基于Ontology和EM方法的网页分类研究

丁 艳 曹 倩 王 超 潘金贵

(南京大学计算机软件新技术国家重点实验室 南京大学多媒体技术研究所 南京210093)

Web Page Classification Research Based on Ontology and EM

DING Yan CAO Qian WANG Chao PAN Jin-Gui

(State Key Laboratory for Novel Software Technology, Multimedia Technology Institute of Nanjing University, Nanjing 210093)

Abstract Works on abstracting semantic information from substantive pages of Web and their usage in search engine can lead to intelligent retrieval, or other individual services. This paper mainly focuses on some research about analysis of Web page classification infor. Ontology as a base, using TFIDF word weights and Rocchio algorithm is combined with EM to improve accuracy of classifier. It's proved that this EM procedure works well on enhancing the veracity by the usage of unlabeled pages when the samples are limited.

Keywords Ontology, VSM, Classifier, Feature vector, Document vector

1. 引言

当前,Internet上广泛流行的各种搜索引擎,为人们寻找资源提供了便利,而且还辅以各种用于提高精确度的技术,但普遍缺乏导引能力,即不能帮助用户确定所需信息所在的领域,使得获得的结果经常是风马牛不相及。所以,目前迫切需要的就是开发一种智能化、个性化的搜索工具,使其能够满足不同用户对不同领域的信息进行发现和积累的要求。

由于网页信息没有很好的条理性,缺乏语义结构,因此需要一种机制来对网页进行分析,为其扩展面向计算机的数据,例如,网页的类别,属性,及多个网页之间的一些关系(可称之为规则)。这些信息都是计算机可自动处理的,如果对搜索引擎数据库中每个网页进行类似的处理,获得这些语义信息,毫无疑问可以其为基础向用户提供智能化的搜索,并且准确率也将大大提高。

我们在Dolphin系统中就进行了这样的尝试(该系统是由南京大学多媒体研究所远程教育实验室开发的课件搜集系统,其前身是DOLTRI-Agent系统(Distance and Open Learing Training Resource Information retrieval Agent),它的中文名称是基于RDF和关键字的专用智能网络信息搜集代理(Intelligent Web Information Retrieval Agent))。该系统对网页资源的分析基于向量空间模型,主要通过对网页元数据进行加注[7],即使用某种规范的元数据语言(RDF),以一定的方式对网页进行描述,使得经过加注的网页有比较丰富的语义信息。为了减少系统维护的工作量,加大系统的自动化程度,我们使用了基于Ontology的网页自动加注方法。其基本思想是:首先分析网页的全文、着重分析一些关键部分(如title、heading等等,并以此为基础分析出网页的类别)、网页之间的链接关系。然后通过相关算法在Ontology中选出恰当的词汇来表达这网页中蕴含的语义。考虑到目前的硬件条件和研究进度,系统仅支持计算机学科领域的英文网页和文档。

本文主要介绍了以Ontology为基础使用EM(Expectation-Maximization)方法对网页进行分类的过程。首先阐述了Ontology的含义及其对分类的指导作用;然后给出了一个简单可行的TFIDF分类方法,并讨论了在已有的样本数量有限的情况下,如何利用已分类和未分类的文档使得分类器的效果最好,并在最后指出这种分类方法还有待完善的地方。

2. Ontology的引入

2.1 Ontology的概念及其作用

Ontology就是关于某一领域对象的类别、属性及它们之间的关系的内容理论(content theory)[1]。在计算机学科中,我们狭义地把它看成是关于某一领域的概念性描述。一个典型的Ontology包含领域中的层次化的概念体系,并通过属性-值这一机制来刻画其中的概念。从某种意义上讲,它是一种领域性的知识表示。

Ontology以其对知识的概括和描述能力在Web上得到广泛的应用。一般Web上的Ontology包括分类和一套推理规则。分类定义了对象的类别及其之间的关系,并且是一个可继承的层次结构,这与面向对象思想中的类、子类、实体间的概念是一致的[1]。通过给类指定属性,允许子类继承类的属性,这样我们就能够表达实体之间的大量关系;Ontology中的推理规则提供进一步的功能。例如,一个Ontology可能表达如下的规则:"如果城市属于某一省,而一个学校在这个城市,则该学校也属于这个省。"程序便能据此进行推理,比方说,南京大学位于南京,则它一定在江苏省,而江苏省属于中国,所以它的地址应该遵照中国的标准。计算机并不会真正"理解"这些话,但是根据这样的规则,它就能以一定的方法来对这些信息进行操作了。

正是由于Ontology本身的这些特点,可以运用它增强网络服务的功能。辅以简单的方法,它们就能改进网上搜索的准确性,使搜索程序只寻找那些指向精确概念的网页,而不是仅仅通过模糊关键字查到的所有页面[2]。更高级的应用就是使用Ontology将页面上的信息关联到相关的知识结构和推理规则。

2.2 Ontology在本项目中的应用

Dolphin中使用的Ontology主要参考了SWRC

丁 艳 硕士,主要研究领域为Web信息搜索,agent技术,数据库技术。曹 倩 硕士,主要研究领域为Agent技术,信息搜索,中文分词技术。王 超 硕士,主要研究领域为Agent技术,信息搜索和中间件。潘金贵 教授,博士生导师,主要研究领域为中间件,Agent技术,多媒体远程教育,多媒体移动教学。

(Semantic Web Research Community Ontology)。使用该Ontology是因为构造Ontology是一项浩大而艰苦的工作,需要具备许多本领域的相关知识,所以Ontology的构造一直是研究的热点。SWRC已经经过了若干版本的修改,它能够很好地描述Web网页的性质。在我们的项目中,搜索的信息主要偏重于科研方面,SWRC这个Ontology正好提供了该方面的一个描述。

图1和图2分别以有向图和XML格式展示了该Ontology的一部分。

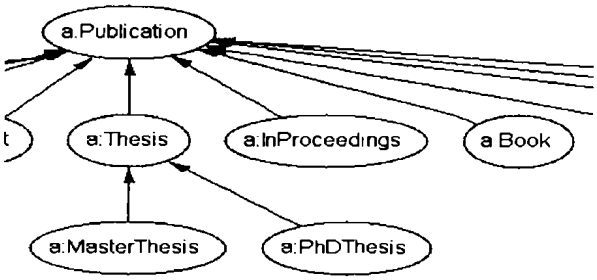

图1 Publication类的一部分(有向图表示)

```
-⟨daml:class rdf:ID="Thesis"⟩
  ⟨rdfs:subClassOf rdf:resource="#Publication"/⟩
-⟨rdfs:subClassOf⟩
 -⟨daml:Restriction⟩
    ⟨daml:onProperty rdf:resource="#author"/⟩
    ⟨daml:toClass rdf:resource="#Person"/⟩
 ⟨/daml:Restriction⟩
⟨/rdfs:subClassOf⟩
```

图2 Thesis类的XML格式描述

Ontology层次化的概念体系可以看成是一组分类,不同的概念形成不同的类别,而概念中的属性描述则对应于类别的属性,概念之间的语义关系也可映射成类别之间的关系。我们首先要做的,就是要确定给出网页的类别,用对应的概念去描述它,然后再进行属性和关系的识别工作。下文介绍的内容就是基于这样的Ontology对网页进行分类的方法。

在本系统中,Ontology的所有信息保存在一个如图2所示的XML文件中,描述了多个类别及其属性,在分类模块启动时,所有分类信息将从该XML文件中装载出来,作为分类的依据。这样的描述形式还有利于扩展和修改。

3. 分类算法和过程

分类的过程对应着Ontology的概念的识别。我们使用了向量空间模型(Vector Space Model)作为文档分类的基本模型,并结合了EM的理论提高分类器的正确率。分类的过程大致可以这样概括:首先对一定数量的已分类的样本网页进行分析,获取它们的文档向量,并以此为基础获得每个类别对应的分类器(classifier)即标准向量,然后用这些分类器与未分类的文档进行比较,与某类文档的特征最接近的网页即归到该类别中;新的分好类的网页同时又可用于调整原有分类器的参数,使得分类器的精确性提高。

3.1 数据的预处理和特征表示

在向量空间模型VSM中,文档可以看成由相互独立的词条组($T_1, T_2, T_3, \cdots, T_n$)构成,对于每一词条$T_i$,根据其在文档中的重要程度赋以一定的权值$W_i$,并将$T_1, T_2, \cdots, T_n$看成一个n维坐标系中的坐标轴,$W_1, W_2, \cdots, W_n$为对应的坐标值,这样由($T_1, T_2, \cdots, T_n$)分解而得的正交词条向量组就构成了一个文档向量空间,文档则映射为空间中的一个点。所有的文档和文档类都可映射到此文档向量空间,而文档类的匹配问题则转化为向量空间中的向量匹配问题[11,12]。假设已知文档类为Q,未知文档为D,两者的相似程度可以用向量之间的夹角来度量,夹角越小说明相似度越高,计算公式如下:

$$Sim(Q,D)=\cos(Q,D)=\frac{\sum_{k=1}^{n} W_{qk}\cdot W_{dk}}{\sqrt{\sum_{k=1}^{n} W_{qk}^2}\cdot\sqrt{\sum_{k=1}^{n} W_{dk}^2}} \quad (1)$$

文档用一组单词来表示是信息检索中常用的技术,这样可以简化对文档的描述,但是如何选取则是一个值得注意的问题。我们需要找的其实是文档中所有单词的一个子集,这些词可以区别不同类别的文档,称之为特征向量[6]。

我们认为,特征单词一般出现在网页的URL、Title/Head、META信息,网页正文、网页中的超链等处。对样本网页进行统计分析后,对以上每个部分,都进行词汇的选择。在选取的过程中,去除出现在stop list中的无意义的单词(一些虚词、介词、html的attribute名等)以及其它无用单词,保留能够反映网页特征的单词。

值得注意的是,相同的单词出现在网页的不同部分,对网页的Ontology分类会产生不同的影响。比如,在Title中出现单词"project",则含有该title的网页很可能是属于Ontology中的project类的,而如果在超链中出现"project",则该网页属于project的可能性不会很大,相反,该网页属于organization的可能性或许会很大。针对这种情况,可以给筛选出来的特征单词加上前缀。如出现在title中的"project"用"title. project"来表示。以此方式,来明确特征单词的出现位置。

经过这些步骤,得到的特征单词向量仍然很大。虽然文档的特征单词太少会影响分类器的精度,但是如果太多也会带来很多噪音数据,反而降低准确率。所以我们采用了一种混合的特征选取方法:

(1)去除低频词(将出现次数少于某个阈值的单词去掉);

(2)去除高频词(将出现次数大于某个阈值的单词去掉);

(3)选择具有较高DF值的单词作为最终的特征单词。

文档频率(Document Frequency,DF)是出现某特征的文档数与文档总数之比,在第三步中忽略掉的就是文档频率较小的特征单词,这些单词更有可能是干扰噪声。根据DF值进行特征选取是一个比较简单的方法,而且容易扩展,它的计算复杂度与训练样本数接近于线性关系。根据已有的实验证明,使用该方法进行特征选取所获得的分类器的正确率与其它方法相比具有明显优势[4,10]。

3.2 分类的主要算法

特征向量选取工作完成后,就可以进行分类器的训练了。Dolphin系统中计算的是特征单词的TFIDF权值,并使用Rocchio算法获得初始分类器,然后结合EM方法以获得更好的分类效果。

3.2.1 TFIDF分类器 该分类器是基于Rocchio相关度反馈算法[8],并采用TFIDF值作为特征单词的权重。算法的基本思想是:文档向量$\vec{d}=(d_1, d_2, \cdots, d_i)$表示一个文档$d$的特征向量,它的每一维代表着经过特征选取后确定下来的特征单词[5]。其中:

$$d_i=TF(w_i,d)\cdot IDF(w_i) \quad (2)$$

TF即为单词频度(Term Frequency),$TF(w_i,d)$表示单词w_i在文档d中出现的次数;如果用$DF(w_i)$表示包含单词w_i的

文档个数，用$|D|$表示样本文档总数，则文档频率的倒数(inverse document frequency)为：

$$IDF(w_i)=\log(\frac{|D|}{DF(W_i)}) \quad (3)$$

如果某个单词 w 出现在许多文档中，则它的 IDF 值就较低，反之如果只出现过一次，则最高。根据式(2)计算出来的 d_i 称为特征单词 w_i 在文档 d 中的权重。这种计算方法使得在文档中经常出现的单词可以作为该文档的索引项，而在许多文档中出现过的单词则可以评估为不太重要的索引项。

通过文档向量，我们就可以求出类的 TFIDF 分类器，即关于某个类的一个基准向量。计算的方法很简单，将同一类的文档向量累加即可。对于每个类 $C_j\in C$，假设 $\vec{c}_j$ 为类 C_j 的基准向量，则

$$\vec{c}_j=\sum_{d\in C_j}\vec{d} \quad (4)$$

对于一个新文档 d'，将它以上述方案获得文档向量$\vec{d}'$，就可以通过比较$\vec{d}'$和$\vec{c}_j$的相似度来决定其类别：

$$R(d',C_j)=\arg\ \underset{C_j\in C}{Max}Cos(\vec{d}',c_j)=\vec{d}'\cdot\vec{c}_j/(\|\vec{d}'\|\cdot\|\vec{c}_j\|) \quad (5)$$

使 $R(d',C_j)$取值最大的 C_j 即为文档 d'的类别。

需要指出的是，$\vec{c}_j$ 的生成是简单地将同一类的文档向量进行累加的过程，但是如果该类中样本数非常多，生成的分类器的各个参数较单一的文本可能会相差很大，这样在计算相似度时得出的值可能普遍较小，所以我们将$\vec{c}_j$ 的计算作了一点修改：

$$\vec{c}_j=(\sum_{d\in C_j}\vec{d})/\|D_j\| \quad (6)$$

D_j 为属于类 C_j 的文档的总数。这样，最后得出的相似度有了较大区别，划分到哪个类也比较明确。

3.2.2 *使用 EM 方法调整分类器参数* EM 方法(Expectation-Maximization)可以利用已分类和未分类的文档训练分类器。它是一种对不完整的数据进行最大可能的后验估计的迭代算法[9]。该算法首先利用已有的分好类的文档训练出一个分类器，然后根据该分类器对每个未分类的文档赋予期望值最大的类别。接着根据所有已分类的文档训练出新的分类器，不断循环直至分类器的参数收敛，不再有较大变化[3]。具体算法[3,6]如下：

输入：已分类的网页集合 D' 和待分类的网页集 D^u；

Step1：根据 D'，对于每个类 $C_j\in C$，通过式(2)～(4)获得一个初始的标准向量 $\hat{\theta}\in\hat{\Theta}$，即分类器；

Step2：以下为一个循环过程，直至分类器的参数不再有较大变化，即对于一个给定的阈值 ε，当循环过后获得的新分类器 $\hat{\theta}'$ 与原有分类器 $\hat{\theta}$ 相比，$\text{distance}(\hat{\theta}',\hat{\theta})=\hat{\theta}'\cdot\hat{\theta}/(\|\hat{\theta}'\|\cdot\|\hat{\theta}\|)\leqslant\varepsilon$ 时，循环终止：

Step2-1：使用目前的分类器 $\hat{\theta}$，对 D^u 中每个网页进行评判，将网页划入某一类别中；

Step2-2：对于某个类别 $C_j\in C$，使用新加入该类的网页的文档向量，和原有数据一起重新获得一个分类器 $\hat{\theta}$；

输出：一个最终的分类器集合 $\hat{\Theta}$，每个可以对为分类的网页进行评估，预测它所属的类别。

其中，如果有一个类别 $C_j\in C$，其初始分类器为$\vec{c}_j$，属于该类的网页集合为 D'_j；使用$\vec{c}_j$ 对 D^u 中网页进行分析比较，结果网页集 $D^u_j\subset D^u$ 属于该类，其中网页的文档向量为 $\vec{d}^u$，则易见新的分类器为：

$$\vec{c}'_j=\frac{c_j\cdot|D'_j|+\sum_{d^u\in C_j}\vec{d}^u}{|D'_j|+|D^u_j|} \quad (7)$$

上述算法和公式将 EM 的思想和 TFIDF 分类方法相结合以期取得较好效果，在实际应用过程中，只需对新增加的网页信息进行计算，而无需将新老网页一起重新分析计算。随着系统搜集的网页数目不断增长，还可以对新加入的网页分批地运行此方法，进行不断的调整，以获得稳定的效果。

4. 实验结果和分析

考虑到系统支持的分析功能是基于英文网页，我们借鉴了 CMU(Carnegie Mellon University)作文档分类时使用的数据集，提取了其中一部分，并加上我们自己搜集的国外大学网页、计算机领域相关网页，约1000多个，依照上述分类方法进行了实验。因为样本数有限，所以将类别精简为 SWRC 中的几个根类(Product、Project、Person、Organization、Topic、Event、Publication)。表1和表2为一部分统计(Organization类)结果。

表1 使用 EM 方法对分类器参数进行调整后的正确率

样本数	20	50	100
初始分类器的正确率	72.5%	76.8%	81.9%
加入20个新网页后的正确率	73.4%	77.4%	82.5%
加入50个新网页后的正确率	75.2%	79.3%	83.7%
加入100个新网页后的正确率	78.6%	81.1%	85.2%

表2 样本数为100时选取的特征单词数目对正确率的影响

选择的特征数目	10	20	35	50
正确率	77.3%	79.2%	81.5%	81.9%

从以上数据可以得出以下结论：

(1)样本集的大小、样本的质量对分类器的正确率影响较大，所以如果条件允许，应搜集尽可能多的并且质量比较好的样本，当然那样需要耗费很多的人力去进行挑选和分类。

(2)使用 EM 方法可以弥补样本较少的不足，可以通过不断的调整以期达到比较满意的效果，对于样本较少的情况，EM 方法显著有效；而样本数比较多时，则影响相对较小。

(3)特征的选取数目对最后结果也能产生影响，当然并不是特征数越多越好，在到一定数目之后，变化就会相当缓慢，有时甚至不升反降，这是由于特征单词中带进了噪音数据。针对处理对象为网页，通常内容不是很多，一般取25～35个即可。

结束语 Dolphin 系统使用了基于 Ontology 和 EM 方法的分类模块后，就能自动地进行网页分类，为下一步其它网页信息的提取打下了基础。将系统数据库中的大量网页与分类信息结合，可以提高用户查询的智能性，获得令用户满意的结果。因为系统目前的硬件资源的限制及样本数量的不足，我们没有将分类进行细化，但是相信随着系统的不断发展和完善，实现层次结构的分类是必然的；此外，系统目前只能支持英文网页的分析，而且 TFIDF 的分类方法虽然简单，但其效果并不非常理想，所以下一步的研究重点将转移到中文网页分类和更优秀的分类算法的实现上来。

参 考 文 献

1 Chandrasokaran B, Josephson J R. Whar Are Ontologies, and Why Do we Need Them? IEEE Intelligent systems, January/February

1999
2 Berners-Lee T, Hendler J, Lassila O, 魏丰译. 语义网. http://www.xml.org.cn
3 Nigam K, Mccallum A, Thrun S, Mitchell T. Text Classification from Labeled and Unlabeled Documents uring EM. Machine Learning, 1999
4 Yang Yiming, Pedersen J O. A Comparative Study on Feature Selection in Text Categorization. In: Proc. of ICML-97, 14th Intl. Conf. on Machine Learning, 1997
5 Joachims T. A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. In: Proc. of ICML-97, 14th Intl. Conf. on Machine Learning, 1997
6 Luke S. Ontology-Based Knowledge Discovery on the World-Wide Web http://www.cs.umd.edu/~seanl/, 1996
7 Iwazume M, Shirakami K, Hatadani K, Takeda H, Nishida T. IICA: An Ontology-based Internet Navigation System. JL41-96 Workshop on Internet-based Information Systems, Portland, OR, 1996
8 Rocchio J. Relevance Feedback in Information Retrieval, in The SMART Retrieval System: Experiments in Automatic Document Processing, Chapter 14, Prentice-Hall Inc. 1971. 313~323
9 Dempster A P, Laird N M, Rubin D B. Maximum likelihood from incomplete data via the EM algorithm. Journel of the Royal Ststistical Society, Series B, 1977, 39(1): 1~38
10 朱明, 王军, 王俊普. Web 网页识别中的特征选择问题研究. 计算机工程, 2000, 26(8)
11 卢增祥, 李衍达. 交互支持向量机学习算法及其应用. 清华大学学报(自然科学版), 1999, 39(7)
12 邹涛, 孙赛. 文档自动分类技术及其实现. 计算机系统应用, 1999(4)

(上接第92页)

则。过滤规则由用户定义,以决定是否接受数据包和需要接收多少数据。每一条规则执行一组操作,具体操作可以分为指令装载、指令存储、执行算术指令、执行跳转指令、执行返回指令等几个类别。

结合图5,过滤过程描述如下:当一个数据包到达网络接口时,链路层驱动程序将其提交到系统协议栈;如果 BPF 正在此接口监听,驱动程序将首先调用 BPF,BPF 将数据包发送给过滤器,过滤器对数据包进行过滤,并将数据提交给过滤器关联的上层应用程序;然后链路层驱动将重新取得控制权,将数据包提交给上层的系统协议栈处理。

3.4 BPF 过滤模型

网络监听应用可能只关心网络流量数据的一个子集,因此包的过滤将大大提高系统的性能,为了减少内存操作,以尽量避开瓶颈,包过滤应该在包的原始位置 (in-place,即 DMA 操作存放数据的内存位置)进行过滤,而不是首先拷贝数据。

包过滤函数一般可用布尔函数表示,如函数返回为 ture,则将包拷贝到上层,反之忽略该包。通常有两种方法可以实现该函数:布尔表达树方式和可控制流图方式(CFG, Control Flow graph),如图6,7用两种方式表示了接受 IP 包和 ARP 包的函数。

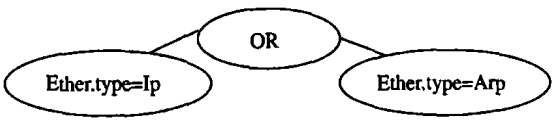

图6

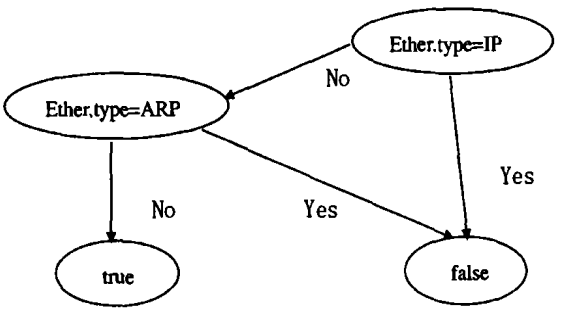

图7

在树型模型中,每一节点代表一个布尔关系如 AND、OR,每个叶子代表一个谓词断语,如 type=IP,边表示布尔操作和操作数的关系。

在 CFG 模型中,每个节点代表一个谓词断语,而边代表控制转换,如果谓词断语为 true,则转向右边,反之转向左边。每个 CFG 图有两个终结节点,表示返回 true 或者 false。

以上两个模型的计算功能大致相同,用一个模型可以表示的函数用另一个模型也同样可以表示,但是在实现的时候却大不相同,树型图的操作往往基于操作数堆栈操作,而 CFG 模型可以基于寄存器操作。

树型模型基于操作数堆栈运算,指令将常数或包数据压入堆栈,在堆栈顶层执行布尔运算或位运算,通过连续运算,在堆栈清空时返回结果。堆栈需要内存模拟,其中的 pop 和 push 操作需要指针的加减运算将数据从内存读取和写入内存。另外树型结构可能进行重复操作,例如某个值可能在多个叶子中分别从内存读取并计算,因此这种方式效率较低。

BPF 采用 CFG 过滤模型而非过滤树模型,因为过滤树模型在解释包时可能进行重复计算,而 CFG 模型解释包时数据包的解释状态和路径是可记忆的,不需要重复计算。

例如采用 CFG 模型接受来自主机 FOO 的所有数据包,包括 IP、ARP 和 RARP 包,在所有包中都需要查询地址,但不同包的格式是不一样的,采用树型模型时,为检查数据包是 ARP 还是 RARP,在表达树模型中,需要7次比较操作和6次布尔操作,而采用 CFG 模型的最长路径需要5次比较操作,平均只需要3次。

结束语 因为危害网络安全的有害信息能顺利地通过防火墙和入侵检测系统,它们不在像几年前那样明目张胆地从某一个地址发出,而往往是包装成合法的报文,或者加载到合法的报文中间,通过合法的用户发布出去。针对这类有害信息,只有通过特定的网络信息审计系统才能将其检查出来。而网络的审计的第一步是网络信息的监听,本文提出采用基于路由器的网络信息监听技术。然而,其面临的一个问题是网络信息监听主机工作负载过大,原因是监听的信息数据中包括大量不需要关心的数据,或者称为垃圾数据。因此论文提出采取高效的 BPF 信息过滤机制对监听的信息数据进行过滤,从而大大提高工作效率。

参 考 文 献

1 监听与隐藏—网络侦听揭密与数据保护技术. 求是科技. 北京:人民邮电出版社, 2000. 12~234
2 Braden R T. A pseudo-machine for packet monitoring and statistics. In: Proc. of SIGCOMM '88 (Stanford, CA, Aug. 1988), ACM
3 Steven McCanney and Van Jacobson. The BSD Packet Filter: A New Architecture for User-level Packet Capture. Lawrence Berkeley Laboratory (December 19, 1992)
4 张兴虎. 黑客攻防技术内幕. 北京:清华大学出版社, 2002. 45~67