文本聚类方法、装置及计算设备

申请号：201510944341.9
申请日：2015-12-16

申请(专利权)人　华为技术有限公司
地址　518129 广东省深圳市龙岗区坂田华为总部办公楼
发明(设计)人　胡斐然　王楠楠

主分类号　G06F17/30(2006.01)I
分类号　G06F17/30(2006.01)I
公开(公告)号　105574156A
公开(公告)日　2016-05-11
专利代理机构
代理人

(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 105574156 A
(43)申请公布日 2016.05.11

(21)申请号 201510944341.9

(22)申请日 2015.12.16

(71)申请人 华为技术有限公司
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

(72)发明人 胡斐然 王楠楠

(51)Int.Cl.
G06F 17/30(2006.01)

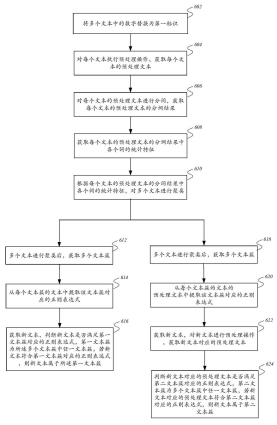
权利要求书2页 说明书9页 附图3页

(54)发明名称

文本聚类方法、装置及计算设备

(57)摘要

本发明实施例公开了一种文本聚类方法。用于文本聚类的设备获取了待聚类的文本后，将待聚类的文本中的数字替换为第一标识，并将待聚类的文本中相邻的第一标识合并获取待聚类的文本的预处理文本，对待聚类的文本的预处理文本进行聚类。通过对待聚类的文本进行预处理，提取了待聚类的文本的格式，根据待聚类的文本的格式对待聚类的文本进行聚类，提升了文本聚类的精度。



CN 105574156 A

1.一种文本聚类方法,所述文本聚类方法由计算机执行,其特征在于,包括:

将多个文本中的数字替换为第一标识;

对每个文本执行预处理操作,获取所述每个文本的预处理文本,所述预处理操作包括:将相邻的两个所述第一标识合并为一个所述第一标识;

对所述每个文本的预处理文本进行分词,获取所述每个文本的预处理文本的分词结果;

获取所述每个文本的预处理文本的分词结果中各个词的统计特征;

根据所述每个文本的预处理文本的分词结果中各个词的统计特征,对所述多个文本进行聚类。

2.如权利要求1所述的方法,其特征在于,所述对每个文本执行预处理操作前,还包括:将所述多个文本中的字素替换为第二标识;

所述预处理操作还包括:将相邻的两个所述第二标识合并为一个所述第二标识。

3.如权利要求1或2所述的方法,其特征在于,所述多个文本进行聚类后,获取多个文本簇;

从每个文本簇的文本中提取该文本簇对应的正则表达式;

获取新文本,判断所述新文本是否满足第一文本簇对应的正则表达式,所述第一文本簇为所述多个文本簇中任一文本簇,若所述新文本符合所述第一文本簇对应的正则表达式,则所述新文本属于所述第一文本簇。

4.如权利要求1或2所述的方法,其特征在于,所述多个文本进行聚类后,获取多个文本簇;

从每个文本簇的文本的预处理文本中提取该文本簇对应的正则表达式;

获取新文本,对所述新文本进行所述预处理操作,获取所述新文本对应的预处理文本;

判断所述新文本对应的预处理文本是否满足第二文本簇对应的正则表达式,所述第二文本簇为所述多个文本簇中任一文本簇,若所述新文本对应的预处理文本符合所述第二文本簇对应的正则表达式,则所述新文本属于所述第二文本簇。

5.一种文本聚类装置,其特征在于,包括:

获取单元,用于将多个文本中的数字替换为第一标识;

处理单元,用于对每个文本执行预处理操作,获取所述每个文本的预处理文本,所述预处理操作包括:将相邻的两个所述第一标识合并为一个所述第一标识;还用于对所述每个文本的预处理文本进行分词,获取所述每个文本的预处理文本的分词结果;还用于获取所述每个文本的预处理文本的分词结果中各个词的统计特征;还用于根据所述每个文本的预处理文本的分词结果中各个词的统计特征,对所述多个文本进行聚类。

6.如权利要求5所述的装置,其特征在于,所述处理单元对每个文本执行预处理操作前,还用于将所述多个文本中的字素替换为第二标识;所述预处理操作还包括:将相邻的两个所述第二标识合并为一个所述第二标识。

7.如权利要求5或6所述的装置,其特征在于,所述处理单元,还用于在所述多个文本进行聚类后,获取多个文本簇;还用于从每个文本簇的文本中提取该文本簇对应的正则表达式;还用于获取新文本,判断所述新文本是否满足第一文本簇对应的正则表达式,所述第一文本簇为所述多个文本簇中任一文本簇,若所述新文本符合所述第一文本簇对应的正则表

达式，则所述新文本属于所述第一文本簇。

8.如权利要求5或6所述的装置，其特征在于，所述处理单元，还用于在所述多个文本进行聚类后，获取多个文本簇；还用于从每个文本簇的文本的预处理文本中提取该文本簇对应的正则表达式；还用于获取新文本，对所述新文本进行所述预处理操作，获取所述新文本对应的预处理文本；还用于判断所述新文本对应的预处理文本是否满足第二文本簇对应的正则表达式，所述第二文本簇为所述多个文本簇中任一文本簇，若所述新文本对应的预处理文本符合所述第二文本簇对应的正则表达式，则所述新文本属于所述第二文本簇。

9.一种计算设备，其特征在于，包括处理器、存储器；

所述处理器用于读取所述存储器中的程序执行以下操作：将多个文本中的数字替换为第一标识；对每个文本执行预处理操作，获取所述每个文本的预处理文本，所述预处理操作包括：将相邻的两个所述第一标识合并为一个所述第一标识；对所述每个文本的预处理文本进行分词，获取所述每个文本的预处理文本的分词结果；获取所述每个文本的预处理文本的分词结果中各个词的统计特征；根据所述每个文本的预处理文本的分词结果中各个词的统计特征，对所述多个文本进行聚类。

10.如权利要求9所述的计算设备，其特征在于，所述处理器对每个文本执行预处理操作前，还将所述多个文本中的字素替换为第二标识；所述预处理操作还包括：将相邻的两个所述第二标识合并为一个所述第二标识。

11.如权利要求9或10所述的计算设备，其特征在于，所述处理器对所述多个文本进行聚类后，获取多个文本簇；从每个文本簇的文本中提取该文本簇对应的正则表达式；获取新文本，判断所述新文本是否满足第一文本簇对应的正则表达式，所述第一文本簇为所述多个文本簇中任一文本簇，若所述新文本符合所述第一文本簇对应的正则表达式，则所述新文本属于所述第一文本簇。

12.如权利要求9或10所述的计算设备，其特征在于，所述处理器对所述多个文本进行聚类后，获取多个文本簇；从每个文本簇的文本的预处理文本中提取该文本簇对应的正则表达式；获取新文本，对所述新文本进行所述预处理操作，获取所述新文本对应的预处理文本；判断所述新文本对应的预处理文本是否满足第二文本簇对应的正则表达式，所述第二文本簇为所述多个文本簇中任一文本簇，若所述新文本对应的预处理文本符合所述第二文本簇对应的正则表达式，则所述新文本属于所述第二文本簇。

文本聚类方法、装置及计算设备

技术领域

[0001] 本发明涉及计算机技术领域，尤其涉及一种文本聚类方法，文本聚类装置以及用于文本聚类的计算设备。

背景技术

[0002] 当存在大量文本时，常需要对这些文本进行聚类，即将大量文本归类到一定数量的簇(英文:cluster)中，以方便后续对这些文本的处理。

[0003] 文本的聚类过程，也即将相似的文本聚集到一起的过程。现有技术中，常根据文本内包含的内容来计算文本之间的相似度，一般包含相同内容较多的多个文本被视为相似程度较高。

[0004] 然而，一些类型的文本，例如日志，包含的内容会随着输入参数和输出参数的变化而变化，因此根据文本包含的内容来对这些文本进行聚类的精度不高。

发明内容

[0005] 本申请提供了一种文本聚类方法，文本聚类装置以及用于文本聚类的计算设备，以提升文本聚类的精度。

[0006] 本申请的第一方面提供了一种文本聚类方法，该方法由计算机执行，包括:接收待聚类的N个文本，N为大于1的整数，将这N个文本中的数字替换为第一标识。对这N个文本执行预处理操作，将这N个文本中相邻的第一标识合并，获得这N个文本对应的N个预处理文本。对N个预处理文本进行分词，获取这N个预处理文本的分词结果，并获取这N个预处理文本的分词结果中各个词的统计特征。根据这N个预处理文本的分词结果中各个词的统计特征，对这N个文本进行聚类。

[0007] 通过对待聚类的文本进行预处理操作，使得文本的预处理文本中保留的不再是文本的内容本身，而是文本的格式，随后根据各个文本的预处理文本来对文本进行聚类，使得聚类过程能够将文本的格式加入考虑，提升了文本聚类的精度。

[0008] 结合第一方面，在第一方面的第一种实现方式中，不仅将N个文本中的数字替换为第一标识，还将这N文本中的字素替换为第二标识。因此，预处理操作还包括:将相邻的两个第二标识合并为一个第二标识。

[0009] 进一步的，不仅仅针对待聚类的文本中的数字进行处理，还对待聚类的文本中的字素进行处理，进一步抽象出待处理的文本的格式，以供后续聚类中使用，能够进一步提升文本聚类的精度。

[0010] 结合第一方面和第一方面的第一种实现方式，在第一方面的第二种实现方式中，对N个文本进行聚类后，获取M个文本簇。从每个文本簇的文本中提取该文本簇对应的正则表达式;获取新文本，判断新文本是否满足M个文本簇中任一文本簇对应的正则表达式，如果该新文本符合任一文本簇对应的正则表达式，则该新文本属于该文本簇。

[0011] 从已经获得的文本簇中提取正则表达式，获取各个文本簇在内容上的共性，获取

了新文本之后，无须将新文本和已经执行过聚类的文本一起重新进行聚类，而是将新文本与各个文本簇对应的正则表达式进行匹配，大幅提升了新文本的聚类速度。

[0012] 结合第一方面和第一方面的第一种实现方式，在第一方面的第三种实现方式中，对N个文本进行聚类后，获取M个文本簇。从每个文本簇包括的文本的预处理文本中提取该文本簇对应的正则表达式；获取新文本，判断新文本是否满足M个文本簇中任一文本簇对应的正则表达式，如果该新文本符合任一文本簇对应的正则表达式，则该新文本属于该文本簇。

[0013] 从已经获得的文本簇的预处理文本中提取正则表达式，获取各个文本簇的预处理文本在格式上的共性，获取了新文本之后，无须将新文本和已经执行过聚类的文本一起重新进行聚类，而是将新文本与各个文本簇对应的正则表达式进行匹配，大幅提升了新文本的聚类速度。

[0014] 本申请的第二方面提供了一种文本聚类装置，该装置包括获取单元和处理单元。获取单元用于，接收待聚类的N个文本，N为大于1的整数，将这N个文本中的数字替换为第一标识。处理单元用于，对这N个文本执行预处理操作，将这N个文本中相邻的第一标识合并，获得这N个文本对应的N个预处理文本；并对这N个预处理文本进行分词，获取这N个预处理文本的分词结果，并获取这N个预处理文本的分词结果中各个词的统计特征；随后根据这N个预处理文本的分词结果中各个词的统计特征，对这N个文本进行聚类。该装置用于实现第一方面提供的文本聚类方法。

[0015] 本申请的第三方面提供了一种计算设备，包括处理器、存储器。该计算设备运行时能够实现第一方面提供的文本聚类方法，用于实现第一方面提供的文本聚类方法的程序代码可以保存在存储器中，并由处理器来执行。

[0016] 本申请的第四方面提供了一种存储介质，该存储介质中存储的程序代码被执行时能够实现第一方面提供的文本聚类方法。该程序代码由实现第一方面提供的文本聚类方法的计算机指令构成。

附图说明

[0017] 为了更清楚地说明本发明实施例或现有技术中的技术方案，下面将对实施例中所需要使用的附图作以简单地介绍，显而易见的，下面描述中的附图是本发明的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

[0018] 图1为本发明提供的文本聚类系统的组织结构示意图；

[0019] 图2为本发明提供的计算设备的组织结构示意图；

[0020] 图3为本发明提供的文本聚类方法的流程示意图；

[0021] 图4为本发明提供的文本聚类装置的组织结构示意图。

具体实施方式

[0022] 下面结合本发明实施例中的附图，对本发明实施例中的技术方案进行描述。

[0023] 贯穿本说明书，术语“无边界语言”指代字符间没有用于划定界限的标点符号或空格的语言，常见的无边界语言包括中文、日文等。相应的，有边界语言指代字符间有用于划

定界限的标点符号或空格的语言，最常见的有边界语言包括英文。

[0024] 贯穿本说明书，术语“聚类”指代根据不同对象的特征，将对象归类到不同的簇的过程。每一个簇包含了有一定共性或者相似程度较高的多个对象。

[0025] 贯穿本说明书，术语“正则表达式”指代一串字符串，该字符串用于描述一系列句法规则，例如包括什么字符、字符位置、字符顺序等。

[0026] 图1为文本聚类系统200的一种实现方式，包括存储设备206、文本聚类设备202。其中存储设备206中存储了用于存储待聚类的文本的文本库，存储设备206可以通过通信网络204与文本聚类设备202建立通信，存储设备206也可以直接设置在文本聚类设备202中，通过输入输入单元2021与文本聚类设备202建立通信。文本聚类设备202中包括输入输出单元2021和处理单元2022。如果存储设备206通过通信网络204与文本聚类设备202通信，则输入输出单元2021可以为网络接口，如果存储设备206部署于文本聚类设备202内，则输入输出单元2021还可以为文本聚类设备202访问本地存储设备的接口。

[0027] 其中，处理器402、存储器404和通信接口406可以通过总线408实现彼此之间的通信连接，也可以通过无线传输等其他手段实现通信。

[0028] 存储器404存储器可以包括易失性存储器(英文:volatile memory)，例如随机存取存储器(英文:random-access memory，缩写:RAM)；存储器也可以包括非易失性存储器(英文:non-volatile memory)，例如只读存储器(英文:read-only memory，缩写:ROM)，快闪存储器(英文:flash memory)，硬盘(英文:hard disk drive，缩写:HDD)或固态硬盘(英文:solid-state drive，缩写:SSD)；存储器404还可以包括上述种类的存储器的组合。计算设备400运行时，存储器404加载存储设备206中文本库中存储的文本，以供处理器402使用。在通过软件来实现本发明提供的技术方案时，用于实现本发明图3提供的文本聚类方法的程序代码可以保存在存储器404中，并由处理器402来执行。

[0029] 计算设备400通过通信接口406获取待处理的文本，当获取文本聚类的结果后，还可以通过通信接口406返回给用户。

[0030] 处理器402可以为中央处理器(英文:central processing unit，缩写:CPU)。处理器402获取文本库中存储的多个文本，并将这些文本中的数字替换为第一标识，第一标识可以为一个特定的字符，例如字母d。对执行完替换操作的文本进行预处理操作，预处理操作即将每一个执行完替换操作的文本中相邻的两个第一标识合并为一个第一标识。如果文本中有多个相邻的第一标识，则可以将多个相邻的第一标识合并为一个第一标识。文本中的空格、标点符号可以保留。

[0031] 一个文本执行完预处理操作后，生成该文本对应的一个预处理文本。因此，N个文本对应于N个预处理文本，N为正整数且N等于待聚类的文本的数量。对每个文本的预处理文本进行分词，如果预处理文本中仅包括标点符号和第一标识，或仅包括有边界语言，例如英文，则根据空格对文本进行分词即可，如果文本中包括无边界语言，则分词还需根据词库中已有词、以及预设的分词方法等对预处理文本进行分词。

[0032] 每个文本的预处理文本的分词结果中每个文本的预处理文本被切分为多个词，例如为M个，获取每个文本的预处理文本中各个词的统计特征，例如为每个词提取一个统计特征，则每个文本的预处理文本中可以提取M个统计特征，根据每个文本的预处理文本的分词结果中各个词的统计特征，对待聚类的多个文本进行聚类。

[0033] 每个文本的预处理文本可以提取M个统计特征，则根据这M个统计特征，对每个文本的预处理文本进行聚类，如果多个文本的的预处理文本被聚类为一个簇，则该多个文本也被聚类到一个簇中。

[0034] 通过预处理操作和分词处理，多个待聚类的文本可以通过一系列词的统计特征来体现，根据这些词的统计特征来对文本进行聚类，使得不再仅根据文本的内容的相似度进行聚类，而是将文本的内容替换为标识并对相邻标识进行合并，用标识来表现文本内容的格式，这样通过文本的格式来对文本进行聚类，可以提升文本的聚类精度。

[0035] 可选的，对每个文本执行预处理操作前，还包括：将待聚类的多个文本中的字素替换为第二标识，则预处理操作还包括将相邻的两个第二标识合并为一个第二标识。进一步的，不仅替换文本中的数字，还将文本中的字素替换为第二标识，使得获得的预处理文本能够更好的表现文本的格式，以提升聚类精度。

[0036] 处理器402将待聚类的多个文本聚类为多个文本簇后，从每个文本簇包括的文本中提取该文本簇对应的正则表达式，每个文本簇对应的正则表达式体现了该文本簇在内容上的一些共同点。获取新文本后，如果需要将新文本也聚类到某一现存的文本簇中，则判断新文本是否满足任一文本簇对应的正则表达式，如果新文本满足某一文本簇对应的正则表达式，则该新文本属于该文本簇。

[0037] 处理器402将待聚类的多个文本聚类为多个文本簇后，从每个文本簇包括的文本的预处理文本中提取该文本簇对应的正则表达式，每个文本簇对应的正则表达式体现了该文本簇中的文本的预处理文本在内容上的一些共同点。获取新文本后，如果需要将新文本也聚类到某一现存的文本簇中，则判断新文本的预处理文本是否满足任一文本簇对应的正则表达式，如果新文本的预处理文本满足某一文本簇对应的正则表达式，则该新文本属于该文本簇。

[0038] 将待聚类的文本分类到不同的文本簇之后，如果文本聚类系统获取了新的文本，无须将全部文本重新聚类，只需从已经获取的文本簇或文本簇对应的预处理文本中提取正则表达式，新文本满足哪个文本簇或文本簇对应的预处理文本中提取出的正则表达式，则该新文本就归类于哪个文本簇中，加快了新文本的聚类速度。

[0039] 本发明还提供了一种文本聚类方法，图1中的文本聚类设备202以及图2中的计算设备400运行时执行该文本聚类方法，其流程示意图如图3所示。

[0040] 步骤602，将多个文本中的数字替换为第一标识。

[0041] 获取待聚类的多个文本，将待聚类的多个文本中的数字替换为第一标识，本说明书中以第一标识为字符“d”为例。文本1为待聚类的多个文本中的一个，文本1包括Aug 1704:27:2203peloton kernel:[pid]uid tgid totalvm，将文本1中的数字替换为第一标识后，文本1包括Aug dd dd:dd:dddd peloton kernel:[pid]uid tgid totalvm。

[0042] 可选的，步骤602中还可以将待聚类的多个文本中的字素替换为第二标识，本说明书中以第二标识为字符“w”为例，则执行完步骤602后，文本1包括www dd dd:dd:dddd wwwwww wwwwww:[www]www wwww wwwww ww。

[0043] 步骤604，对每个文本执行预处理操作，获取每个文本的预处理文本，预处理操作包括：将相邻的两个第一标识合并为一个第一标识。

[0044] 待聚类的多个文本中的数字均替换为第一标识后，对每个文本执行预处理操作，

预处理操作即将每一个文本中相邻的两个第一标识合并为一个第一标识。如果文本中有多个相邻的第一标识,则可以将多个相邻的第一标识合并为一个第一标识。文本中的空格、标点符号可以保留。以文本1为例,文本1执行预处理操作后,文本1的预处理文本包括Aug d d:d:ddd peloton kernel:[pid]uid tgid totalvm,也可以对文本1中相邻的第一标识进一步进行合并,直至文本1的预处理文本中无相邻的第一标识,即文本1的预处理文本包括Aug d d:d:d peloton kernel:[pid]uid tgid totalvm。两个字符之间无标点符号且无空格且无其他字符则称这两个数字相邻。

[0045] 可选的,如果步骤602中还将待聚类的多个文本中的字素替换为第二标识则,预处理操作还包括:将相邻的两个第二标识合并为一个第二标识。合并的过程参考将相邻的两个第一标识合并为一个第一标识的过程。还可以进一步对相邻的第一标识进行合并且对相邻的第二标识进行合并,直至文本1的预处理文本中无相邻的第一标识且无相邻的第二标识,例如文本1的预处理文本包括ww d d:d:ddd wwwww wwwww:[ww]ww www wwwww,则文本1的预处理文本包括w d d:d:d w w:[w]w w w。

[0046] 步骤606,对每个文本的预处理文本进行分词,获取每个文本的预处理文本的分词结果。

[0047] 对文本的预处理文本进行分词的方法有多种,常见对有边界语言的分词方法包括N-Gram分词法,对无边界语言的分词方法一般需要结合词库中的已知词,对预处理文本进行分词后,预处理文本的分词结果中包含预处理文本被切分出来的各个词。以3-Gram分词为例,文本1的预处理文本w d d:d:d w w:[w]w w w的分词结果包括w d d:d:d,d d:d:d w,d:d:d w w:,w w:[w],w:[w]w,[w]w w,w w w,共7个词。

[0048] 步骤608,获取每个文本的预处理文本的分词结果中各个词的统计特征。

[0049] 获取每个文本的预处理文本的分词结果后,进一步获取分词结果中各个词的统计特征,统计特征包括词频、词的方差、词的词频-逆文档频率(英文:term frequency-inverse document frequency,缩写:TF-IDF)等。如果一个文本的预处理文本的分词结果中包括K个词,且为K个词中的每个词提取L个统计特征,则该文本的预处理文本总共可以提取K*L个统计特征,因此,该文本的预处理文本可以通过K*L维的向量表达。每个待聚类的文本的预处理文本均提取了对应的统计特征后,每个待聚类的文本的预处理文本可以通过一个向量表达。

[0050] 步骤610,根据每个文本的预处理文本的分词结果中各个词的统计特征,对多个文本进行聚类。

[0051] 获取每个待聚类的文本的预处理文本对应的统计特征后,根据每个文本的预处理文本的分词结果中各个词的统计特征,通过聚类算法可以对待聚类的文本进行聚类。聚类算法包括k-means,k-medoid,clarans,birch,cure,chameleon,dbscan,optics,denclue等。一个文本对应于一个预处理文本,一个预处理文本对应于一个分词结果,一个分词结果对应于一系列词的统计特征,因此,如果两个文本的分词结果包括的词的统计特征被聚类算法识别为属于同一簇,则这两个文本属于同一文本簇。

[0052] 以待聚类的文本如下文本1至文本7为例:

[0053] 文本1:Aug 17 04:27:22peloton kernel:[pid]uid tgid totalvm

[0054] 文本2:Aug 17 03:41:44peloton kernel:[pid]uid tgid totalvm

[0055] 文本3:Aug 17 03:26:41peloton kernel:Free swap

[0056] 文本4:Aug 17 03:37:33peloton kernel:Total swap

[0057] 文本5:Sep 17 08:51:66peloton kernel:[pid]uid tgid total

[0058] 文本6:Jan 23 08:51:66peloton kernel:?do_page

[0059] 文本7:Jan 27 11:51:66peloton kernel:?security_real

[0060] 经过文本预处理后,文本1至文本7分别对应的预处理文本为:

[0061] 文本1的预处理文本:w d d:d:d w w:[w]w w w

[0062] 文本2的预处理文本:w d d:d:d w w:[w]w w w

[0063] 文本3的预处理文本:w d d:d:d w w:w w

[0064] 文本4的预处理文本:w d d:d:d w w:w w

[0065] 文本5的预处理文本:w d d:d:d w w:[w]w w w

[0066] 文本6的预处理文本:w d d:d:d w w:?w_w

[0067] 文本7的预处理文本:w d d:d:d w w:?w_w

[0068] 通过对文本1至文本7的预处理文本进行聚类后,文本1、文本2、文本5聚类为文本簇1,文本3和文本4聚类为文本簇2,文本6和文本7聚类为文本簇3。

[0069] 获得了多个文本簇后,可以根据已获得的文本簇结果,进一步对文本簇进行合并,例如如果文本簇1和文本簇3中包括的文本相关,因此可以将文本簇1和文本簇3再次合并,合并出来的文本簇4包括文本1、文本2、文本5、文本6以及文本7。具体的,可以通过预设的条件,或用户根据其需求对聚类获得的文本簇进行合并。

[0070] 需要说明的是,本发明中的文本,也可以为待聚类的对象的一部分。常见的待聚类的对象例如日志,包括:CPU:90MEM:80info:Aug 1704:27:22peloton kernel:[pid]uid tgid totalvm,其中info字段包括的内容可以为本发明所述的文本。因此,待聚类的对象除了包括文本之外,还包括了其他字段包括的内容,例如CPU字段包括的内容指示该日志生成时CPU的利用率,MEM字段包括的内容指示该日志生成时内存的利用率。这部分内容已经为数字类型,可以直接用于聚类。

[0071] 文本经过预处理和分词并进一步对这些词提取统计特征后,如步骤608中所述,该文本的预处理文本可以通过K*L维的向量表达。因此,如果该文本是待聚类的日志的一部分,则该日志可以通过K*L+2维的向量表达,即K*L维的从文本的预处理文本中提取的统计特征,以及日志中已有的CPU和MEM两个字段包括的内容。

[0072] 由于文本的预处理文本分词获得的词的个数往往较多,且每个词能够提取的统计特征也可以有多个,因此K*L的数量往往较高。如果日志通过K*L+2个向量表达,且该向量的每个维度对聚类结果影响的权重相同,则日志在聚类的过程中,会过多的被文本的内容所影响,导致日志中其他字段包括的内容对聚类结果的影响太小。因此,从文本中提取了K*L维的统计特征后,还可以为这K*L维的统计特征设置权重,例如这K*L维中每个维度的权重为P/K*L,P为预设的参数,而CPU和MEM两个字段后包括的内容的权重为1,则文本中提取的K*L维的统计特征的权重之和为P,根据P的设置可以调整文本的内容对聚类结果的影响,避免了K*L过大的情况下,待聚类的对象中除文本外的其他字段的内容对聚类结果的影响被过度淡化,提升了聚类精度。

[0073] 可选的,步骤610后还包括步骤612至步骤616。

[0074] 步骤612，多个文本进行聚类后，获取多个文本簇。

[0075] 步骤614，从每个文本簇的文本中提取该文本簇对应的正则表达式。

[0076] 从步骤612获取的每个文本簇中提取该文本簇对应的正则表达式，例如如果一个文本簇中包括的全部文本均以“mytime”开头，且以anomalyScore”结尾，“则这个文本簇可以提取出正则表达式“^mytime.*anomalyScore$”。每个文本簇提取的正则表达式，能够让本文本簇中的每个文本均符合，或本文本簇中超过一定比例的文本符合。

[0077] 步骤616，获取新文本，判断新文本是否满足第一文本簇对应的正则表达式，第一文本簇为所述多个文本簇中任一文本簇，若新文本符合第一文本簇对应的正则表达式，则新文本属于所述第一文本簇。

[0078] 用于文本聚类的设备获取了待聚类的新文本后，判断该新文本能否满足任一文本簇对应的正则表达式。如果该新文本能够满足某一文本簇对应的正则表达式，则该新文本属于该文本簇。如果该新文本同时满足多个文本簇对应的正则表达式，则该新文本可以被归类于这多个文本簇中的任一个。

[0079] 用于文本聚类的设备可以用其已有的文本中的一部分来执行步骤602至步骤612，以获取多个文本簇，然后用已有的文本中的另一部分来执行步骤614与步骤616，则该已有的文本中的另一部分即新文本。该过程类似于机械学习算法中采用一部分样本用于训练模型，采用训练完毕的模型来对剩余的样本进行识别的过程，使得用于文本聚类的设备无须对全部文本进行聚类，提升了文本聚类的效率。用于文本聚类的设备也可以其已有的全部文本来执行步骤602至步骤612，以获取多个文本簇，然后获取到新生成或用户新输入的文本后，用新生成或用户新输入的文本来执行步骤614与步骤616，则新生成或用户新输入的文本即新文本。

[0080] 通过步骤612至步骤616，新文本得以被归类到某一文本簇中，如果新文本无法满足任一文本簇对应的正则表达式，该新文本也可以属于一个新的文本簇。使得获取了新文本的情况下，无须将新文本连同已经执行过聚类的文本重新进行聚类，通过已经获取的文本簇的正则表达式来对新文本进行聚类，提升了新文本的聚类的速度。

[0081] 可选的，步骤610后还包括步骤618至步骤624

[0082] 步骤618，多个文本进行聚类后，获取多个文本簇。

[0083] 步骤620，从每个文本簇的文本的预处理文本中提取该文本簇对应的正则表达式。

[0084] 从步骤618获取的每个文本簇所包括的文本的预处理文本中提取该文本簇对应的正则表达式，例如如果一个文本簇中全部文本的预处理文本均以“mytime”开头，且以“d:d”结尾，则这个文本簇的预处理文本可以提取出正则表达式“^mytime.*d:d$”。每个文本簇对应的正则表达式，能够让本文本簇中的每个文本的预处理文本均符合，或本文本簇中超过一定比例的文本的预处理文本符合。

[0085] 步骤622，获取新文本，对新文本进行预处理操作，获取新文本对应的预处理文本。

[0086] 步骤622中对新文本进行的预处理操作参考步骤604及步骤604的可选方案。

[0087] 步骤624，判断新文本对应的预处理文本是否满足第二文本簇对应的正则表达式，第二文本簇为多个文本簇中任一文本簇，若新文本对应的预处理文本符合第二文本簇对应的正则表达式，则新文本属于第二文本簇。

[0088] 用于文本聚类的设备获取了待聚类的新文本后，判断该新文本对应的预处理文本

能否满足任一文本簇对应的正则表达式。如果该新文本对应的预处理文本能够满足某一文本簇对应的正则表达式，则该新文本属于该文本簇。如果该新文本对应的预处理文本同时满足多个文本簇对应的正则表达式，则该新文本可以被归类于这多个文本簇中的任一个。

[0089] 用于文本聚类的设备可以用其已有的文本中的一部分来执行步骤602至步骤618，以获取多个文本簇，然后用已有的文本中的另一部分来执行步骤620至步骤624，则该已有的文本中的另一部分即新文本。该过程类似于机械学习算法中采用一部分样本用于训练模型，采用训练完毕的模型来对剩余的样本进行识别的过程，使得用于文本聚类的设备无须对全部文本进行聚类，提升了文本聚类的效率。用于文本聚类的设备也可以其已有的全部文本来执行步骤602至步骤618，以获取多个文本簇，然后获取到新生成或用户新输入的文本后，用新生成或用户新输入的文本来执行步骤620至步骤624，则新生成或用户新输入的文本即新文本。

[0090] 通过步骤618至步骤624，新文本得以被归类到某一文本簇中，如果新文本无法满足任一文本簇对应的正则表达式，该新文本也可以属于一个新的文本簇。使得获取了新文本的情况下，无须将新文本连同已经执行过聚类的文本重新进行聚类，通过已经获取的文本簇的正则表达式来对新文本进行聚类，提升了新文本的聚类的速度。

[0091] 上述实施例提供了一种文本聚类方法，对待聚类的文本进行预处理后，对文本的预处理文本进行分词并聚类，使得对文本的聚类能够根据文本的格式进行，提升了文本聚类的精度。

[0092] 本发明实施例还提供了文本聚类装置800，该文本聚类装置800可以通过图1所示的文本聚类设备202实现，还可以通过图2所示的计算设备400实现，还可以通过专用集成电路(英文:application-specific integrated circuit，缩写:ASIC)实现，或可编程逻辑器件(英文:programmable logic device，缩写:PLD)实现。上述PLD可以是复杂可编程逻辑器件(英文:complex programmable logic device，缩写:CPLD)，可编程逻辑门阵列(英文:field-programmable gate array，缩写:FPGA)，通用阵列逻辑(英文:generic array logic，缩写:GAL)或其任意组合。该文本聚类装置800用于实现图3所示的文本聚类方法。

[0093] 文本聚类装置800包括获取单元802，用于将多个文本中的数字替换为第一标识；以及处理单元804，用于对每个文本执行预处理操作，获取每个文本的预处理文本，预处理操作包括：将相邻的两个第一标识合并为一个第一标识；还用于对每个文本的预处理文本进行分词，获取每个文本的预处理文本的分词结果；还用于获取每个文本的预处理文本的分词结果中各个词的统计特征；还用于根据每个文本的预处理文本的分词结果中各个词的统计特征，对多个文本进行聚类。

[0094] 可选的，处理单元804对每个文本执行预处理操作前，还用于将多个文本中的字素替换为第二标识；预处理操作还包括：将相邻的两个第二标识合并为一个第二标识。

[0095] 可选的，处理单元804，还用于在多个文本进行聚类后，获取多个文本簇；还用于从每个文本簇的文本中提取该文本簇对应的正则表达式；还用于获取新文本，判断新文本是否满足第一文本簇对应的正则表达式，第一文本簇为多个文本簇中任一文本簇，若新文本符合第一文本簇对应的正则表达式，则新文本属于第一文本簇。

[0096] 可选的，处理单元804，还用于在多个文本进行聚类后，获取多个文本簇；还用于从每个文本簇的文本的预处理文本中提取该文本簇对应的正则表达式；还用于获取新文本，

对新文本进行预处理操作，获取新文本对应的预处理文本；还用于判断新文本对应的预处理文本是否满足第二文本簇对应的正则表达式，第二文本簇为多个文本簇中任一文本簇，若新文本对应的预处理文本符合第二文本簇对应的正则表达式，则新文本属于第二文本簇。

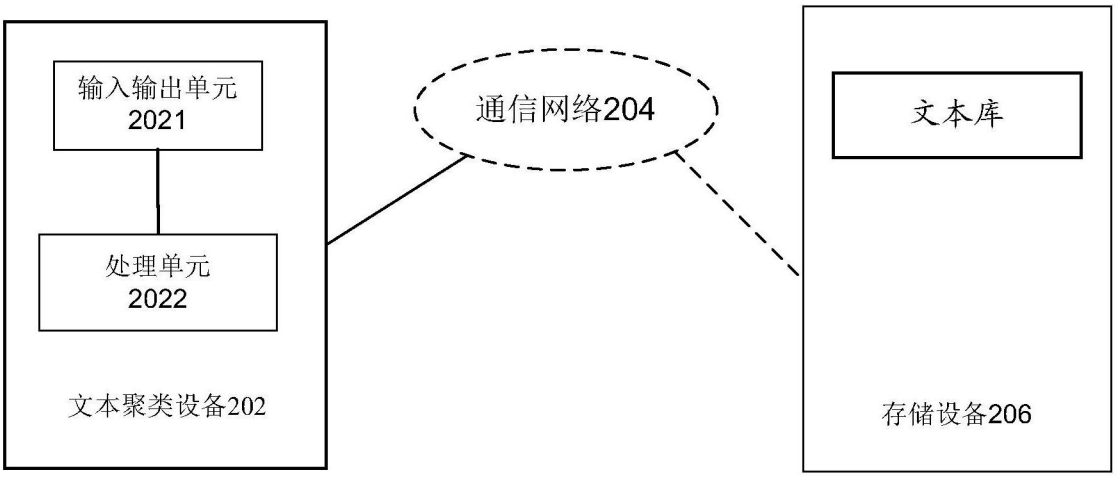
[0097] 上述实施例提供了一种文本聚类装置，该装置对待聚类的文本进行预处理后，对文本的预处理文本进行分词并聚类，使得对文本的聚类能够根据文本的格式进行，提升了文本聚类的精度。

[0098] 需要说明的是：对于前述的各方法实施例，为了简单描述，故将其都表述为一系列的动作组合，但是本领域技术人员应该知悉，本发明并不受所描述的动作顺序的限制，因为依据本发明，某些步骤可以采用其他顺序或者同时进行。其次，本领域技术人员也应该知悉，说明书中所描述的实施例均属于优选实施例，所涉及的动作和单元并不一定是本发明所必须的。

[0099] 在上述实施例中，对各个实施例的描述都各有侧重，某个实施例中没有详述的部分，可以参见其他实施例的相关描述。

[0100] 结合本发明公开内容所描述的方法可以由处理器执行软件指令的方式来实现。软件指令可以由相应的软件模块组成，软件模块可以被存放于RAM、快闪存储器、ROM、可擦除可编程只读存储器(英文：erasable programmable read only memory，缩写：EPROM)、电可擦可编程只读存储器(英文：electrically erasable programmable read only memory，缩写：EEPROM)、硬盘、光盘或者本领域熟知的任何其它形式的存储介质中。

[0101] 最后应说明的是：以上实施例仅用以说明本发明的技术方案，而非对其限制；尽管参照前述实施例对本发明进行了详细的说明，本领域的普通技术人员应当理解：其依然可以对前述各实施例所记载的技术方案进行修改，或者对其中部分技术特征进行等同替换；而这些修改或者替换，并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。



文本聚类系统200

图1

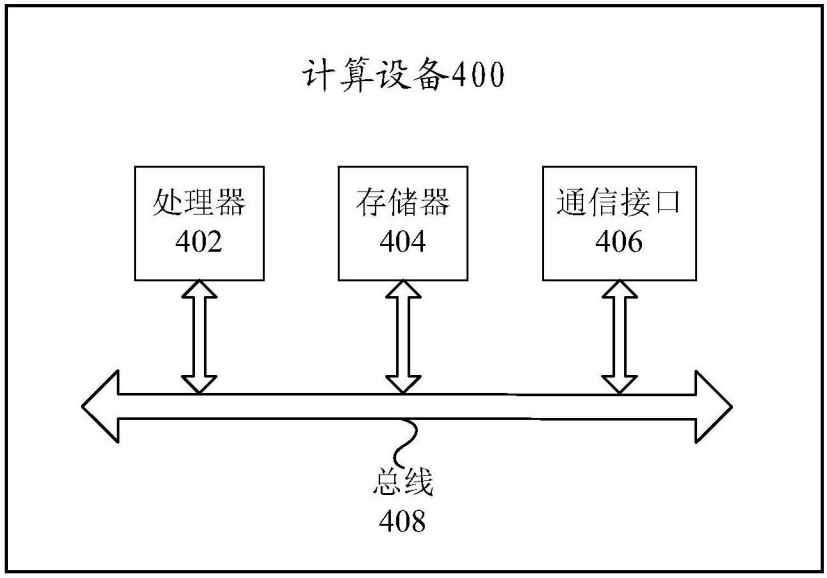


图2

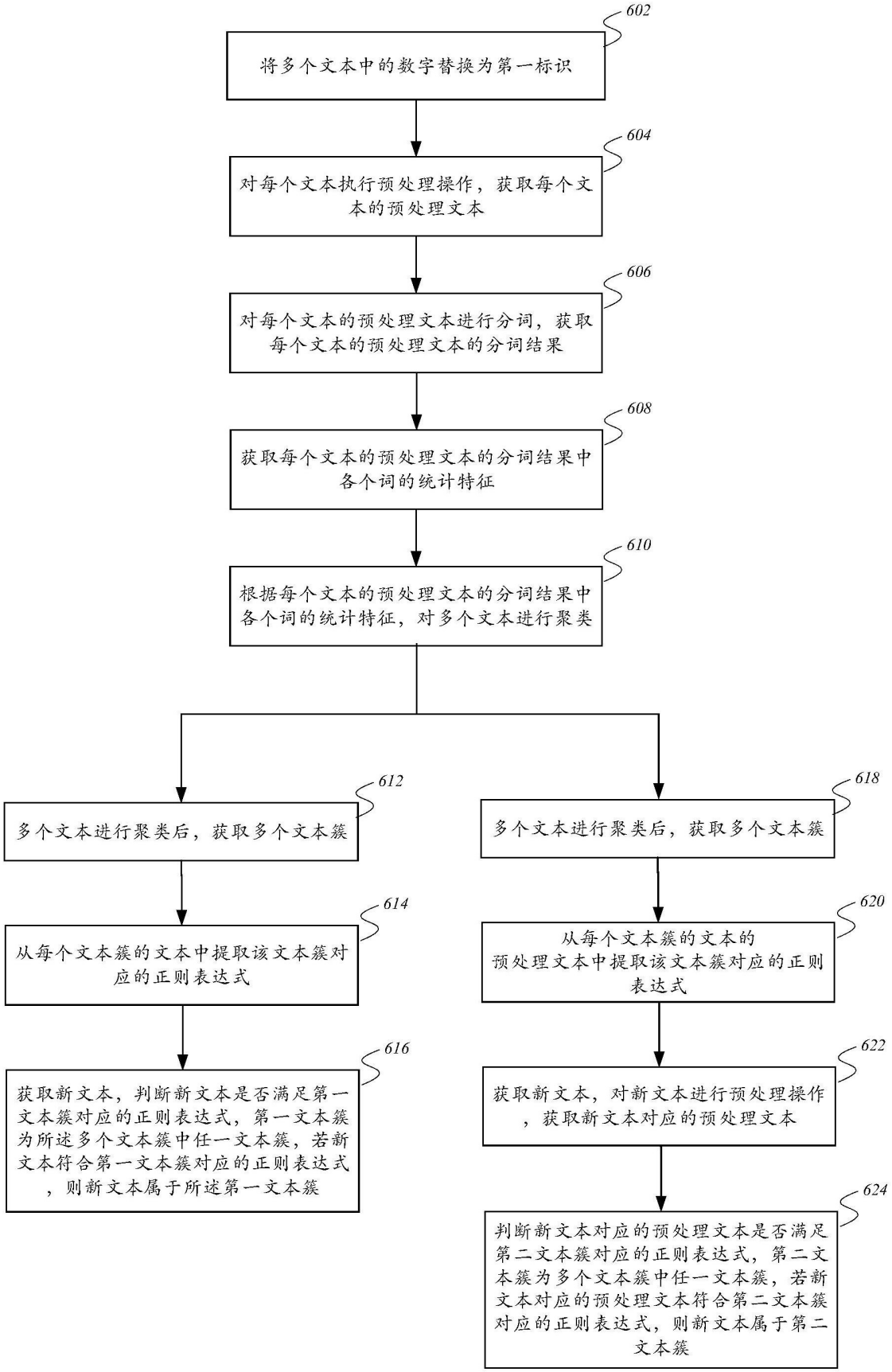

602
将多个文本中的数字替换为第一标识
604
对每个文本执行预处理操作，获取每个文本的预处理文本
606
对每个文本的预处理文本进行分词，获取每个文本的预处理文本的分词结果
608
获取每个文本的预处理文本的分词结果中各个词的统计特征
610
根据每个文本的预处理文本的分词结果中各个词的统计特征，对多个文本进行聚类
612
多个文本进行聚类后，获取多个文本簇
614
从每个文本簇的文本中提取该文本簇对应的正则表达式
616
获取新文本，判断新文本是否满足第一文本簇对应的正则表达式，第一文本簇为所述多个文本簇中任一文本簇，若新文本符合第一文本簇对应的正则表达式，则新文本属于所述第一文本簇
618
多个文本进行聚类后，获取多个文本簇
620
从每个文本簇的文本的预处理文本中提取该文本簇对应的正则表达式
622
获取新文本，对新文本进行预处理操作，获取新文本对应的预处理文本
624
判断新文本对应的预处理文本是否满足第二文本簇对应的正则表达式，第二文本簇为多个文本簇中任一文本簇，若新文本对应的预处理文本符合第二文本簇对应的正则表达式，则新文本属于第二文本簇

图3

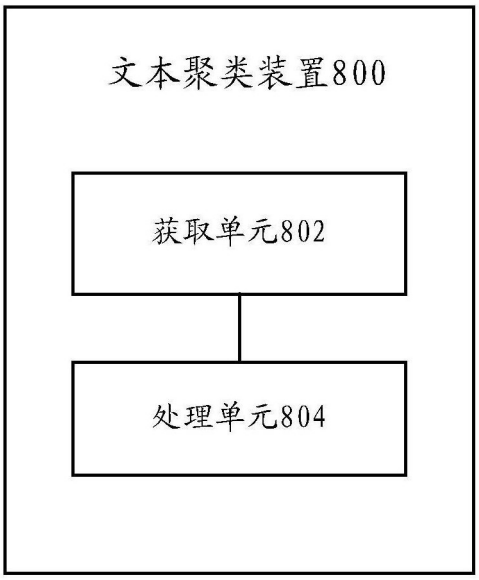


图4