(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2020 年 11 月 19 日 (19.11.2020)

WIPO | PCT

(10) 国际公布号
WO 2020/228182 A1

(51) 国际专利分类号:
*G06F 16/31* (2019.01) *G06F 16/33* (2019.01)

(21) 国际申请号: PCT/CN2019/103446

(22) 国际申请日: 2019 年 8 月 29 日 (29.08.2019)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201910401427.5 2019年5月15日 (15.05.2019) CN

(71) 申请人: 平安科技（深圳）有限公司 (PING AN TECHNOLOGY (SHENZHEN) CO., LTD.) [CN/CN]; 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518000 (CN)。

(72) 发明人: 王保军 (WANG, Baojun); 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518000 (CN)。 江腾飞 (JIANG, Tengfei); 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518000 (CN)。

(74) 代理人: 深圳市世联合知识产权代理有限公司 (SL INTELLECTUAL PROPERTY CO., LTD.); 中国广东省深圳市福田区泰然六路泰然科技园205栋二层203-206, Guangdong 518000 (CN)。

(81) 指定国 (除另有指明，要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS,

(54) **Title:** BIG DATA-BASED DATA DEDUPLICATION METHOD AND APPARATUS, DEVICE, AND STORAGE MEDIUM

(54) **发明名称:** 基于大数据的数据去重的方法、装置、设备及存储介质

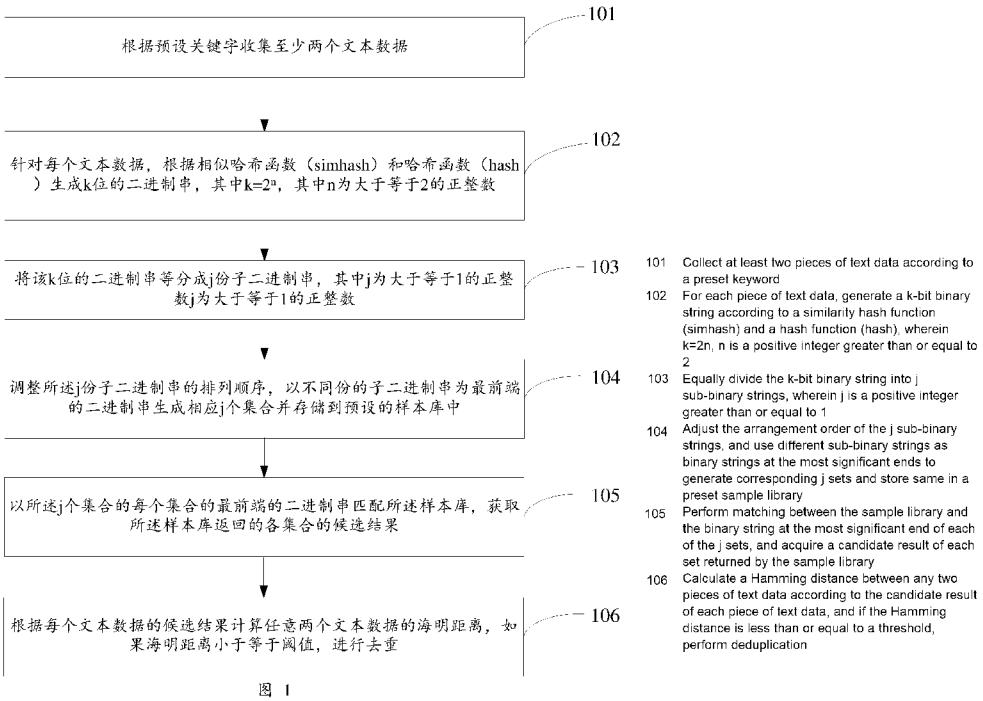


图 1

(57) **Abstract:** A big data-based data deduplication method and apparatus, a device, and a storage medium, which belong to the field of Internet. Said method comprises: collecting at least two pieces of text data according to a preset keyword (101); for each piece of text data, generating a k-bit binary string number according to a similarity hash function and a hash function (102); adjusting the arrangement order of the j sub-binary strings, and using different sub-binary strings as binary strings at the most significant ends to generate corresponding j sets and storing same in a preset sample library (104); performing matching between the sample library and

WO 2020/228182 A1

[见续页]

JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

**(84) 指定国**(除另有指明，要求每一种可提供的地区保护)：ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

**本国际公布**:

— 包括国际检索报告(条约第21条(3))。

the binary string at the most significant end of each of the j sets, and acquiring a candidate result of each set returned by the sample library (105); and calculating a Hamming distance between any two pieces of text data according to the candidate result of each piece of text data, and if the Hamming distance is less than or equal to a threshold, performing deduplication (106). A hash algorithm is used to perform dimension reduction of data, reducing the time for comparison of two texts, and reducing overheads for text storage.

**(57) 摘要**：一种基于大数据的数据去重的方法、装置、设备及存储介质，属于互联网领域。方法包括根据预设关键字收集至少两个文本数据（101）；针对每个文本数据，根据相似哈希函数和哈希函数生成k位的二进制串数（102）；调整所述j份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应j个集合并存储到预设的样本库中（104）；以所述j个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果（105）；根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重（106）。使用哈希算法对数据进行降维，减少两个文本的对比时间，降低对文本储存开销。

# 基于大数据的数据去重的方法、装置、设备及存储介质

## 【交叉引用】

本申请以2019年5月15日提交的申请号为201910401427.5，名称为“一种基于大数据的数据去重的方法、装置及存储介质”的中国发明专利申请为基础，并要求其优先权。

## 【技术领域】

本申请涉及互联网技术领域，尤其涉及一种基于大数据的数据去重的方法、装置、设备及存储介质。

## 【背景技术】

大数据（big data），是指无法在一定时间范围内用常规软件工具进行捕捉、管理和处理的数据集合，是需要新处理模式才能具有更强的决策力、洞察发现力和流程优化能力的海量、高增长率和多样化的信息资产。

大数据具有如下几大特点：

容量（Volume）：数据的大小决定所考虑的数据的价值和潜在的信息；

种类（Variety）：数据类型的多样性；

速度（Velocity）：指获得数据的速度；

可变性（Variability）：妨碍了处理和有效地管理数据的过程；

真实性（Veracity）：数据的质量；

复杂性（Complexity）：数据量巨大，来源多渠道；

价值（value）：合理运用大数据，以低成本创造高价值。

大数据最小的基本单位是比特（bit），按顺序给出所有单位：bit、字节（Byte）、KB、MB、GB、TB、PB、EB、ZB、YB、BB、NB、DB，除了1 Byte =8 bit，其它的单位之间按照进率1024（2的十次方）来计算。

随着信息爆炸时代的来临，以及云技术的应用，大数据吸引了越来越多的关注，大数据的处理技术主要包括大规模并行处理（MPP）数据库、数据挖掘、分布式文件系统、分布式数据库、云计算平台、互联网和可扩展的存储系统。

当在网络中多次从同一目录下备份相同的文件，或者从多个地址处备份相同的文件时，就会出现重复的数据，重复数据大大增加了分析系统的I/0和CPU处理压力，如果不做去重处理，那数据的分析效率会降低，并导致分析系统的硬件开销增大，而对于按照分析总流量进行收费项目，那多余的分析成本花费，是不可接受的。重复数据在进行大数据处理时尤其严重，因为目前互联网上充斥着大量的近重复信息，所以对于大数据挖掘来说，重复的数据会导致对于某方面做出误判，也就是无效的大数据。

因此，有必要对重复数据进行去重，来避免上述问题的发生。

现有的数据去重技术，是根据数据的开销负载（payload）、全数据或自定义规则进行数据比对，从而判断是否有重复，然后做多余数据的过滤去重。

还有一种现有技术的数据去重技术，是比较两个文本相似性，大多是将文本分词之后，

转化为特征向量距离的度量，比如常见的欧氏距离、海明距离或者余弦角度等等。

上述描述的数据去重技术能很好的适用于数据量少的场景，但当互联网存在大量的重复数据时，上述数据去重技术难以适用于海量数据处理的场景，否则会大大增加分析系统的I/O和CPU处理压力，浪费资源。

## 【发明内容】

本申请实施例的目的在于提出一种基于大数据的数据去重的方法、装置、计算机设备及存储介质，使用哈希算法对数据进行降维，可以减少两个文本的对比时间，降低对文本储存开销。

为了解决上述技术问题，本申请实施例提供一种基于大数据的数据去重的方法，采用了如下所述的技术方案：

根据预设关键字收集至少两个文本数据；

针对每个文本数据，根据相似哈希函数和哈希函数生成k位的二进制串，其中$k=2^n$，其中n为大于等于2的正整数；

将该k位的二进制串等分成j份子二进制串，其中j为大于等于1的正整数；

调整所述j份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应j个集合并存储到预设的样本库中；

以所述j个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果；

根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重。

为了解决上述技术问题，本申请实施例还提供一种基于大数据的数据去重的装置，采用了如下所述的技术方案，所述基于大数据的数据去重的装置包括：

收集模块，用于根据预设关键字收集至少两个文本数据；

处理模块，用于针对每个文本数据，根据相似哈希函数和哈希函数生成k位的二进制串，其中$k=2^n$，其中n为大于等于2的正整数；

分裂模块，用于将该k位的二进制串等分成j份子二进制串，其中j为大于等于1的正整数；

调整模块，调整所述j份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应j个集合并存储到预设的样本库中；

匹配模块，用于以所述j个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果；

计算模块，用于根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重。

为了解决上述技术问题，本申请实施例还提供一种计算机设备，该计算机设备包括存储器、处理器，以及存储在所述存储器中并可在所述处理器上运行的计算机可读指令，所述处理器执行所述计算机可读指令时实现所述的基于大数据的数据去重的方法的步骤。

为了解决上述技术问题，本申请实施例还提供一个或多个存储有计算机可读指令的非易失性可读存储介质，所述计算机可读指令被一个或多个处理器执行时，使得所述一个或多个

处理器执行所述的基于大数据的数据去重的方法的步骤。

上述基于大数据的数据去重的方法、装置、设备及存储介质，针对每个文本数据，根据相似哈希函数和哈希函数生成 k 位的二进制串，分成 j 份子二进制串并进行排列，以不同份的子二进制串为最前端的二进制串生成相应 j 个集合并存储到预设的样本库中，以每个集合的最前端的二进制串匹配所述样本库，得到候选结果，再通过计算任意两个文本数据的海明距离判断是否进行去重，因此，使用哈希算法对大数据进行降维，可以减少两个文本的对比时间，降低对文本储存开销。

## 【附图说明】

为了更清楚地说明本申请中的方案，下面将对本申请实施例描述中所需要使用的附图作一个简单介绍，显而易见地，下面描述中的附图是本申请的一些实施例，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。

图 1 根据本申请的一种基于大数据的数据去重的方法的一个实施例的流程图；

图 2 是图 1 中步骤 102 的一种具体实施方式的流程图；

图 3 是根据本申请的一种基于大数据的数据去重的装置的一个实施例的结构示意图；

图 4 是根据本申请的计算机设备的一个实施例的结构示意图。

附图标记：301-收集模块、302-处理模块、303-分裂模块、304-调整模块、305-匹配模块、306-计算模块、307-总线、41-存储器、42-处理器和 43-网络接口

## 【具体实施方式】

除非另有定义，本文所使用的所有的技术和科学术语与属于本申请的技术领域的技术人员通常理解的含义相同；本文中在申请的说明书中所使用的术语只是为了描述具体的实施例的目的，不是旨在于限制本申请；本申请的说明书和权利要求书及上述附图说明中的术语“包括”和“具有”以及它们的任何变形，意图在于覆盖不排他的包含。本申请的说明书和权利要求书或上述附图中的术语“第一”、“第二”等是用于区别不同对象，而不是用于描述特定顺序。

在本文中提及“实施例”意味着，结合实施例描述的特定特征、结构或特性可以包含在本申请的至少一个实施例中。在说明书中的各个位置出现该短语并不一定均是指相同的实施例，也不是与其它实施例互斥的独立的或备选的实施例。本领域技术人员显式地和隐式地理解的是，本文所描述的实施例可以与其它实施例相结合。

为了使本技术领域的人员更好地理解本申请方案，下面将结合附图，对本申请实施例中的技术方案进行清楚、完整地描述。

如图 1 所示，为本申请一实施例的一种基于大数据的数据去重的方法流程示意图，该基于大数据的数据去重的方法可以如下所述。

步骤 101，根据预设关键字收集至少两个文本数据。

例如，利用网络爬虫技术根据预设的关键字抓取与所述关键字相关的至少两个文本数据，

将所述至少两个文本数据保存在缓存器或存储器的数据仓库中。

网络爬虫（又被称为网页蜘蛛、网络机器人、网页追逐者），是一种按照一定的规则，自动地抓取万维网信息的计算机可读指令或者脚本。它为搜索引擎从万维网上下载网页，是搜索引擎的重要组成。传统爬虫从一个或若干初始网页的URL开始,获得初始网页上的URL,在抓取网页的过程中，不断从当前页面上抽取新的URL放入队列，直到满足系统的一定停止条件。

本实施例中，可以通过聚焦网络爬虫收集多个文本数据，该聚焦网络爬虫根据既定的抓取目标（例如，客户的投资类别信息），有选择的访问日志记录、APP反馈、微信或万维网上的网页与相关的链接，获取所需要的信息。例如，通过聚焦网络爬虫搜索时，设置投资数据有关的关键字，例如，所述关键字可以为：姓名、身份证号码、地址、电话号码、银行账号、邮箱地址、所属城市、邮编、密码类( 如账户查询密码、取款密码、登录密码等)、组织机构名称、营业执照号码、银行帐号、交易日期、交易金额等等。然后网络爬虫在日志记录、APP反馈、微信或万维网上的网页上抓取与关键字相关的文本数据，例如，所述相关是指包含所述关键字的文本数据，把这些收集到的文本数据按照各种维度保存在缓存器或存储器的数据仓库中，则该数据仓库的数据即为大数据。

步骤102，针对每个文本数据，根据相似哈希函数（simhash）和哈希函数（hash）生成k位的二进制串，其中$k=2^n$，其中n为大于等于2的正整数。

例如，针对大数据的每个文本数据（例如，Doc文本，web文本），利用simhash函数将该文本数据转化为哈希编码（hashcode），具体如下所述。

例如,以下三段文本为例进行说明:p1=the cat sat on the mat;p2=the cat sat on a mat;p3=we all scream for ice cream，整个过程可以如下所述，如图2所示，为图1中步骤102的一种具体实施方式的流程图。

步骤1021、选择simhash函数的位数k。

例如，根据存储成本以及数据集的大小，选择simhash的位数k，其中，$k=2^n$，n为大于等于2的正整数，例如k=16、32、64或128位。

步骤1022、将simhash函数的各位初始化为0。

步骤1023、将每个文本数据进行分词抽取，抽取出多个分词_权重对。

例如，将每个文本数据进行分词抽取（其中包括分词和计算权重），例如，抽取得到n个分词_权重对（feature_weight_pairs），记为 feature_weight_pairs = [fw1, fw2 ... fwn]，其中fwn = (feature_n,weight_n)，其中n为大于等于2的正整数。

例如,一般采用各种预定数的分词的方式,例如,所述预定数量为2或3,例如,对于"the cat sat on the mat",采用两两分词的方式得到如下结果:{"th", "he", "e ", " c", "ca", "at", "t ", " s", "sa", " o", "on", "n ", " t", " m", "ma"}，其中，空格也算一个字母。

步骤1024、对每个分词_权重对（feature_weight_pairs）中的分词（feature）进行hash函数处理。

例如，使用32位hash函数计算该文本数据的每个预定数量分词字母（word）的哈希代码（hashcode），计算文本数据每2个或3个字母的哈希代码（hashcode），比如："th".hash=-502157718 ，"he".hash =-369049682，…… 。

步骤1025、对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，生成k个数值。

例如，对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，如果该位是 1，则加 1，如果是 0，则减 1，最后生成 k（即 bits_count）个数值。

例如，采用 32 位 hash 函数，则 hash 生成的位数 bits_count = 32，对各分词（word）的 hashcode 的每一位，如果该位为 1，则 simhash 相应位的值加 1；否则减 1，得到 32 个数值（即 simhash 包括 32 个数值）。

步骤 1026、将生成的 k 个数值转换为 k 位的二进制串。

例如，对最后得到的 32 位的 simhash，如果该位大于 1，则设为 1；否则设为 0 。

在本申请的另一实施例中，也可以生成 64 或 128 位的二进制串，本实施例并不限定。

使用 simhash 会应该产生类似如下的结果：

irb(main):003:0> p1.simhash => 851459198 00110010110000000011110001111110

irb(main):004:0> p2.simhash => 847263864 00110010100000000011100001111000

irb(main):002:0> p3.simhash => 984968088 00111010101101010110101110011000。

经过 simhash 函数运算后，这三个文本的海明距离（hammingdistance）为两个二进制串中不同位的数量。

步骤 103，将该 k 位的二进制串等分成 j 份子二进制串，其中 j 为大于等于 1 的正整数 j 为大于等于 1 的正整数。

例如，将 32 位或 64 位的二进制串等分成四份，例如，当将 32 位的二进制串等分成四份时，每份包括 8 位子二进制串，例如，当将 64 位的二进制串等分成四份时，每份包括 16 位二进制串。例如，将 64 位的二进制串等分成四份 16 位的子二进制串：$L_{1-16}$，$L_{17-32}$，$L_{33-48}$ 和 $L_{48-64}$，$L_{1-16}$，$L_{17-32}$，$L_{33-48}$ 和 $L_{48-64}$ 分别包括 16 位的二进制串。

上述实施例仅仅是将 32 位或 64 位的二进制串等分成四份为例进行说明，但本申请的实施例并不限制分成多少分，例如，j 为大于等于 1 的正整数，例如 j 可以为 2、3、4、5、6、7 或 8 等。

步骤 104，调整所述 j 份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应 j 个集合并存储到预设的的样本库中。

当将 64 位的二进制串等分成四份时，可以将任意一份 16 位子二进制串调整作为所述四份子二进制串的最前端的二进制串，例如，子二进制串 $L_{1-16}$，$L_{17-32}$，$L_{33-48}$ 和 $L_{48-64}$ 可以分别调整作为所有二进制串的最前端的二进制串，则存在 4 个集合，可以以表格（table）存储于预设的样本库中，例如，存储到预设的存储器中，即在存储器中存储有 4 个 table，例如所述 4 个集合分别为：（$L_{1-16}$，$L_{17-32}$，$L_{33-48}$，$L_{48-64}$）、（$L_{17-32}$，$L_{1-16}$，$L_{33-48}$，$L_{48-64}$）、（$L_{33-48}$，$L_{1-16}$，$L_{17-32}$，$L_{48-64}$）、（$L_{48-64}$，$L_{1-16}$，$L_{17-32}$，$L_{33-48}$）。

上述实施例仅仅是以最前面的一份子二进制串进行集合分类，后面的子二进制串如何排列并不限定。例如，在本申请的另一实施例中，还可以以其他方式进行集合分类，例如，将 64 位的二进制串等分成两份，每份包括 32 位子二进制串，例如子二进制串 $L_{1-32}$ 和 $L_{33-64}$。可以将任意一份 32 位子二进制串调整作为最前端的二进制串，例如，将子二进制串 $L_{1-32}$ 和 $L_{33-64}$ 分别调整作为最前端的二进制串，则存在 2 个集合，可以以表格（table）存储于存储器的样本库中，即在存储器中存储有 2 个 table，例如，所述 2 个集合分别为（$L_{1-32}$，$L_{33-64}$）和（$L_{33-64}$，$L_{1-32}$）。

步骤 105，以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果。

例如，以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，如果所述样本

库总有 $2^m$ 个哈希指纹，则针对每个集合返回 $2^{m-j}$ 个候选结果，其中，m 为大于 2 的整数，且 m>j。

例如，上述 64 位二进制串生成四个 table 时，利用匹配的方式查找最前 16 位子二进制串，如果样本库中存有 $2^{34}$（差不多 10 亿）的哈希指纹，则每个 table 返回 $2^{(34-16)}$=262144 个候选结果，相对于现有技术返回 $2^{34}$ 的哈希指纹，大大减少了海明距离的计算成本。

在本申请的另一实施例中，所述以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库具体包括：确定所述 j 份二进制串的每份二进制串的最前端的二进制串与存储器存储的最前端的二进制串是否完全相同，如果相同就确定匹配，即确定所述样本库当前反馈的候选结果为正确的候选结果，如果不同就确定不匹配，即确定将所述样本库当前反馈的候选结果为不正确的候选结果。

步骤 106、根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重（即丢弃或删除其中一个文本）。

例如，二进制串 A 和二进制串 B 的海明距离就是 A xor B 后二进制中 1 的个数。

例如，二进制串 A = 100111，二进制串 B = 101010，则 hamming_distance(A, B) = count_1(A xor B) = count_1(001101) = 3。

通过 simhash 算法将高维的特征向量映射成一个 f-bit 的指纹(fingerprint)，通过比较两个文本的 f-bit 指纹的海明距离（Hamming Distance）来确定两个文本是否重复或者高度近似，即海明距离的值越小就越相似，当海明距离等于零时，说明两个比较文本相同，海明距离的值越大就越不相似。

例如，上述三个文本 p1，p2 和 p3 的 simhash 结果，其两两之间的海明距离为(p1，p2)=4，(p1，p3)=16 以及(p2，p3)=12。则两两文本之间的相似度，p1 和 p2 间的相似度要远大于与 p3 的相似度。

综上所述，上述实施例描述的基于大数据的数据去重的方法，simhash 函数运算和 hash 函数运算最大的不同在于 hash 函数虽然也可以用于映射来比较文本的重复，但是对于可能差距只有一个字节的文本也会映射成两个完全不同的哈希结果，而 simhash 函数对相似的文本的哈希映射结果也相似。例如，设置 simhash 函数为 64 位，即 f=64，将文本的加权特征集合映射到一个 64-bit 的哈希指纹（fingerprint）上。

例如，设置 simhash 函数为 64 位，将 64 位的二进制串等分成 4 份子二进制串，然后调整上述 64 位二进制，将任意一份子二进制串作为前 16 位，总共有四种组合，生成四份 table 并存储到样本库中，采用精确匹配的方式查找前 16 位，如果样本库中存有 $2^{34}$（差不多 10 亿）的哈希指纹，则每个 table 返回 $2^{(34-16)}$=262144 个候选结果，大大减少了海明距离的计算成本。

因此，本申请的实施例描述的基于大数据的数据去重的方法，使用哈希算法对大数据进行降维，可以减少两个文本的对比时间，降低对文本储存开销。

需要说明的是，本申请实施例所提供的基于大数据的数据去重的方法一般由服务器/终端设备执行，相应地，基于大数据的数据去重的方法装置一般设置于服务器/终端设备中。所述终端设备可以是无线终端也可以是有线终端，无线终端可以是指向用户提供语音和/或数据连通性的设备，具有无线连接功能的手持式设备、或连接到无线调制解调器的其他处理设备。终端可以是便携式、袖珍式、手持式、计算机内置的或者车载的移动装置。

应该理解，终端设备、网络和服务器的数目仅仅是示意性的。根据实现需要，可以具有任意数目的终端设备、网络和服务器。

本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，是可以通过计算机可读指令来指令相关的硬件来完成，该计算机可读指令可存储于非易失性可读取存储介质中，该计算机可读指令在执行时，可包括如上述各方法的实施例的各个流程。其中，前述的存储介质可为磁碟、光盘、只读存储记忆体（Read-Only Memory，ROM）等非易失性存储介质。

应该理解的是，虽然附图的流程图中的各个步骤按照箭头的指示依次显示，但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明，这些步骤的执行并没有严格的顺序限制，其可以以其他的顺序执行。而且，附图的流程图中的至少一部分步骤可以包括多个子步骤或者多个阶段，这些子步骤或者阶段并不必然是在同一时刻执行完成，而是可以在不同的时刻执行，其执行顺序也不必然是依次进行，而是可以与其他步骤或者其他步骤的子步骤或者阶段的至少一部分轮流或者交替地执行。

进一步参考图 3，作为对上述图 1 所示方法的实现，本申请提供了一种基于大数据的数据去重装置的一个实施例，该装置实施例与图 1 所示的方法实施例相对应，该装置具体可以应用于各种电子设备中。

如图 3 所示，本实施例所述的基于大数据的数据去重装置 300 包括：收集模块 301、处理模块 302、分裂模块 303、调整模块 304、匹配模块 305、计算模块 306 和总线 307。所述收集模块 301、所述处理模块 302、所述分裂模块 303、所述调整模块 304、所述匹配模块 305 和所述计算模块 306 相互之间通过所述总线 307 连接。本实施例的模块划分仅仅是示意性，还可以根据各自方法动作进行各自逻辑划分。

所述总线 307 用于实现这些组件之间的连接通信。例如，所述总线 307 可以是工业标准体系结构（Industry Standard Architecture，ISA）总线、外设部件互连标准（Peripheral Component Interconnect，PCI）总线或扩展工业标准结构（Extended Industry Standard Architecture，EISA）总线等。该总线系统可以分为地址总线、数据总线、控制总线等。为便于表示，图中仅用一条粗线表示，但并不表示仅有一根总线或一种类型的总线。

所述收集模块 301，用于根据预设关键字收集至少两个文本数据；

所述处理模块 302，用于针对每个文本数据，根据相似哈希函数（simhash）和哈希函数（hash）成 k 位的二进制串，其中 $k=2^n$，其中 n 为大于等于 2 的正整数；

所述分裂模块 303，用于将该 k 位的二进制串等分成 j 份子二进制串，其中 j 为大于等于 1 的正整数；

所述调整模块 304，用于调整所述 j 份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应 j 个集合并存储到预设的样本库中；

所述匹配模块 305，用于以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果。例如，所述匹配模块 305 用于以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，如果所述样本库总有 $2^m$ 个哈希指纹，则针对每个集合返回 $2^{m-j}$ 个候选结果，其中，m 为大于 2 的整数，且 m>j；

所述计算模块 306，用于根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重。

以下三段文本为例进行说明：p1=the cat sat on the mat；p2=the cat sat on a mat；p3=we all scream for ice cream。

在本申请的另一实施例中，例如，所述处理模块 302 用于针对大数据的每个文本数据（例

如，Doc 文本，web 文本），利用 simhash 函数将该文本数据转化为哈希编码（hashcode），例如，所述处理模块 302 还包括：选择子单元、初始化子单元、抽取子单元、哈希函数处理子单元、累加子单元和处理子单元，其中，该选择子单元、该初始化子单元、该抽取子单元、该哈希函数处理子单元、该累加子单元和该处理子单元任意两者相互之间可以通信连接。

所述选择子单元，用于选择相似哈希函数的位数 k，例如，所述选择子单元用于根据存储成本以及数据集的大小，选择 simhash 的位数 k，其中，$k=2^n$，n 为大于等于 2 的正整数，例如 k=16、32、64 或 128 位。

初始化子单元，用于将相似哈希函数的各位初始化为 0；

抽取子单元，用于将每个文本数据进行分词抽取，抽取出多个分词_权重对。例如，所述抽取子单元对每个分词_权重对中的分词进行哈希函数处理，例如，所述抽取子单元用于使用 k 位哈希函数计算每个文本数据的预定数量分词字母的哈希代码。例如，所述预定数量为 2 或 3。

所述抽取子单元还用于将每个文本数据进行分词抽取（其中包括分词和计算权重），例如，抽取得到 n 个(分词，权重)(分词_权重)对(feature_weight_pairs)，记为 feature_weight_pairs = [fw1, fw2 ... fwn]，其中 fwn = (feature_n,weight_n)，其中 n 为大于等于 2 的正整数。例如，一般采用各种预定数的分词的方式，例如，所述预定数量为 2 或 3，例如，对于"the cat sat on the mat"，采用两两分词的方式得到如下结果：{"th", "he", "e ", " c", "ca", "at", "t ", " s", "sa", " o", "on", "n ", " t", " m", "ma"}，其中，空格也算一个字母。

哈希函数处理子单元，用于对每个分词_权重对中的分词进行哈希函数处理。例如，例如，哈希函数处理子单元用于使用 32 位 hash 函数计算该文本数据的每个预定数量分词字母(word)的哈希代码（hashcode），计算文本数据每 2 个或 3 个字母的哈希代码（hashcode），比如："th".hash = -502157718，"he".hash = -369049682，..... 。

累加子单元，用于对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，生成 k 个数值，例如，所述累加子单元对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，如果该位是 1，则加权重，如果是 0，则减权重，最后生成 k 个数值。例如，累加子单元采用 32 位 hash 函数，则 hash 生成的位数 bits_count = 32，对各分词（word）的 hashcode 的每一位，如果该位为 1，则 simhash 相应位的值加 1；否则减 1，得到 32 个数值（即 simhash 包括 32 个数值）。

处理子单元，用于将所述生成的 k 个数值转换为 k 位的二进制串。例如，所述 k 为 32、64 或 128，例如，所述处理子单元对最后得到的 32 位的 simhash，如果该位大于 1，则设为 1；否则设为 0 。

在本申请的另一实施例中，处理子单元也可以生成 64 或 128 位的二进制串，本实施例并不限定。

使用 simhash 会应该产生类似如下的结果：

irb(main):003:0> p1.simhash => 851459198 00110010110000000011110001111110

irb(main):004:0> p2.simhash => 847263864 00110010100000000011100001111000

irb(main):002:0> p3.simhash => 984968088 00111010101101010110101110011000。

经过 simhash 函数运算后，这三个文本的海明距离（hammingdistance）为两个二进制串中不同位的数量。

在本申请的另一实施例中，所述匹配模块 305 用于以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库具体包括：所述匹配模块 305 用于将所述 j 个集合的每个集合的

最前端的二进制串与存储器存储的每个集合的最前端的二进制串进行相同性判断，如果相同就确定匹配，即如果相同就确定所述样本库当前反馈的候选结果为正确的候选结果，如果不同就确定不匹配，即如果不相同就确定所述样本库当前反馈的候选结果为不正确的候选结果。

在本申请的另一实施例中，所述分裂模块303还用于将32位或64位的二进制串等分成四份，例如，当将32位的二进制串等分成四份时，每份包括8位子二进制串，例如，当将64位的二进制串等分成四份时，每份包括16位二进制串。例如，所述分裂模块303还用于将64位的二进制串等分成四份16位的子二进制串：$L_{1-16}$，$L_{17-32}$，$L_{33-48}$和$L_{48-64}$，$L_{1-16}$，$L_{17-32}$，$L_{33-48}$和$L_{48-64}$分别包括16位的二进制串。

上述实施例仅仅是将32位或64位的二进制串等分成四份为例进行说明，但本申请的实施例并不限制分成多少分，例如，j为大于等于1的正整数，例如j可以为2、4、6、或8等偶数。

在本申请的另一实施例中，所述调整模块304还用于将64位的二进制串等分成四份时，可以将任意一份16位子二进制串调整作为所述四份子二进制串的最前端的二进制串，例如，子二进制串$L_{1-16}$，$L_{17-32}$，$L_{33-48}$和$L_{48-64}$可以分别调整作为所有二进制串的最前端的二进制串，则存在4个集合，可以以表格（table）存储于存储器中，即在存储器中存储有4个table，例如所述4个集合分别为：（$L_{1-16}$，$L_{17-32}$，$L_{33-48}$，$L_{48-64}$）、（$L_{17-32}$，$L_{1-16}$，$L_{33-48}$，$L_{48-64}$）、（$L_{33-48}$，$L_{1-16}$，$L_{17-32}$，$L_{48-64}$）、（$L_{48-64}$，$L_{1-16}$，$L_{17-32}$，$L_{33-48}$）。

例如，所述计算模块306还用于计算两个文本数据（例如，第一文本数据和第二文本数据）之间的海明距离，二进制串A ()和二进制串B的海明距离就是A xor B后二进制中1的个数。

例如，二进制串A = 100111，二进制串B = 101010，则hamming_distance(A, B) = count_1(A xor B) = count_1(001101) = 3。

所述计算模块306还用于通过simhash算法将高维的特征向量映射成一个f-bit的指纹(fingerprint)，通过比较两个文本的f-bit指纹的海明距离（Hamming Distance）来确定两个文本是否重复或者高度近似，即海明距离的值越小就越相似，当海明距离等于零时，说明两个比较文本相同，海明距离的值越大就越不相似。

例如，上述三个文本p1，p2和p3的simhash结果，其两两之间的海明距离为(p1，p2)=4，(p1，p3)=16以及(p2，p3)=12。则两两文本之间的相似度，p1和p2间的相似度要远大于与p3的相似度。

本实施例中，上述模块均可以通过一个或多个处理器、芯片或集成电路实现，本实施例并不限定。

因此，本申请的实施例描述的基于大数据的数据去重的装置，使用哈希算法对大数据进行降维，可以减少两个文本的对比时间，降低对文本储存开销。

为解决上述技术问题，本申请实施例还提供计算机设备。具体请参阅图4，图4为本实施例计算机设备基本结构框图。

所述计算机设备4包括通过系统总线相互通信连接存储器41、一个或多个处理器42、网络接口43。需要指出的是，图中仅示出了具有组件41-43的计算机设备4，但是应理解的是，并不要求实施所有示出的组件，可以替代的实施更多或者更少的组件。其中，本技术领域技术人员可以理解，这里的计算机设备是一种能够按照事先设定或存储的指令，自动进行数值计算和/或信息处理的设备，其硬件包括但不限于微处理器、专用集成电路(Application Specific

Integrated Circuit，ASIC)、可编程门阵列(Field—Programmable Gate Array，FPGA)、数字处理器 (Digital Signal Processor，DSP)、嵌入式设备等。

所述计算机设备可以是桌上型计算机、笔记本、掌上电脑及云端服务器等计算设备。所述计算机设备可以与用户通过键盘、鼠标、遥控器、触摸板或声控设备等方式进行人机交互。

所述存储器 41 至少包括一种类型的非易失性可读存储介质，所述非易失性可读存储介质包括闪存、硬盘、多媒体卡、卡型存储器(例如，SD 或 DX 存储器等)、随机访问存储器(RAM)、静态随机访问存储器（Static Random-Access Memory，SRAM)、只读存储器（ROM)、电可擦除可编程只读存储器（Electrically Erasable Programmable read only memory，EEPROM)、可编程只读存储器（Programmable read-only memory，PROM)、磁性存储器、磁盘、光盘等。在一些实施例中，所述存储器 41 可以是所述计算机设备 4 的内部存储单元，例如该计算机设备 4 的硬盘或内存。在另一些实施例中，所述存储器 41 也可以是所述计算机设备 4 的外部存储设备，例如该计算机设备 4 上配备的插接式硬盘，智能存储卡（Smart Media Card, SMC)，安全数字（Secure Digital, SD）卡，闪存卡（Flash Card）等。当然，所述存储器 41 还可以既包括所述计算机设备 4 的内部存储单元也包括其外部存储设备。本实施例中，所述存储器 41 通常用于存储安装于所述计算机设备 4 的操作系统和各类应用软件，例如上述数据处理方法的计算机可读指令等。此外，所述存储器 41 还可以用于暂时地存储已经输出或者将要输出的各类数据。

所述处理器 42 在一些实施例中可以是中央处理器（Central Processing Unit，CPU)、控制器、微控制器、微处理器、或其他数据处理芯片。该处理器 42 通常用于控制所述计算机设备 4 的总体操作。本实施例中，所述处理器 42 用于运行所述存储器 41 中存储的计算机可读指令或者数据去重，例如运行所述基于大数据的数据去重方法的计算机可读指令。

所述网络接口 43 可包括无线网络接口或有线网络接口，该网络接口 43 通常用于在所述计算机设备 4 与其他电子设备之间建立通信连接。

所述处理器 42，用于根据预设关键字收集至少两个文本数据；针对每个文本数据，根据相似哈希函数（simhash）和哈希函数（hash）成 k 位的二进制串，其中 k=2n，其中 n 为大于等于 2 的正整数；将该 k 位的二进制串等分成 j 份子二进制串，其中 j 为大于等于 1 的正整数；调整所述 j 份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应 j 个集合并存储到预设的样本库中；以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果；根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重。

在本申请的另一实施例中，所述处理器 42 还用于：将 64 位的二进制串等分成四份时，可以将任意一份 16 位子二进制串调整作为所述四份子二进制串的最前端的二进制串，例如，子二进制串 L1-16，L17-32，L33-48 和 L48-64 可以分别调整作为所有二进制串的最前端的二进制串，则存在 4 个集合，可以以表格（table）存储于存储器中，即在存储器中存储有 4 个 table，例如所述 4 个集合分别为：(L1-16, L17-32, L33-48, L48-64)、(L17-32, L1-16, L33-48, L48-64)、(L33-48，L1-16，L17-32，L48-64)、(L48-64，L1-16，L17-32，L33-48)。

所述处理器 42 还用于计算两个文本数据（例如，第一文本数据和第二文本数据）之间的海明距离，例如，二进制串 A 和二进制串 B 的海明距离就是 A xor B 后二进制中 1 的个数。

例如，二进制串 A=100111，二进制串 B=101010，则 hamming_distance(A, B)=count_1(A xor B) =count_1(001101) =3。

所述处理器还用于通过simhash算法将高维的特征向量映射成一个f比特（f-bit）的指纹(fingerprint)，其中，f为大于等于2的整数，通过比较两个文本的f-bit指纹的海明距离（Hamming Distance）来确定两个文本是否重复或者高度近似，即海明距离的值越小就越相似，当海明距离等于零时，说明两个比较文本相同，海明距离的值越大就越不相似。

例如，上述三个文本p1，p2和p3的simhash结果，其两两之间的海明距离为(p1，p2)=4，(p1，p3)=16 以及(p2，p3)=12。则两两文本之间的相似度，p1 和 p2 间的相似度要远大于与p3的相似度。

本申请还提供了另一种实施方式，即提供一种非易失性可读存储介质，所述非易失性可读存储介质存储有数据处理的可读指令，所述数据处理的可读指令可被至少一个处理器执行，以使所述至少一个处理器执行如上述的数据处理方法的步骤。

例如，所述数据处理的可读指令被至少一个处理器执行时，执行如下内容：根据预设关键字收集至少两个文本数据；针对每个文本数据，根据相似哈希函数（simhash）和哈希函数（hash）成k位的二进制串，其中k=2n，其中n为大于等于2的正整数；将该k位的二进制串等分成j份子二进制串，其中j为大于等于1的正整数；调整所述j份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应j个集合并存储到预设的样本库中；以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果；根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重。

在本申请的另一实施例中，所述数据处理的可读指令被至少一个处理器执行时，执行如下内容：将64位的二进制串等分成四份时，可以将任意一份16位子二进制串调整作为所述四份子二进制串的最前端的二进制串，例如，子二进制串L1-16，L17-32，L33-48和L48-64可以分别调整作为所有二进制串的最前端的二进制串，则存在4个集合，可以以表格（table）存储于存储器中，即在存储器中存储有4个table，例如所述4个集合分别为：（L1-16，L17-32，L33-48，L48-64）、（L17-32，L1-16，L33-48，L48-64）、（L33-48，L1-16，L17-32，L48-64）、（L48-64，L1-16，L17-32，L33-48）。

通过以上的实施方式的描述，本领域的技术人员可以清楚地了解到上述实施例方法可借助软件加必需的通用硬件平台的方式来实现，当然也可以通过硬件，但很多情况下前者是更佳的实施方式。基于这样的理解，本申请的技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质（如ROM/RAM、磁碟、光盘）中，包括若干指令用以使得一台终端设备（可以是手机，计算机，服务器，空调器，或者网络设备等）执行本申请各个实施例所述的方法。

显然，以上所描述的实施例仅仅是本申请一部分实施例，而不是全部的实施例，附图中给出了本申请的较佳实施例，但并不限制本申请的专利范围。本申请可以以许多不同的形式来实现，相反地，提供这些实施例的目的是使对本申请的公开内容的理解更加透彻全面。尽管参照前述实施例对本申请进行了详细的说明，对于本领域的技术人员来而言，其依然可以对前述各具体实施方式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效替换。凡是利用本申请说明书及附图内容所做的等效结构，直接或间接运用在其他相关的技术领域，均同理在本申请专利保护范围之内。

# 权 利 要 求 书

1、一种基于大数据的数据去重的方法，其特征在于，包括：

根据预设关键字收集至少两个文本数据；

针对每个文本数据，根据相似哈希函数和哈希函数生成 k 位的二进制串，其中 $k=2^n$，其中 n 为大于等于 2 的正整数；

将该 k 位的二进制串等分成 j 份子二进制串，其中 j 为大于等于 1 的正整数；

调整所述 j 份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应 j 个集合并存储到预设的样本库中；

以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果；

根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重。

2、根据权利要求 1 所述的基于大数据的数据去重的方法，其特征在于，所述针对每个文本数据，根据相似哈希函数和哈希函数生成 k 位的二进制串具体包括：

选择相似哈希函数的位数 k；

将相似哈希函数的各位初始化为 0；

将每个文本数据进行分词抽取，抽取出多个分词_权重对；

对每个分词_权重对中的分词进行哈希函数处理；

对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，生成 k 个数值；

将所述生成的 k 个数值转换为 k 位的二进制串。

3、根据权利要求 2 所述的基于大数据的数据去重的方法，其特征在于，所述选择相似哈希函数的位数 k 具体包括：

根据存储成本以及数据集的大小，选择相似哈希函数的位数 k。

4、根据权利要求 2 所述的基于大数据的数据去重的方法，其特征在于，所述对每个分词_权重对中的分词进行哈希函数处理具体包括：

使用 k 位哈希函数计算该每个文本数据的预定数量分词字母的哈希代码。

5、根据权利要求 4 所述的基于大数据的数据去重的方法，其特征在于，所述根据所述预设关键字收集所述至少两个文本数据具体包括：利用网络爬虫技术根据预设的关键字抓取与所述关键字相关的至少两个文本数据。

6、根据权利要求 1-5 任意一项所述的基于大数据的数据去重的方法，其特征在于，所述以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果具体包括：

确定所述 j 份二进制串的每份二进制串的最前端的二进制串与存储器所述样本库存储的最前端的二进制串是否完全相同，如果相同，则确定所述样本库当前反馈的候选结果为正确的候选结果，如果不同，则确定将所述样本库当前反馈的候选结果为不正确的候选结果。

7、根据权利要求 2 所述的基于大数据的数据去重的方法，其特征在于，所述对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，生成 k 个数值具体包括：

对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，如果该位是 1，则加 1，

如果是0，则减1，最后生成k个数值。

8、一种基于大数据的数据去重的装置，其特征在于，包括：

收集模块，用于根据预设关键字收集至少两个文本数据；

处理模块，用于针对每个文本数据，根据相似哈希函数和哈希函数生成k位的二进制串，其中$k=2^n$，其中n为大于等于2的正整数；

分裂模块，用于将该k位的二进制串等分成j份子二进制串，其中j为大于等于1的正整数；

调整模块，调整所述 j 份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应j个集合并存储到预设的样本库中；

匹配模块，用于以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果；

计算模块，用于根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重。

9、根据权利要求8所述的基于大数据的数据去重的装置，其特征在于，所述针对每个文本数据，根据相似哈希函数和哈希函数生成k位的二进制串具体包括：

选择子单元，用于选择相似哈希函数的位数k；

初始化子单元，用于将相似哈希函数的各位初始化为0；

抽取子单元，用于将每个文本数据进行分词抽取，抽取出多个分词_权重对；

哈希函数处理子单元，用于对每个分词_权重对中的分词进行哈希函数处理；

累加子单元，用于对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，生成k个数值；

处理子单元，用于将所述生成的k个数值转换为k位的二进制串。

10、根据权利要求9所述的基于大数据的数据去重的装置，其特征在于，所述选择相似哈希函数的位数k具体包括：

位数选取子单元，用于根据存储成本以及数据集的大小，选择相似哈希函数的位数k。

11、一种计算机设备，包括存储器、处理器，以及存储在所述存储器中并可在所述处理器上运行的计算机可读指令，其特征在于，所述处理器执行所述计算机可读指令时实现如下基于大数据的数据去重的方法的步骤：

根据预设关键字收集至少两个文本数据；

针对每个文本数据，根据相似哈希函数和哈希函数生成k位的二进制串，其中$k=2^n$，其中n为大于等于2的正整数；

将该k位的二进制串等分成j份子二进制串，其中j为大于等于1的正整数；

调整所述 j 份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应j个集合并存储到预设的样本库中；

以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果；

根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重。

12、根据权利要求11所述计算机设备，其特征在于，所述针对每个文本数据，根据相似哈希函数和哈希函数生成k位的二进制串具体包括：

选择相似哈希函数的位数 k；

将相似哈希函数的各位初始化为 0；

将每个文本数据进行分词抽取，抽取出多个分词_权重对；

对每个分词_权重对中的分词进行哈希函数处理；

对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，生成 k 个数值；

将所述生成的 k 个数值转换为 k 位的二进制串。

13、根据权利要求 12 所述的计算机设备，其特征在于，所述选择相似哈希函数的位数 k 具体包括：

根据存储成本以及数据集的大小，选择相似哈希函数的位数 k。

14、根据权利要求 12 所述的计算机设备，其特征在于，所述对每个分词_权重对中的分词进行哈希函数处理具体包括：

使用 k 位哈希函数计算该每个文本数据的预定数量分词字母的哈希代码。

15、根据权利要求 14 所述的计算机设备，其特征在于，所述根据所述预设关键字收集所述至少两个文本数据具体包括：利用网络爬虫技术根据预设的关键字抓取与所述关键字相关的至少两个文本数据。

16、一个或多个存储有计算机可读指令的非易失性可读存储介质，其特征在于，所述计算机可读指令被一个或多个处理器执行时实现如下基于大数据的数据去重的方法的步骤：

根据预设关键字收集至少两个文本数据；

针对每个文本数据，根据相似哈希函数和哈希函数生成 k 位的二进制串，其中 $k=2^n$，其中 n 为大于等于 2 的正整数；

将该 k 位的二进制串等分成 j 份子二进制串，其中 j 为大于等于 1 的正整数；

调整所述 j 份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应 j 个集合并存储到预设的样本库中；

以所述 j 个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果；

根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重。

17、根据权利要求 16 所述的非易失性可读存储介质，其特征在于，所述针对每个文本数据，根据相似哈希函数和哈希函数生成 k 位的二进制串具体包括：

选择相似哈希函数的位数 k；

将相似哈希函数的各位初始化为 0；

将每个文本数据进行分词抽取，抽取出多个分词_权重对；

对每个分词_权重对中的分词进行哈希函数处理；

对经过所述哈希函数处理后的分词_权重对进行位的纵向累加，生成 k 个数值；

将所述生成的 k 个数值转换为 k 位的二进制串。

18、根据权利要求 17 所述的非易失性可读存储介质，其特征在于，所述选择相似哈希函数的位数 k 具体包括：

根据存储成本以及数据集的大小，选择相似哈希函数的位数 k。

19、根据权利要求 17 所述的非易失性可读存储介质，其特征在于，所述对每个分词_权重对中的分词进行哈希函数处理具体包括：

使用 k 位哈希函数计算该每个文本数据的预定数量分词字母的哈希代码。

20、根据权利要求 19 所述的非易失性可读存储介质，其特征在于，所述根据所述预设关键字收集所述至少两个文本数据具体包括：利用网络爬虫技术根据预设的关键字抓取与所述关键字相关的至少两个文本数据。

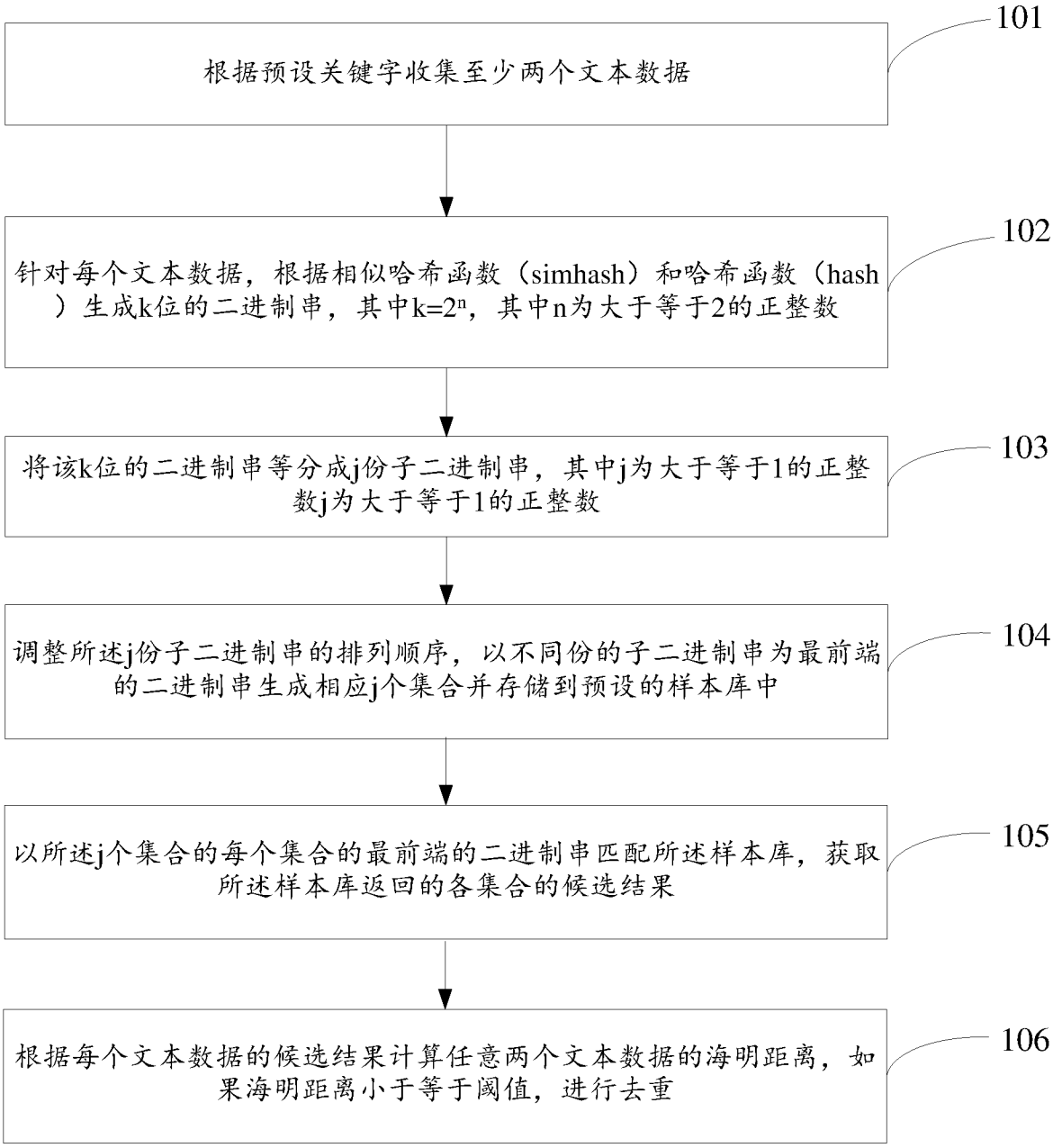
101
根据预设关键字收集至少两个文本数据
102
针对每个文本数据，根据相似哈希函数（simhash）和哈希函数（hash）生成k位的二进制串，其中k=2^n，其中n为大于等于2的正整数
103
将该k位的二进制串等分成j份子二进制串，其中j为大于等于1的正整数j为大于等于1的正整数
104
调整所述j份子二进制串的排列顺序，以不同份的子二进制串为最前端的二进制串生成相应j个集合并存储到预设的样本库中
105
以所述j个集合的每个集合的最前端的二进制串匹配所述样本库，获取所述样本库返回的各集合的候选结果
106
根据每个文本数据的候选结果计算任意两个文本数据的海明距离，如果海明距离小于等于阈值，进行去重

图 1

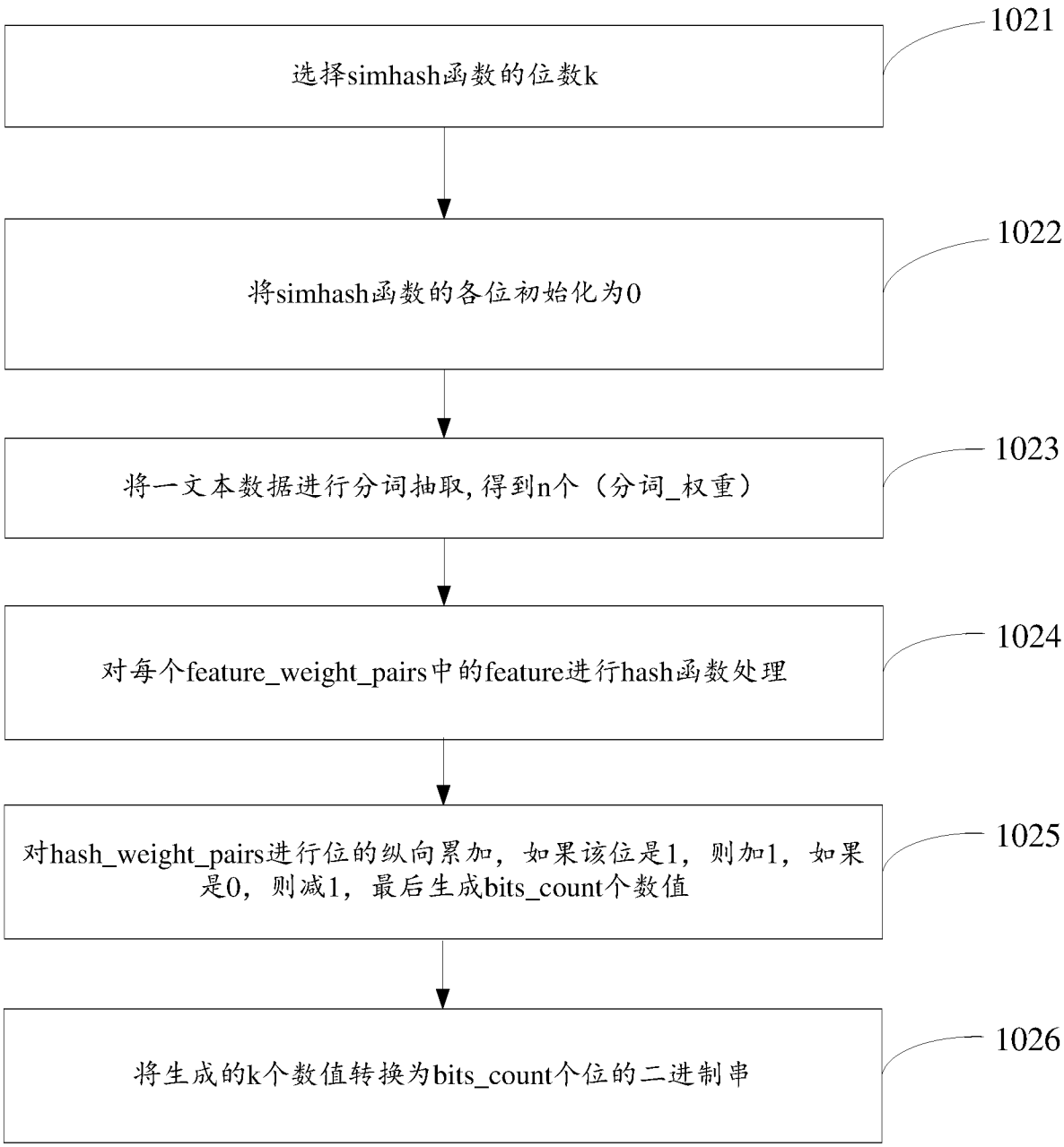
选择simhash函数的位数k
1021
将simhash函数的各位初始化为0
1022
将一文本数据进行分词抽取,得到n个（分词_权重）
1023
对每个feature_weight_pairs中的feature进行hash函数处理
1024
对hash_weight_pairs进行位的纵向累加，如果该位是1，则加1，如果是0，则减1，最后生成bits_count个数值
1025
将生成的k个数值转换为bits_count个位的二进制串
1026

图 2

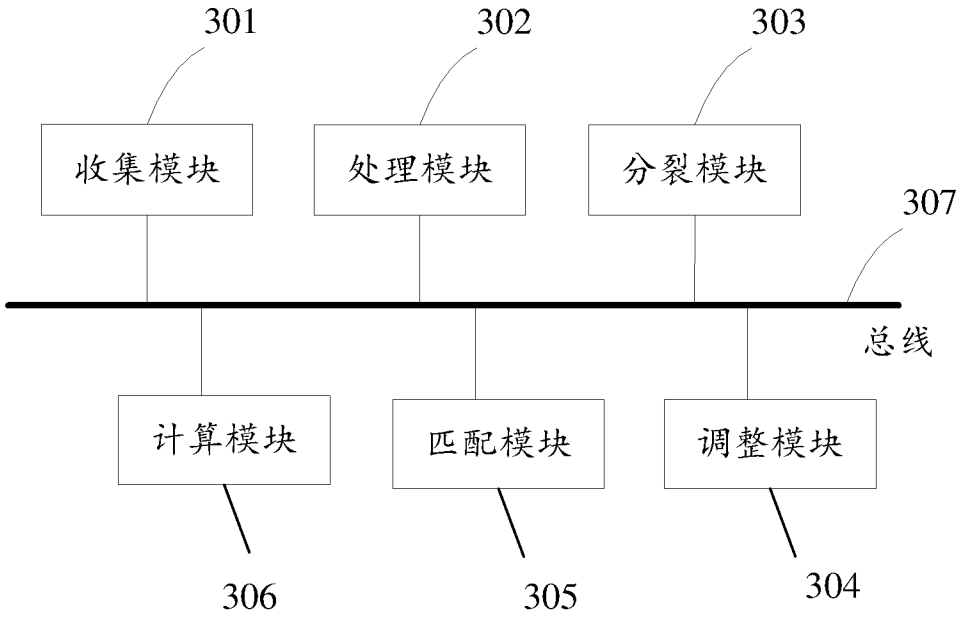


图 3

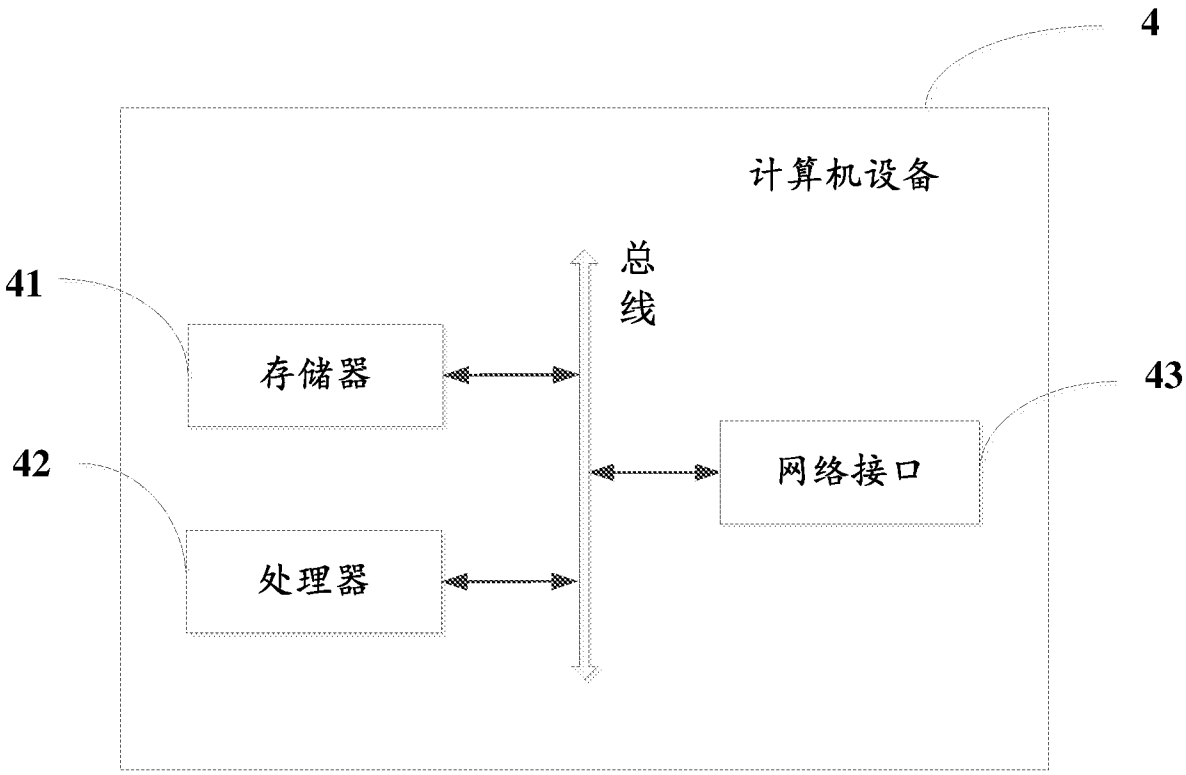


图 4

# INTERNATIONAL SEARCH REPORT

International application No.
**PCT/CN2019/103446**

## A. CLASSIFICATION OF SUBJECT MATTER

G06F 16/31(2019.01)i; G06F 16/33(2019.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WPI, EPODOC, CNKI, CNPAT, IEEE, GOOGLE: 大数据, 去重, 文本, 关键字, 哈希, 串, 海明, 距离, large, big, data, duplication, text, key, word, hash, string, hamming, distance

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| A | CN 108345586 A (CHONGQING YUCUN BIG DATA TECHNOLOGY CO., LTD.) 31 July 2018 (2018-07-31)<br>description, paragraphs 75-112, and figure 1 | 1-20 |
| A | CN 108132929 A (SHANGHAI UNIVERSITY) 08 June 2018 (2018-06-08)<br>entire document | 1-20 |
| A | CN 109271487 A (INSPUR SOFTWARE CO., LTD.) 25 January 2019 (2019-01-25)<br>entire document | 1-20 |
| A | US 2016171009 A1 (INTERNATIONAL BUSINESS MACHINES CORPORATION) 16 June 2016 (2016-06-16)<br>entire document | 1-20 |

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| **10 February 2020** | **02 March 2020** |
| Name and mailing address of the ISA/CN<br>**China National Intellectual Property Administration**<br>**No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088**<br>**China**<br>Facsimile No. **(86-10)62019451** | Authorized officer<br><br>Telephone No. |

**INTERNATIONAL SEARCH REPORT**
**Information on patent family members**

International application No.
**PCT/CN2019/103446**

| Patent document cited in search report | Publication date (day/month/year) | Patent family member(s) | Publication date (day/month/year) |
|---|---|---|---|
| CN 108345586 A | 31 July 2018 | None | |
| CN 108132929 A | 08 June 2018 | None | |
| CN 109271487 A | 25 January 2019 | None | |
| US 2016171009 A1 | 16 June 2016 | US 2018365262 A1 | 20 December 2018 |
| | | CN 105740266 A | 06 July 2016 |

# 国际检索报告

国际申请号
PCT/CN2019/103446

A. 主题的分类

G06F 16/31(2019.01)i; G06F 16/33(2019.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

WPI, EPODOC, CNKI, CNPAT, IEEE, GOOGLE: 大数据, 去重, 文本, 关键字, 哈希, 串,海明, 距离, large, big, data, duplication, text, key, word, hash, string, hamming, distance

C. 相关文件

| 类 型* | 引用文件，必要时，指明相关段落 | 相关的权利要求 |
|---|---|---|
| A | CN 108345586 A (重庆誉存大数据科技有限公司) 2018年 7月 31日 (2018-07-31)<br>说明书第75-112段，附图1 | 1-20 |
| A | CN 108132929 A (上海大学) 2018年 6月 8日 (2018-06-08)<br>全文 | 1-20 |
| A | CN 109271487 A (浪潮软件股份有限公司) 2019年 1月 25日 (2019-01-25)<br>全文 | 1-20 |
| A | US 2016171009 A1 (INTERNATIONAL BUSINESS MACHINES CORPORATION) 2016年 6月 16日 (2016-06-16)<br>全文 | 1-20 |

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

| 国际检索实际完成的日期 | 国际检索报告邮寄日期 |
|---|---|
| 2020年 2月 10日 | 2020年 3月 2日 |

| ISA/CN的名称和邮寄地址 | 受权官员 |
|---|---|
| 中国国家知识产权局(ISA/CN)<br>中国北京市海淀区蓟门桥西土城路6号 100088 | 王艳臣 |
| 传真号 (86-10)62019451 | 电话号码 86-(10)-53961435 |

# 国际检索报告
## 关于同族专利的信息

国际申请号
PCT/CN2019/103446

| 检索报告引用的专利文件 | 公布日<br>(年/月/日) | 同族专利 | 公布日<br>(年/月/日) |
|---|---|---|---|
| CN 108345586 A | 2018年 7月 31日 | 无 | |
| CN 108132929 A | 2018年 6月 8日 | 无 | |
| CN 109271487 A | 2019年 1月 25日 | 无 | |
| US 2016171009 A1 | 2016年 6月 16日 | US 2018365262 A1 | 2018年 12月 20日 |
| | | CN 105740266 A | 2016年 7月 6日 |