零知识下的比特流未知协议分类模型

张凤荔[1] 周洪川[1] 张俊娇[1] 刘 渊[2] 张春瑞[2]
(电子科技大学信息与软件工程学院 成都 611731)[1]
(中国工程物理研究院计算机应用研究所 绵阳 621900)[2]

摘 要 针对在零知识下识别比特流未知协议这一问题,提出了一种协议分类模型。该模型首先利用二进制流的固有特性来计算协议种类个数近似值 K 和初始聚类中心,然后使用改进的 K-Means 聚类算法指定 K 及初始聚类中心以进行聚类,最后使用基于信息熵的混杂度评价方法对聚类结果进行评价,可将评价结果较好的类簇作为一种协议类型进行标记,用于其他分析。使用林肯实验室发布的实验数据进行测试,结果表明该模型能以较高的准确率对未知协议进行分类,基于信息熵的类簇评价方法也具有一定实用性。

关键词 K-Means 聚类,未知协议识别,K 值计算,聚类结果评估
中图法分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.8.008

Unknown Bit-stream Protocol Classification Model with Zero-knowledge

ZHANG Feng-li[1] ZHOU Hong-chuan[1] ZHANG Jun-jiao[1] LIU Yuan[2] ZHANG Chun-rui[2]
(School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China)[1]
(Institute of Computer Application, China Academy of Engineering Physics, Mianyang 621900, China)[2]

Abstract To solve the difficult problem of unknown bit-stream protocol identification with zero knowledge, a protocol classification model was proposed. Firstly, this model calculates the approximation of parameter K and the initial cluster center using the inherent features of bit-stream, then uses the improved K-Means to cluster data set into different clusters by specifying the parameter K and the initial center, and finally evaluates the results of clustering by a hybrid evaluation method based on information entropy. The clusters with good evaluation results can be marked and used to study further. Testing data set published by the Lincoln laboratory shows that unknown bit-stream protocols can be classified with high accuracy by this model, and the evaluation method based on information entropy is also useful and effective.

Keywords K-Means, Unknown protocol identification, K value calculation, Evaluation of clustering results

1 引言

网络信息安全与对抗已成为信息时代备受关注的重要问题。在电子对抗等领域,通信双方使用的协议往往是订制的、非公开的,所截获的通信数据大多是连续的比特流信息;在网络监管等领域,网络通信过程中使用的协议解析工具也会遇到许多无法解析的比特流协议。对于这些协议,协议分析者没有任何先验知识,解析这些完全未知的协议十分困难[1]。由于不知道与未知协议相关的任何信息,如格式说明、开发文档等,可将其视为在零知识条件下解决未知协议识别问题。

对于在零知识条件下识别比特流未知协议的问题,宋等[2]在捕获的数据是连续比特流的情况下,通过改进 AC 算法,提出了一种"基于指纹特征的数据帧定界方法";金等[3]通过频繁串和关联规则挖掘,计算频繁串最小位置差,提出了一种基于关联规则的帧头识别技术。王等[4]在捕获的数据为数据帧(如 ppp 帧、以太网帧等)的情况下提出了一种"基于关联规则识别特定环境下未知协议的方法",该方法通过挖掘关联规则来识别和标识未知协议。这些方案都能在相应的设定条件下得到较好的效果,能为识别未知协议提供有用参考,但它们均是在单协议假定下进行分析和实验的,在实际应用环境下,捕获得到的未知协议数据帧往往是多种协议混合的。而对于如何在零知识情况下将多协议分类为单协议这一至关重要的问题,相关方面的研究还较少。

为解决该问题,本文采用离线分析的方式,先搜集大量的未知协议数据帧(多种协议类型),然后基于聚类理论提出一种协议分类模型。该模型首先对输入的多种类型数据帧进行预处理,利用二进制流的固有特性计算输入数据的协议类型个数的近似值 K 及初始聚类中心;然后使用改进的 K-Means 聚类算法进行聚类分析;最后用基于熵的结果评价方法对聚类结果进行评价,将评价好的类簇加入结果集。其中,关键问题是聚类个数 K 值的计算方法及聚类结果的评价方法。实验表明,该模型执行效率高,使用简单,且具有较高准确率。

到稿日期:2015-07-02 返修日期:2015-10-18 本文受 NASF 基金资助项目(U1230106),中国工程物理研究院科学技术发展基金项目(2012A0403021),四川省科技计划资助项目(2014GZ0109,2015KZ002),国家自然基金项目(61472064)资助。

张凤荔(1963-),女,博士,教授,博士生导师,主要研究方向为网络与信息安全,E-mail:fzhang@uestc.edu.cn;**周洪川**(1988-),男,硕士生,主要研究方向为网络与信息安全;**张俊娇**(1990-),女,硕士生,主要研究方向为网络与信息安全;**刘 渊**(1974-),男,硕士,高级工程师,主要研究方向为网络与信息安全;**张春瑞**(1980-),男,硕士,高级工程师,主要研究方向为网络与信息安全。

本文第 2 节给出协议分类模型设计；第 3 节是实验及结果分析；最后是总结及展望。

1.1 未知协议识别

网络协议指的是为在指定的网络环境下进行数据交换而建立的规则、标准或约定的集合。网络协议由 3 个要素组成：语义、语法和时序。语义是解释控制信息每部分的意义，它规定了需要发出何种控制信息以及完成的动作与做出什么样的响应；语法是用户数据和控制信息的结构与格式以及数据出现的顺序；时序是对事件发生顺序的详细说明[5]。

本文的未知协议指的是没有公开其协议开发文档和协议格式说明的协议，通常是用某些特定的网络设备捕获到的二进制信息。本文假设要分析的二进制信息是以帧为单位进行捕获的，即使捕获到的协议数据是连续的比特流，也可以使用文献[2]、文献[3]或其他方式通过帧切分得到。

协议识别技术经历长期的发展，取得了一系列的成果，典型的代表就是各种协议识别工具的出现，如免费的有 tcpdump，Wireshark 等，商业的有 Capsa Enterprise，Clarified Analyzer 等。它们都能对已知格式的协议进行快速、准确的识别。从识别技术上看，主要有基于端口的协议识别、深度包检测技术和基于模式匹配的协议识别[6]。这些技术大多用于已知协议结构的识别。对于未知协议结构的协议，传统的方式是使用协议逆向工程相关技术对协议进行信息提取[7-9]，根据分析对象的不同，其大致可分为“基于报文序列分析”和“基于指令执行序列分析”两大类，典型的代表有：PI 项目、Discoverer 系统[10]、RolePlayer 项目[11]、Ploygot 项目[12] 等。这些项目都能在一定程度上实现对完全未知协议的识别，但耗费的人力多、时间长，其中 PI 项目还存在无法获取域的语意信息以及对结构嵌套的协议识别不足等问题；RolePlayer 存在依赖先验知识和识别复杂协议效果不佳等问题；Discoverer 系统对协议状态信息获取不理想；Ploygot 项目不能很好地解决域与域之间的关系。

1.2 相关基础理论

(a)聚类算法分析

在零知识条件下，将多协议数据帧分类为单协议数据帧属于无监督的分类过程(聚类)。聚类算法可分为以下几类：基于划分的聚类算法(如 K-Means)、基于层次的聚类算法(如 CURE)、基于密度的聚类算法(如 DBSCAN)、基于网格的聚类算法(如 OPTIGRID)、基于神经网络的聚类算法(如 SOM)、基于统计学的聚类算法(如 COBWEB)以及其他聚类算法。这些聚类算法各有其适用场合和局限性，而本文的协议分类需要一种快速、高效的算法，帧与帧之间的距离可以很好地通过帧间的相似度来衡量，因此在众多的聚类算法中，本文优先选择了 K-Means 算法来处理二进制协议的分类问题，后面的实验部分对各种典型的聚类算法进行了对比分析。

K-Means 聚类算法[13,14]是一种经典的基于划分的算法，该算法简单快速、应用广泛，时间复杂度为 $O(ntk)$，n 为数据规模，t 为迭代次数，k 为指定的类簇个数，在数据量较大的情况下，该算法的时间复杂度接近线性。其基本思想为：对于给定的 n 个对象$\{x_1, x_2, \cdots, x_n\}$，找到 K 个聚类中心$\{m_1, m_2, \cdots, m_K\}$，满足每个数据对象与它最近中心的距离平方和最小。通常将误差平方和函数作为目标函数，记为 E，计算公式如下：

$$E=\sum_{j=1}^{K}\sum_{x\in C_j}\| x-m_j \|^2 \tag{1}$$

其中，K 表示需要得到的类簇个数，m_j 表示类簇 C_j 的均值，C_j 表示第 j 个类簇。在本文中，每个对象是一个数据帧。

K-Means 聚类算法在文本的应用中主要具有以下局限性：1)算法对初始中心的选取非常敏感，容易陷入局部最优解；2)需要用户指定聚类个数 K 值。对此，本文设计了相应算法对其进行改进。

(b)聚类结果评价——信息熵

评价聚类结果的质量是比较困难的，在实验过程中，可以使用准确率、召回率、F 值等指标来对聚类模型进行评价，但在实际应用过程中，聚类结果的好坏是难以判断的。信息熵是独立于算法和输入数据集的，它能评价簇中数据对象来源分布的均衡程度，来源越均衡，混乱度越高，熵值越大，说明簇的质量越差。因此本文采用信息熵的相关理论来对聚类的结果进行评价。

信息熵是对一个随机变量的信息和不确定性的度量，也叫 Shannon 熵[15]，它采用数值形式来表达随机变量取值的不确定程度，以刻画信息含量的多少。信息熵 $E(x)$定义为：

$$E(x)=-\sum_{x\in S(x)}p(x)\ln(p(x)) \tag{2}$$

其中，x 是随机变量，$S(x)$是 x 可能取值的集合，$p(x)$是 x 的概率函数，ln 表示以 e 为底的对数。式(2)中 $E(x)$的单位为 bit，若对数以 2 或 10 为底，则单位为 nat 或 Harley[16]。

信息熵可用来测量一个系统的“混乱”程度。熵值越大，说明系统中的信息越混乱；熵值越小，则说明系统中的信息越“单一”或“纯净”。

对于未知协议分类来说，聚类之后往往难以评价类簇的好坏，可用信息熵来衡量每个类簇的信息混杂度。对于类簇的所有数据帧，以 8bits(或 8bits 的整数倍)为处理单元，以数据帧条数为行数，以数据帧长度为列数，构成二维矩阵，计算每个类簇的各个列的信息熵并将其作为衡量类簇是否纯净的依据。

2 协议分类模型设计

本文提出的未知协议分类模型的功能是实现将多类型的协议数据帧分类为单类型的数据帧。需要解决的关键问题是：K 值的计算和聚类时初始中心的选取、聚类结果的评价方法及合理的数据预处理方法。模型的输入为多协议数据帧，输出为 k 个单协议类簇，如图 1 所示。

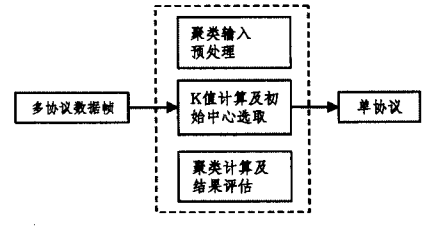

图 1 未知协议聚类模型

2.1 数据预处理方法

数据预处理阶段将二进制数据帧转换为十六进制格式，然后以 8bits(1 字节)为处理单元，比如 1111111100010001 将转换为 ff11，ff 和 11 为处理单元。将输入的数据帧构造成一个 n 行、m 列的二维矩阵，n 表示输入的数据帧的行数，m 表示截取数据帧的前 m 个字节。预处理算法的相关说明及具

体描述如下。

处理单元：指的是二维矩阵的元素，本文算法中处理单元为 8bits(1 字节)。

矩阵的行：对应的是输入数据帧的条数，如输入的数据帧条数为 n，则预处理后二维矩阵的行数为 n。

矩阵的列：对应的是截取输入数据帧前面部分的字节个数，如截取数据帧前 m 个字节，则预处理二维矩阵的列数为 m。

算法 1 数据预处理算法—PABP(Pretreatment Algorithm for Bit-stream Protocol)

参数：m(指定取数据帧的前 m 字节)

输入：n 条数据帧

输出：n 行、m 列的二维矩阵

步骤：

1. 计算输入数据帧的条数，为其赋值 n；
2. 初始化 n 行、m 列的处理单元对象二维数组，记为 a[n][m]；
3. for 循环遍历每条数据帧
4. 　　将数据帧按 8 个 bit 分割为 m 个字节
5. 　　分别将每个字节赋给矩阵对应行的每个元素
6. 输出二维矩阵

2.2 K 值计算及初始中心选取方法

K-Means 聚类算法的 K 值计算及初始中心的选取是本文未知协议分类模型的关键问题。其计算依据是：同一种协议的数据帧之间存在一定相似性，相似性表现为在某些位置会出现相同的字符。如图 2 中，前 6 条数据帧为同一种协议，在第 2 字节和第 7 字节处出现的是相同的字符 11 和 bb。

```
** 11 ** ** ** ** bb ** ** ** ** ** ** ** ** ** ... **
** 11 ** ** ** ** bb ** ** ** ** ** ** ** ** ** ... **
** 11 ** ** ** ** bb ** ** ** ** ** ** ** ** ** ... **
** 11 ** ** ** ** bb ** ** ** ** ** ** ** ** ** ... **
** 11 ** ** ** ** bb ** ** ** ** ** ** ** ** ** ... **
** 11 ** ** ** ** bb ** ** ** ** ** ** ** ** ** ... **
** ** ** ** 22 ** ** ** ** ** ** ** ** ** ** ** ... **
** ** ** ** 22 ** ** ** ** ** ** ** ** ** ** ** ... **
** ** ** ** 22 ** ** ** ** ** ** ** ** ** ** ** ... **
** ** ** ** 22 ** ** ** ** ** ** ** ** ** ** ** ... **
** ** ** ** 22 ** ** ** ** ** ** ** ** ** ** ** ... **
...
** ** ** ** ** ** kk ** ** ** ** ** ** ** ** ** ... **
```

图 2 多协议数据帧示意图

具体的 K 值计算及初始中心选取方法如下。

定义 1(处理单元对象 OneByte) 对应于矩阵中的元素，属性有：

```
class OneByte {
/**此字节所在行*/
public int row;
/**此字节所在列*/
public int line;
/**此字节在该列中出现的次数*/
public int num;
/**此字节在该列中出现的频率*/
public float frequence;
/**此字节的内容*/
public String oneByte;
/**在该列中，出现此字节的行的编号集合*/
public HashSet<Integer> alist=new HashSet<Integer>();
}
```

定义 2(字节出现的频率 *frequence*) 指的是针对矩阵的某一列而言，某字节出现的次数除以矩阵的行数：

$$fre=num/n \tag{3}$$

定义 3(集合间相似度 *simi*) 定义此值为集合 S_i 与集合 S_j 交集的个数的 2 倍除以两集合个数的和。

$$simi=\frac{2*N(S_i\cap S_j)}{N(S_i)+N(S_j)} \tag{4}$$

参数 *lowestSimilar*：可设定值阈值，用于判断两集合是否合并。若两集合的相似度大于或等于 *lowestSimilar*，则将这两集合合并，否则不合并。该阈值越大，则得到的 K 值越大；该值越小，则得到的 K 值越小。

参数 *min_liminal*、*max_liminal*：可设定值，用于筛选频繁字节，当某个字节出现的频率大于或等于 *min_liminal*，并且小于 *max_liminal* 时，该字节被筛选出来。默认值分别为 0.2 和 0.99。

算法 2 计算 K 值及初始中心算法——AKBP(Algorithm for K-value of Bit-stream Protocol)

参数：min_liminal，max_liminal，lowestSimilar

输入：二维处理对象数组 a[n][m]//预处理的结果矩阵

输出：K 值，初始中心行数集合 W

步骤：

1. 将输入数据建立 OneByte 对象的 n 行、m 列的二维数组，将输入的 a[n][m]的每一个字节的内容赋给 OneByte[n][m]对象的 oneByte 域，并且记录该字节所在的行和列；
2. 循环遍历 OneByte 二维数组；
3. 按列统计，将在该列中字符出现的次数赋给 OneByte 的 num 域，按式(3)计算频率 frequence；
4. 将在该列中字符出现过的行号加入 OneByte 的 alist 集合；
5. 找出每一列中出现频率最高的 OneByte 对象，从第 0 列到第 m－1 列有 m 个；
6. 对这 m 个对象进行筛选，将出现频率小于 min_liminal 和大于或等于 max_liminal 的对象去掉；
7. 将以上筛选出来的每个对象的集合(alist)放入结果集 R 中，用 S 表示 R 的元素，按式(4)求 R 中两两元素 S_i 与 S_j 之间的相似度值，若该值大于或等于 lowestSimilar，则将其合并，直到没有可合并的为止；
8. 结果集 R 中的元素个数即为要求的一次 k 值。从结果集 R 中每个集合里取出一条数据帧即构成初始中心集合 W。

为了更准确地确定 K 值，可设置 *lowestSimilar* 值从 0.1 到 1.0 变化，增加步长为 0.05，分别求出 k 值，以 *lowestSimilar* 的值为 X 轴、K 值为 Y 轴作曲线，由于 K 随 L 的增大而增大，所得的曲线为递增曲线，且真实值 K_0 将在曲线上。为尽可能准确地求得 K，使得所求的 K 与真实值相差不会过大或过小，可根据“拉格朗日中值定理”求出区间[0.1，1.0]内的唯一一个切点 P 作为所求 K 值的坐标。方法是将该曲线的首尾用直线相连，记为 L，求出该直线的斜率 k_L，然后在曲线上找出一个切点 P，使得过点 P 的切线的斜率等于 k_L，切点 P 也可视为以 L 为 X 轴时曲线的极值点；并计算出 P 点的 *lowestSimilar* 值以及对应的 K 值，所得的 K 即为要求解的近似值 K，从与 K 最接近的结果集 R 的每一个元素中各取一条数据帧即可组成初始聚类中心 W。

2.3 改进的 K-Means 聚类算法

由以上方法计算出近似的 K 值和初始聚类中心后，改进 K-Means 算法表现在：1) K 值是从零知识环境中得到的；2) 算法不需随机选择 K 个初始聚类中心，直接指定 K 个初始聚类中心进行聚类，这样可以加快 K-Means 算法的收敛速度。

执行步骤如下。

算法 3 改进后的 K-Means 算法——IKBP(Improved K-Means for Bit-stream Protocol)

输入:n 条数据帧,类簇个数 K,K 个初始中心

输出:K 个类簇

步骤:

1. For i=1 to n;
2. 计算数据帧 x_i 到每个聚类中心的距离 d_i,并将数据帧 x_i 划分到距离最近的类簇中;
3. 按式(1)计算误差平方和 E;
4. 重新计算聚类中心,再按式(1)计算误差平方和 E^*;
5. 比较 E 与 E^* 的差的绝对值,若其小于阈值则转到步骤 6,否则转到步骤 1;
6. 输出 K 个类簇。

2.4 基于信息熵的聚类结果评价

在使用改进的 K-means 算法对未知协议进行聚类后,由于未知协议没有先验知识,难以对其进行评估。本文提出了一种使用信息熵的结果评价方案,算法的计算步骤如下。

算法 4 类簇评估算法——CEAE(Cluster Evaluation Algorithm by Entropy)

输入:含 n 条数据帧的类簇

输出:该类簇各列的信息熵

步骤:

1. 按照数据预处理方法得到二维矩阵,并建立相应的二维数组 a[n][m];
2. 循环遍历数组
3. 按列进行统计,计算每一列中每个字节出现的次数;
4. 循环遍历每一列
5. 按照式(2)计算每一列的信息熵值;
6. 输出每一列的信息熵。

将得到的结果以列号为 X 轴、该列的熵值为 Y 轴做图,分析聚类结果的好坏。熵值的大小代表了信息混杂程度的大小,在数据帧量很大的情况下,如果是同一种协议的数据帧,那么总有某些列的熵值接近 0;如果是多种协议混合的,熵值接近 0 的列几乎没有。因此可以用计算熵值的方法来评估未知协议聚类的好坏,可以设定一个评价阈值 $low\_entropy=0.05$,表示越多的列的熵值小于 $low\_entropy$,聚类效果就越好。

3 实验及结果分析

3.1 数据准备

为测试模型的效果,本文采用了林肯实验室发布的 tcpdump 数据集进行实验,测试数据分为以下 4 组。

第一组:5 种协议,分别是 dns,http,ntp,smtp,ssh;

第二组:5 种协议,分别是 ftp,icmp_error,irc,nbss,telnet;

第三组:9 种协议,分别是 arp,dns,http,llc,loop,ntp,rip,smtp,ssh;

第四组:9 种协议,分别是 ftp,icmp_data,icmp_error,irc,nbns,nbss,pop,syslog,telnet。

将以上所有协议每种取 100 条数据帧,组成 4 个测试数据集用于聚类实验,将聚类所得的类簇的部分结果用于测试基于信息熵的评价方案的有效性。由于使用的是已知协议类型的数据作未知的用,因此可使用正确率(记为 C)进行评价,计算公式为:

$$C=\frac{\text{正确分类数据帧条数}}{\text{数据帧总条数}}*100\% \tag{5}$$

3.2 实验结果及分析

3.2.1 K 值计算及聚类实验

将以上 4 组数据作为输入,计算每组数据的 K 值并选出初始中心,然后使用 K-Means 算法进行聚类,计算聚类的正确率。

(1)第一组数据

设置参数 $lowestSimilar(L)$ 从 0.1 到 1.0 变化,每次增加 0.05,计算相应的 K 值,结果如表 1 所列。

表 1 参数 L 与 K 值变化表

L	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5	0.55
K	1	1	1	1	1	1	1	2	3	3
L	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95	1.0	
K	6	7	10	15	17	25	27	29	45	

以 L 为横坐标、K 值为纵坐标做图,如图 3 所示。

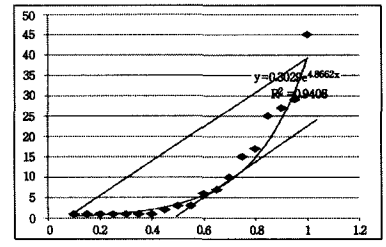

图 3 L-K 曲线图(第一组)

由图 3 可看出,L-K 服从指数分布,该曲线为:

$y=0.3029e^{4.8662x}$ 相关系数 $R^2=0.9408$

可看出,曲线在[0,0.5]区间变化缓慢,而在[0.8,1]区间变化迅速。将曲线的起点和终点相连,得直线 L_0,作 L_0 的平行线并与曲线相切,可得切点坐标为(0.65,7),进而可得 $K=7$ 和相应的初始中心 W。

指定 $K=7$ 及 W 中的初始中心,使用 K-Means 进行聚类,共有 500 条数据,其中 429 条正确聚类,正确率为:

$$C=\frac{429}{500}*100\%=85.5\%$$

(2)其他组实验

使用同样的方法对第二到第四组数据进行实验,所得 L-K 曲线如图 4—图 6 所示。

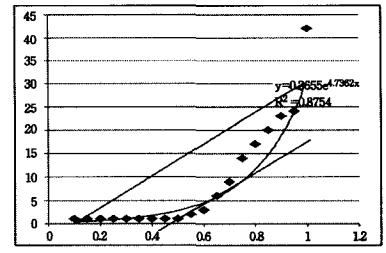

图 4 L-K 曲线图(第二组)

如图 4 所示,对于第二组数据,使用同样的方法可得切点坐标为(0.65,6),计算所得的 K 值为 6,使用 K-Means 聚类得正确率为:

$$C=\frac{464}{500}*100\%=92.8\%$$

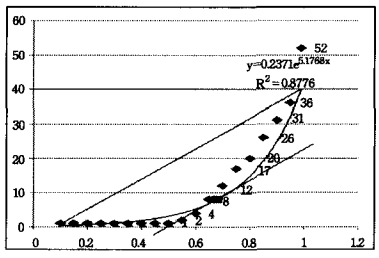

图 5　L-K 曲线图(第三组)

如图 5 所示,对于第三组数据,可得切点坐标(0.68,8),计算得 $K=8$,使用 K-Means 算法进行聚类,正确率为:

$$C=\frac{652}{900}*100\%=72.4\%$$

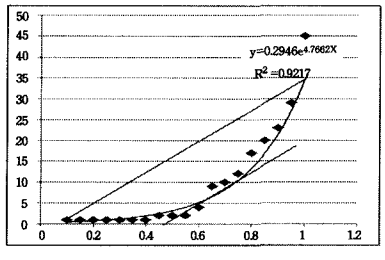

图 6　L-K 曲线图(第四组)

如图 6 所示,对于第四组数据,可得切点坐标(0.678,10),计算得 $K=10$,使用 K-Means 算法进行聚类,正确率为:

$$C=\frac{760}{900}*100\%=84.4\%$$

表 2 对比了 4 组数据实验结果。

表 2　4 组数据的实验结果

比较项	第一组	第二组	第三组	第四组
真实 K 值	5	5	9	9
计算得 K 值	7	6	8	10
聚类正确率(%)	85.5	92.8	72.4	84.4

从表 2 可以看出,用本文提出的 K 值计算方法计算出的 K 值与真实值较接近,且使用计算出来的 K 值进行聚类分析也能得到较高的聚类正确率。

3.2.2　信息熵评估实验

从第一组数据的聚类所得的类簇中取类簇 dns 及 ntp 和 ssh 混合簇进行信息熵计算实验,部分结果如表 3、表 4 所列。

表 3　dns 类簇各列熵值

列号	字符种类数	信息熵值	最高频字符	出现频率
1	26	8.181863	33	3(0.03)
2	83	1.0343713	ac	76(0.76)
3	2	1.0343713	10	76(0.76)
4	2	1.7158424	70	62(0.62)
5	3	1.9301767	14	62(0.62)
…	…	…	…	…

表 4　ntp 和 ssh 混合类簇各列熵值

列号	字符种类数	信息熵值	最高频字符	出现频率
1	85	8.196763	11	5(0.05)
2	3	1.3073595	ac	73(0.73)
3	3	1.3073595	10	73(0.73)
4	5	1.9350747	70	64(0.64)
5	5	2.5482666	32	40(0.4)
…	…	…	…	…

两组数据的信息熵曲线如图 7 所示。从图 7 中的数据可以看出,单类型的数据帧中存在若干列的信息熵值趋近于 0(小于阈值 *low_entropy*),信息熵值不趋近于 0 的列的值仍比较小;而混合类型的数据帧各列的信息熵值普遍较大,不存在信息熵接近 0 的列(均大于设定阈值 *low_entropy*)。因此,可以用基于信息熵的方法来评价聚类后类簇的"纯净度",评价方法为:对于一个聚类得到的类簇,存在信息熵小于 *low_entropy*(设为 0 或接近 0 的值)的列越多,类簇越纯净,说明得到的类簇是某一种协议的数据帧集合。

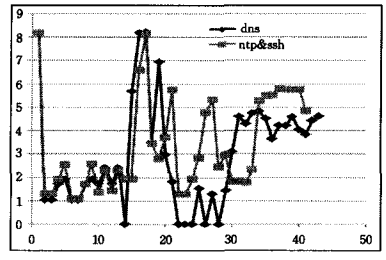

图 7　dns 类簇和 ntp、ssh 混合类簇每列信息熵的对比

3.3　其他聚类算法分析

本文提出的 K 值和初始中心计算方法除了适用于大多数基于划分的聚类算法(如 K-Means),对其他聚类算法也适合。如:1)凝聚型层次聚类算法(AGNES),其思想是将每个对象作为单独的一个类,然后根据相似度相继合并相似的类,直到达到终止条件,例如期望的聚类个数。虽然不需要指定初始聚类中心,但比较恰当的聚类个数 K 的指定能使算法得到较高的正确率。2)最大期望聚类算法(EM),其根据在概率模型中寻找参数最大似然估计或最大后验估计来进行聚类,虽然指定聚类个数 K 和初始中心都不是必须的,但较准确的 K 值能使该算法的性能得到大大提升。3)序列信息最大化算法(sIB),该算法是一种效果较好的文档聚类算法。在文本应用中,将每一条数据帧视为一篇短文档,在指定恰当的 K 值后,该算法也能得到较好的效果。

另外,某些聚类思想则不适合用于本模型。如基于密度的聚类思想:指定一个区域范围 *radius* 及在该范围内期望的最小对象个数 *minCount*,对于给定类中的每个数据点,在 *radius* 范围内必须至少包含 *minCount* 个点。如 *DBSCAN* 算法,需要指定的参数 *radius* 和 *minCount*,与本文要解决的 K 值和初始中心计算问题不太相符。同样地,基于网格的聚类思想为:先将数据空间划分为网络单元,将所有数据对象映射到网络单元中,然后计算每个单元的密度,根据用户输入的密度阈值 *minPoints* 判断每个网络单元是否为高密度单元,最后由邻近的稠密度单元组形成需要的类簇。对于该类方法,影响结果的关键在于密度阈值 *minPoints* 的设定,其同样不适合用于本文的模型。

最后,本文从基于划分、基于层次、基于概率统计、基于序列信息这几大类中各选用了一种适用于本模型的聚类算法进行对比实验,分别是:K-Means,AGNES,EM,sIB 算法。实验数据使用的是 3.1 节提到的 4 组数据,这 4 种算法都采用了本文提出的 K 值计算方法,指定了聚类个数 K 值。各算法对于每组数据的实验结果如表 5—表 8 所列。

表 5　适用于本模型的部分算法对比实验(第一组)

算法	指定 K	指定初始中心	正确率(%)	建模时间(s)
K-Means	7	是	85.5	0.50
AGNES	7	否	77.5	0.46
EM	7	否	83.4	4.91
sIB	7	否	83.7	1.68

表 6　适用于本模型的部分算法对比实验(第二组)

算法	指定 K	指定初始中心	正确率(%)	建模时间(s)
K-Means	6	是	92.8	0.27
AGNES	6	否	90.9	0.44
EM	6	否	92.1	3.24
sIB	6	否	93.5	1.55

表 7　适用于本模型的部分算法对比实验(第三组)

算法	指定 K	指定初始中心	正确率(%)	建模时间(s)
K-Means	8	是	72.4	0.89
AGNES	8	否	51.2	1.13
EM	8	否	64.0	10.92
sIB	8	否	83.3	3.98

表 8　适用于本模型的部分算法对比实验(第四组)

算法	指定 K	指定初始中心	正确率(%)	建模时间(s)
K-Means	10	是	84.4	1.03
AGNES	10	否	55.0	1.48
EM	10	否	79.5	7.96
sIB	10	否	91.0	3.14

从表 5－表 8 的结果中可以看出：EM 算法最费时；sIB 算法虽然正确率高，但也比较费时；AGNES 算法在数据量较大时表现较差；K-Means 算法正确率较高，速度最快，在指定初始中心后计算结果比较稳定，是本模型优先考虑的算法。

结束语　本文提出的零知识条件下比特流未知协议分类模型，使用 AKBP 算法进行 K 值计算和初始中心选择，利用聚类分析和基于信息熵的结果评价方法，能有效地将在实际应用中捕获的多协议数据帧分类为单类型数据帧；解决了聚类过程中 K 值确定、初始中心选择以及在实际应用中对聚类结果评价困难等问题，为以后进行进一步的研究提供了较大参考。

参 考 文 献

[1] Luo Cheng, Zhang Yu-qing, Wang Long, et al. Automatic network protocol analysis and vulnerability discovery based on symbolic expression[J]. Journal of Graduate University of Chinese Academy of Science, 2013, 30(2): 278-284(in Chinese)
罗成，张玉清，王龙，等. 基于符号表达式的未知协议格式分析及漏洞挖掘[J]. 中国科学院研究生院学报，2013，30(2)：278-284

[2] Song Jiang. Unknown protocol identification in wireless environment[D]. Chengdu: University of Electronic Science and Technology of China, 2013(in Chinese)
宋疆. 无线网络环境下未知协议发现探索研究[D]. 成都：电子科技大学，2013

[3] Jin Ling. Study on Bit Stream Oriented Unknown Frame Head Identification[D]. Shanghai: Shanghai Jiaotong University, 2011(in Chinese)
金凌. 面向比特流的未知帧头识别技术研究[D]. 上海：上海交通大学，2011

[4] Wang Yong, Wu Yan-mei, Li Fen, et al. Protocol identification association analysis in mobile network environment[J]. Application Research of Computers, 2015, 32(1): 243-248(in Chinese)
王勇，吴艳梅，李芬，等. 面向比特流数据的未知协议关联分析与识别[J/OL]. 计算机应用研究，2015，32(1)：243-248

[5] 谢希仁. 计算机网络(第五版)[M]. 北京：电子工业出版社，2008：23-30

[6] Wang Yang-de. Study on Bit Stream Oriented Protocol Fram Head Identificaiton[D]. Shanghai: Shanghai Jiaotong University, 2013(in Chinese)
王杨德. 面向比特流的协议帧头结构分析研究[D]. 上海：上海交通大学，2013

[7] Meng Fan-zhi, Liu Yuan, Zhang Chun-rui, et al. Inferring protocol state machine for binary communication protocol[C]// 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA). Ottawa, ON: IEEE, 2014: 870-874

[8] He Yong-jun, Shu Hui, Xiong Xiao-bing. Protocol Reverse Engineering Based on DynamoRIO[C]// International Conference on Information and Multimedia Technology, 2009(ICIMT 09). Jeju Island: IEEE, 2009: 310-314

[9] Wang Yi-peng, Yun Xiao-chun, Shafiq M Z, et al. A semantics aware approach to automated reverse engineering unknown protocols[C]// 2012 20th IEEE International Conference on Network Protocols (ICNP). Austin, TX: IEEE, 2012: 1-10

[10] Cui W, Vern P, Weaver N, et al. Protocol-independent adaptive replay of application dialog[C]// The 13th Annual Network and Distributed System Security Symposium (NDSS). San Diego, 2006: 126-141

[11] Newsome J, Brumley D, Frankinlin J, et al. Replayer: automatic protocol replay by binary analysis[C]// Proc of ACM Conference on Computer and Communications Security. NewYork, 2006: 311-321

[12] Juan C, Heng Yin, Liang Zhen-kai, et al. Polyglot: Automatic extraction of protocol message formatusing dynamic binary analysis[C]// Proceedings of the 14th ACM Conference on Computer and Communications Security. Washington, DC, 2007: 317-329

[13] Wang Qian, Wang Cheng, Feng Zhen-yuan, et al. Summary of K-means clustering algorithm[J]. Electronic Design Engineering, 2012, 20(7): 21-24(in Chinese)
王千，王成，冯振远，等. K-means 聚类算法研究综述[J]. 电子设计工程，2012，20(7)：21-24

[14] Yang Shan-lin, Li Yong-sen, Hu Xiao-xuan, et al. Optimization Study on k Value of K-means Algorithm[J]. Systems Engineering-Theary & Practice, 2006, 26(2): 97-101(in Chinese)
杨善林，李永森，胡笑旋，等. K-MEANS 算法中的 K 值优化问题研究[J]. 系统工程理论与实践，2006，26(2)：97-101

[15] Huang Xiao-yan, Chen Xing-yuan, Zhu Ning, et al. Binary protocol identification based on weighted byte entropy vector[J]. Application Research of Computers, 2015, 32(2): 493-497(in Chinese)
黄笑言，陈性元，祝宁，等. 基于字节熵矢量加权指纹的二进制协议识别[J]. 计算机应用研究，2015，32(2)：493-497

[16] Liu Hua-wen. A Study on Feature Selection Algorithms using Information Entropy[D]. Changchun: Jilin University, 2010(in Chinese)
刘华文. 基于信息熵的特征选择算法研究[D]. 长春：吉林大学，2010