(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 109902289 B
(45)授权公告日 2022.12.13

(21)申请号 201910062048.8

(22)申请日 2019.01.23

(65)同一申请的已公布的文献号
申请公布号 CN 109902289 A

(43)申请公布日 2019.06.18

(73)专利权人 汕头大学
地址 515000 广东省汕头市大学路243号

(72)发明人 姜大志 黄志均 曾文信 黄瑞香 漆原

(74)专利代理机构 广州三环专利商标代理有限公司 44202
专利代理师 周增元 曹江

(51)Int.Cl.
G06F 40/289(2020.01)
G06F 40/30(2020.01)
G06F 40/279(2020.01)
G10L 13/02(2013.01)
G10L 13/08(2013.01)
G10L 15/04(2013.01)
G10L 15/26(2006.01)

审查员 赵芳

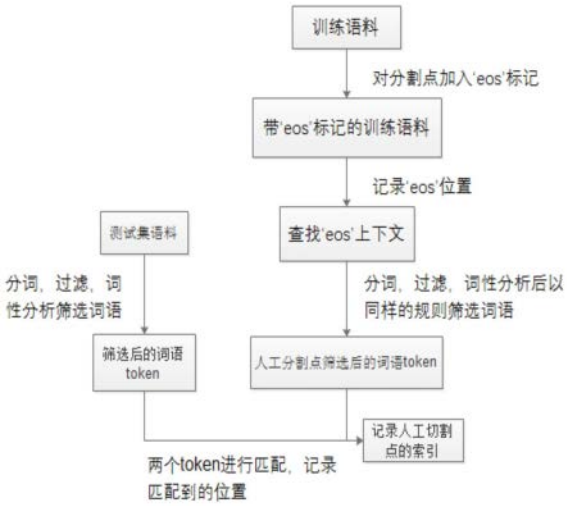
权利要求书2页 说明书8页 附图4页

(54)发明名称

一种面向模糊文本挖掘的新闻视频主题分割方法

(57)摘要

本发明实施例公开了一种面向模糊文本挖掘的新闻视频主题分割方法，包括步骤：将新闻视频转换成音频，使用语音识别技术将音频转换成模糊文本；使用语音识别技术将音频转换成模糊文本；文本的分词；文本的表示；分析音频信息，找出静音点作为潜在分割点；对模糊文本进行分割点识别，更新潜在分割点；把词性分析后筛选出的词语序列，用语言模型转换成句子向量，并根据句子向量的相似度分布来更新潜在分割点；基于PSO确定TextTiling算法参数进行文本主题分割。采用本发明，把视频信息以故事为单位进行语义分割从而形成独立的语义单元，可应用于众多视频检索的应用领域，如影视、监控、交通视频等。



CN 109902289 B

1.一种面向模糊文本挖掘的新闻视频主题分割方法,其特征在于,包括以下步骤:

S1:将新闻转换为音频;

S2:将所述音频转换成模糊文本;

S3:添加用户词典,根据所述用户词典与待分析的语料库,以逆向匹配的分词方式对文本使用字符串匹配进行分词;

S4:使用Filter方法计算得到特征项的子集,并根据权重来判断该特征项对文本的重要程度,

S5:分析音频信息,找出静音点作为潜在分割点;

S6:使用应用TextTiling算法的词性分析和命名实体分析方法确定保留和过滤词语来实现相关的计算;

S7:将所述词性分析后筛选出的词语序列,用语言模型转换成句子向量,并根据句子向量的相似度分布来更新潜在分割点;

S8:使用粒子群优化算法确定所述TextTiling算法参数进行文本主题分割。

2.根据权利要求1所述的面向模糊文本挖掘的新闻视频主题分割方法,其特征在于,所述步骤S3前还包括步骤:针对所述模糊文本的数据清洗。

3.根据权利要求2所述的面向模糊文本挖掘的新闻视频主题分割方法,其特征在于,所述步骤S3的字符串匹配是通过扫描Trie树实现。

4.根据权利要求2所述的面向模糊文本挖掘的新闻视频主题分割方法,其特征在于,所述步骤S5具体包括在使用阈值判断方法后得到初始分割点,之后使用贝叶斯信息准则对初次的初始分割点进行第二次分割。

5.根据权利要求4所述的面向模糊文本挖掘的新闻视频主题分割方法,其特征在于,所述步骤S7还包括根据文本中的特征词,分析文本中词语序列的词性,之后根据词性分析的结果进行相似度的计算。

6.根据权利要求5所述的面向模糊文本挖掘的新闻视频主题分割方法,其特征在于,所述相似度的计算包括采用tf-idf方法对文本信息向量化,

参数tf为在选取的语料库中的主题中出现的次数,参数idf代表的是使用语料库中总的主题数目除以当前词出现过的主题的数目,最后再取对数值,其中,

$$\mathrm{tf}_{i,j}=\frac{n_{i,j}}{\sum_k n_{k,j}},\quad idf_i=\log\frac{|D|}{|\{j:t_i\in d_j\}|}$$

,其中,$n_{i,j}$是当前词在文本d_j中的出现次数,分母表示的是文本d_j中所有词语出现的次数之和,|D|表示的是整个语料库中文档的个数,分母$|\{j:t_i\in d_j\}|$表示的是在整个语料当中包含ti词语的文档的个数,最终的tf-idf值为$\mathrm{tf}_{i,j}\times\mathrm{idf}_i$。

7.根据权利要求1-6任一项所述的面向模糊文本挖掘的新闻视频主题分割方法,其特征在于,所述步骤S8具体包括:

S81:模型初始化,定义词性分析所得到的词语序列得到的长度为word_size,伪句子长度为K,块的大小为W,初始化参数K,W,其中K取值范围为[1,word_size/2],W的取值范围是[1,word_size/k],W取值范围里面的K值为当前K的取值;

S82:计算适应度值,重新划分过滤之后的词语序列,对划分好的块使用TextTiling算

法进行相似度计算，根据相似度的分布情况，寻找极值点，根据极小值点确定对应的分割点，分割点对应着词语序列的索引；

S83：更新种群个体，在计算了种群中所有个体的适应度值之后，记录种群中取得最优值的个体的K，W值，根据当前最优和种群最优的粒子坐标进行更新个体的坐标，再进行适应度的计算。

8.根据权利要求7所述的面向模糊文本挖掘的新闻视频主题分割方法，其特征在于，所述步骤S8还包括：使用TextTiling算法对文本当中的词语序列重新进行句子的划分，生成伪句子，之后根据伪句子和块的大小计算块与块之间的相似度。

一种面向模糊文本挖掘的新闻视频主题分割方法

技术领域

[0001] 本发明涉及人工智能领域，尤其涉及一种面向模糊文本挖掘的新闻视频主题分割方法。

背景技术

[0002] 随着互联网、电子信息以及通信技术的飞速发展，各个行业与领域均积累了海量的数据，但是数据量的急剧增大给有效的信息检索带来了极大的挑战。新闻视频是多媒体传播信息的一种重要的途径，其中的视频信息中包含了丰富和生动的语义信息，但由于新闻视频一般缺乏良好的结构组织以及索引，获取某一方面的新闻需要全篇浏览视频，无法对过往的视频进行检索。如果需要查询某一个主题的新闻视频，则要重新浏览新闻视频，这对于人力、物力以及时间方面都是一个极大的消耗。而通过对视频的检索，可以快速的获取对应的信息，减少阅读和查找的时间，提高搜索效率，因此如何实现高效的新闻视频搜索已经成为了一个研究的热点。

[0003] 目前，可知的研究中对新闻视频主题分割方面的研究和发明有了一段的积累。新闻视频的主题分割技术从信息类型上主要有3种，分别是基于视频镜头的主题分割、基于声学特征的主题分割和基于文本信息的主题分割。

[0004] 新加坡的L Chaisorn在《A Hierarchical Multi-Modal approach to story segmentation in news video》(新闻视频中故事分割的分层多模态方法)论文中提出的故事分割方法，将镜头分成十三个类别，根据类别确定故事边界，以达到新闻视频主题分割的目的。

[0005] 刘华咏在《基于音视频特征和文字信息自动分段新闻故事》论文中通过静音检测和字幕检测来实现新闻主题分割。

[0006] 余骁捷等人在《新闻播报节目的内容自动标注系统》中通过语音识别技术实现了基于语义的主题分割模型。

[0007] 凌坚等人在论文《新闻单元的自动快速分割方法》以及刘群等人在论文《采用多特征融合的镜头边界检测方法》中，根据图片的颜色分布情况及其特征，分析出视频中的说话人和说话人所处的环境变化，进而对视频中的场景进行分类，实现了新闻视频的基于图像之间相邻帧的相似度变化的主题分割。

[0008] 对于上述新闻视频主题分割技术出现的调节效果不理想的问题，可能是多方面的影响，如设备、算法、实现模式、作用模式等。

[0009] 从Hearst在《Segmenting Text Into Multi-Paragraph Subtopic Passages》论文中提出TextTiling算法，利用文本块之间的差异性来实现文本主题的分割。由此我们可以得出文本信息和新闻视频主题两者之间的相互关联。因而我们可以通过设计两者之间的关系模型实现基于基于模糊文本挖掘的新闻视频主题分割技术。

[0010] 其中，对于文本主题分割的实现，目前国内已有大量的研究和专利，并且方案均比较成熟。例如，钟彬彬等人在论文《基于GA的文本子主题切分中的参数优化研究》中提出了

用遗传算法来优化TextTiling算法中的参数的主题分割方法，在中文文本中的主题分割模型中取得了较好的效果。

[0011] 由以上可知，目前对于“新闻视频的主题分割”问题在技术手段上已经有了较大的突破。

[0012] 上述现有技术中存在有以下缺陷：

[0013] 1、基于语音分析的新闻视频分割技术，无法对同一主持人播报不同新闻的情况进行很好的主题变化的切割。

[0014] 2、基于语义理解的分词方法，要求严格，难以得到很好的实现与运用，仍处于试验阶段。

发明内容

[0015] 本发明实施例所要解决的技术问题在于，提供一种面向模糊文本挖掘的新闻视频主题分割方法。可将视频进行有效的主题分割，把视频信息以故事为单位进行语义分割从而形成独立的语义单元，可应用于众多视频检索的应用领域。

[0016] 为了解决上述技术问题，本发明实施例提供了一种面向模糊文本挖掘的新闻视频主题分割方法，包括以下步骤：

[0017] S1：将新闻转换为音频；

[0018] S2：将所述音频转换成模糊文本；

[0019] S3：添加用户词典，根据所述用户词典与待分析的语料库，以逆向匹配的分词方式对文本使用字符串匹配进行分词；

[0020] S4：使用Filter方法计算得到特征项的子集，并根据权重来判断该特征项对文本的重要程度，

[0021] S5：分析音频信息，找出静音点作为潜在分割点；

[0022] S6：使用应用TextTiling算法的词性分析和命名实体分析方法确定保留和过滤词语来实现相关的计算；

[0023] S7：将所述词性分析后筛选出的词语序列，用语言模型转换成句子向量，并根据句子向量的相似度分布来更新潜在分割点；

[0024] S8：使用粒子群优化算法确定所述TextTiling算法参数进行文本主题分割。

[0025] 进一步地，所述步骤S3前还包括步骤：针对所述模糊文本的数据清洗。

[0026] 更进一步地，所述步骤S3的字符串匹配是通过扫描Trie树实现。

[0027] 更进一步地，所述步骤S5具体包括在使用阈值判断方法后得到初始分割点，之后使用贝叶斯信息准则对初次的初始分割点进行第二次分割。

[0028] 更进一步地，所述步骤S7还包括根据文本中的特征词，分析文本中词语序列的词性，之后根据词性分析的结果进行相似度的计算。

[0029] 更进一步地，所述相似度的计算包括采用tf-idf方法对文本信息向量化，

[0030] 参数tf为在选取的语料库中的主题中出现的次数，参数idf代表的是使用语料库中总的主题数目除以当前词出现过的主题的数目，最后再取对数值，其中，

$tf_{i,j}=\frac{n_{i,j}}{\sum_k n_{k,j}}$，$idf_i=\log\frac{|D|}{|\{j:t_i\in d_j\}|}$，其中，$n_{i,j}$是当前词在文本$d_j$中的出现次数，分母表示的是文本$d_j$中所有词语出现的次数之和，|D|表示的是整个语料库中文档的个数，分母$|\{j:t_i\in d_j\}|$表示的是在整个语料当中包含ti词语的文档的个数，最终的tf-idf值为$tf_{i,j}\times idf_i$。

[0031] 更进一步地，所述步骤S8具体包括：

[0032] S81:模型初始化，定义词性分析所得到的词语序列得到的长度为word_size，伪句子长度为K，块的大小为W，初始化参数K，W，其中K取值范围为[1,word_size/2]，W的取值范围是[1,word_size/k]，W取值范围里面的K值为当前K的取值；

[0033] S82:计算适应度值，重新划分过滤之后的词语序列，对划分好的块使用TextTiling算法进行相似度计算，根据相似度的分布情况，寻找极值点，根据极小值点确定对应的分割点，分割点对应着词语序列的索引；

[0034] S83:更新种群个体，在计算了种群中所有个体的适应度值之后，记录种群中取得最优值的个体的K，W值，根据当前最优和种群最优的粒子坐标进行更新个体的坐标，再进行适应度的计算。

[0035] 更进一步地，所述步骤S8还包括:使用TextTiling算法对文本当中的词语序列重新进行句子的划分，生成伪句子，之后根据伪句子和块的大小计算块与块之间的相似度。

[0036] 实施本发明实施例，具有如下有益效果:发明通过词语词性分析的方法对向量化之前的源数据进行过滤和筛选，并实验证明使用了该方法之后相同主题的文本相似度会增加，而不同主题之间的文本相似度会降低，使得主题之间的区分度更好。还提出了基于语言模型和PSO算法的TextTiling模型，通过实验验证发现主题分割的准确率有一定的提升。

附图说明

[0037] 图1是人工分割点流程图；

[0038] 图2是相似度计算过程示意图；

[0039] 图3是相似度计算过程示意图；

[0040] 图4是TextTiling算法流程图；

[0041] 图5是PSO算法流程图；

[0042] 图6是文本主题分割算法流程图。

具体实施方式

[0043] 为使本发明的目的、技术方案和优点更加清楚，下面将结合附图对本发明作进一步地详细描述。

[0044] 本发明实施例的一种面向模糊文本挖掘的新闻视频主题分割方法，通过下述方法进行。

[0045] 1、新闻视频转换成音频。

[0046] 使用FFmpeg将新闻视频转换成音频文件。

[0047] 2、使用语音识别技术将音频转换成模糊文本。

[0048] 使用语音识别系统把语音识别成对应的文本信息，并进一步对文档进行转换，转换成程序可以识别的文本格式和字符。

[0049] 3、针对模糊文本的数据清洗。

[0050] 去除重复的数值、数据表中的空值和空白格、转换数据格式。然后，通过分析新闻文本的特征，把对主题分割没有作用的一部分过滤，比如新闻联播中前面有一部分是对整篇新闻进行概括的，就需要在主题分割算法分析之前识别出来，对这一部分的文字内容进行过滤，提高主题分割的准确性。对一些无意义的内容、停用词等，以及对文本分析影响不大的文本信息进行了过滤，最终构成一篇语义表达相对清晰的文本，为适应后续的分词和相似度的计算最好准备。

[0051] 4、文本的分词。

[0052] 添加用户词典，根据这些词典与待分析的语料库，以逆向匹配的分词方式对文本，对文本使用字符串匹配的分词方法。此匹配的方法是通过扫描Trie树实现。Trie树也就是单词查找树用空间上内存的消耗来减小时间上的消耗，利用Trie树可以缩短查找所需的时间增大查询效率，减少对无效的字符串之间的查询和比较。Trie树里面保存的字符串和对应的值，跟hashmap类似保存的是key值和value的值，出现词频较高的词在Trie树中距离树的root位置越近，查找时从根节点往下查找，检索对应的子树，直到该节点为被标记的结束节点则停止检索，输出的词就是有效的分词结果。检索输出所有可能的分词之后，利用动态规划的方法找出最优可能的分词结果。

[0053] 5、文本的表示。

[0054] 文本表示的方法需要能反映出文本的真实的内容，也要让表示的内容和其他的文本表示出的内容有所区别。使用Filter方法来计算得到特征项的子集。根据特征数据集的属性性质，而不是某个特定的学习函数来得到特征子集，并根据权重来判断该特征项对文本的重要程度。

[0055] 其中，Filter方法是对每一个特征项都赋予一定的权重，然后根据权重来决定该特征项对文本的重要程度。而计算权重的方法也有很多，比如根据信息增益、相关系数、卡方检验等等来计算特征项的权重。特种方法省去了使用学习器的训练过程，比较具有一般性和通用性，复杂程度也比较低，可以快速过滤大量的不相关的特征项。

[0056] 权值的计算方法就是如果文本当中存在该特征项，则该特征项的权值为1，否则权值为0。词频权值法是统计该特征项在所分析的文本当中总共的出现的频次，出现多少次，则权重就为多少。权值的大小就代表了该特征项在文本当中的重要性。TF-IDF是一种使用比较广泛的权值计算方法，TF表示词语在文档中出现的频次，IDF表示该特征词在所有的文本集当中出现的频次。并且与特征词在文本中出现的次数成正比，与特征词在其他文本中出现的次数成反比。而这种方法可以保留比较重要的特征，过滤一些常用的不重要的词，这种方法正好符合需求。TFC权值法和TF-IDF是类似的，知识对文本的长度进行了一个归一化处理，核心思想还是TF-IDF。

[0057] 6、分析音频信息，找出静音点作为潜在分割点。

[0058] 语音主题分割方法可分为两大类，一类是根据语音时域上的振幅大小来判断是否为静音，另一类是利用时域上的信号信息转换成频域上的频谱信息，当频谱的能量信息小于某一个阈值的时候，则判定为静音部分。本实施例在使用阈值判断方法后得到初始分割

点,之后使用BIC(Bayesian Information Criterion),即贝叶斯信息准则对初次的静音分割点进行第二次分割。

[0059] 7、文本的词性分析与命名实体识别。

[0060] TextTiling算法的主要意图是用来弱化原本实施例本中句子的概念,重新定义句子长度,使得每句话表达的信息量一样,然后计算句子块与块之间的相似度,通过相似度之间的关系和一定的分割策略确定主题之间分割的界限。句子块的长度可由人工凭经验设定或者通过策略找到较优的参数,而组成计算相似度的句子块的构成是原本实施例档中的词语,考虑到不同词可能会对主题分割的影响不一样,所以本实施例提出了基于词性分析的TextTiling算法,分析文本中词语的词性在主题分割当中的影响,根据分析的结果对句子块中的词语进行筛选或者加权值的方法来实现相似度的计算,并分析切割边界。使用伪句子计算相似度,构成句子的词语的过滤和选择在一定程度上对相似度的计算有影响,本实施例使用词性分析和命名实体分析来确定保留和过滤词语来实现相关的计算。

[0061] 8、把词性分析后筛选出的词语序列,用语言模型转换成句子向量,并根据句子向量的相似度分布来更新潜在分割点。

[0062] 根据文本中的特征词,分析文本中词语序列的词性,之后根据词性分析的结果进行相似度的计算,相似度计算的基础是将文本信息向量化,本实施例使用的方法为tf-idf,其中tf为term frequency的缩写,其取值代表的是当前词语在整篇文档中出现的次数,在本实施例中代表的是在选取的语料库中的四个主题中出现的次数,一般在计算的过程中还需要对该词频数值进行归一化,防止文档主题过长,则会导致词频的数值比较大,从而不能体现出词语在不同主题之间的重要程度。具体的计算公式如式(1)。

[0063] idf为inverse document frequency的缩写,其取值代表的是逆向文档频率,能很好的表示词语区分主题的能力。如果当前词语在其他主题中出现的次数越少,idf的取值也就越大,表示该词越能代表当前主题的语义,计算方式为文档的总数目除以包含当前词语的文档的数目,再对结果取对数,对应本实施例中的idf的计算则是使用语料库中总的主题数目除以当前词出现过的主题的数目,最后再取对数值。具体的计算公式如式(2)。

[0064] $$\mathrm{tf}_{i,j}=\frac{n_{i,j}}{\sum_k n_{k,j}} \quad (1)$$

[0065] $$idf_i=\log\frac{|D|}{|\{j:t_i\in d_j\}|} \quad (2)$$

[0066] 公式(1)中的$n_{i,j}$是当前词在文本d_j中的出现次数,分母表示的是文本d_j中所有词语出现的次数之和。公式(7-2)中log里的|D|表示的是整个语料库中文档的个数,分母|{j:$t_i\in d_j$}|表示的是在整个语料当中包含t_i词语的文档的个数。最终的tf-idf值为公式(1)与公式(2)结果的乘积。

[0067] 9、基于PSO确定TextTiling算法参数进行文本主题分割

[0068] TextTiling算法的主要思想是计算文本块之间的相似度,根据相似度结果之间的关系,制定出分割策略进行主题的分割。

[0069] 在使用TextTiling算法时,其中最重要的步骤就是在确定参数值伪句子大小和块大小之后的相似度计算。本实施例提出了基于语言特征的词性分析的方法对语音识别后的

文本信息进行分析，根据分词后词语的词性分布情况进行相应的过滤和筛选，最后选择最优的词语分布作为文档的词语分布。

[0070] 根据最优的词语分布，按照由PSO生成的最优参数伪句子和块大小进行相似度的计算，得到整篇文档的一个相似度分布情况，最后根据相似度的分布情况，以一定的分割策略对分割点进行确定。具体过程如下：

[0071] (1)模型初始化

[0072] 定义词性分析所得到的词语序列得到的长度为word_size，伪句子长度为K，块的大小为W。按照上述分析的参数范围进行初始化参数K，W。其中K取值范围为[1,word_size/2]，W的取值范围是[1,word_size/k]，W取值范围里面的K值为当前K的取值。本实施例初始化生成200个个体，取值在该范围内的K，W的种群。迭代次数本实施例设置的是50。种群个体的大小和迭代次数越大，程序运行所花费的时间也越长，而在本实施例的实际应用中，迭代次数在50代以内适应度函数已经收敛到了最优值。

[0073] (2)计算适应度值

[0074] 在初始化K，W之后，重新划分过滤之后的词语序列，对划分好的块使用TextTiling算法进行相似度计算，根据相似度的分布情况，寻找极值点，根据极小值点确定对应的分割点，分割点对应着词语序列的索引。该索引就是算法对应的主题分割点，再结合人工分割点，计算适应度值。

[0075] 人工分割点的步骤，如图1所示，为对训练数据中人工分割点的位置加入标记，比如本实施例使用的分割标记为‘eos’，当遍历训练文档时，查找到出现‘eos’位置，并记录位置，查找人工切割点的上下文，根据上下文通过一定规则找出在翻译出的文本位置，也就是使用上下文的内容通过同样的词性分析过滤词语的方法得到token序列。这里的token序列是分词后，经过过滤和筛选后的词语。然后和原文本进行匹配，找到分割点的位置索引，即为人工分割点。

[0076] 本实施例使用F值为适应度函数的评估值，即根据人工主题分割点的索引和算法主题分割点的索引，计算准确率和召回率。根据准确率和召回率求出F值，即为适应度值。

[0077] (3)更新种群个体

[0078] 在计算了种群中所有个体的适应度值之后，记录种群中取得最优值的个体的K，W值。根据当前最优和种群最优的粒子坐标进行更新个体的坐标，即K，W值，再进行适应度的计算。

[0079] 根据上述三个步骤确定最优的参数取值之后，使用下述的TextTiling算法对文本当中的词语序列重新进行句子的划分，生成伪句子，之后根据伪句子和块的大小计算块与块之间的相似度。使用PSO算法初始化TextTiling算法的两个参数值K，W值，假设初始化的参数值K＝4，W＝2。首先把百度语音识别的新闻文本信息进行分词、过滤、词性分析筛选词语之后，按K值的大小把原来的token序列组成一个个伪句子，根据W大小划分一个个块，按照例子中参数的取值大小，也就是4个词组成一个伪句子，2个伪句子成成一个块，然后再计算块与块的相似度，两个块的大小就是一个滑动窗口的大小，每次滑动的步长为一个伪句子的大小，所以最后计算的相似度结果的个数是K-1+W-1。计算完整篇文档的相似度之后，按步长进行移动，步长大小为一个伪句子大小，重新重复上述步骤，重复的次数为W-1。K＝4，W＝2的情况如图2所示。K＝1，W＝4的情况如图3所示。

[0080] 计算出对应的K,W参数值,每个伪句子间隙处的相似度值,相似度值的个数由上文可知为K+W-2,为了获取每个相似度的值对应伪句子的位置,本实施例同一定义每个伪句子间隙都有一个对应的相似度值,这样相似度值的个数则为K-1,所以本实施例在计算完相似度值之后对相似度矩阵的前部添加0数组,大小为W-1。之后计算出相似度矩阵中的极小值点,记录极小值点所在的token序列的间隙,找到该间隙对应的词语的索引,记录为算法的切割点。具体的算法流程图如图4所示。

[0081] 由上述TextTiling算法的介绍可以知道对分割点影响较大的是伪句子的大小和块的大小,而原算法中是根据经验值进行确定的,并且英文的文本中伪句子和块的大小与中文文本相差还是比较大的,因此本实施例提出了使用粒子种群的方法进行参数的确定。

[0082] 粒子种群算法,下文将直接使用PSO进行表示,全称为Particle Swarm Optimization,即粒子种群的优化算法。最初是由鸟群蚁群鱼群觅食这个问题的解决演化出来的算法。与PSO类似的演化算法还有传统的遗传算法,模拟退火算法,devolution evolution算法,以及基因演化变异算法,都是先初始化种群,参数随机选取,以一个评价标准,即定义和使用一个适应度函数进行评估,然后通过不断的变化和迭代,直到取得最优的解。PSO算法与遗传算法相比,PSO算法没有使用生成的变量值之间进行变异或者交叉取值的操作,而是直接通过初始化的值找到当前这些值的最优解,这个最优解对于全局来说,其实是局部最优解,通过局部最优解逐步取得全局最优解。

[0083] PSO算法的详细步骤为,把问题的求解想象成鸟类觅食的问题。这里的鸟类就可以看做是粒子,每个粒子的取值带入定义的适应度函数可以求得对应的适应度值,然后根据局部最优值,粒子考虑周围粒子的适应度值向局部最优值的方向,以一定的方向和速度进行迭代,跟着当前的最优值去追寻全局最优解。在每一次的更新和迭代的过程当中,粒子通过本身当前的局部最优解pbest,和整个种群的最优值即全局最优值qbest来寻找接下来的最优解。PSO因为能简单的实现和描述出问题方法,也不需要调节很多的参数,常常被用来优化和确定参数,也被广泛的用在其他的遗传算法和神经网络领域。

[0084] PSO首先是初始化种群,生成一群随机的解,然后通过两极值pbest和qbest来不断的迭代生成值,即通过自身的位置以及两个极值点来更新位置,标准的PSO更新速度和位置的方法如式(5)和式(6):

[0085] $V_i = \omega \times V_i + c_1 \times rand() \times (pbest_i - x_i) + c_2 \times rand() \times (gbest_i - x_i)$ (5)

[0086] $x_i = x_i + V_i$ (6)

[0087] 其中ω为非负值,称为惯性因子,i=1,2,…,M,M是该群体中个体的总数;V_i是粒子个体的速度;pbest和gbest分别为当前种群最优个体和全体种群当中的最优个体;rand()是在[0,1]之间的随机数;X_i是粒子的当前位置。c_1和c_2是学习因子。

[0088] (4)相似度计算模型

[0089] 文本的相似度计算应用相当的广泛,包括通过计算需要检索的词与被检索词之间的相似度的信息检索、问答系统、文档聚类、文档分类、网页去重、防作弊等方面的应用。

[0090] 计算相似度的模型主要可以分为三类。

[0091] 第一类是基于向量空间模型的相似度计算。该模型是基于词与词之间是没有关联的一个假设,然后根据向量之间的相似度来计算文本之间的相似度。该模型根据分词后的结果,给每一个词分配权重,则这句话的向量就可以由分词后,每个词的权重组成了,之后

只需要计算这些权重向量之间的相似度就可以了。常见的权重W_i表示的是第i个词在该篇文档中出现的频率，也可以称作词频。第二类是基于集合模型的相似度计算。第三类是基于层次结构的相似度计算。

[0092] 本发明采用基于向量空间模型的相似度计算。

[0093] 常用的计算相似度的模型是基于统计的向量空间模型(Vector space model：VSM)其中每个词都是坐标系中的一个维度或者说是坐标系中一个坐标轴，词向量里的每一个特征项的权重代表其坐标值，如：文本d_i的词向量的集合表示为$\{t_1,t_2,t_3,\cdots\cdots,t_n\}$，其中词$t_j$就表示文本$d_i$的第j个特征词，文本$d_i$表示为一个n维的词向量，权值的一般是根据该词在文本中出现的频率用特定的函数计算，所以该文本d_i的n维向量也可以表示为$\{w_{i1},w_{i2},w_{i3},\cdots\cdots,w_{in}\}$。特征值的权重可用TF-IDF算法计算得出。把特征项的值t_1，$t_2,t_3,\cdots\cdots,t_n$看做坐标系中的n维坐标轴，这样就得到了该篇文本的向量空间图。两个文本的相似度就是计算着两个向量空间图的相似度。通常会采用余弦夹角的方法来计算向量空间的相似度，具体的公式如下：d_1,d_2分别表示两篇不同的文本，n表示文本中有n个词向量，w_i表示第i个特征项的权重。

[0094] $$sim(d_1,d_2)=\frac{d_1*d_2}{d_1\times d_2}=\frac{\sum_{i=1}^{n}w_{1i}\times w_{2i}}{\sqrt{\sum_{i=1}^{n}w_{1i}^2}\times\sqrt{\sum_{i=1}^{n}w_{2i}^2}} \quad (7)$$

[0095] 整个主题模型算法的基本流程图如图6所示。

[0096] 以上所揭露的仅为本发明一种较佳实施例而已，当然不能以此来限定本发明之权利范围，因此依本发明权利要求所作的等同变化，仍属本发明所涵盖的范围。

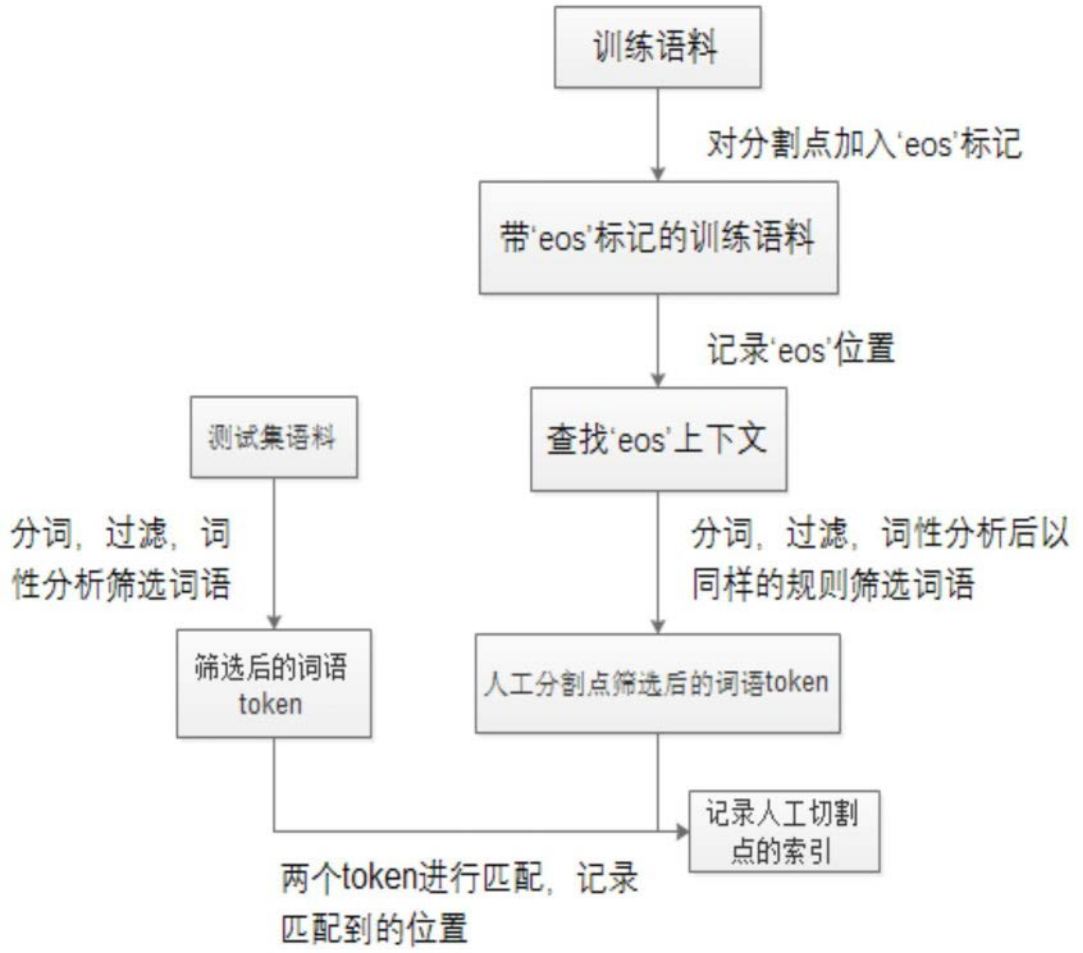

训练语料
对分割点加入'eos'标记
带'eos'标记的训练语料
记录'eos'位置
测试集语料
查找'eos'上下文
分词，过滤，词性分析筛选词语
分词，过滤，词性分析后以同样的规则筛选词语
筛选后的词语token
人工分割点筛选后的词语token
记录人工切割点的索引
两个token进行匹配，记录匹配到的位置

图1


W0W1W2W3
伪句子K1
W4W5W6W7
伪句子K2
W8W9W10W11
W12W12W13W14
W15.....
.....Wn
块W1
块W2

图2

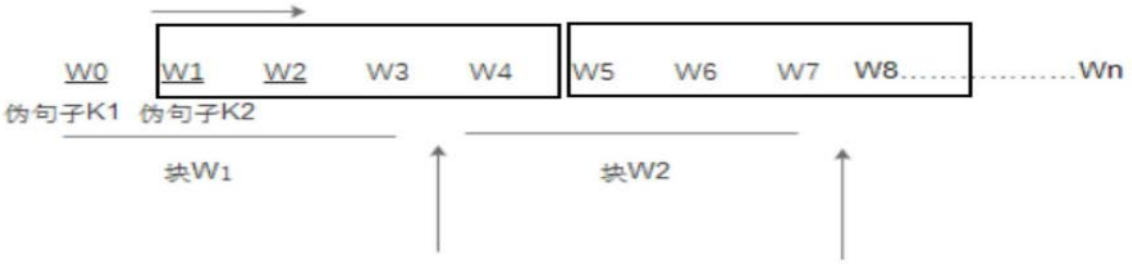

W0
W1
W2
W3
W4
W5
W6
W7
W8.....Wn
伪句子K1
伪句子K2
块W1
块W2

图3

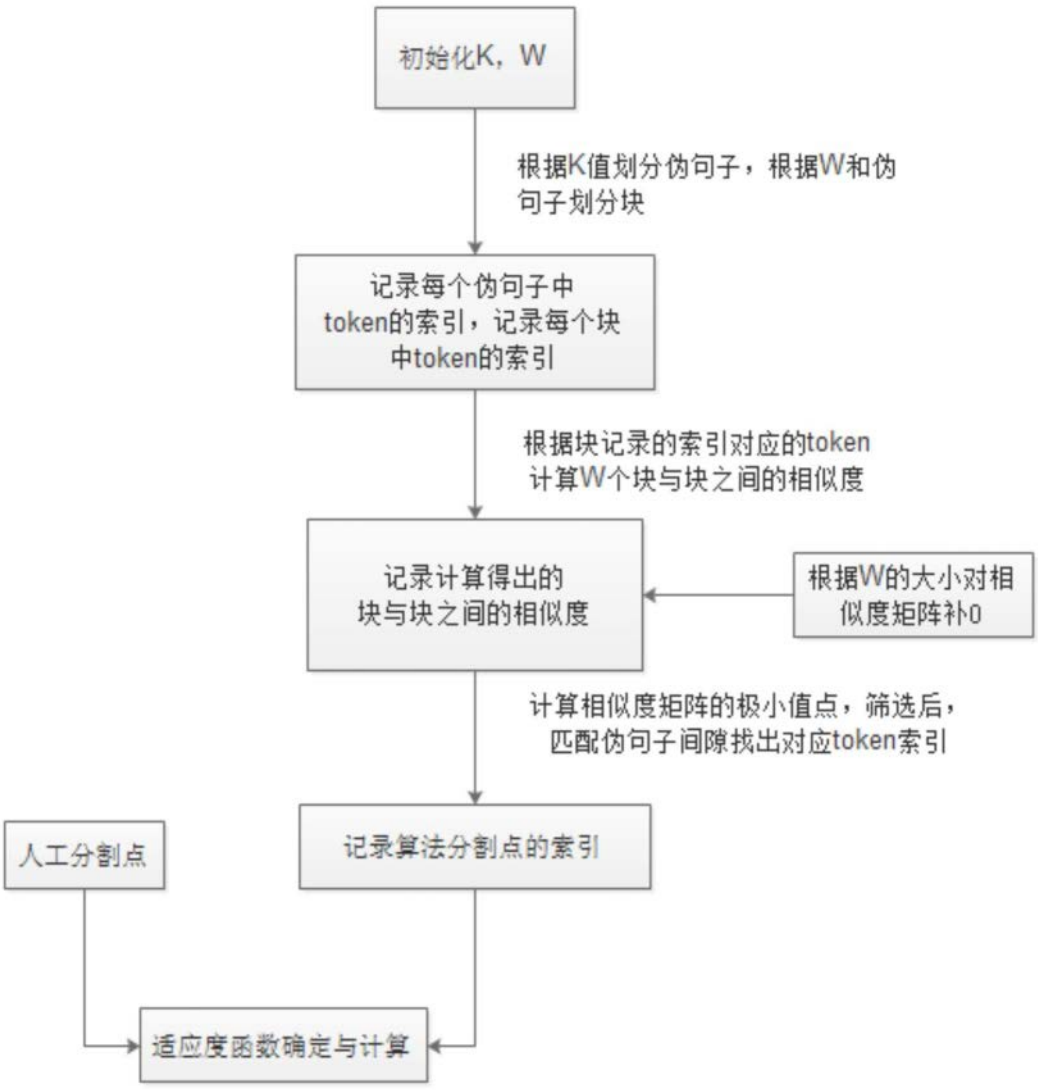

初始化K，W
根据K值划分伪句子，根据W和伪句子划分块
记录每个伪句子中token的索引，记录每个块中token的索引
根据块记录的索引对应的token计算W个块与块之间的相似度
记录计算得出的块与块之间的相似度
根据W的大小对相似度矩阵补0
计算相似度矩阵的极小值点，筛选后，匹配伪句子间隙找出对应token索引
人工分割点
记录算法分割点的索引
适应度函数确定与计算

图4

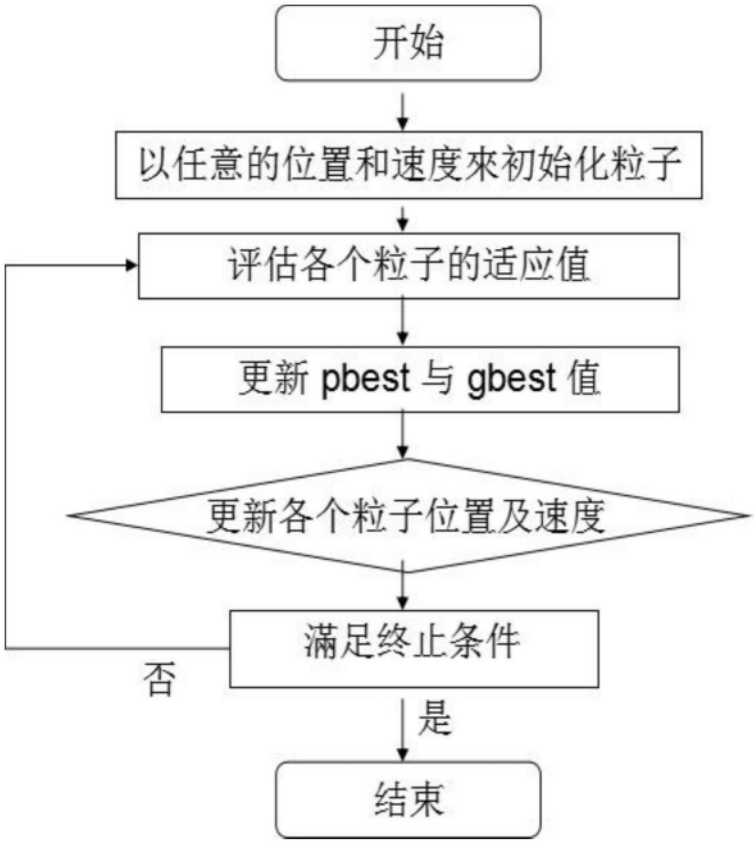

开始
以任意的位置和速度來初始化粒子
评估各个粒子的适应值
更新 pbest 与 gbest 值
更新各个粒子位置及速度
满足终止条件
否
是
结束

图5

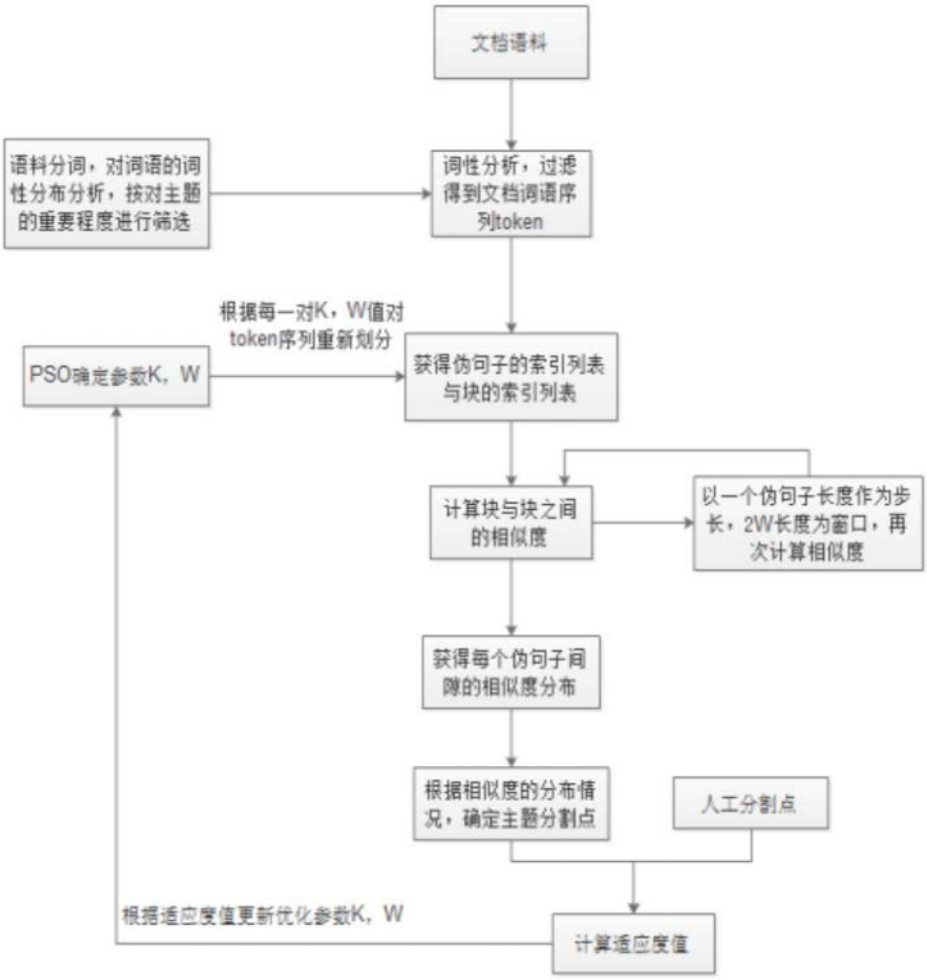

文档语料
语料分词，对词语的词性分布分析，按对主题的重要程度进行筛选
词性分析，过滤得到文档词语序列token
根据每一对K，W值对token序列重新划分
PSO确定参数K，W
获得伪句子的索引列表与块的索引列表
计算块与块之间的相似度
以一个伪句子长度作为步长，2W长度为窗口，再次计算相似度
获得每个伪句子间隙的相似度分布
根据相似度的分布情况，确定主题分割点
人工分割点
根据适应度值更新优化参数K，W
计算适应度值

图6