适用于协议特征提取的关联规则改进算法

龙 文，马 坤，辛 阳，杨义先

(北京邮电大学网络与交换技术国家重点实验室 北京 海淀区 100876;
北京邮电大学网络与信息攻防技术教育部重点实验室 北京 海淀区 100876; 北京邮电大学灾备技术国家工程实验室 北京 海淀区 100876)

【摘要】借鉴关联规则挖掘的思想，引入序列项目集的概念，使算法能够处理集合事物和具有序列特性的项目；通过递推的方法依次得出不同长度的特征字段，并利用偏移属性集加以约束去除无效字段，有效控制约束频繁集的规模；最后依据选择策略从约束频繁集中选出最终的特征字段。实验结果表明只要选取合适的参数，用该方法提取协议特征是行之有效的。

关 键 词 关联规则; 数据挖掘; 协议识别; 特征提取
中图分类号 TN915.08 **文献标识码** A doi:10.3969/j.issn.1001-0548.2010.02.032

Improved Association Rules Algorithm for Protocol Signatures Extracting

LONG Wen, MA Kun, XIN Yang, and YANG Yi-xian

(State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications Haiding Beijing 100876;
Key Laboratory of Network and Information Attack and Defence Technology of MOE, Beijing University of Posts and Telecommunications Haiding Beijing 100876; National Engineering Laboratory for Disaster Backup and Recovery, Beijing University of Posts and Telecommunications Haiding Beijing 100876)

Abstract The notion of sequence itemset is introduced for algorithm to deal with permutations items or transactions of itemsets. A recursive method is provided to generate signatures with different length in turn. The algorithm utilizes offset attributor set to restrict and remove ineffective itemsets. According to selection principle, the optimal constrained frequency sequence itemset as signatures can be found. The result shows that the signatures extracted by this algorithm are reasonable and effective.

Key words association rules; data mining; protocol identification; signatures extracting

目前基于应用层特征字段的检测技术已成为协议识别的主流方法[1-2]，但特征提取还缺乏有效的方法，主要还是依赖于人工方式。随着业务种类快速增长和越来越多地采用非公开的自定义协议，特征字段的提取变得日益困难，工作量也急剧增大。本文借鉴关联规则挖掘的思想，引入序列项目集和偏移属性集的概念，提出了一种采用偏移约束的应用层特征提取算法，可以在离线方式下实现特征字段的自动提取。

1 相关工作

数据挖掘已被应用于网络数据处理的很多领域，但目前仍然没有很有效的协议特征提取方法。文献[3]把数据流中的IP包进行重组，利用还原后的应用层数据生成训练矢量，然后采用朴素贝叶斯、AdaBoost等算法进行样本训练，并把生成的模型用于协议检测，但这种方法本质上是采用概率统计的方式，并不能直接给出各协议的特征字段。文献[4]提出了SA算法用于提取网络攻击的特征值，但不加约束的挖掘会导致规模无谓扩大，产生许多无效的特征字段。

关联规则是应用较多的一种数据挖掘方法，但关联规则挖掘的事务(transaction)是元素，而特征提取需要处理的对象——数据流(由五元组，即源IP、源端口、传输层协议、目的IP和目的端口决定)是一个数据包的集合。同时，关联规则挖掘是研究项目之间的组合关系，而应用层数据是有序列特性的，进而也就无法反映特征字段相对于应用层数据的偏移。因此类似于文献[5]的算法并不能直接适用于特征提取。

收稿日期：2008 – 09 – 12; 修回日期：2009 – 04 – 20
基金项目：国家自然科学基金(60821001、U0835001、60803157)
作者简介：龙 文(1980 –)，男，博士生，主要从事网络信息安全等方面的研究.

2 概念定义

设 $I=\{\infty i_1,\infty i_2,\cdots,\infty i_W\}$ 为包含 W 个项目的重集；事务集 $D=\{T_1,T_2,\cdots,T_M\}$ 由 M 个具有唯一标示TID的事务组成。每个事务 $T_m=\{t_{m1},t_{m2},\cdots t_{mN}\}$ ($1\leqslant m\leqslant M$) 为包含 N 个序列项目集(sequence itemset)的集合，其中序列项目集定义如下。

定义 1 序列项目集是 I 上的一个 r 重排列 $(1\leqslant r\leqslant R)$，记为 $\alpha=\alpha_1\alpha_2\cdots\alpha_r$，其中 $\alpha_i(1\leqslant i\leqslant r)$ 是 I 中的一个项目。序列项目集的长度是它所包含的项目数，记为 $\|\alpha\|$。具有 r 长度的序列项目集称为 r-序列项目集。

定义 2 设序列项目集 $\alpha=\alpha_1\alpha_2\cdots\alpha_{r1}$，$\beta=\beta_1\beta_2\cdots\beta_{r2}$，其中 $r_1\leqslant r_2$；若存在自然数 i，使得 $\alpha_1=\beta_{1+i},\alpha_2=\beta_{2+i},\cdots,\alpha_{r1}=\beta_{r1+i}$，则称 α 是 β 的子序列，或 β 包含 α，记为 $\alpha=\mathrm{sub}(\beta,i)$，$i$ 称之为 α 相对于 β 的偏移。若事务 T_m 中存在序列项目集 β，使得 α 是 β 的子序列，则称 α 包含于事务 T_m，记为 $\alpha\subseteq T_m$。

定义 3 序列项目集 α 在事务集 D 上的支持度是包含 α 的事务在 D 中所占的百分比，即：

$$\mathrm{support}(\alpha)=\|\{T_m\mid\alpha\subseteq T_m,T_m\in D\}\|/M$$

所有不小于用户指定的最小支持度 MIN_SUP 的序列项目集。称为频繁序列项目集。不被其他任何频繁序列项目集包含的频繁序列项目集称为最大频繁序列项目集。频繁序列项目集简称为频繁集。

定义 4 序列项目集 α 相对于事务 T_m 的偏移属性集 $\mathrm{attr}(\alpha,T_m)$ 是 α 相对于 T_m 中所有包含 α 的序列项目集的偏移值的并集，即：

$$\mathrm{attr}(\alpha,T_m)=\bigcup_{n=1}^{N}\{i\mid\alpha=\mathrm{sub}(t_{mn},i),t_{mn}\in T_m\}$$

序列项目集 α 相对于事务数据集 D 的偏移属性集 $\mathrm{attr}(\alpha,D)$ 是 α 相对于 D 中所有包含 α 的事务的偏移属性集的交集，即 $\mathrm{attr}(\alpha,D)=\bigcap_{m=1}^{M}\mathrm{attr}(\alpha,T_m)$，其中 $T_m\in D$。

3 算法描述

假设有 M 条同一应用协议的数据流，每条流中包含 N 个具有应用层负载的数据包，负载的每个字节都取自字符集 Σ，那么特征提取是要在这些数据流中找出出现概率较高的字段及其偏移位置。

如果把字符集 Σ 中的字符作为 I 中的项目，数据流作为事务集 D 中的事务，那么 Σ 中 $r(1\leqslant r\leqslant R)$ 个字符排列成的字段就对应一个序列项目集，每条数据流中的数据包负载就对应事务中包含的序列项目集，R 就是负载的最大长度1 460。偏移属性集 $\mathrm{attr}(\alpha,D)$ 记录字段 α 在不同数据流中的固定偏移位置，如果 $\mathrm{attr}(\alpha,D)=\varnothing$，表示 α 的偏移位置不固定；否则 α 有固定偏移位置。如 $\mathrm{attr}(\alpha,D)=\{3\}$，表示在每条包含 α 的数据流中，α 至少在一个数据包中相对负载首部的偏移为3。

整个特征提取算法可以抽象为以下3个步骤：

(1) 找出所有支持度不小于 MIN_SUP 的序列项目集，得到频繁集 L，即找出所有符合以下条件的字段 p：p 中的每个字符都取值于 Σ；p 至少在 $M\times$MIN_SUP 条数据流中的数据包负载中出现。

(2) 去除 L 中所有不满足约束条件的序列项目集，余下的元素组成约束频繁集 E，约束条件依据 L 中元素的偏移属性集。

(3) 依据选择策略，从 E 的元素中选出最终的特征字段。

3.1 生成频繁集 L

由于频繁集 L 可以划分为若干互不相交的子集，即 $L=\bigcup_{r=1}^{1\,460}L_r$，其中 L_r 表示长度为 r 的频繁集组成的集合，因此可以先计算 $L_1=\{\alpha\mid\mathrm{support}(\alpha)\geqslant\mathrm{MIN\_SUP},\|\alpha\|=1\}$，然后采用递推的方法由 $L_r(1\leqslant r<1\,460)$ 生成 L_{r+1}，将这些不同长度的频繁集集合合并就可以得到 L。在递推过程中，应同时计算 L_r 中每个频繁集的偏移属性集。

为充分利用 L_r 的信息，减少 L_{r+1} 的计算量，可以先通过 L_r 生成候选集 C_{r+1}，然后从 C_{r+1} 中选出支持度不小于 MIN_SUP 的项目集生成 L_{r+1}。候选集 C_{r+1} 采用如下算法生成。

算法1　候选集生成算法。

```
FOR all  α = α1α2⋯αr ∈ Lr
    FOR all  β = β1β2⋯βr ∈ Lr
        IF  r = 1  THEN
            μ = α1β1
        ELSE IF  α2 = β1,⋯,αr = βr−1  THEN
            μ = α1α2⋯αrβr
        ELSE
            Continue;
        将 μ 加入候选集 Cr+1
    END
END
```

算法是基于以下思想：$\mu=\mu_1\mu_2\cdots\mu_{r+1}\in L_{r+1}$ 的

必要条件是 μ 的两个子串 $\alpha=\mu_1\mu_2\cdots\mu_r$ 和 $\beta=\mu_2\mu_2\cdots\mu_{r+1}$ 都是 L_r 中的元素。因此候选集生成算法就是把 L_r 中前r–1字节和后r–1字节相同的两个字段进行合并，生成长度为r+1的字段加入候选集。

3.2 生成约束频繁集E

通过3.1节的处理，生成了频繁集 L，但 L 中包含的元素并不都是有价值的，应该去除 L 中长度过短(小于设定值 S)的非固定偏移字段，同时保留有固定偏移的最长特征字段和满足偏移约束条件的非最长特征字段。最长特征字段是指那些不被 L 中其他字段所包含的字段，用术语描述，也就是利用偏移属性集从 L 中选出以下3类频繁集生成 E：(1) 没有固定偏移(即 $\text{attr}(\alpha,D)=\varnothing$)，但长度不小于 S 的最大频繁集；(2) 有固定偏移(即 $\text{attr}(\alpha,D)\neq\varnothing$)的最大频繁集；(3) 有固定偏移，但经过偏移约束计算后 $\text{attr}(\alpha,D)\neq\varnothing$ 的非最大频繁集。

算法2 偏移约束算法。

FOR ALL $\beta\in L_{r+1}$ DO BEGIN

IF $\text{support}(\beta)<\text{support}(\alpha)$

Continue;

IF $\alpha=\text{sub}(\beta,0)$ THEN

$\text{attr}(\alpha,D)=\text{attr}(\alpha,D)-\text{attr}(\alpha,D)\bigcap\text{attr}(\beta,D)$

ELSE IF $\alpha=\text{sub}(\beta,1)$

$\text{attr}(\alpha,D)=\text{attr}(\alpha,D)-\text{attr}(\alpha,D)\bigcap\text{attr}^{+1}(\beta,D)$

END

其中 $\text{attr}^{+1}(\beta,D)$ 表示 $\text{attr}(\beta,D)$ 中所有的元素加1。偏移约束算法用于判别频繁集 α 是否有自身独有的偏移位置，而不是由包含 α 的频繁集产生的偏移。如字段“abcdef”的偏移属性集为{10}，而字段“abcde”和“bcdef”的偏移属性集分别为{10}和{4,11}，那么就认为前者不满足约束条件，而后者满足约束条件，因为后者的偏移位置4不是由字段“abcdef”造成的。

3.3 选择策略

该步骤从 E 的元素中选出最终的特征字段，选择策略如下：

(1) 如果 E 中存在有固定偏移的字段，则选取固定偏移字段中长度最大的。(2) 如果 E 中不存在有固定偏移的字段，则选取全部字段中长度最大的。

另外，如果 E 中存在多个具有固定偏移的元素，还可以把这些元素作为项目，重新生成频繁集，那么每个频繁集就对应一个多特征字段的检测规则，具体过程和Apriori算法类似，在此不再赘述。

4 实验结果

为验证算法性能和提取特征的有效性，在某城域网出口网关处随机采集了200多种协议的数据，运用本文算法进行特征提取，并将提取的特征应用于网关旁路的协议识别设备上。实验结果表明，只要选取合适的参数，用该方法提取协议特征是行之有效的。为便于描述，以下选取HTTP、PPLIVE、MAZE、XUNLEI(BT功能)和BANACAST五种典型协议重点说明。

4.1 参数的选择

根据协议识别的实际需要，字符集 Σ 取值区间为[0,255]，支持度SUP_MIN应不小于95%。在不同数据流相互独立的前提下，M 取值越大，特征提取的准确性也就越高，但也就意味着提取时间会更长，实际应用时 M 应不小于100。

图1显示了 $M=100$，N 取不同值时，算法产生的约束频繁集 E 中的元素个数$\lVert E\rVert$。从图中可以看出，当 N 取值逐渐增大时，$\lVert E\rVert$的值将趋于稳定，这说明协议的特征主要出现在初始的几个数据包中，因此 N 不宜过大，以避免处理无用的数据。通常取 $N=8$。

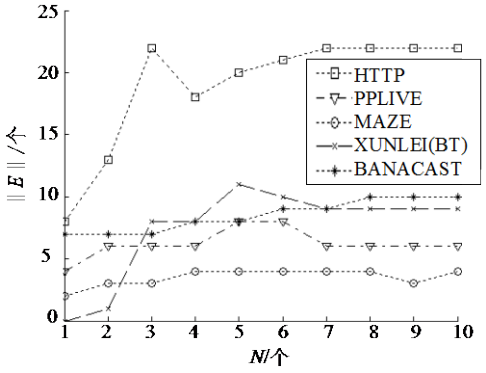

图1 M=100，N取不同值时约束频繁集E的个数

S 用来去除长度过短的非固定偏移字段，对200余种协议的特征字段进行了统计，特征字段共453个，其中非固定偏移字段仅有26个，最短长度为6字节。在实验中，S 的取值统一设置为6。

4.2 效率分析

特征提取的时间效率受数据包长度、负载的字符分布等诸多不确定因素影响，因此算法在处理不同协议时存在较大差异，难以简单地给出时间复杂度。但是，由于特征提取的时间主要消耗在频繁集生成阶段，因此协议频繁集的数量多少会在很大程度上影响特征提取的时间。图2给出了各种协议不同长度的频繁集数量$\lVert L_r\rVert$随长度r变化的情况。

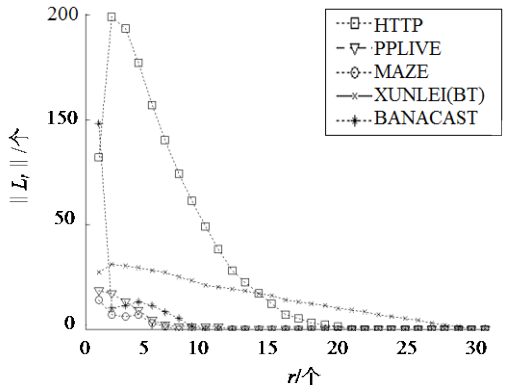

图2　长度r不同时的频繁集个数

从图中可以看出，像HTTP这类基于文本的协议，由于存在大量频繁出现的头字段(Accept、User-Agent等)，因此$|L_r|$取值较大，特征提取时间会较长；而其他类似于PPLIVE、MAZE等采用二进制编码的协议，由于具有严格的格式规定，特征字段较少，$|L_r|$的取值会随r的增大而急剧减少，因此特征提取时间会相对较短。

4.3　检测准确性

表1列出了采用算法生成的特征字段应用在城域网出口协议识别设备中运行一周的结果。其中PPLIVE的两个特征字段依据4.2中所述进一步生成多特征检测规则。

表1　不同协议的应用层特征字段及检测结果

应用	特征字段	偏移	错误率/(%)	召回率/(%)
HTTP	48 54 54 50 2F 31 2E	0	0.012	96.48
PPLIVE	E9 03	0	0.007	99.82
	01 98 AB 01 02	3		
MAZE	10 00 00 00	0	0.016	99.52
迅雷	13 42 69 74 54 6F 72 72 65 6E 74 20 70 72 6F 74 6F 63 6F 6C 65 78 00 00 00 00 00 00 01	0	0.002	97.23
BANACAST	48 00 00 00 1A 02 CA BA	0	0.005	99.93

5　结 束 语

本文通过递推的方法依次得出不同长度的特征字段，并利用偏移属性集加以约束去除无效的字段，可以有效控制特征字段的规模；而且只需做少量改动就可以适用于IDS等其他采用协议特征的检测系统。在未来工作中，可以在以下方面作进一步的研究：(1) 对多协议应用的特征提取。如果某种应用同时采用多种协议进行交互，会影响频繁集的生成，如何合理区分各协议和设定支持度是研究的难点。(2) 与数据包长度等其他要素相结合，进一步提高约束的条件，提高提取特征的有效性和检测的进度。

参　考　文　献

[1] SEN S, SPATSCHECK O, WANG D. Accurate, scalable in-network identification of P2P traffic using application signatures[C]//WWW 2004: Proceedings of Thirteenth International World Wide Web Conference. New York: ACM Press, 2004: 512-521.

[2] HAMZA D, SANDRINE V, DAVID R. A markovian signature-based approach to IP traffic classification[C]// MineNet'07: Proceedings of the Third Annual ACM Workshop on Mining Network Data. San Diego: ACM Press, 2007: 29-34.

[3] HAFFNER P, SEN S, SPATSCHECK O, et al. ACAS: Automated construction of application signatures[C]// Proceedings of ACM SIGCOMM 2005 Workshops: Conference on Computer Communications. Philadelphia: ACM Press, 2005: 197-202.

[4] HAN Hong, LU Xian-liang. Data mining aided signature discovery in network-based intrusion detection system[J]. ACM SIGOPS Operating Systems Review, 2002, 36(4): 7-13.

[5] AGRAWAL R, IMIELINSKI T, WAMI A S. Mining association rules between sets of items in large databases[C]//Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data. Washington: ACM Press, 1993: 207-216.

[6] MOORE A, ZUEV W. Internet traffic classification using Bayesian analysis techniques[C]//SIGMETRICS 2005: Proceedings of International Conference on Measurement and Modeling of Computer Systems. Banff, AB, Canada: ACM Press, 2005: 50-60.

[7] FANG W, PETERSON L. Inter-AS traffic patterns and their implications[C]//Conference Record of 1999 IEEE Global Telecommunications Conference. Rio de Janeiro: IEEE Press, 1999: 1859-1868.

[8] PITKOW J. Summary of WWW characterizations[J]. World Wide Web, 1999, 2: 3-13.

[9] ZANDER S, NGUYENL T, ARMITAGEL G. Self-learning IP traffic classification based on statistical flow characteristics[C]//PAM 2005: Proceedings of 6th International Workshop on Passive and Active Network Measurement. Boston: Springer Verlag, 2005: 325-328.

[10] KRISHNANURTHY B, WANG J. Automated traffic classification for application specific peering[C]//IMW 2002: Proceeds of ACM SIGCOMM Internet Measurement Workshop. Marseille: ACM Press, 2002: 179-180.

编　辑　漆　蓉