



(12) 发明专利

(10) 授权公告号 CN 110622157 B

(45) 授权公告日 2021.06.01

(21) 申请号 201780086816.2

(22) 申请日 2017.12.20

(65) 同一申请的已公布的文献号
申请公布号 CN 110622157 A

(43) 申请公布日 2019.12.27

(30) 优先权数据
15/384,870 2016.12.20 US

(85) PCT国际申请进入国家阶段日
2019.08.19

(86) PCT国际申请的申请数据
PCT/US2017/067551 2017.12.20

(87) PCT国际申请的公布数据
W02018/119051 EN 2018.06.28

(73) 专利权人 里德爱思唯尔股份有限公司雷克
萨斯尼克萨斯分公司
地址 美国俄亥俄州

(72) 发明人 P·F·阿拉贡伊威廉姆斯

(74) 专利代理机构 上海专利商标事务所有限公
司 31100
代理人 周全 张鑫

(51) Int.Cl.
G06F 16/93 (2019.01)
G06F 40/284 (2020.01)

(56) 对比文件
CN 103699598 A, 2014.04.02
CN 104636384 A, 2015.05.20
US 2015278349 A1, 2015.10.01
US 2013262968 A1, 2013.10.03
US 2015143329 A1, 2015.05.21
CN 103578073 A, 2014.02.12

审查员 李婉怡

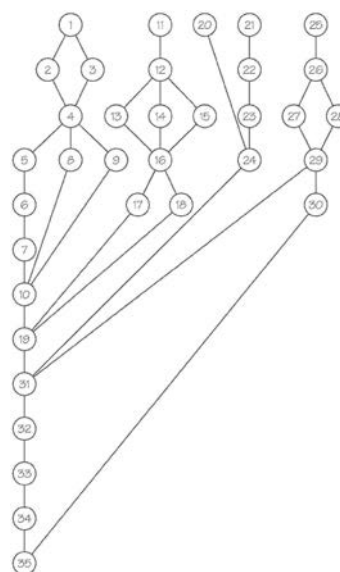
权利要求书3页 说明书17页 附图10页

(54) 发明名称

用于确定专利权利要求中的引用的系统和方法

(57) 摘要

公开了确定专利权利要求之间关系以及生成指示该关系的精简权利要求树的系统和方法。关系的方法包括：确定权利要求列表中的权利要求；搜索每一个权利要求中的引用，该引用涉及权利要求中的另一个权利要求；对于包含唯一引用的每一个权利要求，确定每一个唯一引用的相关联的编号小于包含该唯一引用的权利要求的编号；向与权利要求列表相应的数据文件附加基于相关联的编号指示权利要求的所有先行的先行数据以及与该权利要求对应的自我引用数据；确定包含父权利要求和先行权利要求的权利要求；以及附加关于该父权利要求和先行权利要求的该数据文件，以便指示该权利要求与父权利要求以及该权利要求与先行权利要求之间的关系。



1. 一种用于确定和指示专利权利要求之间的关系的方法,所述方法包括:
 - 由处理设备确定权利要求列表中的一个或多个权利要求;
 - 由所述处理设备查找所述一个或多个权利要求中的每一个权利要求中的至少一个引用,其中所述至少一个引用涉及所述权利要求列表中的所述一个或多个权利要求中的另一个权利要求;以及
 - 对于所述一个或多个权利要求中包含一个或多个唯一引用的每一个权利要求:
 - 由所述处理设备确定所述一个或多个唯一引用中的每一个唯一引用的相关联的编号小于包含所述一个或多个唯一引用的权利要求的编号,
 - 由所述处理设备向与所述权利要求列表相对应的数据文件附加基于所述相关联的编号指示所述权利要求的所有先行权利要求的先行数据,
 - 由所述处理设备向所述数据文件附加与所述权利要求相对应的自我引用数据,
 - 由所述处理设备确定所述权利要求包括一个或多个父权利要求,
 - 由所述处理设备附加所述数据文件以用于相对于与所述一个或多个父权利要求有关的信息指示所述权利要求和所述一个或多个父权利要求之间的关系,
 - 由所述处理设备确定所述权利要求包括一个或多个先行权利要求,以及
 - 由所述处理设备附加所述数据文件以用于相对于与所述一个或多个先行权利要求相关的信息指示所述权利要求和所述一个或多个先行权利要求之间的关系以使得所述数据文件指示专利权利要求之间的所述关系。
2. 如权利要求1所述的方法,进一步包括:
 - 由所述处理设备确定所述权利要求包含所述至少一个引用;以及
 - 由所述处理设备确定所述至少一个引用包含的权利要求编号是否小于包含所述至少一个引用的权利要求的编号。
3. 如权利要求2所述的方法,进一步包括:
 - 当所述至少一个引用包含的权利要求编号大于或等于包含所述至少一个引用的权利要求时,由所述处理设备向所述数据文件附加指示所述权利要求为独立权利要求的独立权利要求数据。
4. 如权利要求1至3中任一项所述的方法,进一步包括:
 - 当所述一个或多个权利要求中的权利要求不包含所述至少一个引用时,由所述处理设备向所述数据文件附加指示所述权利要求为独立权利要求的独立权利要求数据。
5. 如权利要求1至3中任一项所述的方法,进一步包括:
 - 由所述处理设备基于所述数据文件生成精简权利要求树。
6. 如权利要求1至3中任一项所述的方法,其特征在于,确定所述权利要求列表中的所述一个或多个权利要求包括确定所述一个或多个权利要求中的每一个权利要求以及所述一个或多个权利要求中的每一个权利要求的相关联的编号。
7. 如权利要求1至3中任一项中所述的方法,其特征在于,所述权利要求的所有先行权利要求包括由所述权利要求引用并且不是所述权利要求的父权利要求的其他权利要求。
8. 如权利要求1至3中任一项所述的方法,其特征在于,所述一个或多个父权利要求为所述权利要求的一个或多个直接先行权利要求。
9. 如权利要求1至3中任一项所述的方法,其特征在于,所述数据文件为XML文件。

10. 一种用于确定和指示专利权利要求之间的关系的系统,所述系统包括:
处理设备;以及

非瞬态处理器可读存储介质,所述非瞬态处理器可读存储介质与所述处理设备进行沟通,其中,所述非瞬态处理器可读存储介质包括一个或多个编程指令,当所述一个或多个编程指令被执行时使得所述处理设备:

确定权利要求列表中的一个或多个权利要求,

查找所述一个或多个权利要求中的每一个权利要求中的至少一个引用,其中,所述至少一个引用涉及所述权利要求列表中的所述一个或多个权利要求中的另一个,以及

对于所述一个或多个权利要求中包含一个或多个唯一引用的每一个权利要求:

确定所述一个或多个唯一引用中的每一个唯一引用的相关联的编号小于包含所述一个或多个唯一引用的权利要求的编号,

向与所述权利要求列表相对应的数据文件附加基于所述相关联的编号指示所述权利要求的所有先行权利要求的先行数据,

向所述数据文件附加与所述权利要求相对应的自我引用数据,

确定所述权利要求包含一个或多个父权利要求,

附加所述数据文件以便关于与所述一个或多个父权利要求有关的信息,指示所述权利要求和所述一个或多个父权利要求之间的关系,

确定所述权利要求包括一个或多个先行权利要求,以及

附加所述数据文件以便关于与所述一个或多个先行权利要求有关的信息指示所述权利要求和所述一个或多个先行权利要求之间的关系以使得所述数据文件指示专利权利要求之间的所述关系。

11. 如权利要求10所述的系统,其特征在于,所述非瞬态处理器可读存储介质进一步包括一个或多个编程指令,当所述一个或多个编程指令被执行时使得所述处理设备:

确定所述权利要求包含所述至少一个引用;以及

确定所述至少一个引用包含的权利要求编号是否小于包含所述至少一个引用的权利要求的权利要求编号。

12. 如权利要求11所述的系统,其特征在于,所述非瞬态处理器可读存储介质进一步包括一个或多个编程指令,当所述一个或多个编程指令被执行时使得所述处理设备:

向所述数据文件附加独立权利要求数据,所述独立权利要求数据在所述至少一个引用包含的权利要求编号大于或等于包含所述至少一个引用的权利要求的编号时指示所述权利要求为独立权利要求。

13. 如权利要求10所述的系统,其特征在于,所述非瞬态处理器可读存储介质进一步包括一个或多个编程指令,当所述一个或多个编程指令被执行时使得所述处理设备:

当所述一个或多个权利要求中的权利要求不包含所述至少一个引用时,向所述数据文件附加指示所述权利要求为独立权利要求的独立权利要求数据。

14. 如权利要求10所述的系统,其特征在于,所述非瞬态处理器可读存储介质进一步包括一个或多个编程指令,当所述一个或多个编程指令被执行时使得所述处理设备:

基于所述数据文件生成精简权利要求树。

15. 如权利要求10所述的系统,其特征在于,当所述一个或多个编程指令被执行时,使

得所述处理设备确定所述权利要求列表中的所述一个或多个权利要求进一步使得所述处理设备确定所述一个或多个权利要求中的每一个权利要求以及所述一个或多个权利要求中的每一个权利要求的相关联的编号。

16. 如权利要求10所述的系统,其特征在于,所述权利要求的所有先行权利要求包括由所述权利要求引用并且不是所述权利要求的父权利要求的其他权利要求。

17. 如权利要求10所述的系统,其特征在于,所述一个或多个父权利要求为所述权利要求的一个或多个直接先行权利要求。

18. 如权利要求10所述的系统,其特征在于,所述数据文件为XML文件。

用于确定专利权利要求中的引用的系统和方法

[0001] 相关申请的交叉引用

[0002] 本申请要求于2016年12月20日提交的题为“用于确定专利权利要求中的引用的系统和方法 (SYSTEMS AND METHODS FOR DETERMINING REFERENCES IN PATENT CLAIMS)”的美国专利申请序列第15/384,870号的优先权,其以其整体通过引用并入本文。

技术领域

[0003] 本说明书总体上涉及确定专利文档中的权利要求引用,并且更具体地,涉及确定专利文档中权利要求从属关系并且依此创建精简权利要求树的系统和方法。

背景技术

[0004] 当前,成像和/或数字化呈现的专利文档(诸如已公开的专利和已公开的专利申请)需要光学字符识别(OCR)软件将该图像和/或数字化呈现内容转换为计算机可读文本。当前可以用于将专利文档的权利要求部分的图像和/或数字化呈现内容转换为计算机可读文本的OCR软件存在一定不足,因为该OCR软件并未在诸如XML文件之类的数据文件中准确地提供关于从属关系的信息,而无关乎撰写权利要求的语言和句法。此外,该OCR转换可能会包括不易被电子装置检测到的错误。又及,OCR软件生成的数据无法被用于快速创建可以用于验证经确定的权利要求从属关系的准确性的权利要求树。

[0005] 因此,存在对于如下系统和方法的需求,该系统和方法能够获取成像和/或数字化呈现的专利权利要求,快速并且准确地确定权利要求的从属关系而无关乎在撰写权利要求时使用的语言或句法,检测并校正错误并且提供置信度水平,并且基于经确定的权利要求从属关系提供权利要求树。

发明内容

[0006] 在一个实施例中,用于确定专利权利要求之间的一个或多个关系的方法包括由处理设备确定权利要求列表中的一个或多个权利要求;由处理设备查找所述一个或多个权利要求中的每一个权利要求中的至少一个引用,其中该至少一个引用涉及权利要求列表中的一个或多个权利要求中的另一权利要求;以及对于所述一个或多个权利要求中包含一个或多个唯一引用的每一个权利要求,由处理设备确定该一个或多个唯一引用中的每一个唯一引用的相关联的权利要求编号小于包含该一个或多个唯一引用的权利要求的编号;由处理设备向与权利要求列表相对应的数据文件附加基于相关联的编号指示权利要求的所有先行的先行数据;由处理设备向所述数据文件附加与该权利要求相对应的自我引用数据;由处理设备确定该权利要求包括一个或多个父权利要求;由处理设备附加数据文件以便关于与所述一个或多个父权利要求有关的信息,指示该权利要求和一个或多个父权利要求之间的关系;由处理设备确定该权利要求包括一个或多个先行权利要求;以及由处理设备附加数据文件以便关于与一个或多个先行权利要求有关的信息,指示该权利要求和该一个或多个先行权利要求之间的关系。

[0007] 在另一实施例中,用于确定专利权利要求之间的一个或多个关系的系统包括处理设备和与所述处理设备通信的非瞬态处理器可读存储介质。该非瞬态处理器可读存储介质包括一个或多个编程指令,当该编程指令被执行时使得处理设备:确定权利要求列表中的一个或多个权利要求;查找该一个或多个权利要求中的每一个权利要求中的至少一个引用,该至少一个引用涉及权利要求列表中的一个或多个权利要求中的另一个;对于该一个或多个权利要求中包含一个或多个唯一引用的每一个权利要求,确定该一个或多个唯一引用中的每一个唯一引用的相关联的编号小于包含该一个或多个唯一引用的权利要求的编号;向与该权利要求列表相对应数据文件附加基于相关联的编号指示该权利要求的所有先行的先行数据;向该数据文件附加与该权利要求相对应的自我引用数据;确定该权利要求包括一个或多个父权利要求;附加该数据文件以便关于与该一个或多个父权利要求有关的信息,指示该权利要求和一个或多个父权利要求之间的关系;确定该权利要求包括一个或多个先行权利要求;以及附加该数据文件以便关于与该一个或多个先行权利要求相关的信息,指示该权利要求和一个或多个先行权利要求之间的关系。

[0008] 在又一实施例中,用于电子地生成指示专利权利要求之间的一个或多个关系的精简权利要求树的方法包括:对于专利权利要求中的每一个权利要求,由处理设备确定权利要求的纵向顺序;由处理设备向与专利权利要求的权利要求列表相对应的数据文件附加与纵向顺序相应的纵向顺序数据;由处理设备确定所述权利要求的横向顺序;由所述处理设备向所述数据文件附加与所述横向顺序相对应的横向顺序数据;由所述处理设备确定所述权利要求的横向位置;由所述处理设备向所述数据文件附加与所述横向位置相对应的横向位置数据;由所述处理设备确定所述权利要求的一个或多个有序的父权利要求;由所述处理设备向所述数据文件附加与所述一个或多个有序的父权利要求相对应的有序的父权利要求数据;由所述处理设备确定所述权利要求的最右端父权利要求;由所述处理设备向所述数据文件附加与所述最右端父权利要求相对应的最右端父权利要求数据;由所述处理设备确定所述权利要求的一个或多个有序的子权利要求;以及由所述处理设备向所述数据文件附加与所述一个或多个有序的子权利要求相对应的有序的子权利要求数据。所述纵向顺序、所述横向顺序、所述横向位置、所述一个或多个有序的父权利要求、所述最右端父权利要求以及所述一个或多个有序的父权利要求基于生成的关系数据而确定。所述数据文件提供用于生成精简权利要求树的信息。

[0009] 鉴于以下具体描述并结合附图将更完整地理解由本文中描述的实施例所提供的这些特征和附加特征。

附图说明

[0010] 在附图中阐述的实施例在性质上是说明性和示例性的,并不意图限制由权利要求定义的主题。可以在结合以下附图阅读时理解说明性实施例的以下详细描述,其中相同的结构用相同的参考标号表示,并且在附图中:

[0011] 图1是根据本文中所示出和所描述的一个或多个实施例的用于确定专利权利要求中的引用的系统的说明性计算网络的示意性描绘;

[0012] 图2是图1的服务器计算设备的示意性描绘,进一步图示了根据本文中所示出和所描述的一个或多个实施例的可用于确定专利权利要求中的引用的硬件和软件;

[0013] 图3描绘了根据本文示出和描述的一个或多个实施例的用于分析计算机可读文档以便确定专利权利要求的说明性方法的流程图；

[0014] 图4描绘了根据本文示出和描述的一个或多个实施例的用于确定专利权利要求中的引用的说明性方法的流程图；

[0015] 图5A示意性地描绘了根据本文示出和描述的一个或多个实施例的说明性权利要求树；

[0016] 图5B示意性地描绘了根据本文示出和描述的一个或多个实施例的说明性精简权利要求树；

[0017] 图6示意性地描绘了根据本文示出和描述的一个或多个实施例的精简权利要求树的说明性过程；

[0018] 图7描绘了根据本文示出和描述的一个或多个实施例的用于生成权利要求树的说明性方法的流程图；

[0019] 图8描绘了根据本文示出和描述的一个或多个实施例的用于确定权利要求组中的多个权利要求的纵向顺序的说明性方法的流程图；

[0020] 图9描绘了根据本文示出和描述的一个或多个实施例的在精简权利要求树中重新安排权利要求的横向顺序的说明性过程；以及

[0021] 图10描绘了根据本文示出和描述的一个或多个实施例的用于确定权利要求组中的多个权利要求中的每一个权利要求的横向顺序的说明性方法的流程图。

具体实施方式

[0022] 通过总体上参考附图，本文描述的实施例针对用于获取包括专利文档中的权利要求的计算机可读文本（可包括获取专利文档的图像以及进行OCR过程以将权利要求转换为计算机可读和/或人类可读文本）、从计算机可读文本中确定并验证权利要求引用、以及基于所确定的权利要求引用来创建权利要求树的系统和方法。

[0023] 本文中所描述的系统和方法尤其被配置成用于快速确定权利要求引用，而无关乎撰写的权利要求的语言和句法。本文中所描述的系统和方法也能用于通过从多个不同角度分析每一个权利要求来检测OCR转换错误和/或排印错误，从而使得从属关系得到验证或校正。此外，通过本文中所描述的系统和方法所创建的权利要求树可以允许用户更快并且更轻松地将专利权利要求之间的关系可视化，特别是对于专利文档包含大量权利要求、多项从属权利要求、从属于一个或多个从属权利要求的权利要求等的情况。

[0024] 如本文中将更详细地描述的，因为本文中所描述的系统和方法允许计算设备分析任何专利权利要求，而无关乎语言、句法或来源，因此该系统和方法相较于其他计算设备改进了结合在其中的计算设备的功能。即，相同的编程（例如相同的基础引擎）可以用于确定以英语、西班牙语、德语、汉语普通话、韩语、日语或是其他任意语言的权利要求引用，而无需专用于每一种语言的基础编程。相反，语言专用的检测规则能存储在更小且更易访问的外部配置文件中。此外，本文中所描述的系统和方法通过以相较于使用现有的系统和方法所需的时间量大幅减少的时间量（例如，使用本文中所描述的系统和方法训练至多需要两到三天，而不是使用现有的系统和方法的两个以上）针对每一种语言对计算机进行训练来提升计算机的功能。

[0025] 类似地,相同的编程也可用于确定以不同句法格式撰写的权利要求引用。例如,如果方法权利要求5是从属于权利要求1-4中的每一个的多项从属权利要求,则该方法权利要求5可以以下述句法格式中的任意一种来撰写,每一种句法格式都具有相同的含义:“如权利要求1-4中任一项所述的方法,其特征在于, ”、“如权利要求1、2、3或4所述的方法,其特征在于, ”、“如前述权利要求中任一项所述的方法,其特征在于, ”、“根据权利要求1-4中任一项所述的方法,其特征在于, ”、“一种根据权利要求1-4中任一项所述的方法,其特征在于, ”以及“如权利要求1-4中任一项所述的方法。”其他系统或方法可能需要额外的处理步骤和/或潜在更多的处理能力来确定这些句法格式当中的每一个的含义,然后才能进行引用的确定。此外,如果权利要求包含排印错误,或根据编程无法识别的格式布置,则其他过程可能无法和/或错误地识别恰当的权利要求引用。相反,本文中所描述的系统和方法能够准确地确定恰当的权利要求引用,而无论句法(包括该句法是否是可识别的格式)和/或无论排印错误,因为该编程被设计成单独基于每一个权利要求并基于权利要求整体来分析和验证权利要求引用。

[0026] 应当理解,虽然本文中所描述的系统和方法可能尤其可用于包括多个权利要求的权利要求组(包括至少一个独立权利要求和至少一个从属权利要求),但本文中所描述的系统和方法的功能不限于此。例如,本文中所描述的系统和方法还能用于仅包含一个权利要求的专利文档和仅包含独立权利要求的专利文档(例如,不含从属权利要求或多项从属权利要求)。

[0027] 如本文中所使用,术语“专利文档”通常指任何包含专利权利要求的文档。也就是说,专利文档可以是已经公布的专利或专利相关文档(或专利相关文档的一部分)。然而,应当理解在一些实施例中,专利文档也有可以是尚未公布的文档,例如草案等等。说明性专利文档通常可以是已颁布的专利;已由国家专利实体(诸如但不限于美国专利商标局(USPTO)、中国国家知识产权局(SIPO)、日本特许厅(JPO)、韩国知识产权局(KIPO)和澳大利亚知识产权局等)公布的专利申请;已由国际或地区实体(诸如但不限于世界知识产权组织(WIPO)、欧洲专利局(EPO)和非洲地区知识产权组织(ARIPO)等)公布的专利申请;再颁布专利;外观设计专利或申请;植物专利或申请;临时专利申请;更正后专利;附加证明文件;实用新型;工业设计;特开文档;小专利、经济专利、发明人证明文件、前述文档中的任意文档的翻译和/或类似文档等。

[0028] 如本文中所使用,“权利要求”指的是经由特别措辞和构造的定义了专利文档的限制的语句。因此,权利要求定义了专利文档涵盖和不涵盖的范围。用于排除他人制造、使用或销售某物的专利权可以基于权利要求中所定义的内容。专利文档中始终存在至少一个权利要求,但是其具体位置可能取决于专利文档而变化(例如,权利要求可以在专利文档的开头或结尾等)。在一些实施例中,权利要求可以是有序的列表。在一些实施例中,权利要求之前可能会有指示权利要求的起始的标题或短语(诸如“要求保护的是……”,“本人主张……”或“权利要求组权利要求书”)。

[0029] 权利要求可以根据是否引用了其他权利要求来进行分类,如果权利要求引用了其他的权利要求,则可以根据他们如何引用其他权利要求来进行分类。未引用另一权利要求的权利要求是独立权利要求,引用了有序权利要求列表中的至少一项先前出现的权利要求的权利要求为从属权利要求。基于如何引用另一权利要求来指示特定权利要求分类(例如,

权利要求类型)的语言的说明性示例可包括但不限于下表1所提供的示例:

[0030]	表 1	
	权利要求类型	说明性语句
	单项	“如权利要求1所述”
	范围	“如权利要求1-5所述”
	集合	“如权利要求1、2和5所述”
	范围集合(“范围集”)	“如权利要求1、5-10、13以及17-20所述”
	多个前述(“前述”)	“如前述多项权利要求所述”
	单个前述(“前述单个”)	“如前述权利要求所述”
	特定数量的多个前述(“N 个前述”)	“如前述三项权利要求所述”
	前几个(“前N个”)	“如前四项权利要求所述”

[0031] 应当理解,上述参照中仅提供了一些示例,本文所述的其他权利要求类型和示例语句在不背离本公开的范围的情况下也包含在其中。此外,英语以外的语言的使用也被考虑在内。可被用于分类目的的其他语言的说明性示例包括但不限于:保加利亚语、简体中文、繁体中文、捷克语、荷兰语、芬兰语、法语、德语、希腊语、意大利语、日语、韩语、波兰语、标准葡萄牙语、俄语、斯洛伐克语、西班牙语以及瑞典语。应当进一步理解,某些权利要求可以包含分类重叠(即,权利要求的措辞使其可以归为多个权利要求类型)。例如,短语“权利要求1到5”可以被分类为单项(“权利要求1”)和范围(“权利要求1到5”)以用于在恰当分类权利要求时更高的准确度、标识OCR转换错误、和/或保持基础引擎简单且可维护,使得基础引擎无需依赖现有系统中应用的复杂决策树,如本文中将更加详细描述。

[0032] 一般而言,OCR并不完美并且可能会导致无意义的、包含错误和零片段等的电子识别文本。例如,可能发生的OCR错误包括被光学地成像并被识别为小写的L(“1”)的数字“1”。本文中所描述的系统和方法可以通过分析权利要求的各种分类(尤其是重叠分类)并确定不一致性来识别此类错误。这种不一致性可以自动解决或者通过用户查询和/或输入来手动解决。更具体而言,本文中所描述的系统和方法尤其被配置成用于分析作为OCR的结果而生成的一串计算机可读文本,基于根据其他权利要求的引用做出的判断,识别无意义的字符、错误、零片段等,并且使用所成像的文档中出现的原始文本替换此类无意义的字符、错误、零片段等和/或向数据文件(例如相关联的XML文件)附加关于该无意义的字符、错误、零片段等的信息。由此,本文中所描述的系统和方法完全支持更短或更长的替换,以及将内部文本转换到外部文本插入符号位置。一个此类特定的示例可以是使用OCR程序进行了转换的下述文本:“volgens conclusie drie tot vijf”,其经本文所描述的系统和方法识别并且利用以下语句被附加到相关联的数据文件中:“volgens<claim-ref idref=“3 4 5”>conclusie drie tot vijf</claim-ref>”。

[0033] 本文所描述的系统和方法可以进一步用于生成基于对计算机可读文本的校正的

置信度水平,所述计算机可读文本被OCR程序错误地转换和/或由于撰写错误。例如,如果权利要求引用其本身(例如,“2.如权利要求2所述的方法……”)或引用了后续的权利要求(例如,“2.如权利要求8所述的方法”),本文所描述的系统和方法可识别出此类引用是错误的(而无关乎错误是否是由于其被不正确地识别以及OCR转换或者是撰写错误),并且降低该项权利要求和/或与之相关联的权利要求组的置信度。此外,该系统和方法也可用于限定模糊检测匹配,当权利要求和/或权利要求组不符合本文中明确描述的特定规则时降低它们的置信度。

[0034] 现在参考附图,图1描绘了说明性计算网络,该说明性计算网络描绘了根据本文示出和描述的一个或多个实施例的、用于确定专利权利要求中的引用的系统的组件。如图1所示的,计算机网络10可以包括广域网(WAN)(诸如,互联网)、局域网(LAN)、移动通信网络、公共服务电话网络(PSTN)、个人局域网(PAN)、城域网(MAN)、虚拟专用网络(VPN)和/或其他网络。计算机网络10总体上可以配置成用于电子地连接一个或多个计算设备和/或其组件。说明性计算设备可以包括但不限于用户计算设备12a、服务器计算设备12b以及管理员计算设备12c。

[0035] 用户计算设备12a可以总体上用作用户与连接到计算机网络10的其他组件之间的接口。因此,用户计算设备12a可以用于执行一个或多个面向用户的功能,诸如,接收来自用户的一个或多个输入或者将信息提供给用户,如本文中更详细描述的那样。因此,用户计算设备12a可以至少包括显示器和/或输入硬件,如本文中更详细描述的那样。另外,图1中所包括的是管理员计算设备12c。如果服务器计算设备12b需要监督、更新或校正,则管理员计算设备12c可以被配置成用于提供所需的监督、更新和/或校正。管理员计算设备12c还可以用于将附加数据输入到存储在服务器计算设备12b上的语料库中。

[0036] 服务器计算设备12b可以从一个或多个源接收数据、生成数据、存储数据、索引数据、搜索数据和/或将数据以搜索结果、可视表现(诸如权利要求树、文档、来自一个或多个文档的摘录等)的形式提供给用户计算设备12a。

[0037] 应当理解的是,虽然用户计算设备12a和管理员计算设备12c被描绘为个人计算机并且服务器计算设备12b被描绘为服务器,但是这些是非限制性示例。更具体地,在一些实施例中,任何类型的计算设备(例如,移动计算设备、个人计算机、服务器等等)均可以用于这些组件中的任何组件。另外,虽然这些计算设备中的每个计算设备在图1中被图示为单个硬件,但是这仅作为示例。更确切地,用户计算设备12a、服务器计算设备12b和管理员计算设备12c中的每一个可以表示多个计算机、服务器、数据库、组件等等。

[0038] 图2描绘了图1的服务器计算设备12b,进一步示出用于从一个或多个源接收数据、生成数据、索引数据、搜索数据和/或提供数据的系统。另外,根据本文中所示出和所描述的实施例,服务器计算设备12b可以包括用于搜索和提供数据的、被具体化为硬件、软件、和/或固件的非瞬态计算机可读介质。虽然在一些实施例中,服务器计算设备12b可以配置为具有必要硬件、软件、和/或固件的通用计算机,但是在一些实施例中,服务器计算设备12b还可以配置为用于执行本文中所描述的功能而专门设计的专用计算机。在计算设备12b是通用计算机的实施例中,本文所描述的方法总体上通过提升服务器计算设备12b在不为权利要求中出现的每种语言和/或句法进行额外编程的情况下识别权利要求引用的能力,从而提升服务器计算设备12b的功能。

[0039] 同样如图2中所示,服务器计算设备12b可以包括处理器30、输入/输出硬件32、网络接口硬件34、数据存储组件36(其可以存储专利图像数据38a、计算机可读文本数据38b、权利要求引用数据38c、权利要求树数据38d以及其他数据38e)以及非瞬态存储器组件40。存储器组件40可以被配置为易失性和/或非易失性计算机可读介质,并且因此,可以包括随机存取存储器(包括SRAM、DRAM、和/或其他类型的随机存取存储器)、闪存、寄存器、光盘(CD)、数字通用盘(DVD)、和/或其他类型的存储组件。另外,存储器组件40可以配置为存储各种处理逻辑,诸如操作逻辑41、成像逻辑42、OCR逻辑43、引用确定逻辑44和/或权利要求树逻辑45(作为示例,其中的每一个逻辑可以被具体化为计算机程序、固件、或硬件)。本地接口50也包括在图2中并且可以被实现为总线或其他接口以用于促进服务器计算设备12b的组件之间的通信。

[0040] 处理器30可以包括被配置成用于接收和执行指令(诸如来自数据存储组件36和/或存储器组件40)的任何处理组件。输入/输出硬件32可以包括显示器、键盘、鼠标、打印机、相机、麦克风、扬声器、触摸屏和/或用于接收、发送和/或呈现数据的其他设备。网络接口硬件34可以包括任何有线或无线联网硬件,如调制解调器、LAN端口、无线保真(Wi-Fi)卡、WiMax卡、移动通信硬件和/或用于与其他网络和/或设备通信的其他硬件。

[0041] 应当理解的是,数据存储组件36可以驻留在服务器计算设备12b本地和/或远程驻留,并且可以配置成用于存储一条或多条数据并且选择性地提供对所述一条或多条数据的访问。如图2中所示,数据存储组件36可以存储专利图像数据38a、计算机可读文本数据38b、权利要求引用数据38c、权利要求树数据38d和/或其他数据38e,如本文中更加详细地描述的。

[0042] 存储器组件40中包括的是操作逻辑41、成像逻辑42、OCR逻辑43、引用确定逻辑44和/或权利要求树逻辑45。操作逻辑41可以包括用于管理服务器计算设备12b的组件的操作系统和/或其他软件。成像逻辑42可以包括专利文档的一个或多个图像(例如,实体专利文档的扫描页面)。OCR逻辑43可以从专利文档图片中识别文本,并将该文本转换成计算机可读文本(即,从专利文档图像生成计算机可读文本)。引用确定逻辑44可以使用计算机可读文本为专利文档内的一个或多个权利要求中的每一个确定一个或多个权利要求引用,如下文详细描述。权利要求树逻辑45可以生成基于所确定的权利要求引用的权利要求树,如本文更加详细描述的。

[0043] 应当理解,图2中所示出的组件仅是说明性的并且不旨在限制本公开的范围。更具体地,虽然图2中的组件被图示为驻留在服务器计算设备12b内,但是这是非限制性示例。在一些实施例中,这些组件中的一个或多个组件可以驻留在服务器计算设备12b外部。类似地,虽然图2涉及服务器计算设备12b,但是其他组件(诸如用户计算设备12a和管理员计算设备12c)可以包括相似的硬件、软件和/或固件。

[0044] 本文中所描述的各种过程可以总体上需要专利文档中的专利权利要求的识别。因此,专利文档被分析以用于确定其中的专利权利要求。图3中描绘了用于分析计算机可读文本以确定专利权利要求的说明性方法。除了确定计算机可读文本中是否包含专利权利要求之外,参照图3所描述的方法还可以用于为计算机可读文本中的专利权利要求中的每一个确定权利要求编号以及一个或多个引用。

[0045] 在一些实施例中,为了获得计算机可读文本,可以通过OCR程序将专利文档成像

和/或转换为计算机可读文本。此类过程应总体上理解并且在本文中不再详细描述。图2中描绘的服务器计算设备12b的各个部分可以被用于获得计算机可读文本,这些部分包括但不限于成像逻辑42和/或OCR逻辑43。此外,数据可能在数据存储组件36的各个部分中访问和/或存储,这些部分包括但不限于专利图像数据38a和或计算机可读文本数据38b。

[0046] 更具体而言,在获得与专利权利要求的一个或多个图像的相对应的数据的实施例中,此类数据可以是例如从专利图像数据38a和/或从诸如扫描设备、相机等之类的成像设备获得的。说明性数据可以包括图像文件,诸如TIFF图像文件、JPG/JPEG图像文件、GIF图像文件、BMP图像文件和PDF图像文件等。一个或多个图像可以使用OCR程序或模块(诸如目前已知或后续开发的各种OCR程序或模块)被进一步转换成为计算机可读文本。生成的计算机可读文本可以作为计算机可读文本数据38b(图2)存储在数据存储组件36中,并且可以进一步被分析以用于确定专利权利要求,如本文中更为详细描述的。

[0047] 在获取到包含与专利要求相对应的计算机可读文本的数据的实施例中,此类数据可以例如从计算机可读文本数据38b(图2)和/或来自本地和/或远程数据库的数据中获取。计算机可读文本可被分析以确定专利权利要求,如本文更为详细描述的。

[0048] 仍参考图3,权利要求列表可以在步骤305确定。权利要求列表通常是专利文档的一个章节,包含了专利的权利要求。每个权利要求均附有标识该权利要求的编号。因此,权利要求列表可以是经编号的列表,其中列表中的每一个权利要求都具有不同于列表中的其他权利要求的编号。权利要求列表中的权利要求的编号可以是大于零的任何数字。也就是说,权利要求列表可以具有1、2、3、4、5、6、7、8、9、10或是编号更大的权利要求。在一些实施例中,权利要求的编号可以由专利文档来源地的管辖方(例如,对权利要求编号的法规限制)、出于经济考虑(例如,权利要求由于施加的超权费而受到限制)等等来设定。

[0049] 权利要求列表的确定可以还包括分析文档文本的数字符号,诸如,例如欧洲/西方阿拉伯自然数字(1、2、3等)以及罗马数字(I、II、III等)。阿拉伯数字(١、٢、٣等)、汉语大写数字(壹、贰/貳、叁/叁等)以及韩语数字(일、이、삼等)等等,特别是排列成有序列表的数字符号。然而,由于专利文档可能仅包含单个专利权利要求,专利文档可能用于其他目的而包含数字符号列表(例如,说明书可能涉及有序的列表,或可能包含编号的段落)等等,这种权利要求列表的确定可以考虑指示所发现的数字符号是否确实是专利权利要求的其他因素。在非限制性示例中,该确定可以进一步包括确定数字符号之前是否存在特定的符号(例如,句号、逗号、冒号和括号等,以及这些符号的组合)。在另一非限制性示例中,该确定可以进一步包括确定是否在单词“权利要求”(或其翻译)之前的特定数量的字符或单词内存在数字符号。即,可以确定词语“权利要求”(或其翻译)是否出现在数字符号之后的10个字符、15个字符或20个字符等之内。在又另一非限制性示例中,该确定可以进一步包括确定(多个)数字符号之前是否存在权利要求章节的特定指示符。即,该确定可以包括查找紧接着第一个数字符号之前的文本和/或其他文档特征并且确定此类文本或文档字符是否指示权利要求章节。例如,说明性文本可以包括但不限于:“权利要求”、“权利要求列表”、“对权利要求的修改”、“要求保护的是”以及“本人主张”等等。在另一示例中,说明性文档特征可以包括但不限于:与文档剩余部分所用的字体不同的字体格式、与文档剩余部分所用的段落格式不同的段落格式、分页符以及分节符等等。

[0050] 权利要求列表的确定可以进一步包括确定特定的权利要求列表的起始和结束位

置。即,此类确定可以分析文本并且放置指示权利要求列表中的第一项权利要求(例如,有序列表的第一项权利要求以及带有诸如1等等之类的首个编号标识符的权利要求等)的起始的记号;并且放置指示列表中的最后一项权利要求的结束的记号。指示列表中的最后一项权利要求的结束的记号通常放在权利要求列表中的权利要求的最后一个指示符(例如,句号等)处或其之后,其可以是有序的权利要求列表中的最后一个权利要求之后、与权利要求中存在的最大编号相关联的最后一个权利要求之后等。此类指示符并不受本公开的限制,通常可以包括对应数据文件(例如XML文件)中的代码(诸如元数据标签等),该代码可以由计算设备用于识别权利要求列表的起始和/或结束。在一些实施例中,指示符可以由机构作为书签等来提供。

[0051] 在步骤307处,可以确定权利要求列表中的权利要求的语言。也就是说,在一些实施例中,可以在文档元数据中提供权利要求列表中的权利要求的语言的指示符,并且因此可以通过分析元数据来进行确定。在其他实施例中,权利要求的文本可以被分析以用于确定权利要求所使用的字符的类型(例如,拉丁字母、西里尔字母、简体中文字符和繁体中文字符等)、基于权利要求中所使用的词语确定语言(例如,识别出词语“aparato”是“装置”的西班牙语,这可能指示权利要求是以西班牙语撰写的)、确定仅存在于特定语言中的句子结构等。其他识别权利要求语言的方法也可以在不脱离本公开范围的情况下使用。

[0052] 在步骤310处,可以确定权利要求列表中的权利要求。即,可以确定权利要求的数量(包括它们的编号)以及权利要求的顺序。此类确定通常可以通过以下步骤进行:分析每一个权利要求的文本、确定与该权利要求相关联的数字(例如,权利要求列表中的第一项权利要求可以与数字1相关联,权利要求列表中的第二个权利要求可以与数字2相关联,等等)以及确定每一个权利要求起始和结束的位置。此类确定可以进一步包括放置指示权利要求列表中每一个权利要求的起始和结束的记号。指示权利要求列表中每一个权利要求的起始的记号通常可以被放置在权利要求的第一个指示符(例如数字指示符)之前或该第一个指示符处,并且指示权利要求列表中每一个权利要求的结束的记号通常可以被放置在同一个权利要求的最后一个指示符(例如句号等)处或其之后。此类指示符并不受本公开的限制,通常可以包括对应数据文件(例如XML文件)中的代码(诸如元数据标签等),该代码可以由计算设备用于识别权利要求列表中每一个权利要求的起始和/或结束。

[0053] 在步骤315处,可以确定每一个权利要求是否为列表中的第一项权利要求。该确定可以通过以下步骤进行:确定与权利要求相关联的编号、确定权利要求是否引用了另一个权利要求、和/或确定权利要求是否紧邻指示权利要求列表的起始的记号。通常来讲,列表中的第一项权利要求将与数字1或其他类似字符相关联、可以与列表中的所有权利要求中最小的编号相关联、和/或可以不包含对其他权利要求的任何引用(即,由于该权利要求是独立权利要求)并且可以包含标识权利要求组和/或第一项权利要求的起始的标签(例如元数据标签)权利要求组。由于只有一项权利要求能够是列表中的第一项权利要求,因此一旦确定了第一项权利要求,这种确定就可以结束。

[0054] 如果权利要求是列表中的第一项权利要求,其将会在步骤320处被标识为独立权利要求。这是由于有序的专利权利要求列表中的第一项权利要求始终是独立权利要求。标记权利要求可能包括,例如,生成指示该权利要求为独立权利要求的数据,诸如创建被附加到权利要求列表的文本中的元数据标签等。除了标记权利要求之外,可以在步骤325处生成

或附加数据文件。此类数据文件通常指示权利要求、与该权利要求相关联的编号以及其作为独立权利要求的状态。可以之后访问该数据文件以用于创建权利要求树，如本文中详细描述。在一些实施例中，数据文件可以是用于权利要求的新的或已有的XML文件，其中XML文件被修改/附加以便包含标记信息。过程可以前进至步骤355，如本文中更为详细描述的。

[0055] 对于被确定为并非列表中的第一项权利要求的每一项权利要求，过程可以前进至步骤330，以查找位于权利要求中的至少一个引用（例如，其他编号）。即，权利要求的文本可被分析以用于确定其是否包含对于另一权利要求的引用，并且如果包含，则确定其引用了哪些权利要求。此类查找通常查找指示权利要求引用的词语、短语等，诸如但不限于：“权利要求1中的”、“如权利要求1所述”、“如第一项权利要求所述”、“如前述权利要求中的任一项所述”和“如权利要求1-5所述”等等。应注意，仅在其文本内包含数字的权利要求在没有指示引用的附加文本的情况下不一定表示对另一权利要求的引用。这是由于某些权利要求可能会在不包含权利要求编号的情况下引用其他权利要求（例如，“如前述权利要求中的任一项所述”）或可能包含某些不是权利要求引用的数字（例如，描述为短语“大约1mm”的权利要求，其中的“1”代表长度而不是对权利要求1的引用）。下表2中示出了某些权利要求引用的特定示例：

表 2	
权利要求语句	引用
“16. According to claim 1,3 to 5, and 13 to 15, wherein said...”(英语)	权利要求16的权利要求引用为权利要求1、3、4、5、13、14和15。
[0056] “10.제 1 항 또는 제 3 항, 제 6 항 또는 제 7 항 중 어느 한 항에 있어서” (韩语)	权利要求10的权利要求引用为权利要求1、3、6和7。
“5.Οι παραπάνω αξιώσεις (1,2,3 & 4) μπορούν” (希腊语)	权利要求5的权利要求引用为权利要求1、2、3和4。

[0057] 在步骤335中，如果权利要求中不存在其他引用，则该权利要求可以在步骤320中被标记为独立权利要求并且数据文件可以在步骤325中被生成/附加，如本文之前所描述的。该过程可前进至步骤355，如本文中更为详细描述的。

[0058] 如果权利要求中存在其他引用，则进一步的验证步骤可被完成以用于确保该引用是实际上恰当的引用（即，引用了有序的权利要求列表中较早出现的权利要求），由此进一步指示该权利要求是独立权利要求或是从属权利要求。即，权利要求不能引用自身或是引用在有序列表中后续出现的权利要求。由此，在步骤340处确定引用（例如引用中的数字）是否小于该权利要求的编号。即，该确定可以包括确定引用中的数字、确定当前权利要求的编号、以及比较相应编号以便确保引用中的数字小于当前权利要求的编号。例如，如果特定权利要求为权利要求4，则该权利要求可以恰当地引用权利要求1、2和/或权利要求3（包括诸如“前述权利要求中的任一项”等之类的语句）。权利要求4不应恰当地引用权利要求5、权利要求6、权利要求7等等。

[0059] 如果权利要求不包含小于该权利要求编号的引用，则该权利要求将会在步骤320

中被标记为独立权利要求,数据文件被生成/附加,并且过程会前进至步骤355,如本文中更为详细描述。所设想的是,权利要求中可能不包含由于OCR转换、排印错误或其他类似问题造成的小于该权利要求编号的引用。在此类实例中,该权利要求可能包含与(多个)错误引用相关的指示符等,使得该权利要求在权利要求的进一步分析中被重新标记为从属权利要求,如本文中更加详细描述的。

[0060] 如果该权利要求中确实包含小于该权利要求的编号的引用,那么该权利要求可以在步骤345中被标记为从属权利要求。标记权利要求可以包括例如生成指示该权利要求为从属权利要求的数据,诸如创建附加到权利要求列表的文本中的元数据标签等。除了标记权利要求之外,可以在步骤350中生成数据文件或对数据文件进行附加。此类数据文件通常指示权利要求、与权利要求相关联的编号以及其作为独立权利要求的状态。在一些实施例中,数据文件可以是用于权利要求的已有的XML文件,其中已有的XML文件通过标记信息来更新。例如,短语“according to claim 1the”可以按照下述方法标记:“according to<i>clai<u>ml</u></i>the”,在之后步骤350中,则变成了“according to<i><claim-ref idref=“1”>clai<u>ml</u></claim-ref></i>the”。可以之后访问数据文件以用于创建权利要求树,如本文中详细描述的。然后,过程可以前进到步骤355。

[0061] 为了确保所有权利要求都如本文关于图3所描述的那样进行分析,可以在步骤355中确定权利要求列表中是否存在额外的权利要求。如果确实存在,则可以在步骤330处重复该过程。如果不存在额外的权利要求,则可以在步骤360处确定是否存在另一权利要求列表。例如,某些专利文档可能包含多个权利要求列表(例如,专利文档中可能有包含了一组权利要求组的多个翻译版本中的每一版的权利要求列表;多个权利要求列表,其中每个权利要求列表均采用了特定的措辞或以其他方式格式化来符合特定管辖区的要求;以及权利要求组一组权利要求的每一个版本的权利要求列表,诸如最初提交的权利要求、经修改的权利要求、授权的权利要求以及颁布的权利要求)。即,专利文档可能包括例如针对美国而特别措辞/格式化的第一权利要求列表并且包括3个独立权利要求和17个从属权利要求,这些权利要求都不是多项从属权利要求;以及针对欧洲而特别措辞/格式化的第二权利要求列表并且包括一个独立权利要求和14个从属权利要求,这些权利要求中一些是多项从属权利要求。

[0062] 如果仅存在单个权利要求列表,则该过程可以结束。反之,如果存在多个权利要求列表,则该过程可以返回到步骤310以针对多个权利要求列表中的每个后续权利要求列表来完成。

[0063] 图4描绘了用于确定专利权利要求中的引用的说明性方法的框图。在步骤405中,可以获取特定文档中所有需要分析的专利权利要求。也就是说,可以获取每一个专利权利要求或者可以获取专利权利要求的子集。获取专利权利要求可以包括例如获取包括与专利权利要求相对应的计算机可读文本以及相应的数据文件等的的数据。

[0064] 在权利要求包含多个权利要求组的实施例中,可以在步骤407中选择特定的权利要求组。也就是说,权利要求组权利要求组可以从多个权利要求组中选择第一权利要求组,并且可以针对该权利要求组完成下文关于图4所描述的后续过程。该过程可随后针对每一个后续选择的权利要求组重复。

[0065] 在步骤408中可以确定权利要求列表的语言。也就是说,在一些实施例中,可以在

文档元数据中提供权利要求列表中的权利要求的语言的指示符,并且因此可以通过分析元数据来进行确定。在其他实施例中,权利要求的文本可以被分析以用于确定权利要求所使用的字符的类型(例如,拉丁字母、西里尔字母、简体中文字符和繁体中文字符等)、基于权利要求中所使用的词语确定语言(例如,识别出词语“aparato”是“装置”的西班牙语,这可能指示权利要求是以西班牙语撰写的)、确定仅存在于特定语言中的句子结构等。其他识别权利要求语言的方法也可以在不脱离本公开范围的情况下使用。

[0066] 在步骤410中,可以从多组权利要求中选择权利要求以用于分析。在一些实施例中,对特定权利要求的选择可以包括从有序权利要求列表中选择第一项权利要求或者选择要分析的下一项权利要求(如果先前的权利要求已被分析)。

[0067] 可以在步骤415中确定权利要求编号,并在步骤420中存储权利要求编号。确定权利要求编号通常可以包括分析文本以找到指示权利要求开始的记号和/或分析相应的数据文件(如上文所述)并确定与该权利要求相关联的权利要求编号。例如,如果权利要求记号紧接在数字3之前,就可以确定当前分析的权利要求为权利要求编号3。也如本文之前所述,权利要求编号可以被存储为数据文件。数据文件可以存储在例如数据存储组件36的各个部分中(诸如计算机可读文本数据38b、其他数据38e等)。步骤420可以被编码并且算法格式为“Store CurrentClaimNumber”。

[0068] 在步骤425中,可以从每一项权利要求中确定小于该权利要求便号的唯一引用。也就是说,可以检索和分析与该权利要求有关的数据(例如,根据图3的步骤325或350创建或附加的用于权利要求的数据文件)以确定该权利要求是否包含小于该权利要求编号的引用。例如,如果所确定的权利要求编号是5,则它可以具有1、2、3和/或4的唯一引用,其分别对应于权利要求1-4。步骤425可以以算法形式编码为“Calculate ClaimReferences”。

[0069] 在步骤430中,可以确定该权利要求的所有先行。特定权利要求的先行通常是由该特定权利要求直接或间接引用的权利要求。例如,如果一组权利要求包括权利要求1-5并且权利要求2引用权利要求1(例如,“2.如权利要求1所述的方法……”)并且权利要求5引用权利要求2(例如,“5.如权利要求2所述的方法”,那么权利要求5的先行是权利要求1和2,因为它们都被权利要求5引用(权利要求2被直接引用并且权利要求1被间接引用)。下表3提供了权利要求引用和先行的另一个说明性示例:

表 3		
权利要求	引用	先行
1		
2	1	1
3	2	1, 2
4	1, 2, 3	1, 2, 3
5	4	1, 2, 3, 4

[0071] 根据步骤430确定权利要求的先行还可以包括存储与该权利要求及其先行有关的信息(例如,先行数据)。例如,如本文中更详细描述,可以生成或附加相关联的数据文件(例如,XML文件)以记录权利要求和相关联的引用。这样的数据文件可以存储为例如权利要

求引用数据38c(图2)。步骤430可以以算法格式编码为“Calculate ClaimAncestors”。在步骤435中,可以进一步生成或附加相应的数据文件,以将权利要求的自我引用(例如,自我引用数据)添加到先行信息。这样,数据文件可以标识该权利要求以及该权利要求的所有相关联的先行。步骤435可以以算法格式编码为“Calculate ClaimAncestorsAndSelf”。

[0072] 在步骤440中,可以确定权利要求的父权利要求。权利要求的父权利要求通常指的是该权利要求的一项或多项直接先行。也就是说,参照表2,权利要求5的父权利要求是权利要求4,权利要求4的父权利要求是权利要求1、2或3,权利要求3的父权利要求是权利要求2,权利要求2的父权利要求是权利要求1。权利要求1不包含父权利要求,因为它是独立权利要求并且没有先行。由此,所有父权利要求均为先行权利要求,但所有先行权利要求都不一定是父权利要求。作为根据步骤440确定父权利要求的一部分,可以按降序创建权利要求引用的临时列表,使得能用权利要求引用对父权利要求进行交叉检查。也就是说,可以访问临时权利要求引用中的每一项权利要求以验证它是否实际上是特定权利要求的父权利要求列表的成员,从而优化每个权利要求的父权利要求列表。如果权利要求实际上是另一个权利要求的真正父权利要求,则可以从先行权利要求的列表中删除该权利要求,使得每项权利要求具有两个不同的列表:一个列表包含不是父权利要求的先行权利要求,第二个列表包含父权利要求。这种交叉检查也可用于验证和纠正OCR转换错误。步骤440可以以算法格式编码为“Calculate ClaimParents”。

[0073] 在步骤445中,对于由特定权利要求(但不是父权利要求)引用的每一项唯一权利要求,可以为该唯一权利要求添加或附加数据文件以添加该特定权利要求的权利要求编号(例如,插入指示关系的交叉引用数据)。也就是说,如果一组权利要求包含权利要求1、2和3,其中权利要求3引用权利要求2并且权利要求2引用权利要求1,则权利要求1的数据文件可以被添加或附加以包括与权利要求3有关的元数据。然而,在该步骤中将不添加或附加权利要求2的数据文件,因为权利要求2是权利要求3的父权利要求。步骤445可以以算法形式编码为“Initialize ClaimReferencedBy empty and Update ClaimReferencedBy”。

[0074] 类似地,对于由特定权利要求引用的每一项父权利要求,可以在步骤450中为该父权利要求添加或附加数据文件以添加该特定权利要求的权利要求编号(例如,插入指示该关系的交叉引用数据)。也就是说,如果一组权利要求包含权利要求1、2和3,其中权利要求3引用权利要求2并且权利要求2引用权利要求1,则权利要求1的数据文件可以被添加或附加以包括与权利要求2有关的元数据,并且权利要求2的数据文件可以被添加或附加以包括与权利要求3有关的元数据。步骤450可以以算法形式编码为“Initialize ClaimChildren empty and Update ClaimChildren”。

[0075] 在步骤455中,对于特定权利要求的每一项先行权利要求(包括父权利要求),可以为该先行权利要求添加或附加数据文件,以将该特定权利要求的权利要求编号添加到用于对该先行权利要求的所有后代的引用的字段,如上文所描述。也就是说,如果一组权利要求包含权利要求1、2和3,其中权利要求3引用权利要求2,权利要求2引用权利要求1,则权利要求1的数据文件可以被添加或附加以包括与权利要求2和3有关的元数据作为其先行。步骤460可以以算法形式编码为“Initialize ClaimDescendants empty and Update ClaimDescendants”。

[0076] 在步骤460,对于特定权利要求的每一项先行权利要求(包括父权利要求),为该先

行权利要求添加或附加数据文件,以将该特定权利要求的权利要求编号以及该先行权利要求自身的权利要求编号加入到用于对该先行权利要求的所有后代的引用的字段中,如本文先前所描述。也就是说,如果一组权利要求包含权利要求1、2和3,其中权利要求3引用权利要求2并且权利要求2引用权利要求1,则权利要求1的数据文件可以被添加或附加以包括与权利要求2和3有关的元数据作为其先行,并包括与权利要求1有关的元数据作为自我引用。步骤460可以以算法形式编码为“Initialize ClaimDescendantsAndSelf with CurrentClaimNumber and Update ClaimDescendantsAndSelf”。

[0077] 关于图4描述的过程可以针对权利要求组中的每一项权利要求重复,以确保它们被利用元数据恰当地标记和/或恰当地创建/附加引用文件。这样,在步骤465中,可以确定是否存在下一项权利要求。如果存在,则过程可以返回到步骤410。反之,可以针对该权利要求组结束该过程。

[0078] 然后,可以使用从本文描述的权利要求生成的所有信息来创建精简权利要求树,该精简权利要求树允许用户以更精简的方式可视化权利要求之间的关系。图5A描绘了典型的权利要求树。独立权利要求1、11、20、21和25示于图5A的顶部,并且从属权利要求2-35示于下方。权利要求之间的线指示引用。例如,权利要求14、16和24均从属于权利要求20,因此权利要求20和权利要求14、16和24的每一相之间均画有一条线。然而,大量的权利要求从属/引用导致大量的线,使得难以可视化权利要求之间的相互关联。例如,权利要求之间的某些线与其他线重叠和/或在权利要求编号后面穿过,因此可能难以辨别特定线在何处开始和结束(例如,权利要求1和19之间的线在权利要求13后穿过并与其他几条线重叠)。为此,本文描述的系统和方法可以提供如图5B所示的精简权利要求树,其使得更容易可视化权利要求之间的关系。可以使用如本文关于图1-图4所描述那样生成和存储的数据来生成精简权利要求树,并且对于允许/准许对多项固有权利要求的引用的权利要求可以是有用的。

[0079] 图6是根据实施例的精简权利要求树的过程的示意图。如表600所示,示例权利要求组具有7项权利要求,其中权利要求1为独立权利要求,权利要求2-7为从属权利要求。权利要求2和3仅引用了权利要求1(例如,“如权利要求1所述的方法”),权利要求4引用了权利要求2和3(例如,“如权利要求2或3所述的方法”),权利要求5引用了权利要求1和4(例如,“如权利要求1和4所述的方法”),权利要求6引用了权利要求3和4(例如,“如权利要求3或4所述的方法”),权利要求7引用了权利要求2-6(例如,“如前述权利要求中任一项所述的方法”)。因为权利要求1是独立权利要求,所以“ClaimReferences”字段、“ClaimParents”字段和“ClaimAncestors”字段是空白的;权利要求1被提供在“ClaimAncestorsAndSelf”字段中,因为权利要求1没有任何先行权利要求;权利要求2、3、5和7被提供在“ClaimReferencedBy”字段中,因为那些权利要求中的每一项都直接引用权利要求1(即,权利要求4和6,虽然通过从属于其他从属于权利要求1的权利要求而间接从属于权利要求1,但并非直接引用权利要求1);权利要求2和3被提供在“ClaimChildren”字段中,因为权利要求2和3是权利要求1的直接后代(即,虽然权利要求7引用权利要求1,但它不是子权利要求,因为它也引用了权利要求2和3);权利要求2-7被提供在“Claim Descendants”中,因为权利要求2-7中的每一项均直接或间接地从属于权利要求1;并且权利要求1-7被提供在“ClaimDescendantsAndSelf”字段中,因为这样的字段包括权利要求1的所有后代和权利要

求1本身。

[0080] 为了从标准权利要求树605生成精简权利要求树610,本文描述的系统和方法可以使用各种字段来确定哪些权利要求被直接引用(例如,“ClaimReferences”),哪些权利要求被间接地引用(例如,“ClaimAncestors”-“ClaimReferences”),哪些权利要求直接引用所选权利要求(例如,“ClaimReferencedBy”),和/或哪些权利要求间接引用所选权利要求(例如,“ClaimDescendants”-“ClaimReferencedBy”)。结果,精简权利要求树移除了权利要求的非父先行权利要求之间的线,替代地利用两条或更多线(例如,权利要求与父权利要求之间的第一条线以及父权利要求与祖父权利要求之间的第二条线)指示这样的非父先行权利要求与该权利要求之间的连接。

[0081] 表4提供了与上述图表600类似的另一说明性权利要求图表:

[0082]

表4				
<u>CurrentClaimNumber</u>	<u>ClaimReferences</u>	<u>ClaimAncestors</u>	<u>ClaimReferencedBy</u>	<u>ClaimDescendants</u>
1			2	2, 3, 4, 5
2	1	1	3	3, 4, 5
3	2	1, 2	4	4, 5
4	3	1, 2, 3	5	5
5	4	1, 2, 3, 4		

[0083] 根据表4中的权利要求图表,权利要求3直接引用权利要求2,间接引用权利要求1,被权利要求4直接引用,并且被权利要求5间接引用。

[0084] 现在参考图7,生成精简权利要求树的过程包括确定每个权利要求指示符在精简权利要求树中的定位。出于解释图1-图10的目的,术语“权利要求”可以指精简权利要求树中的实际专利权利要求或权利要求指示符(例如,具有围绕它的圆圈的权利要求编号等)。

[0085] 在步骤705中,确定权利要求的纵向顺序(“VerticalOrder”)。权利要求的纵向顺序通常是指权利要求在纵向意义上的定位。例如,独立权利要求可以具有如下纵向顺序,使得该权利要求的标识符出现在权利要求树的顶部,并且其从属权利要求的任何标识符都出现在该独立权利要求的标识符之下。应当理解,独立权利要求的标识符位于权利要求树的顶部仅仅是说明性的,并且可以在另一个位置,这取决于权利要求树的方向。例如,独立权利要求的标识符可以位于权利要求树的一侧、权利要求树的底部或另一个位置。权利要求的纵向定位(即,“VerticalPosition”)并不被创建精简权利要求树的目的而需要,因为每个垂直平面是以固定偏移绘制的,如本文更详细描述。在步骤710中,可以用对应于权利要求的纵向顺序的数据生成或附加到精简权利要求树的引用文件(例如,XML文件等)。

[0086] 除了权利要求的纵向顺序之外,可以在步骤715中确定权利要求的横向顺序(“HorizontalOrder”),并且可以在步骤720中向引用文件附加相应的横向顺序数据。横向顺序通常是指在同一纵向顺序平面上水平绘制权利要求所采用的顺序。横向顺序可以允许重新排序权利要求的能力,以便最小化连接线之间的重叠次数。

[0087] 在步骤725中,可以确定权利要求的横向位置(“HorizontalPosition”),并且可以在步骤730中向引用文件附加相应的横向位置数据并存储。横向位置通常是指权利要求将被放置在精简权利要求树中时所在的绝对横向位置。

[0088] 在步骤735中,可以确定权利要求的有序父权利要求(“OrderedParents”),并且可以在步骤740中向引用文件附加相应的有序父权利要求并存储。权利要求的有序父权利要求指的是权利要求的父权利要求(如果有的话),其中,这些父权利要求将被定位成从该权利要求向上一个纵向顺序,并按其各自的横向顺序排序。

[0089] 如果权利要求具有父权利要求,则可以在步骤745中确定该权利要求的最右父权利要求(“RightMostParent”)。另外,可以在步骤750中向引用文件附加相应的最右父权利要求数据并存储。最右父权利要求通常指的是有序父权利要求中的最后一项。因此,最右父权利要求通常对应的是具有最大权利要求编号的父权利要求。在权利要求仅具有单个父权利要求的实施例中,最右父权利要求将始终是该单个权利要求。

[0090] 如果权利要求有子权利要求,则可以在步骤755中完成对该权利要求的有序子权利要求(“OrderedChildren”)的确定。另外,可以在步骤760中向引用文件附加相应的有序子权利要求数据并存储。权利要求的有序子权利要求通常是指权利要求的、驻留在从该权利要求向下一个纵向顺序处、并且按照横向顺序被进一步排序的子权利要求。

[0091] 为了确定所有权利要求的排序,本文参考图7描述的过程可以针对每一项权利要求重复。这样,在步骤765中确定是否有额外的权利要求需要排序。如果是,则过程返回到步骤705。否则,该过程结束。

[0092] 从图7的各个步骤生成的结果数据文件提供了生成图5B中描绘的精简权利要求树所必需的信息。也就是说,数据文件包含:每一项权利要求的信息,其指示该权利要求相对于精简权利要求树中的其他权利要求的位置;以及权利要求树的权利要求之间的互连线。这样,可以根据图7中描绘的步骤从所生成和附加的数据文件生成精简权利要求树。

[0093] 每一项权利要求的横向顺序和纵向顺序可以根据图8、图9和图10中描绘的流程图来确定。如本文将更详细描述,每一项权利要求的横向顺序设置为0到N,其中N对应于权利要求组中的最高权利要求。具有零引用的权利要求的纵向顺序被设置为零。否则,该纵向顺序被设为-1。

[0094] 参考图8,为了计算权利要求的纵向顺序,将深度变量设置为0。然后在步骤805中确定特定权利要求是否具有等于该深度的纵向顺序。如果是,则在步骤810中收集该权利要求的所有不同子权利要求。在步骤815中,确定是否已找到任何子权利要求(例如,至少1项子权利要求)。如果没有找到,则过程结束。否则,在步骤820中使深度增加1。然后在步骤825中将每个所收集的子权利要求中的每一相的纵向顺序设置为新的深度(例如,增加了1的深度),并且过程可以返回到步骤805以便重复。

[0095] 为了计算权利要求的横向顺序,采用两个相邻的纵向有序平面,并且确定权利要求需要采用哪个顺序以最小化线交叉。也就是说,权利要求以最小化将在权利要求之间延伸的线上发生的线交叉量(例如,线重叠)的任何方式重新排列。例如,如图9所示,权利要求4和5的横向顺序以如下方式重新排列,使得在权利要求2和5之间延伸的线不跨越在权利要求3和4之间延伸的线。当引用线被移除时,关于直接引用的重要和/或关键信息可能会丢失。因此,当选择树中的权利要求圆(例如,在树的该部分上移动鼠标光标、突出显示树的该

部分等)时,通过在父和子方向上使用直接和间接从属关系的颜色编码(等)来与权利要求树的交互以揭示丢失的信息可能是必要的。过滤和突出显示技术也可以应用于权利要求文本本身。例如,可以使用户能够选择权利要求,这将视图转换为在一个或多个组中示出直接引用、间接引用、被直接引用、被间接引用和不相关的权利要求。例如,这可以提供权利要求律师的有用信息,并且可以节省分析权利要求之间重要关系的时间。

[0096] 参照图10,为了计算权利要求的横向顺序,在步骤1005中确定权利要求的有序父权利要求,在步骤1010确定权利要求的最右父权利要求,在步骤1015确定索赔的有序子权利要求,并且在步骤1020中将横向位置初始设置为0。为了确保对所有权利要求完成权利要求的横向顺序,可以通过在步骤1025中确定是否存在额外的权利要求来对每个权利要求重复该过程,如果存在,则返回到步骤1005。一旦对于权利要求组中的所有权利要求完成该过程,该过程前进到步骤1030。

[0097] 在步骤1030中,为特定权利要求设置输入参数,其对应于权利要求编号(“InputClaimItem”)。在步骤1035中,将权利要求的子权利要求列表定义为其有序子权利要求(“OrderedChildren”)。在步骤1040中,将本地树(“LocalTree”)定义为权利要求项的列表,并且在步骤1045中,遍历子项中的每个权利要求项。如果权利要求的最右父权利要求等于该输入参数,则该过程可以用输入参数权利要求编号重复步骤1030并且捕获权利要求编号的列表中的返回值。可以在返回的列表的开始处插入权利要求编号,并且可以在权利要求的纵向顺序相同的列表和本地树之间发现任何相邻权利要求,并且列表的横向顺序等于本地树的水平顺序加上一。

[0098] 现在应该理解,本文描述的系统和方法可以准确地确定专利文献中的权利要求,确定每一项权利要求中的任何引用,并提供精简权利要求树,该精简权利要求树可以提供使权利要求之间的关系可视化的更容易的手段。

[0099] 尽管本文示出和描述了特定实施例,但应理解可作出其他变化和改型而不偏离所要求保护主题的精神和范围。另外,虽然本文描述了所要求保护主题的各种方面,但不需要以组合的方式来使用这些方面。因此,所附的权利要求旨在覆盖在所要求保护主题范围内的全部变化和修改。

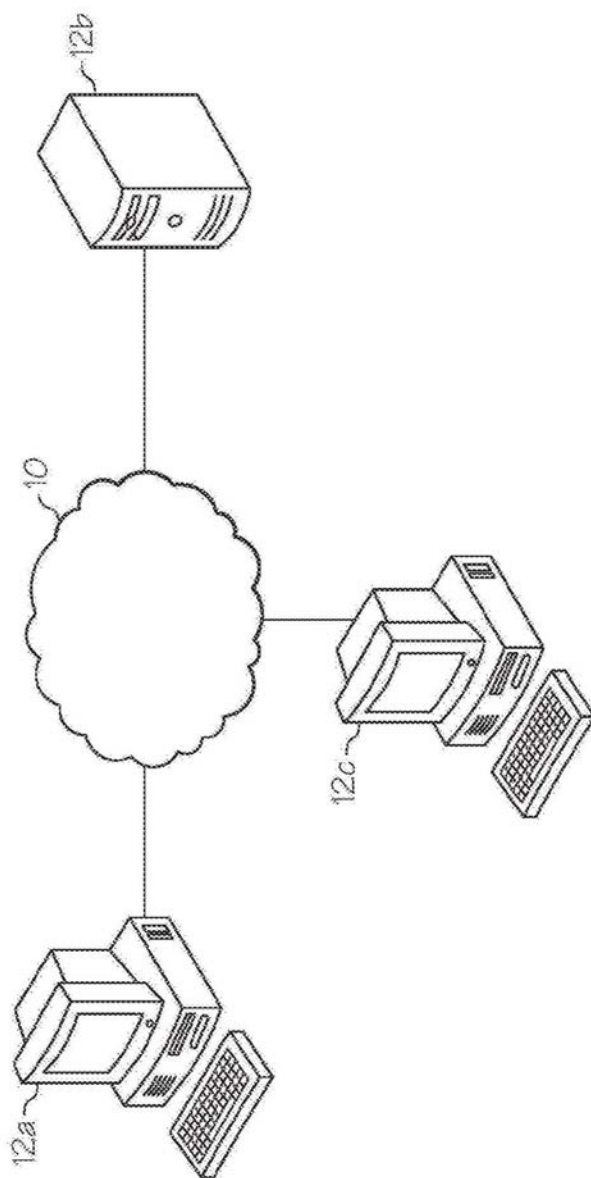


图1

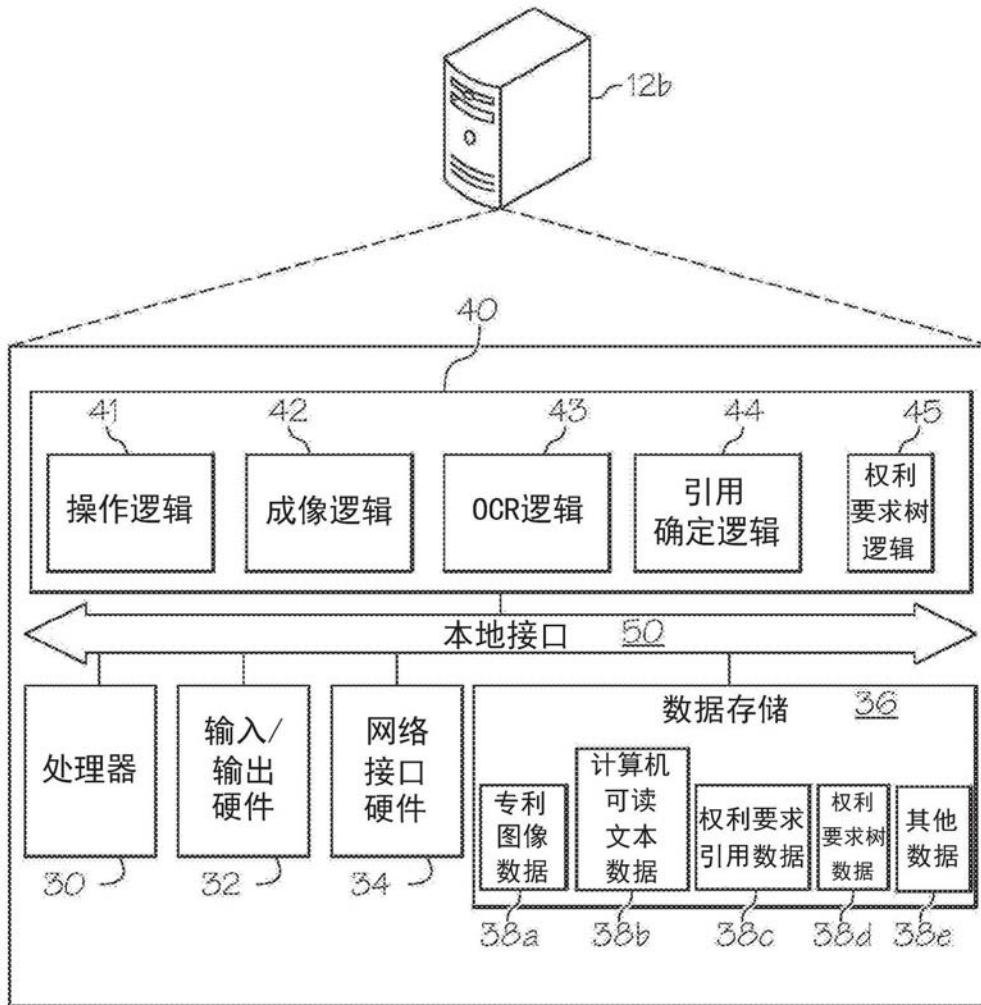


图2

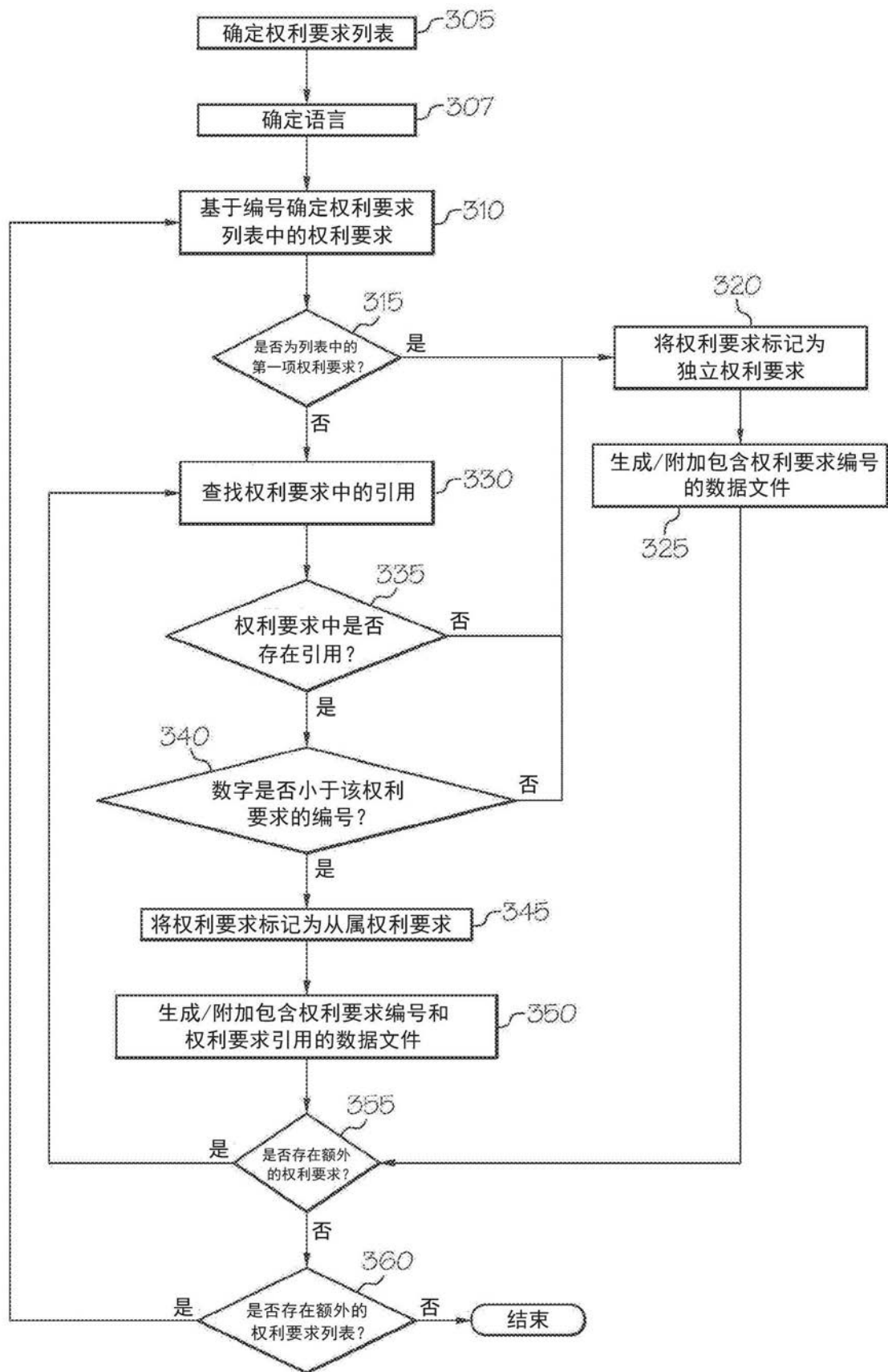


图3

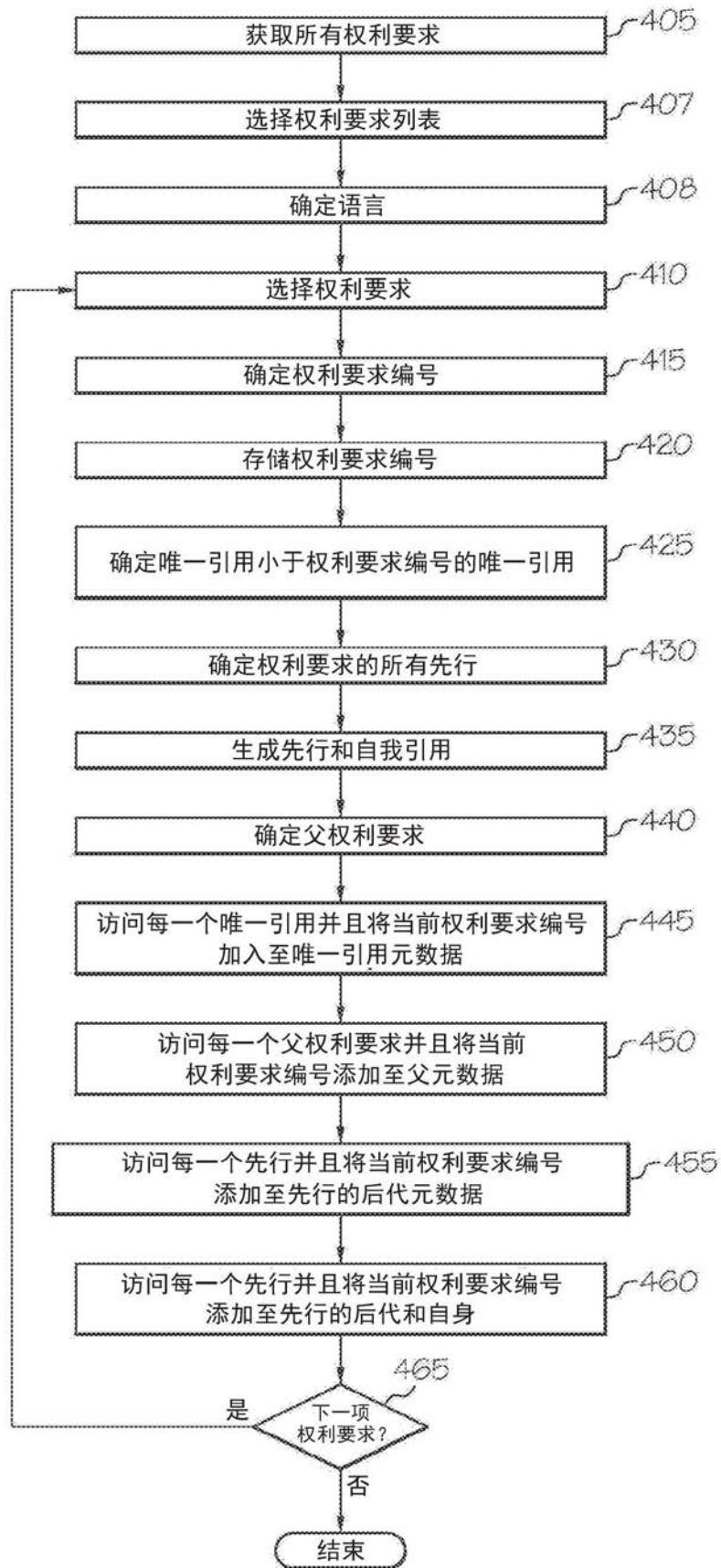


图4

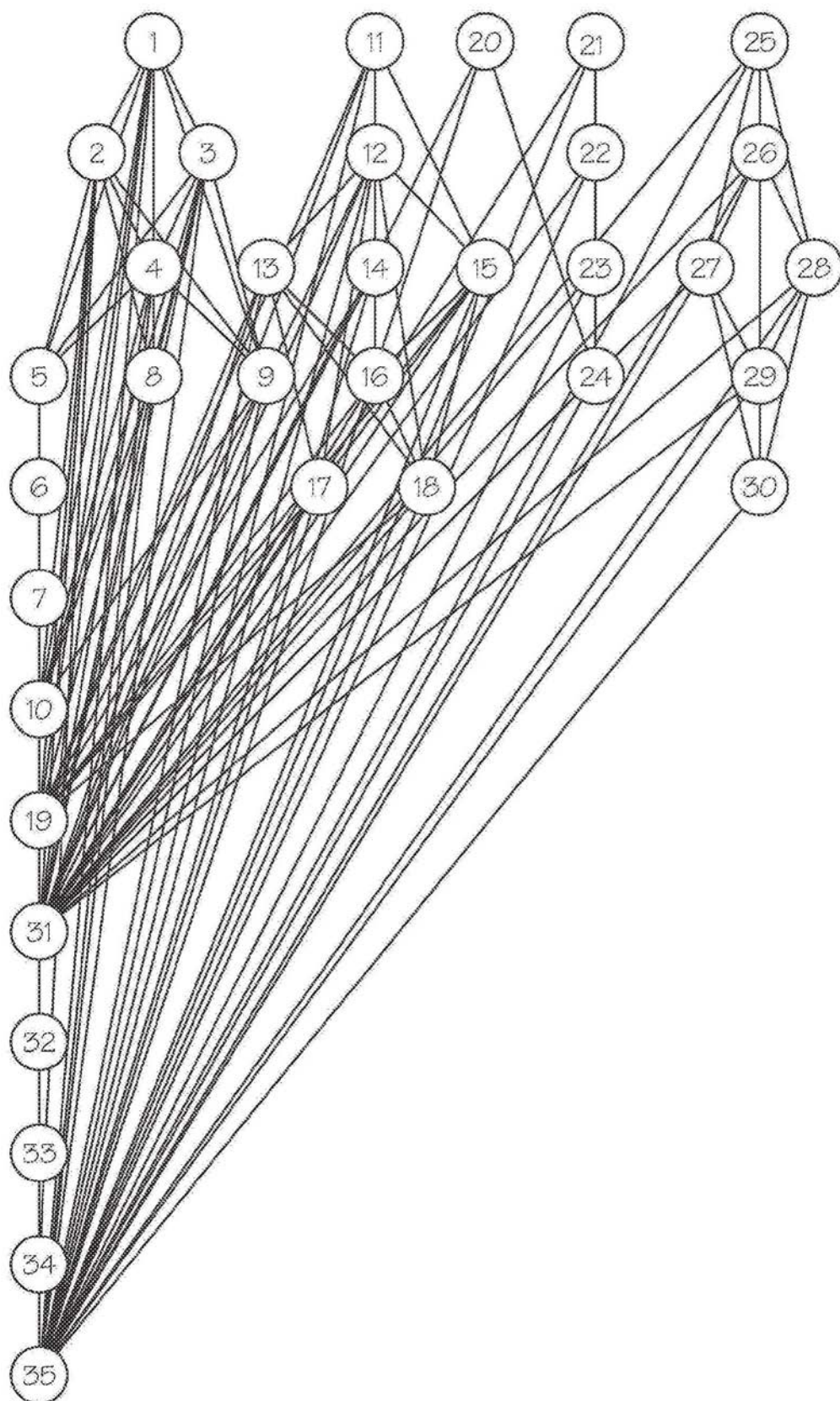


图5A

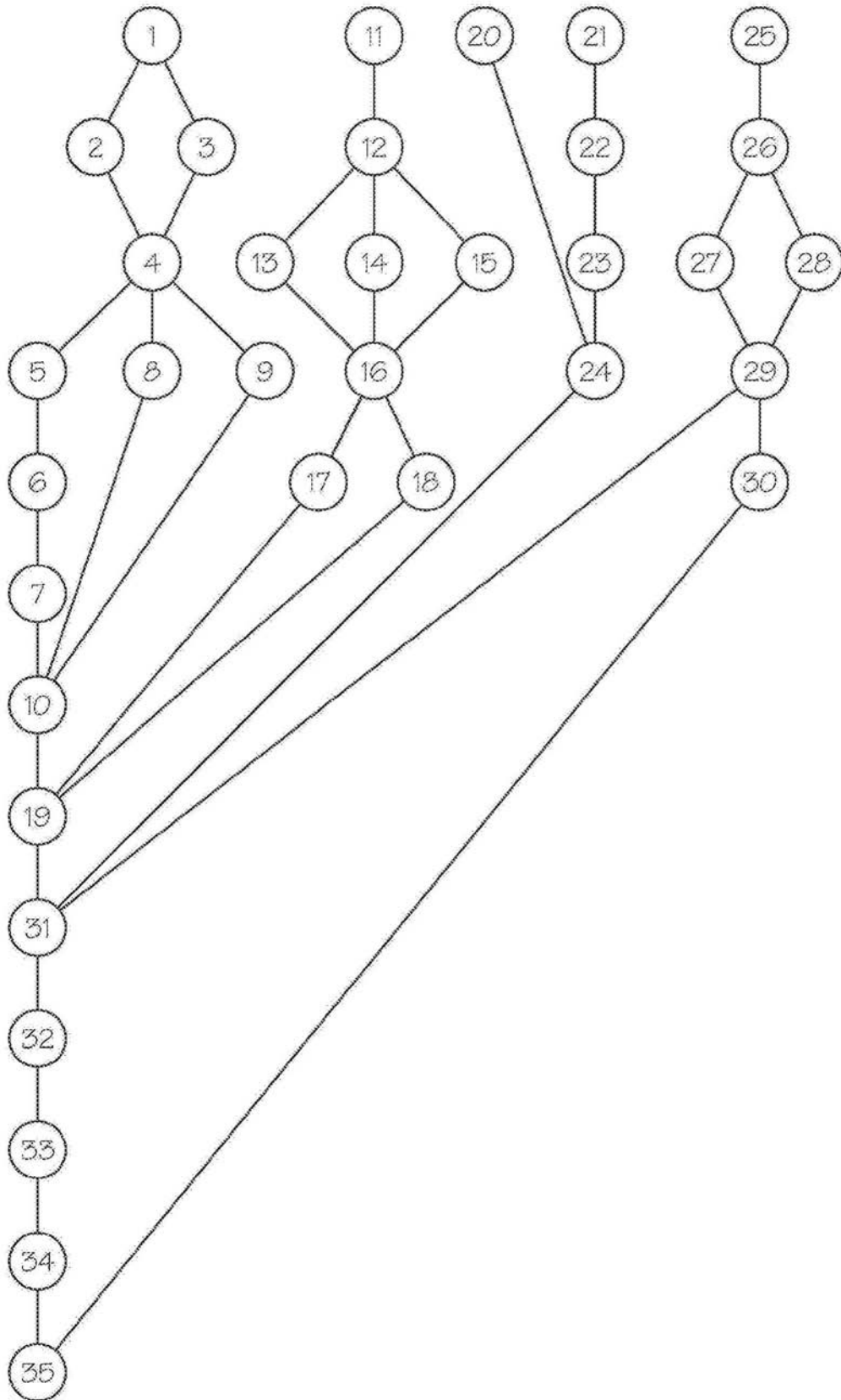


图5B

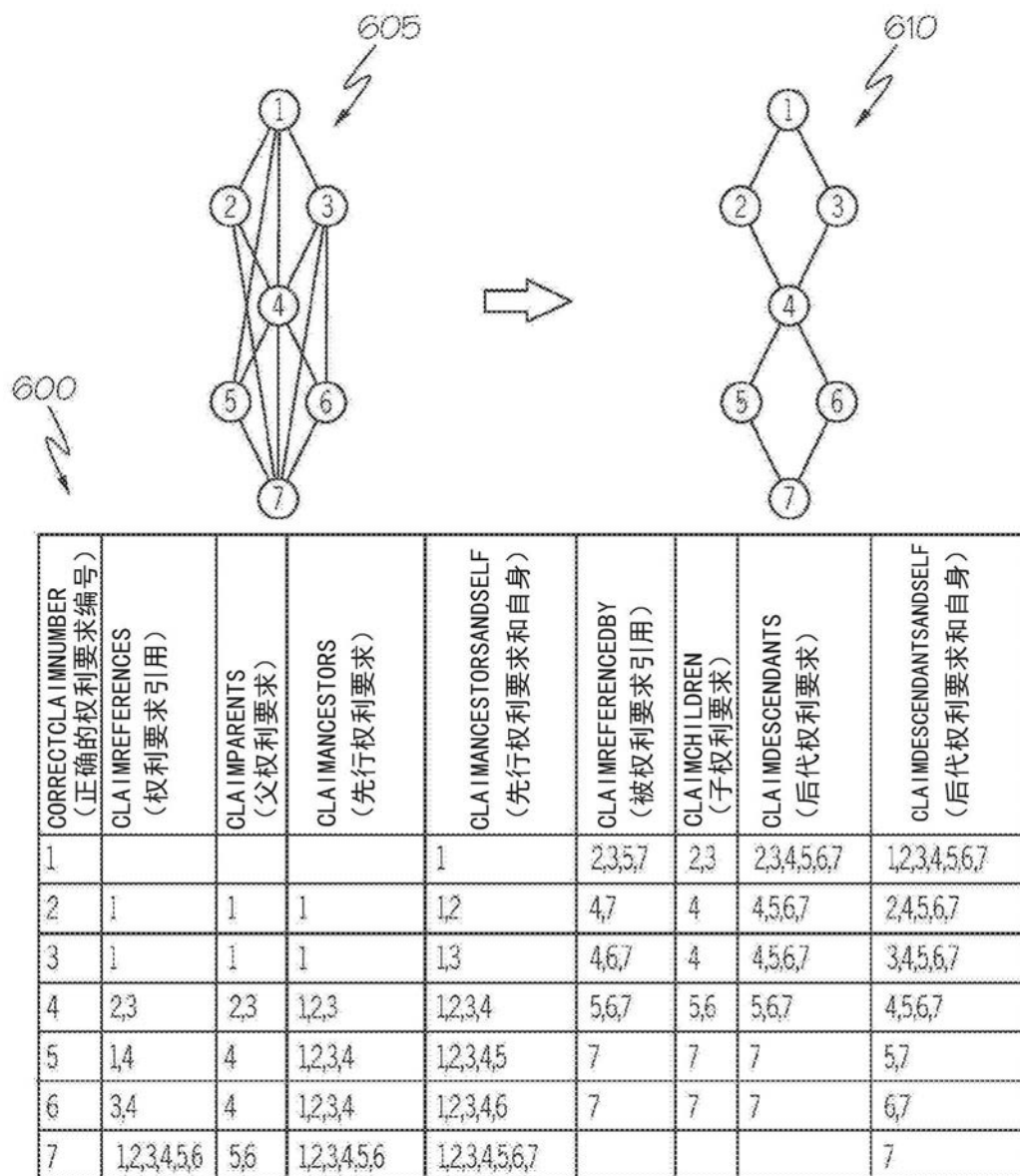


图6

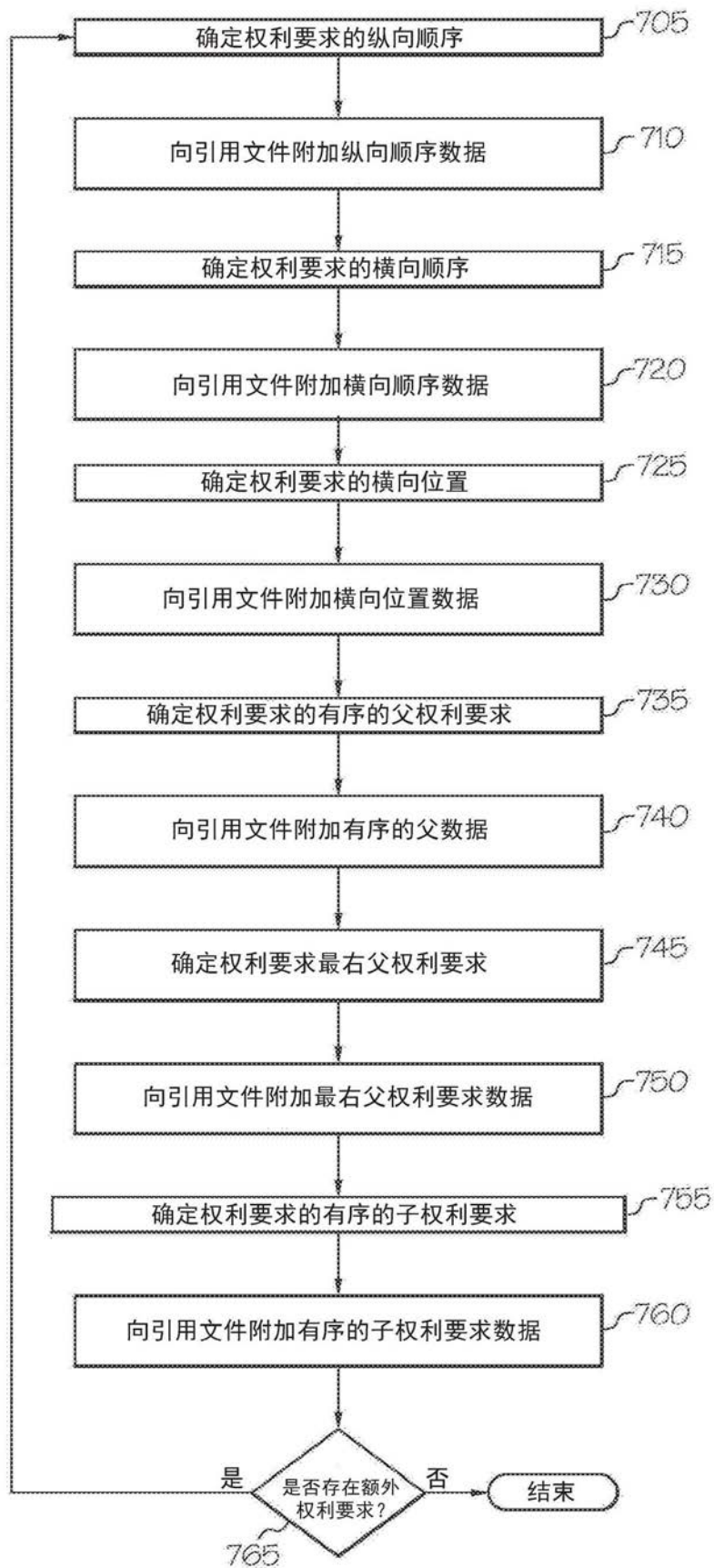


图7

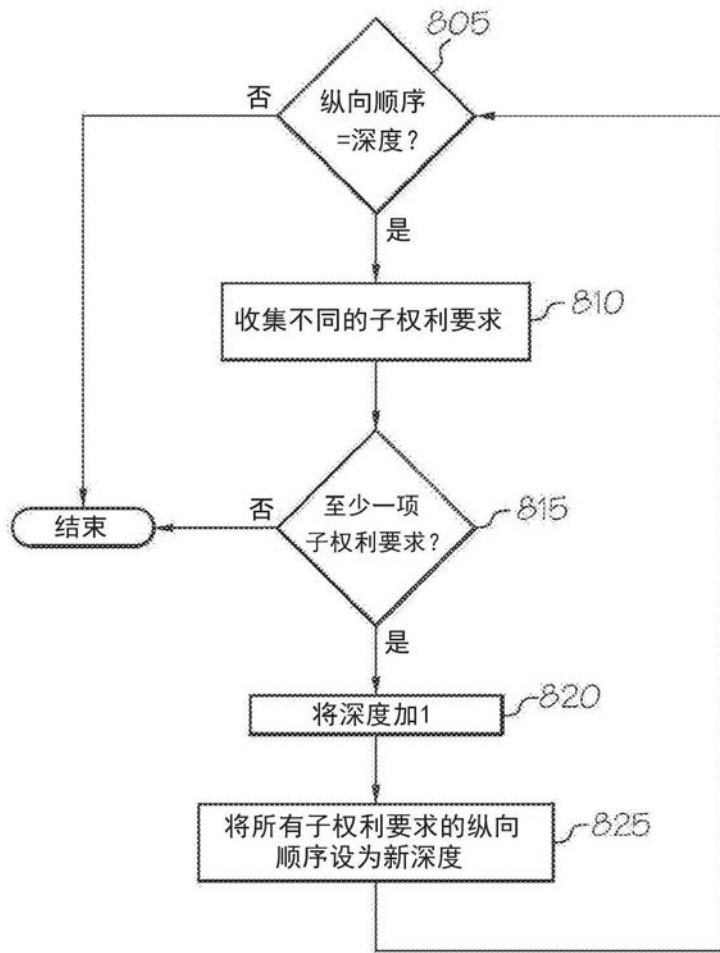


图8

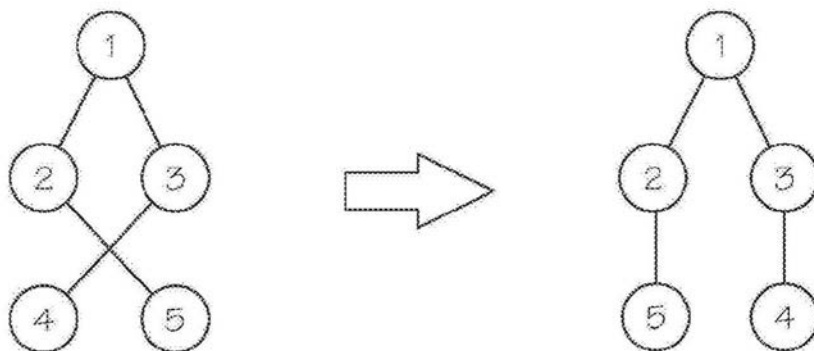


图9

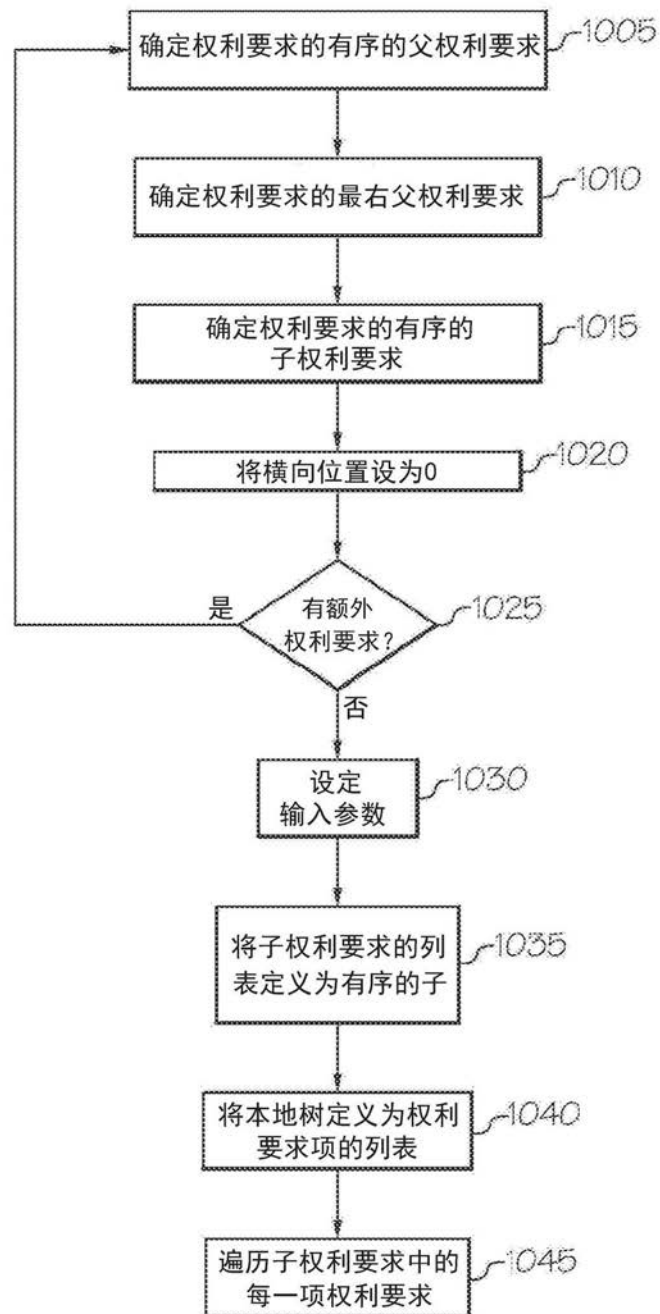


图10