文章编号:1671-9352(2007)11-0032-05

RealMon:处理低质量SNMP数据流的实时监测系统

李人和[1],宫学庆[1],常建龙[2],周游弋[1],周红福[1],周傲英[1]
(1.复旦大学 计算机科学与工程系,上海 200433;2.上海电信,上海 200120)

摘要:提出了一个新颖的数据流监测系统RealMon的设计和实现。该系统能够在大量的网络流量数据中通过分析不同数据流之间的关联关系及时地检测出数据异常。通过应用数据流挖掘算法,该系统能够对电信骨干网络的SNMP流量数据进行监测。同时为了解决所采集SNMP数据中存在着的大量数据质量问题,该系统集成了数据流清洗算法,该算法能够实时处理SNMP数据来提高所采集数据的质量。在模拟环境中的测试表明,该系统能够在SNMP数据流上同时对数千条链路进行有效监测。

关键词:数据清洗;数据流监测;SNMP数据
中图分类号:TP311 **文献标志码**:A

RealMon: a real stream monitoring system for low quality SNMP data

LI Ren-he[1], GONG Xue-qing[1], CHANG Jian-long[2], ZHOU You-yi[1], ZHOU Hong-fu[1], ZHOU Ao-ying[1]
(1. Computer Science and Engineering, Fudan University, Shanghai 200433, China;
2. Shanghai Telecom, Shanghai 200120, China)

Abstract: A data stream monitoring system, named RealMon, was designed to discover correlations and detect anomalies among thousands of network links. Some renowned algorithms for data stream analysis were implemented in this system to monitor the huge amount of simple network management protocol messages, which were collected from routers in the telecom backbone network. Some data cleansing algorithms were also integrated into the system to address the data quality problem among SNMP messages, and such a function features RealMon system. Moreover, a user-friendly console interface was developed for network administrators. The experiments show that the system could perform efficiently in a simulated environment.

Key words: stream data cleansing; stream data mining; SNMP flow data

0 引言

随着网络技术及其应用的飞速发展,如何更好地了解网络的运行状况是所有电信服务提供商都面临的挑战。目前,通过对关键网络设备采集SNMP(simple network management protocol)数据并进行分析,是网管部门了解网络运行现状的一个重要手段。

InterMon[1]是一个能够对大规模网络数据簇进行监测和分析的系统,它能够通过检测数据集隐变量的变化来查找异常。WeatherMan[2]则采用神经网络技术对设备运行时的环境数据进行分析,并根据分析结果对设备的散热和冷却系统进行管理。然而,这些系统都只能对理想的数据流进行处理,由于电信系统所采集的原始SNMP数据通常存在着各种数据质量问题,如数据时间漂移,数据缺失,读数错误等,直接利用已有数据分析方法来处理和分析原始的SNMP数据,所得到的结果往往会存在严重的

收稿日期:2007-06-20
基金项目:国家自然科学基金资助项目(60673134);上海市电信公司(Shanghai Telecom Co.,LTD)"网络流量监测数据处理工具开发项目"支持
作者简介:李人和(1982-),男,硕士研究生,主要研究数据流挖掘及数据流清洗.Email: lirenhe@fudan.edu.cn

误差,并不能真实反映网络流量的现状。

现有的数据清洗算法和工具主要针对数据库中的静态数据进行处理[3]。文献[4]主要对从传感器网络中采集的数据进行清洗,作者假设采集到的传感器数据服从高斯分布,并利用高斯分布的性质来实时清洗数据。文献[5]提出了对 RFID 数据进行清洗的方法,该方法利用二项分布(binomial distribution)的性质来过滤原始数据流中的错误数据来得到较为准确的结果。在电信骨干网络中所采集的大量 SNMP 数据以数据流的形式持续不断,将这些数据保存在数据库中,并进行离线的清洗和处理无法满足对网络流量实时监控的需要,并且无法得到一个有效的数据分布模型能描述所有的 SNMP 流量数据,因此现有的数据清洗方法不能被直接应用于 SNMP 数据流的处理。

本文设计并实现了一个数据流实时分析和监测系统:RealMon。该系统能够对从电信骨干网路由器中采集的大量 SNMP 数据进行实时分析处理并报告网络流量的异常变化。通过该系统,能够及时找出链路中的流量异常并分析不同链路之间的关联性,从而更好地了解网络的运行情况。本文的贡献主要包括以下 3 个方面:

(1) 设计并实现了一个实时的数据流处理系统,能够对从网络设备中采集的 SNMP 流量数据进行处理和分析,从而实现对网络流量现状的实时监测。该系统可以实时监测数千条网络链路。

(2) 在 RealMon 系统中实现了最新的数据流分析算法,实验表明该系统能够高效地检测出网络链路中的异常和关联情况。同时,该系统具有良好的可扩展性,可以方便地集成其它分析算法。

(3) 为了提高系统的稳定性以及降低系统的误报,在 RealMon 中设计并实现了针对数据流的数据清洗方法。

1 数据描述

1.1 SNMP 数据特征

SNMP 是一种被广泛应用的网络管理协议,它使用嵌入到各个网络设备中的代理软件来收集网络的流量信息和不同网络设备的统计信息。检测流量的端口每隔 5 min 向中心服务器传输各条链路的实时流量数据,并且对于同一条链路,采集系统将会产生多条数据流来传输链路的多个流量指标。本文主要关注其中 4 种指标,它们分别是流入一条网络链路的包(packet)数和字节(byte)数以及流出这条链路的包数和字节数。如图 4(a)展示了监测同一链路的多条数据流,这些数据流之间相互关联度是非常高的,而一旦这种关联被打破,那么在这条链路中极有可能发生异常。

SNMP 流量数据被封装在数据包中进行传输,每一个数据包都包含了大量信息,而 RealMon 系统并不需要分析所有的信息,而只需要关注其中的 4 个数据域,它们分别是 Task_ID, Group, Value 和 Time。其中 Task_ID 是每一条流的标识符,它主要是用来说明数据流具体监测的链路以及数据流的类型(流出或流入某一条链路的包或字节)。系统使用 Group 域将所有的 SNMP 流量数据按链路进行分组使得每一组包含所有监测同一条链路的数据流。而 Time 域则被用来表示生成每个数据包的时间,Value 是数据包中表示所监测流量的数值。

1.2 SNMP 数据质量问题

为了更好地说明 SNMP 数据质量问题,首先简单介绍 SNMP 数据的采集过程。在上海电信中,网络管理员主要通过轮询(polling)的方式获取 SNMP 数据:管理进程定时向各个设备的代理进程发送查询请求消息,当代理进程收到查询请求后,向缓冲区发送流量数据,这些数据经过处理后在内存队列中等待直到它们依次被管理进程接收。然而,由于数据排队等待,传输延时以及网络故障等问题的存在,管理进程所接受到的数据存在着以下一些数据质量的问题:

(1) 流量数据缺失:电信网络并不是一个十分稳定的环境,而这种不稳定性往往会造成数据包的丢失。

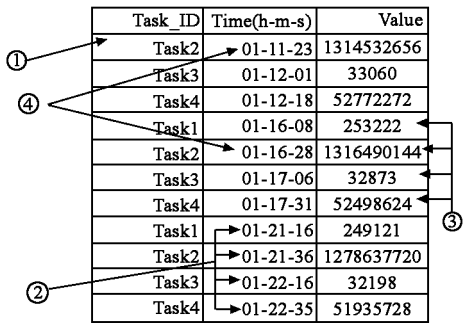
(2) 数据采集时间漂移:管理进程以 5 min 为间隔,周期性地向各个设备发送请求,但是,所收到前后数据包的时间间隔却往往并不是 5 min,称这种情况为时间漂移。这主要是因为网络传输存在延时并且延时长短不一,不同的消息在传输过程中的延时时间并不一致。

(3) 数量级不匹配:尽管同一组中的数据流变化趋势非常相似,但它们数量级的差异却非常大。例如,监测一条链路字节数的数据流,它的值往往是同一条链路包数据流值的 10 000 倍;

(4) 数据接收时间不同步:所有的数据流分析算法和系统都需要在同时接受同一组数据流的新数据,然而,这在实际情况中是很难满足的。由于网络延时长度的不确定性,无法预测每一条数据流数据的到达时间。

图 1 给出了数据质量问题的具体实例,其中①

指出 Task1 数据缺失,②表示的是同一组数据流异步接收数据,③说明了同组数据数量级差距巨大,④表示 Task2 的数据采集时间漂移。



Task_ID	Time(h-m-s)	Value
Task2	01-11-23	1314532656
Task3	01-12-01	33060
Task4	01-12-18	52772272
Task1	01-16-08	253222
Task2	01-16-28	1316490144
Task3	01-17-06	32873
Task4	01-17-31	52498624
Task1	01-21-16	249121
Task2	01-21-36	1278637720
Task3	01-22-16	32198
Task4	01-22-35	51935728

图 1　SNMP 流量数据的质量问题

Fig.1　Quality problem of SNMP flow data

2　系统设计和实现

2.1　系统结构

图 2 展示了 RealMon 整体的系统结构。系统的核心由 4 个模块组成,分别是数据抽取和转换模块,数据清洗规整模块,数据清洗模块以及实时交互控制台。

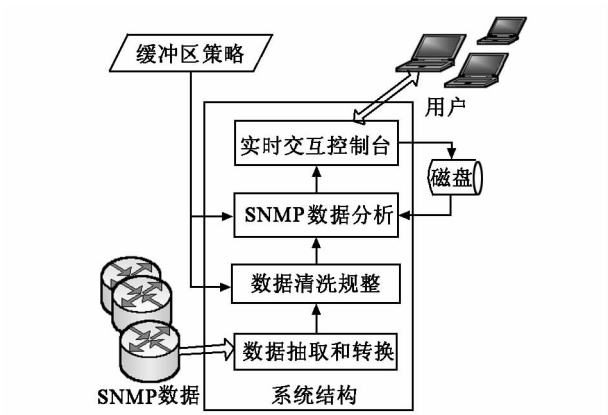


图 2　系统结果

Fig.2　System structure

当该系统从服务器接收到 SNMP 数据包时,这些数据包将被首先发送到数据抽取和转换模块中。只需要对数据包中的 Task_ ID, Group, Time, Value 的值进行分析,而原始数据包含了大量多余的信息。因此,该模块的任务是拆开原始的 SNMP 数据包,从中抽取出所需要的信息,并将这些信息封装成预定的格式传输给上层模块。由于原始数据存在着大量的数据质量问题,因此,在对这些数据进行深入分析之前,需要在数据清洗规整模块中对数据进行实时预处理。这一模块的主要功能就是对数据进行清洗使它能够有效地满足数据流挖掘算法对数据质量的需求。将清洗和规整之后 SNMP 数据传输到数据分析模块中。本工作通过大量实验发现数据降维技术分析 SNMP 数据是非常有效的。在 RealMon 中,实现了基于 PCA(principle component analysis)的方法对流量数据的分析。通过采用该方法,数据分析模块能够实时地对 SNMP 数据流进行监测而且一旦有数据异常发生,该模块将发出告警信号。当实时交互控制台接收到告警信号时,它将及时通知网络管理员并报告异常发生时间及其所在链路。同时,由于数据清洗及数据分析都需要设置相应的参数,当网络管理员认为系统存在一定的偏差时,他可以通过控制台调整相应的参数使系统正确运行。

2.2　数据清洗方法

在该系统中,数据清洗的主要任务是对原始数据进行规整并使它们能够被正确地分析。本文提出的数据清洗算法是简单有效的,如果在清洗过程中能够考虑更多的因素,将会得到更好的方案。

调整时间间隔:当一条数据流的数据包 Package $(Group_1, Task_1, Time_1, Value_1)$ 到达时,首先将该数据包放入这条数据流对应的组 $Group_1$ 中。然后根据该组的信息,决定下一步的清洗流程。如果该组中其他数据流没有实时数据,就将数据包中的 $Value_1$ 值填入内存数据块对应的 $Task_1$ 中,并重置采样时间 $ST_{new} = ST_{old} + CT$,其中 CT 表示采样周期的长度。随后根据新的采样时间划定范围 $[ST_{new} - \alpha * CT, ST_{new} + (1 - \alpha) * CT]$,其中 α 为用户设置的参数,用于调节范围的选取。如该组其它流的数据在这个时间范围内到达,就认为这些数据是在同一个采样周期采集的,当同一个 Task_ ID 有多个数据包在同一个范围内到达,表示选取的生成时间 $Time_1$ 最靠近采样时间 ST_{new} 的数据包。

填补缺失值:在 SNMP 数据包传输的过程中,由于网络的不稳定导致的数据缺失每时每刻都有可能发生,而需要及时地填补数据缺失。对于填补方法,有多种选择:

(1) 基于历史数据的方法:

$$V_T = (\alpha_1 V_{T-1} + \alpha_2 V_{T-2} + \cdots + \alpha_n V_{T-n}) / (\alpha_1 + \alpha_2 + \cdots + \alpha_n)。 \quad (1)$$

其中,V_T 为缺失值,$V_{T-1} \sim V_{T-n}$ 为系统保存的同一条数据流的历史数据,$\alpha_1 \sim \alpha_n$ 为每个历史数据对应的权重。

(2) 基于 SVD(singular value decomposition)的方法:该方法保存同一组 m 条链路 n 个时间段的流量数据以组成矩阵 $\boldsymbol{A}_{m \times n}$,通过 SVD 方法填补缺失数据:

$$\boldsymbol{A}_{m\times n} = \boldsymbol{A}_{m\times r}\boldsymbol{\Sigma}_{r\times r}\boldsymbol{V}^{\mathrm{T}}_{n\times r}, \quad (2)$$

$$\boldsymbol{V}'_{\mathrm{new}} = k'_1\ \boldsymbol{V}'_1 + k'_2\ \boldsymbol{V}'_2 + \cdots + k'_r\ \boldsymbol{V}'_r, \quad (3)$$

$$\boldsymbol{V}_{\mathrm{new}} = k'_1\ \boldsymbol{V}_1 + k'_2\ \boldsymbol{V}_2 + \cdots + k'_r\ \boldsymbol{V}_r. \quad (4)$$

其中式(2)为标准 SVD 分解,矩阵 $\boldsymbol{V}[v_1, v_2 \cdots v_r]$ 中每一列为矩阵 $\boldsymbol{A}$ 的单位特征向量,$\boldsymbol{V}'_{\mathrm{new}} = (v_1 \cdots v_{j-1}, v_{j+1} \cdots v_m)$ 为同一组 m 条链路的实时数据,其中数据 v_j 缺失,$\boldsymbol{V}'$ 为不包含缺失值对应行的 $m-1$ 维特征向量。当收到存在缺失值的数据 $\boldsymbol{V}'_{\mathrm{new}}$ 后,首先应用式(3)估算特征值 k'_i, $i = 1 \sim r$,然后将 k'_i 代入式(4)中,计算出完整的向量 $\boldsymbol{V}_{\mathrm{new}}$ 以补充缺失值。

以上方法都能有效地填补缺失值,但由于 RealMon 系统需要实时分析大量数据,为了节省系统开销,本文采用较小时间和空间复杂度的基于历史数据的方法来填充缺失数据。

规整数据的数量级:由于同一组中的不同数据流的数量级相差很大,需要对 SNMP 数据的数量级进行调整。图 3 给出了算法的描述,算法中 α 和 β 为用户设置的参数,算法的基本策略是通过数据采样的方式实时估算每一条数据流的流量均值,然后将原始数据除以均值来对数量级进行调整。当一条流的实时数据与均值差异较小时,可以降低数据的采样率,反之将提高采样率。同时,为了避免因均值波动而导致规整后的数据异常,本文设定了衰减系数 β 来动态调整历史数据和实时数据的权重。

```
//调整数量级算法
float AdjustMag(float V_cur)
//V_cur为新到的数据,V_avg为已有数据的均值
//SampleRate 为数据采样率,T 为所有周期数
//N 为常量
  if(需要对 V_cur进行采样)
    //V_cur变化超过预设值
    if( V_cur > (1+β) V_avg || V_cur < (1-β) V_avg);
      //增加采样频率,重新计算均值
      SampleRate = SampleRate * (β+1);
      V_avg = ( V_avg * T * α + V_cur )/( T * α + 1);
      T++;
    //否则,降低采样率
    else SampleRate = SampleRate/(β+1)
Return V_cur * N/V_avg
```

图 3 调整数量级算法

Fig.3 Algorithm for adjusting magnitude

2.3 数据分析算法

在已有的工作中,研究者提出利用不同的降维技术对数据流进行分析的方法。本工作也从实验中发现,这些算法能非常有效地处理 SNMP 流量数据。文献[7]提出通过基于 PCA 的技术来检测数据流之间的异常。该方法通过主向量分析的策略计算数据流之间隐变(hidden variable)的数量。当隐变量的数量发生变化时,多条数据流之间已有的关联关系将会发生变化,这表明数据流之间可能存在了异常。在文献[8]中,作者提出了通过离散傅立叶变换(discrete fourier transform)检测数据流之间关联关系的方法。在该算法中,每条数据流被划分成滑动窗口(sliding window)及基本窗口(basic window)。计算每个基本窗口的傅立叶系数,并通过分析这些系数来实时判断数据流之间的关联关系。在文献[6]中,对已有的数据降维方法做出了总结,并提出了对数据流进行分段计算相似性的方法 APCA (adaptive piecewise constant approximation)。在 RealMon 中,由于同组的数据存在着一定的相似性,采用基于 PCA 的数据流分析算法来检测异常。

3 实验结果

本章介绍 RealMon 系统在模拟环境下的运行结果。在一台 2.4 G CPU,1 G 内存,操作系统为 RedHat Linux 的微型机上实验 RealMon 系统。

由于实验室的网络无法同上海电信的内部网络相连接,只能在模拟环境中验证系统的有效性。首先手工采集电信的 SNMP 数据并存放在磁盘中,然后通过程序模拟 SNMP 数据的实时发送过程,同时运行 RealMon 来接受并分析这些 SNMP 数据。

表 1 列出了图 1 中的数据在清洗后的结果,从结果中可以看到,RealMon 填补了 Task1 的缺失数据,并动态地规整了所有数据的时间属性,最后一列是调整了数量级之后的流量数据。尽管调整之后的流量数据与原始值有较大的变化,但这些数据仍能有效地保留原始数据中存在的关联性。对规整后的数据进行分析能够有效地找出链路中的异常。

表 1 规整后数据

Table 1 Data after cleansing

Task_ID	Time(h-m-s)	Value	After Adjust
Task1	01-13-59	249 121	8 004
Task2	01-13-59	1 314 532 656	8 153
Task3	01-13-59	33 060	6 744
Task4	01-13-59	52 772 272	7 620
Task1	01-18-59	253 222	8 135
Task2	01-18-59	1 316 490 144	8 165
Task3	01-18-59	32 873	6 706
Task4	01-18-59	52 498 624	7 581
Task1	01-23-59	249 121	8 004
Task2	01-23-59	1 278 637 720	7 931
Task3	01-23-59	32 198	6 568
Task4	01-23-59	51 935 728	7 500

图 4 给出了 RealMon 系统的运行结果。原始的 SNMP 流量数据在图 4(a)中展示。该图显示的是监测同一条链路的 3 条不同类型的 SNMP 数据流。可以看出尽管这些数据流的变化趋势非常相似,但它们的数量级分别为 108,109,104,差异非常大。图 4(b)显示了经 RealMon 数据清洗模块处理之后数据流。在该组数据流中存在着 2 个异常,这些异常在图中用圈标示。图 4(c)显示了链路隐变量的变化,通过它来监测链路。从图中发现,这些异常能够有效地被 RealMon 系统所检测。另外,在该组数据中存在着许多瞬时的数据突增点,这些点被称为毛刺点,只是电信网络中的链路噪音,告警这样的毛刺点并没有实际应用价值。本文提出的系统能够有效地忽略这些小的链路噪音。

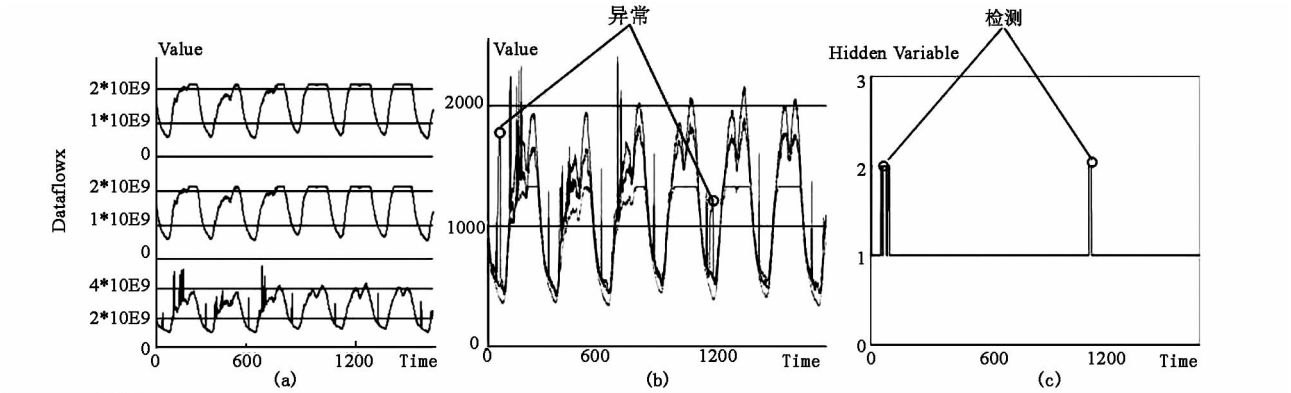


图 4　系统运行结果
(a)原始数据; (b)清洗后数据; (c)异常检测
Fig.4　System result
(a) raw data; (b) data after cleansing; (c) outlier detection

4　结语

本文提出了一个新颖的数据流监测系统 RealMon,它的主要功能是在低质量的 SNMP 数据流中通过分析链路中的关联关系来实时检测数据异常。在 RealMon 中,通过应用基于降维技术的数据流分析算法来有效地监测电信网络的数据。同时,由于原始数据存在着许多数据质量问题,在系统中集成了数据流的实时清洗算法。RealMon 是一个同时具有数据流分析功能和数据流清洗功能的系统。在模拟环境中的实验表明,RealMon 系统能够有效地查找 SNMP 流量数据中的异常情况。希望通过开发 RealMon 来说明数据流清洗和数据流分析在实际应用中的重要性。

参考文献:

[1] HOKE E, SUN J, FALOUTSOS C. InteMon: Intelligent system monitoring on large clusters[C]// D Umeshwar, W Young. Proceeding of 32nd International conference on Very large data bases, September 12-15, 2006. Seoul, Korea: ACM, 2006: 1239-1242.

[2] MOORE J, CHASE J, RANGANATHAN P. Weatherman: Automated, online, and predictive thermal mapping and management for data centers[C]// G Anastasios, S Rizos. IEEE International Conference on Autonomic Computing (ICAC'2006), June 2006. Dublin, Ireland: American Scientific, 2006: 155-164.

[3] GUO Z, ZHOU A. Research on data quality and data cleaning: a survey[J]. Journal of Software, 2002, 13(1): 2076-2082.

[4] ELNAHRAWY E, NATH B. Cleaning and querying noisy sensors[C]// ACM. Proceeding of the 2nd International conference on Wireless Sensor Networks and Applications, Sep19, 2003. San Diego, U.S: ACM Press, 2003: 78-87.

[5] JEFFERY R, GAROFALAKIS M, FRANKLIN J. Adaptive cleaning for RFID data streams[C]// D Umeshwar, WYoung. Proceedings of 32nd International conference on Very large data bases, September 12-15, 2006. Seoul, Korea: ACM Press, 2006: 163-174.

[6] KEOGH E, CHAKRABARTI K, PAZZANI M, et al. Locally adaptive dimensionality reduction for indexing large time series databases[C]// ACM. In Proceeding of the 20th International conference on management of data, May 21-24, 2001. Santa Barbara, California: ACM Press, 2001: 151-162.

[7] PAPADIMITRIOU S, SUN J, FALOUTSOS C. Streaming pattern discovery in multiple time-series[C]// H Klemens, J Christian. Proceeding of 31st international conference on Very large data bases, Aug. 30 - Sep. 2, 2005. Trondheim, Norway: ACM Press, 2005: 697-708.

[8] ZHU Y, SHASHA D. StatStream: Statistical monitoring of thousands of data streams in real time[C]// W Gerhard. Proceeding of 28th International Conference on Very Large Databases, August 20-23, 2002. Hong Kong, China: Morgan Kaufmann, 2002: 358-369.

(编辑:孙培芹)