一种高性能计算网络下的 TCP 查找哈希算法

张立武[1],冯 宝[1],周建华[2],李 洋[1],茅天奇[3]
(1.南瑞集团有限公司(国网电力科学研究院有限公司),江苏 南京 211000;
2.国网江苏省电力有限公司电力科学研究院,江苏 南京 211103;
3.南京邮电大学 通信与信息工程学院,江苏 南京 210003)

摘 要: 在计算机与通信网络紧密结合的时代,智能电网中的数据处理需要依靠计算机集群来完成,数据传输主要依靠高性能计算网络完成,而高性能计算网络在广域网中主要依赖于 TCP 协议来实现。由于一般的基于哈希表的 TCP 查找算法的性能会在 TCP 会话过多的情况下急剧恶化,且会导致计算机查找 TCP 会话时产生缓存占用过多的情况,因此提出了一种优化后用来支持高性能计算网络中计算机查找大量 TCP 会话的高效的 TCP 查找算法。该算法主要对计算机通过哈希函数生成 TCP 会话的哈希值的方法以及哈希表的数据结构和映射方式两方面进行优化,并实现了一种适合现代计算机体系的数据结构。为了验证该算法的性能,在 Intel 多核处理器上进行了并行化堆栈。实验表明,该算法减少了大量 TCP 会话情况下计算机的 TCP 会话查找时间和占用的计算机缓存大小,并能在并行平台上同时处理百万级个会话。

关键词: TCP 查找; 哈希表; 高性能计算网络; 数据结构
中图分类号: TP301.6 **文献标识码:** A **文章编号:** 1673-629X(2018)05-0094-05
doi: 10.3969/j.issn.1673-629X.2018.05.022

A High Performance Computing Network TCP Optimization Algorithm Based on Hash Table

ZHANG Li-wu[1],FENG Bao[1],ZHOU Jian-hua[2],LI Yang[1],MAO Tian-qi[3]
(1.NARI Group Corporation Co.,Ltd.,(State Grid Electric Power Research Institute Co.,Ltd.),
Nanjing 211000,China;
2.Electric Power Research Institute of State Grid Jiangsu Electric Power Co.Ltd.,Nanjing 211103,China;
3.School of Communication and Information Engineering,Nanjing University of Posts and
Telecommunications,Nanjing 210003,China)

Abstract: In the era of the close connection between computer and the communication network,the data processing in the smart grid needs to be done by computer clusters,the data transmission is mainly done by the high performance computing network,and the high performance computing network mainly depends on TCP in the wide area network.As the performance of the typical hash based TCP lookup algorithm is drastically deteriorated in the case of too many TCP sessions,it will contribute to the situation where too much cache of the computers is occupied by TCP sessions.Therefore,we present a high-efficient TCP lookup algorithm that aims at supporting large number of sessions in high performance computing network.It mainly optimizes the TCP session in both the way of generating the hash value and the data structure and mapping mode of the hash table that well fits the modern computer architectures well.In order to verify the performance of the algorithm,we parallelize the stack on the Intel multi core processors.Experiments show that the algorithm reduces the TCP session lookup time and the cache size of a computer in the situation of a large number of TCP sessions,and can handle millions of conversations on a parallel platform at the same time.

Key words: TCP lookup; hash table; high performance computing network; data structure

0 引 言

随着计算机技术和电力行业的迅猛发展,智能电网与高性能计算机集群的联系越来越密切。由于电力系统本身对数据实时性及计算性能的高要求,智能电

收稿日期: 2017-04-14 **修回日期:** 2017-08-25 **网络出版时间:** 2017-12-05
基金项目: 国家自然科学基金(61471203);教育部博士点基金资助项目(20133223120002);国家电网公司 2016 年科技项目
作者简介: 张立武(1974-),男,工程师,从事电力系统通信技术研究工作。
网络出版地址: http://kns.cnki.net/kcms/detail/61.1450.TP.20171205.1429.074.html

网中的数据处理主要由计算机集群提供的高性能计算服务完成,其中的数据传输主要由计算机集群之间的高性能计算网络完成,高性能计算网络就是指能够在集群主机之间提供极高通信性能的网络[1]。由于通信机制越来越难设计,所以通信往往成为开发的瓶颈,如何使高性能计算机集群运行得更快、更高效一直是研究的热点[2]。事实上,高性能计算已被公认为是继理论科学和实验科学之后,人类认识世界改造世界的第三大科学研究方法,是科技创新的重要手段[3]。因此这就对计算机集群的计算性能提出了更高要求。目前,高性能计算网络在广域网中主要依赖于 TCP 来实现[4],但随着高性能计算网络中 TCP 会话数量的快速增长,传统 TCP 会话的查找算法很难在查找速率和查找时所占用的计算机处理器的缓存大小之间达到平衡。TCB(传输控制块)是一种用于维持每个 TCP 会话状态的数据结构。一般来说,TCB 的大小为 280~1 300字节。在如今的高性能计算网络中,TCP 会话的数量一般可以达到一百万个,也就是说 TCB 将占用 280 MB~1.3 GB 的储存空间,而主流的计算机处理器的最后一级高速缓存(LLC)的规模通常为 10 MB,这就使得 TCB 的存储器占用空间是 LLC 的大小的几十万倍。对于这种巨大的工作量,几乎每个沿链表的搜索步骤都会导致查找 TCP 会话时计算机的缓存不够。已有研究[5-7]表明,TCP 会话的查找时间主要是由主存储器访问的 CPU 性能决定的,而不是由指令的执行时间决定的。因此,即使只是次要的存储器访问也会显著增加 TCP 查找的时间,也就是说,传统的 TCP 查找算法中哈希表的数据结构已经不能满足查找高性能计算网络中大量 TCP 会话的要求,且会对计算机的处理器缓存造成极大负担。为了解决这些问题,必须提出一种适合现代计算机处理器的 TCP 会话的哈希查找算法。

1　技术背景

哈希表是在当前 TCP 过程中计算机查找 TCB 最广泛使用的方法。当 TCP 会话到达时,计算机按照哈希函数将 TCP 会话标识符映射为哈希值,然后使用哈希值定位哈希桶,最后在发生哈希冲突时对冲突链表进行搜索。只有当装填因子较低时,哈希表才有较好的性能。在数百万个 TCP 会话并行的情况下,如果装填因子仍要保持较低水平,则 TCB 哈希表将变得特别大,这就会占用计算机极大的存储空间,然而,限制哈希表大小又会大大增加发生哈希冲突的可能性,这就使哈希表的优点不再存在。

文献[8]介绍了目前的算法会导致计算机查找 TCP 会话时因 TCP 会话过多而产生性能急剧恶化的现象,这是因为哈希表的大小与会话数量成比例增长会导致计算机占用的内存空间增加。此外,由于对 TCB 的访问缺乏规律,当有大量的 TCP 会话需要处理时,增加计算机缓存大小并不能带来较大的性能优化。为了改善哈希表的查找瓶颈,文献[9-13]提出了一些优化方案来减少计算机的查找时间和内存占用,然而它们都不能适应高性能计算网络的要求,特别是不能适应在实际系统中对高性能网络的缓存有效使用。

近年来也有许多采用硬件来优化 TCB 查找的方案。文献[14]中利用网络会话的特点设计出了服务器专用的 TCB 缓存硬件。文献[15]设计了一种复杂的函数,该函数将会话签名和 TCB 位置转换为 32 位代码,存储在专用 TCB 缓存硬件中。由于资源有限,上述解决方案都只能用来处理少量 TCP 会话,且它们的扩展成本非常高,这无法满足高性能计算网络中百万数量级的 TCP 会话的要求。除了性能和价格这对矛盾之外,硬件解决方案还需要修改现有的网络栈结构,这是很难实现的。

2　一种基于哈希表的 TCP 查找优化算法

2.1　签名算法

文中提出了一种基于哈希查找的签名算法,它不再使用 TCP 会话的 4 元组,即源 IP 地址、目的地 IP 地址、源端口、目的端口来生成哈希值,而是使用 32 位的短签名来标记 TCP 会话。由于不需要将完整的 TCB 标识符存储在哈希表中,而只需要存储短签名,哈希表的大小大大减少,查找时需要的缓存也大大减少。签名算法的主要作用是数据压缩,这就有可能产生匹配冲突的现象,即不同的 TCP 会话恰好具有相同的签名。因此,每当在哈希表中匹配到对应的 TCP 会话的签名时,应访问相应的 TCB,并比较 4 元组,对实际匹配的 TCP 会话进行确认。由于签名匹配出错会导致额外的存储器访问,低匹配出错率是签名算法最重要的特性。另外,签名算法必须对每个 TCP 会话执行,故其计算次数不能太多。

文中采用一种较为简单的签名算法,对第一个 TCP 会话,简单地对其 4 元组执行异或来获得签名,即计算源 IP 地址⊕目的地 IP 地址⊕源端口⊕目的端口来得到签名,而对于之后的 TCP 会话,则将该 TCP 会话与前一个 TCP 会话按上述步骤得到的两个 32 位数进行异或,以作为该 TCP 会话的短签名。

2.2　数据结构

图 1 描述了优化后的 TCB 查找数据结构,一般用 TCP 会话标识符作为哈希函数的输入,每个哈希桶包含 16 个槽,每个槽为 32 位长。在对 TCB 阵列预分配时,引入参数 N 表示哈希表中的槽的数量。前 N 个会

话与每个槽一一映射,剩余元素则被保留在 *TCB* 池中用于下次分配。当冲突的 TCB 的数量大于哈希桶中的最大的槽的数量时,这些冲突的 TCB 签名将被分配到 TCB 池中,它们的位置与签名一起明确地存储在相应的冲突列表中。另外,在这种数据结构下,TCB 位置并不是明确地存储在查找表中,而是通过它们对应的槽的位置计算得出。TCB 会话与槽的映射关系是将阵列中的序号为 $(i-1)\times 16+j$ 的 TCB 会话映射到第 i 个哈希桶中的第 j 个槽。

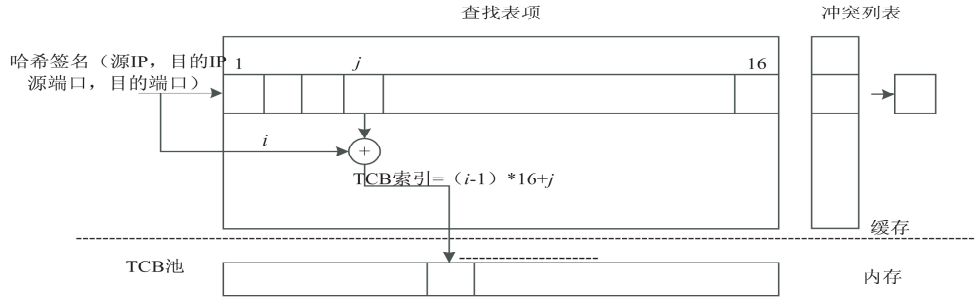

图 1 优化后的哈希算法的数据结构

2.3 算法描述

(1) 查找会话: 如图 2 所示,当接收到 TCP 分组时,用 TCP 的会话标识符计算该 TCP 会话的 TCP 签名。从第一个槽开始向桶的末端进行搜索签名匹配时,访问相应的 TCB,并且进一步比较完整的 4 元组。如果发现误判,则继续进行搜索。若在哈希桶中找不到匹配,则检查相应的冲突列表,这个过程的最终返回值为相应的 TCB 位置或 NOT FOUND。

(2) 添加会话: 添加会话的过程与查找会话类似,其区别在于添加会话是要在哈希桶找到一个空的槽。当在哈希桶中找到空槽时,新的会话签名被存储在其中,并且返回与之对应的 TCB。如果在哈希桶中没有找到空槽,则将包含会话签名和 TCB 位置的新元素添加到冲突列表。

(3) 删除会话: 当会话关闭时,需要清除 TCB 签名。若能在表中找到匹配的 TCB 签名,则可以通过将该槽清零来删除该会话。如果在冲突列表中找到该 TCB 会话签名,则首先将该 TCB 放回到 TCB 池中,然后再删除冲突列表中的 TCB。

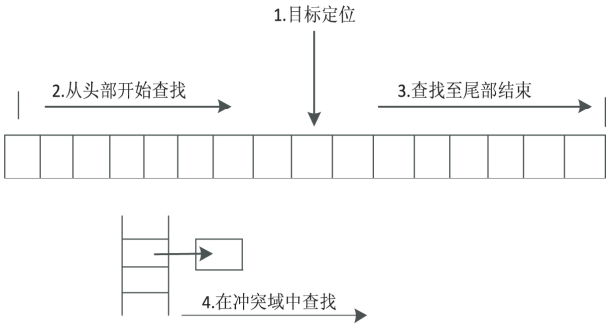

图 2 查找会话的过程

文中提出的哈希表的结构可等效为一种两级哈希表结构,其中第一级表含有 N 个哈希桶,每个第二级表含有 2^b 个哈希桶。在采用优化后的基于哈希算法的 TCP 查找算法时,采取签名算法来生成作为哈希索引的 b 个比特的 TCB 签名的第二级哈希函数。然而,在优化算法下哈希索引并不用于定位哈希桶,而是通过记录哈希索引来标识对象,这能很好地解决与开放寻址法的冲突。因此,在优化后的算法中,每个桶的 TCB 签名的误判率等于第二级哈希表的冲突率。

2.3.1 装填因子

哈希表的性能很大程度上取决于哈希表的装填因子。文中研究的优化后的 TCB 查找算法等效哈希表见图 3 中表 B,两者的装填因子相等 。图中 N 表示哈希表中的哈希桶数,b 表示 TCB 签名的 32 位位数。

当哈希表均匀装填了 M 个 TCP 会话时,装填因子可以计算为 $(M/N)\times(1/2^b)$ 。在该优化算法中,设定 $b=32$,且一般 M/N 不超过 16,则有:

$$(M/N)\times(1/2^b)\leqslant 3.72\times 10^{-9} \quad (1)$$

由此可见,优化后算法实现了一种具有极地装填因子的哈希表结构,这种结构大大减少了哈希表占用的存储空间。例如,当有 1 000 000 个 TCP 会话到达且装填因子为 3.72×10^{-9} 时,传统的哈希表在 64 位系统中需要占用 2 000 TB 的容量,而优化后的算法仅仅需要占用 4.5 MB 的容量。

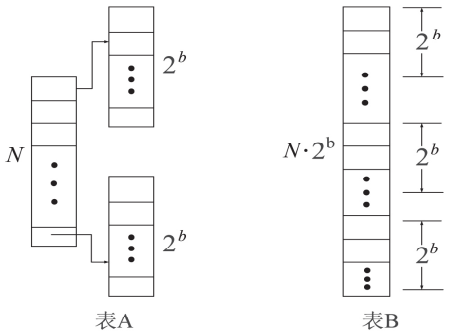

图 3 优化后哈希算法的识别率和装填因子

2.3.2 误判率

如前所述,优化后算法中每个哈希桶的误判率等于图 3 中表 A 中第二级哈希表的冲突率。表 A 中每

个第二级哈希表含有 $n = 2^{32}$ 个桶，包含在该表中 TCP 会话的签名的平均数量不大于 16。假设在第二级哈希表中均匀存储了 k 个 TCP 会话签名，并且定义 E_k 为 k 个会话在表中不冲突的事件，则其概率为:

$$\Pr\{E_k\} = 1 \cdot \left(\frac{n-1}{n}\right)\left(\frac{n-2}{n}\right) \cdots \left(\frac{n-k+1}{n}\right) = 1 \cdot \left(1-\frac{1}{n}\right)\left(1-\frac{2}{n}\right) \cdots \left(1-\frac{k-1}{n}\right) \quad (2)$$

在 $n = 2^{32}$ 和 $k \leqslant 16$ 的情况下，项 $-\frac{1}{n}, -\frac{2}{n}, \cdots, -\frac{k-1}{n}$ 可以被认为是无穷小的，并且 e^x 无限接近于 $1+x$，即 $e^x = 1 + x + o(x^2)$。因此有:

$$\Pr\{E_k\} \approx e^{-\frac{1}{n}} e^{-\frac{2}{n}} \cdots e^{-\frac{k-1}{n}} = e^{-\sum_{i=1}^{k-1} i/n} = e^{-k/(k-1)/2n} \approx 1 - k(k-1)/2n \quad (3)$$

根据概率知识可知，至少两个会话冲突的概率等于 1 减去没有会话冲突的概率，从而在优化后的算法中每个哈希桶的预期误判率为:

$$1 - \Pr\{E_k\} \approx k(k-1)/2n \leqslant 2.79 \times 10^{-8} \quad (4)$$

也就是说，与传统的哈希表相比，优化后的哈希算法具有很低的误判率和装填因子，且占用了更少的计算机内存空间。

3　仿真结果与分析

3.1　单核性能

在开源 TCP/IP 协议栈中，文中使用优化后的 TCB 查找算法代替了原来的 TCP 查找协议，并对查找性能进行了评估。

生成了三种不同大小的跟踪文件，如表 1 所示。

表 1　跟踪文件的特性

文件	数据包总数	数据包平均大小/B	最大 TCP 会话数	平均 TCP 会话数
文件 1	5 242 760	67	263 252	248 128
文件 2	10 394 870	67	523 377	497 986
文件 3	21 862 430	67	1 048	987 290

针对三种不同的跟踪文件，将优化后的算法与原来的算法进行了对比。优化后的哈希算法的平均查找时间如图 4 所示，其中原始算法(N 桶- M 寄存器) 表示以 N 个区段占用了 M 个存储器空间的原始算法。

仿真结果表明，原始算法的性能受 TCP 会话数的变化而显著变化，而优化后算法的表现非常稳定，更加可靠。

3.2　并行性能

不同哈希表大小的优化算法和传统 TCP 查找算法的性能对比如图 5 所示。

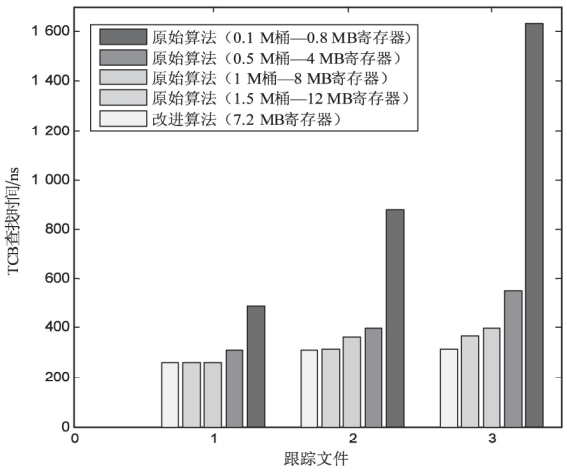

图 4　TCB 查找性能

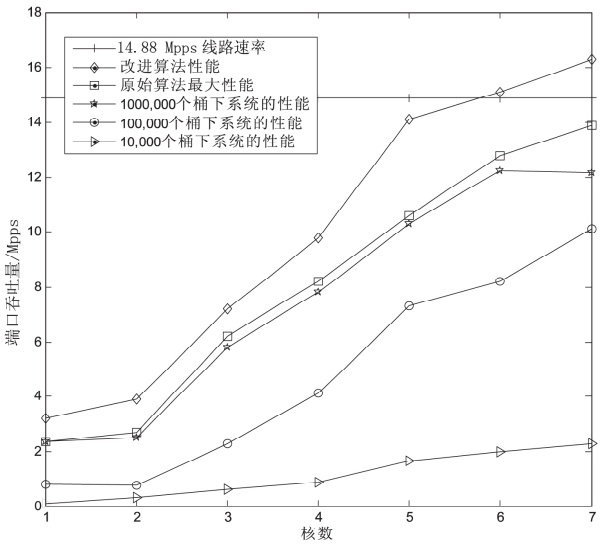

图 5　优化算法与原始 TCP 查找算法的性能对比

设置 150 万桶为阈值，在此情况下采用文中签名算法，仅有 11 257 个 TCP 会话的签名发生冲突，另一方面，在使用传统算法时，系统性能会随着哈希表大小的增加而增加，直到达到阈值。而在 10 Gbps 的以太网下，采用 150 万个哈希桶(约 12 MB) 的传统算法的端口吞吐量不能高于 14.88 Mpps，而优化后的算法能在同等条件下用六个内核实现 15.2 Mpps 的吞吐量，在七个内核上实现 16.5 Mpps 的吞吐量，且其查找表和冲突列表只占用 7.5 MB 的内存空间。显然，优化后的哈希表查找算法的系统性能更加优越。

4　结束语

哈希表是目前计算机查找 TCP 会话中最广泛使用的方法，但其并不能很好地处理国家电网中广域高性能计算网络中大量的 TCP 会话，且会导致计算机查找 TCP 会话的时间和占用的内存较大。为了解决这一问题，对传统的哈希查找算法进行了优化。首先设计了一种签名算法用查找表中的较短的会话签名来替换完整的 TCB 标识符，减少了计算机的内存占用。其

次使用顺序访问替代随机访问,以增加计算机的存储器访问效率,降低了误判率。仿真结果表明,优化后的哈希查找算法的系统性能更加优越。

参考文献:

[1] 刘 颖,陈 煜,林 林,等.高性能计算集群中的网络技术研究与实践[J].中国水利水电科学研究院学报,2016,14(2):90-95.

[2] 岳菲菲,王海军,王 新,等.高性能计算通信机制分析与研究[J].计算机工程与科学,2009,31(A1):27-30.

[3] 李根国,桂亚东,刘 欣.浅谈高性能计算的地位及应用[J].计算机应用与软件,2006,23(9):3-4.

[4] 熊 兵,李 峰,姜腊林,等.面向高速网络连接记录管理的高效哈希表[J].华中科技大学学报:自然科学版,2011,39(2):19-22.

[5] CLARK D D,JACOBSON V,ROMKEY J,et al.An analysis of TCP processing overhead[J].IEEE Communications Magazine,2002,40(5):94-101.

[6] CHIANG M L,LI Y C.LyraNET:a zero-copy TCP/IP protocol stack for embedded systems[J].Real-Time Systems,2006,34(1):5-18.

[7] LI Z,MAKINENI S,ILLIKKAL R,et al.Efficient caching techniques for server network acceleration[C]//Advanced networking & communications hardware.[s.l.]:[s.n.],2004.

[8] MAKINENI S,BHUYAN L.TCP/IP cache characterization in commercial server workloads[C]//Proceedings of seventh workshop on computer architecture evaluation using commercial workloads.[s.l.]:[s.n.],2004.

[9] 马如林,蒋 华,张庆霞.一种哈希表快速查找的改进方法[J].计算机工程与科学,2008,30(9):66-68.

[10] 王 果,徐仁佐.结合哈希过滤的一种改进多连接查询优化算法[J].计算机工程,2004,30(7):57-59.

[11] SONG H,DHARMAPURIKAR S,TURNER J,et al.Fast hash table lookup using extended bloom filter:an aid to network processing[C]//Proceedings of the 2005 conference on applications,technologies,architectures,and protocols for computer communications.New York,NY,USA:ACM,2005:181-192.

[12] KUMAR S,CROWLEY P.Segmented hash:an efficient hash table implementation for high performance networking subsystems[C]//Proceedings of 2005 ACM symposium on architecture for networking and communications systems.New York,NY,USA:ACM,2005:91-103.

[13] HASAN J,CADAMBI S,JAKKULA V,et al.Chisel:a storage efficient,collision-free hash-based network processing architecture[C]//Proceedings of the 33rd annual international symposium on computer architecture.Washington,DC,USA:IEEE Computer Society,2006:203-215.

[14] LIAO G,BHUYAN L N,WU W,et al.A new TCB cache to efficiently manage TCP sessions for web servers[C]//ACM/IEEE symposium on architectures for networking and communications systems,New York,NY,USA:ACM,2010.

[15] PONG F.Fast and robust TCP session lookup by digest hash[C]//Proceedings of the 12rd IEEE international conference on parallel and distributed systems.[s.l.]:IEEE,2006.

(上接第 93 页)

数据的局限性,只研究了微博中的诈骗团体,对于其他平台的和沟通工具的诈骗团体有待进一步挖掘;其次,采用结巴分词进行断词,产生了大量的数据集,影响了运行效率,因此提高该算法的效率是后续的研究方向。

参考文献:

[1] 孙 孟.微博营销-新媒体时代的营销宠儿[J].通信企业管理,2011(7):38-39.

[2] 吴继飞,邓安平.基于互联网时代微博营销的 SWOT 分析[J].中国集体经济,2011,21:52-53.

[3] 王 利.基于数据挖掘技术的微博营销系统的设计与实现[D].武汉:华中科技大学,2013.

[4] 邵 笑.新媒体诈骗的言语行为研究[D].锦州:渤海大学,2014.

[5] 张劭捷.基于微博社交网络的舆情分析模型及实现[D].广州:华南理工大学,2011.

[6] 缪茹一.基于文本数据挖掘的微博情感分析与监控系统[D].杭州:浙江工业大学,2015.

[7] ZHOU X,CHEN L.Event detection over twitter social media streams[J].VLDB Journal,2014,23(3):381-400.

[8] 康泽东,余旌胡,丁义明.微博社交网络的对称程度实证分析[J].计算机应用,2014,34(12):3405-3408.

[9] FARINE D R,WHITEHEAD H.Constructing,conducting and interpreting animal social network analysis[J].Journal of Animal Ecology,2015,84(5):1144-1163.

[10] 孙怡帆,李 赛.基于相似度的微博社交网络的社区发现方法[J].计算机研究与发展,2014,51(12):2797-2807.

[11] 范超然,黄曙光,李永成.微博社交网络社区发现方法研究[J].微型机与应用,2013,31(23):67-70.

[12] NASON G J,FARDOD O,KELLY M E,et al.The emerging use of Twitter by urological journals[J].Bju International,2015,115(3):486-490.

[13] CHEN P,FU X,TENG S,et al.Research on micro-blog sentiment polarity classification based on SVM[C]//International conference on human centered computing.[s.l.]:Springer International Publishing,2014:392-404.

[14] FLEUREN WW M,ALKEMA W.Application of text mining in the biomedical domain[J].Methods,2015,74:97-106.

[15] IRFAN R,KING C K,GRAGES D,et al.A survey on text mining in social networks[J].Knowledge Engineering Review,2015,30(2):157-170.