基于分布式倒排索引和VSM算法的P2P复杂搜索

李 想,吴国新,郭 晶
(东南大学 计算机网络与信息集成教育部重点实验室,江苏 南京 210018)

摘 要:传统的基于DHT的结构化P2P系统有一定的局限性,如不支持多特征词的复杂搜索,无法对搜索结果进行排序等。通过改进的分布式倒排索引,支持多特征词的复杂搜索,并极大改善了传统的倒排索引技术引起的网络流量消耗;通过改进的VSM算法,对搜索结果进行排序;提出了新的资源发布算法。

关键词:结构化;对等网;倒排索引;向量空间模型
中图分类号:TP393 **文献标识码**:A **文章编号**:1673-629X(2009)04-0025-03

Distributed Inverted Index and VSM Algorithm Based Complex Peer-to-Peer Search

LI Xiang, WU Guo-xin, GUO Jing
(Ministry of Education Key Lab. of Computer Network and Information Integration, Southeast University, Nanjing 210018, China)

Abstract: The traditional DHT-based structured peer-to-peer system has its limitation. It does not support multi-keywords-based complex search and cannot sort the searching results. By improving the distributed inverted index algorithm, to support multi-keywords-based complex search, and to address the problem of huge network traffic consumption caused by traditional inverted index. By improving VSM algorithm, to sort the searching result. Present a new resource-publishing algorithm.

Key words: structured; peer-to-peer; inverted index; vector space model

0 引 言

P2P技术由于其固有的优点,如容错性好、扩展灵活和自治性等,被广泛应用于资源共享系统中。一般来说,P2P系统可以分为以Gnutella等为代表的无结构化P2P,以及Chord[1]、Pastry[2]等为代表的结构化P2P。无结构化P2P的覆盖网以松散的方式连接,节点加入退出简单,维护成本低。但是无结构化P2P中的查询多采用泛洪的方式,效率低下,并且无法保证查询结果的准确性。基于DHT(Distribute Hash Table)的结构化P2P有良好的可扩展性和查询效率,是当前的主流P2P系统。

DHT-P2P是基于单特征词进行搜索的,然而,在实际的应用中,人们经常发现使用单特征词无法准确描述要搜索的目标,只能通过几个方面进行描述。同时,当前的P2P系统多不支持对搜索结果的排序,给用户的使用带来不便。为了更好地完成搜索功能,DHT-P2P需要解决和完善如下两个问题:1)如何根据多个特征词进行搜索;2)对搜索结果进行有效的排序,使更符合用户要求的搜索结果排序靠前。

文中在DHT-P2P中引入分布式倒排索引和VSM(Vector Space Model)算法尝试解决上述问题。通过改进的分布式倒排索引,支持基于多特征词的搜索;通过改进的VSM算法,实现了对搜索结果的排序。

收稿日期:2008-07-28
基金项目:国家高技术研究发展计划(863计划)(2007AA01Z422)
作者简介:李 想(1985-),男,江苏宿迁人,硕士研究生,研究方向为结构化P2P网络、P2P网络信用管理;吴国新,教授,博士生导师,研究方向为企业信息化关键支撑技术、网络协议及标准、P2P网络信用管理。

1 分布式倒排索引及改进

1.1 分布式倒排索引

按特征词建立的索引称为倒排索引,是目前搜索引擎中最常用的存储方式。假设doc1和doc2表示两个不同的文档,doc1中包含特征词c1和c2,doc2中包含特征词c2和c3,对应的倒排索引如下:

c1:doc1
c2:doc1,doc2
c3:doc2

将倒排索引机制应用于DHT-P2P系统时,可以

根据文档的特征词进行文档的发布，对应节点维护相关特征词的倒排索引。覆盖网中所有节点上倒排索引的集合构成一个分布式的倒排索引。

1.2 改进的分布式倒排索引

文献[3]介绍了一种典型的利用传统的分布式倒排索引进行多特征词搜索的算法。该算法在进行搜索时，需要在网络上传输搜索的中间结果，存在带宽浪费的问题。为了解决这个问题，文献[3]中介绍了很多传输压缩算法，但是这些都不能从根本上解决问题。文献[4]讨论了文献[3]中相关算法的实现问题。

笔者等对分布式倒排索引和相应的算法进行了改进，通过扩充倒排索引的存储信息，避免了搜索算法产生的中间结果在网络上的传输。首先对分布式倒排索引中的每个文档项包含的信息进行如下扩充：

c1:doc1($n_1, n_2, n_3, \cdots, n_m$)

c2:doc1($n_1, n_2, n_3, \cdots, n_m$) doc2($n_1, n_2, n_3, \cdots, n_m$)

c3:doc2($n_1, n_2, n_3, \cdots, n_m$)

文档项中的 n_i 表示特征词 c_i 在该文档中出现的频率。若特征词 c_i 没有在某文档中出现过，则该文档项中的 n_i 取值为 0。通过扩充，每个文档项中都包含了所有特征词的频率信息，避免了搜索过程中此类信息的传输，可以极大地节省网络带宽。

1.3 可行性分析

改进后的分布式倒排索引需要更多的空间存储词频信息。假设在一个大型 P2P 系统中，有 6 万条左右的特征词，平均每个节点存储 1 万个文档项的信息，每个文档项纪录了所有特征词在该文档中出现的频率，每个出现频率用 2 个字节表示，则平均每个节点大约需要 10 000 * 60 000 * 2/1024 * 1024 * 1024≈1.2GB 的空间来存储相关信息，这样的存储空间要求在一台普通的微机上完全可以得到满足。同时，可以设计一个轻量级的协议用于特征词的扩充。

2 搜索与排序算法

2.1 TFIDF - VSM

VSM 是信息检索领域经典的统计算法，它将描述文档的多个特征词以向量的形式来表示，向量中的每个分量对应了文档的不同特征。用户进行文档搜索时，将用户描述文档的特征词也用向量表示，通过比较两个向量的相似程度来进行文档的匹配。文献[5]讨论了如何利用 VSM 算法和本体论构建支持复杂搜索的 P2P 系统，但是没有解决排序问题。

假设整个 P2P 网络中包含的文档集合为 D，D 由 N 个不同的文档构成，即 $D=\{d_1, d_2, \cdots, d_n\}$。$D$ 中所有文档包含的特征词集合构成 C，定义 $C=\{c_1, c_2, \cdots, c_m\}$，即总共有 m 个特征词。则文档 $d_i \in D$ 可以表示为 $d_i=\{n_{i1}, n_{i2}, \cdots, n_{im}\}$，其中 n_{ij} 是文档 d_i 包含的第 j 个特征词的数量。

计算 n_{ij} 的值有很多种方法，最简单的是二值表示法，即如果文档 d_i 包含第 j 个特征词则 n_{ij} 取 1，否则 n_{ij} 取 0。这种表示方法不能准确地表达不同特征词对文档的重要程度。TFIDF(Term Frequency Inverse Document Frequency)技术可以比较好地解决上述问题。TF 表示特征词在文档中出现的次数，DF 表示文档集中出现该特征词的文档的总数量。TF 除以 DF 即是 TFIDF 的值，直观看来，在很多文档中都会出现的特征词没有只在少量文档中出现的特征词的区分作用大。采用 TFIDF 计算 n_{ij} 的值，可以使用公式(1)：

$$n_{i_j} = f_{i_j} * \log(N/M_j + 0.5) \tag{1}$$

f_{i_j} 是第 j 个特征词在文档 d_i 中出现的次数(即 TF)，N 是文档集中文档的总数，M_j 是包含第 j 个特征词的文档数量(即 DF)。

根据用户输入的特征词提取出相应的特征向量 $i=(i_1, i_2, \cdots, i_m)$，如果用户输入的特征词中包含第 j 个特征词，则 i_j 取 1，否则取 0。将该特征向量与文档向量进行点乘，即可得到二者的相似度。如公式(2)：

$$r_t = \sum_{k=1}^{m} n_{tk} * i_k \tag{2}$$

相似度(r_t)越高，则该文档与特征词的关联度越高，越符合用户的要求。

根据 OneStat.com 的调查，在实际应用中，使用 4 个及 4 个以下特征词进行的检索占到了总检索数的 86% 以上。因此向量中绝大多数分量的值为 0。利用这个特点，可以对公式(2)进行优化，首先提取出向量 i 中值不为 0 的项，将它们出现的位置保存在数组 $V[x]$ 中，在 86% 以上的情况下，x 的值小于等于 4。对公式(2)优化后得到公式(3)：

$$r_t = \sum_{k=1}^{x} n_{tv}[k] \tag{3}$$

采用改进后的分布式倒排索引和 TFIDF - VSM 算法后，搜索算法如下：根据输入的特征词，提取特征向量，根据特征词中任意一个定位到维护该特征词的节点，该节点检索倒排索引选取包含此特征词的文档项，使用 TFIDF - VSM 算法对选取的文档项进行排序，并返回结果。

2.2 算法的改进

在进行基于多特征词的搜索时，上述算法可能导致没有包含所有特征词的文档与特征词的相似度高于包含所有特征词的文档，使搜索结果不符合用户的预

期,如下例所示:

表 1 统计了三个特征词在三个文档中的分布情况,假设用户输入特征词"分布式"和"网络",抽象后得到特征向量 $i=(1,1,0)$,则计算出文档 $D1$ 与特征词相似度:

$r_1 = 1 * 5 * \log(1000/20 + 0.5) + 1 * 0 * \log(1000/100 + 0.5) + 0 * 10 * \log(1000/500 + 0.5) = 8.52$

同理可得 $r_2 = 2.72$, $r_3 = 5.47$。

因此排序的结果是 $D1, D3, D2$,事实上 $D1$ 并没有包含特征词"网络"。排序显然不正确。

表 1　特征词在文档中的分布

文档名	特征词 1(TF/DF)	特征词 2(TF/DF)	特征词 3(TF/DF)	总文档数
$D1$	分布式(5/20)	网络(0/100)	计算机(10/500)	1000
$D2$	分布式(1/20)	网络(1/100)	计算机(50/500)	1000
$D3$	分布式(2/20)	网络(2/100)	计算机(80/500)	1000

文中提出了一种改进的算法以解决这个问题。对基于多特征词的搜索,先通过基于二值的 VSM 算法进行初始排序,然后使用基于 TFIDF 的 VSM 算法进行二次排序。

二值 VSM 算法即在计算文档的特征向量时,如果文档包含相应的特征向量则取 1,否则取 0。计算相似度时采用以前同样的算法,此时得到的相似度即是文档中包含的不同的被检索特征词的数目。在上述例子中,使用二值的 VSM 计算出的三个相似度分别为:$r_1=1, r_2=2, r_3=2$。对相似度等于检索特征词数目的文档项进行二次排序。在上述例子中,则仅对 $D2$ 和 $D3$ 进行二次排序,因为 r_1 的值不为 2,所以二次排序的时候,其不用参与。因为仅对相似度等于检索特征词数目的文档进行二次排序,这将过滤掉大部分的文档,因此改进算法使用的总时间总是小于原有算法的 2 倍。改进后的算法流程如图 1 所示。

2.3　资源发布算法

使用 TFIDF 时使用到的数据中有一些全局的统计数据,如文档的总数,出现某特征词的文档数,如何得到和维护这些数据是一个需要解决的问题。

文中提出了一个新的通用的资源发布算法用于对上述数据的维护。典型的 DHT - P2P,如 Chord[1],Pastry[2]等资源发布算法的跳数复杂度都是 $\log_n$,本算法的跳数复杂度为 n,但是考虑到 P2P 网络中查询操作远多于发布操作,因此带来的性能损失是有限的。算法描述如图 2 所示。

3　结束语

对基于多特征词的 P2P 复杂搜索进行了研究。

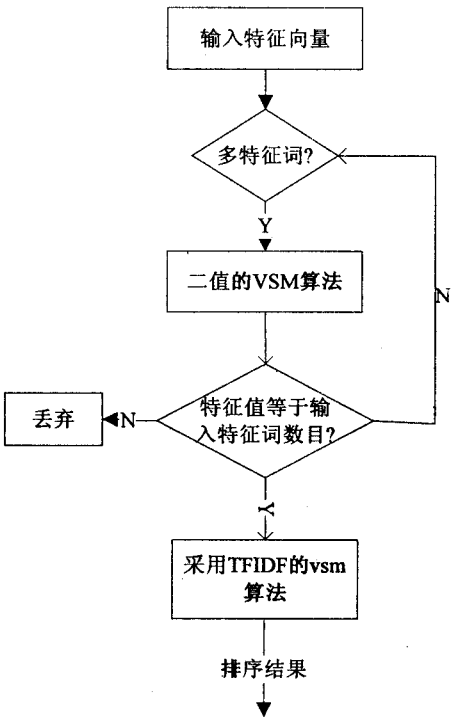

图 1　改进后的算法流程

d[m] //表示一个文档,d[i]表示文档 d 中包含的第 i 个特征词的数量
N //p2p 网络中文档总数,每个节点上都维护一个相同的 N 值
F[m] //F[i]表示包含了第 i 个特征词的文档总数,每个节点上有一个相同该数组
Bi = {d1,d2,…,dn} //节点 i 上的缓存,保存最近接收的 n 个文档

$$Ti=\begin{cases} c_{main1}:d1,d2\cdots \\ c_{main2}:d2,d4\cdots \\ \cdots \\ c_{main1}:d1,d3\cdots \end{cases}$$ //节点 i 上的二维倒排索引表

```
Each node i:
Receive d;
If(d∈Bi){
    Do nothing;
}else{
    Add d to Bi;
    N++;
    For(int i = 1; i <= m; i++){
        If(d[i] > 0)
          F[i]++;
    }
    If(node i is destination)
        Add d to Ti;
    If(node i is not sender){
        Foreach neighbor j of node i:
          If d is not received from j
            Send d to j;
    }
}
```

图 2　算法

(下转第 187 页)

处理建议;下方为该图像的基本信息内容,是采集图像在远程提交时上传的相关信息,该部分内容保存在信息数据库的图像信息表中。并且,诊断结果还可以以报表形式导出。

针对电力系统实际运营中的设备故障检测问题,本系统可以起到很好的辅助作用,最大限度地帮助操作人员完成设备检修。此外,本系统还具有以下优点:

(1)运用数字图像处理中的特征提取和经典分割算法获取图像中的目标区域,得到目标的边缘信息以及目标区域信息,并把基于内容的图像检索技术应用于共享数据库的检索中。

(2)红外热像仪输出结果的解读与控制。红外热像仪大都输出标准格式的数字图像(如 jpg 格式),通过数字图像处理中图像格式的理论,对红外热像仪输出的标准格式的图像进行分析处理,解读出图像中每一个像素的温度数据以及相关参数信息。

(3)采用 B/S 和 C/S 结合的多层结构,实现最大限度的发挥不同构架的不同优势。

(4)系统采用模块化开发,具有良好的可靠性和扩展性,便于系统的维护和二次开发。

5　结束语

所开发的带电设备红外辅助诊断系统集现场预检、远程提交、故障诊断、图像库管理为一体,不仅可以使采集到的红外图像信息及时传送,实现对设备的及时诊断,而且可以使诊断结果再次利用。该系统能及时检测出带电设备的事故隐患,能有效地降低事故发生率,从而提高设备运营的安全性和可靠性。可见,该系统可以广泛应用于在电力、冶金、石化等工业领域。但是由于系统缺乏完备的典型故障图像库,在故障诊断时,缺少相应的典型故障图像作参考,所以不利于为操作人员提供直观的对比分析,这是进一步要解决的问题。

参考文献:

[1] 田裕鹏.红外检测与诊断技术[M].北京:化学工业出版社,2006.

[2] 叶　风,张晓宁.红外技术在电力设备外部故障检测中的应用[J].辽宁工学院学报,2002,22(5):13-15.

[3] 于　勇,孟广军.带电设备红外线检测及诊断[J].青海电力,2007,26(1):16-19.

[4] 胡世征,程玉兰,廖福旺,等.DL/T 664—1999 带电设备红外诊断技术应用导则[S].北京:中国电力出版社,2005.

[5] 付小宁,殷世民,吴志鹏,等.红外图像的动态阈值分割[J].光电工程,2002,29(6):57-60.

[6] 蒋锡健,吴功平,肖晓晖,等.高压输电线路巡检数据库及其管理系统[J].电力建设,2006,27(8):65-68.

(上接第 27 页)

通过对分布式倒排索引进行改进,引入二次的 VSM 排序,避免了传统算法中传输中间结果消耗的网络流量,并根据文档与特征词的关联度,对搜索结果进行了有效的排序。同时提出了一种通用的资源发布算法支持 TFIDF-VSM 中相应数据的维护,有较高的可行性。

参考文献:

[1] Stoica I, Morris R, Karger D, et al. Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications[M]. USA: ACM Press, 2001:149-160.

[2] Rowstron A, Druschel P. Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems [M]. Germany: Springer verlag, 2001.

[3] Reynolds P, Vahdat A. Efficient Peer-to-Peer Keyword Searching[M]. Rio de Janeiro, Brazil: [s.n.], 2003:21-40.

[4] 郑仲伟,郑有才.一个 P2P 搜索引擎的架构和实现[J].电子科技,2007(6):39-42.

[5] 王志晓,张大陆,刘　雷,等.基于本体的 P2P 复杂搜索[J].计算机应用,2007,27(4):780-783.

(上接第 68 页)

[3] CHEN Hong-na, ZU Xu, ZHOU Feng. On the Developing Situation, Research Content and Trend of Workflow Technology[J]. Journal of Chongqing Institute of Technology, 2006, 20(2):65-67.

[4] 郝丽波,李建华,夏明伟.工作流事务性研究综述[J].计算机工程与设计 2007,28(13):3209-3212.

[5] 郭科翔.工作流异常和常见的处理办法[J].闽江学院学报,2007,28(5):79-82.

[6] Shukla D, Schmidt B. Essential Windows Workflow Foundation[M]. [s.l.]: Addison-Wesley, 2006.

[7] Allen K S. Programming Windows Workflow Foundation: Practical W F Techniques and Examples using XAML and C#[M]. Birmingham: PACKT publishing, 2006.

[8] 吕成戍.工作流系统事务处理的研究与应用[D].大连:大连理工大学,2005.

[9] WANG Ni-hong, YU Hai-hao. Research on workflow technology and its developing trend[J]. information technology, 2007(6):67-69.

[10] 林春莺.工作流事务处理的应用解决方案[J].集美大学学报:自然科学版,2006,11(1):58-60.