第33卷 第10期 计 算 机 工 程 2007年5月
Vol.33 No.10 Computer Engineering May 2007
·网络与通信· 文章编号：1000—3428(2007)10—0092—03 文献标识码：A 中图分类号：TP393

基于NP的垃圾邮件分析系统的设计与实现

翟伟斌[1,2]，叶进星[1,2]，陈 宇[1,2]，许榕生[1,2]
(1. 中国科学院高能物理研究所计算中心，北京 100049；2. 中国科学院研究生院，北京 100049)

摘 要：垃圾邮件的泛滥成灾给人们的正常生活带来了很大的不便和危害。该文设计并实现了基于NP的垃圾邮件分析系统，具有邮件抓取、还原和类别识别功能，能够有效识别垃圾邮件。实验结果表明，该系统对于垃圾邮件的追踪具有良好的实用价值。
关键词：网络处理器；垃圾邮件；向量空间模型

Design and Implementation of Spam Mail Analysis System Based on Network Processors

ZHAI Weibin[1,2], YE Jinxing[1,2], CHEN Yu[1,2], XU Rongsheng[1,2]
(1. Computing Center, Institute of High Energy Physics, Chinese Academy of Sciences, Beijing 100049;
2. Graduate School, Chinese Academy of Sciences, Beijing 100049)

【Abstract】 The spam mails are inconvenient and harmful to people's life. The paper designs a spam mail analysis system based on network processors, which has the ability of snatching, restoring mails and dispatching mails, which can identify spam mails effectively. The good performance of this model is presented.
【Key words】 Network processor; Spam mail; Vector space model

垃圾邮件一般是指未经请求而发来的电子邮件，通常包含一些商业广告以及其它不良信息。自从互联网普及以来，电子邮件逐渐成为人们生活中便捷的通信手段之一。然而，随之产生的垃圾邮件迅速蔓延，污染网络环境，占用大量传输、存储和运算资源，影响了网络的正常运行。根据2005年1月份中国互联网信息中心CNNIC发布的第15次互联网统计报告，中国用户每周收发的正常邮件数分别为4.4封和3.6封，收到的垃圾邮件数达到7.9封，垃圾邮件数量已经超过了正常邮件数量。对于用户来说，处理垃圾邮件不但造成时间和费用的浪费，而且有可能收到对系统安全构成威胁的病毒邮件。因此，消除垃圾邮件具有非常重要的意义。

目前处理垃圾邮件主要采用在邮件服务前面放置垃圾邮件处理器。在垃圾邮件达到邮件服务器前进行过滤，一般采用基于IP、域名的过滤。这种过滤方法只是在垃圾邮件到达目的地时才过虑掉，而不能有效地找到垃圾邮件的起源地，在源头上切断垃圾邮件。本文提出的基于NP垃圾邮件分析系统，可以部署在各个交换机上，能够对经过交换机的垃圾邮件进行截获分析，并把分析结果反馈给管理人员，管理人员根据反馈结果进行分析，用来准确定位垃圾邮件的来源。

1 邮件抓取

目前，网络监控系统在硬件上一般采用PC架构，抓包软件则采用Linux下的Libpcap工具包，实现对网络数据包的捕获。这种架构只能适应于百兆带宽，对于高速网络，由于PC机CPU处理能力和存储总线吞吐能力的限制会出现严重的丢包现象。为了实现高速网络垃圾邮件监控，本文采用网络处理器（Network Processor，NP）架构。NP是为宽带网络应用而设计的专用处理器，它具有很高的数据包处理能力，可以完全满足高速网络的数据包抓取任务。

本系统将NP串接在想要监控的网络上，能自动对所有流经的网络包进行抓取，然后传送给垃圾邮件处理器。设计支持透明接入的方式，这给系统部署带来极大的方便，因为不需要对路由器或交换机进行任何设置，用户甚至感觉不到NP的存在。

邮件抓取流程如图1所示， 对NP传送来的一个包，先将其存入内存中，而后对其进行协议分析，协议分析的顺序从底层到高层，同时进行协议过滤，将符合条件的数据包传给下一步处理程序。由于电子邮件一般采用POP3和SMTP协议，因此本系统只抓取POP3和SMTP数据包，其余数据包全部过滤掉。

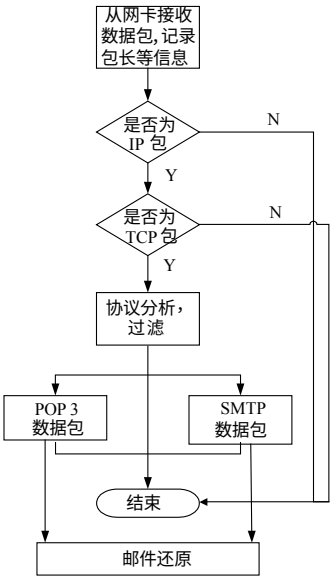

图1 邮件抓取流程

2 邮件还原

电子邮件在网络上是以包的形式传输，一封邮件根据长

基金项目：国家自然科学基金资助项目(70471064)
作者简介：翟伟斌(1975－)，男，博士生，主研方向：信息提取；叶进星、陈 宇，博士生；许榕生，研究员、博导
收稿日期：2006-05-25 **E-mail：**zhaiwb@gmail.com

度的不同，会拆分为数个不同的数据包，这些数据包在网络上传输时，不一定会走相同的路径，而且还有丢包的可能，所以抓取到的邮件数据包不一定是完整的。即便是完整的数据包，顺序一般也是打乱的。所以要对数据包进行重组。本文采用 Linux 操作系统下的 Libnids 函数库实现网络数据包的重组。还原后的邮件存储后供下一步进行邮件分析。Libnids 的网络数据处理流程如图 2 所示。

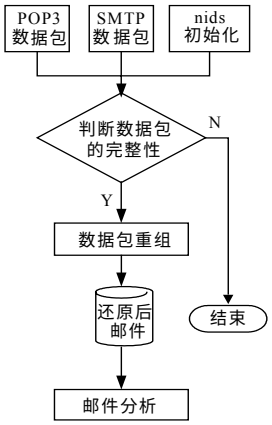

图 2 邮件还原流程

3 邮件分析

本邮件分析系统采用基于改进的向量空间模型，通过机器学习的方法，在大量语料中自动获取特征词，即垃圾邮件和非垃圾邮件的特征集，利用规则统计相关信息，并用向量空间表示，当抓获到新邮件时，对邮件进行处理，用向量空间表示，分别与垃圾邮件向量空间和非垃圾邮件向量空间进行比较，得出抓取到的邮件的类别。原理如图 3 所示。

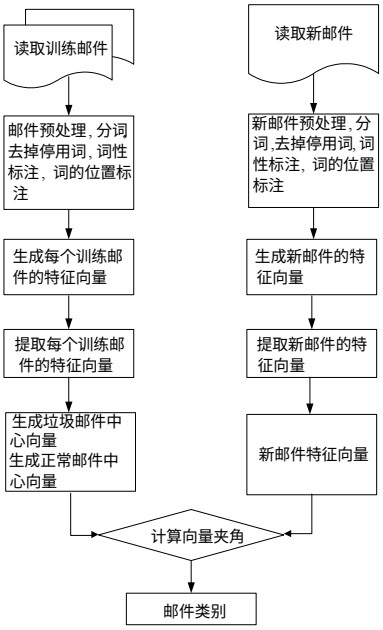

图 3 邮件分析流程

3.1 邮件分类技术

(1)定义。邮件分类就是在给定的分类模型下，根据邮件内容自动判别垃圾邮件和非垃圾邮件类别的过程。

(2)邮件的表示。邮件还原后为文本形式，但是计算机只能识别二进制码，不可能像人一样读懂文本，所以必须将邮件内容转换为计算机可识别格式。根据“贝叶斯假设”，假定字和词在确定文本内容的作用上相互独立，就可以使用文本中出现的字或词的集合来代替文本。

目前，在信息处理方向上，文本的表示主要采用向量空间模型(VSM)。向量空间模型的基本思想是以向量来表示文本：$\vec{d} = < t_1, w_1; t_2, w_2; ...; t_n, w_n >$，其中 t_i为文本的特征项，w_i是t_i在该文档中的权值。特征项一般可以选择字、词或词组，普遍认为词作为特征项要优于字和词组，因此，要将文本表示为向量空间中的一个向量，就首先要提取文本中的特征词，由这些特征词及其权重作为向量的维数来表示文本，权重的计算方法主要运用 TFIDF公式，我们在系统中采用了一种比较普遍的 TFIDF 公式[1]：

$$W_i(t_i, \vec{d}) = tf(t_i, \vec{d}) \times \log(N / n_{t_i} + 0.01) \times Key_{t_i}$$

其中，$W_i(t_i, \vec{d})$ 为词 t_i在文本 $\vec{d}$ 中的权重，而 $tf(t_i, \vec{d})$ 为词t_i在文本 $\vec{d}$ 中的词频，Key_{t_i} 为词t_i的加权值，取值按照表 1 所示，N 为训练文本的总数，n_{t_i} 为训练文本集中出现t_i的文本数。为了便于处理，通常对$W_i(t_i, \vec{d})$作归一化处理，限制其取值位于区间[0, 1]中，即

$$W_i'(t_i, \vec{d}) = \frac{W_i(t_i, \vec{d})}{\sqrt{\sum_{j=1}^{K} W_j^2(t_j, \vec{d})}} \quad (1)$$

其中，K 为向量 $\vec{d}$ 的维数。

邮件文本经过分词程序预处理后，统计词频，最终表示为上面描述的向量。采用 TFIDF 方法计算权重没有考虑邮件的结构特性和特征词的词性对权重的影响，事实上同一个关键词出现在邮件的不同位置，它所能表达邮件内容的能力是有差别的，同时同一位置出现的不同词性的特征词，在表达邮件内容的能力上也存在很大差异。本文引入了权重系数概念，对于一个邮件，其标题、正文、附件中出现的特征词，根据词性，给予不同程度的加权(见表 1)。

表 1 不同域特征词的加权值

	名词	动词	形容词	副词
标题	1.5	1.45	1.4	1.4
正文	1.2	1.15	1.1	1.1
附件	1.1	1.05	1.025	1.025
其它	1	1	1	1

3.2 特征项的抽取

由于邮件长度的随机性，因此在对邮件进行特征提取时，有可能得到的向量空间维数很大，随着维数的增加，用来训练邮件分类器和测试性能所需的对象个数会随着系统维数呈指数级增长[2]。同时还有很多词对于区分垃圾邮件和非垃圾邮件所起的贡献很小，可以完全忽略。 因此需要进行维数压缩，这样做还可以提高程序的效率。对于每一类邮件，应去除那些表现力不强的特征项，筛选出针对该类的特征项集合，本文采用词和类别的互信息量进行特征项抽取[3]。其计算公式如下：

$$I(t, C_j) = \log[\frac{P(t|C_j)}{P(t)}] \times Key_t \quad (2)$$

其中，$P(t \mid C_j)$ 为特征词t在类别C_j中出现的比重，Key_t为特征词t加权值，取值如表 1 所示，$P(t)$是特征词t在所有训练文本中的比重。

对计算出来的所有的互信息量 I 进行从大到小排序，根据需要抽取一定数量的特征项。

3.3 邮件训练和分类算法

给定一定数量的垃圾邮件和正常邮件进行训练，对于各个类别中的每一个邮件按照图 3 所示进行训练，最后得出该类邮件的中心向量，中心向量为该类别所有训练文本向量的简单算术平均。

对于需要处理的新的邮件，进行处理，用特征向量表示。最后计算新邮件和每类邮件中心向量的相似度。公式为

$$sim(\vec{d}_i, \vec{d}_j) = \frac{\sum_{n=1}^{K} W_{in} \times W_{jn}}{\sqrt{(\sum_{n=1}^{K} W_{in}^2)(\sum_{n=1}^{K} W_{jn}^2)}} \quad (3)$$

其中，$\vec{d}_i$为新邮件的特征向量，$\vec{d}_j$为第 j 类邮件的中心向量，K 为特征向量的维数，W_n为向量的第 n维。

比较每类中心向量与新邮件的相似度，将邮件分到相似度最大的那个类别中。

4 实验及结果

为了有效地评价垃圾邮件监控系统性能的好坏，我们使用两个评定指标GP(邮件识别的准确率)和GR(邮件识别的查全率)[4]。

$$GP = \frac{N_p}{N_p + N_w} \tag{4}$$

其中，N_p为正确识别出的垃圾（正常）邮件数，N_w为误识别为垃圾（正常）邮件数。

$$GR = \frac{N_p}{N_s} \tag{5}$$

其中，N_s为应该识别出的垃圾（正常）邮件数。

首先本文对中科院高能物理研究所mail用户举报的垃圾邮件进行筛选，去除一些包含加密信息、图片信息以及一些内容过短的垃圾邮件，选取 1 000 封作为训练样本。按照图 3 方式进行训练，得出垃圾邮件的中心向量。然后对 NP 抓取的邮件进行手动分析，选取 1 000 封正常邮件作为正常邮件训练样本，同理按照图 3 进行训练，得出正常邮件的中心向量。对随后在中心交换机上抓取的 1 000 封邮件进行分类，结果如表 2 所示。

表 2 邮件过滤的准确率，查全率

	垃圾邮件	正常邮件	GP	GR
垃圾邮件类别	483	26	94.8%	91.3%
正常邮件类别	35	441	92.6%	93.6%
无法识别邮件	11	4		

本系统对抓取的邮件分类后，如果是垃圾邮件，会向管理员发出警报信息，管理员根据得到垃圾邮件警报信息(包含垃圾邮件的来源)，进行垃圾邮件追踪，便于从源头上消除垃圾邮件。邮件分类错误的主要原因是一些邮件内容过短，只有单单几个字。对于这样的邮件，很难判断类别。

邮件种类无法识别是因为表示邮件的向量和垃圾邮件中心向量，正常邮件的中心向量的差别都超过了设定的限度值。主要原因是：(1)用来参与训练的样本邮件内容不够全面，系统训练不够充分。(2)邮件中包含了无法识别内容，邮件无法正常还原。

5 结束语

从系统得到的实验结果可以看出，采用基于 NP 架构的垃圾邮件分析系统，可以有效地抓取经过高速网络交换机的垃圾邮件，并能快速地对邮件进行还原、分类，同时邮件类别识别的准确率也达到了可以接受的程度。但是只能对邮件中的文字内容进行处理，对于图片信息无法处理。由于越来越多的垃圾邮件(如商业广告)采用图片形式，因此必须能够监控到这部分垃圾邮件。下一步工作将是带图片信息的邮件还原，这将是一个很有挑战性的工作。

参考文献

1 Sebastiani F. Machine Learning in Automated Text Categorization[J]. ACM Computing Surveys, 2002, 34(1): 11-12, 32.

2 Meisel W. Computer-oriented Approaches to Patten Recognition[M]. New York: Academic Press, 1972.

3 庞剑峰, 卜东波, 白 硕. 基于向量空间模型的文本自动分类研究与实现[EB/OL]. http://www.ict.ac.cn/xueshu/2001/115.doc.

4 Sakkis G, Droutsopoulos I, Paliouras G. Stacking Classifiers for Anti-spam Filtering of E-mail[C]//Proc. of EMNLP'01. 2001: 45.

(上接第 65 页)

表 2 软中断进程化之前与之后的内核不确定延迟（μs）

软中断进程化之前				软中断进程化之后			
200/次	Min	Max	Avg	200/次	Min	Max	Avg
1	0	38	6.15	1	0	1	0.03
2	0	11	2.99	2	0	0	0.00
3	0	15	3.16	3	0	1	0.01
4	0	27	4.46	4	0	1	0.01
5	0	31	3.66	5	0	2	0.04
总计	0	38	4.084	均值	0	2	0.018

表 3 软中断进程化之前与之后的执行延迟（μs）

软中断进程化之前				软中断进程化之后			
200/次	Min	Max	Avg	200/次	Min	Max	Avg
1	0	38	5.28	1	0	2	0.16
2	0	15	4.69	2	0	2	0.20
3	0	14	4.66	3	0	2	0.22
4	0	15	4.57	4	0	3	0.27
5	0	15	4.98	5	0	2	0.17
总计	0	38	4.836	总计	0	3	0.204

4 结束语

本文实现了中断进程化方法，不仅大大减少了原中断机制造成的大量不确定内核延迟，而且可以实现与系统中其他任务在统一调度框架下的资源竞争。实验表明这种改进是显著的，较大地减少了内核的不确定延迟。该研究是基于 Linux 的嵌入式实时控制系统，因此今后的工作将包括：

(1)高精度时钟管理系统的实现；

(2)在中断进程化基础上，研究不同的实时调度策略，以支持不同实时性需求的应用；

(3)研究网络资源调度以实现实时远程控制等。

参考文献

1 李小群, 赵慧斌, 叶以民, 等. RFRTOS: 基于 Linux 的实时操作系统[J]. 软件学报, 2003, 14 (7).

2 Michael B. A Linux-based Real-time Operating System[D]. Socorro, New Mexico: New Mexico Institute of Mining and Technology, 1997.

3 House S B, Niehaus D. KURT-Linux Support for Synchronous Fime-grain Distributed Computations[C]//Proc. of the 6th IEEE Real Time Technology and Applications Symposium. 2000.

4 Srinivasan B. A Firm Real-time System Implementation Using Commercial Off-the-shelf Hardware and Free Software[D]. Department of Electrical Engineering and Computer Science, University of Kansas, 1998-02.

5 赵慧斌. RFRTOS——基于 Linux 的 Qos 实时操作系统[D]. 北京: 中国科学院软件研究所, 2003.