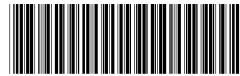


(19) 中华人民共和国国家知识产权局



(12) 发明专利申请

(10) 申请公布号 CN 102915321 A

(43) 申请公布日 2013. 02. 06

(21) 申请号 201210227570. 5

(22) 申请日 2012. 07. 02

(30) 优先权数据

13/173,028 2011.06.30 US

(71) 申请人 波音公司

地址 美国伊利诺伊州

(72) 发明人 L · J · 夸特西 K · M · 纳卡摩德
B · 沃恩

(74) 专利代理机构 北京纪凯知识产权代理有限
公司 11245

代理人 赵蓉民

(51) Int. Cl.

G06F 17/30 (2006. 01)

权利要求书 2 页 说明书 16 页 附图 14 页

(54) 发明名称

用于处理数据的系统和方法

(57) 摘要

本发明提供一种用于处理至少部分未结构化数据的方法。该方法包括在数据处理工具从至少一个数据源接收至少部分未结构化数据，以及处理该至少部分未结构化数据以生成包括标签化数据的至少部分结构化的数据，其中处理至少部分未结构化数据包括以下中的至少一个：利用关联存储器应用程序来处理至少部分未结构化数据；以及利用正则表达式处理程序来处理至少部分未结构化数据。该方法进一步包括传输至少部分结构化数据到主应用程序，以及至少部分基于标签化数据来合并至少部分结构化数据到主应用程序，其中合并该至少部分结构化数据包括基于标签的存在、内容和 / 或类型来进行以下中的至少一个：包括数据和排除数据。

1. 一种处理至少部分未结构化数据的方法,该方法包括 :

在数据处理工具处从至少一个数据源接收至少部分未结构化数据 ;

处理所述至少部分未结构化数据,以实现包括标签化数据的至少部分结构化数据的生成,其中所述标签化数据包括至少一个感兴趣的项目,并且其中处理所述至少部分未结构化数据包括下列中的至少一个 :

利用关联存储器应用程序来处理所述至少部分未结构化数据 ;以及

利用正则表达式处理程序来处理所述至少部分未结构化数据 ;

传输所述至少部分结构化数据到主应用程序 ;以及

至少部分基于所述标签化数据合并所述至少部分结构化数据到主应用程序,其中合并所述至少部分结构化数据包括基于标签的存在、内容和类型中的至少一个来进行以下中的至少一个 :包括数据和排除数据。

2. 根据权利要求 1 所述的方法,其进一步包括 :

验证至少部分结构化数据被正确标签化 ;以及

释放至少部分结构化数据,使得所述至少部分结构化数据可以被合并到所述主应用程序中。

3. 根据权利要求 2 所述的方法,其中验证至少部分结构化数据包括检查所述至少部分结构化数据中的一个或更多识别标签。

4. 根据权利要求 1 所述的方法,其中利用关联存储器应用程序处理至少部分未结构化数据包括 :

将至少部分未结构化数据语法分析成至少部分未结构化数据的一个或更多段 ;

用所述至少部分未结构化数据的至少一个段查询所述关联存储器应用程序 ;

生成与所述至少部分未结构化数据的至少一个段和所述关联存储器应用程序中的数据的至少一个段关联的分数 ;以及

基于所述分数来标签化所述至少部分未结构化数据的所述至少一个段。

5. 根据权利要求 4 所述的方法,其中查询所述关联存储器应用程序包括查询如下关联存储器应用程序,其包括包含样板文件的数据的至少一个段,并且其中标签化至少部分未结构化数据的至少一个段包括标签化至少部分未结构化数据的包括样板文件的至少一个段。

6. 根据权利要求 1 所述的方法,其进一步包括 :

在用户接口显示所述至少部分结构化数据,其中所述至少部分结构化数据包括被不正确标签化和被不正确未标签化中至少一个的错误识别的数据的至少一个段 ;

在所述用户接口接收错误识别的数据的至少一个段的用户选择 ;以及

基于所述错误识别的数据的至少一个段来更新所述数据处理工具 ;

输出至少部分结构化数据到输出表格和输出超文本标记语言(HTML)页中的一个。

7. 根据权利要求 1 所述的方法,其中利用正则表达式处理程序来处理所述至少部分未结构化数据包括 :

应用至少一个源正则表达式模式到至少部分未结构化数据 ;

将所述至少部分未结构化数据的至少一个段和所述至少一个源正则表达式模式匹配 ;

以及

标签化所述至少部分未结构化数据的至少一个匹配段包括用识别标签来标签化至少部分未结构化数据中的至少一个匹配段。

8. 一种用于处理至少部分未结构化数据的系统,所述系统包括:

处理装置;

通信耦合到所述处理装置的用户接口;以及

通信耦合到所述处理装置的存储器和通信耦合到所述处理装置的通信接口中的至少一个,所述处理装置被编程为:

从所述存储器和所述通信接口中的至少一个接收所述至少部分未结构化数据;以及

利用在其上执行的数据处理工具来处理所述至少部分未结构化数据,以通过以下方式中的至少一个实现包括标签化数据的至少部分结构化数据的生成,所述标签化数据包括至少一个感兴趣的项目:

利用在其上执行的关联存储器应用程序来处理所述至少部分未结构化数据;以及

利用在其上执行的正则表达式处理程序来处理所述至少部分未结构化数据;以及

基于标签化来合并所述至少部分结构化数据到主应用程序,其中合并所述至少部分结构化数据包括基于标签的存在来进行以下中的至少一个:包括数据和排除数据。

9. 根据权利要求8所述的系统,其中所述处理装置进一步被编程为:

使所述用户接口显示所述至少部分结构化数据,其中所述至少部分结构化数据包括被不正确标签化和被不正确未标签化中至少一个的错误识别的数据的至少一个段;

接收所述错误识别的数据的至少一个段的用户选择;以及

基于所述错误识别的数据的至少一个段来更新在其上执行的数据处理工具。

10. 根据权利要求9所述的系统,其中利用关联存储器应用程序来处理所述至少部分未结构化数据,所述处理装置进一步被编程为:

将所述至少部分未结构化数据语法分析成所述至少部分未结构化数据的一个或更多段;

用所述至少部分未结构化数据的至少一个段查询在其上执行的所述关联存储器应用程序;

生成与所述至少部分未结构化数据的至少一个段和所述关联存储器应用程序中的数据的至少一个段关联的分数;以及

基于所述分数来标签化所述至少部分未结构化数据的所述至少一个段;

利用正则表达式处理程序来处理所述至少部分未结构化数据,所述处理装置进一步被编程为:

应用至少一个源正则表达式模式到所述至少部分未结构化数据;

将所述至少部分未结构化数据的至少一个段和所述至少一个源正则表达式模式匹配;以及

标签化所述至少部分未结构化数据的至少一个匹配段,输出所述至少部分结构化数据到所述存储器中的输出表格和输出超文本标记语言(HTML)页中的一个,以便经由用户接口显示。

用于处理数据的系统和方法

技术领域

[0001] 本公开的领域总体涉及数据分析,尤其是涉及处理未结构化数据和 / 或部分结构化数据以生成结构化数据,以便由应用程序处理。正如本文所使用的,未结构化数据指为自由形式以及基于生成该数据人员的语法 / 语言而变化的数据。

背景技术

[0002] 在数据分析系统中,数据,例如未结构化文本和 / 或部分结构化文本或其他数据类型(例如字母数字串和非字母数字数据(图像、元数据等))在被添加到系统之前,常常需要被处理和 / 或组织成更结构化的形式。然而,从未结构化文本和 / 或部分结构化数据中识别、语法分析和提取相关信息会是困难的和耗时的。利用类属语法分析器(generic parsers)和 / 或提取器(extractor)来识别这类信息,数据会被忽略、错误识别和 / 或不适当当地解构。

[0003] 为了纠正这些错误,常常编写专用代码以正确地识别该信息。然而,编写和实现这类专用代码会是耗时的,而且得到的代码仅适用特定情形。进一步地,定期更新未结构化文本和 / 或部分结构化数据会加重这些问题,因为其引入了可能需要其他专用代码的新情形。进一步地,专用代码通常仅能由有经验的人员编写和更新。

[0004] 也可以实现自然语言方法来处理和 / 或组织未结构化数据和 / 或部分结构化数据。然而,根据未结构化数据和 / 或部分结构化数据的来源,自然语言在组织未结构化数据和 / 或部分结构化数据时可能不是有效的。进一步的,自然语言方法要求需要本体论(ontology)专家和数据挖掘专家,以便正确地编程和更新。最后,可以使用人工智能工具(例如基于规则的系统、神经网络和 / 或 Bayesian 网络)来处理和 / 或组织未结构化数据和 / 或部分结构化数据。然而这些系统也要求有经验的人员来实现和 / 或更新。

发明内容

[0005] 一方面,提供了用于处理至少部分未结构化数据的方法。该方法包括在数据处理工具从至少一个数据源接收至少部分未结构化数据并处理该至少部分未结构化数据以生成包括标签化数据的至少部分结构化数据,其中标签化数据包括至少一个感兴趣的项目,以及其中处理该至少部分未结构化数据包括以下中的至少一个:利用关联存储器应用程序来处理该至少部分未结构化数据;以及利用正则表达式处理程序来处理该至少部分未结构化数据。该方法进一步包括传送至少部分结构化数据到主应用程序,并且至少部分基于标签化数据合并至少部分结构化数据到主应用程序中,其中合并至少部分结构化数据包括基于标签的存在、内容和 / 或类型来进行以下中的至少一个:包括数据和排除数据。

[0006] 另一方面,提供了一种具有体现在其上的计算机可执行指令的一个或更多计算机可读存储介质。当由至少一个处理器执行时,计算机可执行指令使该至少一个处理器在数据处理工具处从至少一个数据源接收至少部分未结构化数据,并处理至少部分未结构化数据以生成包括标签化数据的至少部分结构化数据,其中标签化数据包括至少一个感兴趣的

项目，并且其中处理至少部分未结构化数据，计算机可执行指令使处理器执行以下中的至少一个：利用关联存储器的应用程序来处理至少部分未结构化数据；以及利用正则表达式处理程序来处理至少部分未结构化数据。该指令进一步使至少一个处理器传送至少部分结构化数据到主应用程序中，并且至少部分基于标签化数据合并至少部分结构化数据到主应用程序中，其中合并至少部分结构化数据包括基于标签的存在进行以下中的至少一个：包括数据和排除数据。

[0007] 在又一个方面，提供了一种用于处理至少部分未结构化数据的系统。该系统包括处理装置、通信耦合到处理装置的用户接口以及通信耦合到处理装置的存储器和通信耦合到处理装置的通信接口中的至少一个。处理装置被编程为从存储器和通信接口中的至少一个接收至少部分未结构化数据；利用在其上执行的数据处理工具来通过以下中的至少一个处理至少部分未结构化数据以生成包括标签化数据的至少部分结构化数据，标签化数据包括至少一个感兴趣的项目；利用在其上执行的关联存储器应用程序来处理至少部分未结构化数据；和利用在其上执行的正则表达式处理程序来处理至少部分未结构化数据；以及基于标签化合并至少部分结构化数据到主应用程序中，其中合并至少部分结构化数据包括基于标签的存在来进行以下中的至少一个：包括数据和排除数据。

[0008] 已经讨论的特征、功能和优点可以在各种实施例中独立实现或可以在其他实施例中组合，其进一步细节可以参考下列描述和绘图看出。

附图说明

- [0009] 图 1 是用于处理文本的方法的流程图。
- [0010] 图 2A-2D 是图解说明在图 1 示出的方法的图示。
- [0011] 图 3 是用于对未结构化文本标签化以生成结构化文本的示例性方法的流程图。
- [0012] 图 4 是图解说明利用正则表达式处理程序来标签化未结构化文本的示例性方法的图示。
- [0013] 图 5 是图解说明利用关联存储器应用程序来标签化未结构化文本的示例性方法的图示。
- [0014] 图 6 是利用关联存储器应用程序来识别和标签化未结构化文本的示例性方法的图示。
- [0015] 图 7 是用于生成识别得分的示例性方法的流程图。
- [0016] 图 8A- 图 8C 是识别和选择错误识别的文本的示例性用户接口的实施例。
- [0017] 图 9 是示例性文本处理系统的框图。
- [0018] 图 10 是数据处理系统的图示。

具体实施方式

[0019] 本文所述的方法和系统涉及可能在数据源(例如，文本文件、数据库字段(database field))中发现的感兴趣的项目的识别。虽然本文所述的示例和实施例涉及文本处理，但是应当理解，实施例不应该解释为如此限制。描述文本处理的示例和实施例是为了清晰起见。本文使用的示例无意被视为限制性的，而仅仅用作说明性示例。更确切说，这里描述的实施例涉及包括任何类型信息和 / 或数据的处理，包括文本、字母数字数据

(alphanumeric data)、嵌入式对象、图像、元数据、视频、音频、多媒体和所有类型的数据和信息流中的一个或多个，而不限于任何特定形式或类型的这类数据和信息。

[0020] 因此该方法和系统涉及，例如利用数据处理工具来提供数据的标签化，这给数据提供了“结构”，以及发生在处理期间的数据的任何结构化的验证。虽然本文做了进一步描述，但是应当理解，实施例不仅涉及在文件内的未结构化数据的“结构化”，而且涉及包含部分结构化数据的文件的进一步结构化。为了进一步清晰起见，正如本文所使用的，未结构化数据是指通常由人员输入的数据，例如文本，其为自由形式并且基于该人的语法 / 语言而变化。例如，电子邮件和注记字段通常使用户能够输入自由形式的响应。进一步地，正如本文所使用的，若数据中的信息被标签化或以有组织化方式调用(call out)，那么结构化数据被称为结构化的和 / 或部分结构化的。前述将标签添加到文件内感兴趣的项目类似于将文件内的数据结构化。

[0021] 与现有的数据处理方法相比，这类实施例提供了改进的效率和性能。正如本文进一步所述的，可以利用关联存储器应用程序(associative memory application)和 / 或正则表达式处理程序中的一个或两者来识别数据内的感兴趣项目，通过标签化来结构化数据内的感兴趣项目，以及验证数据内的感兴趣项目。关联存储器包括多个数据和该多个数据之间的多个联合。关联存储器应用程序还指代利用关联存储器引擎将数据源合并在一起从而创建的关联存储器。关联存储器引擎是控制关联存储器创建、维护和存取的应用程序，类似于数据库软件如何控制多个数据库。关联存储器包括与其他实体和属性相关和 / 或关联的实体和属性。实体是在感兴趣的特定项目的关联存储器中的实例，属性是关联的实体的特性和 / 或描述。关联存储器记住属性、实体以及他们之间的联合。

[0022] 进一步地，在未结构化数据和 / 或部分结构化数据被处理成进一步结构化的数据后，任何由数据处理工具已经错误识别的数据能够被识别。错误识别(不正确标签化)的数据的这类实例用于改进和改善数据处理工具对进一步数据样本的识别、处理和验证的能力。正如本文所使用的，错误识别的数据是指被不正确标签化的数据和 / 或不正确地未标签化的数据(即，在处理期间应该已被标签化的未识别的数据，但不是例如之前没有被识别为需要被标签化而后来发现需要标签化的数据)。

[0023] 进一步地，在某些实施例中，用户接口使得用户能够识别和选择错误识别的数据，而不要求用户熟悉复杂的数据处理方法和系统和 / 或关联存储器系统和正则表达式处理程序。由于本文所描述的方法和系统中的至少某些不要求专门人员维护和 / 或更新数据处理工具，因此本文所描述的方法和系统有利于降低与已知数据分析系统相关的成本。

[0024] 图 1 是图解说明用于文本处理的方法 100 的流程图。方法 100 包括识别 102 待处理的文本，例如，如上所述的未结构化文本和 / 或部分结构化文本。在未结构化文本和 / 或部分结构化文本中识别 104 感兴趣的项目。例如，在一个实施例中，客户可以可视化地识别 104 对数据分析员感兴趣的项目。接着，标签化 106 感兴趣的项目以至少部分结构化该文本。可以利用手工或自动进程标签化 106 感兴趣的项目。

[0025] 验证 108 得到的包括标签的结构化文本(和 / 或部分结构化文本)该标签给文本提供结构(如下面进一步描述)。验证 108 可以包括将结构化文本显示在耦合到文本处理系统的一个或更多部件的用户接口上，并且观察给文本提供结构的各种标签。通过观察这类标签，能够快速验证是否正确地标签化未结构化文本和 / 或部分结构化文本。进一步地，在某

些实施例中,可以由用户选择已经被不正确地标签化或未标签化的文本,并将其用于更新正被使用的一个或更多文本处理工具。在验证 108 结构化文本后,释放 110 该结构化文本,以便进一步处理。释放的文本可以被传送到任何合适的数据挖掘应用程序和 / 或数据处理应用程序,其基于标签化来处理和 / 或合并该结构化文本。例如,该结构化文本可以被传送到主应用程序,如下面进一步描述的。

[0026] 图 2A-2D 是图解说明处理未结构化文本和 / 或部分结构化文本的示例性方法的图示,其通过以下步骤进行:识别感兴趣的项目并相应地标签化它们,由此给文本提供结构或额外结构。该方法可以利用各种文本处理方法和系统来实现。图 2A 包括在其原始形式的未结构化文本 202 的样本。未结构化文本 202 和 / 或部分结构化文本(未在图 2 示出)可以被存储在例如数据源中。为了清晰起见,在图 2B 中,以粗体字示出在未结构化文本 202 中的多个感兴趣的项目 204。在示例性实施例中,感兴趣的项目 204 包括在未结构化文本 202 中的作者、年份、大学名称、城市、零件号以及书名。

[0027] 在文本样本包括部分未结构化文本的实施例中,可能已经标签化了某些感兴趣的项目。例如,虽然之前已经标签化了作者和年份,但是仍然需要标签化大学名称。替换地,感兴趣的项目 204 可以包括在如本文所述的通过标签化而可能被识别和处理的未结构化文本和 / 或部分结构化文本内的任何类别和 / 或类型项目。例如,在本文所述的具体实施例中,感兴趣的项目 204 包括动物、日期和 / 或样板文件文本(*boilerplate text*)。

[0028] 应当理解,“样板文件”是基于应用领域来描述文本类别的通用术语,这些文本类别在风格、格式和 / 或内容上往往是类似的,特别是当文本由多个源创建时。在一个应用领域,样板文件包括署名块、法律免责声明、专有标记(*proprietary markings*)和 / 或电话会议信息。虽然在本文中常常称为文本,但是应当明白,样板文件还可以包括字母数字数据、嵌入式对象(图像、元数据等)中的一个或更多。在一个实施例中,客户可以视觉地识别在未结构化文本和 / 或部分结构化文本 202 中的感兴趣的项目 204。

[0029] 一旦感兴趣的项目 204 被识别,就标签化感兴趣的项目 204,这使文本 202 结构化和 / 或部分结构化。在示例性实施例中,客户例如利用用户接口视觉地识别感兴趣的项目 204。该用户接口可以耦合到文本处理系统的一个或更多组件。在一个实施例中,客户向数据分析员描述感兴趣的项目 204。为了确定附加的感兴趣的项目 204 是否应该被标签化以进一步结构化该文本,数据分析员可以和客户讨论在未结构化文本和 / 或部分结构化文本 202 中的模式和 / 或项目。接着,数据分析员利用同一用户接口或利用耦合到文本处理系统的一个或更多组件的单独的用户接口标签化附加的感兴趣的项目 204。

[0030] 替换地,可以通过自动进程来标签化感兴趣的项目 204 以结构化和 / 或部分结构化该文本。在一个实施例中,自动进程爬行(*crawl*)穿过适当名词、零件号和 / 或用于特定类型信息的任何其他值集合的已知列表。进一步地,可以利用关联存储器应用程序和 / 或正则表达式处理程序实现自动进程,如下所述。此外,自动进程还可以利用基于本体论的方法识别这类值集合。在这样的情况下,以及其他未在这里描述的情况下,可应用的标签可以被应用到在自动进程期间未被覆盖的得到的感兴趣的项目 204,以给这样的文本添加结构。

[0031] 在图 2C 中,插入标签 206 以继续进行识别的感兴趣的项目 204,从而结构化该文本。例如,包括日期标签可能是尤其重要的,而排除标签(*exclude-tag*)可能是不重要的。因此,这类标签 206 的存在指示至少部分结构化文本 207。例如,在结构化文本 207 中,利用

“author”标签 208 标签化“Henry David Thoreau”，利用“year”标签 210 标签化“1862”，以及利用“city”标签 212 标签化“Concord”。在图 2C 示出的示例中，标签 206 还包括“part_number”标签 214 和“book_title”标签 216。如上所解释的，由数据分析师或通过使用自动进程插入标签 206 到未结构化文本和 / 或部分结构化文本 202 内。这类标签的插入为文本生成了结构。

[0032] 如图 2D 所示，每个类型的标签 206 还可以包括唯一识别标签，或“i- 标签”。标签和“i- 标签”在形式上可以变化并使用不同的格式，包括使用 HTML / XML 类型标签或完全不同的格式。在图 2D 中，i- 标签以粗体字示出并具有形式“[ixx]”。在下述段落中分别引用图 2D 中的各 i- 标签中的若干。i- 标签使用户（例如客户和 / 或数据分析师）能够确定每个标签 206 应用到感兴趣的项目 204 的良好程度。更具体地说，i- 标签使用户能够快速确定给定的标签 206 是否成功地被应用并且如所期望的那样标签化感兴趣的项目 204，一个标签 206 的应用是否和另一个应用冲突，和 / 或一个标签 206 的应用是否类似于另一个标签 206 的应用和 / 或是另一个标签 206 的应用的复制品。为方便确定标签 206 的正确应用，得到的结构化文本 207 被显示在耦合到文本处理系统的一个或更多组件的用户接口上。

[0033] 例如，在图 2D 中，author 标签 208 包括 i- 标签 “[i01]”，book_title 标签 216 包括 i- 标签 “[i02]”。author 标签 208 和 book_title 标签 216 两者都正确地标签化感兴趣的项目 204。然而，如图 2D 所示，不正确的标签 220 错误识别在未结构化文本和 / 或部分结构化文本 202 中的“1234-1”。即，包括 i- 标签 “[i05]” 的 part_number 标签 214 不正确地识别“1234-1”为短语“The distance from his porch to the water's edge was 1234-1255feet”中的零件号。即，如在该短语中所使用的“1234-1”不是感兴趣的项目 204，并且不应该被标签化为 part_number 标签 214。另外，i- 标签 “[i14]”也紧挨着“1234-1”出现，指示另一个标签 206 被应用到该特定文本。通过在用户接口上观察不正确 i- 标签，数据分析师能够迅速地确定包括 i- 标签 “[i05]”和 “[i14]”的标签 206 中的至少一个操作不正确和 / 或不成功，并采取恰当的步骤纠正这个错误。

[0034] 一旦包括标签 206 的结构化文本 207（其可以是仅部分结构化）被验证（即，确定所有标签 206 操作正确），就释放结构化文本 207，以便进一步处理。在一个实施例中，用户验证应用程序数据源中的得到的结构化文本以确定文本处理工具是否正确地处理来自主数据源的未结构化文本和 / 或部分结构化文本。若用户验证文本被正确处理，则用户释放该文本（结构化文本和 / 或部分结构化文本）到应用程序数据源中，使得主应用（如本文进一步描述）能够合并该结构化文本。若该用户确定该文本被不正确地处理，则用户更新处理工具数据源和 / 或处理工具以纠正任何文本处理错误和 / 或过失。在实施例中，验证和更新是自动的或部分自动的。

[0035] 图 3 是用于标签化未结构化文本以生成结构化（或部分结构化）文本的示例性方法的流程图 300。应当注意，根据接收的文本内容和感兴趣的项目，同一方法用于部分结构化文本的进一步标签化以进一步结构化该文本以及可能得到仅部分结构化文本的未结构化文本的标签化。为进一步清晰起见，如本文所使用的，未结构化文本是指通常由人员输入的文本，其为自由形式并且基于该人员的语法 / 语言而变化。例如，电子邮件和注记字段通常使用户能够输入自由形式的响应。进一步地，正如本文所使用的，若文本中的信息被标签

化或以组织方式调用，则文本被称为结构化的和 / 或部分结构化的。在示例性实施例中，结构化文本是指包括识别文本中信息的一个或更多标签的文本。为了处理，未结构化文本和 / 或部分结构化文本被供应给文本处理工具 304。

[0036] 在本文所述的示例性实施例中，文本处理工具 304 包括正则表达式处理程序 309 和关联存储器引擎 308 内的关联存储器应用程序 306 中的一个或两者，用于通过标签的插入结构化未结构化文本和 / 或部分结构化文本 302，如本文详细描述的。关联存储器应用程序 306 包括关联存储器。如本文所使用的，关联存储器是指利用一个或更多数据源生成的信息储藏。该信息储藏包括与其他实体和属性相关和 / 或关联的实体和属性。

[0037] 实体是在感兴趣的特定项目的关联存储器中的实例，属性是关联实体的特性和 / 或描述。关联存储器应用程序 306 使用户能够通过属性与实体和 / 或实体类型的联合两者做相似度分析和执行类比查询。因此，关联存储器应用程序 306 使得能够发现之前未识别的属性和实体之间的关联。关联存储器引擎 308 使关联存储器应用程序 306 能够搜索关于存储在关联存储器中的实体和实体关系的信息。

[0038] 在示例性实施例中，文本处理工具 304 还包括正则表达式处理程序 309，用于处理未结构化文本和 / 或部分结构化文本 302，如下面详细描述。替换地，文本处理工具 304 可以仅包括关联存储器应用程序 306 和正则表达式处理程序 309 中的一个。进一步地，在某些实施例中，关联存储器应用程序 306 或正则表达式处理程序 309 构成完整的文本处理工具 304。文本处理工具 304 利用关联存储器应用程序 306 和 / 或正则表达式处理程序 309 来处理未结构化和 / 或部分结构化文本 302 并且输出结构化文本 310，如本文所述。

[0039] 图 4 是图解说明利用正则表达式处理程序 (REPP) 400 (例如正则表达式处理程序 309 (如图 3 所示)) 来对未结构化文本和或部分结构化文本进行标签化(结构化)的图示。REPP400 可以和本文进一步描述的系统一起使用。根据应用，REPP400 可以是文本处理工具的一个组件或可以构成完整的文本处理工具。待处理的未结构化文本和 / 或部分结构化文本被存储在源表格 402 中，该源表格可以是主数据源的一部分。未结构化文本和 / 或部分结构化文本在源表格 402 中被组织为文本的列。

[0040] 在示例性实施例中，为了给未结构化文本和 / 或部分结构化文本添加标签，用户利用用户接口选择所期望的文本段，例如，用户接口耦合到文本处理系统的一个或更多组件。某些实施例也允许用户简单地手工编辑源以添加标签。选择的文本段从源表格 402 传送到 REPP400，以便处理添加标签到文本，并因此添加结构到文本。替换地，未结构化文本和 / 或部分结构化文本的段和 / 或列可以从源表格 402 自动地传送到 REPP400 (即用户没有选择文本)。REPP400 可以由嵌在计算机可读介质中的可执行指令编程。

[0041] 在 REPP400，一个或更多源正则表达式模式 (SREP) 404 被应用到选择的文本段和 / 或列。在示例性实施例中，SREP404 被存储在处理工具数据源中。在 SREP404 中的正则表达式是在大多数编程语言 (例如，Java，PERL) 中可用的标准字母数字字符和非字母数字字符，其用于匹配文本中的一系列字符。

[0042] 在示例性实施例中，给定的 SREP404 包含包括四种类型实体的行：捕捉所期望系列字符的正则表达式模式；替换模式；REPP400 用来执行特定动作 (例如，递归应用具体模式) 的特殊字符；记载给定的 SREP404 的目的任务的注记字段。REPP400 在 SREP404 中读取，按从顶部到底部的顺序应用每个 SREP404 行，并输出输出表格 406 和输出 HTML 页 408 中

的至少一个。在某些实施例中，如本文进一步描述，输出表格 406 是应用程序数据源的一部分。在示例性实施例中，输出表格 406 和 HTML 页 408 两者具有数据列，其包含如输出 HTML 页 408 的“MODIFIED”列中所示的标签化文本，该标签化文本在本文称为结构化文本。

[0043] 如上所述，SREP404 匹配并标签化选择的文本中的预定模式以提供这种文本的结构化。例如，在图 4 中，Animal SREP 匹配并标签化文本段中的动物名称，Date SREP 匹配并标签化文本段中的四个字符作为年份。Animal SREP 和 Date SREP 是可以应用于一个实施例的 SREP 的具体示例。应当明白，Animal SREP 和 Date SREP 不是必然地关联在 404 中示出的类属 SREP 示例(例如，模式 1，模式 2)。

[0044] 接着，标签化的文本段被传送到输出表格 406 和 / 或输出 HTML 页 408。在示例性实施例中，用户利用用户接口选择是否传送标签化的文本段到输出表格 406 和 / 或输出 HTML 页 408。进一步地，在一个实施例中，结构化的文本段被传送到应用程序，以便进一步处理。在下述的一个示例中，应用程序至少部分基于置入文本中的标签来合并结构化文本。例如，应用程序可以包括或排除某些标签化单词和 / 或短语。

[0045] 输出 HTML 页 408 显示应用 SREP404 到未结构化文本和 / 或部分结构化文本段的结果。例如，在图 4 中，输出 HTML 页 408 示出，“fox”在文本 410 的第一段中被标签为 animal，“1942”在文本 412 的第二段中被标签为 year。在一个实施例中，输出 HTML 页 408 被显示在显示装置的用户接口上。通过观察输出 HTML 页 408，该用户能够确定结构化文本的任何段是否被正确地标签化。在某些实施例中，利用该用户接口，错误识别的文本能够用于更新 SREP404，例如，SREP404 将被更新以纠正生成不正确标签的一个或更多现有模式。例如，当用户识别和 / 或选择错误识别的文本时，该错误识别的文本能够用于修改现有的 SREP404 和 / 或创建要被应用于新的未结构化文本和 / 或部分结构化文本的新的 SREP404。

[0046] 在示例性实施例中，每个 SREP404 包括唯一识别标签，或“i- 标签”。该“i- 标签”使用户能够确定在 REPP400 操作期间每个 SREP404 工作得如何。更具体地说，该 i- 标签使用户能够确定给定的 SREP404 是否成功匹配并标签化所希望的文本段，确定一个 SREP404 是否和另一个 SREP404 的运行冲突，和 / 或确定一个 SREP404 执行的操作是否类似于另一个 SREP404 操作和 / 或是另一个 SREP404 操作的复制。

[0047] 例如，在图 4 中，Animal SREP 包括 i- 标签 “[i21]”，Date SREP 包括 i- 标签 “[i22]”。因此，在输出 HTML 页 408 中，第一文本段 410 包括 “[i21]”，其指示利用 Animal SREP 标签化第一文本段 410，第二文本段 412 包括 “[i22]”，其指示利用 Date SREP 标签化第二文本段 412。虽然在示出的实施例中，两个 SREP404 用于应用标签到未结构化文本和 / 或部分结构化文本，但是可以应用使 REPP400 能够起如本文所述的作用的任何数量的 SREP。

[0048] 图 5 是图解说明关联存储器应用程序 500 (例如关联存储器应用程序 306)如何识别和标签化未结构化文本以提供结构化文本结果的图示。在示例性实施例中，未结构化文本和 / 或部分结构化文本被存储在数据源中的一列或更多列中。该未结构化文本可以被分开成多个列，使得该未结构化文本被拆分成开的列中的多个段。文本处理工具(例如文本处理工具 304)利用关联存储器应用程序 500 来识别和标签化在未结构化和 / 或部分结构化文本中的感兴趣的项目，如本文所述。

[0049] 在图 5 示出的示例中，关联存储器应用程序 500 识别和标签化在未结构化 / 部分

结构化数据中的样板文件文本,由此添加结构到未结构化 / 部分结构化数据中。虽然图 5 中所示的示例图解说明了识别和标签化样板文件,但是该示例仅仅是说明性的,因为关联存储器应用程序 500 可以用于识别和标签化在未结构化和 / 或部分结构化的文本和 / 或数据中的任何感兴趣相关项目。

[0050] 在描述该示例中,应当理解,“样板文件数据”是描述文本和 / 或其他数据(例如,字母数字数据、嵌入式对象、图像、元数据等)的类别的通用术语,这些文本类别在风格、格式和 / 或内容方面往往类似,特别是当文本 / 数据由多个源创建时。对于本示例目的,样板文件数据包括签名块、法律免责声明、专有标记和 / 或电话会议信息,但是该术语不应该视为如此限制。由于样板文件通常与特定应用程序无关,而且若其由主应用程序接收,则可能负面影响使用这类应用程序的结果,因此期望从这类应用程序中排除(即,不合并)样板文件。

[0051] 在这个特定示例中,若文本段类似于现有的样板文件,则其被标签化为样板文件。在一个实施例中并且不以限制本文所述的方法和系统的范围的方式提供这个示例,以展示文本处理工具如何利用关联存储器应用程序来识别和标签化文本。更具体地说,若关联存储器被如此配置,则可以利用关联存储器应用程序来识别与样板文件的识别和标签化不相关的感兴趣文本项目。

[0052] 为了识别和标签化文本,文本处理工具(例如文本处理工具 304)查询关联存储器应用程序 500 (例如关联存储器应用程序 306 (如图 3 所示))。在示例性实施例中,关联存储器应用程序 500 由数据库生成。例如,图 5 示出包括标记列 504、文本列 506 和识别列 508 的数据库 502,标记列 504 包括用于不同文本串的唯一整数,文本列 506 包括不同的文本串,识别列 508 识别文本串是否是感兴趣的项目。

[0053] 例如,在数据库 502 中,文本“BOILERPLATE IS HERE.”被识别为样板文件,而文本“TESTING ON NEW EQUIPMENT.”被识别为不是样板文件。虽然在示例性实施例中,数据库 502 具有三个列,但是数据库 502 可以具有使测试处理工具和关联存储器应用程序能够如本文所述起作用的任何数量的列。在某些实施例中,数据库 502 被视为与正则表达式模式(例如 SREP (如图 4 所示))并列(parallel)。

[0054] 在示例性实施例中,为了生成关联存储器应用程序 500,标记列 504 和识别列 508 被直接合并到关联存储器应用程序 500。在示例性实施例中,在文本列 506 中的文本段被直接合并到关联存储器应用程序 500,使得文本列 506 和关联文本段形成关联存储器应用程序 500 的一部分。替换地,在文本列 506 中的文本段可以利用类属语法分析器和 / 或提取器合并到关联存储器应用程序 500,使得在文本列 506 中的文本可以进一步分成和 / 或语法分析成关键项目,例如在关联存储器应用程序 500 中形成一个或更多文本段的关键字和 / 或关键短语。

[0055] 例如,文本列 506 可以分成和 / 或语法分析成名词、动词和 / 或形容词。替换地,可以利用使文本处理工具能够如本文所述起作用的任何进程实现关联存储器应用程序 500。当利用关联存储器应用程序 500 时,未结构化和 / 或部分结构化文本被分成和 / 或语法分析成段,并与关联存储器应用程序 500 的文本列 506 中的文本段分成的组件和 / 或关键词比较,如下列详细描述。

[0056] 在示例性实施例中,文本处理工具从数据库源接收未结构化和 / 或部分结构化文本,例如样本文本 510。在示例性实施例中,通过利用类属语法分析器和 / 或提取器将未结

构化和 / 或部分结构化文本语法分析成分离的文本段,以此生成样本文本 510。通过利用样本文本 510 查询关联存储器应用程序 500,文本处理工具识别和标签化样本文本 510 的段为感兴趣的项目,从而生成结果文本 512。

[0057] 例如,在结果文本 512 中,文本“BOILERPLATE IS HERE.”被标签化为样板文件,文本“NEW EQUIPMENT TESTING.”没有被标签化为样板文件。在替换实施例中,文本“NEW EQUIPMENT TESTING.”可以被标签化为非样板文件。因为文本处理工具利用在关联存储器应用程序中的文本列 506 的内容识别和标签化文本,所以未结构化文本和 / 或部分结构化文本的段不需要精确匹配在关联存储器应用程序中的文本段。例如,“THIS IS BOILERPLATE.”被识别和标签化为样板文件,即使关联存储器应用程序包括文本短语“THIS IS A BOILERPLATE TEST.”。

[0058] 图 6 是利用关联存储器应用程序(例如关联存储器应用程序 306)来识别和标签化文本的示例性方法 600 的流程图。文本处理工具(例如文本处理工具 304)接收 602 待处理的未结构化和 / 或部分结构化文本。为了识别目的,未结构化和 / 或部分结构化文本被分成和 / 或语法分析成分离的文本段,例如段落、句子和 / 或单词。

[0059] 对于未结构化和 / 或部分结构化文本的每一个段,文本处理工具查询 604 关联存储器应用程序,并且基于未结构化和 / 或部分结构化文本的内容分割和 / 或段的关键词与关联存储器应用程序中文本列 506 的内容分割和 / 或段的关键词的比较,关联存储器应用程序生成 606 识别分数。文本处理工具确定 608 识别分数是否高于预定的阈值。若识别分数高于预定的阈值,则未结构化和 / 或部分结构化文本的段被标签化 610 为感兴趣的项目。若识别分数低于预定的阈值,则未结构化和 / 或部分结构化文本不被标签化 612。

[0060] 接着,根据识别分数可以被标签化的文本段被供应给主应用程序,以便基于标签化合并。该标签化文本是结构化文本。在一个实施例中,结构化文本被发送到输出表格,接着其被主应用程序使用。在示例性实施例中,文本处理工具利用关联存储器应用程序相应地识别和标签化未结构化和 / 或部分结构化文本的剩余段。

[0061] 图 7 是为关联存储器应用程序所应用的未结构化和 / 或部分结构化文本段生成识别分数的示例性方法 700 的流程图。对于关联存储器应用程序中文本的每个段(即,来自文本列 506 的文本的每个串),文本处理工具确定 702 未结构化和 / 或部分结构化文本的段与关联存储器应用程序中文本(文本列 506)的段比较的相似度分数, s_i 。

[0062] 例如,相似度分数 s_i 可以定义为未结构化和 / 或部分结构化文本的段和关联存储器应用程序中文本的段之间的匹配项目(例如,单词)的数目除以未结构化和 / 或部分结构化文本的段中的项目的总数目。文本处理工具确定 704 相似度分数 s_i 是否高于预定的相似度阈值。若相似度分数低于预定的相似度阈值,则文本处理工具给关联存储器应用程序中文本段分配值为“0”,并开始确定 702 未结构化和 / 或部分结构化文本的同一段和关联存储器应用程序中下一个段比较的相似度分数 s_i 。

[0063] 若相似度分数 s_i 高于预定的相似度阈值,则文本处理工具,例如利用来自数据库 502 的识别列 508 的信息确定 706 关联存储器应用程序中的文本段是否是感兴趣的项目。在示例性实施例中,若关联存储器应用程序中的文本段是感兴趣的项目,则关联存储器应用程序中的文本段被分配等于相似度分数的值。

[0064] 若关联存储器应用程序中的文本段不是感兴趣的项目,则关联存储器应用程序中

的文本段被给予值“0”。在相对于未结构化和 / 或部分结构化文本的特定段为关联存储器应用程序中的每一个文本段(即,为来自列 506 的每个文本串)确定值后,通过合计 708 分配给关联存储器应用程序中的每一个文本段的值,以此计算未结构化和 / 或部分结构化文本段的识别分数。

[0065] 虽然图 7 示出生成识别分数的示例性方法 700,但是可以利用任何使文本处理工具能够如本文所述起作用的方法。例如,在某些实施例中,当相似度分数 s_i 低于预定的阈值和 / 或当关联存储器应用程序中的文本段不是感兴趣的项目时,则关联存储器应用程序中的文本段被分配非零值。进一步地,在其他实施例中,可以利用相似度分数和值,利用其他更复杂度量方式计算识别分数。

[0066] 图 8A-8C 示出使用户能够添加错误识别的文本到如上所述的关联存储器应用程序中的示例性用户接口的截屏。在示例性实施例中,用户接口显示在被文本处理工具处理后的结构化文本。例如,对于上面讨论的关联存储器应用程序示例,该用户接口显示与电子邮件 802 关联的文本。该文本包括第一样板文件部分 804 和第二样板文件部分 806。如图 8A 所示,文本处理工具识别和标签化第二样板文件部分 806 为样板文件文本,但没有识别和标签化第一样板文件部分 804 为样板文件文本。因此,第一样板文件部分 804 是错误识别的文本。

[0067] 利用用户接口,该用户能够视觉地识别错误识别的文本。进一步地,该用户能够拷贝该错误识别的文本到窗口 808,如图 8B 所示。通过选择语法分析按钮(parse)810,该错误识别的文本被加载到处理工具数据源。一旦错误识别的文本被供应给文本处理工具中的关联存储器应用程序,确认窗口 812 就被显示在用户接口上,提醒该用户关联存储器应用程序已经被更新为包括该错误识别的文本,如图 8C 所示。

[0068] 因此,当文本处理工具处理包含错误识别的文本的未结构化文本和 / 或部分结构化文本,并且通过例如用户交互通知文本处理工具时,该文本处理工具将被更新以正确地处理接下来的错误识别的文本。因此,该文本处理工具被重复地更新,从而改善文本处理工具处理来自数据源的新的未结构化文本和 / 或部分结构化文本的能力。进一步地,更新文本处理工具不需要对文本处理工具进行复杂的编程和 / 或关联存储器系统和方法的专家知识。更确切说,用户能够使用用户接口相对较快和容易地更新该文本处理工具。

[0069] 图 9 是可以合并上述实施例的某些或全部的示例性文本处理系统 900 框图。系统 900 包括主数据源 902,其接收和 / 或包括将要最终合并到例如主应用程序 904 的未结构化文本和 / 或部分结构化文本(即,未处理文本)。正如本文所使用的,合并文本到主应用程序 904 是指输入正确标签化的(结构化的)文本到主应用程序 904。主数据源 902 可以包括使系统 900 能够如本文所述起作用的任何数量的单独数据源。在示例性实施例中,主应用程序 904 合并来自应用程序数据源 905 的文本。

[0070] 主数据源 902 耦合到文本处理工具 906,例如文本处理工具 304(如图 3 所示)。在示例性实施例中,文本处理工具 906 从主数据源 902 接收未结构化文本和 / 或部分结构化文本,并通过如上所述添加适当标签来将该未结构化文本和 / 或部分结构化文本处理为至少部分结构化文本。该结构化文本包括已被标签化的文本的至少一个段。

[0071] 正如本文所使用的,文本段是指文本的一个或更多单词,其中单词可以是任何一组连续的字符。文本处理工具 906 包括关联存储器应用程序(例如关联存储器应用程序 306

(如图 3 所示)) 和 / 或正则表达式处理程序(例如正则表达式处理程序 309 (如图 3 所示))中的一个或两者,以便处理未结构化文本和 / 或部分结构化文本,如上详细描述。

[0072] 文本处理工具 906 通过应用程序数据源 905 耦合到主应用程序 904,使得来自自主数据源 902 的未结构化文本和 / 或部分结构化文本由文本处理工具 906 处理,并作为结构化文本输出到应用数据源 905,以便用在主应用程序 904 中。替换地,从文本处理工具 906 输出的结构化文本可以在被传送到应用数据源 905 之前经受额外的处理。应用数据源 905 可以包括例如输出表格和 / 或输出超文本标记语言(HTML)页,其用于验证文本的结构化,然而也可以考虑其他的格式。在示例性实施例中,主应用程序 904 合并来自应用程序数据源 905 的结构化文本。

[0073] 为了处理来自自主数据源 902 的未结构化文本和 / 或部分结构化文本,文本处理工具 906 查询关联存储器应用程序和 / 或应用至少一个源正则表达式模式到未结构化文本和 / 或部分结构化文本。例如,在一个实施例中,文本处理工具 906 通过以下步骤来处理未结构化文本和 / 或部分结构化文本:用未结构化文本和 / 或部分结构化文本的段查询关联存储器应用程序,计算相似度分数,以及基于相似度分数确定是否标签化未结构化文本和 / 或部分结构化文本。

[0074] 文本处理工具 906 处理未结构化文本和 / 或部分结构化文本而生成的结构化文本从文本处理工具 906 传送到应用数据源 905,在这里其能够被合并到主应用程序 904。主应用程序 904 基于标签化的文本段合并结构化文本。例如,在某些实施例中,标签化文本被合并到主应用程序 904,非标签化文本不被合并到主应用程序 904。为了清晰起见,在本文描述的示例中,用样板文件(boilerplate)标签标签化的文本被忽略,由主应用程序合并其他的一切。

[0075] 在示例性实施例中,主应用程序 904 是数据分析应用程序,而且可以包括例如商业智能应用程序、关联存储器应用程序和 / 或搜索引擎。替换地,主应用程序 904 可以是使系统 900 能够如本文所述起作用的任何应用程序。在示例性实施例中,文本处理工具 906 在主应用程序 904 合并结构化文本之前处理未结构化文本和 / 或部分结构化文本。

[0076] 基于文本处理工具 906 对未结构化文本和 / 或部分结构化文本的标签化,主应用程序 904 合并结构化文本。处理文本以便由主应用程序 904 合并减少了合并到主应用程序 904 的文本总量,改善了合并文本到主应用程序 904 的速度,减少了主应用程序 904 所使用的存储器数量,和 / 或改善了从主应用程序 904 可以获取文本的速度,以及改善了结果。

[0077] 在示例性实施例中,主应用程序 904 耦合到用户接口 908。用户接口 908 可以包括显示装置,例如阴极射线管(CRT)、液晶显示器(LCD)、有机 LED(OLED)显示器和 / 或“电子墨水”显示器。进一步地,用户接口 908 可以包括使用户能够和用户接口 908 交互的输入装置,例如键盘、定位装置、鼠标、尖笔、接触感应面板(例如,接触板或触摸屏)、陀螺仪、加速计、位置检测器和 / 或音频用户输入接口。

[0078] 利用用户接口 908,用户能够查看结构化文本。用户接口 908 使用户能够从结构化文本中选择并且提取错误识别的文本。即,用户能够选择并且提取文本处理工具 906 不正确处理的或根本没有处理的文本段。在示例性实施例中,与错误识别的文本相关的数据和 / 或错误识别的文本本身被接着转发到和 / 或存储在耦合到用户接口 908 的处理工具数据源 910 上。在某些实施例中,处理工具数据源 910 还包括要被供给到文本处理工具 906 且

不是错误识别的文本的初始数据。

[0079] 文本处理工具 906 利用初始数据并且从用户接口 908 处接收的用户输入进行更新,以根据本文所述的方法和系统处理未结构化和 / 或部分结构化文本。在某些实施例中,一个或更多额外的用户接口耦合到文本处理系统 900 的一个或更多组件以有助于本文所述的方法和系统能够实现。如图 9 所示,文本处理,将处理过的文本应用到主应用程序 904,通过用户接口 908 查看额外的文本处理需求,这会是迭代和重复的过程,其能够提供改善的结果,因为改进了文本的标签化。

[0080] 在文本处理工具 906 包括关联存储器应用程序的实施例中,处理工具数据源 910 例如基于用户输入更新关联存储器应用程序,如上所述。进一步地,在文本处理工具 906 包括正则表达式处理程序的实施例中,可以更新源正则表达式模式,以正确处理包括之前错误识别的文本的未结构化文本和 / 或部分结构化文本。

[0081] 类似于主数据源 902,处理工具数据源 910 可以包括使系统 900 能够如本文所述起作用的任何数量的单独数据源。在一个实施例中,处理工具数据源 910 基于通过用户接口 908 接收的输入,定期地供应任何错误识别的文本到文本处理工具 906 的关联存储器应用程序中。替换地,处理工具数据源 910 可以连续地或无论何时用户识别到新的错误识别的文本段时供应错误识别的文本到文本处理工具 906。

[0082] 用来自处理工具数据源 910 的错误识别的文本更新文本处理工具 906,以改善对来自主数据源 902 的未结构化文本和 / 或部分结构化文本的进一步处理。因此,通过把由文本处理工具 906 初始错误识别的文本供应回到文本处理工具 906,文本处理工具 906 正确处理未结构化文本和 / 或部分结构化文本的能力随着时间而改善,因为当处理新的未结构化文本和 / 或部分结构化文本时,文本处理工具 906 利用错误识别的文本。虽然在示例性实施例中仅示出一个文本处理工具 906,但是系统 900 可以包括使系统 900 能够如本文所述执行的任何数量的文本处理工具 906。例如,系统 900 可以包括用于处理来自不同主数据源 902 的不同类型的未结构化文本和 / 或部分结构化文本的不同文本处理工具 906,和 / 或利用不同文本处理方法的文本处理工具 906。

[0083] 如上所述,在示例性实施例中,文本处理工具 906 供应结构化文本到应用程序数据源 905,其向主应用程序 904 提供数据。进一步地,该结构化文本可以被包括在应用程序数据源 905 中的输出表格和 / 或输出 HTML 页中。在本文解释的示例中,主应用程序 904 例如基于文本是否利用正则表达式处理程序和关联存储器应用程序中的一个或两者如所述的被标签化来处理文本。例如,在一个具体示例中,主应用程序 904 不合并已经被标签化为样板文件的文本。替换地,主应用程序 904 可以以使系统 900 能够如本文所述起作用的任何方式合并来自应用程序数据源 905 的结构化文本。

[0084] 系统 900 通过设置如下架构而运行,其使数据分析系统 904 的用户(没有任何专业技能)能够通过为数据处理工具 906 建立数据源 910 来改善系统 904 的性能。在一个实施例中,应用语法分析能力 906 包括应用关联存储器数据标记进程,该进程包括:始于数据比较;语法分析该数据以确定关联存储器实体和属性;基于从数据导出的实体和属性查询关联存储器应用程序,寻找相似结果;利用相似结果组对结果排名和计算分数;以及基于该分数,暗示关于实体和属性的额外信息。

[0085] 额外信息转换类属实体和属性成更多特定域实体和属性。利用特定域实体和属

性,该数据被标记,以便稍后用于改善数据分析系统 904(例如,关联存储器系统、商业智能应用程序、搜索引擎等)。进一步地,可以检查从这些分析系统的输出,以识别和提取能够通过用户接口 908 提供到“数据处理”关联存储器应用程序 906 的数据源 910 的错误识别的数据。

[0086] 图 10 是可以用于实现本文所述的一个或更多实施例的示例性数据处理系统 1000 的图示。例如,可以利用数据处理系统 1000 实现文本处理工具 304(数据处理工具 906)、关联存储器应用程序 306、正则表达式处理程序 309 和 / 或文本处理系统 900 的一个或更多组件。在示例性实施例中,数据处理系统 1000 包括通信构造 1002,其提供处理器单元 1004、存储器 1006、永久存储介质 1008、通信单元 1010、输入 / 输出(I/O) 单元 1012 和显示器之间的通信。

[0087] 处理器单元 1004 用于执行可以被加载到存储器 1006 的软件指令。处理器单元 1004 根据特定实现可以是一个或更多处理器的集合或可以是多处理器核。进一步地,可以利用一个或更多异质处理器系统来实现处理器单元 1004,在其中单个芯片上具有主处理器和二级处理器。

[0088] 如另一个说明性示例,处理器单元 1004 可以是包含多个相同类型处理器的对称多处理器系统。进一步地,可以利用任何合适的可编程电路实现处理器单元 1004,其包括一个或更多系统和微控制器、微处理器、精简指令集电路(RISC)、专用集成电路(ASIC)、可编程逻辑电路、现场可编程门阵列(FPGA),以及任何其他能够执行如本文所述功能的电路。

[0089] 存储器 1006 和永久存储介质 1008 是存储装置的示例。存储装置是能够临时和 / 或永久存储信息的任何硬件。在这些示例中,存储器 1006 可以是例如但不限于,随机存取存储器或任何其他合适的易失性或非易失性存储装置。永久存储介质 1008 可以根据特定实现采用各种形式。

[0090] 例如但不限于,永久存储介质 1008 可以包含一个或更多组件或装置。例如,永久存储介质 1008 可以是硬盘驱动器、快闪存储器、可重写光盘、可重写磁带或上述的某些组合。由永久存储介质 1008 使用的介质还可以是可移除的。例如但不限于,可移除硬盘驱动器可以用于永久存储介质 1008。

[0091] 永久存储介质 1008 可以是在其上具有计算机可执行指令的计算机可读存储介质,其中当由至少一个处理器执行时,该计算机可执行指令使处理器接收并且处理部分未结构化数据,以包括标签,使得其可以由关联存储器应用程序进一步处理。这可以进一步使部分结构化数据能够传送到主应用程序,在这里部分未结构化数据能够基于标签化而经受进一步处理,包括和排除数据。额外的技术效果是,部分结构化数据可以语法分析成一个或更多段,并且能够用部分未结构化数据中的至少一个段查询。其还可以生成与部分未结构化数据中的至少一个段和关联存储器应用程序中至少一个数据段相关的分数。这些结果可以通过在用户接口上的显示器上观察,使得用户能够识别标签化是正确的,并且释放部分未结构化数据,用于合并到主应用程序中。用户可以观察不正确标签化的部分未结构化数据,从而提供更新错误识别的数据段的机会。当正则表达式处理程序处理部分未结构化数据时,其可以匹配至少部分未结构化数据的至少一个段到至少一个源正则表达式模式,并且标签化一个匹配的段。计算机可执行指令可以使处理器输出至少部分结构化数据到输出表格和输出超文本标记语言(HTML)页中的一个。

[0092] 在这些示例中,通信单元 1010 提供和其他数据处理系统或装置的通信。在这些示例中,通信单元 1010 是网络接口卡。通信单元 1010 可以通过使用物理通信链接或无线通信链接中的任意一个或两者提供通信。

[0093] 输入 / 输出单元 1012 允许用可以连接到数据处理系统 1000 的其他装置输入和输出数据。例如但不限于,输入 / 输出单元 1012 可以通过键盘和鼠标为用户输入提供连接。进一步地,输入 / 输出单元 1012 可以发送输出给打印机。显示器 1014 提供给用户显示信息的机构。

[0094] 用于操作系统和应用程序或程序的指令位于永久存储介质 1008 上。这些指令可以被加载到存储器 1006,以便由处理器单元 1004 执行。不同实施例的进程可以由处理器单元 1004 利用位于存储器(例如存储器 1006)中的计算机实现指令执行。这些指令称为程序代码、计算机可用程序代码或计算机可读程序代码,其可以由处理器单元 1004 中的处理器读取和执行。在不同实施例中,程序代码可以体现在不同物理或有形的计算机可读介质中,例如存储器 1006 或永久存储介质 1008。

[0095] 程序代码 1016 以功能形式位于选择性可移除的计算机可读介质 1018 上,并且可以被加载到或转移到数据处理系统 1000,以便由处理器单元 1004 执行。这些示例中,程序代码 1016 和计算机可读介质 1018 形成计算机程序产品 1020。在一个示例中,计算机可读介质 1018 可以是有形的形式,例如光盘或磁盘,其被插入或置入驱动器或是永久存储介质 1008 的一部分的其他装置中,以便转移到存储装置(例如是永久存储介质 1008 的一部分的硬盘驱动器)上。在有形形式中,计算机可读介质 1018 也可以采用永久存储介质的形式,例如硬盘驱动器,拇指驱动器,或连接到数据处理系统 1000 的快闪存储器。有形形式的计算机可读介质 1018 也可以称为计算机可刻录存储介质。在某些示例中,计算机可读介质 1018 是不可以移除的。

[0096] 替换地,程序代码 1016 可以从计算机可读介质 1018 通过至通信单元 1010 的通信链接和 / 或通过至输入 / 输出单元 1012 的连接转移到数据处理系统 1000。在说明性示例中,该通信链接和 / 或连接可以是物理的或无线的。计算机可读介质还可以采用非有形介质的形式,例如包含程序代码的通信链接或无线传输。

[0097] 在某些说明性实施例中,程序代码 1016 可以通过网络从另一个装置或数据处理系统下载到永久存储介质 1008,以便在数据处理系统 1000 内使用。例如,存储在服务器数据处理系统中的计算机可读存储介质的程序代码可以通过网络从服务器下载到数据处理系统 1000。提供程序代码 1016 的数据处理系统可以是服务器计算机、客户端计算机或某些能够存储和传输程序代码 1016 的其他装置。

[0098] 为数据处理系统 1000 示出的不同组件并不意味着对不同实施例实现的方式提供架构限制。不同的说明性实施例可以实现在包括除了那些针对数据处理系统 1000 示出的组件之外的组件的数据处理系统中,或在包括替换那些针对数据处理系统 1000 示出的组件的组件的数据处理系统 1000 中。在图 10 示出的其他组件可以根据示出的说明性示例而改变。

[0099] 作为一个示例,数据处理系统 1000 中的存储装置是可以存储数据的任何硬件装置。存储器 1006、永久存储介质 1008 和计算机可读介质 1018 是有形形式存储装置的示例。

[0100] 在另一个示例中,总线系统可以用于实现通信构造 1002 并且可以由一个或更多

总线组成,例如系统总线或输入 / 输出总线。当然,总线系统可以利用任何合适类型的提供附着于总线系统的不同组件或装置之间的数据转移的架构来实现。另外地,通信单元可以包括一个或更多用于发送和接收数据的装置,例如调制解调器或网络适配器。进一步地,存储器可以是例如但不限于,存储器 1006 或在接口和通信构造 1002 中可能存在的存储器控制器集线器中找到的高速缓存。

[0101] 本文描述的实施例使用数据处理工具来提供对未结构化和 / 或部分结构化数据的改进处理,从而提供超过现有数据处理方法的改善的效率和性能。可以利用关联存储器应用程序和 / 或正则表达式处理程序来处理数据。进一步地,在未结构化和 / 或部分结构化数据被处理后,用户能够识别数据处理工具错误识别的和 / 或未识别的(例如,被忽略的文本或不恰当标签化的文本)数据。

[0102] 这个错误识别的数据用于改善和改进数据处理工具处理和识别新的未结构化和 / 或部分结构化数据的能力。进一步地,在某些实施例中,用户接口使用户能够识别和选择错误识别的数据,而不要求用户对复杂的数据处理方法和系统和 / 或关联存储器系统有经验。由于至少某些在本文描述的方法和系统不要求专职人员维护和 / 或更新数据处理工具,因此本文描述的方法和系统有助于减少与已知数据分析系统相关的成本。

[0103] 实施例至少部分涉及未结构化数据内的两个项目之间的相互关系的识别和 / 或观察的一致性的识别。描述的实施例操作为设置未结构化数据,使得关联存储器软件能够处理它。这种预处理开创了进一步处理的机会,例如该技术可以应用于图像中的元数据、元数据标准以及网站中元数据的检查。总之,这些实施例识别和标签化未结构化数据内的相关数据段以建立改善的数据分析系统,例如关联存储器系统、商业智能应用程序、搜索引擎和 / 或图像关联存储器系统。

[0104] 有利地,本文描述的方法和系统允许用户利用来自自主应用程序本身的具体数据建立数据处理工具。例如,利用上述实施例基于“实际数据”(示例案例)生成数据处理工具,这可以改善数据处理工具,使其比许多常规的基于规则的系统更鲁棒、精确、准确。例如,许多常规的基于规则的系统需要专家(例如天赋的编程语言专家)去捕捉一个或更多特定域项目,例如零件号、序列号等,和 / 或识别感兴趣模式并且生成正确识别信息的规则 / 代码。

[0105] 而且,利用本发明的实施例,系统用户可以识别示例案例并使用识别的示例在例如数据处理的下一个周期更新期间回流信息(例如数据片),由此建立数据处理系统。因此,本发明的实施例可以用初始数据的仅一部分工作。因此,与许多常规神经网络相比,这个新颖的系统避免了大量训练数据的要求。最终,非常熟悉数据(例如实际数据)的用户可以识别感兴趣的项目(例如样板文件)并且将其内容输入到数据处理工具中;因此,可以在下一次包含未结构化和 / 或部分结构化数据的问题空间被处理或当更新数据被添加到系统时应用更新到数据处理工具。

[0106] 依照本文所述的系统和方法处理数据减少了合并到主应用程序的数据(例如文本)的总量,提高了数据合并的速度,减少了用于存储数据的存储器数量,以及提高了数据能够被获取的速度。进一步地,由于至少某些本文所描述系统和方法不要求专职人员维护和 / 或更新数据处理工具,因此本文所描述的方法和系统有助于减少与已知数据分析系统相关的成本。

[0107] 本文所描述的方法和系统可以被编码为包括在计算机可读介质(包括但不限于,

存储装置或计算机装置的存储器区域)中的可执行指令。当由一个或更多处理器执行时,这类指令使处理器执行本文所描述方法的至少一部分。正如本文所使用的,“存储装置”是有形物件,例如可操作来存储数据的硬盘驱动器、固态存储器装置和 / 或光盘。

[0108] 虽然本发明各实施例的具体特征可能在某些图中示出而在其他图中没有示出,但是这仅是为了方便。根据本发明的原理,附图的任何特征可以与任何其他附图的任何特征结合引用和 / 或要求。

[0109] 书面描述使用示例公开了各种实施例,包括最佳模式,使本领域技术人员能够实施那些实施例,包括制造和使用任何装置或系统以及执行任何合并的方法。专利性范围由权利要求限定,而且可以包括本领域技术人员想到的其他示例。若这些其他示例具有并非不同于权利要求的文字语言的结构元件,或若它们包括与权利要求的文字语言相比非实质性区别的等价结构元件,则它们意在处于权利要求的范围内。

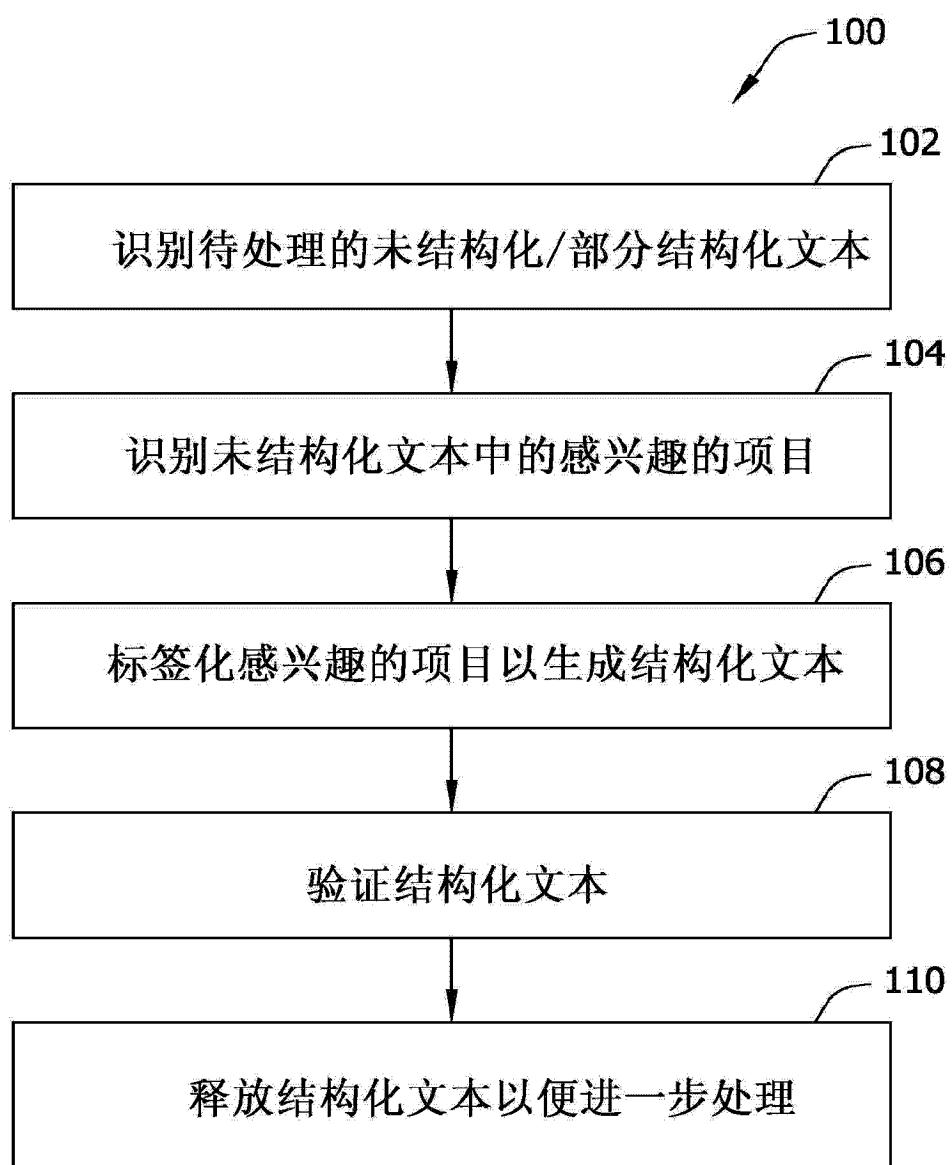


图 1

200

Henry David Thoreau (b. 1817, d. 1862). This eccentric American author and naturalist was born at Concord, Mass. He graduated at Harvard University in 1837. He was a good English and classical scholar, and was well acquainted with the literature of the East. His father was a maker of lead pencils, and he followed the business for a time, but afterwards supported himself mainly by teaching, lecturing, land surveying, and carpentering. In 1845 he built himself a small wooden house near Concord, on the shore of Walden Pond, where he lived about two years. The distance from his porch to the water's edge was 1234-1255 feet depending upon the height of the lake on any given year. The current caretaker of his estate recently installed a Kohler K-2345-1 model sink. The part number of the sink is 2345-1. He was intimate with Hawthorne, Emerson, and other literary celebrities. His principle works are "Walden, or Life in the Woods," "A Week on Concord and Merrimac Rivers," "Excursions," "Maine Woods," "Cape Cod," "A Yankee in Canada," and "Letters to Various Persons." In descriptive power Mr. Thoreau has few, if any, superiors.

202

图 2A

204 204 204 202
Henry David Thoreau (b. **1817**, d. **1862**). This eccentric American author and naturalist was born at **Concord**, Mass. He graduated at **Harvard University** in **1837**. He was a good English and classical scholar, and was well acquainted with the literature of the East. His father was a maker of lead pencils, and he followed the business for a time, but afterwards supported himself mainly by teaching, lecturing, land surveying, and carpentering. In **1845** he built himself a small wooden house near **Concord**, on the shore of Walden Pond, where he lived about two years. The distance from his porch to the water's edge was 1234-1255 feet depending upon the height of the lake on any given year. The current caretaker of his estate recently installed a Kohler K-2345-1 model sink. The part number of the sink is **2345-1**.
He was intimate with **Hawthorne**, **Emerson**, and other literary celebrities. His principle works are "**Walden, or Life in the Woods**," "**A Week on Concord and Merrimac Rivers**," "**Excursions**," "**Maine Woods**," "**Cape Cod**," "**A Yankee in Canada**," and "**Letters to Various Persons**." In descriptive power Mr. **Thoreau** has few, if any, superiors.
204

图 2B

[author]Henry David Thoreau[/author] (b.[year] 1817[/year], d. [year] 1862[/year]). This eccentric American author and naturalist was born at [city] Concord[/city], Mass. He graduated at [college_name] Harvard University[/college_name] in [year] 1837[/year]. He was a good English and classical scholar, and was well acquainted with the literature of the East. His father was a maker of lead pencils, and he followed the business for a time, but afterwards supported himself mainly by teaching, lecturing, land surveying, and carpentering. In [year] 1845[/year] he built himself a small wooden house near [city] Concord[city], on the shore of Walden Pond, where he lived about two years. The distance from his porch to the water's edge was [part_number]1234-1[/part_number]255 feet depending upon the height of the lake on any given year. The current caretaker of his estate recently installed a Kohler K-[part_number]2345-1[/part_number] model sink. The part number of the sink is [part_number]2345-1[/part_number]. He was intimate with [author]Hawthorne[/author], [author]Emerson[/author], and other literary celebrities. His principle works are "[book_title]Walden, or Life in the Woods[/book_title]," "[book_title]A Week on Concord and Merrimac Rivers[/book_title]," "[book_title]Excursions[/book_title]," "[book_title]Maine Woods[/book_title]," "[book_title]Cape Cod[/book_title]," "[book_title]A Yankee in Canada[/book_title]," and "[book_title]Letters to Various Persons[/book_title]." In descriptive power Mr. [author]Thoreau[/author] has few, if any, superiors.

图 2C

206 208 204 207

[author][i01]Henry David Thoreau[/author] (b.[year][i03]1817[/year], d. [year][i03]1862[/year]). This eccentric American author and naturalist was born at [city][i04]Concord[/city], Mass. He graduated at [college_name][i18]Harvard University[/college_name] in [year][i03]1837[/year]. He was a good English and classical scholar, and was well acquainted with the literature of the East. His father was a maker of lead pencils, and he followed the business for a time, but afterwards supported himself mainly by teaching, lecturing, land surveying, and carpentering. In [year][i03]1845[/year] he built himself a small wooden house near[city][i04]Concord[/city], on the shore of Walden Pond, where he lived about two years. The distance from his porch to the water's edge was

220 [part_number][i05][i14]1234-1[/part_number]255 feet depending upon
 214 the height of the lake on any given year. The current caretaker of his estate recently installed a Kohler K-[part_number][i05]2345-1[/part_number] model sink. The part number of the sink is [part_number][i05]2345-1[/part_number]. He was intimate with [author][i01]Hawthorne[/author],
 [author][i01]Emerson[/author], and other literary celebrities. His principle works are "[book_title][i01]Walden, or Life in the Woods[/book_title]," "[book_title][i02]A Week on Concord and Merrimac Rivers[/book_title]," "[book_title][i02]Excursions[/book_title]," "[book_title][i01]Maine Woods[/book_title]," "[book_title][i02]Cape Cod[/book_title]," "[book_title][i01]A Yankee in Canada[/book_title]," and "[book_title][i01]Letters to Various Persons[/book_title]." In descriptive power Mr. [author][i01]Thoreau[/author] has few, if any, superiors.

216 204 206

图 2D

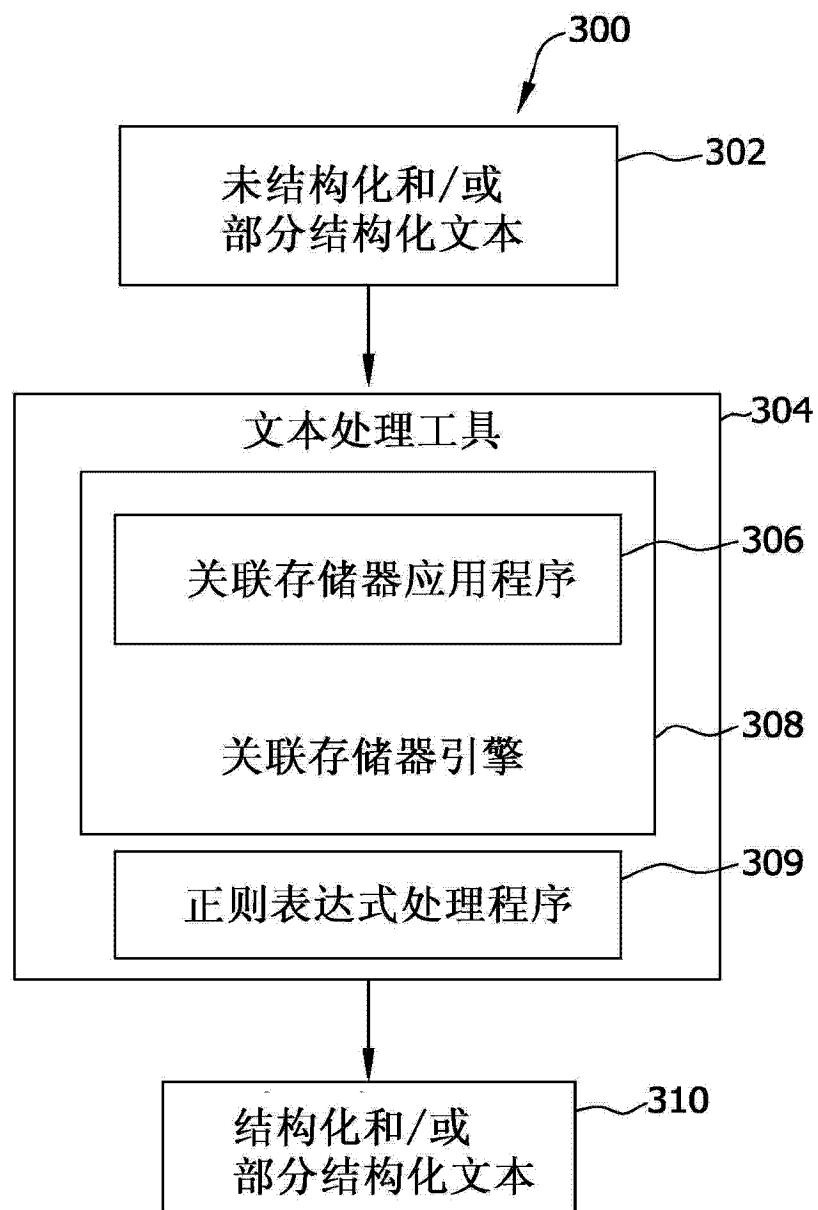


图 3

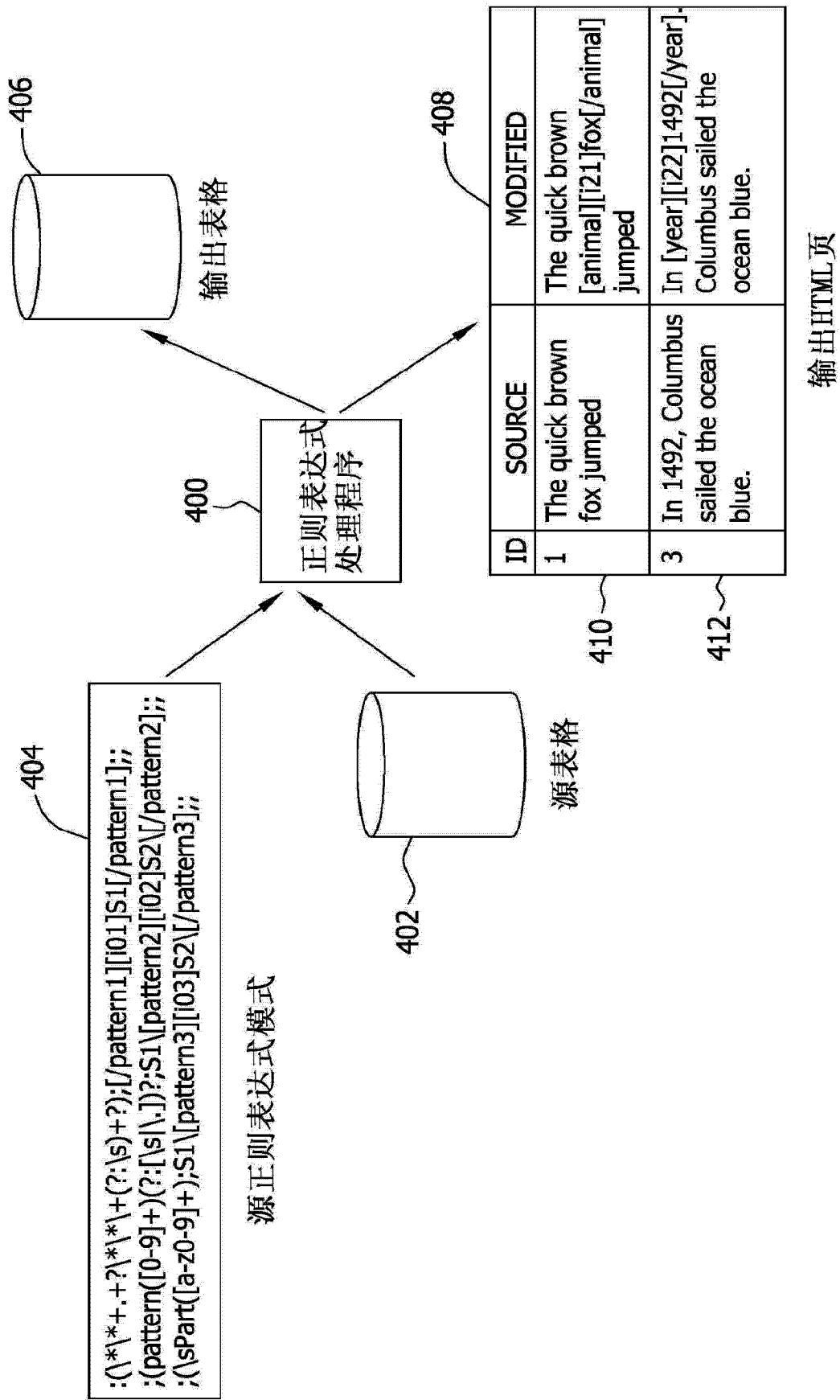


图 4

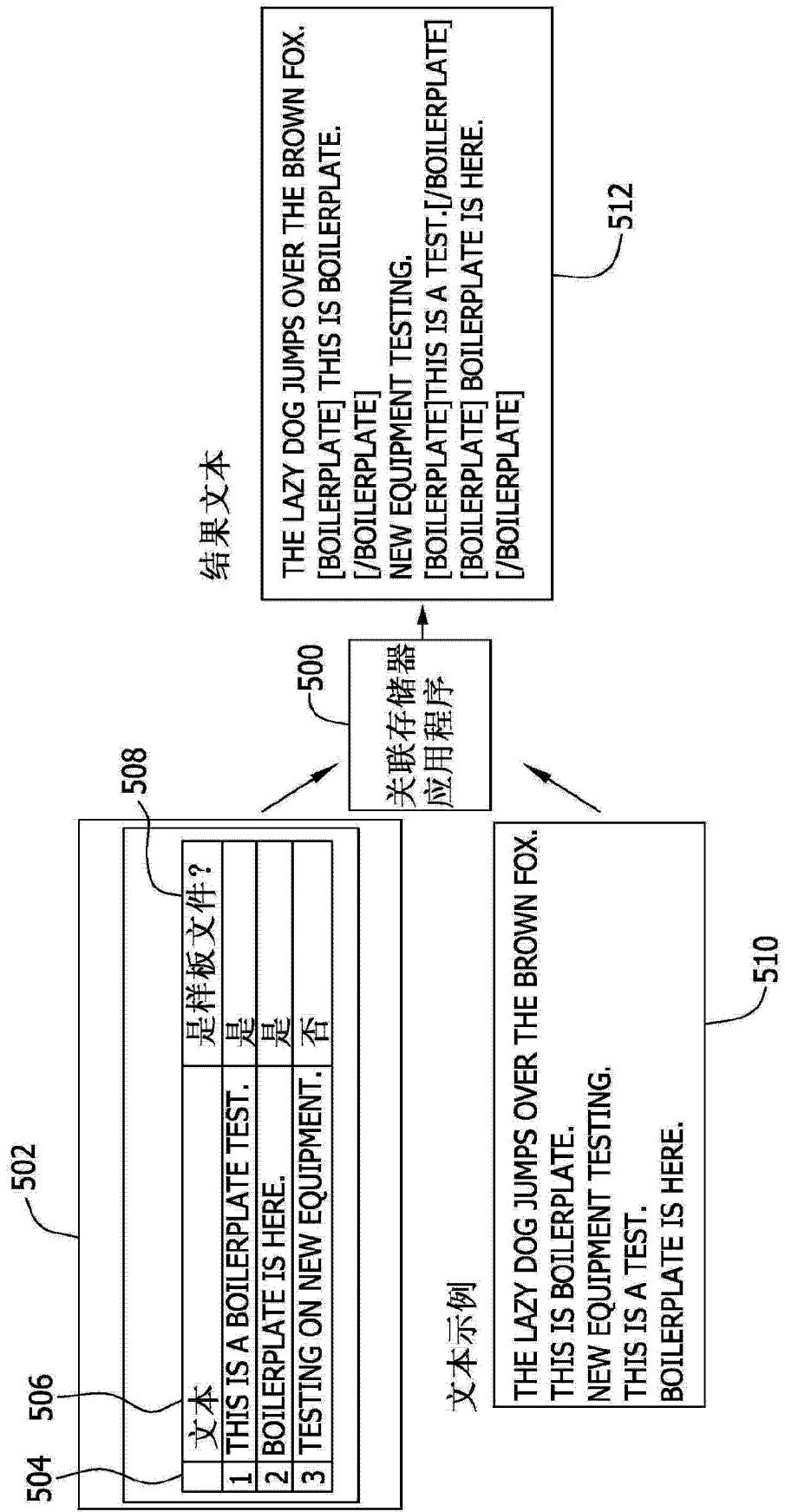


图 5

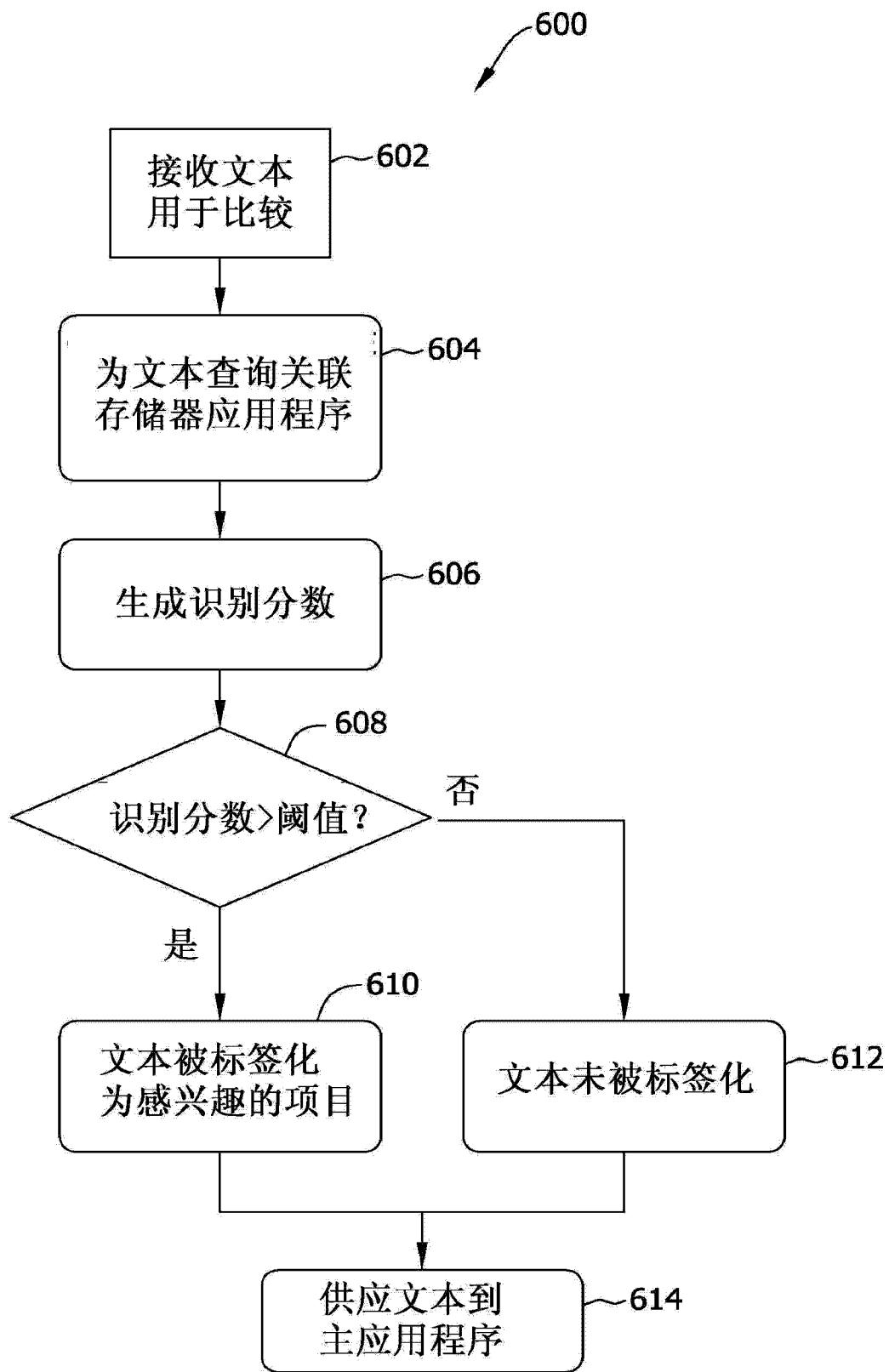


图 6

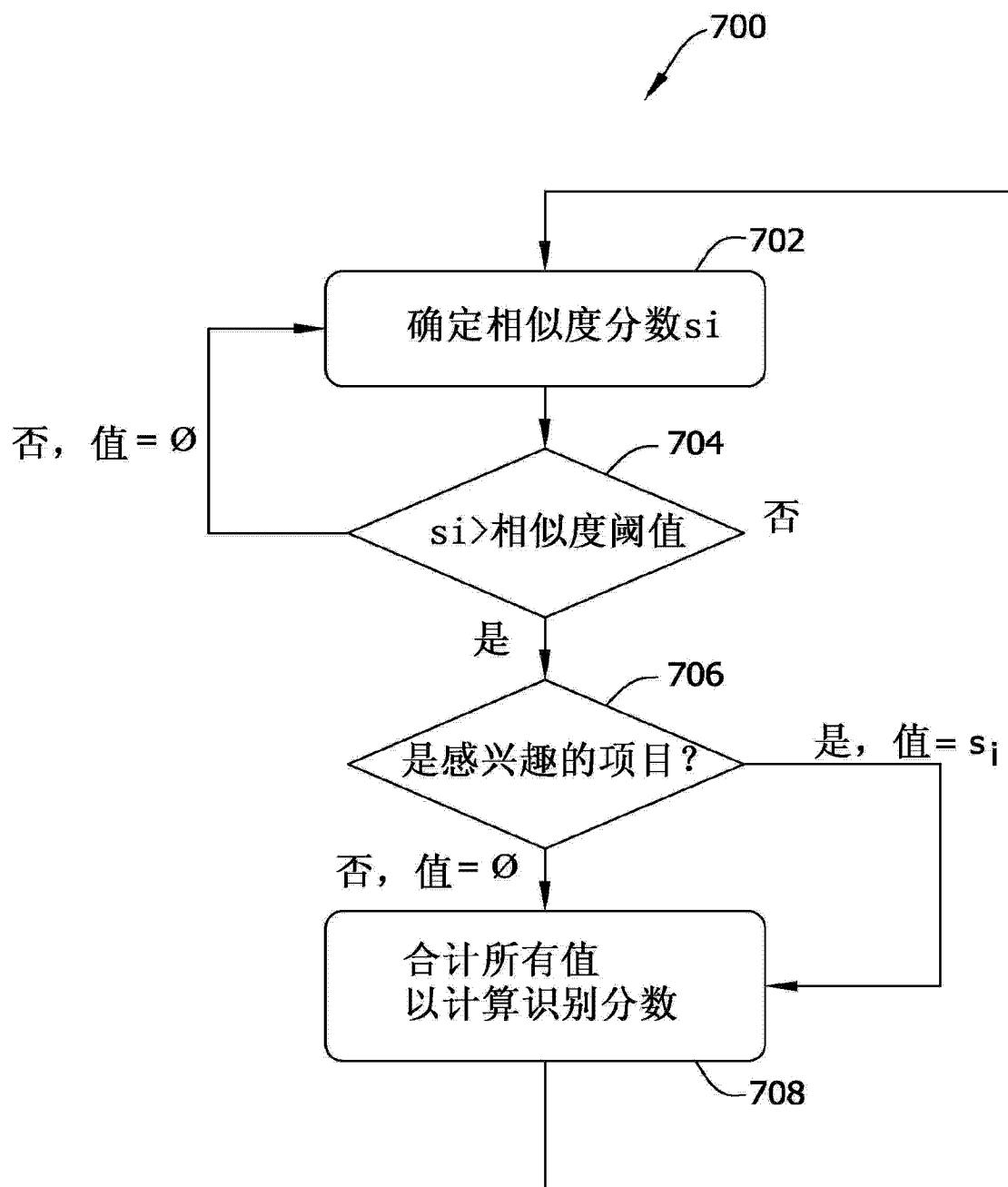


图 7

802

THE BOEING COMPANY

**SUBJECT: Inspections of carriage spindle of outboard mid-flaps-FAA
Proposed Rule Withdrawal**

Summary:

The following is a notification of ref/a/Proposed Rule Withdrawal Docket Number 2002-NM-219-AD. Since the issuance of the supplemental NPRM, the Federal Aviation Administration (FAA) has issued another AD that adequately addresses the identified unsafe condition. Accordingly, the proposed rule is withdrawn.

This action withdraws a supplemental notice proposed rulemaking (NPRM) that proposed a new airworthiness directive (AD), applicable to all Boeing Model 737-100,-200,-200C,-300,-400 and -500 series airplanes. That action would have superseded an existing AD that currently requires repetitive inspections to find cracks, fractures, or corrosion of each carriage spindle of the left and right outboard mid-flaps; and corrective action, if necessary. That action would also have mandated the previously optional overhaul or replacement of the carriage spindles, which would have ended the repetitive inspections required by the existing AD.

804

The complete text of the subject Proposed Rule withdrawal was noted in the Federal Register and will be accessible via the following World Wide Web address:

<http://www.gpoaccess.gov/fr/browse.html>

806

[boilerplate]

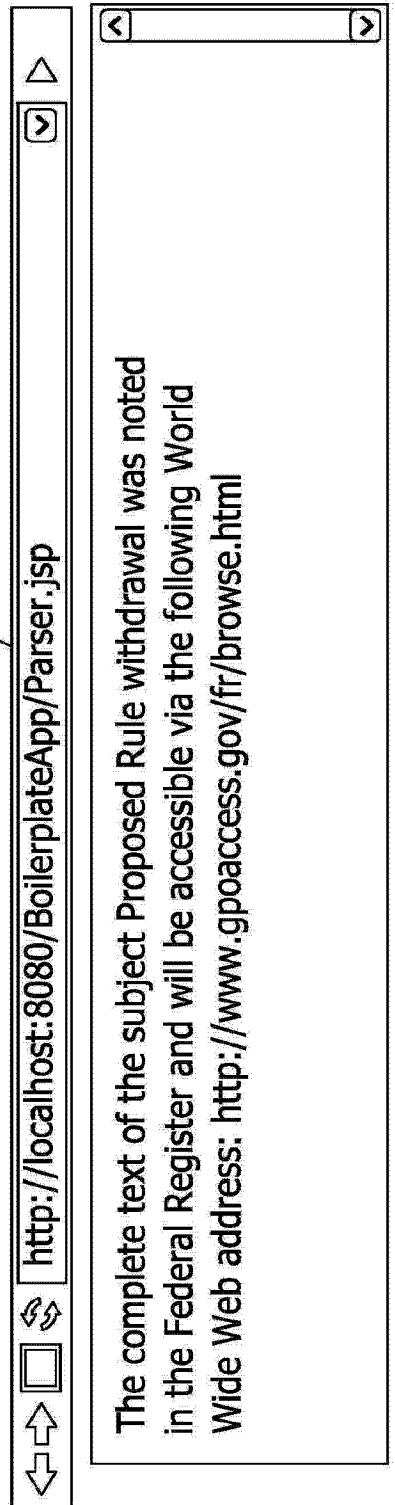
If you need further information regarding the subject, please direct your request to your local Boeing Field Service Representative. If your local Field Service Representative is not available you may contact Airline Support Manager at the address noted on the top of this message or call (123) 456-7890.

John V. President
Vice President, Fleet and Airline Support

Boeing
Commercial Aviation Services **[/boilerplate]**

图 8A

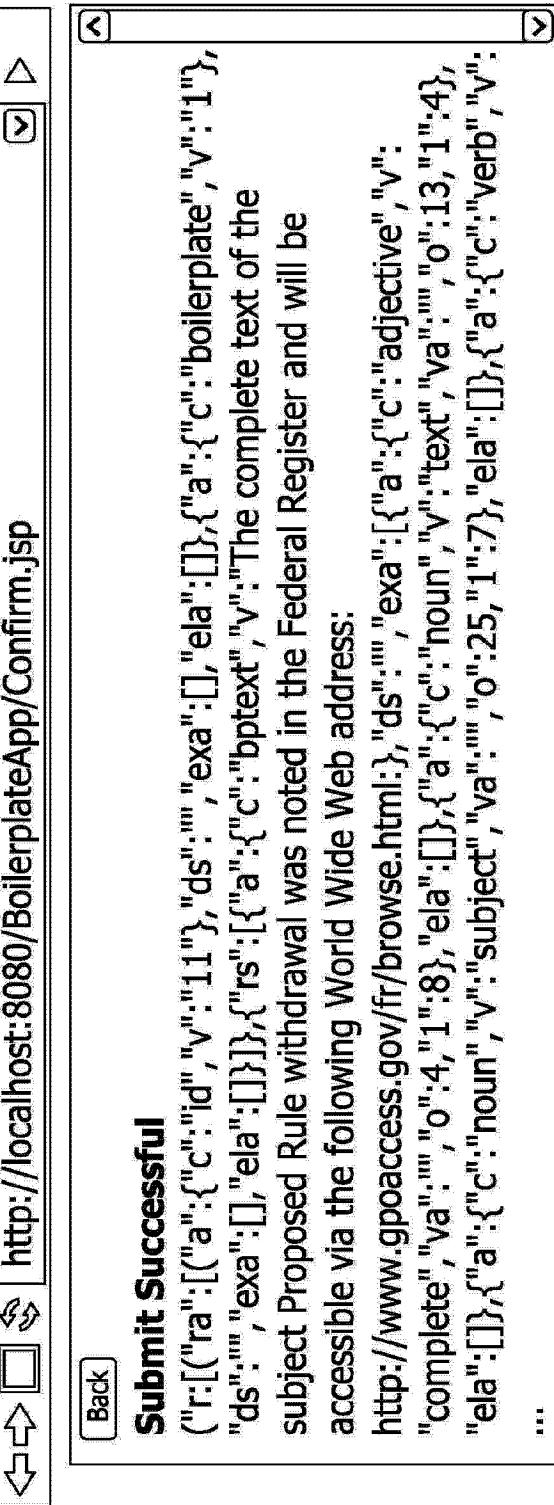
808



The complete text of the subject Proposed Rule withdrawal was noted in the Federal Register and will be accessible via the following World Wide Web address: <http://www.gpoaccess.gov/fr/browse.html>

810

812



Submit Successful

```
{"r": [{"ra": [{"c": "id", "v": "11"}, {"ds": "", "exa": "", "ela": []}], "a": [{"c": "boilerplate", "v": "1"}], "ds": "", "exa": "", "ela": []}], "rs": [{"a": [{"c": "bpptext", "v": "The complete text of the subject Proposed Rule withdrawal was noted in the Federal Register and will be accessible via the following World Wide Web address: http://www.gpoaccess.gov/fr/browse.html;"}]}]}
```

图 8C

图 8B

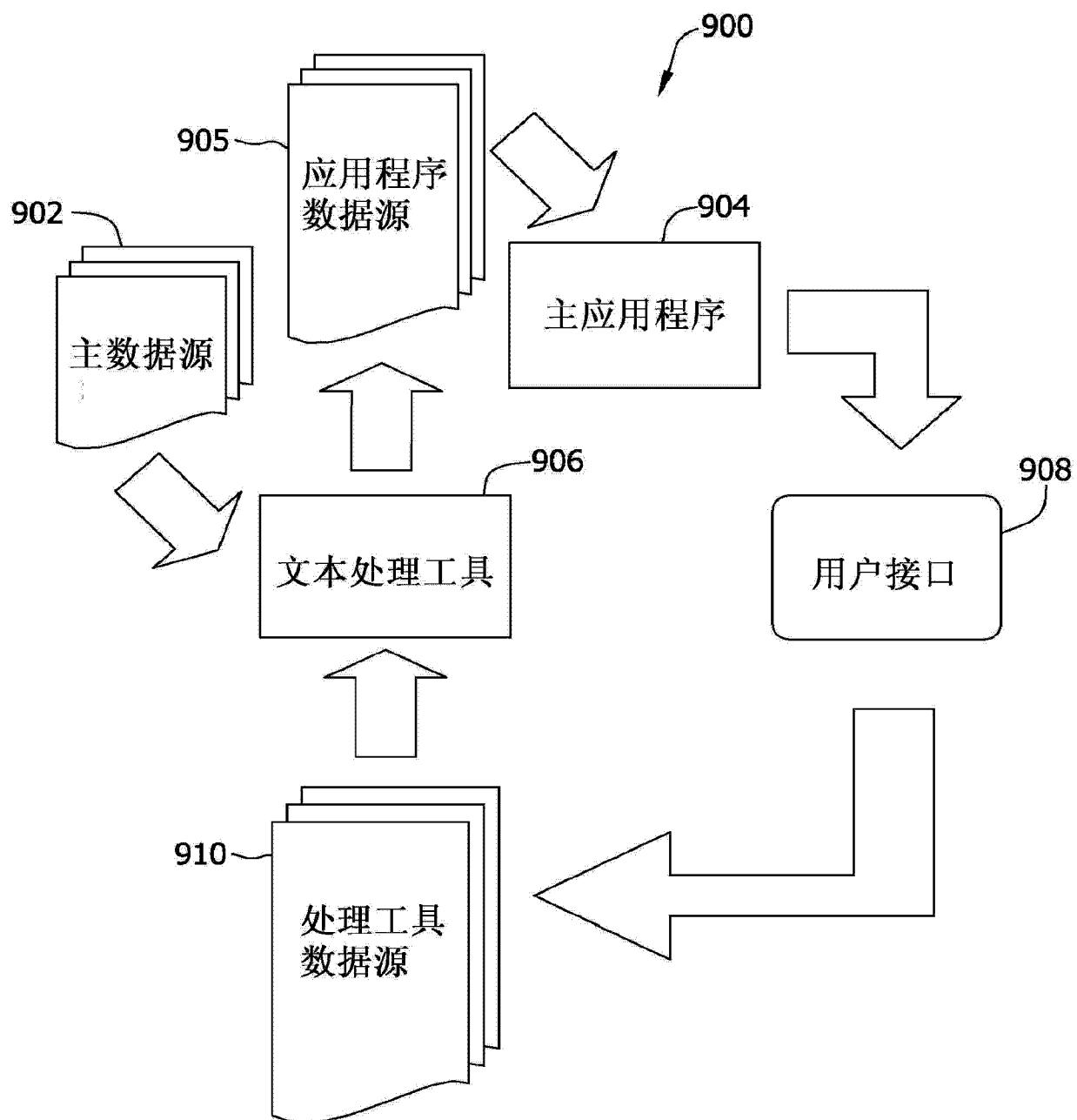


图 9

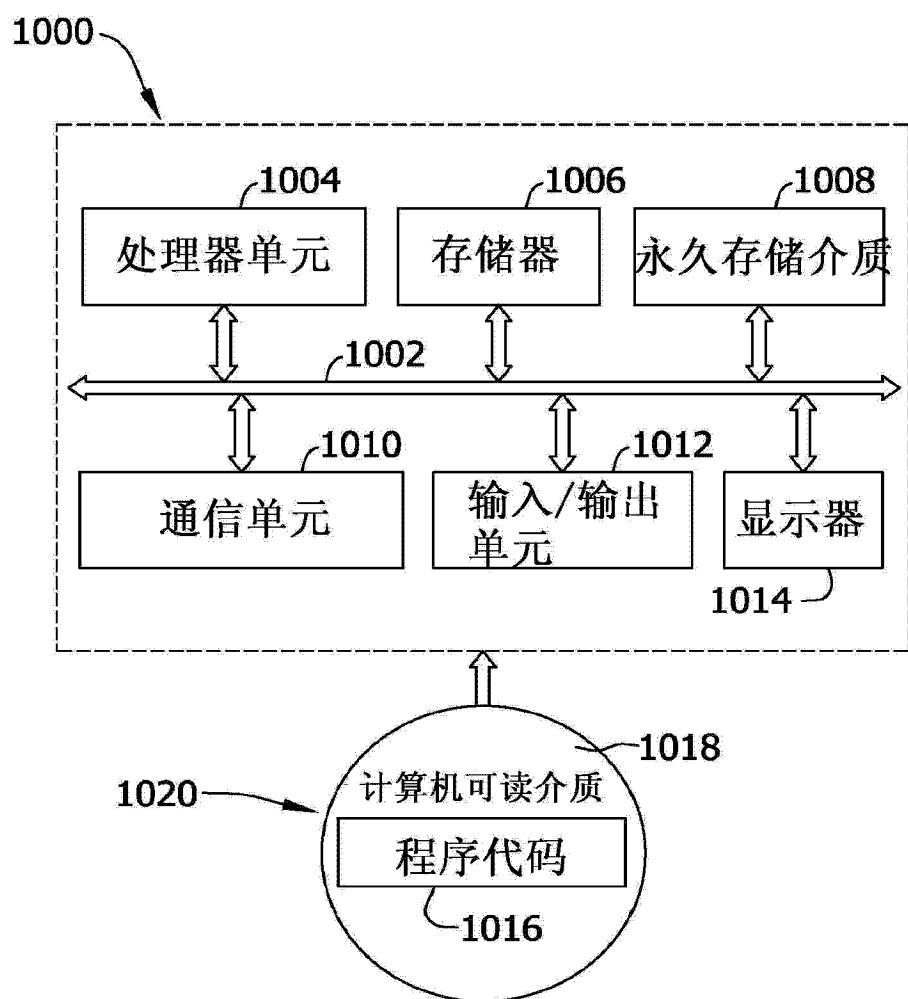


图 10