和共性的特征。以“乌干达事件”为例，经过文摘系统抽取的 10 个文摘句如图 4.2 所示。

A：中国日报网站消息：乌干达一名地方议员 2 月 22 日表示，数十名叛乱武装分子利用步枪、迫击炮和火箭推进式手榴弹袭击了乌干达北部一个难民营，造成 192 人死亡和另外数十人受伤。

B：当地一个传教士在接受电话采访时透露，Abia 难民营共居住着约 1 万乌干达平民，袭击事件发生后他探访了这个难民营并得知，至少 46 人当场死亡，8 人在附近的医院死亡，另外还有70 多人受伤。

C：以苏丹南部地区为基地的“圣灵抵抗军”18 年来不断骚扰乌干达北部地区，甚至把战火扩大到东部地区，目前已经造成大约一百万人逃离家园无家可归。

D：营地由当地的警察部队负责看守，但参加这次的袭击的叛军人数远远草果警卫的人数，而且他们的活力更强，以致发生了这次惨剧。

E：该发言人还称政府军徽在同一地区继续对叛军展开行动，他说：“叛军在这一地区还有另外一个相关联的据点，政府军会继续对他们进行追击。”

F：十几年来，圣灵抵抗军一直试图推翻穆塞韦尼总统的统治，但他们多数时候都是在袭击平民以夺取食物或是绑架儿童充当士兵。

G：中国日报网站消息：罗马天主教牧师赛巴特·阿耶莱 2 月 22 日说，乌干达北部的反政府武装“圣灵抵抗军”的数十名成员一天前袭击了一个难民营，并打死了 100 人。

H：在乌干达政府军的重兵围剿下，这支反政府武装力量近年来被不断消弱，已从原来的几千人减少到目前的数百人。

I：“圣灵抵抗军”反抗乌干达政府已经长达 17 年，这是最近几年来该叛乱组织发动的最严重的一次袭击。

J：据报道，这名发言人表示同政府军发生战斗的叛军分别从属于乌干达北部的 2 个军阀，也有 1 名政府军士兵在战斗中身亡。

图 4.2 “乌干达事件”文摘句集合

以上文摘句分别通过不同的读者对其进行了排序，结果如表 4.3 所示。

表 4.3 同一篇文摘不同人工排序结果

人员编号	文摘句排序
1	GABDJECIFH
2	AGCBHIFDEJ
3	AGBJDEICFH
4	AGBCIFDHJE
5	GBDAJECFIH
6	AGBDICFHJE
7	ACGBDFHJEI
8	AGBJEDCIFH
9	AGBDCFIJEH
10	GABDFCHIJE

通过对表 4.3 人工排序结果的分析，我们可以发现：

(1) 尽管所有的排序并不完全一样，但还是有规律可循的，如部分句子排序的一致性等，可以看到在这十个排序中都存在 A->C->E 这样一个顺序，九个排序中存在 A->B->C->E；

(2) 文摘排序的非唯一性，给采用机器学习方法和统计学方法带来了挑战。

(3) 不同的人对文摘的排序是不同的，说明文摘的理想排序并不是唯一的；

通过文摘句排序必要性的验证实验，可以从不同的排序结果中找到共性的东西，用来指导今后文摘的排序。总的来讲，文摘句的排序方法大致分为两种， 一种是基于形态学的方法，另一种是基于机器学习的方法。

基于机器学习的方法，实际上是根据已经建立的标准的人工文摘的排序获得一定的特征，学习由这些特征所确定的排序的方法，从而指导任意的文摘句排序。该方法也存在不足：(1)语料库的建立，构建语料库代价较大；(2)不同的人对同一篇文档生成的文摘是有差别的，存在一定的主观性，这样的条件不适合统计机器学习，所以本章只讨论通过形态学的方法得到文摘句的排序。

4.2 相关研究

2004 年，Okazaki 教授[118]介绍了一种使新闻文摘具有流畅结构的方法。该方法的

基本思路是：影响文摘流畅的原因是有些文摘句缺乏必要的先行句子，也就是说先行句子不明确使得文摘在内容上出现了跳跃，从而使文摘的流畅性变差。解决的方法是在“年代顺序法”得到的排序结果之上，根据话题的分割与对齐找到缺乏先行内容的文摘句，然后在原文档中找到合适的先行句并将其安置在相应文摘句前。

美国哥伦比亚大学的 Barzilay 教授[119]指出了句子的顺序对文摘可读性的影响，并给出了用于多文档文摘的句子排序策略。根据 Barzilay 的实验，一些朴素的方法过于简单且鲁棒性不够。如 MO 算法(Majority Ordering)根据各个原文档对顺序的投票，选择多数原文档赞成的顺序。当原文档的顺序有较高的一致性时，这种方法能够产生很好的排序结果。“时间顺序法”CO 算法(Chronological Ordering)是根据原文档出版或发布日期的顺序对文摘句进行排列。这种方法对基于事件的原文档有较好的效果，但如果原文档中的信息不是基于事件的，则参照发布时间得到的排序结果并不理想。在研究结果“话题相关的句子倾向于排列在一起”的基础上，Barzilay 提出了一种新的方法，该方法首先按话题将话题相关的句子放在一组，然后再按时间信息对各组进行排序。人工对重排序结果的评测显示，这种组合方法较年代顺序法排序相比有一定的改善，但是文档中的时间特征并不总是可得的。

2003 年，Lapata[120]提出了一种基于无监督概率模型的句子排序方法，而没有采用传统方法中的“年代顺序”。其基本出发点是：在一篇文档中，某个句子的出现与其前一句直接相关，也就是由其前一个句子决定；句子可通过一系列特征来表达，而从一个句子转移到其后一个句子的转移概率可以用这些特征集上的笛卡尔积来衡量。在对句子的表达上，他选择了动词、句词和依存关系。这样，整篇文档就可用马尔可夫模型[121]HMM(Hidden Markov Model)表示。给定多个文摘句，有多个可能的排序结果，每个排序结果对应一个概率，其中概率最大的排序结果就是最终的排序结果。对于排序模型的评测，Lapata 选择了模型排序结果与人工排序结果之间的 Kendall 队列相关系数。Barzilay 教授[122]对 Lapata 算法进行了改进，研究了内容结构对话题表示及话题之间转移的作用，并将这种内容结构模型应用于句子排序和抽取型文摘。这种内容结构实际上是 HMM，其中状态对应于信息特征的类型，状态的转移表达了这些信息在句子间的转移情况。

2006 年，Bollegala 提出了一种类似于层次聚类的自底向上的句子排序方法[123]，考察文档集段落(由一个或多个句子组成)两两之间的先后关系及这种先后关系的强度，对先后关系强度最大的两个段落进行组合得到一个段落，重复这个操作直到最后

组合成一个段落。对于段落 *A* 和 *B*，该方法定义了 4 种先后关系判定的量化准则，分别为：话题关系、年代顺序关系、前驱关系和后继关系，并视两文本段量化后 4 种关系为 4 维向量空间中的一个点，并用支持向量机 SVM 对该点进行分类：*A* 在 *B* 前(＋1)，或 *B* 在 *A* 前(－1)。而 *A* 和 *B* 先后关系的强度则用该点归于＋1 或－1 的概率估计值来表达。

国内，哈尔滨工业大学的秦兵教授于2004年通过借鉴单文档文摘的排序思想以及利用多文档集合的特征，提出了基于参考文本框架，基于文摘句位置参数以及基于文本框架与文摘句位置参数相结合的排序方法[170]，有效地解决了文摘句的排序问题，并提出了一种多文档文摘句排序的新方法，既基于顺序比较、基于*n-gram*的比较方法以及两种方法相融合的流利度评价方法。武汉大学的刘德喜博士提出了一种文摘句排序混合模型[171]。该模型综合了文摘句之间的四种关系：位置关系、时间关系、依赖关系、话题关系，以句子为节点、句间关系为边，构建句子优先关系有向图并通过对已有的PageRank方法进行改进，对优先关系有向图中的各节点进行排序。

研究人员对文摘句的排序提出了很多有益的探索，所提的方法确实能够提高多文档文摘的可读性，但其中也存在一些亟需解决的问题。Barzilay 的方法注意到了句子的时间和句子所属话题之间的关系，却忽视了句子在原文档中的位置关系。下面我们将详细介绍常用的两种文摘句排序算法(CO 算法和 MO 算法)的算法思想和存在的问题，并将这两种算法的排序结果与本章提出的改进算法进行对比实验。

4.3 CO 算法

CO 算法依据局部主题发布时间的先后对文摘句进行排序，主要应用于包含时间信息且具有相同或相近主题的新闻文档集合的摘要系统。这些新闻文档集合是指同一主题事件下不同文档的集合，特点是文档之间具有很多的共同信息，各个文档中包含与主题相关的不同信息的文档集合。新闻报道中一般先交代事件的背景，然后是事件的后续追踪，最后是评论报道。基于这一规则，根据事件发生的先后顺序对文摘句进行排序应该是一个可以接受的方法。

4.3.1 算法描述

对文摘句的排序问题可以归结到如何对局部主题进行排序，每个文摘句在最终摘要中的位置由其所属局部主题的位置确定，这样就需要为每个局部主题进行时间标

注。文献[124]中假设整个新闻文档集中每个新闻文档都标识了由日期、小时和分钟组成的出版时间，并且不存在相同时间标识的两个文件。根据上述假设，采用 CO 算法的文摘句排序步骤如下[123-125]：

（1） 对局部主题中的每个句子进行时间标注，标注的时间为该句子所在文档的发布时间；

（2） 将局部主题的所有句子中最早的标注时间作为该局部主题的标注时间；

（3） 如果两个局部主题的标注时间相同，则认为这两个局部主题的第一次出现是在同一文档中，它们的排序与其在该文档中出现顺序一致；

（4） 文摘句的顺序由其所在局部主题的标注时间的相对位置决定。

4.3.2 存在问题

CO 算法用于中文多文档文摘时存在如下不足：

（1） 每个局部主题的标注时间过分依赖其下的每个句子的标注时间，这样对局部主题划分精确性的要求特别高；

（2） 大多数中文新闻文件的发布时间并没有精确到小时和分钟，更多的只是提及了发布日期，这给局部主题标注时间的比较带来困难；

（3） 当两个局部主题的标注时间相同时，它们的第一次出现并不一定在同一个文档中，即 CO 算法描述步骤(3)中的假设并非成立；

（4） CO 算法仅仅适用于文档中带有时间信息的多文档摘要系统。

4.4 MO 算法

定义 4.1 A 和 B 是两个局部主题，$A \succ B$ 表示如果 A 的文摘句 A_i 和 B 的文摘句 B_j 出现在同一个文档中，则文摘句 A_i 总位于文摘句 B_j 之前。

定义4.2 A和B是两个局部主题，当$A \succ B$不成立时，用$A \propto B$表示文档集中满足$A \succ B$的文档数。

对于任意两个局部主题A和B，如果$A \succ B$成立，那么A中的文摘句在摘要中应该位于B的文摘句之前，进而假设各局部主题之间的排列顺序存在传递关系(若$A \succ B, B \succ C$，那么$A \succ C$ 成立)，则最终摘要将是一个$(A,B,\ldots,N)$的线性排列。但要求任意两个主题在文档集中保持一致的顺序并不现实，例如，有三个局部主题A、B 和C，存在这样的关系：$A \succ B$，$B \succ C$，$C \succ A$，可以看出(A,B,C)和(C,A)之间存在矛盾，这三个局部

主题的顺序间不具有传递性，从而单纯使用上述方法并不能获得文摘句的线性排列。

为了解决上述问题，Barzilay[124]借鉴了单文档文摘中文摘句排序的思想(即按照原始位置信息对文摘句排序)，提出了根据局部主题相对位置的多数原则进行文摘句排序的MO算法，即对于A和B两个局部主题，如果 $A \propto B > B \propto A$，则$A$在摘要中排列在$B$之前，反之，$B$在摘要中位于$A$之前。

4.4.1 算法描述

在利用聚类分析对文档集进行局部主题的确定和对每个局部主题进行文摘句抽取后，为每对局部主题A和B计算两个计数：C_{AB}和C_{BA}，其中$C_{AB}=A \propto B$，$C_{BA}=B \propto A$。以局部主题为顶点，C_{AB}和C_{BA}作为连接顶点的边的权重，这些顶点和边可以构成一个有向图，如图4.3所示。每个局部主题在摘要中的位置可以由有向图中一条最优路径P来决定，这条路径满足：(1)遍历每个顶点一次且仅一次；(2)该路径具有最大权重。

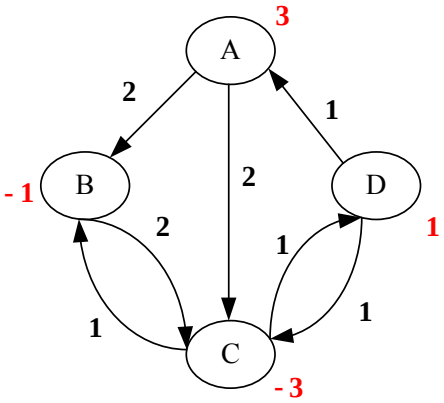

图 4.3 局部主题关系有向图

由于寻找该最优路径是一典型的NP问题，故Barzilay[67,119,124]提出了近似寻找最优路径P的算法，具体步骤如下：

（1） 设置初始路径P为空；

（2） 计算每个顶点的权值(入边权重和与出边权重和的差值)；

（3） 选择权值最大的顶点，将其插入路径P中并从有向图删除该顶点及与其相连的所有边，重新计算剩余顶点的权值；

（4） 如果有向图不为空，转至步骤(2)继续迭代运行，否则结束，路径P就是各局部主题在最终摘要中的位置；

路径 P 决定了每个局部主题在摘要中的位置，则各文摘句在摘要中的位置与其所属局部主题在路径 P 中的位置保持一致。

4.4.2 存在问题

在实验过程中，我们发现MO算法整体表现一般，仍存在一些不可忽视的问题，具体问题如下：

（1） 在寻找路径过程中，当遇到两个顶点的权值相同时，MO 算法不能提供足够的约束来决定选择哪个顶点更好，而只是随机选择其中一个，这样做比较盲目；

（2） 当某两个局部主题不同时在文档集中任何一个文档出现的时候，MO 算法不能很好解决它们的排序问题，如图 4.3 中 B 与 D 的排序；

（3） 当各局部主题在文档集中的位置相对固定时，用 MO 算法排序生成的摘要可读性较好，而当局部主题之间的相对位置变化频繁时，排序生成的摘要连贯性和可读性都较差。

4.5 改进的 MO 算法

当同一个局部主题在不同文档中出现的相对位置频繁发生变化时，MO算法排序生成的文摘可能会出现不流畅、不太容易被理解的情况。如图4.4所示，T_1、T_2、T_3和T_4是输入文档集，A_1、A_2、A_3和A_4属于同一个局部主题A，同样，B、C和D分别是其它几个局部主题。S_1是基于MO算法生成的摘要，S_2是人工排序生成的摘要。从图4.4可看出，虽然S_1和S_2中只有B和C两个主题的出现次序发生颠倒，但就整个摘要的效果而言，S_2的排序效果要明显优于S_1。

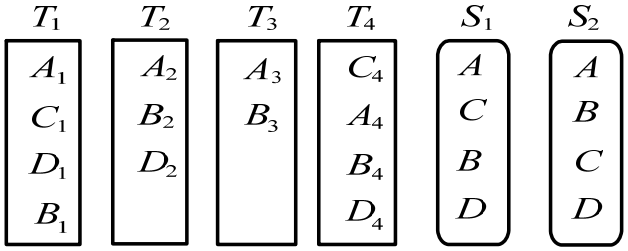

图 4.4 MO 算法排序与人工排序的结果比较

在对图4.4所示的文档集进行局部主题确定和对每个局部主题进行文摘句抽取的基础上，利用MO算法进行文摘句的排序，排序过程如图4.5所示。从图4.5中可以发现，MO算法每次从有向图中选择一个权值最大的顶点(即局部主题)，便将该顶点及与其相连的所有边从有向图中删除，并重新计算剩余顶点的权值。当一个顶点被选择后，下一个顶点的选择只与剩余的顶点之间的关系有关，与已选顶点无关，这样容易使得一些内聚程度不高的甚至文档集中根本就不存在该顺序的局部主题在摘要中处于相

邻位置，从而导致摘要的内容表达显得比较跳跃而不易于理解。例如图4.4所示文档集的4个文档中有3个文档存在出现顺序相连的A和B，某种程度上可以说明这两个局部主题是紧密相关的，因而在摘要中将A和B置于相邻的位置应该是比较理想的排序结果，这正与人工排序的结果一致，但MO算法生成的摘要S_1中A和B并不相连。

根据前面分析知道，MO算法在排序过程中仅仅考虑了局部主题间的先后关系，没有考虑到它们之间的内聚程度，使得一些关联紧密的局部主题的文摘句在摘要中处于离散的状态，生成的摘要不连贯和不清晰。为此，在MO排序算法的基础上，我们提出了将局部主题间的内聚度融入到MO排序过程中的方法，具体步骤如下：

（1） 使用 MO 算法构建局部主题关系有向图(在 4.4.1 节的已详细介绍)；

（2） 计算两两局部主题间的内聚度，即紧密关联程度；

（3） 结合局部主题间内聚度和MO算法对文摘句进行排序。

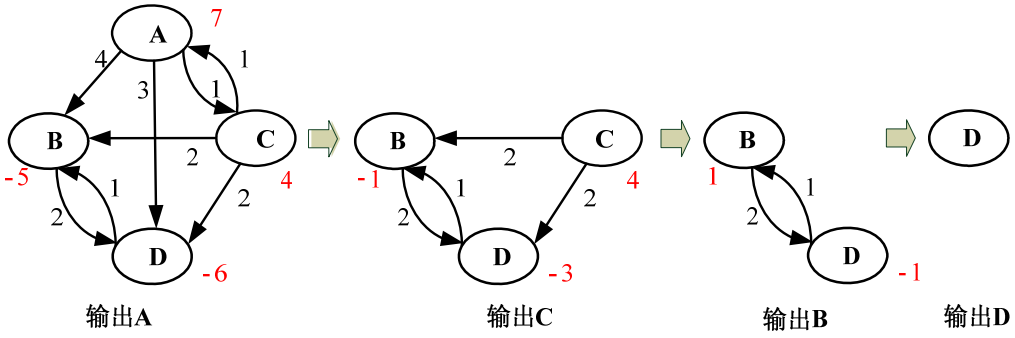

图 4.5 MO 算法的排序过程

4.5.1 内聚度计算

我们在对大量的人工排序分析发现：虽然对同一篇文章每个人给出的排序结果会不同，但对这篇文章的多个排序中有些句子会总是出现在一起。这些句子虽然不是都以同一顺序出现，但它们会有着相邻的关系(这些句子的集合称为块)，同时我们发现每个块都是由一些话题相关的句子组成的。换句话说，一个好的排序结果是由话题相关的句子构成的块组成的。将话题相关的句子聚到一起这种想法就叫内聚，内聚度越高的两个句子在文摘中越不能将其分开，否则就会影响到文摘的可读性。内聚度和排序结果的好坏间有着紧密的联系，鉴于这种想法我们将内聚度融入到排序策略中。

假设$A(A_1,A_2,\ldots,A_n)$和$B(B_1,B_2,\ldots,B_m)$是两个局部主题，其中A_i和B_j分别是局部主题A和B中包含的句子。当句子A_i与B_j出现在同一文档中时，用A_iB_j=1表示，否则A_iB_j=0；当句子A_i与B_j出现在同一文档，并且A_i与B_j先后出现在同一段落或相邻段落中时，用

$A_iB_j^+$=1表示，否则$A_iB_j^+$=0。$\#AB=\sum_{i=1}^{n}\sum_{j=1}^{m}A_iB_j$ 表示局部主题A和B中(A_i，B_j)句子对出现在同一个文档的个数；$\#AB^+=\sum_{i=1}^{n}\sum_{j=1}^{m}A_iB_j^+$ 表示局部主题A和B中(A_i，B_j)句子对出现在同一个文档并且A_i和B_j先后出现在相同或相邻段落中的个数。

局部主题 A 与 B 之间的内聚度用 $R_{AB}=\#AB^+/\#AB$ 来表示，同样，B 与 A 之间的内聚度用 $R_{BA}=\#BA^+/\#BA$ 来表示。当 $R_{AB}=0$ 时，说明局部主题 A 与 B 之间无内聚关系；当 $R_{AB}=1$ 时，说明局部主题 A 与 B 之间内聚关系紧密。因此，当 R_{AB} 大于设定阈值时，则认为 A 与 B 之间是内聚紧密相关的。在对 540 个不同主题的中文文档集进行内聚度计算和分析观察的基础上，文中采用经验值 0.7 为阈值。通过该方法计算可以获得文档集中每对局部主题之间的内聚度，从而生成一个 $N\times N$ 的非对称内聚度数组，N 是局部主题的个数。图 4.4 所示的文档集中各局部主题的相互内聚度数组为

$$\begin{array}{c} \\ A \\ B \\ C \\ D \end{array}\begin{array}{c} \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{bmatrix} 1.00 & 0.75 & 0.50 & 0.00 \\ 0.00 & 1.00 & 0.00 & 0.67 \\ 0.50 & 0.00 & 1.00 & 0.50 \\ 0.00 & 0.33 & 0.00 & 1.00 \end{bmatrix} \end{array}$$

4.5.2 文摘句排序

由于文摘句的位置与其所属局部主题在摘要中的位置一致，因此，文摘句的排序归结为对局部主题的排序。用MO算法建立局部主题关系有向图后，可以通过以下步骤求得局部主题的排序路径P：

(1) 设置初始路径P为空；

(2) 计算每个顶点的权值；

(3) 从有向图中选择权值最大的顶点i(即局部主题)，将其插入路径P中并从有向图中删除该顶点及与其相连的所有的边，并重新计算剩余顶点的权值；

(4) 计算i与所有剩余顶点间的内聚度，并取其中的最大值，顶点j与其具有最大内聚度R_{ij}，如果R_{ij}小于阈值，转至步骤(3)；

(5) 将顶点j插入路径P中的顶点i之后(如果步骤(4)中最大值是R_{ji}，则插入到顶点i之前)，并从有向图中删除顶点j及与其相连的所有边，重新计算剩余顶点的权值；

(6) 如果有向图为空，则输出路径P，否则，将j作为当前顶点，转至步骤(4)；

(7) 根据路径P对文摘句进行排序。

在对图4.4所示的文档集进行局部主题抽取的基础上，利用上述的MO改进算法进行局部主题排序过程如图4.6所示。

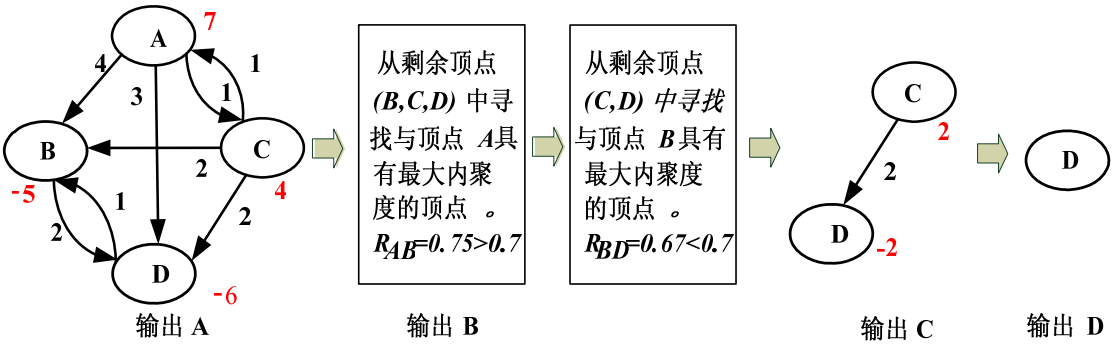

图 4.6 基于改进 MO 算法的局部主题排序过程

4.6 实验与评价

通过搜索引擎获得 20 个不同主题的语料集合，经整理后每个主题语料来自同一事件的报道 5~8 篇文档，在对每个主题进行局部主题划分和文摘句抽取的基础上，分别使用 MO、CO 和改进 MO 算法对文摘句进行了排序，分别生成 20 个文摘。

4.6.1 人工评价

为了便于对每个文摘的排序效果进行客观评价，我们将每个文摘的排序效果分为三类：差、一般和好。如果文摘的可读性较差，但通过调整文摘句的顺序能极大地提高可读性，则认为该文摘的排序效果较差；如果能够基本理解文摘的大意，但通过调整文摘句间的顺序能进一步提高文摘的可读性，则认为该文摘的排序效果一般；如果一个文摘的可读性较好，仅仅通过调整文摘句的顺序不能进一步提高文摘的可读性，则认为该文摘的排序效果较好。

为了使文摘中的代词不影响对排序效果的评价，我们首先对文摘中出现的代词进行了手工指代消解，然后请五位专家在不浏览文档内容的情况下，按照上述标准分别对 60 个文摘进行了评价，通过对评价数据的综合分析发现，专家们对每个文摘的评价基本一致，评价结果如表 4.4 所示。

表 4.4 三种排序算法的评价结果

算法	差	一般	好
MO	4	12	4
CO	5	7	8
改进 MO 算法	2	8	10

表 4.4 表明, 改进 MO 算法的排序结果被评价为好的要多于其它两种算法，且被评价为差的也少于其它两种算法，总体上来看，由改进 MO 算法排序得到的摘要效果要明显优于 MO 和 CO 算法。根据局部主题间的内聚程度，将一些内聚紧密的局部主题置于摘要中相邻的位置，从而有效避免了 MO 算法排序生成的文摘中可能将某些内聚程度较大的局部主题分散到整个摘要中的不足，大大提高了文摘的连贯性和可读性。但是通过对两个评价为差的文摘及其相关原文进行分析，发现当各局部主题包含的句子在文档中比较分散或者在位置上内聚紧密但逻辑意义并不连贯时，生成的文摘仍然存在逻辑性不强和可读性较差的问题，所以改进 MO 算法对文档集中各文档的结构还具有一定的依赖性。

4.6.2 ROUGE 自动评测

由于人工评价的工作量非常大，并且对文摘测试的结果具有一定的主观性，受 BLEU[85]方法在机器翻译评价中获得巨大成功的启发，Lin 和 Hovy[89]提出了与其类似自动文摘评测系统 ROUGE（Recall-Oriented Understudy for Gisting Evaluation），该系统通过统计 *n-gram* 的共现对单文档文摘和多文档文摘进行评价，并用于目前多文档文摘领域最有影响的评测会议 DUC（Document Understanding Conference）的评价。

为考察本章所给改进 MO 排序算法对提高文摘质量方面是否有帮助，我们请多个专家共同对上述 20 个主题中的随机 3 个主题进行了文摘（即理想摘要），并实验对比了 MO、CO 和改进 MO 算法的 ROUGE 分值的差别。表 4.5 给出了评测结果。

表 4.5 三种排序算法的 ROUGE 得分比较

主题	排序算法	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4	ROUGE-L	ROUGE-W
	MO	0.37321	0.04646	0.03752	0.00231	0.18822	0.12236
1	CO	0.37321	0.04656	0.03755	0.00272	0.10647	0.12348
	改进 MO 算法	0.37321	0.04773	0.04063	0.01069	0.20288	0.13563
	MO	0.35317	0.06479	0.01527	0.01029	0.31179	0.10189
2	CO	0.35317	0.06315	0.02022	0.01054	0.31326	0.06537
	改进 MO 算法	0.35317	0.07063	0.02815	0.01965	0.41630	0.12065
	MO	0.26238	0.03859	0.00993	0.00321	0.22335	0.09325
3	CO	0.26238	0.03861	0.00799	0.00489	0.23748	0.09499
	改进 MO 算法	0.26238	0.04136	0.01204	0.01017	0.26303	0.12256

从表 4.5 可以看出，ROUGE-1 的分值在三种算法排序前后几乎没有发生变化，ROUGE-2, 3,4 的分值在用改进 MO 算法排序后变化也很小，但对 ROUGE-L 的影响

却比较大。原因在于，ROUGE 是一种基于 *n-gram* 召回率的评测方法，当 *n* 为 1 时，只对比机器文摘与理想文摘之间词的召回率，因此句子排序与否，对评测结果并不影响。而 ROUGE-L 是考察机器文摘与理想文摘间最长的匹配情况，因此，在利用改进 MO 算法排序后会改善这种最长匹配，特别是这种匹配跨越两个文摘句时。尽管整体上 ROUGE-L 会在排序后得到提出，但这种提高对于“好”文摘并不明显。

为了考察改进 MO 算法对不同长度 n-gram 评测结果的影响，我们分别对三个主题的六个 ROUGE 得分进行了分析比较(即比较改进 MO 算法的 ROUGE 得分与 MO 和 CO 算法 ROUGE 平均得分的增量)，从图 4.7 可以看出：(1)ROUGE 评测中用于比较的内容单元（*n-gram*)越长，重新排序对其评测结果的影响就越大；(2)对“好”文摘进行重新排序时，文摘质量并不能得到明显改善；(3)文摘句排序对文摘质量有很大的影响，它是导致文摘“差”的重要因素之一，因为对“差”文摘重排序后，其质量有较大的改善。

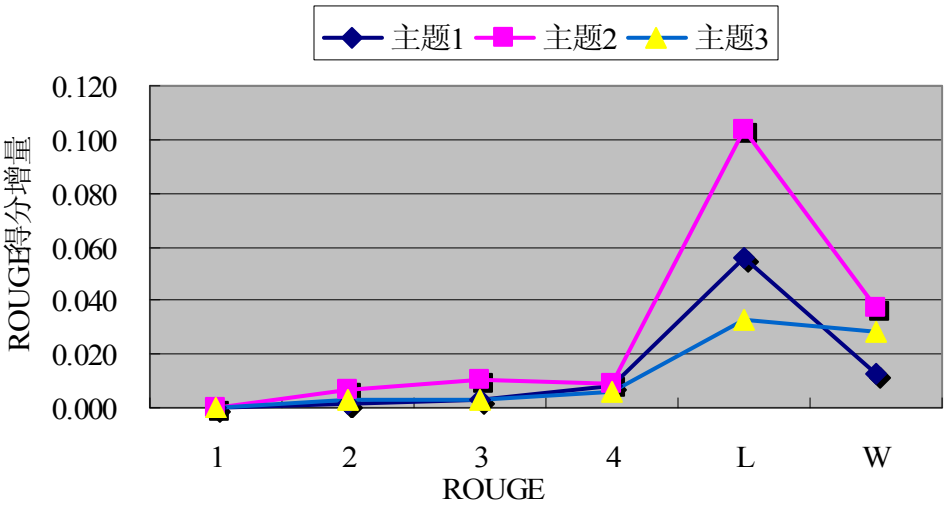

图 4.7 改进 MO 算法对不同 ROUGE 得分的影响

总体而言，自动评价与人工评价在测试结果上基本保持一致，即改进 MO 算法比 MO 和 CO 算法排序生成的文摘更加连贯和更具可读性。由于 ROUGE-L 与 ROUGE-N 的主要区别在于：ROUGE-L 考虑了单词的顺序，而在图 4.7 中，改进 MO 算法比 MO 和 CO 算法的 ROUGE 得分有较大幅度的增加，这说明了用改进 MO 算法进行排序的文摘与理想文摘具有更高的一致性，从而使得文摘在语义上更加连贯和易于理解。

4.6.3 流利度自动评测

由于多文档文摘中组成文摘的句子大多是来自不同的文章，这些从不同的原文章中抽取出来的句子组合在一起，使得文摘的连贯性和易读性成为一个很重要的问题。为了避免人工评价的过多地主观性和减少工作量，秦兵教授[111]提出了流利度自动评测的方法，并给出了两个计算文摘流利度值的方法：顺序比较和 *n-gram* 模型。基于顺序比较计算流利度的方法就是通过比较自动文摘与专家文摘相匹配的句子在顺序上是否一致，统计出顺序一致的句子的个数，从而得到文摘的流利度的值。计算公式如下：

$$Fluent_1 = \frac{the\ number\ of\ same\ ordering\ sentences\ in\ A\ and\ B}{|B|} \tag{4.1}$$

其中：*A* 为系统文摘，*B* 为专家文摘，|*B*|为专家文摘中包含的句子数，|*A*|=|*B*|。

Unigram 是指系统文摘中是否出现和人工文摘一样的句子，不论顺序是否一样，由于仅仅是对抽取后句子进行排序评价，三种排序算法采用的文摘局和理想文摘出现的句子是一样的，并且 *Unigram* 不会对排序有所贡献，故在评价中不考虑此因素。*Bigram* 是指是否有两个邻近的句子的顺序与人工文摘中的句子相同，*Trigram* 是指是否有 3 个邻近的句子顺序与人工文摘中的句子相同。更多的 *n-gram* 出现的概率很小，在此暂不予以考虑。具体的公式如下：

$$Fluent_2 = \frac{\alpha \times NumBigram + \beta \times NumTrigram}{|B|} \tag{4.2}$$

其中，α=0.4，β=0.6，*NumBigram* 表示系统文摘 *A* 和专家文摘 *B* 中相同的 *Bigram* 数，*NumTrigram* 表示系统文摘 *A* 和专家文摘 *B* 中相同的 *Trigram* 数，|*B*|为专家文摘中包含的句子数。

由于基于顺序比较的评价方法侧重于对整体排序的评价，基于 *n-gram* 的方法侧重于对局部排序的评价，所以理论上选择上述两种方法相结合的策略，可以提供更准确的评价方法，具体公式如下：

$$Fluent_3 = \phi \times Fluent_1 + \psi \times Fluent_2 \tag{4.3}$$

其中，Φ 和 Ψ 分别为权重系数，满足 Φ+Ψ= 1。

秦兵教授对 $Fluent_3$ 和人工评价相关性进行了实验，发现当 Φ=0.4，Ψ=0.6 时，$Fluent_3$ 可以获得 0.7551 的相关性系数，本文在此基础上，对 MO 算法、CO 算法和改进的 MO 算法生成文摘的流利度进行了评价，并与专家打分进行了比较，具体评价结果如

表 4.6 所示。其中专家评价分为五个等级进行打分，打分标准如下：

（1） 流利度十分好，可读性很强，句子连贯为 0.8~1 分；

（2） 文章较通顺，大多数句子语义连贯为 0.6~0.8 分；

（3） 有一些句子之间缺乏连贯性为 0.4~0.6 分；

（4） 文章较不通顺，由多句之间缺乏连贯性为 0.2~0.4 分；

（5） 文章可读性十分差，句子前后产生很大的歧义为 0~0.2 分。

表 4.6 三种排序算法的流利度得分比较

主题编号	排序算法	$Fluent_1$	$Fluent_2$	$Fluent_3$	专家打分
	MO	0.5000	0.2000	0.3200	0.4
1	CO	0.5833	0.2333	0.3733	0.5
	改进 MO 算法	0.6667	0.3167	0.4567	0.55
	MO	0.4286	0.1571	0.2657	0.4
2	CO	0.5714	0.2000	0.3486	0.45
	改进 MO 算法	0.6429	0.3429	0.4629	0.55
	MO	0.5625	0.2625	0.3825	0.5
3	CO	0.4375	0.1375	0.2575	0.3
	改进 MO 算法	0.8125	0.5000	0.6250	0.75

表 4.6 实验结果表明，$Fluent_3$ 流利度与专家打分基本上保持一致，改进的 MO 算法在 $Fluent_3$ 流利度和专家打分上都要明显好于 MO 和 CO 算法，在主题 1 和主题 2 的评价中，MO 算法要略差于 CO 算法，而主题 3 中 MO 算法要明显好于 CO 算法，分析其原因，我们发现在主题 1 和 2 的文档中包含大量的时间信息，这非常利于 CO 算法的排序，而主题 3 的文档中包含的时间信息较少或者是难于识别和排序，所以 CO 算法对文档集中包含的时间信息具有一定的依赖性，比较适用于包含时间信息较多的新闻事件；整体来说，MO 算法的排序效果一般，但是相对比较稳定，其对文档集中包含局部主题间的顺序的稳定性具有一定的依赖性，如果稳定性好，则排序效果较好，否则排序效果较差。改进的 MO 算法将局部主题间的内聚度融入到 MO 排序过程中，从而弥补了仅用 MO 方法对文摘句排序的效果急剧下降的缺点。

4.7 本章小结

对于多文档文摘，两个来自不同文档的文摘句间并没有明确的位置关系，文档中也可能没有明确的发布时间，因此需要考虑更多的因素来确定各文摘句的先后顺序。本章在分析了 MO 和 CO 算法的基本原理和不足之处的基础上，提出了一种将局部主

题间的内聚度与 MO 算法相结合进行文摘句排序的新方法，即在统计局部主题间相对位置的基础上，建立它们之间的关系有向图并计算其内聚度；排序过程中每从有向图中输出一个顶点时，从剩余顶点中查找与其具有最大内聚度的顶点，若该内聚度大于阈值，则将这两个顶点所代表的局部主题文摘句置于摘要中相邻的位置，并从人工评价、ROUGE 自动评价和流利度评价三个方面对 MO 算法、CO 算法和改进的 MO 算法生成的文摘进行了评测，测试结果发现改进的 MO 算法较其它两种算法要好，而且具有很好的鲁棒性。但是该改进算法的不足之处也比较明显，当各局部主题包含的文摘句在文档集中比较离散或者位置内聚紧密的文摘句在语义意义上缺乏连贯性时,生成的文摘仍然存在逻辑性不强且不易理解的问题，对文档集中各文档的结构还具有一定的依赖性。

第5章 基于多特征融合和查询短语识别的查询文摘算法研究

5.1 引言

随着 Internet 的快速发展，网络已成为一个巨大的信息源，在线可获得的信息量以指数级迅猛增长，导致了人们所说的信息爆炸的发生。一方面我们生活在信息的海洋中，处处充斥着各种电子信息；而另一方面却很难从信息海洋中找到我们所需的正确信息，无法最大限度的从这些信息中获益。如何在海量信息中搜寻所需要的知识，获取信息的主旨，如何快速的阅读每天涌现出来的各种新信息，是一个目前被广泛讨论和研究的热点课题。

自然语言处理技术(NLP，Natural Language Processing)最大的应用成功案例就是搜索引擎的引入和开发。通过搜索引擎，可以检索到特定范围信息池(万维网，数据库)内的信息，有助于解决在信息世界中快速获得信息的问题，正是在这样的背景下，造就了 Google，Yahoo 和百度等搜索引擎服务提供商的成功。然而，目前的搜索引擎还是有很多的缺陷，自动检索到的结果是一个数量庞大的网页集合，其中包含有用信息和无用信息，而选取有用信息的过程则完全取决于用户个人，从而造成了用户在获取信息过程中的巨大负担。显然，通过阅读网页文摘而不是全文人们能够极大地提高获取信息的速度，更容易地获取有用的信息，但是单单靠手工来完成对文本信息的摘要编撰工作除了无法应付互联网上不断涌现的海量信息外，还存在着以下缺点：首先，每个人对网页文本内容的理解必然受到其自身知识背景的影响，这使得人工撰写的摘要存在很大的主观性；其次，人工摘要也很难满足面向特定任务或者基于用户请求的文摘任务，比如在信息检索过程中不同的用户会有不同的关键字或者自然语言指出他所特别关心的内容，这就需要在撰写文摘时必须根据用户请求，对包含特定信息的语句有所侧重，用户请求是很难预知的，由人来编写相关的摘要将无法满足用户对信息获取的实时性要求。

如果能够通过网页自动文摘技术将网页文档压缩成一两句主旨，将会大大提高用户在判断信息是否为有用信息的速度。基于用户查询的自动文摘正是提供给用户在检索得到的大量候选项选择过程中有力的帮助工具之一，世界上几乎所有的搜索引擎都能够使用网页自动摘要的技术对其搜索结果提供摘要，以此提高网页相关性判断的速度。目前搜索引擎的自动摘要有以下几种表达形式：

（1） 提供命中网站或者网页的前几行信息，典型的代表为 http://www.yahoo.cn。目前很多网站都在效仿这种做法。该方法处理简单，但反映信息不全面，无法概括网页主题信息，有用信息较少。

（2） 截取关键词(即检索词)周围的文字或者句子，这是目前大多数搜索引擎采用的方法，典型代表有 http://www.google.cn 和 http://www.baidu.com 等网站。该方法方便动态处理，容易实现，但是内容杂乱，不易理解主题内容。

（3） 人工方法为所搜集到的网页编写摘要。典型代表为 http://www.cei.gov.cn。该方法摘要质量高，信息量大，方便阅读，但是要耗费大量的人力，难以适应快速增长的网络信息资源的需求。

5.2 相关研究

当搜索引擎返回信息时，用网页文档的通用摘要代替仅含有查询关键词的句子构成的摘要可以用户提高网页相关性判断的速度准确性，为了进一步帮助用户快速准确定位信息，近年来国外一些研究人员提出了一些面向查询的自动文摘代替通用文摘返回给用户的方法，实验表明这样更有利用户进行相关性判断。

所谓面向查询的自动文摘就是基于特定的查询语句或查询关键词，将查询结果中每个文档的相关内容分别浓缩为一个覆盖主要相关主题、简洁、组织良好的摘要[126]。它除了能为用户提供所需信息的一个较为全面摘要外，还能帮助用户判断和浏览感兴趣的具体信息内容，该工作更能适应当前 Internet 环境下对于信息获取的个性化需求。

理想的面向查询的自动文摘应该既是相关文档的“压缩版”，同时又能满足用户对信息的个性化需求。因此采用何种策略从文档集合中选取文摘句，成为该技术的关键所在，其很大程度上决定了文摘质量的好坏。目前，国际国内研究人员就文摘句的选择问题陆续开展了许多探索性的研究工作。

Wauter Bosma[127]根据相关的修辞结构理论，构造文档与查询条件所包含的各个句子间的语义距离图，并利用语义距离图判定与用户查询需求最相关的句子，实现文摘句的挑选。White[128]和 Goldstein[129]构建了一种依赖查询的句子抽取模型。通过把相关文档处理成抽取单元的集合后，结合抽取单元与查询在词语重现上的特征，抽取单元自身在原文档中的位置以及包含的词语特征等给句子打分，根据得分高低抽取相关的文摘单元。大连理工大学的林鸿飞[130]依照覆盖区域中句子的关键字数目和句子内部的密度均值抽取文摘句。北京大学的李素建[131]通过抽取候选句与查询的多种关联

信息筛选文摘句。

针对面向查询的自动文摘，考虑到网页文档的半结构化特征以及主题与查询的关联，本章提出了一种多特征融合的句子重要度计算的策略，即对查询输入进行短语识别和网页结构分析的基础上，将关键词短语、网页结构等启发式规则信息融入到句子重要度计算，并分析句子与查询输入的关联特征，使文摘内容能与用户需求一致，最后利用句子相似度计算的方法减少文摘句的冗余度，在冗余度最小的前提下，得到最终的文摘句集合，从而实现文摘信息的内聚性和全面性，便于用户进行网页相关性判断。

为了客观评价本章提出的面向查询的自动文摘方法的性能，我们与目前常用的一些基于统计与语义相结合的文摘方法进行了比较。本节将介绍三种基于抽取的文摘方法：基于查询的段落抽取[132-133]，基于查询和特征词频统计的句子抽取[134-135]，基于查询和词汇链统计的句子抽取[136-137]。

5.2.1 基于查询的段落抽取

首先，通过网页清洗将网页文件转换成纯文本文件(即将网页中的标签、图片以及导航等信息过滤掉)；然后，利用段落标签(<P>)和标题标签(<Title>，<Hn>)等信息进行段落识别，其中网页标题和子标题都按段处理；最后，计算每个段落与查询语句的相似度，具有最大相似度的段落被抽取出来作为文摘，如果该段落长度大于长度阈值时，则从该段落抽取相应长度的内容，如果小于时，则从相似度排列次之的段落中抽取。查询语句 $Q(t_1,t_2,\cdots,t_n)$ 与段落 $P_j(t_{j1},t_{j2},\cdots,t_{jn})$ 的相似度计算如公式(5.1)所示：

$$sim\left(Q,P_j\right)=\sum_{i=1}^{n}\left(tf_{q_i}\times log\left(N/df_i\right)\right)^2\times\left(tf_{ji}\times log\left(N/df_i\right)\right) \tag{5.1}$$

其中，t_i 是构成向量空间的一个特征项；tf_{qi} 为 t_i 在查询 Q 中的出现频率；N 为搜索引擎返回的网页文档集清洗后的段落数；df_i 为 t_i 在文档集出现的段落数；tf_{ji} 为 t_i 在段落 P_j 中的特征频率。

5.2.2 基于查询和特征词频统计的句子抽取

首先，依据公式(5.2)计算网页文档集中特征 t_i 基于查询关键词的权值 W_i；然后，计算网页中每个句子的权重(组成该句子的所有特征的权重之和)；最后，按照权重对所有句子进行降序排序并依次抽取系统指定长度的句子作为文摘。

$$W_i = \lambda \times tf_i \times log\left(N / df_i\right) \tag{5.2}$$

其中，λ为调节参数，当 t_i 不是查询关键词时，λ=1；当 t_i 是查询关键词时，λ取一个大于 1 的经验值；tf_i 为 t_i 在文档集中的出现频率；N 为搜索引擎返回网页文档集中文档个数；df_i 为 t_i 在文档集中的文档频率。

5.2.3 基于查询和词汇链统计的句子抽取

所谓词汇链是一组词义内聚相关的词语集合。借助《知网》计算特征之间的语义相似度[107]，当两个特征间的语义相似度大于指定阈值时，将这两个特征归于同一个词链中。首先，将查询和文档集中的每个特征分别归于相应的词汇链中；然后，依据公式(5.3)计算词汇链 i 的权值 W_i，同时计算每个句子的权重；最后，依据权重对所有句子进行降序排列并依次抽取系统指定长度的句子作为文摘。

$$W_i = \gamma \times |i| \times log\left(N / df_i\right) \tag{5.3}$$

其中，γ 为调节参数，当词汇链 i 中不包含查询关键词时，γ=1；当词汇链 i 包含查询关键词时，γ 取一个大于 1 的经验值；$|i|$为所在词汇链 i 中包含特征的个数；N 为搜索引擎返回网页文档集中文档个数；df_i 为词汇链 i 在文档集中的文档频率。

5.3 WEB 文档预处理

与传统的文本文档相比，WEB网页的一个显著特点是：它通过HTML标记(Hypertext Markup Language)提供了对自动文摘更为有利的一些辅助信息，包括：文档标题(<TITLE>)，各级子标题(<H1>，<H2>，…，<Hn>)，段落标记(<P>，</P>)等等。但是WEB文档中也存在着一些对文档摘要不利的因素，即WEB文档不像传统的文本那样整齐、干净，其中包含了大量噪声，例如：为了增强用户交互性而加入的SCRIPT代码段；为了便于用户浏览或出于商业因素所加入的广告链接、导航链接、标注版权等信息。因此，对于WEB网页文档的预处理主要包括以下两个步骤：

(1) 网页去噪，即过滤掉网页文档中的无关信息，保留主题内容及HTML标记；

(2) 从步骤(1)处理完的文件中提取正文信息。

5.3.1 网页去噪

WEB网页通常包含两部分内容：(1)体现网页主题信息的内容(“主题”内容)；

(2)与主题无关的内容(“噪音”内容)。由于网页中的“噪音”给自动文摘带来了很大的干扰，所以如何快速准确地识别并清除网页内的“噪音”是提高WEB文档自动文摘的重要环节之一，因为去噪后的网页没有“噪音”内容的干扰，简化了网页标签结构的复杂性，提高了系统处理结果的准确性。

在网页去噪的研究方面，Buyukkokten[138]率先提出了将网页分割为平行的STU(Semantic Textual Unit)模型，每个STU对应网页中的块，但是这种方法改变了网页的内容和结构，而且保留了无关的内容和链接。Gupta[139]提出了直接从网页中删除无关部分的方法，维持了源网页的结构和内容，但在删除链接时没有考虑上下文的语义，容易导致提取的结果不完整。

在国内，张志刚[140]提出以一组启发式规则为基础，利用信息检索的方法以及WEB网页的特征提取网页的主题内容。常育红[141]用标记树表示页面文档，采用自底向上的算法，抽取出具有不同语义的页面内容。荆涛[142]利用网页中的布局信息对网页进行划分，并在此基础上去除“噪音”信息。孙承杰[143]提出了一种依靠统计信息从中文网页中抽取正文内容的方法，先根据网页中的HTML标记把网页表示成一棵树，然后利用树中每个节点包含的中文字符数，从树中选择包含正文信息的节点。王琦[144]基于DOM(Document Object Model)规范，针对HTML缺乏语义描述的不足，提出了将语义信息融入STU模型的STU-DOM树模型。

实验结果表明，上述一些算法把正文内容误判为“噪音”信息而清除掉的情况是无法杜绝的。但是对于自动文摘系统而言，网页去噪的目标强调的是内容的重要性而非完整性，也就是说，由于文摘本身是来自于网页的重点段落，因此在实际的“噪音”内容过滤过程中，把一些虽然是正文内容但重要度极低的部分过滤掉，并不会对最终的文摘造成很大影响。根据这一思想，以STU-DOM树模型为基础，本章采用文献[88]提出的基于信息块重要度计算的网页去噪算法。

5.3.1.1 网页布局与结构模型

网页常见的布局形式及其有关的描述定义[144]为：网页通常可以划分为多个不同区域，每个区域称为块。每个块还可以继续嵌套子块。图5.1是一种很常见的网页布局形式。

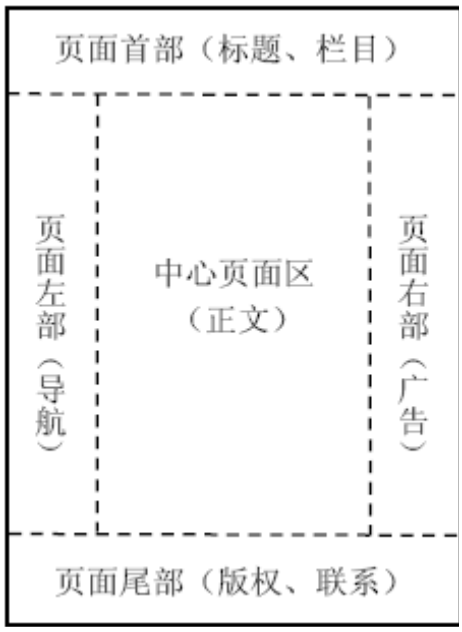

图 5.1 常见页面布局图

STU中的每个语义文本单元对应于网页布局图中的一个区域，这些STU相互嵌套便构成了STU树。每个STU树由具有强大的语义描述能力的DOM树节点组成[145]，即STU-DOM树。DOM（文档对象模型）是万维网联盟W3C（World Wide Web Consortium）制定的标准接口规范。图5.2是一个WEB网页源文件及其对应的DOM树模型例子。

```
<html><head><title>welcome</title></head>
<body>
  <table width="100%">
      <tr>   <td>  </td>   </tr>
  </table>
  <div>
  <table width="100%">
      <tr>
      <td> 
            <table width="100%">
                <tr><td>???? </td></tr>
            </table>
      </td>
      <td>
        <table width="100%">
          <tr><td><a herf="">1</td></tr>
          <tr><td><a herf="">2</td></tr>
          <tr><td><a herf="">3</td></tr>
        </table>
      </td>
      </tr>
  </table></div></body></html>
```

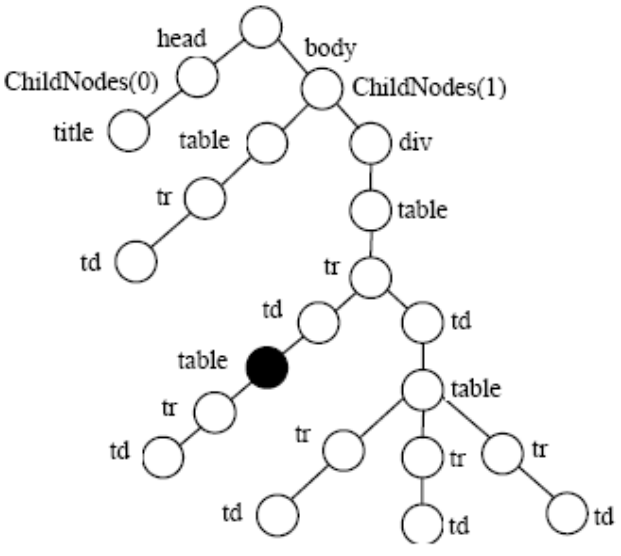

图 5.2 网页源文件及其对应的 DOM 树

利用HTML与DOM树的结构一致的关系，可以将STU与DOM树结合，向DOM树中的某些节点添加描述语义的属性，生成的DOM树就是STU-DOM树，STU-DOM树

中具有语义属性的节点称为STU节点。因为STU-DOM树同时具有DOM树和STU树的结构和语义，所以节省了存储空间、简化了处理流程，而且保证了提取正文后的页面保持与原始页面一致的结构。

5.3.1.2 网页去噪系统与算法

网页去噪系统[88]包括五个部分：页面解析器、DOM过滤器、分块器、语义解析器和剪枝器。页面解析器负责将WEB文档转化为DOM树；DOM过滤器负责将无关节点从DOM树中删除；分块器是指向STU节点添加语义属性，将DOM树转化为STU-DOM树；语义解析器负责语义属性值的计算；剪枝器是指从STU-DOM树中删除无关链接和没有内容的块，最后输出只含有主题信息的WEB文档。

5.3.1.2.1 过滤和分块算法

过滤和分块是将DOM树转换为STU-DOM树的过程。过滤器从DOM树的根节点开始递归遍历DOM树，删除所有无关节点，遇到分块节点时调用分块器，向该节点添加语义属性，使其成为STU节点，当STU节点的语义属性值满足剪枝条件时，调用剪枝器处理该节点。

王琦[144]通过对WEB页面的抽样结果分析，发现与主题无关的块总是包含大量无关链接，提出了利用这一特征计算STU-DOM节点的主题相关度的算法，但是该算法用于自动文摘的网页过滤方法会带来如下问题：(1)如果一个包含与网页主题有关的链接块内存在大量的超链接的说明文字，这样会导致计算出来的节点相关度很容易低于预先设定的阈值而被剪枝，但这些超链接内容对基于上下文分析的WEB文档自动文摘却有着重要意义；(2)不包含超链接文字的节点也有可能并不包含与文章主题相关的内容。

为了弥补上述算法的不足，我们采用的语义信息是块中包含标题词的比例。所谓标题词是指出现在网页标题、文章标题以及各级子标题中的词项，在STU-DOM模型中对应子树中的标题词总数。根据对文档摘要的分析发现：标题中出现的词是非常重要的，常常出现在最终生成的文摘之中。基于上述思想，改进的信息块重要度计算过程包括以下两个步骤：

(1) 获取网页标题及其各级标题的标题内容，对它们进行分词、停用词过滤等操作，得到只包含实词的标题词序列 $S_{title}=\{w_{t_1},w_{t_2},\cdots,w_{t_m}\}$ 以及各级标题词序列

$S_{head}=\{w_{h_1},w_{h_2},\cdots,w_{h_n}\}(i=1,2,\cdots,6)$;

(2) 遍历STU-DOM的每个节点tw，对每个节点$tw_i(i=1,2,\cdots,l)$内的字符串分词，得到字符串序列。根据公式(5.4)计算字符串词序列s_{tw_i}中出现的标题词的比例。

$$P_t=|\sum_{i=1}^{l}s_{tw_i}|/\left(|\sum_{i=1}^{m}w_{t_i}|+|\sum_{i=1}^{n}w_{h_i}|\right) \tag{5.4}$$

5.3.1.2.2 剪枝算法

设信息块重要度阈值为δ_{BI}，如果$P_t>=\delta_{BI}$，则表明该信息块是重要的；如果$P_t<\delta_{BI}$，则表明该信息块是与主题无关的。

如果剪枝器判断STU节点与主题无关，那么将其所有子树中的链接删除。如果一个STU为空则删除该节点。这样通过剪枝算法可以将无关链接和没有内容的块删除。

经过上述过滤、分块与剪枝后的STU-DOM树仅仅包含与文章主题相关的重要内容，它所对应的WEB文件中不再包含无关的、不重要的文本及其超文本标记。

5.3.2 正文提取算法

正文提取是网页去噪之后，进一步提取出文本内容的操作，即根据HTML语言的特征，分析判断正文内容所在的位置，并过滤掉HTML 标记，将正文内容提取出来转化成纯文本文件[146-148]。正文提取的关键在于正文内容的准确定位。

本章讨论的网页正文提取是针对于有主题网页的(所谓有主题网页是指通过成段的文字描述一个或者多个主题的WEB文档)。主题型的网页可以将正文内容作为一个整体放置于一个或多个表格中。正文提取算法的关键之处在于如何准确定位包含正文的表格，而大部分的中文新闻网页的正文存放在多个表格中。文献[143]提出的算法仅仅适用于正文内容全部放在同一个表格中的情况。为了可以处理正文内容放在多个表格中的情况，需要进一步改进孙承杰的算法。

由于有主题网页中的正文通常是用成段的文字来描述的，中间通常不会加入大量的超链接，因此，在包含正文的表格中，超链接所占的比例一般很小，而包含非正文信息的表格中超链接所占的比例一般很大，根据这一点可以作为判断一个表格内包含的内容是否为正文信息。利用上述思路进行正文提取的算法如下：

(1) 利用前述的算法将WEB文档表示成一棵STU-DOM树；

(2) 找到STU-DOM树的所有表格节点，对每一个表格节点，删除<FORM></FORM >，

<STYLE></ STYLE >，<SCRIPT></SCRIPT>等标记内嵌的全部内容，得到表格内所有文字节点的内容；

(3) 利用公式(5.5)计算：

$$P = Length(T_a) / Length(T_t) \tag{5.5}$$

其中，T_a指表格节点下所有的超链文本内容，T_t是TABLE节点下所有的文本内容，$Length(T_a)$是T_a中所有字符串的长度，$Length(T_t)$是T_t中所有字符串的长度。

可以根据P值判定正文表格：如果P小于某一阈值t_P，则认为该表格节点包含正文内容，将其作为候选节点；如果P大于阈值t_P，则认为该表格节点不包含正文内容。

如果阈值t_P过大，超链接内容不能完全清除；如果阈值t_P过小，又可能过滤掉正文内容。通过对大量WEB网页文档的对比实验分析，将阈值t_P定为0.36可以取得较好的正文提取效果。

5.3.3 未登录词识别算法

在大规模真实文本处理中，有一些仅仅依靠分词词典是无法识别出来的词汇被称为未登录词，以中文人名、地名、机构名为 3 种主要形式。未登录词的识别，是中文文本自动分词中遇到的除歧义识别外的另一难题。未登录词的识别不但可以提高分词系统分词的正确率，而且对于改善分词词典也有很重要的作用。依靠未登录词识别功能识别出的新词，可以补充进词典，作为下一次分词的依据，以此下去，词典不断被丰富完善，分词的正确率也会得到提高，可以说这是一个互助的过程。

本章采用最小切分和逆向最大匹配法结合的复合分词算法对提取后的网页文本进行分词，但是由于分词词典的容量有限等问题，实验中，我们发现分词后的文本中有很多单字构成的散串，并且在同一文档或文档集中出现频率比较高，通过人工分析发现，这些散串往往是一些未登录词或是一些专业术语。所谓散串是指文本经过分词后，在文本中出现连续的若干个单字单独构成词汇，而这种情况在中文领域是不多见的，当然，我们约定这样散串中一般不包含“的”、“有”、“是”等单字高频虚词。为了提高文摘句重要度计算和文摘句抽取的效果，有必要对这些散串进行未登录词的识别[149]。具体识别算法为：

(1) 将每个句子中的散串分解为两字或两字以上的组合。例如：散串“北 理 工”被分解为三个子串：“北理”，“理工”和“北理工”；

(2) 统计每个子串的频度。例如：文档中出现“北理工”2 次、“理工”4 次、“北

理”3 次；

(3) 对每个子串 S 进行权重 W_S 的计算，加权方法如公式(5.6)所示：

$$W_S = Length(S) \times Frequency(S) \tag{5.6}$$

其中，W_S 是子串 S 的权重，*Length*(*S*)是子串 S 的长度，*Frequency*(*S*)是子串 S 出现的频率；

(4) 权重 W_S 高于预定阈值 δ 的子串将被认定为未登录词(实验中采用阈值 δ=5.0)；

(5) 根据子串 S 的长度 *Length*(*S*)和权重 W_S 对所有满足条件的子串进行降序排列；

(6) 依次对原分词结果进行替换，例如：用“北理工”替换“北 理 工”。

通过上述识别算法能够提高未登录词的识别率，提高分词的准确率，更利于统计词频以及文档句子重要度的计算。实验证明这种通过散字识别进行未登录词识别的方法是有效的。

5.4 查询短语识别

搜索引擎开始工作时，首先对用户输入的查询语句进行分词和停用词过滤操作，并将处理后剩下的词语作为关键词进行搜索，但是经常存在如下问题：假如用户输入的查询语句是“操作系统”，经分词后的“操作/v 系统/n”两个词被单独用于特征权值和句子权重的计算，从而导致“……操作系统……”和“……操作……系统……”两个句子在权重计算过程中没有任何区别，但前句显然与用户查询短语所需的信息更为相关，故将查询中的短语分成多个关键词并单独用于句子权重计算是不合理的。因此在搜索引擎返回网页信息列表的基础上，为了更加合理有效地对句子进行权值计算，有必要对查询语句中的短语进行识别。

5.4.1 互信息

互信息(MI，Mutual Information)是信息论中的一个重要概念，它是一种检测事件集之间相关程度的方法。在自然语言处理领域中，互信息一般反映的是字与字之间的静态结合，因为它计算的就是相邻字出现的频率，根据这个频率与字单独出现频率进行比较，计算出互信息来判断是否可以组成词语。互信息的概念最早见于信息论，其中互信息被作为一种衡量两个信号之间相互依赖的尺度[150]。在信息论中，这种二元互信息可以表示为两个信号发生概率的函数。具体在 NLP 领域中，就是把句子中词或词序列作为一系列可能有关联的随机事件，然后用互信息对它们进行分析与研究。

对有序汉字串 *xy*，汉字 *x* 和 *y* 之间的互信息定义如公式(5.6)所示：

$$I(x,y)=log_2\frac{P(x,y)}{P(x)\times P(y)} \tag{5.6}$$

其中 *P*(x, y)是 x 与 y 的邻接同现概率，*P*(*x*)和 *P*(*y*)分别代表 *x* 和 *y* 的独立概率。

如果在字容量为 *N* 的汉语语料库中，*x* 与 *y* 邻接同现的次数为 *r*(*x*, *y*)，*x* 与 *y* 间的互信息反映了汉字之间结合关系的紧密程序：

$$P(x,y)=\frac{r(x,y)}{N},P(x)=\frac{r(x)}{N},P(y)=\frac{r(y)}{N} \tag{5.7}$$

互信息反映了汉字串 x 与 y 间相关的程度：

（1） *I*(*x*, *y*)≥0，则 *P*(*x*, *y*)≥*P*(*x*)×*P*(*y*)，则 *x* 与 *y* 之间是相关的，随着 *I*(*x*, *y*)增加，相关度增加，如果 *I*(*x*, *y*)大于给定的一个阈值，可以认为 *xy* 是一个词；

（2） *I*(*x*, *y*)≈0，则 *P*(*x*, *y*)≈*P*(*x*)×*P*(*y*)，此时 *x* 与 *y* 之间是不相关的；

（3） *I*(*x*, *y*)≤0，则 *P*(*x*, *y*)≤*P*(*x*)×*P*(*y*)，此时 *x* 与 *y* 之间是互斥的，*xy* 基本上不会结合成词。

5.4.2 查询短语识别算法

以用户输入查询语句搜索引擎反馈的网页集作为汉语语料库，计算每个词之间的互信息，判断查询语句中是否含有短语，具体短语识别的具体步骤如下：

(1) 对用户输入的查询语句进行停用词过滤和分词操作，不妨假设分词后的关键词集合为 $W(K_1,K_2,\cdots,K_m)$；

(2) I=0；

(3) *I*=*I*+1，若 *I* 大于阈值 γ，则转至(7)；否则，根据公式(5.8)计算两两关键词的互信息 $I(K_i,K_j)$；

(4) 如果 $I(K_i,K_j)$越大，则表示关键词 K_i 和 K_j 搭配成短语 K_iK_j 的关系越紧密，故当 $I(K_i,K_j)$大于指定阈值 ζ 时，则认为 K_iK_j 是短语；

(5) 将 K_iK_j 加入短语候选集合 W^*中；

(6) 重新定义查询关键词新组合 $W=W\cup W^*$，即 $W(K_1,K_2,\cdots,K_i,K_iK_j,K_j,\cdots,K_m)$，清空 W^*，转至(3)；

(7) 将确定的关键词短语在已分词的文档集中进行短语标注(例如：如果 *I*(*A*,*B*)大于指定阈值时，将文档中“……*A*　*B*……”的句子转化为“……*AB*……”)。

$$I(A,B)=log_2\frac{P(A,B)}{P(A)\times P(B)}=log_2\frac{C_{AB}/N}{(C_A/N)\times(C_B/N)} \tag{5.8}$$

其中，A 和 B 分别为两个不同的关键词；$P(A,B)$为 A 和 B 先后相邻出现在搜索返回文档集中的出现概率；$P(A)$和 $P(B)$为 A 和 B 分别在搜索返回文档集中的出现概率；C_{AB}、C_A 和 C_B 分别为特征 AB，A 和 B 在文档集中出现次数；N 为文档集中的特征总数；γ 通常取 4 或者 5，ζ 取 1.75。

5.5 文摘生成

5.5.1 网页结构处理

网页文档是一种用 HTML 编写的半结构化文本，HTML 中的标签代表了网页制作者对该网页内容的理解和认识，比如：<B>和<Strong>是对内容的强调，表示该内容重要；<Title>是标识文档的标题，是整个文档内容的概括；<H1>~<H6>是用来标识小标题，是对下文内容的总结性概括等等。因此，在网页去噪的同时，需要挖掘网页中结构信息，并将网页结构信息融入特征 t_i 的权值 $W_i^{'}$ 计算中去，如公式(5.9)所示：

$$W_i^{'}=tf_i\times log_2(N/df_i)\times log_2(2+Tag\_Weight) \tag{5.9}$$

其中，tf_i 为 t_i 在网页文档集中的特征频率；N 为搜索引擎返回网页文档集的文档个数；df_i 为 t_i 在文档集中的文档频率；Tag_Weight 为特征所在标签的权值，我们参考了北京大学网络实验室的天网搜索引擎中考虑的标签及其权值，如表 5.1 所示：

表 5.1 HTML 标签及其对应的权值

标签	权值	标签	权值	标签	权值
<B>	4	<Title>	40	<Font Size=7>	16
<I>	4	<H1>	12	< Font Size=6>	12
<Blink>	4	<H2>	8	< Font Size=5>	8
<U>	4	< H3>	4	< Font Size=4>	4
<Em>	2	< H4>	1	< Font Size=3>	1
<OL>	4	< H5>	1	< Font Size=2>	1
<Li>	4	< H6>	1	< Font Size=1>	1
<A>	4	<Big>	4	<Caption>	6
<Cite>	8	<Strong>	4	<UL>	4

5.5.2 特征权值计算

在查询短语识别和网页结构分析完成之后，应该对二者中有利于文摘生成的信息进行整合。其中网页结构信息已经在特征 t_i 的权值 $W_i^{'}$ 的计算中得到体现；查询语句中每一个被识别的短语都被分解为三个关键词。假设短语识别前查询语句的分词结果是“*A*,*B*,*C*”，短语识别后查询语句由“*A*,*AB*,*B*,*C*”四个关键词组成，很显然它们间的重要性关系是 *AB*>*C*>*A*=*B*，故特征（包括关键词）t_i 的权值计算 W_i 如公式（5.10）所示：

$$W_i = \mu \times W_i^{'} = \mu \times tf_i \times log_2\left(N/df_i\right) \times log_2\left(2+Tag\_Weight\right) \tag{5.10}$$

其中，μ 为基于关键词的调节参数，当 t_i 不是查询关键词时，μ=1；当 t_i 为查询短语时，μ 取实验经验值 3.6；当 t_i 为查询短语中的词语时，μ=1.78；否则，μ=2.3。

5.5.3 基于启发式规则的句子权重计算

由于本章主要研究的是如何对搜索引擎反馈的网页进行摘要来提高用户相关性判断的速度和准确率，显然，基于理解的自动文摘方法不仅受领域限制，而且需要大量的知识库，文摘生成的速度很慢，不适应搜索引擎快速反馈的要求，所以我们采用传统的基于句子抽取的机械式自动文摘方法，即利用网页文档的一些启发式规则对句子进行权重计算和抽取，具体通过公式（5.11）计算文档中每个句子 S_i 的权重 $W^{'}(S_i)$。

$$W^{'}\left(S_i\right) = log_2\left(2+\alpha\right) \times \sum_{k=1}^{n}\left(C_k \times W_k\right) \tag{5.11}$$

其中，n 为特征向量空间的维数；C_k 为特征 t_k 在句子 S_i 中出现的次数；W_k 为特征 t_k 的权值；α为句子位置信息参数，当句子 S_i 位于网页的首段、尾段或者段落的首句、尾句，以及句中出现类似于“综上所述”等的指示性词语时，将α设置为大于零的值。通过对中文网页语料文摘的统计分析后，当句子分别位于首段、尾段、段落的首句、尾句和句中含有指示词语时，分别将α设置为 1.8、1.2、1、0.8 和 2。

5.5.4 基于查询条件的句子权值计算

如果返回给用户的文摘的句子中包含用户查询关键词的话，这更有利于用户判断搜索引擎返回的文档是否是他们所需，基于这样的假设，我们在计算句子重要度时增加了“查询权值”的概念，即句子与查询语句之间的关联特征。所谓查询权值[151]是指每个句子针对查询关键字统计出来的权值，具体查询权值计算如公式（5.12）所示：

$$W(q,S_k) = C^2(q,S_k)/Len(q) \tag{5.12}$$

其中，$C(q,S_k)$表示句子 S_k 中包含查询条件 q 中的关键词的总数；$Len(q)$表示查询条件中关键字个数。

5.5.5 句子重要度计算及文摘生成

句子重要度计算考虑了两个因素：文档内部存在的启发式规则和查询条件词与句子之间的关联特征，通过公式(5.13)计算文档中每个句子 S_i 的权重 $W(S_i)$，根据句子权重将网页文档中所有的句子进行降序排列，依次抽取指定长度的句子作为文摘并以原文档中的顺序返回给用户。

$$W(S_i)=W'(S_i)+W(q,S_i)=log_2(2+\alpha)\times\sum_{k=1}^{n}(C_k\times W_k)+C^2(q,S_i)/Len(q) \tag{5.13}$$

通过上述步骤可以得到一个粗略的文摘，但是这个文摘中往往会出现文摘句的冗余度较大的问题。为了利于用户进行网页相关性判断，应该在网页文摘中提供给更多的信息，有必要对文摘句进行排除冗余的处理，我们通过句子相似度计算的方法减少文摘句的冗余度，具体方法在第二章 2.6.1.4 节已经介绍，在此不再阐述。

5.6 实验与评价

5.6.1 实验数据

为了测试本文提出的方法是否有利于网页相关性判断，我们首先确定了十个不同的信息需求明确的主题，每个主题用一个或者多个关键词来描述；然后在 http://www.sogou.com 中分别通过相应的关键词对这些主题进行搜索，并从返回的网页列表中随机选择 25 个网页作为该主题的测试数据；最后，通过人为地浏览网页内容来标注每个网页是否与所需信息相关。

5.6.2 评价标准

我们从相关性判断的速度和准确率两个方面对不同算法进行了比较：

(1) 速度：即进行判断花销的时间，由实验中各种方法在所有主题的网页相关性判断中花销时间之和来表示；

(2) 准确率：由召回率 R、精确率 P 和 F_β 值来衡量。召回率(Recall)指被正确判断为相关的文档数占全部文档的比率；精确率(Precision)指被判断为相关的文档中真正相关的文档所占的比率；F_β 综合了精确率和召回率两个指标, 是算法对相关性

判断影响的整体评价，计算如公式(5.14)所示：

$$F_{\beta=1}=\frac{(\beta^2+1)\times P\times R}{\beta^2\times P+R}=\frac{2\times P\times R}{P+R} \tag{5.14}$$

5.6.3 实验方法

为方便测试者进行相关性判断，实验中采用了搜索引擎信息返回的模式：网页标题+网页摘要，其中出现的关键词用红色标记出来。本文分别采用了 Google、百度、通用摘要、基于查询的段落抽取文摘等 7 种方法对每个主题下的 25 个网页进行文摘生成，并相应的按照网页标题+文摘的模式将这 25 个网页的信息罗列出来，这样每个主题被分成了 7 组，于是我们邀请了 7 位同学参与了测试。为了避免每个测试者对信息相关的理解不一致和确保实验结果的客观性，针对每个主题，在测试者了解该主题的需求信息和不查看网页内容的前提下，分别将他们随机地分配到其中一组中进行相关性判断并统计判断过程中花费的时间。具体实验环境如图 5.3 所示：

图 5.3 网页相关性判断的实验环境

图 5.3 中标题前的多选框用来指示该网页是否与所需信息相关，如果测试者认为相关，那就选中多选框，这样可以利用 WEB 技术方便系统进行统计分析。

5.6.4 实验结果

本文采用了七种方法对网页进行文摘处理，分别是搜索引擎(Google，baidu)的返回信息，网页的通用文摘(gen-sum)、基于查询的段落抽取(q-para)、基于查询和特征词频统计的句子抽取(q-tfidf)、基于查询和词汇链统计的句子抽取(q-cfidf)和本文提出的基于多特征融合的句子重要度计算和句子抽取(qr-stf.idf)。根据评价标准对实验中的统计数据进行了处理，7 种方法分别对 10 个主题中共 250 个网页进行相关性判断的平均速度统计结果如图 5.4 所示：

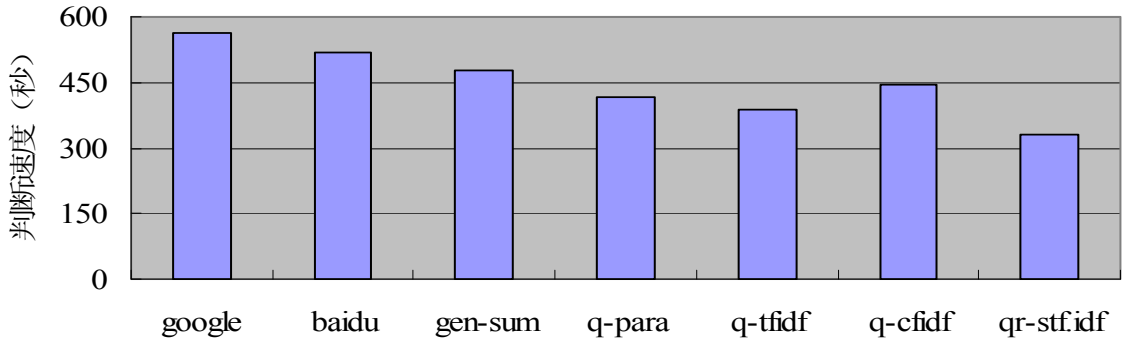

图 5.4 七种方法相关性判断的平均速度对比结果

将 Google 和百度的返回摘要与其他五种方法在 10%和 20%压缩比下生成摘要进行相关性判断测试，根据公式(5.14)准确率的统计结果如图 5.5 和如图 5.6 所示：

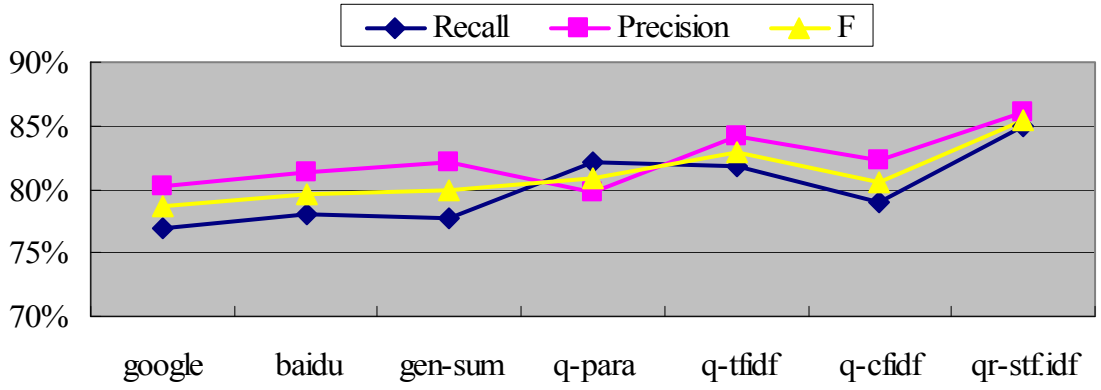

图 5.5 七种方法的准确率对比结果(压缩比=10%)

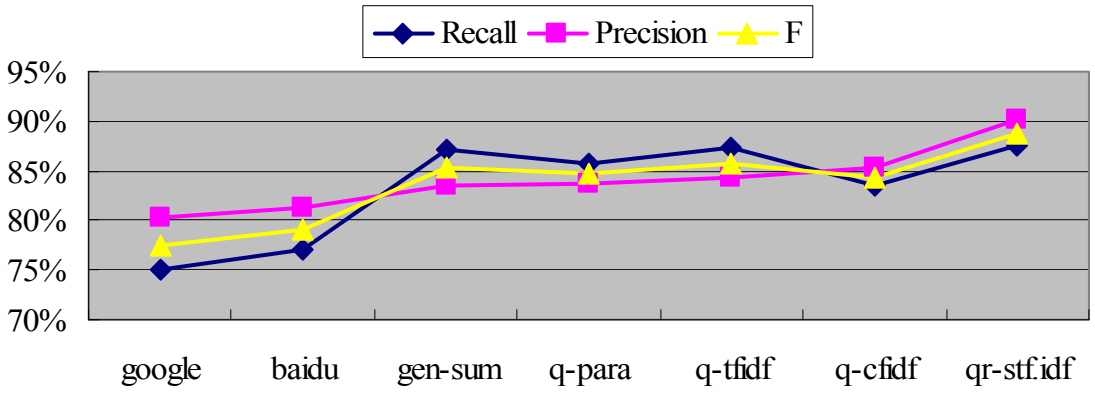

图 5.6 七种方法的准确率对比结果(压缩比=20%)

5.6.5 实验分析

从上述的对比实验结果可以看出，本文提出的面向查询的自动文摘算法在相关性判断花费的时间和准确性上都要优于其他六种方法，分析其原因，主要是通过对关键词识别和网页结构分析，可以将网页中有效的启发式规则信息和查询语句与句子之间的关联特征用于句子重要度计算，使得生成文摘同时反映了网页的重要信息和用户查询的需求信息，从而更有利于网页相关性的判断。通过上述的实验结果，我们可以得出以下两个结论：(1)无论是通用文摘还是查询文摘，都比仅用含有关键词的句子返回的信息更有利于网页相关性判断；(2)当摘要的压缩比达到 20%时，各种摘要生成的方法在网页相关性判断的性能上趋于一致；(3)针对中文网页，百度在判断速度和准确率上都要略好于 Google；(4)关键词的正确选择和关键词的个数对最终生成的网页文摘的质量有较大的影响，从而进一步影响用户进行网页相关性的判断。

5.7 本章小结

面向查询的网页自动文摘的优劣直接影响到用户判断搜索引擎返回信息是否与自己所需信息相关，如何更好地为网页进行文摘一直是信息检索领域研究的热点。本章提出了一种多特征融合的句子重要度计算的策略，即对查询输入进行短语识别和网页结构分析的基础上，将关键词短语、网页结构等启发式规则信息融入到句子重要度计算，并分析句子与查询输入的关联特征，使文摘内容能与用户需求一致，最后利用句子相似度计算的方法减少文摘句的冗余度。实验结果表明，该方法比其他基于统计和抽取方法具有更好的相关性判断表现。

由于该算法仍旧是基于用户输入的关键词的文摘生成方法，当进行复杂问题查询

时，让用户提炼出几个关键词来描述信息需求，这对一般用户来说比较困难，下一步我们研究的重点是：

(1) 能否直接用自然语言来描述信息需求，并且让系统返回一个针对该问题的摘要(即面向问题回答的自动文摘)；

(2) 本章主要讨论的是如何对 HTML 格式的网页进行面向查询的自动文摘，但是在网络中还有一些其它的文档格式(WORD、PS、PDF 和 PPT 等等)，对这些格式的文档如何进行面向查询的自动文摘；

(3) 本章只考虑了 A and B 类型的查询条件的文摘生成，对 A and (.NOT. B)类型的查询条件如何进行文摘生成需要进一步研究。

第6章 应用自动文摘提升 WEB 文本分类的性能

随着互联网技术的迅速发展和普及，大量的文字信息开始以计算机可读的形式存在，并且其数量每天都在急剧增加，人们已经从信息缺乏时代过渡到了信息极大丰富的时代。如何对浩如烟海的文献、资料和数据进行自动归类、组织和管理，已经成为一个具有重要用途的研究课题。自动文本分类技术可以在没有人工干预的情况下，通过对训练样本的智能化学习来自动产生分类模型，从而大大降低了人工工作量。因此它具有学习效率高、分类速度快、成本低的特点，适合于当今社会与日俱增的海量信息的分类任务。

6.1 引言

文本分类是在预定义的分类体系下，根据文本的特征(内容或属性)，将给定文本与一个或多个类别相关联的过程。因此，文本分类研究涉及文本内容理解和模式分类等若干自然语言理解和模式识别问题，一个文本分类系统不仅是一个自然语言处理系统，也是一个典型的模式识别系统，系统的输入是需要进行分类处理的文本，系统的输出则是与文本关联的类别。开展文本分类技术的研究，不仅可以推动自然语言理解相关技术的研究，而且可以丰富模式识别和人工智能理论研究的内容，具有重要的理论意义和实用价值。

F.Sebastiani[152]用如下的数学模型描述文本分类任务：

文本分类的任务可以理解为获得这样的一个函数 Φ：$D\times C\rightarrow\{T,F\}$，其中，$D=\left\{d_1,d_2,\cdots,d_{|D|}\right\}$ 表示需要进行分类的文档，$C=\left\{c_1,c_2,\cdots,c_{|C|}\right\}$ 表示预定义的分类体系下的类别集合。T 值表示对于<d_j, c_i>来说，文档 d_j 属于类 c_i，而 F 值对于<d_j, c_i>而言文档 d_j 不属于类 c_i。也就是说，文本分类任务的最终目的是要找到一个有效的映射函数，准确地实现域 $D\times C$ 到值 T 或 F 的映射，这个映射函数实际上就是我们通常所说的分类器。因此，文本分类中有三个关键问题：(1)文本的表示；(2)文本特征的选择或抽取；(3)分类器设计。一个文本分类系统可以简略地用图 6.1 表示。

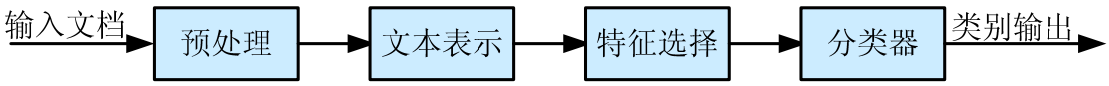

图 6.1 文本分类系统示意图

6.2 分类关键技术

6.2.1 文本表示

普通的文本文档是不能由分类器和构建分类器的算法直接处理的，需要将其转换成一个既能表示其内容，又能让分类器处理的形式。因选择文本特征的不同，目前常用的文本表示方法有：向量空间模型、*n-gram*方式、词组表示法等等。

6.2.1.1 向量空间模型

向量空间模型(VSM，Vector Space Model)是由美国的G.Salton教授在20世纪60年代率先提出的，它的基本思想就是用词袋法表示文本，即将单词作为文本的表示特征，将每一个不同的词条都看成是特征空间中的独立一维，将每一个文本看成是特征空间中的一个向量，但这种模型通常存在一个非常严重的问题，那就是维数过高和数据稀疏问题。

有些研究人员尝试用短语代替单词作为特征进行文本表示，但实验结果表明这种复杂特征的效果并不比用单词做特征时更好。Lewis[153]指出了结果不好的原因：虽然短语做特征比用词在语义质量上有一定的优势，但是在统计质量上则明显处于下风。用短语做特征的方法会造成有更多的特征，更多的同义词或近义词，在分布上更缺乏连续性，特征的文档频度更低。

6.2.1.2 *n-gram* 方式

用VSM表示文本首先要对各个文档进行分词处理。在英语等西方语言中显得非常简单的分词问题，在汉语等东方语言中却显得十分困难。而独立于语言的文本表示方式，即*n-gram*方式[154]则根本不考虑组成文本的语义单位是字、词还是词组，它是将整个文本看成是由不同字符组成的字符串，因而可以方便地表示包括汉语在内的各种语言文本文档。但*n-gram*表示法存在着数据噪声大、特征生成复杂、计算量大、易于出现过学习或过训练等缺点，通常认为其表示能力远不如向量空间模型。

6.2.1.3 词组表示法

词组表示法[155]与VSM的基本原理相同，主要区别在于前者使用词组作为基本语义单位，而后者则用单词作为基本语义单位。词组表示法通常要进行“去停用词”和“词根化处理”等操作。在基于单词的*2-gram* 表示法中，英文词组“information

retrieval”，“retrieval of information”，“retrieved information”， “informative retrieval”，都对应于同一个由词根组成的短语“inform retrieve”。

词组表示法在英文自动文本分类领域[156]取得了一系列成果。但总的来说，其文本表示能力并不明显优于VSM，主要原因在于词组表示法虽然提高了特征向量的语义含量，却降低了特征向量的统计质量，使得特征向量变得更为稀疏，使得机器学习算法难以从文本中提取有利于分类的统计特性。

6.2.2 降维技术

在自动文本分类问题中遇到的一个重要困难就是高维的特征空间，通常普通的文本经过分词后有几千甚至几万个单词，即它的特征空间维数达到几千甚至几万维，大多数的机器学习算法无法处理这么大的维数，因此，对特征空间进行降维处理是很有必要的，即减小文本的特征向量维数，保留有区分能力的特征。很多研究成果表明，特征降维可以带来几方面的好处：(1)减小问题处理的规模，提高分类效率；(2)好的特征选择方法可以删除噪声词等特征，使文本之间的相似度更为准确，即提高语义上相关文本之间的相似度，同时降低语义上不相关文本之间的相似度，提高自动分类的效率；(3)通过降维可以生成一个更紧凑、各维之间更独立的特征空间，提高分类器的推广能力。所以，对特征空间进行降维处理的结果直接影响到分类器性能的好坏，当在测试中发现分类器的效果较差时，不仅需要调整分类器的参数，更要考虑使用的特征选择方法是否合适。大部分降维方法可以归结为两类：特征选择和特征重构。

6.2.2.1 特征选择

特征选择是一个在文本分类/聚类中都要解决的一个关键问题。选择哪些特征将是一个关键的问题，这个问题一般要通过特征选择算法加以解决。特征选择就是从特征集$T=\{t_1,t_2,\cdots,t_s\}$中选择一个真子集$T=\{t_1,t_2,\cdots,t_{s'}\}$（$s'<<s$）。其中：$s$为原始特征集的大小，$s'$为选择后的特征集大小。选择的依据是特征对分类作用的大小，通常用一个统计量来度量。

常用的特征选择方法有文档频率(DF，Document Frequency)、信息增益(IG，Information Gain)、χ^2统计量等等。

1. 文档频率法

文档频率(DF)是指出现某个特征项的文档的频率。基于文档频率的特征选择法通

常的做法是：从训练语料中统计出包含某个特征的文档的频率，然后根据设定的阈值，当该特征项的 DF 值小于某个阈值时，同特征空间中去掉该特征项，因为该特征项使文档出现的频率太低，没有代表性；当该特征项的 DF 大于另外一个阈值时，从特征空间中也去掉该特征项，因为该特征项使文档出现的频率太高，没有区分度。

基于文档频率的特征选择方法可以降低向量计算的复杂度，并可能提高分类的准确率，因为按这种选择方法去掉一部分噪声特征，这种方法简单、易行。但严格地讲，这种方法只是一种借用算法，其理论根据不足。根据信息论，我们知道，某些特征虽然出现频率低，但往往包含较多的信息，对于分类的重要性很大，对于这类特征就不应该使用 DF 方法将其直接排除在向量特征之外。

2. 信息增益法

信息增益法(IG)依据特征项 t_i 为整个分类所能提供的信息量多少来衡量该特征项的重要程度，从而决定对该特征项的取舍。特征项 t_i 的信息增益是指有该特征或没有该特征时，为整个分类所能提供的信息量的差别，其中，信息量的多少由熵来衡量。因此，信息增益是不考虑任何特征时文档的熵和考虑该特征后文档的熵的差值：

$$\begin{aligned} Gain(t_i) &= Entropy(S) - Expected\ Entropy(S_{t_i}) \\ &= -\sum_{j=1}^{M} P(C_j) log_2 \left(P(C_j)\right) + P(t_i) \sum_{j=1}^{M} P(C_j \mid t_i) log_2 P(C_j \mid t_i) \\ &\quad + P(\overline{t_i}) \sum_{j=1}^{M} P(C_j \mid \overline{t_i}) log_2 P(C_j \mid \overline{t_i}) \end{aligned} \tag{6.1}$$

其中，$P(C_j)$表示 C_j 类文档在语料中出现的概率，$P(t_i)$表示语料中包含特征项 t_i 的文档的概率，$P(C_j|t_i)$表示文档包含特征项 t_i 时属于 C_j 类的条件概率，$P(\overline{t_i})$表示语料中不包含特征项 t_i 的文档的概率，$log_2 P(C_j|\overline{t_i})$表示文档不包含特征项时属于 C_j 类的条件概率，M 表示类别数。

从理论上讲，信息增益应该是最好的特征选取方法，但实际上由于许多信息增益比较高的特征出现频率往往较低，所以，当使用信息增益选择的特征数目比较少时，往往会存在数据稀疏的问题，此时分类效果也比较差。

3. χ^2 统计量

χ^2 统计量(CHI)衡量的是特征项 t_i 和类别 C_j 之间的相关联程度，并假设 t_i 和 C_j 之间符合具有一阶自由度的 χ^2 分布。特征对于某类的 χ^2 统计值越高，它与该类之间的相关性越大，携带的类别信息也越多，反之则越少。

如果令 N 表示训练语料中文档的总数，A 表示属于 C_j 类且包含 t_i 的文档频数，B

表示不属于 C_j 类但包含 t_i 的文档频数，C 表示属于 C_j 类但不包含 t_i 的文档频数，D 是既不属于 C_j 类也不包含 t_i 的文档频数。

特征项 t_i 对类别 C_j 的 CHI 值为：

$$\chi^2\left(t_i,C_j\right)=\frac{N\times\left(A\times D-C\times B\right)^2}{\left(A+C\right)\times\left(B+D\right)\times\left(A+B\right)\times\left(C+D\right)} \tag{6.2}$$

对于多类问题，基于 CHI 统计量的特征提取方法可以采用两种实现方法：一种方法分别计算 t_i 对于每个类别的 CHI 值，然后在整个训练语料上计算：

$$\chi^2_{MAX}\left(t_i\right)=\max_{j=1}^{M}\chi^2\left(t_i,C_j\right) \tag{6.3}$$

其中，M 为类别数。从原始特征空间中去除统计量低于阈值的特征，保留统计量高于给定阈值的特征作为文档特征。另一种方法是计算各特征对于各类别的平均值：

$$\chi^2_{AVG}\left(t_i\right)=\sum_{j=1}^{M}P\left(C_j\right)\chi^2\left(t_i,C_j\right) \tag{6.4}$$

6.2.2.2 特征重构

特征重构主要是将原有的特征集T加以联系和转化以构建新特征集T'的过程，一般$|T'|<<|T|$，从而达到降维的目的。由于一词多义、多词一义的现象大量存在于文本信息中，导致文本的原始项可能不是文档内容表示的最佳维度，特征重构就是试图通过重构新项来避免上述问题，一般有项聚类(Term Clustering)、潜在语义索引(LSI，Latent Semantic Index)、主成分分析(PCA，Principal Component Analysis)、多维尺度变换、自组织特征映射(SOM，Self-organizing Map)等特征重构的方法。

其中，最为常用的特征重构方法是潜在语义索引 LSI，它基于这样一种假设：“文档中存在潜在语义结构，这种语义结构由于部分的被文档中词的语义和形式上的多样性所掩盖而不明显”。LSI 通过对原文本集中的词-文档矩阵进行奇异值分解计算，计算出文档集合中潜在内涵的概念之间关系，通过潜在概念集表示所有的概念空间，减少了概念表示之间的模糊性，从而避免了向量空间模型中各维之间概念正交的假设。它将特征项-文档关联数据的不可靠性看作是一种统计问题，认为在数据中存在一种潜在的语义结构，这种结构由于检索词出现的多样性被干扰，LSI 使用统计技术去估计这些潜在的语义结构，去掉这种“噪声”，因此，LSI 也可以被看成是向量空间模型的一种映射。

6.2.3 分类算法

对文档进行特征项的提取和表示之后，就可以对文档运用分类算法进行分类了。目前已经有很多文本分类算法，其中包括 K 近邻[157]、Rocchio[158]、朴素贝叶斯[159]、决策树[160]、支持向量机[161]等算法。在分类时，根据不同的情况选取针对其特征的不同分类方法。下面就其中的几种常用算法进行介绍：

1. K 近邻算法（KNN，K Nearest Neighbor）

K 近邻算法是一种传统的基于统计的分类方法[157]，是根据测试样本在特征空间中 k 个最近邻样本中的多数样本的类别来进行分类。算法思想为：对于一篇待分类文档 $\vec{x}$，系统在训练集中找到与该文档距离最近（最相似）的 k 个最相近的邻居，如果这 k 篇文本多数属于 C_j 类，则新文档 $\vec{x}$ 属于类别 C_j。计算公式如公式（6.5）所示：

$$y\left(\vec{x},C_j\right)=\sum_{\vec{d}_i\in KNN} sim\left(\vec{x},\vec{d}_i\right)y\left(\vec{d}_i,C_j\right) \tag{6.5}$$

其中，$\vec{x}$ 为一篇待分类文档的向量表示；$\vec{d}_i$ 为训练集中文档的向量表示；C_j 为一类别；$y(\vec{d}_i,C_j)\in\{0,1\}$（当 $\vec{d}_i$ 属于 C_j 时取 1；当 $\vec{d}_i$ 不属于 C_j 时取 0）；$sim\left(\vec{x},\vec{d}_i\right)$ 为相似度计算公式，计算待分类文档与训练集中文档之间的距离，从而从训练文档集中选出与新文档最相似的 k 篇文本。

2. Rocchio 算法

Rocchio 算法来源于向量空间模型理论，它训练的过程其实就是建立类别特征向量的过程，并且按照正例和反例为组的形式处理每一个特征词[158]。

Rocchio 算法的突出优点是容易实现，计算（训练和分类）特别简单，它通常用来实现衡量分类系统性能的基准系统，而实用的分类系统很少采用这种算法解决具体的分类问题。

3. 朴素贝叶斯（NB，Naïve Bayes）

NB 算法是基于贝叶斯全概率公式的一种分类算法[159]。其基本思路是计算文档属于类别的概率，等于文档中每个词属于类别的综合表达式，根据计算结果将文档分到概率最高的类别中。

NB 算法假设文档之间的特征项都是相互独立的。但是，这一假设对语义丰富的语言文字信息往往过于简单，这也在一定程度上限制了算法的性能。NB 算法需要使用训练集对分类器进行训练，也就是需要分别计算每个 $P(a_i|C)$。假设训练集共有 m 个类别，n 个特征项，待分类文档共有 k 个特征项，那么训练的时间复杂度为 $O(m\times n)$，

分类的时间复杂度为 $O(k)$。

4. 决策树算法(Decision Tree)

决策树是一个类似于流程图的树结构[160]，其中每个内部节点表示在一个属性上的测试，每个分支代表一个测试输出，而每个树叶节点代表类或类的分布。

决策树算法是一种贪心算法，通过对训练数据的学习，总结出一般化的规则，然后再利用这些规则解决问题。即它以自顶向下的方式在训练集的基础上为预先定义的每一个类构造一棵决策树，之后取未知文本的属性在决策树上测试。路径由根结点到叶结点，从而得到该文本所属的类别。

5. 支持向量机(SVM，Support Vector Machine)

SVM 的算法思想为：首先通过非线性变换将输入空间变换到一个高维空间，然后在这个新空间中求最优线性分类面，而这种非线性变换是通过定义适当的内积函数实现的[161]。

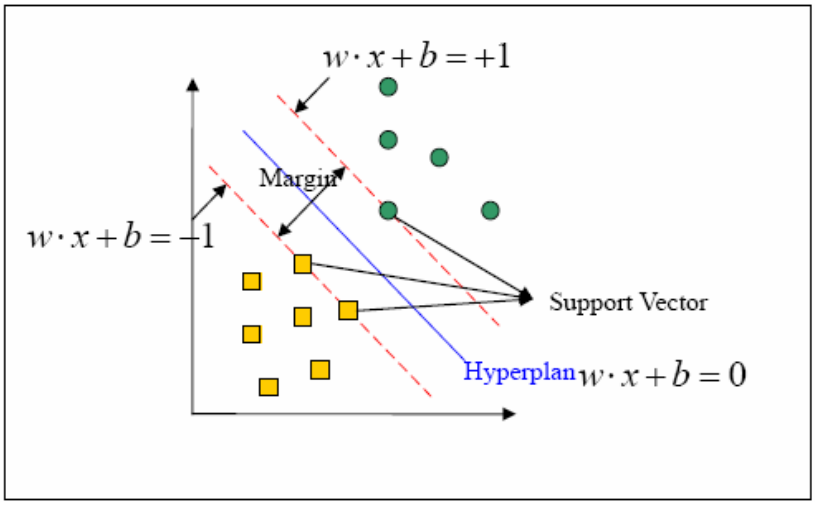

图 6.2 SVM 分类模型

6.3 应用自动文摘的文本分类系统

无论选择哪一个特征评价函数来选择特征项，特征空间的维数都是非常高，在中文文本分类中，该问题表现得更为突出，而高维特征向量对分类器训练和测试存在不利的影响，很容易出现模式识别中的“维数灾难”现象。同时，通过人工分析发现并非所有的特征项对分类都是有利的，很多提取出来的特征可能是噪声。因此，如何降低特征向量的维数，并尽量减少噪声，仍然是文档特征提取亟需解决的问题。

基于句子抽取的自动文摘系统生成的摘要在一定程度上覆盖了原文档中所有的重要信息，用户完全可以通过阅读摘要了解原文的意思表达，所以摘要可以作为原文

档的某种替代。既然通过阅读原文 10%~30%的摘要，用户可以了解原文的大概信息，那么就一定能够依据摘要确定文章所属的类别，鉴于上述考虑，我们设想是否可以直接用摘要参与传统的文本分类的特征选择或者是在摘要的基础上挑选特征参与分类器训练，以降低特征选择和训练的运算量和提高文本分类的性能。

6.3.1 系统框架

基于上述想法，我们在传统的文本分类流程中增加了对训练文本进行自动文摘操作，具体流程如图 6.3 所示：

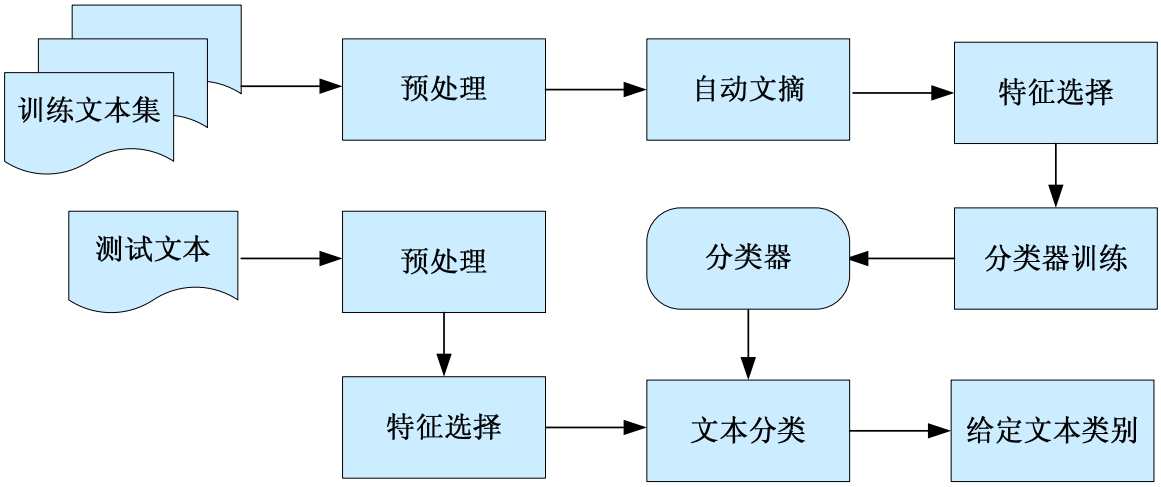

图 6.3 应用自动文摘的文本分类系统体系结构

6.3.2 预处理

预处理的功能是将非结构化文档或半结构化文档表示成计算机可表示、可计算的向量空间模型，预处理主要包括网页去噪、分词、未登录词识别、分段或分句和停用词过滤等等操作。其中，网页去噪主要是在过滤掉 WEB 网页文档中的 HTML 标签、导航栏和广告图片等信息的同时，识别 WEB 网页的大小标题和正文信息；分段或分句主要根据自动文摘系统所采用的粒度来决定；停用词过滤是将文档集中出现的一些词频极高或者虚词、副词和数词等不包含任何文本信息的词汇去除。

6.3.3 自动文摘

在中文自动文摘领域，很多研究成果表明特征的选择应以名词和动词为主，其根据的理由是，如果将文件中的冠词、副词、形容词以及介词等词汇删除，读者仍然能够知道文件的意思表达，充分说明名词和动词相对比较重要，对文本所属类别的判断

起决定的作用，又因为本文的自动文摘系统生成的摘要主要用于文本分类，所以我们在进行文摘的过程中仅仅考虑了名词和动词，具体步骤如下：

（1） 计算文档名词和动词的权重

$$W(t)=tf_t\times log_2\left(\frac{N}{n_t}\right) \tag{6.6}$$

其中，$W(t)$表示单词 t 的权重；tf_t 表示单词 t 在其所属文档中出现的频率；N 表示训练集的文件总数；n_t 表示单词 t 在训练集中多少个文档出现。

（2） 计算动词和名词相互间的距离

基于文档是有生命的文字组合在一起的观点，语义上相关的单词在文档中出现的距离间距不会太长，否则彼此之间的连贯性和意思表达的效果就会大打折扣。因此，采用距离的概念能够确实反映作者的撰写行为[162]。词汇 X 与词汇 Y 之间的距离可用公式(6.7)求得：

$$D(X,Y)=Arg\,min\left(ABS\left(C(X)-C(Y)\right)\right) \tag{6.7}$$

其中，函数 ABS 是求绝对值；$C(X)$表示单词 X 在分词后文档中所处的位置；函数 $Argmin$ 表示当文档中出现多个 X 和 Y 时，取最小值作为单词 X 与 Y 间的距离 $D(X,Y)$。

（3） 计算动词和名词间的强度（CS , Connective Strength）

根据文献[162]，句子 S_i 的重要度可由其包含名词的强度之和决定；名词的强度 $CS(n)$可由该名词与其它名词的强度 $SNN(n)$及该名词与其它动词的强度 $SNV(n)$来获得。

$$CS(n)=SNN(n)+SNV(n) \tag{6.8}$$

$$SNN(n_i)=\sum_j\frac{W(n_i)\times W(n_j)}{D(n_i,n_j)} \tag{6.9}$$

$$SNV(n_i)=\sum_j\frac{W(n_i)\times W(v_j)}{D(n_i,v_j)} \tag{6.10}$$

由于动词本身也包含文档的重要信息，因此本章在计算句子重要度时同时考虑了动词的强度。

$$CS(v)=SVN(v)+SVV(v) \tag{6.11}$$

$$SVN(v_i)=\sum_j\frac{W(v_i)\times W(n_j)}{D(v_i,n_j)} \tag{6.12}$$

$$SVV(v_i)=\sum_j \frac{W(v_i)\times W(v_j)}{D(v_i,v_j)} \tag{6.13}$$

（4） 计算句子的重要度

假设句子 S_i 包含 m 个名词(n_j)和 n 个动词(v_k)，则句子 S_i 的重要度 $W(S_i)$可由公式(6.14)求得：

$$W(S_i)=\alpha\times\sum_{j=1}^{m}CS(n_j)+\beta\times\sum_{k=1}^{n}CS(v_k)+\lambda\times Title_{S_i}+\delta\times Position_{S_i} \tag{6.14}$$

其中，α、β、λ 和δ是权重系数，满足α+β+ λ+δ=1；$Title_{S_i}$ 表示 S_i 是否为大小标题的权重，如果 S_i 为标题，$Title_{S_i}$=1，否则$Title_{S_i}$=0； $Position_{S_i}$ 表示 S_i 在文档中所处位置的权重，如果 S_i 位于文档的篇首或者段落的开头，$Position_{S_i}$=1；如果 S_i 位于文档的尾部或者段落的结尾，$Position_{S_i}$=0.6；否则$Position_{S_i}$=0。

在实验中，把参数设为：α=0.4，β=0.35，λ=0.15，δ=0.1。

（5） 摘要生成

在对文档中每个句子进行重要度计算后，首先根据重要度对所有句子进行降序排序，然后按照预定的比例抽取句子组成摘要。由于摘要只是作为文本分类的中间载体，不连贯、不简洁的文摘对分类结果没有任何影响，故对文摘句的顺序不需做排序处理；又因为重要度更高的句子中包含的信息对类别的判断更是有利，故不需要进行文摘句的冗余处理。

6.3.4 特征选择

文本分类的特征选择有两种方式：全局选择和局部选择。全局选择是指从整个文档集的原始特征集中选择特征，选择后得到的特征对于各类别都是一致的[163]。局部选择是指以类别为单位对原始特征集进行选择，选择后各类别具有不同的特征集[164]。一些研究的分类结果表明，采用局部特征选择的分类效果要优于全局特征选择[165]，但是全局特征选择能够从总体上控制特征集的大小，有利于对特征选择问题进行深入的研究。

在生成摘要的基础上，本章分别采用了三种特征选择的方法：(1)直接将摘要代替原文用常用的特征选择算法(互信息、交叉熵等等)进行特征选择；(2)针对KNN分类算法，本文采用为训练集中的每个文件单独选择特征，即将文件对应的摘要中的所有词汇(除停用词)作为该文件的特征词；(3)采用局部选择特征的方式，为每个类别单独进行候选特征选择，首先，为每个类别的训练文件进行摘要生成，从所有摘要中抽

取出名词和动词；然后，将这些名词和动词组成一个候选特征词的集合，并对这些候选特征词进行权重计算；最后，根据权重将这些候选特征词进行降序排列，按照预设的特征个数确定分类使用的特征集合，从而完成特征选择。

6.3.5 特征权值计算

在特征选择结束后，需要分别对每个文件或类别的候选特征进行权值计算，我们采用了文献[166]中表现最好的权重计算公式，并对其进行了归一化：

$$W(t_i,c)=\frac{log_2\left(tf_{t_i,c}\right)\times log_2\left(idf_{t_i}+0.01\right)}{\sqrt{\sum_{k=1}^{n}\left(log_2\left(tf_{t_i,c}\right)\times log_2\left(idf_{t_i}+0.01\right)\right)^2}} \tag{6.15}$$

其中，$W(t_i, c)$是特征 t_i 在类别 c 或文件 c 中的权重；$tf_{t_i,c}$ 是特征 t_i 在类别 c 或文件 c 中出现的次数；idf_{t_i} 是文档总数与含有特征 t_i 的文档数的比值。

6.3.6 分类

在对每个文件进行特征选择和权重计算后，采用公式(6.5)计算待分类文本 $\vec{x}$ 与所有训练集文件之间的距离，并从其中找到与文本 $\vec{x}$ 距离最近的 K 个最相近的邻居，如果这 K 篇文本多数属于 C_j 类，则新文本 $\vec{x}$ 被归于类别 C_j。

在每个类别的特征选择和权值确定后，对每个需要分类的文本采用公式(6.16)计算该文本与每个类别的相关度[166]，并将这一文本归类到相关度最大的一个类别或者相关度大于某一阈值的某几个类别。

$$R(c,d)=\sum_{t\in F_c\cap d} log_2\left(tf_{t,d}\right)\times W(t,c) \tag{6.16}$$

其中，$R(c, d)$是测试文档 d 与类别 c 的相关度；F_c 是类别 c 的特征集；$tf_{t,d}$ 是特征 t 在文档 d 中出现的次数；$W(t, c)$是特征 t 在类别 c 中的权值，由公式(6.15)计算而得。

6.4 实验与评价

6.4.1 实验数据

6.4.1.1 训练集和测试集

在文本分类系统中，通常将用于实验的语料库被分为两部分：训练集和测试集。所谓训练集是由一些已经完成分类(即已给定类别标号)的文本组成，用于归纳出各个

类别的特性以构造分类器；根据分类体系的设定，每一个类别都应含有一定数量的训练文本。测试集是用于测试分类器的分类效果的文档的集合，其中每个文本都通过分类器分类，然后与正确决策的分类结果相对比，从而得到对分类器效果的评价，但测试集不参与分类器的建设。

6.4.1.2 语料库的选取

由于中文分类系统没有统一的语料供系统测试使用，所以我们以国际 TREC 语料格式为标准，从中华网、新浪、搜狐和雅虎等权威网站获取特定主题网页，结合手工标注，建立了 20 个类别的语料库，其中训练语料库共 7856 篇文档，测试语料库共 6879 篇文档。由于有些类别中包含的文档数量过少，使得分类器从文本中学习到的知识较少，会导致分类器在这些类别上的性能不佳，所以本文在实验中选取了文档数较多的 10 个类别进行测试，共包括 4637 篇训练文档，并从这 10 个类的测试集中每类选取 300 个文档，共 3000 篇文档作为测试文档。这 10 个类别的文档分布如表 6.1 所示：

表 6.1 语料中类别文档数目分布

类别名称	训练集	测试集	类别名称	训练集	测试集
农业	465	300	经济	425	300
艺术	502	300	医药	480	300
教育	510	300	军事	424	300
交通	456	300	政治	505	300
环境	420	300	体育	450	300

6.4.2 评价标准

文本分类从根本上说是一个映射过程，评估文本分类系统的指标是映射的准确程度和映射的速度。映射的速度取决于映射规则的复杂程度，而评估映射准确程度的参照物是通过专家思考判断后对文本的分类结果，与人工分类结果越相近，分类的准确程度就越高，这里隐含了评估文本分类系统的两个指标：精确率(Precision)和召回率(Recall)。对于某一文档类别，它们分别定义为：

$$\mathrm{Pr}\,ecison = \frac{TP}{TP + FN} \tag{6.17}$$

$$\mathrm{Re}\,call = \frac{TP}{TP + FP} \tag{6.18}$$

其中，*TP*是判断属于该类并且正确的文档数目；*FN*是判断属于该类但错误的文档数目；*FP*是判断错误的本属于该类的文档数。

精确率和召回率反映了分类质量的两个不同方面，但这两个比率是两个互相矛盾的衡量标准。一般情况下，Precision 会随着Recall 的升高而降低，两者不可兼得，所以很多情况下需要将它们综合在一起考虑。最常用的综合方法就是F-Measure，其数学公式如下：

$$F_{\beta}(P,R)=\frac{(\beta^2+1)\times P\times R}{\beta^2\times P+R} \tag{6.19}$$

其中，β是一个调整参数用于以不同的权重综合Precision 和Recall。当β等于1时，Precision 和Recall 被平等对待，这时F-Measure 又被称为F_1 值，如公式(6.20)所示。

$$F_{\beta=1}(P,R)=\frac{(\beta^2+1)\times P\times R}{\beta^2\times P+R}=\frac{2\times P\times R}{P+R} \tag{6.20}$$

6.4.3 实验一：摘要代替原文的分类实验

为了验证将摘要代替原文进行特征选择和分类器训练对分类性能的影响，我们采用了信息增益IG的特征选择算法，为整个训练集进行全局选择2500个特征，分别在原文档内容、10%文档摘要、20%文档摘要和30%文档摘要的基础上，采用朴素贝叶斯(NB，Naïve Bayes)分类算法对分类器进行训练，并对特征选择和分类器训练花费时间(包括网页清理、分词、停用词过滤、文摘、特征选择、特征权重计算和排序等等)及分类的召回率和精确率进行了比较，具体实验结果如图6.4、图6.5和图6.6所示：

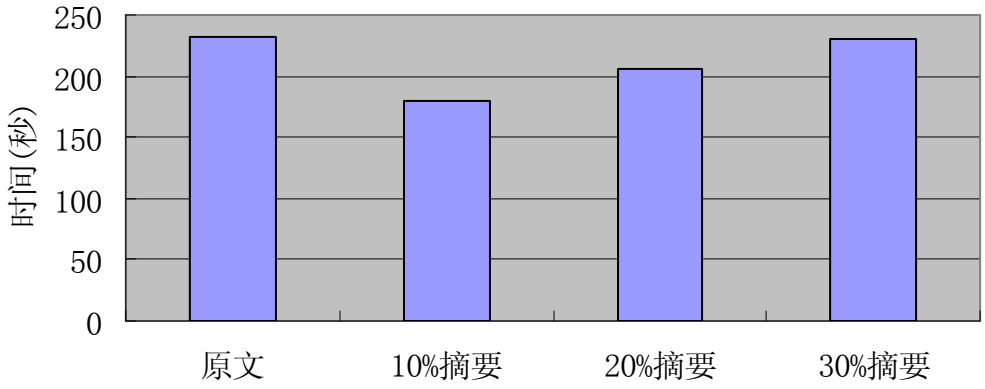

图 6.4 摘要和原文档内容用于特征选择和分类器训练花费时间比较

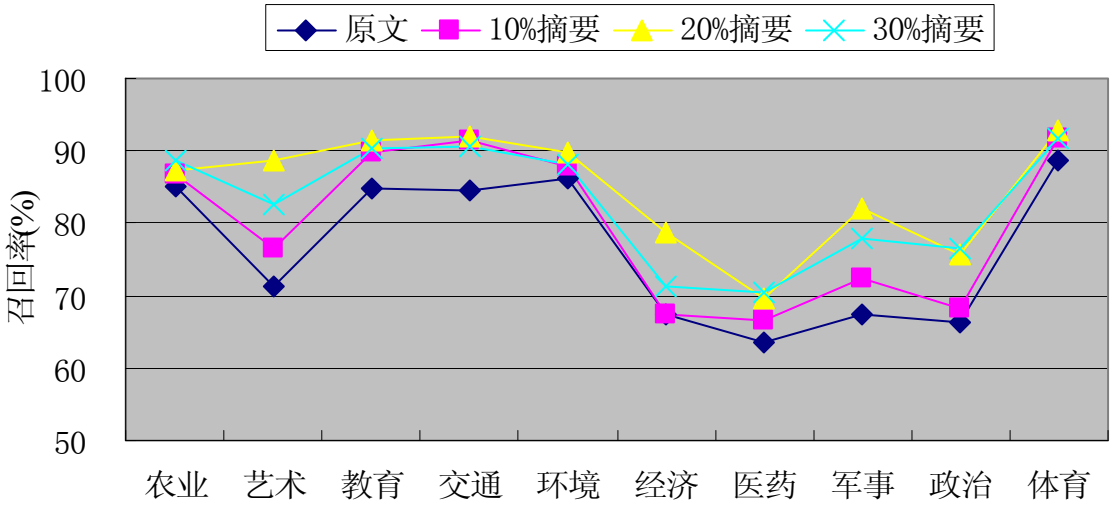

图 6.5 摘要与原文档内容进行分类的召回率比较

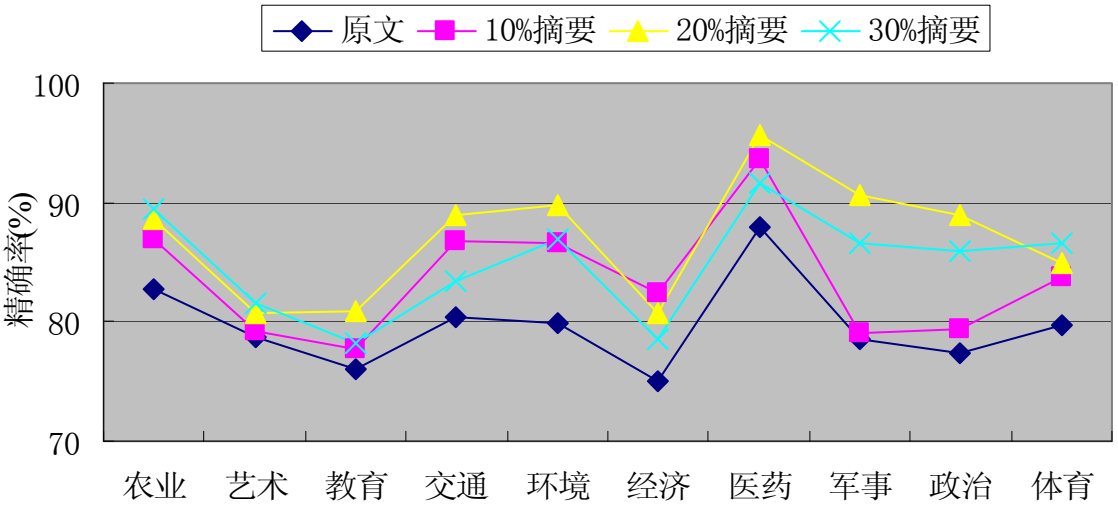

图 6.6 摘要与原文档内容进行分类的精确率比较

6.4.4 实验二：利用摘要进行文档单独特征选择的分类实验

为了验证利用文档摘要对每个文档单独进行特征选择是否有利于 KNN 分类算法，我们对 4 个分类方法进行了测试比较：(1)利用 20%摘要对每个文档单独特征选择 + KNN 算法(Method A)；(2)利用 30%摘要对每个文档单独特征选择 + KNN 算法(Method B)；(3)利用 MI 进行局部特征选择(类别)+ KNN 算法(Method C)；(4)利用 MI 进行全局特征选择 + KNN 算法(Method D)。具体实验对比结果如表 6.2 和表 6.3 所示：

表 6.2 用摘要对文档单独特征选择与 MI 特征选择进行 KNN 分类的精确率比较

	类别	农业	艺术	教育	交通	环境	经济	医药	军事	政治	体育
K=50	Method A	80.64	78.78	76.18	84.01	86.76	78.66	91.65	81.65	83.89	82.97
	Method B	81.56	76.06	74.56	82.04	87.45	80.13	87.28	75.33	81.64	78.49
	Method C	79.35	76.54	71.38	81.41	83.05	78.47	88.02	76.05	79.26	78.99
	Method D	75.61	72.48	65.14	78.95	82.68	73.24	82.36	72.51	76.65	73.46
K=100	Method A	84.36	79.09	82.28	83.64	85.06	75.44	89.52	83.36	80.17	79.27
	Method B	81.05	76.77	77.10	79.35	87.12	77.37	84.73	79.79	75.77	78.99
	Method C	76.69	76.54	73.23	77.26	80.69	71.47	87.47	81.67	79.10	80.10
	Method D	72.13	73.90	69.42	78.62	82.49	63.11	81.53	74.42	73.81	75.97

表 6.3 用摘要对文档单独特征选择与 MI 特征选择进行 KNN 分类的召回率比较

	类别	农业	艺术	教育	交通	环境	经济	医药	军事	政治	体育
K=50	Method A	85.22	83.69	88.43	87.05	89.68	73.85	64.68	78.02	70.42	88.56
	Method B	82.96	79.63	86.87	82.35	85.89	75.69	70.64	73.49	71.55	86.03
	Method C	82.58	81.69	83.32	77.13	86.13	77.56	63.34	69.65	70.31	83.63
	Method D	80.21	77.46	78.71	72.24	79.25	68.38	56.99	64.86	66.02	78.61
K=100	Method A	83.15	83.23	89.81	85.58	86.56	77.97	66.74	76.31	79.39	84.08
	Method B	81.65	81.58	87.66	79.09	80.60	81.53	74.06	73.68	77.14	81.63
	Method C	78.39	78.36	84.01	74.30	83.45	75.12	63.69	71.55	75.89	79.42
	Method D	74.14	71.32	80.23	67.78	75.24	70.64	59.25	68.19	72.16	73.68

6.4.5 实验三：利用摘要进行类别特征选择的分类实验

为了验证利用文档摘要对每个类别进行特征选择是否有利于分类，我们用 6 个特征选择方法为每个类别选择了 1500 个特征，进行了分类测试比较：(1)利用 10%摘要对每个类别特征选择 + SVM 算法；(2)利用 20%摘要对每个类别特征选择+ SVM 算法；(3)利用 30%摘要对每个类别特征选择+ SVM 算法；(4)利用 DF 进行类别特征选择+ SVM 算法；(5)利用 MI 进行类别特征选择 + SVM 算法；(6)利用 IG 进行类别特征选择 + SVM 算法。精确率、召回率和 F 值的对比实验结果如图 6.7、图 6.8 和图 6.9 所示：

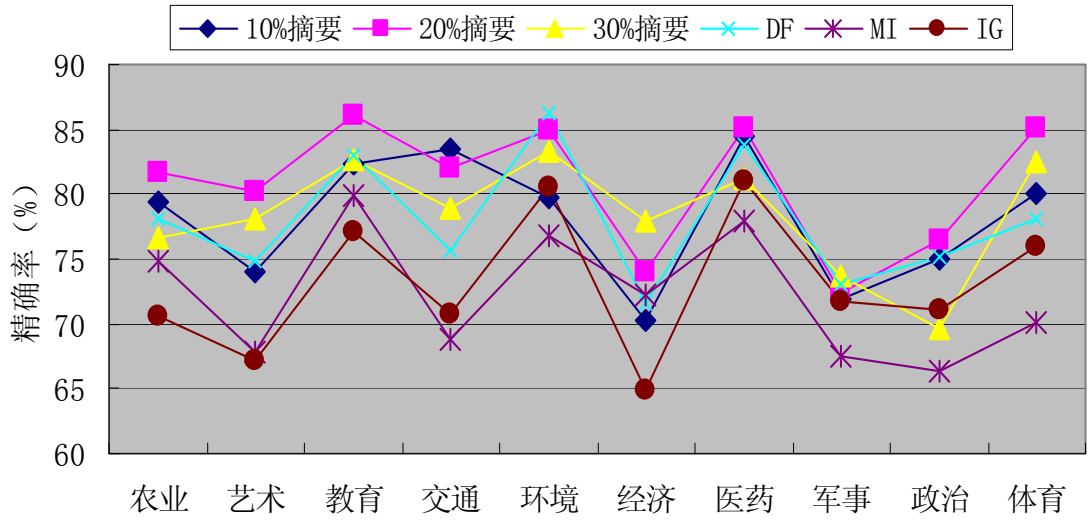

图 6.7 四种类别特征选择进行 SVM 分类的精确率比较

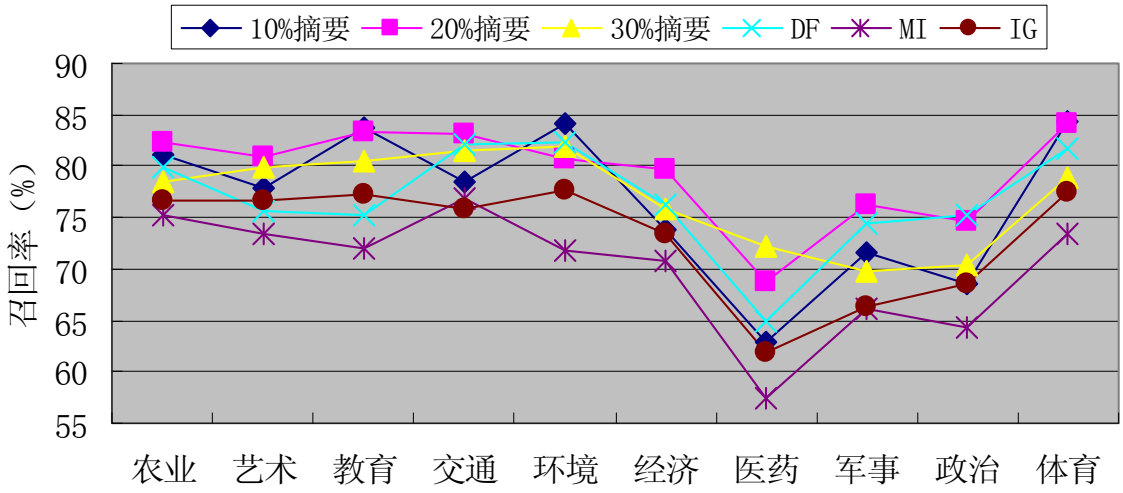

图 6.8 四种类别特征选择进行 SVM 分类的召回率比较

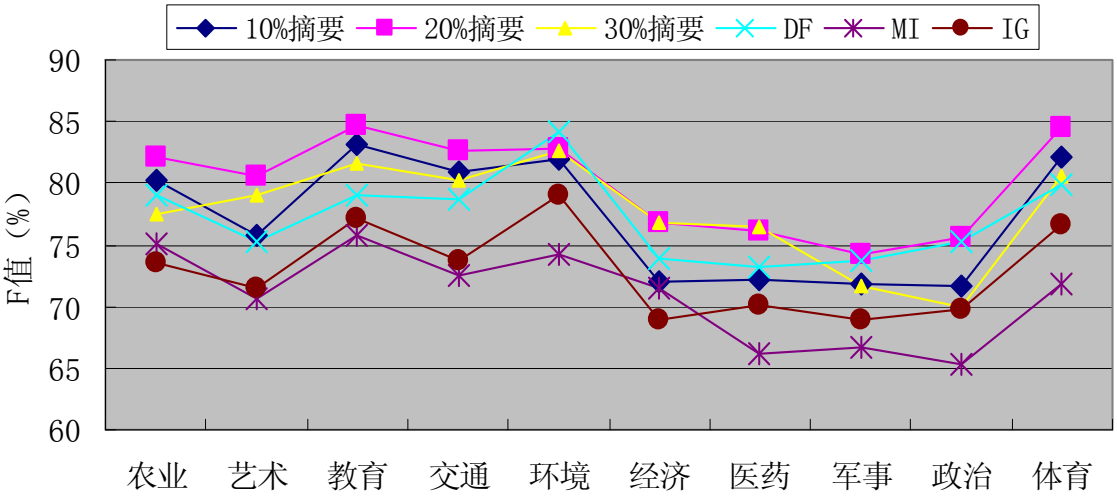

图 6.9 四种类别特征选择进行 SVM 分类的 F 值比较

6.4.6 实验分析

从实验 1 的对比实验结果可以看出，用摘要代替原文档进行特征选择和分类器训练花费的时间会更少，分析每个阶段的时间开销，我们发现虽然用原文档训练的方法省略了文摘的时间，但是其对原训练集中所有文档中每个词汇进行权重计算和按照权重对所有词汇进行排序的时空开销要远远大于用文摘训练的方法。此外，用摘要进行特征选择和分类训练的方法还可以获得更好的分类性能，分析其原因，主要是摘要是原文档重要信息的浓缩，其保留了对特征选择和分类训练的关键信息的同时，也排除了影响特征选择和分类训练的噪音，从而大大降低了特征选择的计算量和提高了分类的性能，其中，压缩率为 20%的摘要表现出来的分类性能最佳，主要是 10%摘要的内容过少，不能提供原文档所有的重要信息，有待选择的特征过少；相反，30%摘要在提供有效特征的同时，也隐含了很多影响分类的噪音。

在面向 KNN 分类算法的实验 2 中，利用文档摘要为每个文档单独特征选择与 MI 特征选择方法(全局和局部两个方面)进行了比较，从对比实验结果可以看出，利用文摘为文档单独选择特征和 KNN 分类的效果要好于 MI 特征选择方法，但是我们也发现实验 2 中的分类效果要略低于实验 1，分析其原因，主要是单文档中包含类别的特征是有限的，仅仅依赖于两个文档之间词汇进行硬匹配式的相似度计算不足以说明两文档的相关程度，因此，能否在相似度计算的过程中，融入基于 HOWNET 和《同义词词林》等知识库的语义概念提高相似度计算的准确性和可靠性，从而进一步提高 KNN 分类性能还有待进一步研究。

在进行类别特征选择的实验 3 中，用摘要进行类别特征选择和 SVM 算法进行分类的性能要优于传统的特征选择方法(MI、DF 和 IG 等)，分析其原因，主要是针对中文文档，其内容主要是通过名词和动词来表述且对文本所属类别的判断起决定的作用，而本章提出的用摘要进行类别特征选择的方法中，首先利用摘要将文档的重要信息抽取出来，去除了大量的噪音信息；然后在摘要的基础上考虑将名词和动词作为特征的候选项，从而使得该方法选择的特征更有代表性、更利于类别判定。此外，虽然 DF 用于文本分类的特征选择的理论根据不足，但是其表现要远远好于其它特征选择算法(例如：IG 和 MI)。

综上所述，本章提出的应用自动文摘进行文本分类的方法在降低特征选择计算量、缩短分类器训练时间和提升分类性能有着明显的改善作用，此外，用该方法进行分类的副产品是为训练集中的每个文档提供了提示性摘要，供用户评估分类结果是否

恰当使用。

6.5 本章小结

文本分类是信息处理领域的一个重要分支，向量空间模型 VSM 以其有利于数字化处理的特点而在文本分类任务中被广泛应用。但是，向量空间模型的数据稀疏，特征空间维数过高的问题是限制它进一步发挥作用的主要方面，多年来研究人员针对该模型的特点提出了多种算法来解决降维和数据稀疏的问题。但这同时也带来了一个新问题，即大量的分类特征被舍弃的是否合理，以及这些舍弃的特征对分类性能有何影响尚未有人研究。

为了有效地舍弃噪音的同时，保留有利于分类的特征，在基于文档摘要在一定程度上覆盖了原文档中所有的重要信息和可以将摘要作为原文档的某种替代的假设，我们提出了利用机器文摘进行特征选择和分类器训练的方法。为了验证利用文档摘要进行特征选择和分类是否能有效提高分类性能，我们设计了 3 个实验，对比实验结果表明，文档摘要能够保留原文档中相对重要的特征，去除了大量不利于分类的噪音，大大降低了特征选择和分类的计算量，提高了分类的速度和性能。

但是，随着互联网技术的迅速发展和普及，对网络内容管理、监控和过滤有害(或垃圾)信息的需求越来越大，网络信息的主观倾向性分类受到越来越多的关注[167]。这种倾向性分类与传统的文本分类不同，传统的文本分类所关注的是文本的客观内容，而倾向性分类所研究的对象是文本的“主观因素”，即作者所表达出来的主观倾向性，分类的结果是对于一个特定的文本要得到它是否支持某种观点的信息，这种独特的文本分类任务被又称为文本情感分类(Sentiment Classification)。

目前针对中文情感分类的研究相对较少，且采用的方法大多数只是利用从训练语料中分别估计出代表“赞扬”和“批评”两种情感倾向的语言模型的方法进行情感分类的[168-169]，没有考虑到情感中还有一类是“中立”。因此是否可以用“半监督聚类”的方法实现三种情感的文本情感分类(即：首先利用训练语言进行情感信息统计，然后并将统计信息融入到文本信息的聚类过程中，最后对每一类的信息进行评估加权来确定文本的情感所属)，下一步我们将对此做一些尝试性的工作。

结 论

本论文首先介绍了自动文摘的分类、发展历程、主要研究内容、技术路线和评测方法。然后，对于单文档文摘，给出了一种在关键词抽取的基础上进行文摘句抽取的方法；对于多文档文摘，提出了基于聚类分析的局部主题识别和基于语义分析的文摘句润色处理的多文档文摘方法，旨在提高获取多文档集合信息的全面性和文摘信息的平衡性；对于抽取得到的文摘句，还给出了一种用于多文档文摘句排序的改进MO算法，旨在提高文摘的可读性。此外，针对目前网页相关性判断的速度慢和准确率低的情况，提出了一种多特征融合的句子权重计算和查询文摘生成的方法；为了进一步降低特征空间的维度和减轻分类计算的复杂性，提出了利用摘要进行特征选择和分类器训练的方法。具体说来，本文主要的创新工作如下：

(1)提出了一种基于词汇链构建的中文关键词抽取的算法。首先利用相邻次共现进行未登录词识别；然后利用《知网》作为知识库，通过计算词义相似度构建词汇链；最后结合词汇所在词汇链的强度、信息熵和出现位置等属性进行关键词抽取。该方法考虑了术语识别和词汇之间的语义信息，能够避免抽取的关键词局限于某一语义范围内，明显改善了关键词抽取的效果。

(2)给出了基于主题聚类和语义分析的多文档文摘系统(CSA-MDSS)的体系结构。首先通过对局部主题个数的自动探测，采用 K-均值聚类算法对句子进行聚类，形成多文档集合的局部主题；然后在局部主题确定的基础上，采用一个重要性抽取和平均抽取相结合的方法进行文摘句抽取；最后用改进的 MMR-SS 句子冗余消除算法对候选文摘句抽取进行冗余处理、平滑处理和连贯性处理。实验结果表明该结构生成的文摘在信息的平衡性和覆盖率上具有很大的优势。

(3)提出了基于内聚度与文摘句位置相结合的文摘句排序策略，提高了文摘的连贯性和可读性。该策略在统计局部主题间相对位置的基础上，建立它们之间的关系有向图并计算其内聚度；排序过程中每从有向图中输出一个顶点时，便从剩余顶点中查找与其具有最大内聚度的顶点，若该内聚度大于阈值，则将这两个顶点所代表的局部主题文摘句置于摘要中相邻的位置，从而有效地解决了句子的排序问题。

(4)提出了一种多特征融合的句子权重计算和查询文摘生成的方法。在对用户查询输入进行短语识别和网页结构分析的基础上，将关键词短语、网页结构等启发式规则信息融入到句子重要度计算，并分析句子与查询输入的关联特征，使文摘内容能与用

户需求一致，实现文摘信息的内聚性和全面性。对比实验结果表明，此方法生成的查询摘要大大提高了网页相关性判断的速度和准确率。

(5)提出了利用自动文档摘要进行特征选择和文本分类的方法。该方法的特点是文档摘要能够保留原文档中相对重要的特征，去除了大量不利于分类的噪音，大大降低了特征选择和分类的计算复杂度，提高了分类的速度和性能。

由于自动文摘的研究是一个富有挑战性和探索性的课题，许多相关问题的认识以及求解都需要长期研究和不断积累。本论文仅仅在中文自动文摘及应用方面进行了一些初步探索，研究工作非常有限，要丰富和完善这一领域的研究还有大量的工作要做。作者认为应该从以下几个方面继续展开研究工作：

(1) 本文提出的文摘句润色处理是通过对句法树中的节点进行增加、替代、合并、删除等操作实现的。由于这些操作依据规则进行(判断条件和操作体两部分组成)，为此需要为每一种润色处理类型分别制定了相应的规则。但是，目前我们定义的判断条件和操作体还不完善，只能覆盖文摘平滑处理中的小部分语法与句法现象，如何进一步完善文摘润色处理规则的制定是提高文摘质量的一个非常重要的研究课题。

(2) 本文提出的面向用户查询的文摘虽然能够加快用户判断文档是否为有用信息的速度，但是用户对信息的获取仍然离不开对大量返回文档摘要的阅读，这种负担同样会令很多用户难以承受，是否可以根据用户的查询，利用多文档自动文摘技术对多个内容相近的网页文档的内容进行融合，生成较原文档集有大幅度压缩的短文，进一步减轻用户获取信息的压力，是需要深入研究的内容。

(3) 虽然本文对文摘句排序进行了探索，提出了一种基于局部主题内聚度的改进MO 算法，但是我们并没有考虑文档集中蕴含的时间信息。对于来自于不同时间段的多文档集合进行多文档文摘，下一步我们需要在改进 MO 算法的基础上加强时间信息的利用，借助于文档的时间信息解决信息矛盾的处理和进一步提高文摘的可读性。

(4) 多文档文摘目前还是基于句子抽取的文摘，如何打破句子结构生成文摘，一直是自动文摘研究者致力要解决的问题。在文本生成以及文摘句的压缩与合并方面我们将会做进一步的研究工作。

由于作者水平有限，论文中肯定存在不足或不妥之处，真诚希望各位老师和同学批评指正。

参考文献

[1] 郭庆琳. 基于文本聚类和语义分析的自动文摘的研究与实现[D]. 北京: 北京理工大学, 2005.

[2] 李蕾,钟义信. 全信息理论在自动文摘系统中的应用[J], 计算机工程与应用, 2000 (1):4-7.

[3] George Giannakopoulos, Vangelis Karkaletsis, George Vouros, Panagiotis Stamatopoulos. Summarization System Evaluation Revisited: N-gram Graphs. ACM Transactions on Speech and Language Processing[C].2008, 5(3):1-39.

[4] 刘德喜. 基于基本要素的多文档自动文摘研究[D]. 武汉: 武汉大学,2007.

[5] GB6447-86,文摘编写规则[S]. 北京: 中国标准出版社,1986.

[6] H.P.Luhn. the Automatic Creation of Literature Abstracts [J]. IBM Journal of Research and Development, 1958, 2 (2):159-165.

[7] Udo Hahn. Automatic Text Summarization: Methods, Systems, Evaluation [EB/OL]. 2001. http://www.coling.uni-freiburg.de/~hahn.

[8] H.P.Edmundson. New Methods in Automatic Abstracting [J]. Journal of the Association for Computing Machinery, 1969, 16(2):264-285.

[9] 刘开瑛, 郭炳炎. 自然语言处理[M]. 北京: 科学出版社, 1991.

[10] 马希文,李小滨,徐越. 自然语言处理与自动文摘[M]. 北京:电子工业出版社,1990: 99-117.

[11] 姚天顺.自然语言理解—一种让机器懂得人类语言的研究[M].北京: 清华大学出版社, 1995.

[12] 刘挺,王开铸.自动文摘的四种主要方法[J], 情报学报,1999.18(1):10-19.

[13] L.F.Rau, E.S.Jacobs, Uri Zernik. Information and Text Summarization Using and Management [J]. Lingusitic Knowledge Acquisition, Extracting Information Processing, 1998, 25(4): 419– 428.

[14] 宋今, 赵东岩. 基于语料库与层次词典的自动文摘研究[J],软件学报, 2000,11(3):308-314.

[15] G.Salton, J.Allan, C.Buckleym, Amit Singhal. Automatic Analysis, Theme Generation and Summarzation of Machine-Readable Texts [J]. Science, 1994, 264(3):1421-1426.

[16] Kupiec, Julian, Jan O. Pedersen, Francine Chen. A Trainable Document Summarizer [J], Research and Developmentin Information Retrieval, 1995, 68-73.

[17] Dragomir R.Radev, Eduard Hovy, Kathleen McKeown. Introduction to the Special Issue on Summarization [J]. Computational Linguistics, 2002, 28(4): 399-408.

[18] B.Endres-Niggemeyer, E.Neugebauer. Professional Summarizing: No Cognitive Simulation without Observation [J]. Journal of the American Society for Information Science, 1998,

9(6):486-506.

[19] D.R.Radev, K.R.McKeown. Generating Natural Language Summaries from Multiple On-Line Sourse [J]. Computational Linguistics, 1998, 24(3):469-500.

[20] Lin, C.E.Hovy. Identifying Topics by Position. In Proceedings of the 5th Conference on Applied Natural Language Processing, Association for Computational Linguistics[C]. 1997, 283-290.

[21] Jaime Carbonell, Jade Goldstein. The Use of MMR, Diversity-Base Reranking for Reordering Documents and Producing Summaries. In Proceedings of ACM SIGIR'98[C]. 1998: 335-336.

[22] Yi-hong Gong, Xin Liu. Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis. In Proceedings of ACM Special Interest Group on Information Retrieval [C]. 2001:19-25.

[23] Conroy, John. Dianne Oleary. Text Summarization via Hidden Markov Models. In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval[C]. 2001: 406-407.

[24] Inderjeet Mani. Automatic Summarization [M]. Amsterdam/Philadephia, John Benjamins Publishing Company, 2002.

[25] 史磊. 中英文自动文摘系统及其若干相关技术研究[D]. 上海: 上海交通大学, 2000.

[26] 王永成, 许慧敏.OA 中文文献自动摘要系统[J],情报学报,1997,16(2):128-132.

[27] 刘挺,王开铸.基于篇章多级依存结构的自动文摘研究[J],计算机研究与发展, 1999, 36(4):479-488.

[28] 刘挺, 吴岩, 王开铸.基于信息抽取和文本生成的自动文摘系统设计[J],情报学报, 1997, 16(增刊):24-29.

[29] 吴立德. 大规模中文文本处理[M]. 上海: 复旦大学出版社,1997.

[30] 郑义,黄萱菁,吴立德.文本自动综述系统的研究与实现[J],计算机研究与发展, 2003, 40(11):1606-1611.

[31] 薛翠芳,李晓黎,郭炳炎. 汉语文摘系统中文本结构的自动分析.全国第四届计算语言学联合学术会议论文集[C].北京: 清华大学出版社,1997:338-344.

[32] 薛翠芳,郭炳炎.中文自动文摘系统. 第五届全国 Artificial Intelligence 联合会会议论文集[C]. 西安:西安交通大学,1998:200-206.

[33] 杨晓兰,钟义信.基于全信息词典的自动文摘系统研究与实现[J],情报学报, 1997, 16(6):408-414.

[34] 李蕾,钟义信,郭祥昊.面向特定领域的理解型中文自动文摘系统[J],计算机研究与发展, 2000,37(4):493-497.

[35] 胡舜耕,刘晓宇,钟义信.基于多 Agent 技术的自动文摘系统的研究和设计[J],电子学报, 2001,29(2):247-249.

[36] 万敏,罗振声,季垣,等.基于概念统计的英文自动文摘研究[J],计算机工程与应用, 2002,24:7-16.

[37] 季垣,罗振声,万敏,等.基于概念统计和语义层次分析的英文自动文摘研究[J],中文信息学报,2003,17(2):14-20.

[38] 王萌,何婷婷,姬东鸿.基于 HowNet 概念获取的中文自动文摘系统[J],中文信息学报, 2005,19(3):87-93.

[39] 徐永东, 徐志明,王晓龙. 基于信息融合的多文档自动文摘技术[J], 计算机学报, 2007, 30(11):2048-2054.

[40] 赖茂生, 王知津. 文摘的概念与方法[M].北京:书目文献出版社,1991:4-224.

[41] 刘挺. 基于篇章多级依存结构的自动文摘研究[D]. 哈尔滨: 哈尔滨工业大学,1998.

[42] 孙春葵.自动文摘及其知识获取技术研究[D]. 北京: 北京邮电大学,2000.

[43] Udo Hahn, Inderjeet Mani. The Challenges of Automatic Summarization [J]. Information Retrieval, 2000: 29-36.

[44] 王志琪, 王永成, 刘传汉. 论自动文摘及其分类[J], 情报学报, 2005,24(2): 214-221.

[45] R.Brandow, K. Mitze, L. F. Rau. Automatic Condensation of Electronic Publications by Sentence Selection [J]. Information Processing & Management, 1995, 31(5): 675-685.

[46] H.P.Edmundson, R.E.Wyllys. Automatic Abstracting and Indexing—Survey and Recommendations [J]. Communications of the ACM, 1961, 4(5): 226-234.

[47] H.P.Edmundson. Automatic Abstracting, TRW Computer Division, Thompson Ram Wooldridge, Inc., Canoga Park, California, AD 406 155, 1963.

[48] M.Mitra, Amit Singhal, Chris Buckley. Automatic Text Summarization by Paragraph Extraction. In Proceedings of ACL/EACL-1997 Workshop on Intelligent Scalable Text Summarization[C]. Madrid, Spain, July 1997: 31-36.

[49] J. Hutchins. Summarization: Some Problems and Methods. In Proceedings of INFORMATICS 9: Meaning—the Frontier of Informatics[C]. Cambridge, UK, 1987:151-173.

[50] T.A. van Dijk. Semantic Macro-Structures and Knowledge Frames in Discourse Comprehension. M.A. Just and P.A. Carpenter (eds.): Cognitive Processes in Comprehension, 1977:3-32.

[51] U. Hahn, U. Reimer. Knowledge-Based Text Summarization: Salience and Generalization Operators for Knowledge-Based Abstraction, Advances in Automatic Text Summarization, I. Mani and M. Maybury, eds., MIT Press, Cambridge, Mass.1999: 215-232.

[52] D. Fum, G. Guida , C. Tasso. Forward and Backward Reasoning in Automatic Abstracting. In Proceedings of COLING 1982[C]. 1982: 83-88.

[53] R. Kuhlen. Some Similarities and Differences between Intellectual and Macine Text Understanding for the Purpose of Abstracting. In Proceedings of the Fifth International Research Forum in Information Science[C]. 1984: 87-109.

[54] T. Strzalkowski, J. Wang, B. Wise. A Robust Practical Text Summarization System. In AAAI Intelligent Text Summarization Workshop[C]. Stanford, CA, Mar 1998:26-30.

[55] Dragomir R. Radev, Vasilis Hatzivassiloglou, Kathleen McKeown. A Description of the CIDR System as Used for TDT-2. In Proceedings of the DARPA Broadcast News Workshop[C]. Herndon, Virginia. 1999.

[56] Dragomir Radev, Hongyan Jing, Malgorzata Budzikowska. Centroid-Based Summarization of Multiple Documents: Sentence Extraction, Utility-Based Evaluation and User Studies [J]. Information Processing and Management, 2004(40): 919–938.

[57] Patrick Pantel, Dekang Lin. Document Clustering with Committees. In Proceedings of ACM (SIGIR'02) [C]. New York: ACM, 2002: 199-206.

[58] J.G.Carbonell, J.Goldstein. The Use of MMR, Diversity-based Reranking for Reordering Documents and Producing Summaries. In Proceedings of the 21th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval [C]. Melbourne, Australia, August 24-28, 1998: 335-336.

[59] Endre Boros, Paul. Kantor, David J. Neu. A Clustering Based Approach to Creating Multi-Document Summaries. In Proceedings of Document Understanding Conference DUC 2001[C]. 2001.

[60] Jianhui Wang, Shuigeng Zhou, Yunfa Hu. Sentences Clustering Based Automatic Summarization. In Proceedings of the Second International Conference on Machine Learning and Cybernetics[C]. Xi'an, 2003, 57-62.

[61] Ricardo Baeza-Yates, Berthier Ribeiro-Neto. Modern Information Retrieval [M]. ACM Press, 1999:27-30.

[62] Po Hu, Tingting He, Donghong Ji, et al. A Study of Chinese Text Summarization Using Adaptive Clustering of Paragraphs. In Proceedings of the 4th International Conference on Computer and Information Technology (CIT'04) [C]. Wuhan, 2004, 1159-1164.

[63] S.T.K. Tang, Yen Jerome, C. C. Yang. Multidocument Summarization Based on Concept Space. In Proceedings of Information Technology: Research and Education[C]. 2003, 385-389.

[64] E Hovy, C Y Lin, L Zhou. Evaluating DUC 2005 Using Basic Elements. In Proceedings of Document Understanding Conderence (DUC-2005) [C]. 2005.

[65] Dexi Liu, Yanxiang He, Donghong Ji, et al. Multi-document Summarization Based on BE-Vector Clustering. In Proceedings of 10th International Conference on Intelligent Text Processing and Computational Linguistics [C]. 2006: 470-479.

[66] 王建会,申展,胡运发. 一种实用高效的聚类算法[J], 软件学报, 2004, 15(5): 697-705.

[67] Regina Barzilay, Kathleen R. McKeown, Elhadad Michael. Information Fusion in the Context of Multi-Document Summarization. In Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics[C]. New Jersey: Association for Computational Linguistics, 1999: 550-557.

[68] Rie Kubota Ando, Branimir K. Boguraev, Roy J. Byrd, et al. Multi-document Summarization by Visualizing Topical Content. In Proceedings of the ANLP-NAACL 2000 (Workshop on Automatic Summarization) [C]. Advanced Summerization Workshop, Seattle, WA, 2000:12-19.

[69] Jaoua Kallel Fatma, JAOUA Maher. Summarization at LARIS Laboratory. In Proceedings of the 5th Document Understanding Conference (DUC2004) [C]. 2004.

[70] Junichi FUKUMOTO, Tomoya SUGIMURA. Multi-document Summarization Using Document Set Type Classification. Working Notes of NTCIR-4, Tokyo, National Institute of Informatics, 2004.

[71] M. White, T. Korelsky, C. Cardie, et al. Multidocument Summarization via Information Extraction. In Proceedings of Human Language Technology Conf. HLT 2001 [C]. San Diego, CA, 2001.

[72] G. Salton. Automatic Text Structuring and Summarization [J]. Information Processing & Management, 1997, 33(2): 193-207.

[73] C. D. Paice. Constructing Literature Abstracts by Computer: Techniques and Prospects [J]. Information Processing & Management, 1990: 171-186.

[74] LIN Chin-yew, CAO Gui-hong, GAO Jian-feng. An Information-Theoretic Approach to

Automatic Evaluation of Summaries. In Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL[C]. New York.2006:463-470.

[75] K. S. Jones. Automatic Summarizing Factors and Directions Advances in Automatic Text Summarization [M]. Cambridge MA: MIT Press. 1998.

[76] Inderjeet Mani, Summarization Evaluation: An Overview. In Proceedings of the NTCIR Workshop, Meeting on Evaluation of Chinese and Japanese Text Retrieval and Text Summarization[C]. 2001.

[77] I. Mani, T. Firmin, B. Sundheim. The TIPSTER SUMMAC Text Summarization Evaluation. In Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics[C]. Bergen. 1999: 77-85.

[78] 沈洲,王永成,许一震. 自动文摘系统评价方法的研究与实践[J], 情报学报, 2001, 20(1): 66-72.

[79] R. E.Johnson. Recall of Prose as a Function of Structural Importance of Linguistic Units [J]. Journal of Verbal Learning and Verbal Behavior, 1970 (9): 12-20.

[80] Jianhui Wang, Shuigeng Zhou, Yunfa Hu. Sentences Clustering Based Automatic Summarization. In Proceedings of the Second International Conference on Machine Learning and Cybernetics[C]. Xi'an, 2003: 57-62.

[81] S. Miike, E. Itoh., K. Ono. A Full-Text Retrieval System with Dynamic Abstract Generation Function. In Proceeding of SigIR[C]. 1994:152-161.

[82] D.R.Radev, H.Jing, M.Budzikowska. Summarization of Multiple Documents: Clustering, Sentence Extraction and Evaluation. In Proceedings of the Workshop on Automatic Summarization[C]. New Brunswick, New Jersey: Association for Computational Linguistics, 2000: 21-30.

[83] R.L.Donaway, K.W.Drummey, L. A.Landauer, et al. Indexing by Latent Semantic Analysis [J]. Jounal of the American Society for Information Science, 1990, 41(6):391-407.

[84] C.Y Lin. Summary Evaluation Environment [OL]. http://www.isi.edu/~cyl/SEE.

[85] K.Papineni, S.Roukos, T.Ward, et al. BLEU: a Method for Automation Summarization of Machine Translation [R].IBM Research Report RC22176 (W0109-022), 2001.

[86] C.Y.Lin, E.Hovy. Automatic Evaluation of Summarization Using N-gram Co-Occurrence Statistics. In Proceedings of the Human Technology Conference[C]. Edmonton, Canada, 2003.

[87] T.R.Cormen, C.E.Leiserson, R.L.Rivest. Introduction to Algorithms [M]. The MIT Press, 1989.

[88] 索红光. 中文 WEB 文档关键词抽取与自动文摘研究[D]. 北京: 北京理工大学, 2007.

[89] 丁春. 关键词标引的若干问题探讨[J],编辑学报, 2004, 16 (2) : 105 -106.

[90] Turney P.D. Learning to Extract Keyphrases from Text[R]. NRC Technical Report ERB-1057, National Research Council, Canada, 1999: 1-43.

[91] Witten I.H., Paynter G.W., Frank E., et al. KEA: Practical Automatic Keyphrase Extraction. In Proceedings of the 4th ACM Conference on Digital Libraries [C]. Berkeley, California, US, 1999: 254-256.

[92] Hulth. An Improved Automatic Keyword Extraction Given More Linguistic Knowledge. In Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing[C]. 2003: 216-223.

[93] Yang Wenfeng. Chinese Keyword Extraction based on Max-duplicated Strings of the Documents. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval[C]. Tampere, Finland, 2002: 439-440.

[94] 李素建, 王厚峰, 俞士汶. 关键词自动标引的最大熵模型应用研究[J], 计算机学报, 2004, 27 (9) : 1192-1197.

[95] 王军. 词表的自动丰富——从元数据中提取关键词及其定位[J],中文信息学报, 2005, 19 (6): 36-43.

[96] Karen Sparck Jones.What might be in Summary? [J]. Information Retrieval, 1993.

[97] Kathleen McKeown, Dragomir Radev. Genersting Summaries of Multiple New Articles .In Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval [C]. 1995: 74-82.

[98] Michael Hasan, Ruqaiya Halliday. Cohesion in English [M].Longman, London, 1976.

[99] M.Hoey. Patterns of Lexis in Text [M]. Oxford University Press, Oxford, 1991.

[100] J Morris, G Hirst. Lexical Cohesion Computed by Thesaural Relations as an Indicator of the Structure of Text [J].Computational Linguistics, 1991, 17(1):21-48.

[101] Mark A.Stairmand. A Computational Analysis of Lexical Cohesion with Applications in Information Retrieval [D]. PhD thesis, Center for Computational Linguistics, UMIST, Manchester, 1996.

[102] Graeme Hirst, David St-Onge. Lexical Chains as Representation of Context for the Detection and Correction of Malapropisms [M].In Christiane Fellbaum, Editor, WordNet: An Electronic Lexical Database and Some of its Applications.Cambridge, MA: The MIT Press, 1997, Chapter 13,

305-332.

[103] 孔翔勇. 基于《知网》的汉语词相似度计算[D]. 哈尔滨: 哈尔滨工业大学,2002.

[104] 姚建民, 周明, 赵铁军, 李生. 基于句子相似度的机器翻译评价方法及其有效性分析[J], 计算机研究与发展, 2004, 41 (7): 1258-1265.

[105] 颜伟, 荀恩东. 基于 WordNet 的英语词语相似度计算[EB/OL], 2004-12, http://lib.blcu.edu.cn/per/scbar/pdf/wordnetsem.pdf.

[106] Yi Guan, Xiaolong Wang, Xiangyong Kong. Quantifying Semantic Similarity of Chinese Words from HowNet .In Proceedings of 2002 International Conference on Machine Learning and Cybernetics [C]. 2002: 234-239.

[107] 刘群, 李素建. 基于《知网》的词汇语义相似度计算. 第三届中文词汇语义学研讨会[C]. 中国台北, 2002.

[108] 王灿辉, 张敏,马少平,等.基于相邻词的中文关键词自动抽取[J],广西师范大学学报: 自然科学版, 2007,25(2):161-164.

[109] 尤文建, 李绍滋, 李堂秋. 基于词汇链的文本过滤模型[J],计算机应用研究, 2003, 20(9): 32-35.

[110] 王继成, 武港山. 一种篇章结构指导的中文 WEB 文档自动摘要方法[J]. 计算机研究与发展, 2003，40(3).

[111] 秦兵. 基于子主题的多文档文摘技术的研究[D]. 哈尔滨: 哈尔滨工业大学, 2005.

[112] Addrew R. Webb. Statistical Pattern Recognition, 2nd edn. John Wiley &Sons, 2002: 376–379.

[113] 刘寒磊. 中文多文档自动文摘中若干重要技术的研究[D]. 哈尔滨: 哈尔滨工业大学,2005.

[114] 张冬茉, 李锦乾. 汉语自然语言生成的句子结构优化[J],计算机工程, 1998,24(7):14-17.

[115] 李锦乾, 张冬茉.自然语言生成中的句子结构优化处理[J],计算机应用研究, 1998, 14(1): 53-55.

[116] 何文忠. 谈谈扩展句[J],益阳师专学报,2000,21(1):84-85.

[117] Dalianis H, Hovy E. On Lexical Aggregation and Ordering. In Demonstrations and Posters of the 8th International Workshop on Natural Language Generation[C]. 1996, 1(1):6-16.

[118] N. Okazaki, Y. Matsuo, M. Ishizuka. Coherent Arrangement of Sentences Extracted from Multiple Newspaper Articles. In Proceedings of Trends in Artificial Intelligence PRICAI 2004 [C]. 2004, 882-891.

[119] R. Barzilay, E. Elhadad, K. McKeown. Inferring Strategies for Sentence Ordering in

Multidocument Summarization [J]. Journal of Artificial Intelligence Research, 2002, 17: 35-55.

[120] M. Lapata. Probabilistic Text Structuring: Experiments with Sentence Ordering. In Proceedings of the 41st Meeting of the Association of Computational Linguistics[C]. 2003, 545-552.

[121] John Conroy, Judith Schlesinger, Jade Goldstein, Dianne O'Leary. Left-Brain/Right-brain Multi-document Summarization. In Proceedings of the 4th Document Understanding Conference (DUC'04) [C]. 2004.

[122] R. Barzilay, L. LEE. Catching the Drift: Probabilistic Content Models, with Applications to Generation and Summarization. In Proceedings of HLT-NAACL 2004[C]. 2004: 113-120.

[123] Danushka Bollegala, Naoaki Okazaki, Mitsuru Ishizuka. A Bottom-up Approach to Sentence Ordering for Multi-document Summarization. In Proceedings of the 21th International Conference on Computational Linguistics and 44th Annual Meeting of the ACL[C]. 2006: 385-392.

[124] Barzilay R, Elhadad E, McKeown K. Sentence Ordering in Multi-Document Summarization. In Proceedings of the 1th Human Language Technology Conference[C]. San Diego, CA, 2001: 149-156.

[125] N. Okazaki, Y. Matsuo, M. Ishizuka. Improving Chronological Ordering of Sentences Extracted from Multiple Newspaper Articles. ACM Transactions on Asian Language Information Processing[C]. 2005, 4(3): 321-339.

[126] 邵伟, 何婷婷, 胡珀, 肖华松.一种面向查询的多文档文摘句选择策略.内容计算的研究与应用前沿——第九届全国计算语言学学术会议论文集[C]. 2007:637-642.

[127] Wauter Bosma. Query-Based Summarization using Rhetorical Structure Theory. In Proceedings of CLIN04[C]. 2004.

[128] R.W.White, I.Ruthven and J.M.Jose. Finding Relevant Documents Using Top Ranking Sentences: An Evaluation of Two Alternative Schemes. In Proceedings of Special Interest Group on Information Retrieval (SIGIR2002) [C]. 2002.

[129] J.Goldstein, M.Kantrowitz, V.Mittal, J.Carbonell. Summarizing Text Documents: Sentence Selection and Evaluation Metrics. In Proceedings of Special Interest Group on Information Retrieval [C]. 1999.

[130] 林鸿飞,杨志豪, 赵晶. 基于段落匹配和分布密度的偏重摘要实现机制[J],中文信息学报, 2007, 21(1):43-48.

[131] Sujian Li, You Ouyang, Bin Sun. Peking University at DUC2006 .In Proceeding of DUC2006[C].

2006.

[132] A.Tombros, M. Sanderson. Advantages of Query Biased Summaries in Information Retrieval. In Proceedings of 21th Annual International ACM Special Interest Group on Information Retrieval Conference on Research and Development in Information Retrieval[C]. 1998: 2-10.

[133] M.Sanderson. Accurate User Directed Summarization from Existing Tools. In Proceedings of the 7th International Conference on Information and Knowledge Management (CIKM 98) [C]. 1998: 45-51.

[134] 索红光,刘玉树,曹淑英.一种基于词汇链的关键词抽取方法[J],中文信息学报, 2006, 20(6):25-30.

[135] H.Alam, A.Kumar, M.Nakamura, et al. Structured and Unstructured Document Summarization: Design of a Commercial Summarizer using Lexical Chains. In Proceedings of 7th International Conference on Document Analysis and Recognition [C]. IEEE Computer Society, 2003: 1147-1150.

[136] H.Mochizuki, M.Iwayama, M. Okumura. Passage-level Document Retrieval using Lexical Chains [J]. Journal of Natural Language Processing, 2000, 6(3):101-126.

[137] H.Mochizuki, M.Okumura. A Comparison of Summarization Methods based on Task-based Evaluation. In Proceedings of 2th International Conference on Language Resources and Evaluation[C]. LREC-2000, Athens, Greece.

[138] O Buyukkokten, H Garcia-Molina.A Packet Accordion Summarization for End-game browsing on PDAs and Cellular Phones. In Proceedings of ACM Conference on Human Factors in Computing Systems (CHI 2001) [C]. New York: ACM Press, 2001:213-220.

[139] S Gupta, G Kaiser, D Neistadt. DOM-based Content Extraction of HTML Documents. In Proceedings of the 12th International World Wide Web Conference[C]. New York: ACM Press, 2003: 207-214.

[140] 张志刚, 陈静, 李晓明.一种 HTML 网页净化方法[J],情报学报, 2004,23(4):387-392.

[141] 常育红,姜哲,朱小燕.基于标记树表示方法的页面结构[J],计算机工程与应用, 2004(16):130-132.

[142] 荆涛, 左万利. 基于可视布局信息的网页噪音去除算法[J],华南理工大学学报:自然科学版, 2004,32(增刊):85-86.

[143] 孙承杰, 关毅. 基于统计的网页正文信息抽取方法的研究[J],中文信息学报,2004, 18(5):1-6.

[144] 王琦,唐世渭,杨冬青.基于 DOM 的网页主题信息自动抽取[J], 计算机研究与发展, 2004,41(10):1786-1792.

[145] 李效东, 顾柳清. 基于 DOM 的 Web 信息提取[J],计算机学报,2002,25(5):528-529.

[146] 吕津, 赵明生. 对因特网上自动信息提取技术的研究[J],数据通信, 2000:1-2.

[147] Line Eikvil. Information Extraction from World Wide Web [M]. Norwegian Computing Center. 1999,3-4.

[148] Yohei Seki. Sentence Extraction by TF/IDF and Position Weighting from Newspaper Articles. In Proceedings of the Third NTCIR Workshop on Research in Information Retrieval, Automatic Text Summarization and Question Answering[C]. Tokyo, 2002: 55-59.

[149] 蒋效宇, 樊孝忠, 陈康. 基于用户查询的中文自动文摘研究[J],计算机工程与应用, 2008, 44(5):48-50.

[150] 高升,贾文举,王晓龙.一个基于互信息的规则量化方法[J],计算机研究与发展, 2000(8): 984-989.

[151] H.Alam, A.Kumar, M.Nakamura. Structured and Unstructured Document Summarization: Design of a Commercial Summarizer using Lexical Chains. In Proceedings of 7th International Conference on Document Analysis and Recognition [C]. IEEE Computer Society, 2003: 1147-1150.

[152] FABRIZIO SEBASTIANI. Machine Learning in Automated Text Categorization [J]. ACM Computing Surveys, 2002(34):1-47.

[153] David. D. Lewis. An Evaluation of Phrasal and Clustered Representations on a Text Categorization Task. In Proceedings of the 15th Annual International ACMSIGIR Conference on Research and Development in Information Retrieval[C]. Copenhagen,Denmark, 1992: 37-50

[154] W. B. Cavnar. N-Gram-Based Text Filtering For TREC-2. In Proceedings of the Second Text Retrieval Conference[C]. Gaithersburg, Maryland, 1993.

[155] M. F. Caropreso, S. Matwin, F. Sebastiani. A Learner-Independent Evaluation of the Usefulness of Statistical Phrases for Automated Text Categorization. In Proceedings of Text Databases and Document Management: Theory and Practice[C]. Hershey, US, 2001: 78-102.

[156] J. Furnkranz, T. Mitchell, E. Riloff. A Case Study in Using Linguistic Phrases for Text Categorization on the WWW. In AAAI/ICML Workshop on Learning for Text Categorization[C]. Madison, WI, 1998: 5-12.

[157] Cui Yu,Bin Cui,PShuguang Wang, et al.Efficient Index-based KNN Join Processing for High-dimensional Data[J].Information and Software Technology. 2007, 49(4):332-344.

[158] Fabrizio Sebastiani. Machine Learning in Automated Text Categorization [J].ACM ComPuting Surveys.2002, 34(1):1-47.

[159] PAleksander Kolcz, PWen-tau Yih.Raising the Baseline for High-precision Text Classifiers. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining[C]. Aug, 2007: 400-409.

[160] Hillol Kargupta, Byung-Hoon Park.A Fourier Spectrum-Based Approach to Represent Decision Trees for Mining Data Streams in Mobile Environments [J]. IEEE Transactions on Knowledge and Data Engineering. Feb, 2004, 16(2):216-229.

[161] Tatjana Eitrich, PBruno Lang.On the Optimal Working Set Size in Serial and Parallel Support Vector Machine Learning with the Decomposition Algorithm .In Proceedings of the 5th Australasian Conference on Data Mining and Analystics[C].Nov 2006: 121-128.

[162] Chen, K.H., Chen, H.H. A Corpus-Based Approach to Text Partition. In Proceedings of the Workshop of Recent Advances in Natural Language Processing [C]. Sofia, Bulgaria. 1995: 152-161.

[163] Yang Ymiing, Jan O. Pedersen. A Comparative Study on Feature Selection in Text Categorization. In Proceedings of 14th International Conference on Machine Learning[C]. 1997: 412-420.

[164] Apte C, Damerau F.J, Weiss S.M. Automated Learning of Decision Rules for Text Categorization. In Proceedings of ACM Transactions on Information Systems[C]. 1994, 12 (3): 233-251.

[165] Dumais S.T, Plat J, Heckerman D, et al. Inductive Learning Algorithms and Representations for Text Categorization [J]. Technical Report, Microsoft Research, 1998.

[166] Sue J.Ker, Jen-Nan Chen. A Text Categorization on Summarization Technique. In Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics IR&NLP Workshop[C]. Hong Kong, China: Association for Computational Linguistics, 2000: 79-83.

[167] 唐慧丰,谭松波, 程学旗. 基于监督学习的中文情感分类技术比较研究[J],中文信息学报, 2007, 21(6):88-95.

[168] 胡熠, 陆汝占, 李学宁. 基于语言建模的文本情感分类研究[J], 计算机研究与发展, 2007, 44 (9) :1469-1475.

[169] 李钝,曹付元,曹元大. 基于短语模式的文本情感分类研究[J],计算机科学, 2008, 35(4):

132-134.

[170] 秦兵,刘挺,李生. 基于局部主题判定与抽取的多文档文摘技术[J], 自动化学报, 2004, 30(6): 906-910.

[171] Yanxing He, Dexi Liu, Donghong Ji, et al. A Multi-Document Summarization System Based on Genetic Algorithm. In Proceedings 2006 International Conference on Machine Learning and Cybernetics[C]. Dalian, China, Aug. 2006, 7(4): 2659-2664.

攻读学位期间发表论文与研究成果清单

发表的论文：

1. Xiao-Yu Jiang, Xiao-zhong Fan, Zhi-fei Wang, Ke-liang Jia. Improve the Performance of Web Pages Relevance Judgment using Query-biased Summary [J]. Journal of Computational Information Systems, 2009(5).(EI Compendex)
2. 蒋效宇, 樊孝忠, 陈康. 用于多文档文摘句排序的改进 MO 算法[J]. 华南理工大学学报:自然科学版, 2008,36(9):43-47.(EI Compendex: 084511687163)
3. Xiao-Yu Jiang, Xiao-zhong Fan, Kang Chen. Chinese Text Classification based on Summarization Technique. In Proceedings of the Third International Conference on Semantics, Knowledge and Grid (SKG2007) [C]. Xi'an, China, Oct. 2007: 362-365. (EI Compendex: 083511496833)
4. Xiao-Yu Jiang，Xiao-zhong Fan, Kang Chen, Jie Liu. Applied Research of Query-biased Summary in the Chinese Web Pages Relevance Judgment.In Proceedings of 2008 International Conference on Machine Learning and Cybernetics (ICMLC2008) [C].Kunming, China, July 2008, 7(3):1604-1609. (EI Compendex: 085211811598)
5. Xiao-Yu Jiang. A Keyword Extraction Method based on Lexical Chains .In Proceedings of Third International Symposium on Intelligence Computation and Applications (ISICA 2008) [C]. Lecture Notes in Computer Science, v5370LNCS Springer, Advances in Computation and Intelligence, Wuhan, China, December 2008:360-367. (EI Compendex : 090511883897)
6. Xiao-Yu Jiang, Xiao-zhong Fan, Zhi-fei Wang, Ke-liang Jia. Improving the Performance of Text Categorization using Automatic Summarization. In Proceedings of 2009 International Conference on Computer Modeling and Simulation (ICCMS 2009) [C]. Macau, China, Feb.2009:347-351.(EI Compendex: 20091712052586)
7. Xiao-Yu Jiang .Chinese Automatic Text Summarization based on Keyword Extraction. In Proceedings of 2009 International IEEE Workshop on Database Technology and Applications (DBTA2009) [C]. Wuhan, China, April 2009:225-228. (EI Compendex)
8. 蒋效宇, 樊孝忠, 陈康. 基于用户查询的中文自动文摘研究[J].计算机工程与应用, 2008,44(5):48-50.(中文核心)
9. 蒋效宇, 樊孝忠, 陈康. 自动文摘在网页分类中的应用研究[J].计算应用研究. 2008 年第 08 期.

(中文核心)

10. Ke-liang Jia, Xiao-zhong Fan, Xiao-Yu Jiang.Research of Chinese Question Answering System based on Clustering [J]. Journal of Computational Information Systems, 2008, 4(3): 875-882.(EI Compendex: 083211445077)
11. Jie Liu, Xiao-zhong Fan, Xiao-Yu Jiang. Research on Chinese Restricted Domain Question Answering System.In Proceedings of 2008 International Conference on Machine Learning and Cybernetics (ICMLC2008) [C]. Kunming, China, July 2008, 7(3):2585-2590. (EI Compendex: 085211817238)

参与的科研课题：

1. 国家教育部高等学校博士学科点专项科研基金(受限领域自动问答系统研究，项目编号：20050007023)
2. 北京理工大学自然语言处理实验室与扬州万方电子技术有限公司合作项目“特定领域网络信息提取服务系统”
3. 北京理工大学自然语言处理实验室与山西省太原软件园合作项目“银行领域问答系统”
4. 北京市教委科技面上项目“信息抽取技术在服装领域的应用研究”

致　谢

值此论文完成之际，向我的导师樊孝忠教授表示衷心的感谢，博士期间的每一个跋涉的脚印，无不得益于恩师的指引。樊老师在我攻读博士学位的过程中给予了全面指导和深切关怀，为本文的完成更是倾注了大量的心血。恩师创造的和谐、民主、自由的学术气氛极大地激发了我的创造力；恩师渊博的知识、严谨的治学态度、非凡的科学洞察力和勤奋的工作作风深深地感染着我，将使我终生受益。樊老师诲人不倦的精神和杰出的学者风范将鼓舞我在今后的学习科研中不断攀登新的高度。我以导师为骄傲，深深地向恩师说一声谢谢。

感谢武汉大学的刘德喜博士、哈尔滨工业大学的秦兵教授和北京理工大学的索红光博士公开发表的重要研究成果对本人在研究方向的选择和研究方法的正确引导的帮助。

感谢已经毕业的陈康、邓擘、尹继豪、于江德、余正涛、顾益军、贾可亮和郭庆琳等博士，在我攻读博士学位的三年时间里，在和他们一起学习和讨论中，学到了许多知识，为我的课题研究奠定了基础，向他们致以最诚挚的谢意！

感谢实验室的刘杰、许进忠、傅继彬、庞文博、朱俭、王函石、陈岳、齐全、王知非、毛金涛、周怀南、王金帅、陈晓阳、刘里、刘小明和申艳超等同学，大家在一起相处和讨论的日子里，让我获益不少，我为在这样一个团结向上的集体中工作学习而骄傲！

感谢北京服装学院商学院的宁俊教授、殷文生书记、牛继舜教授、单红忠副教授、赵乃东博士、冯复平老师、龙琼老师、邵丽娜老师、谢杰红老师、张红玉老师等同事在学习、科研和工作中给予的热情帮助。

感谢我敬爱的父亲、伯父、姑姑和姑父，是你们多年来含辛茹苦养育了我，殷切希望鞭策着我，从未改变的鼓励我和支持我，是你们默默无私的奉献给予了我莫大的关怀、爱护、理解与开导。感谢岳父、岳母经济上的支持，生活上的关心，学业上的督导。感谢我的妻子方君，我博士学业中的每一点成绩都离不开你的理解、支持和奉献。

感谢将为我评审论文的各位专家和将出席我论文答辩会的各位委员，你们提出的宝贵意见和建议，将使我受益匪浅。

值此论文结束之际，再次向关心、鼓励、帮助过我的所有老师、同学、亲人和朋友们表示由衷的感谢！

作者简介

蒋效宇，男，汉族，1979 年 1 月 24 日生，江苏省盐城市人

教育经历：

1997.9-2001.7	南京信息工程大学信息工程专业	工学学士
2001.9-2004.4	北京理工大学计算机软件与理论专业	工学硕士
2006.9-至今	北京理工大学计算机应用技术专业	在职博士

工作经历：

2002.9-2003.4	北京美髯公科技有限公司
2003.5-2004.3	北京炎黄信息科技有限公司
2004.4-至今	北京服装学院商学院信息管理教研室

研究方向：自然语言处理，多文档自动文摘，信息检索

主要研究工作及成果：

2006-2007	参与了北京理工大学自然语言处理实验室与扬州万方电子技术有限公司合作项目“特定领域网络信息提取服务系统”
2006-2008	主要参加了国家教育部高等学校博士学科点专项科研基金（受限领域自动问答系统研究，项目编号：20050007023）
2009-至今	主持了北京市教委科技面上项目“信息抽取技术在服装领域的应用研究”
2006-至今	作为第一作者，在国内外期刊会议发表多篇文章，其中核心期刊 4 篇，会议论文 6 篇，EI Compendex 检索 7 篇。