学位论文使用授权书

根据《中央民族大学关于研究生学位论文收藏和利用管理办法》，我校的博士、硕士学位获得者均须向中央民族大学提交本人的学位论文纸质本及相应电子版。

本人完全了解中央民族大学有关研究生学位论文收藏和利用的管理规定。中央民族大学拥有在《著作权法》规定范围内的学位论文使用权，即：(1)学位获得者必须按规定提交学位论文(包括纸质印刷本及电子版)；(2)为教学和科研目的，学校可以将公开的学位论文作为资料在图书馆等场所提供校内师生阅读等服务；(3)根据教育部有关规定，中央民族大学向教育部指定单位提交公开的学位论文；(4)学位论文作者授权学校向中国科技信息研究所及其万方数据电子出版社和中国学术期刊(光盘)电子出版社提交规定范围的学位论文及其电子版并收入相应学位论文数据库，通过其相关网站对外进行信息服务。同时本人保留在其他媒体发表论文的权利。

本人承诺：本人的学位论文是在中央民族大学学习期间创作完成的作品，并已通过论文答辩；提交的学位论文电子版与纸质本论文的内容一致，如因不同造成不良后果由本人自负。

本人同意遵守上述规定。

（保密的学位论文在解密后适用本授权书，本论文：☑不保密，☐保密期限至　　年　　月止）

作者暨授权人签字：王笑琨

20 13 年 5 月 30 日

学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下进行研究工作所取得的研究成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签字：王笑琨

20 13 年 5 月 30 日

摘 要

移动互联网的迅速增长使得搜索引擎面临巨大的挑战，搜索引擎如何适应这种变化以及如何提供更优质的检索服务成为了一个备受关注的问题，作为其重要组成部分的网络爬虫算法成为人们研究的热点。通用网络爬虫由于爬行的规模较大，爬行页面内容比较杂乱，不能满足用户对于特定信息以及兴趣主题的集中爬行。面向主题的网络爬虫可以有选择的爬行与主题相关的网页，有效的减少了爬行页面的数量，而且提高了抓取的准确度并满足了用户对特定主题的搜索需求。

形式概念分析是一种基于概念格的数据分析方法，自从形式概念分析理论提出以来，它就因为知识表示的直观、简洁等特点受到研究者的广泛关注，已经在软件工程、图书馆和信息科学、数据挖掘等诸多领域得到了广泛的应用。

本文通过研究现有主题爬虫的原理，提出了将形式概念分析这一数据分析工具应用到主题爬虫的有关算法中，将概念格应用到主题相关性分析以及排序算法，从而改进了爬虫的相关算法。本文的研究工作主要有：

首先，本文通过对形式概念分析理论的学习，认真研究了其核心概念格上概念间的关系以及概念格的结构，联想到将概念格融入到主题爬虫的算法中。

其次，重点研究了主题爬虫的原理，包括对其结构，搜索策略，pagerank排序算法和主题相关度的研究，改进了基于概念格的主题相关度算法并将其

用来计算爬虫的主题相关度。分析了 pagerank 排序算法的缺陷，并在此基础上结合概念格提出了改进的 pagerank 算法。

关键词: 形式概念分析; 概念格; 主题相关度; pagerank

ABSTRACT

The rapid growth of mobile Internet makes search engine facing the enormous challenge. How the search engine adapts to this change and provides better retrieval service has become a major concern. As an important part of it, the web crawler algorithm is becoming the research focus. General web crawler cannot satisfy the users for specific information and interest topic crawl, due to the large scale and the disorderly content of the web page. Topic web crawler can selectively crawl the web page relevant to the theme, effectively reducing the number of pages in the crawl and improving the accuracy, so that it meets the users' demands for the topic-oriented search.

Formal concept analysis (FCA) is a data analytical method based on concept lattice. Because of the intuitive and concise representation features of knowledge, FCA has attracted a great attention of researchers since it had been put forward. Now it has been widely used in a wide range of areas such as software engineering, library, information science, data mining etc.

This paper is based on the principle of current topic crawler, advancing that apply the formal concept analysis, a data analysis tool to the relevant topic crawler algorithm, apply the concept lattice to the theme correlation analysis, thus improving the calculation method of theme related degree. The main research work of this paper include:

Firstly, this paper studies the FCA theory, focusing on the core part—concept lattice, especially the relationship between the concepts on concept lattice and the structure of concept lattice, then associates the concept lattice into the theme crawler algorithm.

Secondly, this paper studies the principle of theme crawler, including its structure, search strategy, Pagerank scheduling algorithm and theme related degree, then improves the calculation method of theme related degree on the base of concept lattice. Continually, this paper analyses the defects of Pagerank scheduling algorithm. With the combination of concept lattice, this paper finally proposes an improved Pagerank algorithm combining.

KEY WORDS: Formal concept analysis; Concept lattice; theme related degree; Pagerank

目录

Contents

第一章 绪论

1.1 论文选题的依据与意义

网络爬虫(Web Crawler)，又被称为网络机器人或者蜘蛛（Spider），它的主要作用是获取在互联网上的信息。网络爬虫利用网页中的超链接遍历互联网，通过 URL 引用从网页爬行到另一个网页。网络爬虫是搜索引擎的重要组成部分，是一个功能十分强大的能够自动提取网页信息的程序。网络爬虫收集到的信息具有多种用途，可以用来建立索引、验证 HTML 文件、获取更新信息等 。

随着网络信息资源爆炸式地增长，使用传统搜索技术准确、快速地查找用户所需要的信息变得十分困难。对于与之俱增的海量数据以及数以亿计的网页，通用搜索引擎在及时、准确地更新索引数据库方面面临巨大问题。此外，通用搜索引擎也很难深入抓取信息以及定向抓取信息，其采用的主要是基于关键词匹配的检索方式，而没有挖掘词语之间的语义关联，使最终的检索结果很难满足用户的需求。为了解决以上问题，面对特定主题定向抓取相关网页资源的聚焦爬虫（又称主题爬虫）应运而生。聚焦爬虫和传统的通用爬虫不同的是，它并不追求较大的覆盖，而是将抓取的目标限定为与某一类特定主题内容有关的网页，为面向主题的用户查询准备数据资源。从主题相关的领域内，获取、加工与搜索行为相匹配的结构化数据和元数据信息。

主题爬虫的设计是以通用爬虫为基础的，事实上它是对通用爬虫功能上的扩充。聚焦爬虫的设计主要包括如下几部分：[①] 确立主题、初始种子选取、主题相关度分析、查询结果排序。确立主题是指确定爬虫爬取的主题；初始种子是指事先指定的面向特定主题的起始种子网页，能够使爬行模块顺利开展爬行工作；主题相关度分析是指对网页进行主题相关度的计算，主题相关度是聚焦爬虫的核心，

[①] 汪涛，樊孝忠：《主题爬虫的设计与实现》，《计算机应用》第 24 卷第 6 期，2004 年。

它决定了待爬取页面的取舍；排序是对爬取页面的最后一步处理，是指给主题相关的页面评分，然后按照评分进行排序。

形式概念分析 (Formal Concept Analysis，FCA)[①]是 R. Wille 提出的一种从形式背景进行数据分析和规则提取的强有力工具，它是格论的分支，以数学为基础。形式概念分析对组成本体的对象、属性以及他们间的关系等用形式化的背景表示出来，然后根据形式背景，构造出概念格(concept lattice)，从而清晰地描述本体的结构。这种构建本体的方法是半自动化的，在形成概念的阶段，需要识别出领域内的对象、属性，以及构建它们间的关系，在概念生成之后，可以构造语境，然后利用概念格的生成算法 CLCA，自动产生本体。形式概念分析强调以人的认知为中心，提供了一种与传统的、统计的数据分析和知识表示完全不同的方法，成为了人工智能学科的重要研究对象，在机器学习、数据挖掘、信息检索等领域得到了广泛的应用。

本文拟利用形式概念分析理论结合现在比较成熟的算法，提出一种基于形式概念的聚焦爬虫设计，改进原有的主题爬虫相关算法。

1.2 研究相关动态及最新动态

1.2.1 形式概念分析研究现状

形式概念分析是知识发现和数据分析的有力数学工具，近些年来被广泛应用于信息检索、软件工程等领域。Neuss 等人[②]利用概念格对因特网上的文档信息进行自动分类和分析。Eklund 等人[③]将网页文档索引和导航进行了概念层次的展

---

① Ganter B and Wille R:《形式概念分析》，马垣，张学东，迟呈英，王丽君，等译，北京：科学出版社，2007 年，第 1—75 页。

② 董占兵:《基于形式概念分析的主题搜索策略研究》，硕士学位论文，西华大学，2007 年，第 18—19 页。

③ Eklund P and Martin P:*WWW indexation and document navigation using conceptual structures*, 2nd IEEE Conference on Intelligent Information Processing Systems(IVIPS'98),pp.217-221.

示。Lengnink[①]将向量空间模型中的相似度和距离等应用到格中，由此传统信息检索模型中的自动分类、聚类分析等方法能够应用于格结构中。Godin[②]等提出了基于增量的概念格构造方法并提出在概念格上提取蕴含规则的方法。Carpineto等人[③]用最短路径的方法来计算文档相似度，将查询插入文档集的概念格中，用来处理词语不匹配问题。Carpineto 和 Romano[④⑤]利用 FCA 的思想通过支持查询过滤和将查询和导航的结合，将 Google 的搜索结果构建成为基于格的元搜索引擎。此外，Mail-Sleuth[⑥]利用 FCA 挖掘大量的 Email 文档文件，并将其开发成为专业的软件。

1.2.2 聚焦爬虫研究现状

面向主题搜索的聚焦爬虫要解决的最主要的问题是搜索的网页要尽可能多的与搜索主题相关，同时要尽可能少的访问与主题无关的网页。主题爬行（聚焦爬行）作为网页信息检索的重要技术，是 1994 年在搜索引擎诞生的几乎同时间由 De. Bra[⑦]提出的，他把网络爬虫的爬行过程模拟成鱼群在网络上进行迁徙的活动方式，称为“鱼群搜索方法”(Fish Search)，这种方法利用二元分类方法对关键词进行简单匹配来判断网页的相关性。随后，1998 年 Hersovici[⑧]等人改进了 Fish Search 方法并改进为“鲨鱼搜索方法”(Shark Search)，其思想是综合

---

① 王莹煜:《基于多 Agent 系统的主题爬虫理解与协作研究》，硕士学位论文，西华大学，2010 年，第 3—4 页。

② Godin R,Mili H,Mineau G W,Missaui R:Design of class hierarchies based On concept lattices, Theory and application of object systems,vol.4,no.2(1997), pp.117-134.

③ Carpineto C and Romano G:Order-theoretical ranking, Journal of the American Society for Information Science,vol.51,no.7(2000), pp.587-601.

④ Carpineto C and Romano G:*Exploiting the Potential of Concept Lattices for Information Retrieval with CREDO,* Journal of Universal Computing,vol.10, no.8(2004),pp.985-1013.

⑤ Carpineto C and Romano G:*A lattice conceptual clustering system and its application to browsing retrieval,* Machine Learning,no.24(1996),pp.1-28.

⑥ Eklund P, Ducrou J and Brawn P:*Concept Lattices for Information Visualization: Can Novices Read Line Diagrams*, 2nd International Conference on Formal Concept Analysis(LNCS 2961),Berlin: Springer,2004,pp14-27.

⑦ De Bra P,Houben G,Kornatzky Y and Post R:*Information Retrieval in Distributed Hypertexts,*In Proceedings of the 4th RIAO Conference,New York,1994,pp.481-491.

⑧ Hersovici M, Jacovi M, Maarek Y, Pelleg D, Shtalhaim M:*The Shark-Search Algorithm—An Application: Tailored Web Site Mapping*, In Proceedings of the Seventh International World Wide Web Conference, Brisbane, Australia,1998.

考虑网页和链接锚文本的相关性，按照优先级对网页中的 URLs 进行排序，然后乘以一个衰减因子来继承上一级（父亲）页面的相关性。1998 年斯坦福大学 Cho[①] 等人提出了一些基于主题的爬行策略，主要对宽度优先策略进行了改进，虽然改动不太大，但是该方法在计算 URLs 优先权值时采用了 Pagerank[②]方法。除此之外，McCallum 等人采用朴素贝叶斯分类器(Naive Bayes Classifiers)将超链接分类[③④]，Diligenti 等人利用语境图(Context Graph)来指导主题爬行[⑤]。另外，Menczer [⑥]等基于神经元网络与进化计算，结合增强学习思想，使得其适应性较好。2006 年，Allna Formica[⑦]结合领域本体的知识对概念格中概念之间的相似度进行了分析，而且结合信息内容又对其进行了改进。

近年来，也有很多人提出了各种不同的主题爬行策略，但是其中的大多数也都停留在实验和理论阶段。如何抽象地描述用户所需的主题，怎样在爬行的过程中对所爬网页主题进行相关度判定，快速地穿越与主题不相关的网页，提高爬虫预测的准确性，以及如何对用户兴趣进行良好的学习，降低时间复杂度的计算，都成为聚焦爬虫亟待解决的问题。

1.3 本文的主要工作和论文结构

---

① Cho J, Garcia-Molina H, Page L:*Efficient Crawling Through URL Ordering*, Proceedings of the 7th ACM-WWW International Conference, Brisbane:ACM Press,1998,pp.161-172.

② Brin S, Page L:*The Anatomy of a Large Scale Hypertextual Web Search Engine*, Computer Networks and ISDN Systems Archive,no.30(1998),pp.107-117.

③ McCallum A, Nigam K, Rennie J:*Building domain-specific search engines with machine learning technique*, Proceedings of AAAI Spring Symposium on Intelligent Engine in Cyberspace,1999,pp.100-108.

④ Rennie J,McCallum A:*Using reinforcement learning to spider the web efficiently*, Proceedings of the 16th International Conference on Machine Learning ICML-99,1999,pp.335-343.

⑤ Diligenti M, Coetzee F M, Lawrence S:*Focused crawling using context graphs*,Proceedings of the International Conference on Very Large Database (VLDB'00),2000,pp.527-534.

⑥ Menczer F, Gant G, Srinivasan P:*Topic-driven crawlers*,Machine Learning Issues, ACM TOIT,2002,pp.58-70.

⑦ Formica A:*Ontology-based Concept Similarity in Formal Concept Analysis*, Information Sciences,2006,pp.2624-2641.

本文在通用爬虫的基础之上，通过形式概念分析的方法，对网页的内容和链接结构做深入细致的分析，改进了传统的 pagerank 权值计算方法以及主题相关度算法，本文的主要内容如下：

1、介绍形式概念分析相关知识；

2、介绍网络爬虫以及聚焦爬虫原理及相关知识；

3、研究并改进了基于形式概念分析的概念相似度计算模型。在基于概念格的相似度计算模型中考虑概念深度的影响加入概念距离，对网页进行相关度排序。

4、研究将形式概念分析中的概念格来分配 Pagerank 值，从而改进了 Pagerank 算法，使其 Pagerank 值的分配更合理。

本文共分五章，篇节安排如下：

第一章，绪论：介绍了形式概念和爬虫的发展历史，国内外相关的研究动态，以及本文研究内容和篇章结构。

第二章，形式概念分析简介：介绍了形式概念分析的相关知识，包括形式背景，形式概念，以及概念格的构造。

第三章，主题爬虫简介：分析介绍了网络爬虫的结构及基本原理，然后分析了主题爬虫的结构，以及主题爬虫的搜索策略。介绍了网络爬虫设计到的关键算法：首先讲述了主题爬虫的排序算法，着重研究了 Pagerank 和 HITS 算法。然后介绍了主题爬虫相关度的计算方法，主要是介绍了使用最为广泛的向量空间模型算法。

第四章，基于形式概念分析的主题爬虫算法改进：介绍了概念格距离并将其应用于改进基于概念格的相似度计算模型。接着重点分析了 Pagerank 算法的缺点，提出了通过形式概念分析将 Pagerank 算法改进用来对抓取来的网页排序。

第五章，总结与展望。

第二章 形式概念分析基本理论

形势概念分析（Formal Concept Analysis,简记 FCA）是由德国数学家 Wille[①]提出的，它属于应用数学和格论的一个分支，建立在概念和概念层次的数学化基础之上。一个概念最大限度的收集对集合中共同特点有帮助的元素，并且运用形式概念分析的方法，可以发现、构造和展示由属性和对象构成的概念及其之间的关系。形式概念分析的基本概念是形式背景和形式概念，背景中的所有概念可以通过一定的算法生成概念格。概念格是形式概念分析的核心，可通过哈斯图反映出其概念层次结构。

2.1 格论基本概念

2.1.1 偏序关系

定义 1[②③④] 在集合 A 上的一个二元关系 R, $\forall x, y, z \in A$满足以下条件：

xRy （自反性）

xRy, yRx⇒x=y （反对称性）

xRy, yRz⇒xRz （传递性）

则称 R 是集合 A 上的一个偏序关系，记为“≤”。集合 A 以及其上的序≤形成的有序二元组（A, ≤）称为偏序集。

定义 2[②③④] 设(A, ≤)为偏序集，如果$a, b \in A$且对于$\forall x \in B$，都满足$x \leq a$，

---

① Ganter B,Wille R:*Formal Concept Analysis*, Mathematical Foundations, Berlin, Germany: Springer,1999,pp.1-80.

② 陈杰：《格论初步》，内蒙古大学出版社，1990 年，第 1—15 页。

③ Davey B A, Priestly H A: *Introduction to Lattices and Order(Second Edition)*,Cambridge University Press,2002,pp.1-25.

④ 郑崇友，樊磊，崔宏斌：《Frame 与连续格(第二版)》，北京：首都师范大学出版社，2000 年，第 44—55 页。

则称 a 是子集 B 的上界。对偶的，如果$x \in B$都满足$b \leq x$，则称 b 为子集 B 的下界。

定义 3[①②③] 设（A, ≤）为偏序集，B 是 A 的子集，a 是 B 的任意上界，如果对于 B 的所有上界 y 均有$a \leq y$，则称 a 为 B 的最小上界，也称为上确界，即为 sup(B)。利用对偶原理，若 b 为 B 的任意下界且对于$\forall x$，都有$x \leq b$，则称 b 为 B 的最大上界，也称为下确界，记为 inf(B)。

2.1.2 完备格

定义 4[①②③] 设（A, ≤）是一个偏序集，如果 A 中的任意两个元素 a, b 都有上确界和下确界，则称 A 是一个格。

定义 5[①②③] 设（A, ≤）是一个偏序集，如果对于任意非空集合$S \subseteq A$都存在$\vee S$，则称（A, ≤）是一个完全并半格。如果对于任意非空集合$S \subseteq A$都存在$\wedge S$，则称（A, ≤）是一个完全交半格。如果（A, ≤）既是完全并半格，又是完全交半格，则称其为完备格。

2.2 背景与概念

定义 6[④⑤] 一个形式背景 K:=(G, M, I)是由两个集合 G 和 M 以及 G 与 M 间的关系 I 组成。G 的元素称为对象，M 的元素称为属性（严格地说是“形式对象”与“形式属性”）。(g, m)∈I 或 gIm 表示对象 g 具有属性 m。一个简单的形式背

---

① 陈杰：《格论初步》，内蒙古大学出版社，1990 年，第 1—15 页。

② Davey B A, Priestly H A: *Introduction to Lattices and Order(Second Edition)*,Cambridge University Press,2002,pp.1-25.

③ 郑崇友，樊磊，崔宏斌：《Frame 与连续格(第二版)》，北京：首都师范大学出版社，2000 年，第 44—55 页。

④ Ganter B,Wille R:*Formal Concept Analysis*, Mathematical Foundations, Berlin, Germany: Springer,1999,pp.1-80.

⑤ Ganter B and Wille R：《形式概念分析》，马垣，张学东，迟呈英，王丽君，等译，北京：科学出版社，2007 年，第 1—75 页。

景如表 2-1 所示。

表 2-1　形式背景 K=(Object={01, 02, 03, 04, 05}, Attribute={a, b, c, d, }, R)

A(Attribute) / 0(object)	a	b	c	d
1	×			
2		×	×	
3	×	×		×
4	×	×	×	×
5	×		×	

定义 7[①②]　形式概念(Concept)：序偶(E,I)是形式背景$K=(O,A,R)$的一个形式概念C(简称概念)，当且仅当$E\subseteq O$, $I\subseteq A$, 则称E为概念的外延(Extent)，而I为概念的内涵(Intent)。

设$C_1=(E_1,I_1)$和$C_2=(E_2,I_2)$是格中的两个概念，其中偏序关系“≤”定义为$C_1\le C_2\Leftrightarrow I_1\subset I_2$。此时称$C_1$是$C_2$的子概念(Sub-concept)，$C_2$是$C_1$的超概念(Super-concept)

根据偏序关系可以生成概念格的 Hasse 图，如果有概念$C_1\le C_2$，并且不存在另一个概念C_3使得$C_1\le C_3\le C_2$，则从C_1到C_2就存在一条边，即C_1是C_2的直接子概念，反之C_2是C_1 的直接超概念，满足直接子概念——超概念关系的所有概念节点的集合是一个完全格，每个概念节点(E,I)都是完全对，使得$E'=I$且

---

① Ganter B,Wille R:*Formal Concept Analysis*, Mathematical Foundations, Berlin, Germany: Springer,1999,pp.1-80.

② Ganter B and Wille R:《形式概念分析》，马垣，张学东，迟呈英，王丽君，等译，北京：科学出版社，2007 年，第 1—75 页。

$I^{'}=E$，其中

$$E^{'}=\{m\in A\mid gRm,\forall g\in E\}$$

$$I^{'}=\{g\in O\mid gRm,\forall m\in I\}$$

这个性质使对于同一个形式背景K而言，概念格的构造不受数据或属性排序的影响，是唯一的。由表 2-1 生成的 Hasse 图如图 2-1 所示：

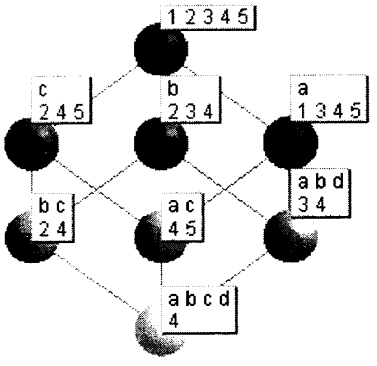

图 2-1 概念格的 Hasse 图

定义 8[①②] 背景(G, M, I)上的一个形式概念是二元组(A,B)，其中$A\subseteq G$，$B\subseteq M$，而且满足$f(A)=B, g(B)=A$。我们称 A 是概念(A,B)的外延，B 是概念(A,B)的内涵。B(G,M,I)表示背景(G,M,I)上的所有概念的集合。

2.3 概念格

定义 9[①②]若$(A_1,B_1),(A_2,B_2)$是某个背景上的两个概念，而且$A_1\subseteq A_2$（等价于$B_2\subseteq B_1$），则我们称(A_1,B_1)是(A_2,B_2)的子概念，(A_2,B_2)是(A_1,B_1)超概念，并记作$(A_1,B_1)\leq(A_2,B_2)$，关系≤称为是概念的“层次序”，（简称“序”）。(G,M,I)的所有概念用这种序组成集合用$\underline{B}$(G,M,I)表示，称它为背景(G,M,I)上的概念格。

---

① Ganter B,Wille R:*Formal Concept Analysis*, Mathematical Foundations, Berlin, Germany: Springer,1999,pp.1-80.

② Ganter B and Wille R:《形式概念分析》，马垣，张学东，迟呈英，王丽君，等译，北京：科学出版社，2007 年，第 1—75 页。

2.4 概念格的构造

在形式概念分析中，构造概念和概念格是最基本和首要的任务，概念格的构建过程实际上是聚类对象的过程。在用形式概念分析处理问题时，一般都需要根据已有的形式概念背景构建出概念格，因此，高效的概念格生成算法是解决问题的关键。自从提出形式概念分析理论以来，国内外很多相关研究人员已经提出了许多概念格的生成算法，归结起来主要分为两类：批处理算法和增量算法。

2.4.1 批处理算法

批处理算法一般适用于构建静态数据的概念格，对于在固定背景中构造概念格十分有效。该算法的构造过程主要包括两个方面：一是找出数据背景中所有概念，也就是概念格的结点；二是建立概念格中的概念（父结点与子结点）之间的关系。对于这两方面的实现有很多不同的方法，一种是在产生概念格的时候，生成少量概念，随后再将这些概念插入到各个结点的集合中，比较典型的是 Bordat 算法[①]。另一种是数据集的形式背景中生成所有的概念，然后再找出概念之间的父子关系，这种方式的典型的算法是 Chein 算法[②]。批处理算法主要分为三类：

（1）自顶向下算法

自顶向下算法首先生成概念格最上层的结点，然后从上到下，一级一级的构造概念格，典型算法有 Osham 算法，Bordat 算法等[③④]。

（2）自底向上算法

该算法与自顶向下算法恰好相反，首先生成概念格的最下层结点，然后逐步向上构造概念格，下层结点的构造要与其上一层的序对进行判断合并，典型算法

---

① Bordat J P:*Calculpratique du treillis de Galois d'une correspondence*, Math.EtSci.Humaines, 4eme annee,1986,no.96,pp.31-47.

② Chein M:*Algorithm de recherche des sous-matrices premieres d'une matrice*, Bull.Math.Soc.Sic.Roumanie,vol.13,no.61(1969),pp.21-25.

③ Ho T B:*An approach to concept formation based on formal concept analysis*,EICE Trans.Information and Systems, no.5(1995),pp.553-559.

④ 李鸿儒，魏平：《基于不可约元的概念格属性特征识别方法》，《计算机科学》第 33 卷第 6 期，2006 年。

有 Chein 算法等[①]。

（3）枚举算法

枚举算法思想是：首先按照一定的顺序列举出概念格中的所有概念，然后构造各结点间的关系，生成与之对应的哈斯图。比较著名算法有 Ganter 算法，Nourine 算法等[②③]。

2.4.2 增量算法

增量算法的思想是：首先将形式背景按照一定的算法生成概念格，然后将当前要插入的对象的属性集与之前生成概念格中的概念结点的内涵求交，根据交的结果的不同采取不同的办法，从而更新得到新的概念格。因此，增量算法不仅具有批处理算法的功能，还能处理动态的形式背景，其典型的增量算法有 Godin,Capineto 等[④⑤]。

2.5 概念格的特点

对于给定的数据集，利用一定的方法能够找出对应的形式背景，而概念格是对形式背景最直观的反映，如图 2-1。概念格所具有特点如下[⑥⑦]：

（1）概念格中每个结点都表示一个形式概念，而每个形式概念由两部分组

---

① 李树青，韩衷愿：《个性化搜索引擎原理与技术》，科学出版社，2008 年，第 50—200 页。

② Godin R:*Incremental concept formation algorithm based on Galois (concept) attics*, Computational Intelligence,vol.11,no.2(1995),pp.246-267.

③ Nourine L, Raynaud O: *A Fast Algorithm for Building Lattices*,Information Processing Letters,1999,pp.199-204.

④ Carpineto C, Romano G:Galois-*an order-theoretic approach to conceptual clustering*, Proceedings of ICML-93,1993,pp.33-40.

⑤ McCallum A, Nigam K, Rennie J:*Building domain-specific search engines with machine learning technique*, Proceedings of AAAI Spring Symposium on Intelligent Engine in Cyberspace,1999,pp.100-108.

⑥ 胡建，杨炳儒：《增量式广义概念格结构的生成算法研究与实现》，《计算机科学》第 36 卷第 5 期，2009 年。

⑦ 李鸿儒，魏平：《基于不可约元的概念格属性特征识别方法》，《计算机科学》第 33 卷第 6 期，2006 年。

成：外延和内涵，外延表示概念的对象，内涵表示概念的属性。

（2）概念格中的每个结点都标有其对应的概念的对象和属性，这样可以很直观的从 Hasse 图中把握每个结点的特性，便于从中找出所需信息。

（3）在概念格中能够轻易的找出对象或属性包含在哪些结点中，因为每个对象都出现在包含该对象结点的上升路径上。类似地，每个属性都出现在包含该属性结点的下降路径上。

（4）概念格的构造实现了对象聚类和分层。概念格中每个概念都包含多个对象和属性，也就是说这些属性相同的对象属于同一类。另外沿着概念格的某一条边自底向上，概念的属性不断减少，对象不断增加。

第三章 主题爬虫的基本原理

Crawler 即网络爬虫也称为蜘蛛，是一种自动抓取万维网网页信息的机器人，是搜索引擎的重要组成部分。世界上第一个网络爬虫有 MIT 的马休.格雷在 1993 年写成，命名为“万维网漫游者”。传统网络爬虫从一个或若干个初始网页的 URL 开始，获得初始网页上的 URL，在抓取网页的过程中，不断从当前页面上抽取新的 URL 放入队列，直到满足所设定的停止条件。主题爬虫的设计则相对复杂，需要经过网页分析算法去掉与主题无关的网页链接，保留其中有用的链接然后放入等待抓取的网页地址队列。接下来，再根据相应的爬取策略从队列中选取下一步要抓取的网页地址，之后循环上述过程，直到达到某一停止条件。此外，主题爬虫抓取过的网页会被系统保存，并进行分析、过滤，以及建立索引，为今后的查询和检索做好准备。

3.1 主题爬虫的结构

主题爬虫的基本思想是确定爬行队列中 URL 的优先级，然后从中选择可能性最大的也就是和主题相关度最高的网页爬行，这样就保证了爬行的效率和准确度。因此，主题爬虫的核心问题是要根据当前信息来确定队列的访问次序，也就是在爬行之前，要先计算该网页与主题的相关性。

主题爬虫是以通用网络爬虫为基础的，首先从初始种子网页开始，解析出其中的链接并保存到 URL 队列，爬行模块负责下载目标网页，然后相关度分析模块计算种子页面到该网页的链接数目及网页到种子网页集的链接数目，分析网页内容主题相似度，决定网页的取舍。最后，排序分析模块对爬行并保存下来的网页进行全面的评价和排序。主题爬虫系统的系统结构如图 3-1 所示[①]。

[①] 李树青，韩衷愿：《个性化搜索引擎原理与技术》，科学出版社，2008 年，第 50—200 页。

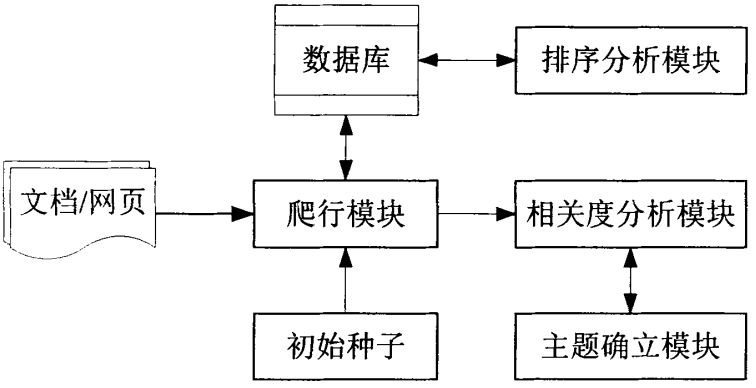

图 3-1　主题爬虫结构图

3.2 主题爬虫的搜索策略

主题爬虫的爬行策略是只爬取某一类特定主题的网页，主题爬虫会给其下载的页面计算出一个评价得分，之后根据这个评价得分将网页排序，最后再插入到一个队列中。通常情况下，最优的下一步爬取对象是将队列中的第一个页面进行分析并执行爬行程序，这种方式可以保证主题爬虫优先跟踪那些最有可能链接到目标网页的页面。网络爬虫爬取策略的重点是如何评价链接的评分，不同的评分方法计算出的链接的评分不同，表现形式就是链接的“重要程度”也不同，这也就决定了所采用的搜索策略不同。由于链接存在于页面之中，而一般情况下具有较高评分的网页所包含的链接也具有比较高的价值，因此对链接的评价得分有时也可以转变为对页面价值的评价。这种策略经常应用在专业的主题搜索引擎中，因为这类搜索引擎只关注某一类特定主题的页面.

目前，网页的抓取策略分为深度优先、广度优先和最佳优先三种。深度优先由于很有可能会导致爬虫的陷入黑洞，因此最常用的爬虫策略是广度优先和最佳优先策略。图 3-2 展示了爬虫抓取策略的过程

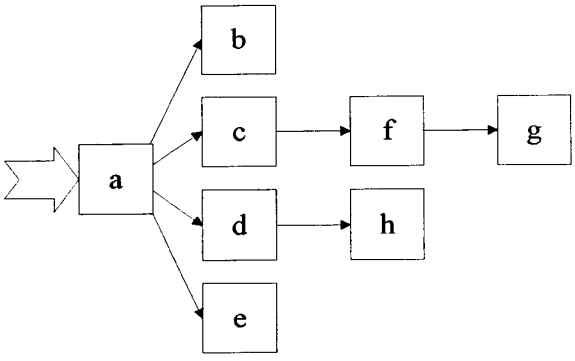

图 3-2　爬虫抓取策略过程图

3.2.1 深度优先策略

深度优先策略是指网络爬虫从起始页开始，一个链接一个链接跟踪下去，处理完这条线路之后再转入下一个起始页，继续跟踪链接，直到所有链接都被访问过为止。事实上，深度优先策略属于图算法的一种，其过程简言之就是对每一个可能的分支路径深入到不能再深入为止。如图 3-2，a 为种子节点，则其遍历顺序为 a→b→c→f→g→d→h→e。

3.2.2 广度优先策略

广度优先策略是指网络爬虫会先抓取起始网页中链接的所有网页，然后再选择其中的一个链接网页，继续抓取在此网页中链接的所有网页。广度优先策略的设计和实现方法相对比较简单，用这种方式也可以让网络爬虫并行处理，提高抓取速度。它的基本思想指出与起始 URL 在一定链接距离内的网页具有较高主题相关度的概率很大。此外，还可以将广度优先策略与网页过滤技术相结合，先利用广度优先策略抓取网页，然后再利用网页过滤技术将其中无关的网页过滤。这些方法的不足之处在于，当抓取的网页较多时，大量的无关网页将被下载和过滤，算法的时间效率变的较低。如图 3-2，首先遍历第一层 a，接着是第二层 bcde，然后遍历 fg，最后 h，即顺序为 a→b→c→d→e→f→g→h。

3.2.3 最佳优先策略

最佳优先策略也称为“页面选择问题”，通常保证在有限带宽的条件下，尽

可能地照顾到重要性高的网页。这种策略主要思想是利用一些网页分析方法，计算候选 URL 与目标网页的主题的相关度或相似度，并从中选取评分最高的一个或几个网页进行抓取。也就是说这种方法首先访问经过主题相关性分析被预测为最有价值的网页。这其中的问题是，在爬虫爬行路径上可能会有很多有用的网页被忽略，因而最佳优先策略只能算是一种局部最优的爬行策略。因此，很多时候需要将最佳优先策略与实际的情况相结合进行改进，从而达到更好的效果。如图 3-2，假设其重要性为 d>b>c>a>e>f>g>h，则遍历过程为 a→d→b→c→e→f→h→g。

3.3 网络爬虫的排序算法

网页质量评测标准最主要的方式为网页链接关系评价，即对页面之间相互引用关系的分析来确定链接的重要性，进而决定链接爬行的顺序，通常认为有较多入链或出链的页面具有较高的价值。基于链接的评价最有代表性的算法是 Pagerank 和 Hits。

3.3.1 Pagerank 算法

Pagerank 算法是 1998 年由斯坦福大学的 Sergey Brin 和 Lawrence Page 提出的，同时他们也是谷歌的创始人，该算法是 Google 算法的重要内容。

它独创的“链接评价体系”（Pagerank 算法）是基于这样一种认识，为了得到更好的搜索结果，使搜索引擎抛弃垃圾网页，需要计算网页本身的重要性。假设用户是随机访问网页的，浏览完一个网页后，可能再通过其链出链接访问其他网页。为了更好的对网页排序，把用户可能访问的页面排在前面，因此被链接次数多的网页，访问的可能性更大。

Pagerank 算法最初是用来对 Google 搜索引擎检索结果的排序，近年来逐渐被应用于网络爬虫对链接重要性的评价。

标准的 Pagerank 算法[①]是：

$$PR(A) = (1-d)/N + d \cdot (PR(T1)/C(T1) + \ldots + PR(Tn)/C(Tn))$$

---

① 罗刚：《解密搜索引擎技术实战》，电子工业出版社，2011 年，第 25—96 页。

其中：

N：网页总数；

PR(A) ：网页 A 的 Pagerank 值；

PR(Ti) ：链接到 A 的网页 Ti 的 Pagerank 值；

C(Ti) ：网页 Ti 的出站链接数量；

d ：阻尼系数，0<d<1。

这里，公式中第一个部分可以认为是固有得分，第二部分可以认为是因为被其他网页指向而得到其他网页的一部分得分。一方面，Pagerank 算法并不是将整个网站排名，而是以单个页面为单位计算的。另一方面，页面 A 的 Pagerank 值取决于那些链接到 A 的页面的 Pagerank 值，PR(A)的值并不是平均的受到 PR(Ti)值影响。另外，由 Pagerank 的计算公式能够看出，链接T_i对 A 的影响还受到T_i的出链接数$C(T_i)$的影响。也就是说，T_i的出链接越多，网页 A 受到T_i链接的影响就越小。由于 PR(A)是所有 PR(Ti)的和，因此，每增加一个入链接都会增加 PR(A)值。另外，所有 PR(Ti)的和还要乘以阻尼系数 d，d 的值介于 0 到 1 之间。可以看出，阻尼系数 d 的引入，减少了链接页面对当前页面 A 的排序贡献。

网络爬虫可以利用 Pagerank 值决定某个 URL 所需要抓取的网页数量和深度。重要性高的网页抓取的页面数量相对多一些，而 pagerank 值第的网页则抓取的页面少一些，低于一定的阈值就不抓取。

3.3.2 HITS 算法

HITS 算法是网页结构挖掘中权威性最高和使用最广泛的算法[①]。

它的算法思想是利用网页之间的引用来挖掘隐含在其中的有用信息，计算方法简单且效率高。HITS 算法通过内容权威度（Authority）和链接权威度（Hub）来对网页的质量进行评价。

内容权威度与网页本身提供的内容信息有关，网页被其他网页引用的越多，其内容权威度就越高。链接权威度与网页超链接页面的质量相关，引用越多高质

[①] 杨炳儒，李岩，陈心中，王霞：《Web 结构挖掘》，《计算机工程》第 29 卷第 20 期，2003 年。

量页面的网页，则其链接权威度就越高。

HITS 算法指出将每个网页的内容权威度和链接权威度分开考虑，在评价网页内容权威度的基础上再对页面的链接权威度进行评价，最后给出该页面的综合评价。

HITS 算法和 Pagerank 算法虽然均为链接分析算法，但是二者也有着明显的不同点。Pagerank 算法是基于随机冲浪(Random Surfer)模型的，它将网页权值直接从 authority 网页传递到 authority 网页，而 HITS 算法则是将 authority 网页的权值经过 hub 网页的传递进行传播。HITs 的 authority 值只是相对于某个检索主题的权重，而 Pagerank 算法是独立于检索主题的。

3.4 主题相关性算法

主题爬虫的主题相关度的计算有多种方法，如坐标匹配、内积相似度、向量空间模型(VSM)等等。目前，最为常用的为向量空间模型方法，由于向量空间模型对文档训练的要求比较低，能够从较少的训练文档中提取出主要的目标特征，有利于网络信息的探索和发现。基于向量空间模型[①]VSM 的主题相关度计算基本思想为计算两个文档向量之间夹角的余弦值。

令 X 和 Y 是两个 n 维向量，且 X=($x_1, x_2 \ldots x_n$)，Y=($y_1, y_2 \ldots y_n$)，向量间夹角θ满足公式：$X \cdot Y = |X||Y|\cos\theta$，也就是内积公式，其中$|X| = \sqrt{\sum_{i=1}^{n} x_i^2}$。夹角$\theta$可通过如下公式计算：

$$\cos\theta = \frac{X \cdot Y}{|X||Y|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

上面公式描述了两层含义，首先描述了归一化问题，即用欧式长度来描述文

[①] Witten I.H,Moffat A, Bell T.C:《深入搜索引擎—海量信息的压缩、索引和查询 》,梁斌 译，电子工业出版社，2009 年，第 200—350 页。

档 n 维文档术语权重集合的长度；其次，将一个文档集合表示为一个 n 维空间中的一个点，短文档接近原点，文档越长距离原点越远。VSM 模型将特征项作为文档的坐标，特征项可以选择字、词和词组等(一般选择词)，表示向量中的各个分量。由于θ为 0 时，$\cos\theta=1$；θ为 1 时，$\cos\theta=0$。主题相关度的度量转化为文档向量和查询向量的夹角余弦，余弦值越大，相关度越高。

综上所述，基于向量空间模型的主题相关度为：

$$\cos(D,D_i)=\frac{D\cdot D_i}{|D||D_i|}=\frac{\sum_{m=1}^{n}w_m w_{i,m}}{\sqrt{\sum_{m=1}^{n}w_m{}^2}\sqrt{\sum_{m=1}^{n}w_{i,m}{}^2}}$$

其中，D 为查询主题，D_i为待爬行页面，w_m和$w_{i,m}$分别为查询主题和待爬行页面向量的第 m 维在相应文本 D 和D_i中对应的权值。

第四章 基于形式概念分析的主题爬虫算法改进

本章在前文对形式概念分析和主题爬虫的学习与研究的基础上，提出了改进的基于形式概念分析的计算主题相关度的方法，提供了一种新的爬行策略思路。另外，对于爬行结果的排序，将 pagerank 算法和形式概念分析相结合，从形式概念分析的角度改进了 pagerank 算法。

4.1 基于形式概念分析的主题相关度

形式概念分析可以利用概念格反映概念之间的关系，其实质是将概念间的关系抽象成了语义网络，通过关系网能够发现概念之间直接的或间接的关系。将概念格应用到主题爬行中，不但能实现用户所感兴趣主题的表示，而且还将主题以概念的形式汇集起来。形式概念分析将主题搜索提高到了概念的层次，使用户不仅能获取主题资源，还能发现与这些主题相关或者相近的主题，一定程度上丰富了主题特征，提高了发现大量相关主题资源的预测。

4.1.1 概念格上的概念层次距离

众所周知，概念格是由概念组成的，概念是概念格的基本的单位，一个概念可以看作是一个类，因此概念格同时反映了类之间的关系。概念距离反映了他们的相似度，与语义距离类似，也描述两个不同类之间的继承关系或二元关系链中最短关系链的长度。经常采用的计算概念间距离的方法是 WhiteBoard 和 ChatingRoom 的 GCSM 距离[①]，其思想是：概念（结点）间距离由概念格中结点和共同父结点决定，概念深度越大，概念涵盖属性越多即表达越清楚，两结点距离父节点越小，则结点间关系越紧密。其公式为：

---

① 董占兵：《基于形式概念分析的主题搜索策略研究》，硕士学位论文，西华大学，2007 年，第 18—19 页。

$$dis(a,b)=\frac{dep(a)+dep(b)}{2\times dep(LCA(a,b))}$$

其中，a，b 表示两个概念，LCA(a,b)表示 a 和 b 的共同父概念，dep(a)和 dep(b)表示各子结点距离根结点的深度。这里我们定义根节点到其本身的深度为 1，子节点在父节点深度的基础上加 1。这样就解决了有些概念间父节点为根节点的问题，否则求得的距离为无穷大。通过这一细小改动，概念间的层次距离就相对减小了，使得概念间的关联加强了。

下面举出一个实例，如图 4.1。

在图 4.1 的概念层次树中，d 和 e 的深度分别都为 3（dep(d)=dep(e)=3），即到根结点 Root 的距离为 3，它们的父节点为 a，其深度 dep(a)=2。由此，根据距离计算公式可得 d 和 e 之间的距离为

$$dis(d,e)=\frac{dep(d)+dep(e)}{2\times dep(a)}=\frac{3+3}{2\times 2}=1.5$$

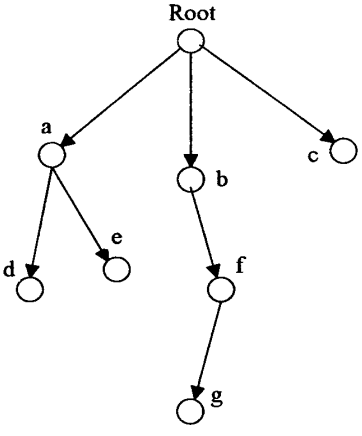

图 4-1 距离的概念层次树

4.1.2 基于概念格的相似度计算模型

Souza 和 Davis[①]利用概念格中概念的不可约下确界为依据提出了相似度计算模型，其主要思想是利用格理论中的∨运算与∧运算操作概念的不可约下确界，

① 王凯:《基于概念格的领域本体概念相似度提取方法研究》，硕士学位论文，安徽农业大学，2011 年，第 26—29 页。

计算公式为：

$$sim(a,b)=\frac{\left|(a\vee b)^{\downarrow}\right|}{\left|(a\vee b)^{\downarrow}\right|+\alpha\left|(a-b)^{\downarrow}\right|+(1-\alpha)\left|(b-a)^{\downarrow}\right|}$$

式中，

$a\vee b$表示概念 a 和 b 的上确界；

$(a\vee b)^{\downarrow}$表示形式概念 a 与 b 的上确界特征中的不可约下确界元素集；

$(a-b)^{\downarrow}$表示在 a 中，不在 b 中的不可约下确界元素集；

$(b-a)^{\downarrow}$表示在 b 中，不在 a 中的不可约下确界元素集。

4.1.3 改进的基于概念格的主题相关度

对于 4.1.2 中的概念格相似度计算模型，其仅仅考虑用概念的上层结点个数计算，缺乏对于概念深度及距离的考虑。因此，本文将概念格上距离引入 4.1.2 计算模型，提出改进概念格相似度计算模型的新算法。概念相似度计算的基本思想是：概念相似度可以由概念在概念格层次结构中的距离来度量，直观上可以看出：距离越大，相似度越低；相反地，两个概念之间距离越小，其相似度越大。也就是说，距离为 0 时，其相似度为 1；概念距离为无穷大时，其相似度为 0；相似度为概念距离的单调减函数。概念间的距离在概念格中能够用对象和属性的相似度来衡量：两个概念间的相似度越大，它们之间共有的对象和属性就越少；相反，两个概念间的距离越小，它们之间共有的对象和属性就越多。此外，我们还应考虑概念层次距离对概念相似度的影响。改进后的公式为：

$$sim((x_1,y_1),(x_2,y_2))=((\frac{|x_1\cap x_2|}{\gamma})\times\alpha+(\frac{|y_1\cap y_2|}{\lambda})\times(1-\alpha))\times(1-q)^{dis(a,b)}$$

式中，

$(x_1,y_1),(x_2,y_2)$表示两个概念，x_1,x_2表示概念的对象，y_1,y_2表示概念的属性；

$x_1\cap x_2,y_1\cap y_2$分别表示两个概念间共有的对象和属性；

$\gamma=\max(|x_1|,|x_2|),\lambda=\max(|y_1|,|y_2|)$；

dis(a,b)表示概念 a 与 b 之间的概念层次距离；

q 是为了体现概念层次距离对相似度的影响而作的修正，取值区间为(0,0.1]，其特点是两概念间的概念层次距离越大，概念间相似度越小。

4.1.4 改进的主题相关度的应用

主题爬虫首先要建立用户兴趣主题页面集合，也就是通常所说的爬虫爬行的初始种子。在获取了用户兴趣主题集合后，就要利用其建立主题特征模型，利用这个模型就能很好的表现主题。可以利用形式概念分析的方法，通过建立概念格来描述用户的搜索背景。可以将要爬取的网页集合描述为概念格中的对象集，网页主题的关键字集合描述为属性。例如，用户兴趣主题属性集为{a：人工智能；b：知识发现；c：机器学习；d：模式识别；e：数字图像处理；f：数据挖掘}，1-7 分别表示涵盖上述属性的网页，{1,2,3,4,5,6,7}组成了网页主题集，相应的形式背景如表 4-1，由表 4-1 建立的概念格如图 4-2。

表 4-1 用户主题形式背景

Attribute / Object	a	b	c	d	e	f
1	*	*	*	*		
2				*		*
3				*	*	*
4			*		*	
5			*	*	*	
6			*			*
7	*	*	*	*	*	

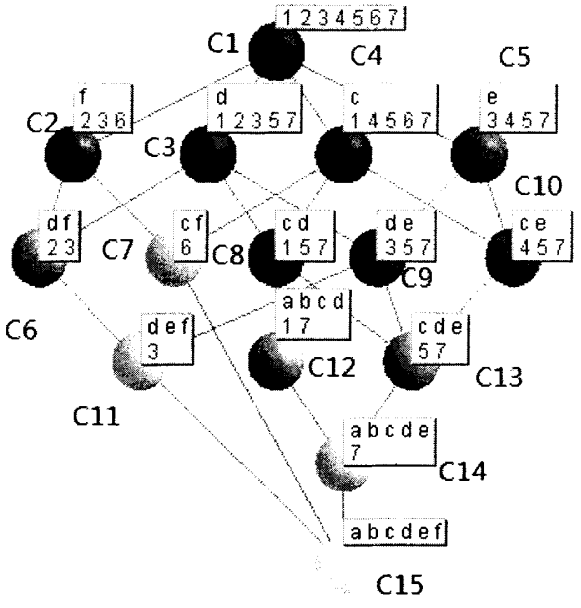

图 4-2 用户主题概念格

例如，现在用户所感兴趣的主题属性集为{d,e}，则首先从概念格中找出包含{d,e}的概念，即({3,5,7},{d,e}),({5,7},{c,d,e}),({7},{a,b,c,d,e}) 。很明显 3,5,7 网页跟用户主题属性集相关性最高的，在这种情况下，可以选取({3,5,7},{d,e})(即 C9)概念为主题概念（主题概念表示抽象出的用户主题的网页与关键词所构成的概念格中的概念），并从这其中的网页开始爬取。而对于其他网页，爬取的顺序则需计算其与核心概念间的概念相似度，也就是主题相关度，之后就可以确定爬虫爬行路径。

根据 4.1.3 中改进的概念相似度计算公式，可计算其他概念与核心概念间的相似度。一般情况下，由对偶原理可知概念的属性和对象在概念格中具有同样的地位。但是，由于主题爬虫更多的是使用关键字，也就是主题属性集作为判别主题相关度的依据，所以这里α应当取的小一些，这里我们取$\alpha=0.2$。此外，由于 q 是为了体现概念层次距离对相似度的影响，这里不妨取为 q=0.05，则可以计算其他概念与主题概念间的相似度。例如选取(C2,C6,C7,C8,C11,C13)计算其概念相似度如下：

sim(C9,C2)=0.06

sim(C9,C6)=0.43

sim(C9，C8)=0.49

sim(C9,C11)=0.57

sim(C9,C13)=0.63

sim(C9,C14)=0.36

从计算的结果来看，概念与主题概念的共有属性越多，其相似度值就越高，即主题相关度越高，这就给主题爬行提供了策略依据。由此，主题爬虫就可以确立爬行网页的主题相关度，进而指导爬行模块的爬行。

算法如下：

算法 1：用来计算概念格上概念到主题概念间的层次距离

```
Input：L,TopicConcept    //概念格
Input：Struct Concept   //定义概念格的结点
        {
            Attributes;   //存储概念信息（对象和属性）
            Objects;
            Distance;   //定义结点到根结点的距离
        }
Output：Distance(TopicConcept,Concept)    //输出概念格中概念到主题概念的
层次距离
Begin
    FindMaxConcept(L)→Concept(M)   //找出格的顶点
    FindMinConcept(L)→Concept(0)   //找出格的底部点
    Concept(M) →Queue(L).enqueue   //
    Concept(M).Distance=1
    while(Queue(L)!=empty)
        {
            NextConcept=Queue(L).dequeue
            If(NextConcept!= Concept(0))
                {
                    NextConcept →VisitedSets
                    If(NextConcept. NextConcept ∉ VisitedSets)
```

```
                    {
                        NextConcept.NextConcept.Distance=
                        NextConcept.Distance+1
                        NextConcept.NextConcept→ Queue(L).enqueue
                    }
                }
            }
    Concept(0). Distance= NextConcept. Distance+1
    Initialize(TopicConcept,VisitedConcept)
    TopicConcept →Queue(L).enqueue
    TopicConcept →VisitedConcept
    While(Queue(L)! =empty)
        {
            Queue(L).dequeue →TestConcept
            If(TestConcept!= Concept(0))
            {
            FindFatherConcept(TestConcept, TopicConcept)→FatherConcept
            TestConcept.Distance=(TestConcept.Distance+TopicConcept.
            Distance)/2* FatherConcept. Distance
            }
        }
End
```

算法 2：计算概念格上概念主题概念的概念相似度（主题相关度）

Input：概念对集合 Compare1(TopicConcept,Concept1)，

Compare2(TopicConcept1,Concept2) ……

Concept

Output：主题相关度 sim

Begin

```
    Initialize(TopicConcept,VisitedSim)
    TopicConcept →VisitedSim
    While(Queue(L)! =empty)
        {
            Queue(L).dequeue →SimConcept
  If(SimConcept!= Concept(0))
    {
    SimConcept.Objects ∧ TopicConcept.Objects →M   //M、N 分别为两概念
 SimConcept.Attributes ∧ TopicConcept.Attributes→ N   //共有对象和属性个数
    P=Max{SimConcept.Objects，TopicConcept.Objects}
    Q=Max{SimConcept.Attributes，TopicConcept.Attributes}
  Compare(TopicConcept,Concept(i)).Sim=((M/P)*α+(N/Q)*(1−α))*
  (1-q)^ SimConcept.Distance              // SimConcept.Distance 为算法 1 计算
                                              的概念层次距离
  Compare(TopicConcept,Concept(i)).Sim→Queue1.enqueue //将计算出的主题
                                                  相关度存入队列 Queue1
        }
  }
End
```

4.2 改进的 Pagerank 算法

4.2.1 原有 Pagerank 算法的缺点

近些年， Pagerank 算法一直是很多学者学习和改进的兴趣点，主要是因为传统 Pagerank 算法存在一些问题：

（1）首先，Pagerank 算法利用链接结构来作为衡量网页重要性的依据，却忽略了行为、内容等对网页重要性的影响。

（2）其次，Pagerank 算法采用的是随机冲浪思想，由公式中的 d (PR(T1)/C(T1) + ... + PR(Tn)/C(Tn))可以看出，网页把自己的 Pagerank 值 PR(Ti)平均分配给它链

出的网页，这就在较大程度上影响了链出网页的 Pagerank 值。然而，这并没有考虑到用户更多的时候浏览的是与当前网页相关性大的网页，而不是随机浏览。也就是说，实际用户浏览当前网页所链接的网页的概率是不同的，这也是 Pagerank 值主要问题所在。

4.2.2 基于形式概念的 Pagerank 算法

下面本文结合形式概念分析，主要针对 4.2.1 中第二个问题提出 Pagerank 算法改进的新算法，改进后的 Pagerank 公式为：

$$\mathrm{PR(A)}=\frac{1-d}{N}+d\sum_{i=1}^{n}PR(T_i)\cdot B(A_{T_i})$$

式中 $B(A_{T_i})$ 为由 rank 格计算得出的 A 网站在 T_i 网站的权值(由 T_i 网站所有出链接所占比重权值计算得出)。

首先来看下面的网络简化图。

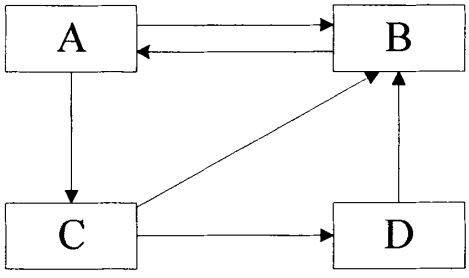

图 4-3 网络链接图

假设上图描述的是四个网页之间链接关系，设网页 A 的 Pagerank 值 PR(A) 为 0.8，则按照原有的 Pagerank 算法，会将其 Pagerank 值平均的分给 B 和 C，即都为 0.4。但是，实际上 B 和 C 的重要程度是不一样的。那么，如何来衡量他们之间重要性的大小呢？按照 Pagerank 算法的思想，网页的入链接都是给自己增加 Pagerank 值，也就是说入链接越多本网页就越重要，所以在将网页 Pagerank 值分配给出链接时，应该考虑出链接的入链接数。

基于此，本文的新想法是结合形式概念分析，来划分出链接权重的分配。首先，将图 4-3 绘制成表格 4-2。

表 4-2 网站背景表

出 / 入	A	B	C	D
a		×		
b	×		×	×
c	×			
d			×	

表格 4-2 的含义为：表格中大写的 A,B,C,D 和小写的 a,b,c,d 都表示图 4-3 中的网站 A,B,C,D，大写表示的是链接的起点，小写表示的为链接的终点，例如表中第一行第二列有一个×表明有一个链接从 B 指向 a，如图 4-3 所示。按上述方式将图 4-3 绘制成表 4-2，上述表格描述的含义类似于图论中有向图的邻接矩阵。

上述表格非常类似于形式概念分析中的形式背景，但是实质上并不是形式背景，因为形式背景描述的是对象和属性以及他们之间的关系。但是表格描述的是网站及他们之间的链接关系，鉴于其相似性，我们不妨命名其为 rank 背景。我们知道，在确定了形式背景后就能够利用概念格的生成算法绘制概念格。类似地，我们有了 rank 背景，也可以利用概念格的生成算法绘制其格形式，我们暂且称为 rank 格。根据表 4-2 网站 rank 背景绘制其 rank 格如图 4-4。

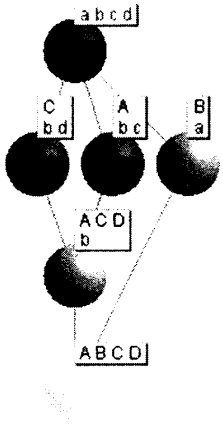

图 4-4 网站 rank 格

从图 4-4 中，我们能够清晰的看出网站入与出链接的情况，例如图中结点((A,C,D),b)表示网站 b 有三条如链接，分别是从 A，C，D 三个网站链接过来的。为了改进 Pagerank 算法中 Pagerank 值分配的问题，我们可以利用绘制的 rank 格

来给出链接网站分配权值。我们设 A,B,C,D 同时指向的链接（A,B,C,D 表示链接的起点）为 1，而对于接下来的结点按照其入链接的所占比重分配权值。如图 4-2 所示，底结点权值为 1，与其相邻的结点只有((A,C,D),b)，(B,a)，b 由于有三个入链接，而 a 只有一个入链接，所以按照比重 b 的权值为$\frac{3}{4}$，a 的权值为$\frac{1}{4}$。依此类推，d 的权值为$\frac{3}{4}\times\frac{2}{3}\times\frac{1}{2}$，即其父节点（从下向上看）权值乘以出链接所占比重（$\frac{2}{3}$）再平分给当前网站（b,d)。则计算出的权值分别为 a：$\frac{1}{4}$，b：$\frac{3}{4}$，c：$\frac{1}{4}$，d：$\frac{1}{4}$。因此，计算 Pagerank 值时就可以将利用 rank 格计算出的权值加入 Pagerank 计算公式。

下面我们将传统 pagerank 算法和改进后的算法进行比较，将上面例子 pagerank 值分别计算(设 A 起始 pagerank 值为 1)，如表 4-3。

算法 / 网页	原算法	改进算法
B	0.38	0.44
A	0.33	0.38
C	0.16	0.11
D	0.13	0.07

从表中可以看出，改进算法并没有改变原有算法的最终排名，但是通过改进，使得重要网页的 pagerank 值提高了，而不重要网页的 pagerank 值下降了。这样，既保证了改进算法的正确性，又使得 pagerank 值在迭代的过程中不是均匀的分配，而是更侧重分配给重要的网页，使得 pagerank 的分配更加合理。这样，对于杜绝垃圾网页等提供了现实的参考意义。

第五章 总结与展望

随着移动互联网的迅速发展，网络信息也呈现出爆炸式增长的现象，人们从海量信息中获取所需信息的要求也随之改变，搜索引擎技术就称为很多大公司和学者研究的热点。其中，网络爬虫作为搜索引擎基石，自然而然也就称为热点，通用爬虫因以搜集海量信息为目标，无法满足人们对特定信息的查询，主题爬虫就称为解决这类问题的关键。

概念格以图的结构来表现数据间的关系，能够针对指定用户形成一个表现该用户兴趣的概念背景，与主题爬虫的有关算法结合，可以展现主题背景的模型。本文通过对主题爬虫的结构和特点的学习和分析，提出了将形式概念分析中概念格应用到主题爬虫的相关度分析中，来改进主题爬虫的爬行。此外，类比概念格的特点改进了 Pagerank 算法。因此，本文在主题爬虫的两个关键算法的改进，具有很现实的理论意义。

由于时间、精力和水平有限，本文的研究工作也存在很多缺点和不足。首先，当将结合了概念格的相关算法应用到数据比较庞大或复杂领域时，相应的概念格势必会非常复杂，对于概念格的构造算法要求较高，然而在相关领域这一问题始终还没有解决。其次，由于本文研究只是对主题爬虫中主题相关性和排序算法的改进，而且由于缺乏强有力的工具和帮手，并没有搭建一个完整的爬虫系统进行实现。最后，对于形式概念分析和主题爬虫原理可能理解和研究不够深入，还需要对改进的算法进行更加深入的论证。

参考文献

[1] Bordat J P:*Calculpratique du treillis de Galois d'une correspondence*, Math.EtSci.Humaines, 4eme annee,1986,no.96,pp.31-47.

[2] Brin S, Page L:*The Anatomy of a Large Scale Hypertextual Web Search Engine*, Computer Networks and ISDN Systems Archive,no.30(1998),pp.107-117.

[3] Carpineto C and Romano G:*A lattice conceptual clustering system and its application to browsing retrieval*, Machine Learning,no.24(1996),pp.1-28.

[4] Carpineto C and Romano G:*Exploiting the Potential of Concept Lattices for Information Retrieval with CREDO*, Journal of Universal Computing,vol.10, no.8(2004),pp.985-1013.

[5] Carpineto C, Romano G:*Galois-an order-theoretic approach to conceptual clustering*, Proceedings of ICML-93,1993,pp.33-40.

[6] Carpineto C and Romano G:*Order-theoretical ranking*, Journal of the American Society for Information Science,vol.51,no.7(2000), pp.587-601.

[7] Chein M:*Algorithm de recherche des sous-matrices premieres d'une matrice*, Bull.Math.Soc.Sic.Roumanie,vol.13,no.61(1969),pp.21-25.

[8] 陈杰：《格论初步》，内蒙古大学出版社，1990年，第1—15页。

[9] Cho J, Garcia-Molina H, Page L:*Efficient Crawling Through URL Ordering*, Proceedings of the 7th ACM-WWW International Conference, Brisbane:ACM Press,1998,pp.161-172.

[10] Davey B A, Priestly H A: *Introduction to Lattices and Order(Second Edition)*,Cambridge University Press,2002,pp.1-25.

[11] De Bra P,Houben G,Kornatzky Y and Post R:*Information Retrieval in Distributed Hypertexts*,In Proceedings of the 4th RIAO Conference,New York,1994,pp.481-491.

[12] Diligenti M, Coetzee F M, Lawrence S:*Focused crawling using context graphs*,Proceedings of the International Conference on Very Large Database (VLDB'00),2000,pp.527-534.

[13] 董占兵：《基于形式概念分析的主题搜索策略研究》，硕士学位论文，西华大学，2007年，第 18—19页。

[14] Eklund P, Ducrou J and Brawn P:*Concept Lattices for Information Visualization: Can Novices Read Line Diagrams*, 2nd International Conference on Formal Concept Analysis(LNCS 2961),Berlin: Springer,2004,pp14-27.

[15] Eklund P and Martin P:*WWW indexation and document navigation using conceptual structures*, 2nd IEEE Conference on Intelligent Information Processing Systems(IVIPS'98),pp.217-221.

[16] Formica A:*Ontology-based Concept Similarity in Formal Concept Analysis*, Information Sciences,2006,pp.2624-2641.

[17] Ganter B,Wille R:*Formal Concept Analysis*, Mathematical Foundations, Berlin, Germany: Springer,1999,pp.1-80.

[18] Ganter B and Wille R:《形式概念分析》，马垣，张学东，迟呈英，王丽君，

等译，北京：科学出版社，2007年，第1—75页。
[19] Godin R:*Incremental concept formation algorithm based on Galois (concept) attics*, Computational Intelligence,vol.11,no.2(1995),pp.246-267.
[20] Godin R,Mili H,Mineau G W,Missaui R:*Design of class hierarchies based On concept lattices*, Theory and application of object systems,vol.4,no.2(1997), pp.117-134.
[21] Hersovici M, Jacovi M, Maarek Y, Pelleg D, Shtalhaim M:*The Shark-Search Algorithm—An Application：Tailored Web Site Mapping*, In Proceedings of the Seventh International World Wide Web Conference, Brisbane, Australia,1998.
[22] Ho T B:*An approach to concept formation based on formal concept analysis*,EICE Trans.Information and Systems, no.5(1995),pp.553-559.
[23] 胡建，杨炳儒：《增量式广义概念格结构的生成算法研究与实现》，《计算机科学》第36卷第5期，2009年。
[24] 李鸿儒，魏平：《基于不可约元的概念格属性特征识别方法》，《计算机科学》第33卷第6期，2006年。
[25] 李树青，韩衷愿：《个性化搜索引擎原理与技术》，科学出版社，2008年，第50—200页。
[26] 罗刚：《解密搜索引擎技术实战》，电子工业出版社,2011年，第25—96页。
[27] McCallum A, Nigam K, Rennie J:*Building domain-specific search engines with machine learning technique*, Proceedings of AAAI Spring Symposium on Intelligent Engine in Cyberspace,1999,pp.100-108.
[28] Menczer F, Gant G, Srinivasan P:*Topic-driven crawlers,Machine Learning Issues*, ACM TOIT,2002,pp.58-70.
[29] Nourine L, Raynaud O: *A Fast Algorithm for Building Lattices*,Information Processing Letters,1999,pp.199-204.
[30] Rennie J,McCallum A:*Using reinforcement learning to spider the web efficiently*, Proceedings of the 16th International Conference on Machine Learning ICML-99,1999,pp.335-343.
[31] 王凯：《基于概念格的领域本体概念相似度提取方法研究》，硕士学位论文，安徽农业大学，2011年，第26—29页。
[32] 汪涛，樊孝忠：《主题爬虫的设计与实现》，《计算机应用》第24卷第6期，2004年。
[33] 王莹煜：《基于多Agent系统的主题爬虫理解与协作研究》，硕士学位论文，西华大学，2010年，第3—4页。
[34] Witten I.H,Moffat A, Bell T.C: 《深入搜索引擎—海量信息的压缩、索引和查询 》，梁斌 译，电子工业出版社，2009年，第200—350页。
[35] 杨炳儒，李岩，陈心中，王霞：《Web 结构挖掘》，《计算机工程》第29卷第20期，2003年。
[36] 郑崇友，樊磊，崔宏斌：《Frame与连续格(第二版)》，北京：首都师范大学出版社，2000年，第44—55页。

致谢

三年的研究生生活转瞬即逝，在论文完成及行将毕业之际，我由衷的感谢三年里给予我帮助的老师，同学和家人。

首先要郑重的感谢我的导师何伟教授和樊磊教授，从毕业论文创作自始至终无不寄托着老师的希望和付出，正是他们的指导给了我指引和方向，可以说本文从选题到最后的完稿都凝聚了两位老师的心血。感谢何老师三年来学习上的督促和指导，生活上的关心和操劳，以及何老师事事处处为学生着想，一直给予我鼓励和勇气。感谢樊老师组织的讲座和讨论课堂，加深了我对专业领域的认识，开阔了视野。

其次，感谢各位指导过我老师，让我掌握了扎实的专业基础知识，也感谢中央民族大学理学院提供的优良的学习环境，感谢各位同学给予我的关心和帮助。

最后，感谢我的家人，谢谢你们一直照顾我、关心我，并给予我物质上的支持和精神上的帮助，使我最终完成学业。

真挚的感谢所有关心、帮助和支持我的老师、同学和家人！

攻读学位期间发表的学术论文目录

[1]王笑琨,马迪.基于形式概念的旅游资源开发分析.中央民族大学学报（自然科学版）,2013,(1):83-87

[2]王笑琨.基于嵌入式声音报警系统设计.中央民族大学学报（自然科学版）,2013增刊

基于形式概念分析的聚焦爬虫算法

作者：王笑琨
学位授予单位：中央民族大学