



(12) 发明专利

(10) 授权公告号 CN 102918532 B

(45) 授权公告日 2015. 12. 16

(21) 申请号 201180027027. 4

代理人 陈斌

(22) 申请日 2011. 04. 19

(51) Int. Cl.

(30) 优先权数据

12/791, 756 2010. 06. 01 US

G06F 17/30(2006. 01)

G06F 9/44(2006. 01)

(85) PCT国际申请进入国家阶段日

2012. 11. 30

审查员 程小梅

(86) PCT国际申请的申请数据

PCT/US2011/033125 2011. 04. 19

(87) PCT国际申请的公布数据

W02011/152925 EN 2011. 12. 08

(73) 专利权人 微软技术许可有限责任公司

地址 美国华盛顿州

(72) 发明人 V·坦科维奇 D·梅耶泽

V·波兹南斯基

(74) 专利代理机构 上海专利商标事务所有限公

司 31100

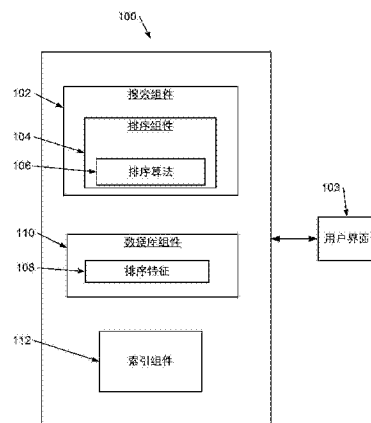
权利要求书2页 说明书11页 附图5页

(54) 发明名称

在搜索结果排序中对垃圾的检测

(57) 摘要

各实施例涉及使用垃圾简档来排序搜索结果。对于给定文档语料库，可以创建和维护一个或多个垃圾简档。垃圾简档提供了表示已知的垃圾文档的参考度量。例如，垃圾简档可以包括自动地插入到使用特定系统或模板创建的文档中的文档数据的词典。垃圾简档还可以包括已知垃圾文档的特定垃圾变量的分布的一个或多个表示(例如，直方图)。垃圾简档提供已知垃圾文档的可使用的表示，本系统和方法使用垃圾简档来预测语料库中的文档是垃圾的可能性。在各实施例中，计算垃圾分数，并响应于搜索查询，将其用于将这样的文档排序得高一些或低一些。



1. 一种用于响应于搜索查询来对候选文档进行排序的计算机实现的方法, 包括下列步骤:

由至少第一处理器, 创建语料库中的多个文档的索引;

计算所述语料库中的至少第一文档的垃圾分数, 其中所述计算垃圾分数包括:

为语料库中的第一文档生成第一候选直方图, 其中所述第一候选直方图特定于所述第一文档;

将所述第一候选直方图与垃圾简档进行比较, 其中所述垃圾简档包括第一已知垃圾文档的第一参考直方图, 所述第一参考直方图特定于所述第一已知垃圾文档并且基于第一垃圾变量;

接收搜索查询;

基于所述搜索查询和所述索引, 从所述语料库中的所述多个文档中标识候选文档, 其中, 所述候选文档包括至少所述第一文档;

对所述候选文档进行排序。

2. 如权利要求 1 所述的计算机实现的方法, 其特征在于, 所述垃圾简档包括至少一个已知垃圾文档的表示, 其中, 对所述候选文档进行排序包括至少部分地基于所述第一文档的所述垃圾分数来对所述候选文档进行排序。

3. 如权利要求 1 所述的计算机实现的方法, 其特征在于, 还包括:

为至少所述第一文档, 创建至少第一垃圾变量的候选直方图;

其中, 所述垃圾简档包括至少第一已知垃圾文档的所述第一垃圾变量的第一参考直方图; 以及

其中, 计算所述垃圾分数包括将所述候选直方图与所述第一参考直方图进行比较, 以确定第一相似度度量。

4. 如权利要求 3 所述的计算机实现的方法, 其特征在于, 所述垃圾简档包括第二已知垃圾文档的所述第一垃圾变量的第二参考直方图, 其中, 计算所述垃圾分数包括将所述候选直方图与所述第二参考直方图进行比较, 以确定第二相似度度量。

5. 如权利要求 4 所述的计算机实现的方法, 其特征在于, 计算所述垃圾分数包括下列各项中的至少一项: 计算所述第一和第二相似度度量中的最大值, 以及计算所述第一和第二相似度度量的平均值。

6. 如权利要求 1 所述的计算机实现的方法, 其特征在于, 还包括显示已排序的候选文档和显示至少所述第一文档的垃圾状态的步骤。

7. 如权利要求 1 所述的计算机实现的方法, 其特征在于:

所述垃圾简档包括自动地生成的数据的词典;

计算所述垃圾分数还包括将来自所述语料库中的所述多个文档的文档数据与自动地生成的数据的所述词典进行比较; 以及

创建所述索引包括在所述索引中描绘匹配所述自动地生成的数据的文档数据。

8. 如权利要求 7 所述的计算机实现的方法, 其特征在于, 标识所述候选文档包括将所述搜索查询与所述索引中的文档数据进行比较, 且其中, 对所述候选文档进行排序包括判断匹配所述搜索查询的文档数据是否被描绘为匹配所述自动地生成的数据。

9. 一种用于响应于搜索查询来对候选文档进行排序的系统, 包括:

用于创建语料库中的多个文档的索引的装置；

用于计算所述语料库中的至少第一文档的垃圾分数的装置，其中所述计算垃圾分数包括：

为语料库中的第一文档生成第一候选直方图，其中所述第一候选直方图特定于所述第一文档；

将所述第一候选直方图与垃圾简档进行比较，其中所述垃圾简档包括第一已知垃圾文档的第一参考直方图，所述第一参考直方图特定于所述第一已知垃圾文档并且基于第一垃圾变量；

用于接收搜索查询的装置；

用于基于所述搜索查询和所述索引，从所述语料库中的所述多个文档标识候选文档的装置，其中，所述候选文档包括至少所述第一文档；

用于至少部分地基于所述第一文档的所述垃圾分数，对所述候选文档进行排序的装置。

10. 如权利要求 9 所述的系统，其特征在于，所述系统进一步包括：

用于为至少所述第一文档，创建至少第一垃圾变量的候选直方图的装置；

其中，所述垃圾简档包括至少第一已知垃圾文档的所述第一垃圾变量的第一参考直方图；

其中，计算所述垃圾分数包括将所述候选直方图与所述第一参考直方图进行比较，以确定第一相似度度量；

其中，所述垃圾简档包括自动地生成的数据的词典；

其中，计算所述垃圾分数还包括将来自所述语料库中的所述多个文档的文档数据与自动地生成的数据的所述词典进行比较；以及

其中，创建所述索引包括在所述索引中描绘匹配所述自动地生成的数据的文档数据。

在搜索结果排序中对垃圾的检测

[0001] 背景

[0002] 计算机用户具有不同的方式来定位可以本地或远程存储的信息。例如，搜索引擎可以被用来使用搜索查询来定位文档。搜索引擎尝试基于特定搜索查询来返回相关结果。

[0003] 概述

[0004] 提供本概述是为了以精简的形式介绍将在以下详细描述中进一步描述的一些概念。本概述并不旨在标识出所要求保护的主题的关键特征或必要特征，也不旨在用于帮助确定所要求保护的主题的范围。

[0005] 各实施例被配置成使用垃圾简档来排序搜索结果。在各实施例中，可以计算诸如垃圾分数之类的排序特征，并且排序特征被排序算法用来响应于搜索查询来排序候选文档。在一个实施例中，创建索引，以促进响应于搜索查询对候选文档的标识。在各实施例中，通过消除或分开地索引当创建文档时可能已经被自动地插入的文档数据来优化索引。在各实施例中，可以通过确定一个文档和已知垃圾文档之间的相似度来进一步计算文档的垃圾分数。在各实施例中，垃圾分数基于该文档的直方图与已知垃圾文档的直方图的比较。直方图可以基于诸如词频和块大小之类的不同的垃圾变量。在各实施例中，可以基于计算出的垃圾分数，将搜索结果标识为可能的垃圾文档，不管搜索结果是否是基于垃圾分数来排序的。这样的作为可能的垃圾的标识可以向用户显示，并可以被维护为可搜索的参数。

[0006] 通过阅读下面的“详细描述”并参考相关联的图形，这些及其他特点和优点将变得显而易见。应该理解，前面的一般性的说明和下面的详细描述只是说明性的，不会对如权利要求所述的本发明形成限制。

[0007] 附图简述

[0008] 图 1 示出了根据本发明的用于排序搜索结果的系统的实施例。

[0009] 图 2 示出了根据本发明的用于排序搜索结果的方法。

[0010] 图 3 示出了根据本发明的用于创建索引的方法。

[0011] 图 4 示出了根据本发明的用于计算垃圾分数的方法。

[0012] 图 5 示出了根据本发明的示例性操作环境。

[0013] 详细描述

[0014] 响应于搜索查询返回的垃圾文档会使用户有挫败感并浪费宝贵的时间和精力。在各实施例中，“垃圾”文档可以包括不怎么包含人可读取的文档数据的文档或除由用于创建文档的系统或模板自动地添加的数据以外包含很少文档数据的文档。例如，在企业环境中，垃圾文档可以是生成的，但是不曾添加有意义的内容的文档。这样的文档常常可以具有与特定搜索查询项非常匹配的标题，流行的类型的统一资源定位符(“URL”)和匹配的锚文本。因此，默认的搜索结果排序器可能没有办法区别这样的垃圾文档与具有有用信息的文档，并可能将这样的垃圾文档排序在很高的位置。

[0015] 如上文简要描述的，此处所公开的各实施例涉及使用垃圾简档来排序搜索结果。对于给定文档语料库，可以创建和维护一个或多个垃圾简档。一般而言，垃圾简档提供了表示已知的垃圾文档的参考度量或模型。例如，垃圾简档可以包括自动地插入到使用特定系

统或模板创建的文档中的文档数据的词典。垃圾简档还可以包括已知垃圾文档的特定垃圾变量的分布的一个或多个表示(例如,直方图)。垃圾简档有效地提供已知垃圾文档的可使用的表示,本系统和方法使用垃圾简档来预测语料库中的文档是垃圾的可能性。在各实施例中,计算垃圾分数,并响应于搜索查询,将其用于将这样的文档排序得高一些或低一些。

[0016] 本系统和方法,虽然不如此限制,可以在其中文档语料库通过一个或多个已知系统和/或模板创建的企业环境中特别有用。在各实施例中,本发明的搜索组件可以使用诸如 MICROSOFT OFFICE SHAREPOINT **SERVER**® 系统之类的集成的服务器平台的功能,来计算、收集、存储,以及更新垃圾分数及可以被用作排序判断的一部分的其他排序特征。由于 MICROSOFT OFFICE SHAREPOINT **SERVER**® 系统包括可以用来创建文档的标准的以及可自定义的“模板”,因此,垃圾简档的创建可以得到简化。

[0017] 在一个实施例中,系统包括搜索组件,该搜索组件包括可以作为计算机可读取的存储介质的一部分被包括的搜索应用程序。搜索应用程序可以被用来部分地基于用户查询来提供搜索结果。例如,用户可以向搜索应用程序输入关键字或其他搜索参数,搜索应用程序可以使用搜索参数来标识候选文档。候选文档可以部分地根据使用垃圾简档被排序,并呈现给用户。

[0018] 图 1 是包括索引、搜索,及其他功能的系统 100 的框图。例如,系统 100 可包括索引、搜索,及其他应用程序,它们可以被用来作为索引的数据结构的一部分来索引信息并使用已索引的数据结构,搜索相关数据。如下面所描述的,系统 100 的组件可以被用来至少部分地基于文档的一个或多个垃圾分数,来排序和返回搜索结果。用户可以使用诸如,例如,浏览器或搜索窗口之类的用户界面 103,向搜索组件 102 提交查询。

[0019] 如图 1 所示,系统 100 包括诸如,例如,可以被配置成部分地基于查询输入来返回结果的搜索引擎之类的搜索组件 102。例如,搜索组件 102 可以操作以使用一个单词、多个单词、短语、及其他数据来定位候选文档。搜索组件 102 可以操作以定位信息,并可以被操作系统(OS)、文件系统、基于 web 的系统,或其他系统使用。搜索组件 102 也可以作为插件组件被包括,其中,搜索功能可以被主机系统或应用程序使用。如此处进一步描述的,搜索组件 102 还可以使用垃圾分数作为对候选文档进行排序的排序特征。

[0020] 搜索组件 102 可以被配置成提供可以与文档相关联的搜索结果(例如,统一资源定位符(URL))。例如,当返回与本地文件、远程联网文件,本地和远程文件的组合等等相关联的搜索结果时,搜索组件 102 可以使用文本、属性信息,格式,和/或元数据。在一个实施例中,当提供搜索结果时,搜索组件 102 可以与文件系统、虚拟 web、网络或其他信息源进行交互。

[0021] 搜索组件 102 包括排序组件 104,该排序组件 104 可以被配置成至少部分地基于排序算法 106 和一个或多个排序特征 108,对搜索结果(诸如候选文档)进行排序。在一个实施例中,排序算法 106 可以被配置成提供可以被搜索组件 102 用于排序目的的多个其他变量。排序特征 108 可以被描述为当标识搜索结果的相关性时可以使用的基本输入或原始数字。排序特征 108 可以被收集、存储,和维护在数据库组件 110 中。

[0022] 可另选地,诸如垃圾分数之类的排序特征 108 可以被存储和维护在专用存储器中,包括本地、远程,及其他存储介质。排序特征 108 中的一个或多个可以被输入到排序算法 106,而排序算法 106 可以操作以作为排序判断的一部分来对搜索结果进行排序。如下面

所描述的,在一个实施例中,排序组件 104 可以作为排序判断的一部分,使用一个或多个排序特征 108。

[0023] 相应地,当作为排序判断的一部分使用排序特征 108 中的一个或多个时,搜索组件 102 可以使用排序组件 104 以及相关联的排序算法 106 来提供搜索结果。可以基于相关性排序或某种其他排序,提供搜索结果。例如,搜索组件 102 可以至少部分地基于由排序组件 104 使用包括垃圾分数的排序特征 108 中的一个或多个提供的相关性判断,从最相关到最不相关,呈现搜索结果。

[0024] 继续参考图 1,系统 100 还包括可以被用来索引信息的索引组件 112。索引组件 112 可以被用来索引和编目要存储在数据库组件 110 中的信息。此外,当针对多个完全不同的信息源进行索引时,索引组件 102 可以使用元数据、内容和 / 或其他文档数据。例如,索引组件 112 可以被用来构建将关键字及其他文档数据映射到文档(包括与文档相关联的 URL)的倒排索引数据结构。

[0025] 当根据由排序组件 104 所提供的排序返回相关搜索结果(诸如候选文档)时,搜索组件 102 可以使用被索引的信息。在一个实施例中,作为搜索的一部分,搜索组件 102 可以被配置成标识一组包含诸如例如关键字和短语之类的用户的查询信息的一部分或全部的候选文档。例如,查询信息可以位于文档的正文或元数据或与文档相关联的额外的元数据中,该额外的元数据可以被存储在其他文档或数据存储中(诸如例如锚文本)。如下面所描述的,如果整组搜索结果比较大,则并非返回该整组搜索结果,搜索组件 102 可以使用排序组件 104 来就相关性或某种其他准则而论对候选进行排序,并至少部分地基于排序判断,返回整组的一子集。然而,如果该组候选不太大,则搜索组件 102 可以操作以返回整个组。

[0026] 在一个实施例中,排序组件 104 可以使用排序算法 106 来根据相关性,排序与特定查询相关联的候选文档。例如,排序算法 106 可以计算与候选搜索结果相关联的排序值,其中,较高的排序值对应于更为相关的候选。可以将包括一个或多个排序特征 108(诸如垃圾分数)的多个特征输入到排序算法 106 中,然后,排序算法 106 可以计算允许搜索组件 102 按排序或某种其他准则来对候选进行排序的输出。搜索组件 102 可以使用排序算法 106 通过根据排序来限制候选组,来防止用户不得不检查整组候选,诸如例如企业 URL 集合。

[0027] 在各实施例中,当返回候选文档时,搜索组件 102 计算并选择候选文档的一个或多个垃圾分数,作为相关性判断的一部分。候选文档可以具有零个或多个与它们相关联的垃圾分数,使用垃圾分数对候选文档进行排序可以包括平均化、忽略,或查找候选文档的一组垃圾分数之中的最大值或最小值。

[0028] 在一个实施例中,搜索组件 102 可以使用诸如 MICROSOFT OFFICESHAREPOINT **SERVER**®系统之类的集成的服务器平台的功能,来计算、收集、存储,以及更新垃圾分数及可以被用作排序判断的一部分的其他排序特征 108。服务器平台的功能可包括 web 内容管理、企业内容服务、企业搜索、共享的业务进程、商务智能服务,及其他服务。例如,如此处所描述的,使用 MICROSOFT OFFICE SHAREPOINT **SERVER**®系统创建的模板,可以被用来收集已知垃圾文档的参考信息。

[0029] 如下面所描述的,作为相关性判断的一部分,可以使用两层神经网络。在一个实施例中,两层神经网络的实现包括培训阶段和排序阶段,作为使用两层神经网络的前向传播过程的一部分。在培训阶段,可以使用 LambdaRank 作为培训算法,并可以使用神经网络前

向传播模型作为排序判断的一部分(参见Schölkopf、Platt 和 Hofmann 编著的(Ed.) 神经信息处理系统中的进步 19, 2006 年会议学报(MIT 出版社, 2006) 中的 C. Burges、R. Ragno、Q. V. Le 所作的“Learning To Rank With Nonsmooth Cost Functions (学习用非平滑成本函数来排序)”, 该文的全部内容通过参考结合于此)。例如, 作为排序阶段的一部分, 可以使用标准神经网络前向传播模型。可以将一个或多个垃圾分数用作排序特征 108, 并且结合两层神经网络作为基于用户查询来对候选文档排序的一部分。

[0030] 在一个实施例中, 排序组件 104 利用排序算法 106, 该排序算法 106 包括两层神经网络打分函数(此处还称为“打分函数”), 该“打分函数”包括:

$$[0031] \quad \text{分数}(x_1, \dots, x_n) = \left(\sum_{j=1}^m h_j \cdot w_{2j} \right) \quad (1)$$

[0032] 其中,

$$[0033] \quad h_j = \tanh \left(\left(\sum_{i=1}^n x_i \cdot w_{ij} \right) + t_j \right) \quad (1a)$$

[0034] 其中,

[0035] h_j 是隐藏节点 j 的输出,

[0036] x_i 是来自输入节点 i 的输入值, 诸如一个或多个排序特征输入,

[0037] w_{2j} 是向隐藏节点输出应用的权重,

[0038] w_{ij} 是应用于隐藏节点 j 输入的值 x_i 的权重,

[0039] t_j 是对于隐藏节点 j 的阈值,

[0040] 以及, \tanh 是双曲正切函数:

$$[0041] \quad h_j = \tanh \left(\left(\sum_{i=1}^n x_i \cdot w_{ij} \right) + t_j \right) \quad (1c)$$

[0042] 在一个实施例中, 上面可以使用具有与 \tanh 函数类似的属性和特征的其他函数。在各实施例中, 变量 x_i 可以表示一个或多个垃圾分数或其他排序特征。作为相关性判断的一部分, 在排序之前, 可以使用 λ 排序培训算法来培训两层神经网络打分函数。此外, 可以将新特征和参数添加到打分函数中, 而不会显著影响培训准确性或培训速度。

[0043] 当返回基于用户查询的搜索结果时, 当进行相关性判断时, 对于此实施例, 可以输入一个或多个排序特征 108, 并由排序算法 106, 两层神经网络打分函数使用。在各实施例中, 当作为返回基于用户查询的搜索结果的一部分作出相关性判断时, 可以输入一个或多个垃圾分数, 并由排序算法 106 用作排序特征 108。

[0044] 当排序和提供搜索结果时, 也可以使用其他特征。在一个实施例中, 点击距离(CD)、URL 深度(UD)、文件类型或以前的类型(T)、语言或以前的语言(L), 元数据、BM25F, 和 / 或其他排序特征可以被用来排序和提供搜索结果。在 2007 年 10 月 18 日提交的标题为“Ranking and Providing Search Results Based in Part on a Number of Click Through Parameters (部分地基于点进参数来排名和提供搜索结果)”的美国专利申请第 11/874579 号和 2007 年 10 月 18 日提交的标题为“Enterprise Relevancy Ranking Using a Neural Network (使用神经网络的企业相关性排名)”的美国专利申请第 11/874844 号中提供了关于使用两层神经网络来基于排序特征对搜索结果进行排序(包括对排序特征的转换和归一

化)的更多细节,这两个申请的全部内容通过引用结合于此。在各实施例中,可以使用其他类型的排序算法 106。例如,包括垃圾分数在内的这些(或额外的)排序特征 108 中的一个或多个也可以被用作由排序组件 104 所使用的线性排序判断或其他排序算法 106 的一部分。

[0045] 图 2 示出了用于确定并使用垃圾分数作为排序特征以响应于搜索查询来对候选文档进行排序的方法 200 的实施例。在各实施例中,图 2 所示出的方法 200 的步骤以及此处的其他附图可以以不同的顺序执行,并且可以添加、消除,或组合步骤。图 2 的方法可以由诸如系统 100 之类的系统来执行。在步骤 201 中,创建语料库中的文档的索引。在各实施例中,索引是将文档数据映射到语料库内的文档的倒排索引。如此处所使用的,文档数据可以包括单词、数字、短语、文本、格式、元数据,及文档内的其他人可读取的和非人可读取的数据。另外,语料库可以是爬取以创建索引的任何文档集合。如此处所使用的,文档包括文字处理文档、电子表格、网站、列表、文档库、web、演示文稿或其他文件。语料库可以通过特定网络(因特网、外部网,或其他网络),站点,或其他群组内的文件的集合来定义。在各实施例中,优选情况下,可以使用本系统和方法来标识使用特定模板来创建文档的语料库内的可能的垃圾文档。例如,MICROSOFT OFFICE SHAREPOINT **SERVER**®系统包括标准文档模板,并准许用户来定义他们的 MICROSOFT OFFICE SHAREPOINT **SERVER**®环境特定的文档模板。

[0046] 图 3 示出了用于在步骤 201 中创建索引的方法 300 的实施例,该方法 300,在各实施例中,可以由系统的诸如索引组件 112 之类的索引组件来执行。在此实施例中,参考自动地生成的数据的词典,创建索引。在步骤 301 中,爬取文档语料库。在步骤 302 中,选择自动地生成的数据的词典。

[0047] 在各实施例中,可以通过使用与语料库相关联的系统来创建空白文档,来创建自动地生成的数据的词典。如此处所使用的,“空白”意味着,当创建文档时,除由与语料库相关联的系统自动创建和插入在文档中的文档数据以外,实质上缺少文档数据。例如,MICROSOFT OFFICE SHAREPOINT **SERVER**®系统准许用户定义特定文档库的模板,而文档库可以包括可以如此处所阐述的被索引,查询和排序的文档语料库。利用这样的模板创建的空白文档将包括由 MICROSOFT OFFICE SHAREPOINT **SERVER**®系统自动地生成的某些文档数据(诸如文本、格式、元数据等等)。然后,可以通过提取和编译空白文档中的文档数据来创建自动地生成的数据的词典。还可以通过检查语料库中的现有文档中的某些或全部,并标识对话料库中的相当大的比例的文档公共的文档数据,来创建或扩充词典。词典可以包括从空白文档中所提取的文档数据的内容和位置信息两者。

[0048] 在各实施例中,可以为用于生成空白文档的不同的语料库,不同的模板,以及不同的系统创建自动地生成的数据的不同的词典。另外,可以通过准许用户指定要使用的特定空白文档,给用户(诸如管理员)提供创建要对于特定类型的文档使用的新词典的能力。例如,可以提供包括由诸如 MICROSOFT OFFICE SHAREPOINT **SERVER**®系统之类的特定系统所生成的预定义的模板的计算机可读存储介质。例如,一个这样的预定义的模板可以包括联系人管理模板,而第二这样的预定义的模板可以是销售领先管道化模板。为这些模板中的每一个自动地生成的数据的词典可以被编译并作为系统的一部分预先加载。然而,管理员可以使用系统来自定义现有模板或创建新自定义模板。在各实施例中,可以提示管理员(或系统可以自动地)使用这样的自定义模板来生成空白文档,并编译从这样的自定义模板

自动地生成的数据的词典。如此,如下面所阐述的,在各实施例中,可以使用正在被索引的特定文档所特定的自动地生成的数据的词典。

[0049] 在步骤 302 中,选择至少一个自动地生成的数据的词典。如上文所描述的,取决于语料库中的文档的类型,也可以有一个以上的词典可用于选择。在各实施例中,如果语料库中的所有文档是使用同一个系统并使用同一个模板创建的,那么,可以选择单个词典并将其用于语料库中的所有文档。可另选地,在各实施例中,语料库可以包括由完全不同的系统或使用不同的模板创建的文档,可以为不同的文档选择不同的词典。另外,在各实施例中,可以定义组合了使用完全不同的系统或跨语料库的模板创建的文档的内容和位置信息两者的单个词典。在各实施例中,所选一个或多个词典可以被视为垃圾简档的一部分。

[0050] 在步骤 304 中,将爬取的文档的文档数据对照自动地生成的数据的所选一个或多个词典进行比较。在各实施例中,将每一个爬取的文档的文档数据对照为该文档选择的词典(或多个词典)进行比较,以确定什么文档信息可能是由用于创建该爬取的文档的系统(和/或模板)自动地生成的。例如,用于创建文档的系统所定义的模板可以在使用该模板创建的每个文档的标题中自动地包括单词“Task (任务)”。单词“Task”以及其在模板内的位置包括在为该文档选择的自动地生成的数据的词典中。在将文档与词典进行比较之后,文档的标题中的单词“Task”可以被确定为“匹配”所选词典中的对应的条目。如此处所使用的,“匹配”可以包括内容、位置或两者的准确的或显著的关联度。另外,在各实施例中,将文档数据匹配到自动地生成的数据的词典(或多个词典)可以被视为如这里所描述的计算垃圾分数。

[0051] 在步骤 306,描绘了匹配自动地生成的数据的所选词典的文档数据。在各实施例中,这样的匹配文档数据可以通过标记文档或索引中的匹配文档数据,与非匹配文档数据分开地索引匹配文档数据,忽略匹配文档数据并只索引非匹配文档数据,或通过其他方法来描绘。

[0052] 考虑下列简单示例。虽然此示例是使用文本文档数据来提供的,但是,也可以使用任何文档数据(例如,元数据、格式,非人可读取的数据等等)。给定文本 T0=“you know what it is”;T1=“what is it”;T2=“it is abird”,通常将创建下列完整的倒置文件索引(其中,一对数字指代文档编号(Tx)和单词位置)。例如,单词“bird”位于第三文档(T2),它是该文档中的第四单词(位置 3):

[0053] “a”:[(2, 2)]

[0054] “bird”:[(2, 3)]

[0055] “is”:[(0, 4), (1, 1), (2, 1)]

[0056] “it”:[(0, 3), (1, 2), (2, 0)]

[0057] “know”:[(0, 1)]

[0058] “what”:[(0, 2), (1, 0)]

[0059] “you”:[(0, 0)]

[0060] 现在假设为所有三个文档选择了同一个词典,而所选词典包括位置 1 处的单词“is”(例如,因为使用用于创建文档 0, 1, 和 2 的同一个系统和模板创建的空白文档在位置 1 处包含单词“is”)。在各实施例中,位置 1 处的单词“is”匹配文档 1 和 2 中的每一个中的文档数据。可以以多种方式来描绘该匹配文档数据。例如,在索引中可以忽略来自文档

1 和 2 的匹配文档数据。在此实施例中,上面的示例的索引将变为:

[0061] "a":[(2, 2)]

[0062] "bird":[(2, 3)]

[0063] "is":[(0, 4)]

[0064] "it":[(0, 3), (1, 2), (2, 0)]

[0065] "know":[(0, 1)]

[0066] "what":[(0, 2), (1, 0)]

[0067] "you":[(0, 0)]

[0068] 如此,将响应于搜索查询被进行搜索的索引被最小化(用于更快的搜索),并更加聚焦于用户添加的内容(而非由用于创建文档的系统或模板自动地添加的文档数据)。在其他实施例中,匹配文档数据可以被分开地索引,以便可以针对主要索引(从非匹配文档数据导出的)和辅助索引(从匹配文档数据导出的)两者运行随后的搜索查询。因此,在各实施例中,诸如排序算法 106 之类的排序算法,可以给搜索查询项在主索引中的出现比这样的项在辅助索引中的出现赋予更重要的权重。

[0069] 在步骤 308,索引语料库中的文档。在各实施例中,组合步骤 306 和 308,匹配所选词典的文档数据可以通过如上文所讨论的这样的文档数据(以及文档)被索引(或不被索引)的方式被描绘为匹配。

[0070] 在步骤 310,为语料库中的爬取的文档中的某些或全部计算垃圾分数。在各实施例中,在步骤 310,计算垃圾分数,该垃圾分数是爬取的文档和用于创建自动地生成的数据的所选词典的空白文档之间的相似度量度的函数(诸如库尔贝克-莱布勒发散性)。例如,相似度量度可以包括匹配所选词典中的对应的条目的文档数据与不匹配所选词典中的对应的条目的文档数据的比率。在各实施例中,相对较高的相似度量度表示除用于创建所选词典的空白文档中的东西以外不包括许多文档数据的文档。这可以被视为文档可能是“垃圾”的指示,可以将对应的垃圾分数指定给文档,并作为诸如排序特征 108 之类的排序特征存储数据库组件 110 中。垃圾分数可以包括相似度量度本身,或者也可以包括相似度量度的函数,以便归一化和使用垃圾分数作为排序特征。

[0071] 在各实施例中,如果将文档与一个以上的选择的词典进行比较 304,则可以根据文档针对所选词典的相似度量度的平均值、最大值、最小值的函数或其他计算值,来计算 310 垃圾分数。在步骤 310 计算出的垃圾分数可以被用作如此处所提供的排序特征。

[0072] 回头参考图 2,在步骤 202,使用垃圾简档来计算一个或多个垃圾分数。在各实施例中,对垃圾分数的计算可以与创建 201 语料库中的文档的索引相结合地进行。例如,如所讨论的,爬取的文档和用于创建自动地生成的数据的所选词典(或多个词典)的空白文档之间的相似度量度可以被用来确定垃圾分数。在其他实施例中,在步骤 310 计算出的垃圾分数可以不计算,或也可以与使用来自垃圾简档的垃圾变量计算 202 垃圾分数组合或者作为额外的排序特征相结合与其相结合。

[0073] 如此处的各实施例中所使用的,垃圾变量可以意味着可以被用来确定一个文档和已知垃圾文档之间的相似度的变量。例如,如下面所讨论的,在各实施例中,垃圾变量可以包括词频。在其他实施例中,垃圾变量可以包括块大小。垃圾变量可以个别地使用或也可以组合起来使用。

[0074] 在步骤 202,在各实施例中,根据为文档创建的一个或多个直方图和垃圾简档之间的相似度量来计算文档的垃圾分数,其中,垃圾简档包括已知垃圾文档的一个或多个直方图。图 4 示出了用于使用一个或多个垃圾简档来计算 202 文档的垃圾分数的方法 400 的一个实施例。

[0075] 在各实施例中,当在步骤 201 索引文档的同时,或否则,在接收搜索查询之前,对于语料库中的所有文档,执行方法 400。在其他实施例中,可以在接收到搜索查询并标识了候选文档之后,执行方法 400。方法 400 是参考一个文档来描述的,然而,可以对例如语料库中的任何或所有文档,或者对响应于搜索查询标识的候选文档,重复方法 400。

[0076] 在步骤 401,在各实施例中,基于垃圾变量,为文档生成至少一个直方图。如此处所使用的,直方图可以是所定义类别(或柱)内的变量的表示。在其他实施例中,可以使用垃圾变量的分布的替换的表示,来代替直方图或作为其补充。例如,可以计算或估计连续函数,来表示这样的分布,而不将它转换为直方图。此处被分析的文档的直方图被描述为“候选直方图”。

[0077] 在方法 400 所示出的示例实施例中,在步骤 401,为文档生成候选直方图。在下面的示例中,候选直方图基于文档的词频(例如,文档中的具有语料库中的对应的出现频率的唯一检索词的百分比)。例如,在一个简单示例中假设文档包括四个检索词:T1、T2、T3,以及 T4。按如下方式,示出了每一个检索词的示例总的语料库频率(即,语料库中包含此检索词的文档的总数):

[0078] T1:10

[0079] T2:300

[0080] T3:100000

[0081] T4:50

[0082] 可以相对于类别或“柱”定义此文档的词频候选直方图。在此示例中,为总的语料库频率定义了四个柱:[1... 20],[21... 400],[401... 12000],[12001... 最大]。因此,此文档的词频候选直方图可以被表示为:[0.25, 0.5, 0.0, 0.25]。这反映了四个检索词中的一个落入第一柱内,四个检索词中的两个落入第二柱内,四个检索词中没有一个落入第三柱内,而四个检索词中的一个落入第四柱内。

[0083] 在步骤 402,将候选直方图与至少一个垃圾简档进行比较。除上文参考图 2 和 3 所描述的垃圾简档之外,垃圾简档可以进一步包括已知垃圾文档的一个或多个直方图。它还可以包括逼近已知垃圾文档的表示的一个或多个规则。在不同的实施例中,可以以不同的方式定义“垃圾文档”。例如,在某些实施例中,垃圾文档包括整体来看包括较大比例的非人可读取的文档数据的文档。在其他实施例中,垃圾文档可以包括几乎是空的文档。在各实施例中,出于创建参考直方图的目的,管理员被准许(例如,通过用户界面 103)将特定现有的文档定义为“已知垃圾”。

[0084] 例如,参考上文参照步骤 401 所讨论的简单示例,假设已知垃圾文档包括下列检索词(带有每一个检索词的总的语料库频率):

[0085] T1:10

[0086] T3:100000

[0087] T5:500

[0088] T6:1000

[0089] T7:12

[0090] 可以相对于相同类别或“柱”，将此已知垃圾文档的词频参考直方图定义为候选直方图： $[1...20]$, $[21...400]$, $[401...12000]$, $[12001...最大]$ 。因此，此已知垃圾文档的词频参考直方图可以被表示为： $[0.4, 0.0, 0.4, 0.2]$ 。这反映了五个检索词中的两个落入第一柱内，五个检索词没有一个落入第二柱内，五个检索词中两个落入第三柱内，而五个检索词中的一个落入第四柱内。

[0091] 候选直方图与垃圾简档的比较 402（在此示例中，垃圾简档包括参考直方图）可以采用许多形式。例如，可以通过将相似度度量计算为候选直方图和参考直方图之间的距离来比较直方图：

[0092]

$$\text{相似度度量} = \sqrt{\sum_0^n (B1(i) - B2(i))^2}$$

[0093] 在此示例比较函数中，B1 是候选直方图，B2 是参考直方图，B1(i) 是第 i' 个柱的候选直方图的值，而 B2(i) 是第 i' 个柱的参考直方图的值。此计算产生 0 和 1 之间的相似度度量。

[0094] 再次参考上面的示例，比较步骤 402 按如下方式计算候选直方图和参考直方图之间的相似度度量：

$$\text{相似度度量}(B1, B2) = \sqrt{(4-.25)^2 + (0.0-.5)^2 + (4-0.0)^2 + (.2-.25)^2} \approx .66$$

[0096] 在此示例计算中，候选直方图和参考直方图越相似，相似度度量越靠近零（如此，表示文档是垃圾的可能性越高）。

[0097] 在步骤 404，为文档计算至少一个垃圾分数。在各实施例中，垃圾分数可以包括相似度度量本身。在其他实施例中，相似度度量可以被转换为不同的比例。另外，步骤 402 中所使用的垃圾简档可以包括一个以上的参考直方图。例如，一个以上的已知垃圾文档可以被用来创建垃圾简档。在各实施例中，垃圾分数可以包括在比较步骤 402 过程中计算出的多个相似度度量的平均值、加权平均值、最大值、最小值的函数或某种其他函数。另外，可以基于诸如词频之类的垃圾变量，来计算文档的一个以上的垃圾分数，如此处进一步描述的，所有这样的垃圾分数都可以被排序算法使用。垃圾分数可以作为元数据与文档本身一起存储，与文档分开存储，或以其他方式存储。

[0098] 在步骤 406，就是否请求了警告作出判断。例如，在各实施例中，管理员可以在计算出了超出某一阈值的垃圾分数时请求警告。应该理解，取决于特定比例是如何定义的，如此处所使用的“超出阈值”可以是指落在特定阈值之上或之下的测量（诸如垃圾分数）。如果没有请求警告，则方法 400 可以结束（在各实施例中，控制返回到图 2 中的步骤 204）。如果请求了警告，则就计算出的垃圾分数是否超出阈值作出判断 408。在各实施例中，阈值是由管理员调节的。如果没有超出阈值，则方法 400 可以结束。如果超出了阈值，则发送警告 410。警告可以采用不同的形式，包括电子邮件、声音消息、文本等等，并可以向用户、管理员等等发送。如此，可以向管理员及其他人警告可能需要删除的垃圾文档。另外，还可以使用垃圾分数作为可搜索的属性。例如，用户可能希望搜索具有高垃圾分数的文档，因为它们可能是用于存档的好的候选。另一个用户可能希望使用垃圾分数作为搜索准则，自动地使带

有高于特定阈值的垃圾分数的文档从返回的搜索结果中被过滤出来。在其他实施例中,可以由管理员或用户查询垃圾分数,以便发现是用于删除的候选的文档。

[0099] 参考使用基于一个垃圾变量的文档的直方图描述了方法 400,然而,可以对于使用其他垃圾变量的直方图来重复方法 400。例如,可以被用来预测文档是否是垃圾的另一种垃圾变量是“块大小”。块大小是连续的文本的长度。诸如电子表格之类的某些文档类型具有大量的非人可读取的数据。可以预期诸如文字处理文档之类的其他文档具有比较长的连续的文本的块。基于文档的块大小的直方图可以示出块大小在文档内的分布(连续的文本的遍数(run))。取决于文档类型,可以通过扫描文档的文本并测量逻辑中断(诸如单元格、段落、句子、分页等等)之间的距离来以不同的方式测量块。在各实施例中,文档内的块大小分布和已知垃圾文档中的块大小分布之间的相似度是文档是否应当被表征为垃圾的指示符。

[0100] 基于不同的直方图和 / 或垃圾变量的文档和垃圾简档之间的相似度度量可以被合并到单个垃圾分数中,或被用作单个垃圾分数,这些分数被诸如排序算法 106 之类的排序算法用作排序特征。

[0101] 再次参考图 2,方法 200 在步骤 204 继续,在那里,接收搜索查询。在各实施例中,用户可以使用诸如用户界面 103 之类的用户界面来输入搜索查询。搜索查询可以包括关键字、短语或其他搜索参数,包括非文本搜索参数(诸如格式等等)。在步骤 206,标识候选文档。例如,如参考图 1 所描述的,诸如搜索组件 102 之类的搜索组件可以返回匹配搜索查询的候选文档。

[0102] 在步骤 208,对候选文档进行排序。如参考图 1 所描述的,可以使用排序组件 104,使用一个或多个排序算法 106 和一个或多个排序特征 108,对候选文档进行排序。如此处所阐述而计算出的垃圾分数可以被用作排序特征 108。也可以使用其他排序特征 108。在各实施例中,单个文档的排序如此可以受文档的垃圾分数的影响。如此,原本由于文档紧密地匹配搜索查询项而排在较高的位置的文档可能由于高的垃圾分数而排序得较低。在其他实施例中,对文档的实际排序不受垃圾分数的影响。相反,使用垃圾分数来向用户提供可能的垃圾的指示,而不会影响列出文档的顺序。在步骤 210,呈现了排序的候选文档。例如,可以按照文档排序的顺序,向用户显示文档的子集(例如,开头十个)。在各实施例中,具有超出阈值的垃圾分数的文档可能根本不呈现(例如,可以删去,不作为候选文档)。在其他实施例中,文档的垃圾分数与候选文档一起呈现(例如,在候选文档的排序的列表中),以便用户可以作出是否要点击具有特定垃圾分数的文档的独立判断。在其他实施例中,可以使用垃圾阈值。如果文档的垃圾分数超出垃圾阈值,则该文档可以与该文档可能是垃圾的指示一起显示。例如,如果垃圾分数超出垃圾阈值,则可以在候选文档的排序的列表中为该文档显示垃圾符号或实际垃圾分数。如此处所使用的,显示的垃圾分数或垃圾符号应该被视为“垃圾状态”。

[0103] 在各实施例中,本系统和方法对检测“偶然的垃圾文档”(并非包含恶意的或不希望有的信息的文档,即,“敌手的垃圾”)有用。例如,用户可能已经开始创建演示文稿,插入了标题,保存了演示文稿,而没有添加任何额外的内容,然后,忘记了这件事。通常,响应于包含用于该演示文稿的标题中的项的搜索查询,搜索组件可能将返回该演示文稿,并将它排在很高的位置。然而,检测到演示文稿类似于已知垃圾文档(例如,具有类似的词频分布),允许搜索组件将该演示文稿排在较适当的位置。

[0104] 图 5 示出了其中可以实现软件实施例的合适的操作环境 500 的一个示例。这只是合适的操作环境的一个示例,并非旨在对使用范围或功能提出任何限制。适用的其他公知计算系统、环境和 / 或配置包括但不限于个人计算机、服务器计算机、手持式或膝上型设备、多处理器系统、基于微处理器的系统、可编程消费电子产品、网络 PC、小型机、大型计算机、包括以上系统或设备的任一个的分布式计算环境等等。

[0105] 在其最基本配置中,操作环境 500 通常包括至少一个处理单元 502 和存储器 504。取决于计算设备的确切配置和类型,存储器 504(存储,其中,如此处所描述的计算出垃圾分数)可以是易失性(如 RAM)、非易失性(如 ROM、闪存等)或是两者的某种组合。该最基本配置在图 5 中由虚线 506 来示出。此外,环境 500 还可包括存储设备(可移动 508 和 / 或不可移动 510),包括但不限于磁盘、光盘或磁带。类似地,环境 500 也可包括一个或多个输入设备 514,如键盘、鼠标、笔、语音输入设备等等,和 / 或输出设备 516,如显示器、扬声器、打印机等等。环境还可以包括一个或多个通信连接 512,如 LAN、WAN、点对点等等。

[0106] 操作环境 500 通常至少包括某种形式的计算机可读介质。计算机可读介质是可以被处理单元 502 或构成操作环境的其他设备访问的任何可用的介质。作为示例而非限制,计算机可读介质可包括计算机存储介质和通信介质。计算机存储介质包括以用于存储诸如计算机可读指令、数据结构、程序模块或其他数据之类的信息的任何方法或技术实现的易失性和非易失性、可移动和不可移动介质。计算机存储介质包括但不限于, RAM、ROM、EEPROM、闪存或其他存储器技术、CD-ROM、数字多功能盘(DVD)或其他光存储、磁带盒、磁带、磁盘存储或其他磁存储设备、或者可用于存储所需信息的任何其他介质。通信介质通常以诸如载波或其他传输机制的已调制数据信号来体现计算机可读指令、数据结构、程序模块或其他数据,并包括任意信息传送介质。术语“已调制数据信号”是指具有以在信号中编码信息的方式被设定或改变其一个或多个特征的信号。作为示例而非限制,通信介质包括诸如有线网络或直接线连接之类的有线介质,以及诸如声学、RF、红外及其他无线介质之类的无线介质。上述中任一组合也应包括在计算机可读介质的范围之内。

[0107] 操作环境 500 可以是使用对一个或多个远程计算机的逻辑连接在联网环境中工作的单个计算机。远程计算机可以是个人计算机、服务器、路由器、网络 PC、对等设备或其他公共网络节点,通常包括上文所描述的许多或全部元件。逻辑连接可以包括由可用的通信介质支持的任何方法。这些联网环境在办公室、企业范围计算机网络、内联网和因特网中是常见的。

[0108] 应该了解,本发明的各实施例可以实现成(1)计算机实现的动作序列或在计算系统上运行的程序模块和 / 或(2)计算系统内的互连机器逻辑电路或电路模块。实现是取决于实现本发明的计算系统的性能要求的选择问题。因此,包括相关算法的逻辑操作可以被不同地称为操作、结构设备、动作、或模块。所属领域技术人员将认识到,在不偏离在此处所阐述的权利要求书内所列举的本发明的精神和范围的情况下,这些操作、结构设备、动作和模块可以以软件、固件、特殊用途数字逻辑,以及其任何组合来实现。

[0109] 虽然结合各个示例性实施例描述了本发明,但是,本领域技术人员可以理解,在随后的权利要求书的范围内可以对本发明进行许多修改。因此,本发明的范围不以任何方式受上面的描述的限制,而是完全参考随后的权利要求书来确定。

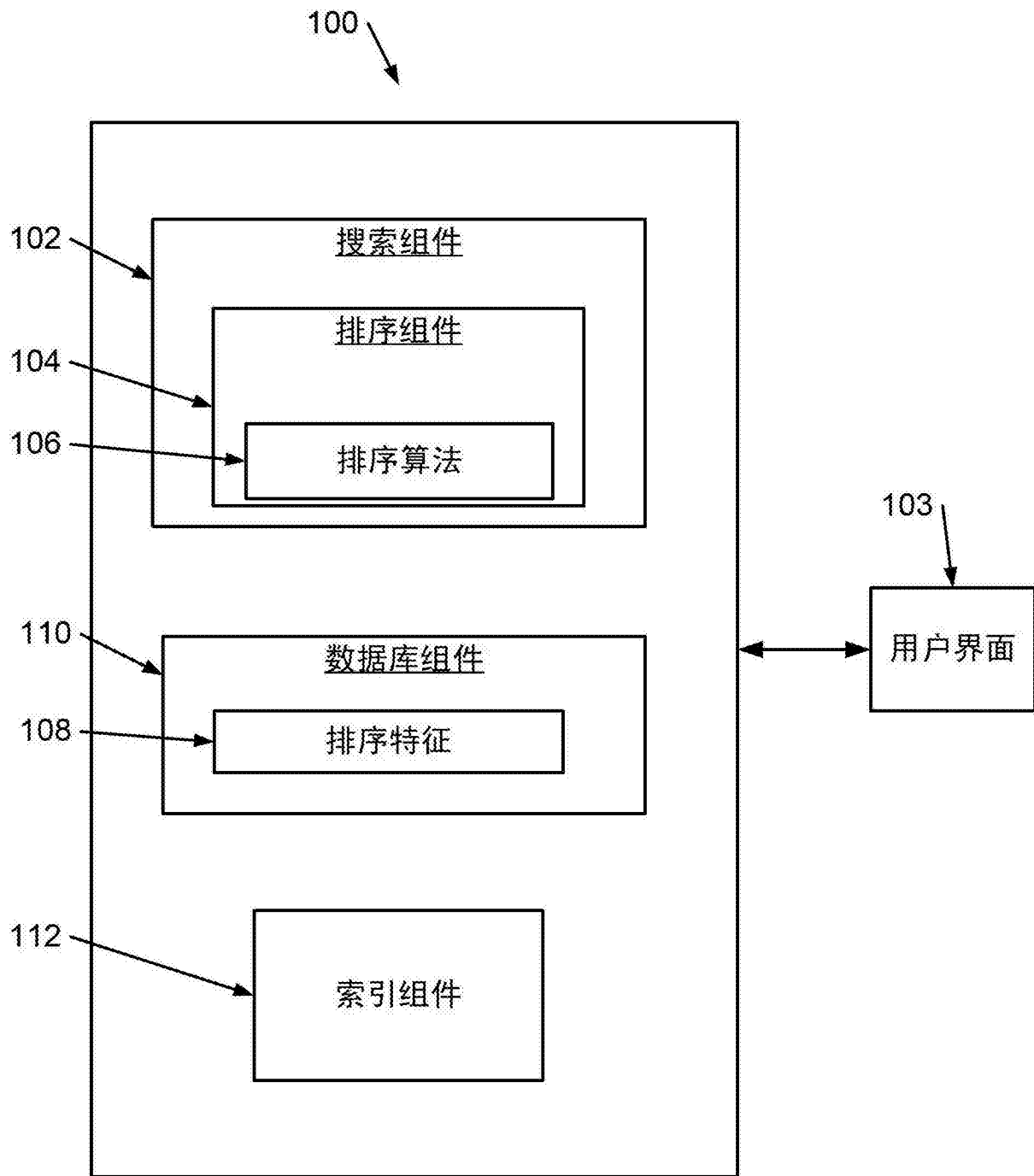


图 1

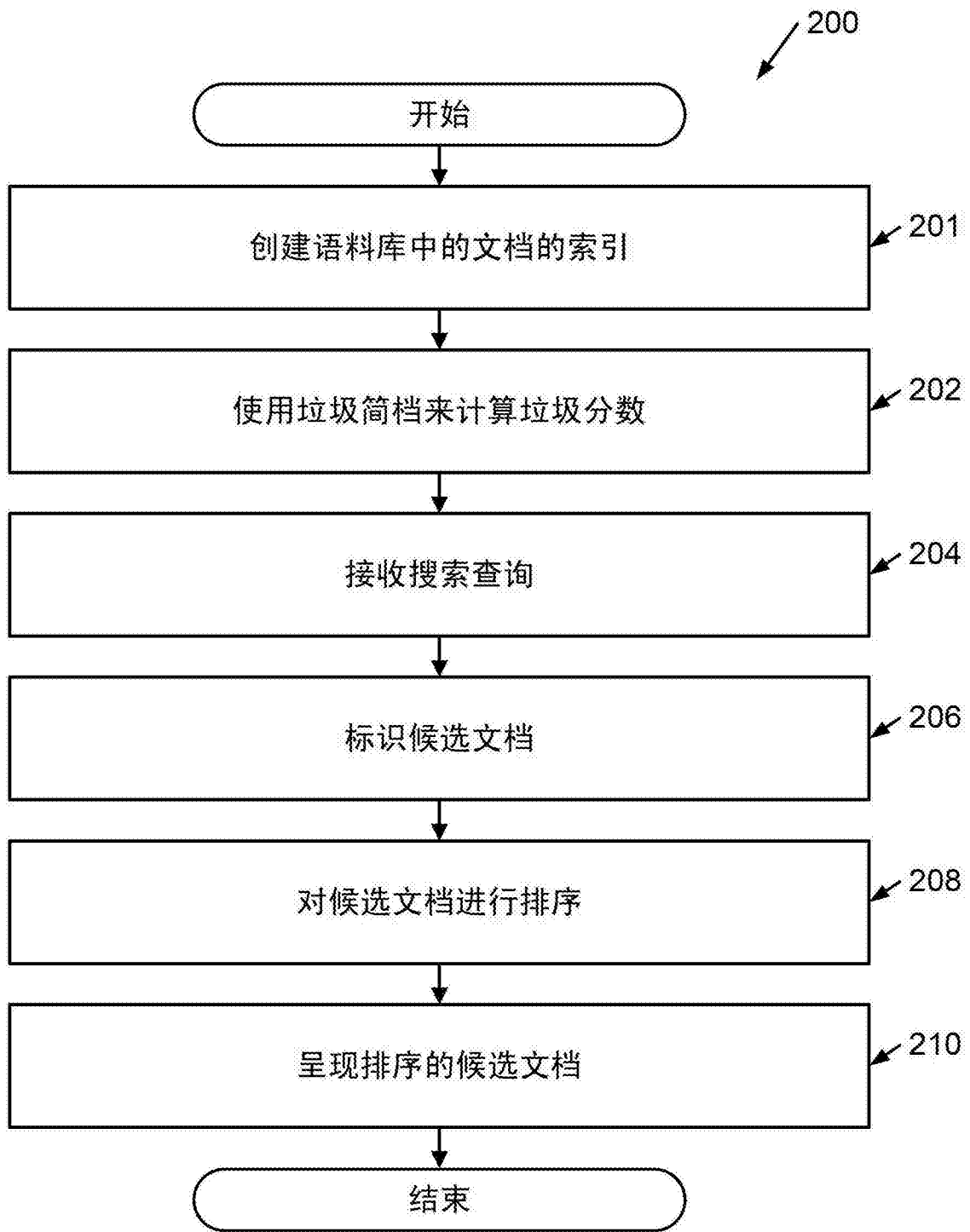


图 2

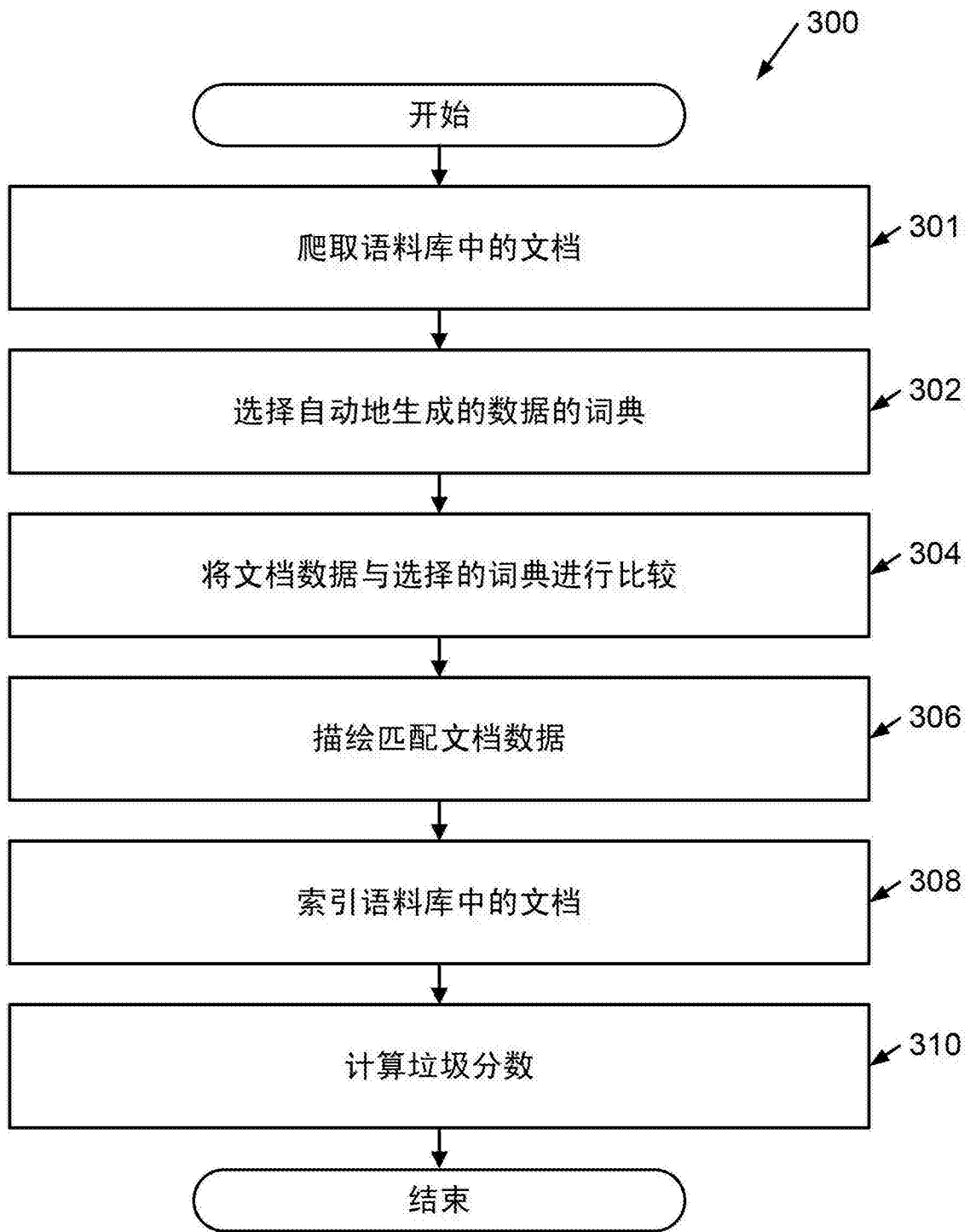


图 3

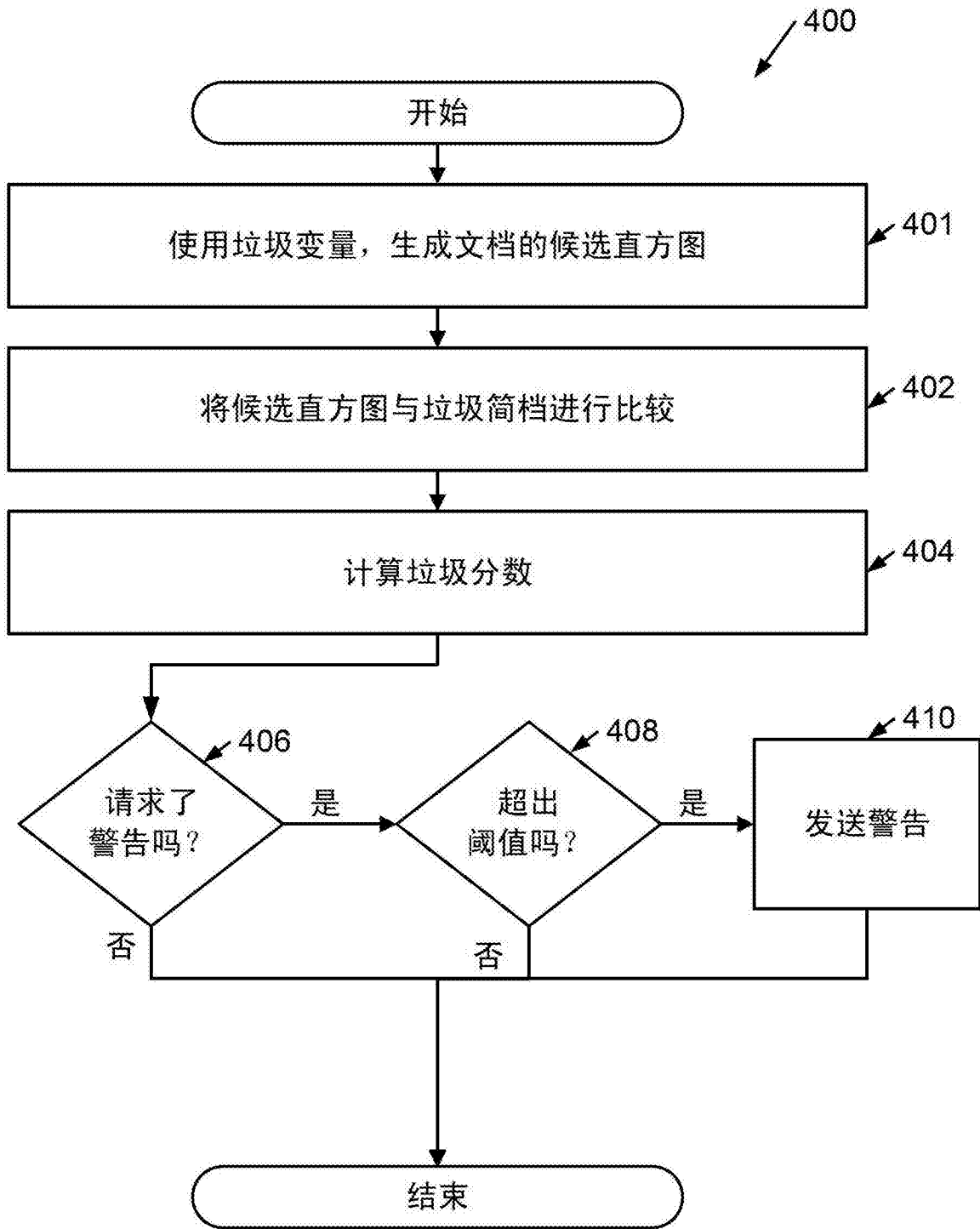


图 4

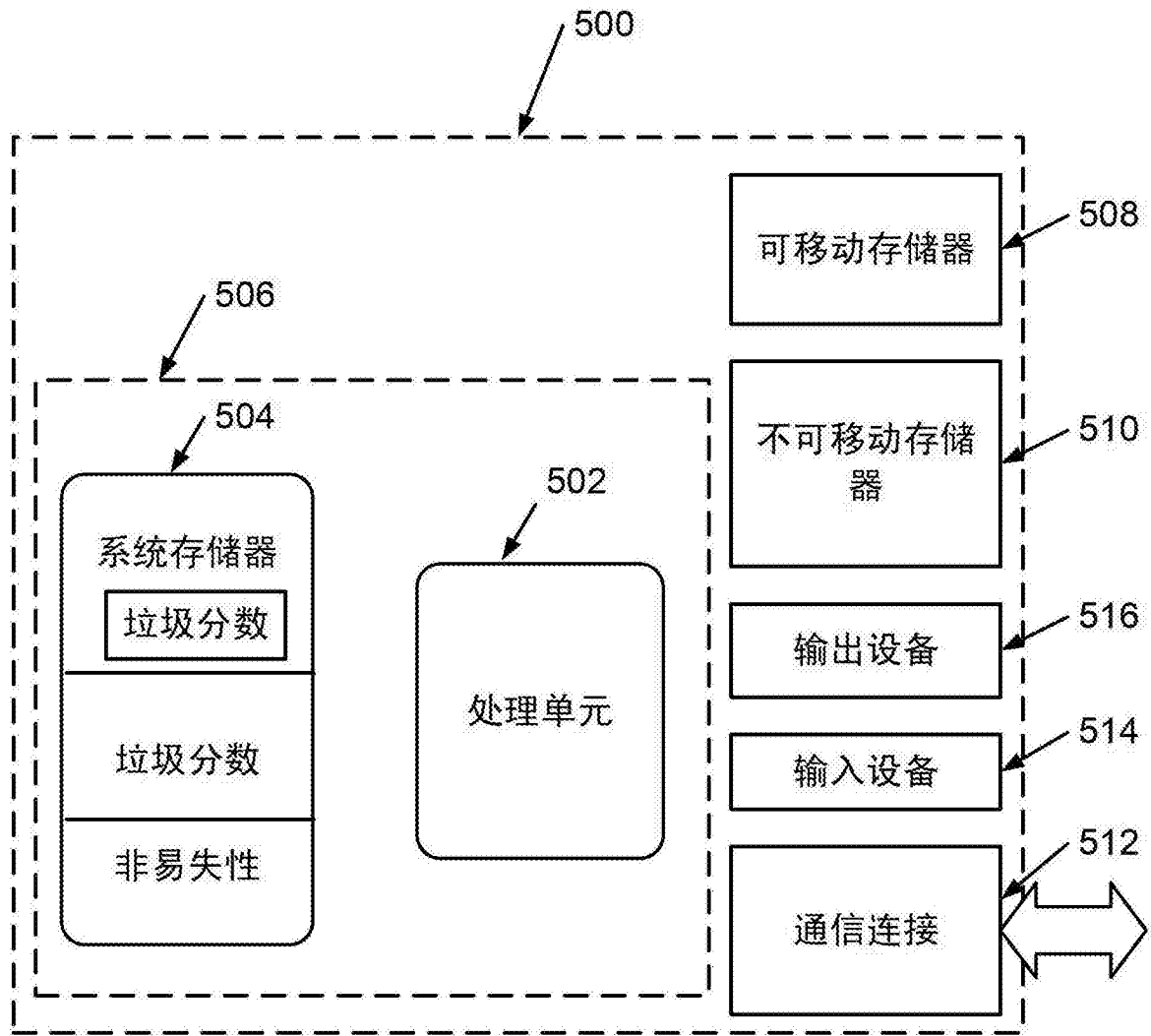


图 5