****

本科毕业设计（论文）

**题目：基于主题模型的文档关键词抽取算法研究**

**学 院 软件学院**

**专 业 软件工程卓越班**

**学生姓名 杨楷**

**学生学号 201130634480**

**指导教师 蔡毅**

**提交日期 2014 年 5 月 23 日**

摘要

关键词是文档主要内容的概要，它是一种快速理解文档主题的重要方法。关键词自动抽取技术可以为文档产生相应的关键词，免去了人工标注花费的巨大人力成本。当前关键词提取比较流行的方法是基于PageRank[8]的关键词提取算法TextRank[5]。该算法通过将文档转化为单词图来描绘句子词语之间的关系，再通过PageRank算法为每个节点求出一个值，选择Top-N个词作为该文档的关键词。另一种TextRank算法[10]中还提到，单词图中每个节点都是可以有权重的，代表着该文档对某一词语的偏好。我们可以为TextRank中每个节点增加一个权重。恰当的为节点增加权重，会提高TextRank算法的效果。

TextRank算法也有不足，它只根据句子结构抽取关键词，在句子结构中较为重要的词会更容易被选中为关键词，而这种做法忽略了句子中每个词的语义重要性。从我们经验上看，句子中的关键词不仅应该在句子结构中重要性强，而且应该在语义层面上有更强的性质。我们通常使用隐含主题模型[1]来分析文档中单词的语义。我们受到启发利用主题模型来为每个词赋予一个主题分布，根据这个分布为节点增加权重。这种算法我们称为基于主题模型的图算法（简称TPR）。在提出这个算法的过程中，我们实现了传统主题模型，并对其主题聚类结果进行分析。结果是我们对传统主题模型的效果不满意，于是在它的基础上修改得到基于背景词的主题模型，经过验证该模型对于关键词分类场景效果明显。我们发现采用基于背景词的隐含主题模型方法将主题聚类的结果应用到TextRank中可以提升该算法的效果。具体做法是针对词图里面每一个词的主题分布进行加权，采用加权后的值作为词图里面节点的权值，再采用TextRank算法算出词图中重要性较高的词作为关键词。

最后，我们分析了基于主题模型的图算法的不足与缺陷，提出引入信息熵[11]的概念对每个词加权，降低混淆度较高的词对关键词抽取效果的影响。为了解决上述问题，我们提出了基于信息熵的主题模型图算法，即ETP算法。实验结果表明，TPR算法与ETP算法相对于传统算法有明显提升。

**关键词**：关键词抽取；主题模型；随机游走；信息熵；

Abstract

. Keyword is a summary of the main content of the document, it is an important method to quickly understand the theme of the document. Automatic Keyword Extraction technology can generate the appropriate keywords for the document, eliminating manual tagging takes enormous human cost. Current keyword extraction method is more popular keyword extraction algorithm based on PageRank TextRank. The algorithm will document into word map to depict the relationship between words in a sentence, and then a value obtained by PageRank algorithm for each node, select the Top-N words as keywords for the document. TextRank algorithm also mentioned the word figure each node can have the right weight, which represents the preferences of a word in the document. We can add a weight to each node TextRank. Appropriate for the node to increase the weight, the effect will increase TextRank algorithm.

TextRank algorithm inspired theme we use models to assign a theme profile for each word, and then increase the distribution of node weights. This algorithm is called graph algorithm model based on the theme. In the proposed algorithm in the process, we realized the traditional theme of the model, and analyze the topic clustering results. The result is the effect of the traditional themes of our model are not satisfied, then modify it to get the topic on the basis of the model based on the background of the word, validated the model for keyword classification scene effect. We found that using the implicit theme background word model based approach would result Clustering topic TextRank can be applied to enhance the effect of the algorithm. This is done for the word graph which every word relating to the distribution of weight, the use of value weighted word graph as weights inside the node, then using TextRank word graph algorithm calculates the higher importance of words as keywords.

Finally, we analyze the topic model based on graph algorithm of deficiencies and defects, introduced the concept of entropy weight each word, reduce the impact of a higher degree of confusing word pairs keyword extraction effect. To solve the problem, we propose a topic model based on information entropy graph algorithms, namely ETP algorithm.

**Keyword :** Automatic Keyword Extraction, Topic Model, Information Entropy, PageRank

目录

[摘要 1](#_Toc419571227)

[Abstract 4](#_Toc419571228)

[第一章 绪论 1](#_Toc419571229)

[1.1 引言 1](#_Toc419571230)

[1.2 研究背景及意义 1](#_Toc419571231)

[1.3 国内外研究成果 3](#_Toc419571232)

[1.4 本文的研究内容 4](#_Toc419571233)

[1.5 本文的组织结构 5](#_Toc419571234)

[第二章 基础知识介绍 7](#_Toc419571235)

[2.1 隐含主题模型 7](#_Toc419571236)

[2.2 吉布斯采样(Gibbs Sampling) 9](#_Toc419571237)

[2.3 PageRank算法 9](#_Toc419571238)

[2.4 信息熵 11](#_Toc419571239)

[2.5 本章小结 12](#_Toc419571240)

[第三章 爬虫的编写与语料库的构建 13](#_Toc419571241)

[3.1 网络爬虫介绍 13](#_Toc419571242)

[3.2 语料库的构成及语料的预处理 13](#_Toc419571243)

[3.3 本章小结 14](#_Toc419571244)

[第四章 利用隐含主题模型建模 15](#_Toc419571245)

[4.1 隐含主题模型 15](#_Toc419571246)

[4.2 隐含主题模型的不足 15](#_Toc419571247)

[4.3 bLDA：基于LDA的修改 16](#_Toc419571248)

[4.4 利用bLDA对语料库进行主题聚类 17](#_Toc419571249)

[4.5 本章小结 17](#_Toc419571250)

[第五章 基于图方法提取文档关键词 18](#_Toc419571251)

[5.1 基于主题模型的PageRank算法 18](#_Toc419571252)

[5.1.1 构建单词图 18](#_Toc419571253)

[5.1.2 利用TPR进行关键词抽取 19](#_Toc419571254)

[5.2 Topical PageRank的不足 19](#_Toc419571255)

[5.3 基于信息熵的PageRank算法 20](#_Toc419571256)

[5.4 本章小结 21](#_Toc419571257)

[第六章 实验设计和结果展示 22](#_Toc419571258)

[6.1 评价指标 22](#_Toc419571259)

[6.2 不同参数对于结果的影响 22](#_Toc419571260)

[6.2.1 滑动窗口大小W 23](#_Toc419571261)

[6.2.2 LDA主题个数K 23](#_Toc419571262)

[6.2.3 PageRank中阻尼系数 24](#_Toc419571263)

[6.3 方法对比 24](#_Toc419571264)

[6.4 本章小结 25](#_Toc419571265)

[结论 26](#_Toc419571266)

[参考文献 27](#_Toc419571267)

[致谢 29](#_Toc419571268)

1. 绪论
   1. 引言

文档关键词指的是一个或几个词可以简明地概括该文档的内容，让读者可以简单的了解整篇文章的概要。在信息量极大丰富的今天，自动化文档关键词提取(Automatic Keyword Extraction)已经成为了一个极具潜力的研究方向。将不同文档中具有概括性语义的关键词准确地提取出来是一个具有挑战性的任务，其在自然语言处理(Natural Language Processing)的众多领域中得到广泛的应用，比如文本挖掘(Textual Mining)，信息检索(Information Retrieval)等。在其他领域，比如金融领域，关键词抽取这一技术也是有广泛的商用市场的。它可以帮助决策者在这个信息爆炸的时代快速的浏览有用的金融类信息，以做出正确合理的决策。

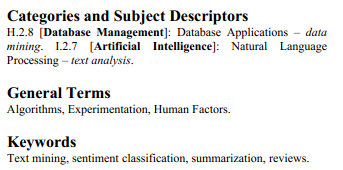
本文主要介绍了当前主流的关键词提取算法，以及介绍如何在实际语料中采用这种算法，并指出该算法的一些不足。本文在上述算法的基础上提出了一个基于该算法的改进算法，并对这两者进行分析比较。

* 1. 研究背景及意义

关键词是文档主要内容的概要，它是一种快速理解文档主题的重要方法。在各个地方都可以看到关键词的身影，比如我们在新闻网站上可以看到每篇新闻的标签，我们在浏览科技论文时可以看到该论文所讨论的关键词。它降低了人们在海量信息中搜寻信息的难度。

当前关键词已被应用在各个领域。在信息检索领域，关键词的应用非常广泛，百度、谷歌等搜索引擎公司都基于网页文本的关键词进行检索，基于文档关键词搜寻出来的结果往往是用户希望得到的。而在社交网络领域，当前许多功能以及相关研究都是基于用户所标注的标签(tags)实现的。用户标签方便用户对标注对象进行管理、收藏和检索，还可以利用这个为用户定制个性化信息推荐。通过提供用户对感兴趣对象（如图片、文章、视频等）的标注功能，利用用户群体智慧(Collective Intelligence)，我们可以获得一大批已标注的文档，对我们的研究工作提供了数据支撑。

图1.1展示了关键词的不同应用。图1.1(a)为某篇学术论文的关键词，该关键词是作者自己为该论文标注的，目的是标明该论文的主要内容以及让读者更容易地理解文章主题。该作者提供了三种不同类型的关键词：1) 分类和主题的描述(Categories and Subject Descriptions)，说明了该论文的领域 2) 通用术语(Ceneral Terms)，也是启到论文说明论文类型的作用 3) 关键词(Keywords)，展示了该文章的主题。图1.1(b)是国内IT类社区CSDN中某篇技术类文章的截图。从该文章中的标签（“算法”，“function”，“string”，“delete”，“数据结构”）我们可以在不看正文的条件下，迅速了解到该文章写的是关于字符串处理采用的算法与数据结构的相关内容，而该文章所描述的确是一种字符串处理算法。关键词技术能够帮助我们迅速浏览一篇文章或者一篇论文的主题，从而帮助我们迅速筛选信息，以得到我们最需要的资讯。



1. 某学术论文关键词



1. CSDN某文章

图1-1 关键词示例：(a)为某学术论文关键词 (b)为CSDN上某篇技术类文章的标签。

关键词在各个领域中应用广泛，一般的生成方法有以下三个：1) 由用户自发产生，用户自己标注感兴趣的内容 2) 请专家对文档进行手工标注关键词 3) 采用自动化文档关键词提取技术提取关键词。由用户自发标注标签的做法适用的场景较少，用户只对特定感兴趣的对象进行标注，当前还无法提供一种有效的模式激励用户标注其他内容。而因为当前信息技术高速发展，互联网上信息量也爆发式的增长，新内容无时无刻在产生，请专家对相关文档进行手工标注成本过高，其标注的文档也只能做研究用，而很难进行商业用途。所以当前对于自动化文档关键词提取技术的需求是迫切的，相关问题的研究也是当前的热点。

综上所述，关键词在信息技术飞速发展的今天，对各个领域的意义重大，它是人们检索信息的重要手段，又是社交网络研究中特征抽取的基础，还是人们对海量信息进行分类整理的重要工具。关键词自动抽取技术可以为文档产生相应的关键词，免去了人工标注花费的巨大人力成本。本文将重点探讨自动化文档关键词提取技术及如何对其进行优化。

* 1. 国内外研究成果

自动化文档关键词提取主要有以下两个步骤：1) 候选关键词抽取 2) 关键词加权。候选关键词抽取指的是从文档中找到符合需要的词语作为关键词的候选词。关键词加权是对之前选出的候选词进行加权处理，可以认为权值越大的候选词越能代表它所处的文档。当前已有很多针对自动化文档关键词提取的相关研究。Danilevsky[2]提出了一种关键词提取策略，先利用主题模型LDA[1]进行聚类，由聚类结果生成候选词，然后根据候选词的覆盖率(Coverage)，纯净度(Purity)，短语性(Phraseness)，完整性(Completeness)等指标为该词赋予权值。根据该权值的大小选择Top-N个词作为该文档的关键词。

Mihalcea[5]提出的TextRank算法将文档中每个词视为一个点，词与词之间的联系视为边构建单词图。词和词之间的联系是通过一个滑动窗口实现的，同在一个窗口内的词相互间会有边的联系。接着利用PageRank[8]算法对该单词图进行权重计算，并根据权重排序得出Top-N个词作为文档的关键词。该算法得到比较好的效果，因为该该算法生成的单词图考虑了有序的单词序列中词和词之前的共现关系，换句话说，它把句子的结构考虑了进去。

Tomokiyo [4]针对长文本使用基于概率的语言统计模型提取关键短语。其主要方法是对不同的语言模型采用K-V分歧（KL-divergence）计算短语的信息丰富程度。而[12]专注于短文本的主题提取，将文本-主题关系使用矩阵建模，并采用矩阵分解（matrix factorization）的方法，得到每个词隐含的主题。

* 1. 本文的研究内容

本文研究如何应用主题模型进行自动化关键词抽取。比如一个简单的文档“至少20名央企高管巡视中落马，央企成为腐败易发地”，凭借我们经验，关键词应该是“央企高管”，“落马”，“腐败”，因为这两个词最能代表改句子的语义内容。为了达到该效果，我们可以先通过利用大量语料进行训练得出一个适用性较强的模型，然后通过该模型对文档中每个词指定一个主题，根据该主题根句子中主题的相关程度判断该词是否是关键词。在上述例子中，“央企高管”，“腐败”，“落马”这三个词相比其他词出现在同一个主题的概率更大，所以对该文档更具代表性。这里有个问题，简单的主题模型并不能完全适用我们当前的研究。简单的主题模型比如LDA对于我们的语料分类效果不佳，经常错误地将不相关的词聚为同一个话题，比如上述例子中得“成为”一词，由于其适用于大多数语境，所以这种词有可能会被划分到“腐败”那一个主题中，对结果造成干扰。于是我们在简单的主题模型的基础上又提出了另外一种主题模型bLDA，可以预先指定一系列背景词到一个主题中，这些背景词一般都像“成为”这种词，适用于大部分语境，而没有明显的主题特征。如此处理可以有效地避免背景词对于非背景词的干扰。

然而单纯采用主题模型抽取关键词还是具有一定的局限性，因为它是基于词袋模型的方法。词袋模型即将每个文档视为一个无词序的单词集合，它只考虑该文档中出现了哪些词，而不考虑单词在文档中的顺序。将文档用建模为词图模型，可以将文档中单词的顺序考虑进来，单词的顺序本质上蕴含了句子结构的相关信息。因此我们在主题模型的基础上，引入了词图模型，并对此进行相关研究。之前已经有相关论文是研究将隐式主题模型和词图模型结合提取关键词，我们将会先介绍其中一种效果较好的方法(Topical PageRank)，然后分析其还尚待改进的地方。接下来我们会针对该方法的不足，提出另外一种基于信息熵(Entropy)的改进算法(Entropy Topical PageRank)。信息熵本质上指的是信息的混淆程度，如果某个词的信息熵很大，那这个词很可能在各个文档中频繁出现，那么这种词很可能是主题不明确的词，在最后赋予其权重时应该降低该词的值。基于上述假设，我们的基于信息熵的改进方法从理论上说是可以相比之前的方法获得提升的，因为我们不仅考虑了句子的结构，还考虑了句子结构内部的内在关系。

* 1. 本文的组织结构

本文将阐述关键词提取的相关算法及其意义，分析这些算法的优缺点，有何种改进措施。我们也会介绍我们如何对数据进行预处理，以适应我们的算法，让算法达到最好的效果。

第一章阐明了我们为何进行该项研究，以及该研究的重要意义，也阐述了我们的研究内容，并介绍了当前国内外对于该领域的研究成果。最后还针对该研究提出了我们的想法和方法。

第二章介绍了与本文相关基础知识，为下文的介绍铺平道路。我们重点对主题模型，TextRank，信息熵进行介绍。

第三章描述了我们对语料库的文档进行预处理的相关方法。本章介绍了我们所采用相关分词工具以及相关预处理方法。

第四章介绍了我们采用隐含主题模型为文档中每个词产生一个主题分布的过程。其中还介绍了一种主题模型的改进方法，增加分词的准确度。

第五章介绍了使用第四章中每个词的主题分布，结合词图模型，采用Topical TextRank算法来为每个词赋予权重的过程。还提出了另外一种基于信息熵的Entropy Topical TextRank算法作为改进。

第六章完成了对上述提到的几种算法TextRank ，Topical TextRank和Entropy Topical TextRank的实验和论证，并且对比这三种方法。在这一章我们结合之前构建的语料库和我们的研究目标设计实验，阐述实验步骤，展示实验效果。

最后是本文的总结和展望，针对上述实验结果，指出我们尚待改进的地方以及未来工作的方向。

1. 基础知识介绍
   1. 隐含主题模型

传统判断文档相似性的方法是基于两个文档的统计学相似性，比如TFID。这种方法没办法考虑到语义层面上的相似性。Blei[1]在2003年提出了隐含狄利克雷分配模型(Latent Dirichlet Allocation, LDA)很好地解决了这个问题。在主题模型中，主题代表的是一个概念，一个相关的方面(Aspect)，或者是一系列相关的词。而在隐含狄利克雷分配模型中，每个词的主题值就是改模型一个隐含变量，需要通过观测值来进行推断。LDA是一个词袋模型，每个文档里面的词都是无序序列，就像是放到一个袋子里。LDA又属于生成模型(Generative model)，它探讨了一个文档是如何被生成的，如何使用随机算法生成一个文档。LDA生成过程可以简单理解为以下过程：

1. 现在有两个袋子，每个袋子装的是不同类型的骰子，第一个袋子装的是doc-topic骰子，第二个袋子装的是topic-word骰子。
2. 先随机从第二个袋子中抽取K个topic-word骰子，编号1到K。
3. 为了生成一篇新的文档，先随机从一个袋子中抽取一个doc-topic骰子，然后重复骰子该骰子，直到生成该文档的所有单词，并为每个单词生成一个topic，编号t；然后重复上述过程，为文档中每个词指定一个topic编号，此时词还没有生成。
4. 对于c)步骤为每个词指定的topic编号t，选择b)步骤中编号为t的骰子，投掷这个骰子得到的单词就是对应的word。

以上就对LDA模型下每篇文档中每个词的生成过程一个比较形象的描述。上述骰子有两种，doc-topic骰子表示的是一篇文档中主题的分布，topic-word骰子表示一个主题里面单词的分布。袋子中骰子也是有一种分布，这种分布一般采用狄利克雷分布(Difichlet Distribution)。如果我们已知上述分布，那么根据生成模型可以随机生成多篇文档。而现实中，我们往往是不知道上述分布，却可以在语料库中观测得到多篇文档，我们根据生成模型以及贝叶斯推断，最大似然估计可以得到对隐含变量topic值的推断。LDA的物理过程如图2-1。

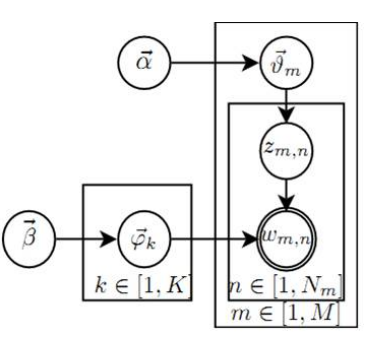


图2-1 LDA的物理模型

模型中总共有M篇文档，有K个主题。代表着doc-topic分布，即文档中的主题分布。代表着topic-word分布，即主题中每个词的分布。和分别是和的狄利克雷先验分布(Dirichlet prior)。先验参数和都由我们事先指定，它们对最终结果都有相当大的影响。图2.1物理过程图用的是盘子记法(plate notation)，里面每个圆代表着一个变量，用两个圆盘框住的变量代表我们可观测的变量。

图2.1可以分解为以下两个主要的物理过程：

1. ，该过程表明，在生成第m篇文档的时候，先从狄利克雷先验中抽取一个分布，再从该分布中抽取一个词，从而生成第m篇文档中第n个词的主题。对应的是狄利克雷分布，对应的是多项分布(Multinomial Distribution)，所以整体是一个狄利克雷-多项分布共轭结构(Dirichlet-Multinomial conjugate)。根据共轭结构的特点，可以得到以下公式：

|  |  |
| --- | --- |
|  | (1) |

1. ，该过程表示，在a步骤已抽取到主题的条件下，生成预料中的m篇文档的第n个词。先从狄利克雷先验中抽取一个topic-word分布，根据这个分布再进行抽样，获得。同样的，也是一个狄利克雷-多项分布共轭结构，所以可以得到以下公式：

|  |  |
| --- | --- |
|  | (2) |

根据上述两个公式，我们可以得到：

|  |  |
| --- | --- |
|  | (3) |

* 1. 吉布斯采样(Gibbs Sampling)

Griﬃths[9]提出了吉布斯采样法训练LDA模型。LDA提出者Blei[1]在其论文中用的是边分推断法(Variational Bayesian)学习LDA，而由于其效率以及内存占用量过高的缺点使得该方法运用不广泛。而Griffiths提出的吉布斯采样法让蒙特卡洛马尔可夫链(Markov-chain Monte Carlo ,MCMC)收敛从而完成LDA的学习过程。Gibbs Sampling算法过程如下：

1. 随机初始化 ，
2. 对循环采样

根据吉布斯采样法，结合章节2.1中推导出来的公式(3)，我们可以得到LDA模型的Gibbs Sampling公式：

|  |  |
| --- | --- |
|  | (4) |

我们根据上述公式(4)计算出每个词对于每个主题的概率分布，然后根据这个概率分布进行随机取样，取样的结果作为该词的主题。重复对文档中的每个词迭代上述过程，直到最后各个词的分布收敛，此时根据每个词的主题分布可以对该词进行分类。

* 1. PageRank算法

当前关键词抽取一般有两种思路，一种是将关键词抽取当作是二分类，对文档中每个词都给予一个二值变量，判断其是否是关键词。这种方法需要使用大量已标注关键词的语料进行训练。而要获得标注关键词的语料需要花费相当大的成本。另一种思路就是使用无监督(Unsupervised)的方法，将文档中的单词进行排序，选择Top-K个单词出来作为该文档的关键词。传统的方法有TFIDF等基于词频的方法。当某个词在文档中出现的概率很高，即词频(TF)很高时，该词对于该篇文档的重要程度就相应较高。而当某个词只在某几个文档集中时，反向文档频率很高，说明该词只在几个文档中被用到，该词对于那些文档重要程度较高。如果只考虑TF那很可能会出现将一些常用的，而对该文档内容重要性不大的词考虑进来，比如一些停用词或者是一些常用形容词，比如“快”，“好”，“坏”等。这时候就需要将IDF考虑进来，常用的词虽然TF很高，但因为其IDF很低，整个TFIDF值将会降低。TFIDF就是将TF和IDF相乘，得到的值的数学意义是，一个词如果在一个文档中出现频率较高，而在其他文档出现的频率不高，那么这个词对于该文档重要程度较大，可以作为关键词。但是TFIDF缺点也是很明显，有些重要的词其实词频并不是很高，单纯以词频来衡量一个词的重要性明显是不行的。TFIDF值较高的词都是词频较高，但是文档频率较低的，而我们在实际应用中通过已标记的文档关键词，我们可以观察到，较多的文档关键词在改文档中的词频并不高，从而导致其TFIDF值较低，无法选中为关键词。为了克服这种误差，又有一种无监督的关键词抽取算法提出来，这种算法成为TextRank。该算法是将一种网页很流行的网页排名算法PageRank[5]算法应用到文本挖掘中的方法。

PageRank算法是最先由谷歌提出并使用的一种算法，根据网页中得链出链接构建一个网页图，再根据形成的网页拓补图计算得出每个网页的影响力。它普遍被应用到网页评价领域，评价一个网页是否重要。PageRank算法应用了如下两个假设：数量假设和质量假设。数量假设指的是如果一个网页的入链数量越多，那这个网页在网络中越重要。质量假设指的是链入某个页面的网页越重要，那这个网页在网络中也相应的越重要。利用这两个假设，可以得到以下的计算PageRank步骤：

1. 构建网络图，初始化每个节点的PageRank值。
2. 不停的迭代并计算PageRank值，每个页面的PageRank值等于其链入的众多页面分配给它的值的总和。其链入页面根据其链出到该页面的链接占总链接比例分配值。 为页面总链出数，而为页面到的链出数。我们可以得到以下公式计算PageRank值。

|  |  |
| --- | --- |
|  | (5) |

上述公式(5)中是阻尼系数(damping factor)，取值范围是[0,1]，表示网页图中节点个数。一个节点不一定有出链，有些节点出链为0。这种节点孤立于其他网页，为了使这种网页能被访问到，就需要阻尼系数来对PageRank进行修正，是每个节点跳到另外一个节点的概率值。不断重复上述步骤，每个页面的PageRank值会趋向于正常和稳定，因为该算法最终将会收敛，而这个收敛值便是PageRank值。

Rada Mihalcea和Paul Tarau提出了一种基于PageRank的关键词提取算法TextRank。该算法通过将文档转化为单词图来描绘句子词语之间的关系，再通过PageRank算法为每个节点求出一个值，选择Top-N个词作为该文档的关键词。将文档转换为单词图的步骤如下：

1. 选取一个移动窗口长度W
2. 将文档开始部分在移动窗口中的单词作为图的节点，移动窗口中的一个词与其他词对应的节点都以边的形式关联。
3. 向文档结尾的方向移动移动窗口，并执行b步骤，直到移动窗口到达文档末尾。

TextRank算法中还提到，单词图中每个节点都是可以有权重的，代表着该文档对某一词语的偏好。当把权重考虑到该算法中时，TextRank中每个词的值的计算公式得有所改变：

|  |  |
| --- | --- |
|  | (6) |

TextRank算法中每个词语的权重值可以利用LDA进行计算得出，将LDA与TextRank算法结合提取关键词的方法在后续章节会详细介绍。

* 1. 信息熵

1984年，Shannon[11]将热力学领域的熵(entropy)引入信息论(information theory)中。热力学熵和香农的信息熵的基本物理意义、测量方法、数学形式完全一致，不存在本质的区别。但两者的应用领域不一样，热力学熵主要用来热机或者是内燃机，而信息熵主要是用来研究通讯或者是计算机等人工领域。信息熵内在含义是数据的混淆度。信息熵的值越大，代表数据混淆度越高。信息熵可以用来衡量一个概率分布，其值越大代表概率分布越分散，而其值越小代表概率分布越集中。熵的公式可以表示成：

|  |  |
| --- | --- |
|  | (7) |
|  |  |

上述式子(7)中H(X)代表随机变量X的熵的值。P(x)代表x的出现概率。对于信息熵，可以举个例子。当我们想预测明天的天气，如果能100%确定明天一定下雨，那其熵就为-1\*log1=0，也就是说不确定性为零。如果能50%确定明天是阴天，50%概率晴天，那熵就是-0.5log0.5-0.5log0.5=(-0.5)(-2)+(-0.5)(-2)=1。如果说明天有25%的机会下雨，25概率晴天，25%概率下雪，25%概率阴天，那熵就是4\*(-0.25)(log0.25)=2。也就是说随着不确定性增加，熵的值也在增加。信息熵中之所以采用概率的对数计算，是因为一个数的对数是这个数在某个值下的拆分。比如一段信息有8bit，在计算机中用 个位来记录。一般来说在熵的计算公式中，采用自然对数ln的效果会更好。信息熵可以用来判断一个分布是否是集中的，如果一个分布的熵比较小，则认为这个分布是集中的。LDA模型产生的结果是每个词对于主题的分布，如果一个分布的信息熵很小，则认为该词被分到少数几个主题中，混淆度较低，效果较好。而如果一个分布的信息熵很大，那我们认为该词被分到大多数的主题中，效果较差。关于信息熵的更加深入讨论在后续章节会继续。

* 1. 本章小结

本章介绍了隐含狄利克雷分配模型的原理和物理解释，又介绍了吉布斯采样法作为LDA的采样方法，完成了对LDA学习过程的介绍。本章还介绍了谷歌网页排名中应用到的一种关键算法PageRank，又称网页排序算法。又从PageRank发引申到TextRank算法，该算法通常关键词提取。最后又介绍了信息熵相关知识，在后续章节中，我们会将信息熵应用到关键词抽取算法中。

2. 语料库的构建及预处理
   1. 语料库介绍

为了验证我们的方法，我们必须做一系列对比实验。在实验中，我们使用Hulth[13]论文中的数据集来构建我们的语料库。这个数据集包含了2000篇论文的摘要和19254个手动标注的关键词。在本实验中，我们采用这些手动标注的关键词和我们提取的关键词作为对比，下一小节我们将介绍一系列指标来反映我们算法的效果。

* 1. 语料的预处理

利用上一小节构建的语料库，我们对语料库中的文章进行了如下的预处理：单词词性划分，功能词/停用词剔除，词干提取（stemming）。词性划分可以使用Stanford POS-tagger ，它是是斯坦福大学自然语言处理小组（Natural Language Processing Group）实现并公布的分词器。使用Stanford POS-tagger对数据进行分词，并划分其词性效果比较理想。在构建中心短语的时候，得用到各个主题词的词性来作为构建的参考，所以在预处理时也对各个词的词性进行标注。在得到各个词的词性后，根据词性移除句子中的停用词，介词等意义无关的词，只留下名词、形容词、动词，减少主题词构建的时候的噪声。词干提取是去除句子中英文单词词态的影响，比如‘dog’和‘dogs’应该算作同个词。词干提取我们采用WordNet来完成。WordNet是由普林斯顿大学教授George Armitage Miller指导下建立和维护的英语字典，它可以根据单词的意义对其进行分组。当前WordNet提供了程序调用接口，我们使用它来将有词态的单词还原。

* 1. 本章小结

本章介绍了我们如何获取语料资源和如何对获取的语料进行预处理。本章完成的内容是我们整个研究的基础，只有先获得数据，以及对数据进行相应的预处理，才能继续我们的研究工作。

2. 利用隐含主题模型建模
   1. 隐含主题模型

我们在章节2.1已经详细介绍过一种隐含主题模型，隐含狄利克雷分配模型。本章介绍如何利用该模型完成对文档中单词的主题聚类。我们将LDA应用到我们预处理过的语料库，训练出语料库中每个词的主题分布。训练出来的主题分布可以应用到下一章节的基于图算法提取关键词中。我们也可以通过计算某个词的主题分布和该文档的主题分布相似度，给予每个词一个值，再取top-K个值对应的单词作为该文档的关键词。这种方法在论文[1]中已经研究过，并且取得不错的效果。

* 1. 隐含主题模型的不足

传统LDA必须事先指定主题数量，即K值，因为在吉布斯采样的过程中是根据k值来为每个词指定一个主题的。如果K值设置不恰当，会使主题聚类结果出现混淆了不同语义的词语。当前自动获取K值的算法还在研究中，本论文没有采用该类型的算法，而是采用手动调整K值，通过观察主题聚类结果判断聚类结果的。

LDA是根据词袋模型，一篇文档中每个词是认为从一个无序的单词集合中不断随机抽取得到的。这种方式忽略了句子中词与词之间的关系，而现实生活中一个句子中不同词之间并不是毫无关系的，而且一个句子中单词的组成顺序对一个句子是有影响的。举个例子，“目前数据挖掘在各行各业应用广泛,而在金融领域也有越来越多的相关研究”。该例子中，“应用广泛”形容的是“数据挖掘”，而“越来越多的相关研究”形容的是“金融领域”而不是“数据挖掘”，这说明词与词之前是有关系的，每个词之间的关系也是不一样的。我们观察上述例子可以得出，词与词之间的相关性很多情况下是基于其在句子中的位置的，句子位置中相邻近的词关系较强，比如“金融领域”和“越来越多的相关研究”这两者的关系。

由于传统LDA是无监督类算法，而其算法的本质在于词语的共现频率，经常出现在同一个文档的词语更有可能被分配到同个主题中。这种无监督类方法会出现的问题是，文档中那些模糊词，比如“好”，“坏”等，它们可以分别修饰不同的语义的名词，从而使得不同语义的词更有可能被聚类到同一个主题中，导致结果的可读性很差。为了去除这种影响，我们可以采用下一小节的方法，bLDA。

* 1. bLDA：基于LDA的修改

本文重点介绍一个基于LDA的修改：bLDA。bLDA与传统LDA不同之处在于增加了一个额外的背景主题 z = 0。这个基于上一小节中提出的原因，文档中信息量较低的词往往是比较模糊的，这种词对于聚类效果是有影响的。将文档中模糊的词放到背景主题中这种方法比较简单易行，对模型的改动相对较小。采用这种相对简单的模型而不是其他复杂模型可以让我们将更多时间放在对候选词加权的效果提升上，而这部分是本文主要研究内容以及贡献。

bLDA利用前景主题(Foreground Topics) z = 1,…,k 和背景主题 z=0来对文档进行建模。对于文档中每个词，我们先确定这个词是前景词还是背景词，然后从相应的topic-word分布中选择一个词出来。

我们让代表主题t = 0,…,k的词频分布。让代表文档d的主题分布。让代表从所有主题中选择背景主题还是前景主题的伯努利分布(Bernoulli Distriution)。bLDA的生成过程如下：

1. 刻画出分布：，对于每个t = 0,…,k。
2. 对于每个文档d D，
3. 刻画分布 。
4. 对于d中所有位置i的词：
5. 刻画分布
6. 如果 = 0，刻画，如果不是则执行以下内容：
7. 从分布中选择一个主题 z
8. 从多项分布中选择一个单词

其中和是分布与分布的狄利克雷先验分布。我们使用吉布斯采样法进行模型推断。我们迭代地对于每篇文档中的每个词的采样其主题。在某些研究中，他们重点研究的是每个词在某篇文档中被指配到的主题，而在我们的研究里，我们更加重视的是每个词对应的主题分布，该分布可以使用章节2.2的公式(3)求出。具体做法是针对每个主题t计算一个概率。我们使用最大后验概率(maximum a posteriori,MAP)的方法对每个词的出现频率进行标记：。

* 1. 利用bLDA对语料库进行主题聚类

本小节主要介绍如何利用bLDA对语料库进行主题聚类。bLDA主要方法在上一小节中已经描述了。而在实际操作的时候需要关注的一个点就是背景主题词的设定。我们探讨如何确定哪些词在句子中信息量较少，并且容易对模型聚类效果有影响。获取背景词最简单的方法就是通过阅读语料库，以人工的方式从中挑选那些表意不明，主题倾向不明确的词出来，将其加入到背景词中。这种方法思路简单，但是实现起来成本较高，当语料很大时，这种方法耗时太长。另一种方法是通过某种排序算法近似的将信息量较小的词选取出来，再通过人工的方式剔除掉分的不准的词。这种方法相对来说节约成本，本论文采用的就是这种方法。我们通过计算每个词的TFIDF值，并选定某个阀值，将低于这个阀值的单词都认为是信息量较小的词。然后再人工浏览这些词，挑选出语义上有些主题倾向的词语剔除掉。将剩下的词作为背景词引入LDA中进行学习。

* 1. 本章小结

本章介绍了我们如何使用隐含主题模型进行建模。最开始我们分析介绍了传统LDA的优缺点，表明传统LDA并不十分适用于本文的研究中。接着我们介绍了一种用于改进传统LDA的模型：bLDA。针对bLDA背景词的选用方法，我们也进行了讨论。在本章中我们重点在于介绍bLDA，因为它是本研究中采用的主题聚类算法，并且在实际中获得了比较好的效果。

2. 基于图方法提取文档关键词
   1. 基于主题模型的PageRank算法

章节2.3中介绍了PageRank算法，而本章将介绍一个PageRank算法的改进，基于主题模型的PageRank算法。在论文[8]中介绍了这种方法。该方法通过两个步骤进行：

1. 构建一个主题模型来得到单词和文档的主题
2. 执行TPR方法，从文档中抽取关键词

第一步已经在第四章详细介绍了，本章关注的内容是如何使用TPR进行关键词抽取。对于给定的文档d，使用TPR算法对关键词进行抽取通过以下四个步骤进行：

1. 将文档d中单词用图模型建模，生成一个单词图。这个单词图代表的是单词之间在结构上的联系。
2. 将TPR算法应用在单词图上，使用主题模型得到的主题分布，对每个单词在不同主题上计算一个PageRank值。
3. 对于不同主题，将单词按照PageRank值大小进行排序
4. 根据文档的主题分布，得到该文档中每个主题的重要程度。结合该重要程度，对候选词进行最终排序，选取前M个词作为关键词。
   * 1. 构建单词图

为了使单词图保存尽可能多的句子结构信息，我们采用一个滑动窗口的方式来构建单词图。文档中的单词看作是一个有序的序列，这样的做法能够体现每个词在句子中的位置信息。文档中得每个词作为单词图中的每一个节点。我们采用一个长度为W的滑动窗口，从文档开始往文档尾部移动，在移动过程中不断向图中增加边。TextRank论文中提到词图是否有向对其算法效果没有太大影响。我们通过移动窗口构建一个有向图，每个滑动窗口的第一个单词指向其他的单词。与论文[5]不同的是，我们在构建单词图的时候将标点符号考虑进去，因为实际生活中往往每个句子中的词相关性较大，特别是每个分句（通过逗号隔开的分句）的词语之间相关程度较大，往往有修饰与被修饰的关系。所以在移动窗口移动过程中将标点符号的因素考虑进来，我们可以通过增加不同的边的权重值来体现这一关系。如果两个词在同一个分句中，那其增加的边的权重设为2，而如果两词不在同一个分句中，其增加的边的权重设为1。我们在构建单词图的时候，只考虑形容词和名词，因为关键词一般都是名词。

* + 1. 利用TPR进行关键词抽取

TPR根据主题模型得到的主题分布，根据这个分布确定单词图中每个单词的权值。利用每个词的权值和公式(6)可以为每个词计算一个PageRank值。在TPR中，对于每篇文档，我们针对每一个主题计算每个词和该主题的相关程度作为其权值。我们对(6)稍加改进，得到下式：

|  |  |
| --- | --- |
|  | (8) |

其中，，即给定单词w的条件下，该词被分配为主题z的概率，这个也反映了单词w有多大程度上属于主题z。对于每篇文档，我们能根据K个主题画出K个不同权重的单词图，然后应用上述公式(8)，计算得到文档中每个词都包含K个PageRank值。对于每个词，我们需要将其在K个不同主题下的K个PageRank值处理为一个PageRank值。p(z|d)为文档d属于主题z的概率，对于每个单词的K个PageRank值，我们可以采用如下公式将它们合并：

|  |  |
| --- | --- |
|  | (9) |
|  |  |

TPR是一种迭代算法，需要为该算法定义一个终止条件。我们规定当迭代次数达到300或者相邻两次迭代中每个词的TR值小于0.001则终止该算法，输出算法结果。根据公式(9)每个单词计算出来的PageRank值，我们对其进行排序，然后选取词序最高的M个词作为该文档的关键词。

* 1. Topical PageRank的不足

TPR算法实质上是先分别针对文档中每个主题计算得出一个关键词重要性排序，再根据文档的总体主题倾向得出一个最终的根据重要性排序的关键词列表。这种方法的前提是主题模型对文档中每个词的主题聚类是准确的，而且一个主题就代表着一个语义上一致的单词集合。这种是理想情况下才会发生，在实际的语料分析过程中，每篇文章中词语的组合是千变万化的，而同一个语义又可以有多种表达方式，所以用主题模型聚出来的同个主题的单词之间往往不是语义一致的。我们称同个主题中语义不与其他词一致的词为‘杂质词’，这种词对TPR算法影响是很大的。同个杂质词往往会出现在多个主题中，会对TPR算法在针对某个主题计算该主题的关键词时造成影响。杂质词并不同于章节4.3中介绍的bLDA中背景词，背景词指的是语义上有模糊的词，无法确定其应该属于哪个主题，而‘杂质词’是可以分辨出其大体的主题的，但是该词混杂在一个其他主题里面。举个例子，一个‘数据挖掘’领域的主题，它的主题词有‘文本挖掘’，‘信息检索’，‘机器学习’，‘中枢神经系统’这四个。其中‘文本挖掘’，‘信息检索’，‘机器学习’都是数据挖掘领域的名词，但是‘中枢神经系统’从经验上看显然是生物学领域的。出现这种情况原因有许多，有一种可能是因为生物学领域词语在该语料库中包含的较少，缺乏和它组合在一起形成一个新主题的词，所以被误分类到其他主题中去。由于我们的主题模型是一种聚类模型，我们无法控制其生成结果每个主题分别具有哪些语义，所以该问题较难解决。我们可以通过不断增加主题数量K的值，然后观察其聚类结果，直到找到一个较好的划分结果。不过这种方法局限性也是很大，因为我们无法控制聚出来的类别是什么。由于‘杂质词’的干扰，TPR模型在计算文档中主题t的单词权重或者是文档d的主题权重时会受到影响，使得结果的TPR值会有偏差。下一章节，我们提出一种方法解决这一问题。

* 1. 基于信息熵的PageRank算法

为了克服由于主题模型聚类不准确产生的问题，我们对TPR算法进行改进，得到一个基于信息熵的PageRank模型，我们称为Entropy Topical PageRank(ETP)。为了使ETP效果更好，我们必须减少‘杂质词’对于我们算法效果的影响。一种合理的方式是在计算过程中降低‘杂质词’的权重。所以我们得设计出一个方法将主题模型的主题聚类结果中的‘杂质词’识别出来。之前提到过，‘杂质词’有个特点就是分布在众多主题中，没有集中分布在某几个主题。所以利用章节2.4中提到的信息熵的方法，根据词的主题分布求其信息熵，信息熵越大的词其主题分布越均匀，即混淆度越高，而信息熵越小代表其主题分布越集中，混淆度越低。文档中词对于其主题分布的信息熵计算公式如下：

|  |  |
| --- | --- |
|  | (10) |

计算每个词的信息熵，在TPR计算每个词的权重时加上熵的影响。修改公式(9)，得到如下公式：

|  |  |
| --- | --- |
|  | (11) |

根据公式(11)，当是‘杂质词’时，将会很大，因而将会变得很小，即该词的权重将会变得很小，在PageRank算法过程中被访问到的概率降低，从而最终该词的PageRank值降低。是一个平滑因子，可以控制信息熵的值对最终PageRank值的影响幅度。

在TPR基础上添加信息熵的方法直接效果就是让主题模型中聚类不准确的词在词图中权重降低。而还有其他方法可以提取出‘杂质词’，比如计算主题中每个词和其他词的互信息的和，并选择互信息较小的词作为‘杂质词’降低其在TPR计算中的权重。本论文只讨论采用信息熵的方法，而暂不讨论基于互信息的方法。

* 1. 本章小结

本章主要介绍了如何使用基于主题模型的图方法抽取关键词。首先我们介绍了基于主题模型的PageRank算法，并从该算法实现步骤的角度深入展示了该算法。我们构建了单词图并且使用相关计算公式，迭代的运行该算法直到达成终止条件。接着我们探讨了TPR算法的不足以及原因。针对这种不足，我们在该算法的基础上再引入信息熵的概念，提出了另一种算法ETP，完成对该算法的改进。

2. 实验设计和结果展示
   1. 数据集

我们使用Hulth[13]论文中的数据集来构建我们的语料库。这个数据集包含了2000篇论文的摘要和19254个手动标注的关键词。在本实验中，我们采用这些手动标注的关键词和我们提取的关键词作为对比，下一小节我们将介绍一系列指标来反映我们算法的效果。

* 1. 评价指标

本实验采用准确率(Precision)，召回率(Recall)和F值(-Measure)作为评价算法的评价标准。准确率和召回率是信息检索和统计分类等领域中重要的评价指标，用于评价结果的质量。一般来说，准确率是指预测的条目和实际的条目有多少条是一致的，而召回率是指实际的条目中有多少条是被预测正确的。然而准确率和召回率有时候会出现矛盾的情况。比如在极端的情况下面，我们只预测了一个结果，且是正确的，那么正确率就是100%，但是召回率就很低；而如果我们把所有结果都返回，那么召回率将是100%，但是准确率就会很低。要综合考虑它们，就要用到F值了。它是综合准确率和召回率的一个指标，一般用F值来作为评价一个算法的标准比较可靠，其计算方法如下：

|  |  |
| --- | --- |
|  | (12) |

公式(12)中，F指的是F值，Pr代表的是准确率，Re代表的是召回率。

* 1. 不同参数对于结果的影响

ETP中有以下四个参数可能对关键词抽取的效果产生影响。

1. 滑动窗口大小W
2. LDA主题个数K
3. PageRank中阻尼系数
   * 1. 滑动窗口大小W

我们设置ETP的参数W从5到20之间变化，结果如表6.1所示。观察结果发现W的变化对于算法效果有一定的影响，但是影响不大，当W选值较大时，算法F值较高，效果较好。

表6-1 滑动窗口大小对结果的影响

|  |  |  |  |
| --- | --- | --- | --- |
| W | 准确率 | 召回率 | F值 |
| 5 | 0.288 | 0.344 | 0.313 |
| 10 | 0.29 | 0.352 | 0.318 |
| 15 | 0.288 | 0.357 | 0.319 |
| 20 | 0.294 | 0.361 | 0.324 |

* + 1. LDA主题个数K

主题个数K不同，主题聚类聚出来的主题品质是不同的，而这个会对ETP的效果有一定影响。我们使用准确率-召回率曲线来观察K对关键词抽取效果的影响，曲线越靠近原点，效果越差。我们选取K等于50，100，500，1000四个值，关键词推荐个数M的值从1到20变化，生成一幅准确率-召回率曲线，如图6-1所示。

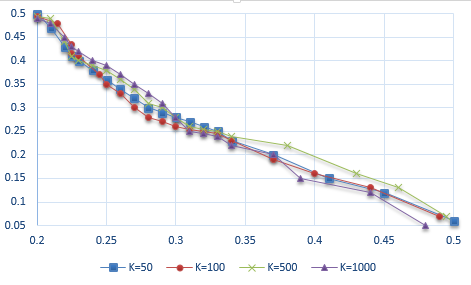


图6-1 主题个数K对结果的影响

* + 1. PageRank中阻尼系数

ETP中的阻尼系数会影响到随机游走算法的效果，因为它决定了随机游走过程中随机跳转的概率，越大，随机跳转概率越大，因而导致单词在每个主题中该词权重所占的比例更大了。图6-2展示了当 = 0.1,0.3,0.5,0.7时，关键词推荐个数M值从1到20的时候，F值的变化情况。从图中可以直观的了解到，当值在0.3附近的时候，F值最高，效果最好。

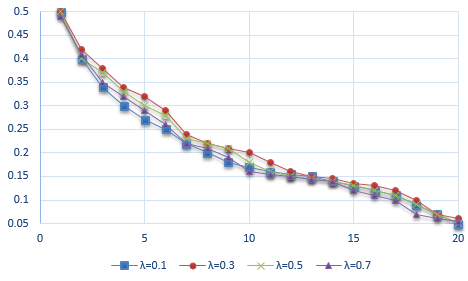


图6-2 阻尼系数对结果的影响

* 1. 方法对比

本文选取四种方法进行实验对比，TFIDF，TextRank，Topical PageRank和Entropy Topical PageRank。我们选取关键词推荐个数M值从1到20，使用这4种不同的方法对比其实验结果，得到图6-3所示准确率-召回率曲线。从该曲线可以看出，ETP效果要稍微好于TPR，TextRank效果大部分M值下略低于TFIDF，而这ETP与TPR效果都是明显好于TFIDF与TextRank的。

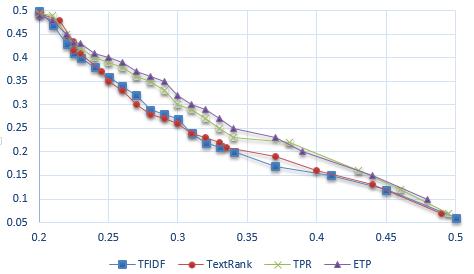


图6-3 不同方法效果对比

* 1. 本章小结

本章针对我们之前提出来的方法进行实验验证其效果，我们先介绍了本论文采用的效果评价指标，接着针对我们提出的模型中的参数进行实验，确定其最佳参数范围。接着将本文提到的四种方法进行实验对比，得出的结论是本文所提出的方法效果好于原有的方法，证实了本研究的意义。

结论

在信息量极大丰富的今天，自动化文档关键词提取已经成为了一个极具潜力的研究方向。文档关键词指的是一个或几个词可以简明地概括该文档的内容，让读者可以简单的了解整篇文章的概要。当前关键词提取比较流行的方法是基于PageRank的关键词提取算法TextRank。该算法通过将文档转化为单词图来描绘句子词语之间的关系，再通过PageRank算法为每个节点求出一个值，选择Top-N个词作为该文档的关键词。TextRank算法中还提到，单词图中每个节点都是可以有权重的，代表着该文档对某一词语的偏好。我们可以为TextRank中每个节点增加一个权重。恰当的为节点增加权重，会提高TextRank算法的效果。于是这个启发了我们利用主题模型来为每个词赋予一个主题分布，然后根据这个分布为节点增加权重。这种算法称为基于主题模型的图算法。在提出这个算法的过程中，我们对传统主题模型的效果不满意，最终采用基于背景词的隐含主题模型方法来提高主题聚类的效果，即TPR算法。最后，我们分析了基于主题模型的图算法的不足与缺陷，提出引入信息熵的概念对每个词加权，降低混淆度较高的词对关键词抽取效果的影响。为了解决上述问题，我们提出了基于信息熵的主题模型图算法，即ETP算法。最后，我们采用已完成关键词标记的新浪财经新闻作为语料库，对本文提出的方法与原有的方法进行对比实验，得出的结论是本文提出的方法效果优于传统的关键词提取算法，这也证明了本研究的意义。

参考文献

1. Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. the Journal of machine Learning research, 2003, 3: 993-1022.
2. Danilevsky M, Wang C, Desai N, et al. Automatic construction and ranking of topical keyphrases on collections of short documents[C]. SDM, 2014.
3. Zhao W X, Jiang J, He J, et al. Topical keyphrase extraction from Twitter[C]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics, 2011: 379-388.
4. Tomokiyo T, Hurst M. A language model approach to keyphrase extraction[C]//Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment-Volume 18. Association for Computational Linguistics, 2003: 33-40.
5. Mihalcea R, Tarau P. TextRank: Bringing order into texts[C]. Association for Computational Linguistics, 2004.
6. Newman D, Asuncion A, Smyth P, et al. Distributed Algorithms for Topic Models. Journal of Machine Learning Research, 2009, 10:1801–1828.
7. Liu Z, Huang W, Zheng Y, et al. Automatic keyphrase extraction via topic decomposition[C]//Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2010: 366-376.
8. Page L, Brin S, Motwani R, et al. The PageRank citation ranking: Bringing order to the web[J]. 1999.
9. Griffiths T. Gibbs sampling in the generative model of latent dirichlet allocation[J]. 2002.
10. Wu W, Zhang B, Ostendorf M. Automatic generation of personalized annotation tags for twitter users[C]//Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics. Association for Computational Linguistics, 2010: 689-692.
11. Shannon, Claude E. (July–October 1948). "A Mathematical Theory of Communication". Bell System Technical Journal 27 (3): 379–423.
12. Yan X, Guo J, Liu S, et al. Learning topics in short texts by non-negative matrix factorization on term correlation matrix[C]//Proceedings of the SIAM International Conference on Data Mining. 2013.
13. Hulth A. Improved automatic keyword extraction given more linguistic knowledge[C]//Proceedings of the 2003 conference on Empirical methods in natural language processing. Association for Computational Linguistics, 2003: 216-223.

致谢

首先感谢我的导师蔡毅老师。本论文是在蔡老师的指导下和同学们的帮助下修改完成的。在此，我要向他们的细心帮助和指导表示由衷的感谢。在这段时间里，我从蔡老师身上不仅学到了许多的专业知识，更感受到他们工作中的兢兢业业，生活中的平易近人。此外，蔡老师严谨的治学态度和忘我的工作精神值得我去学习。

非常感谢实验室的大家在我的毕业设计中，给予我极大的帮助，使我对整个毕业设计的思路有了总体的把握，并耐心的帮我解决了许多实际问题，使我有了很大的收获。同时，他们在整个过程中提出了许多建设性意见，并给我解决了一些专业性问题。

还要感谢我的母校——华南理工大学，感谢多年来传授我知识的老师们，感谢母校为我的成长付出的一切！

总之，在以后的学习生活中我将以加倍的努力对给予我帮助的学校、老师及同学们的回报。