第四届全国网络与信息安全防护峰会

基于类别向导信息增益与恶意程序检测

谭营 教授

北京大学

提纲

1. 引言
2. 恶意程序检测研究现状
3. 类别向导信息增益(CIG)
4. 基于CIG的恶意程序检测方法
5. 实验与讨论
6. 结语

1. 引言

- 恶意程序(Malware or Malicious Executables)是
- *任何破坏计算机系统或未经用户授权访问计算机系统的程序都可以称为恶意程序。*
- *主要特性：传染性、破坏性、隐藏性、潜伏性、触发性，等。*

- 常见种类：
  - ✓ 病毒、后门、构造器、木马、蠕虫等。

1. 引言

恶意程序增长趋势

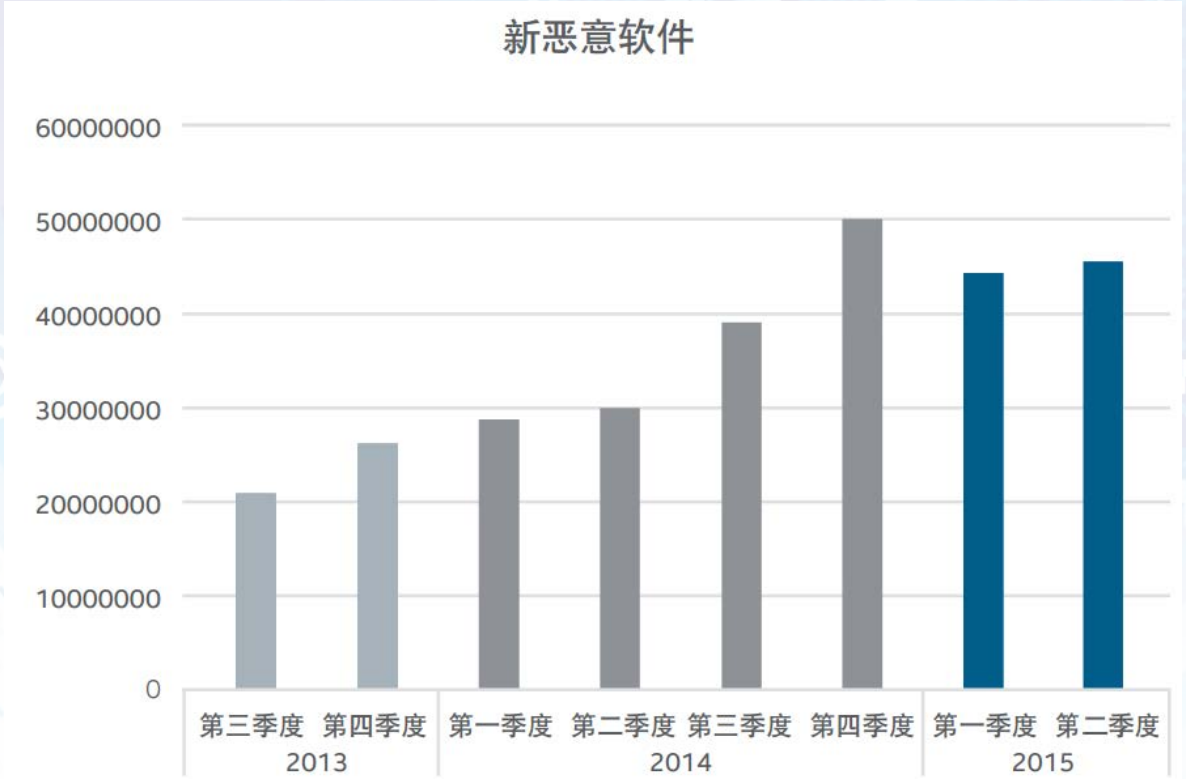


图1. 近2年新增恶意程序数量

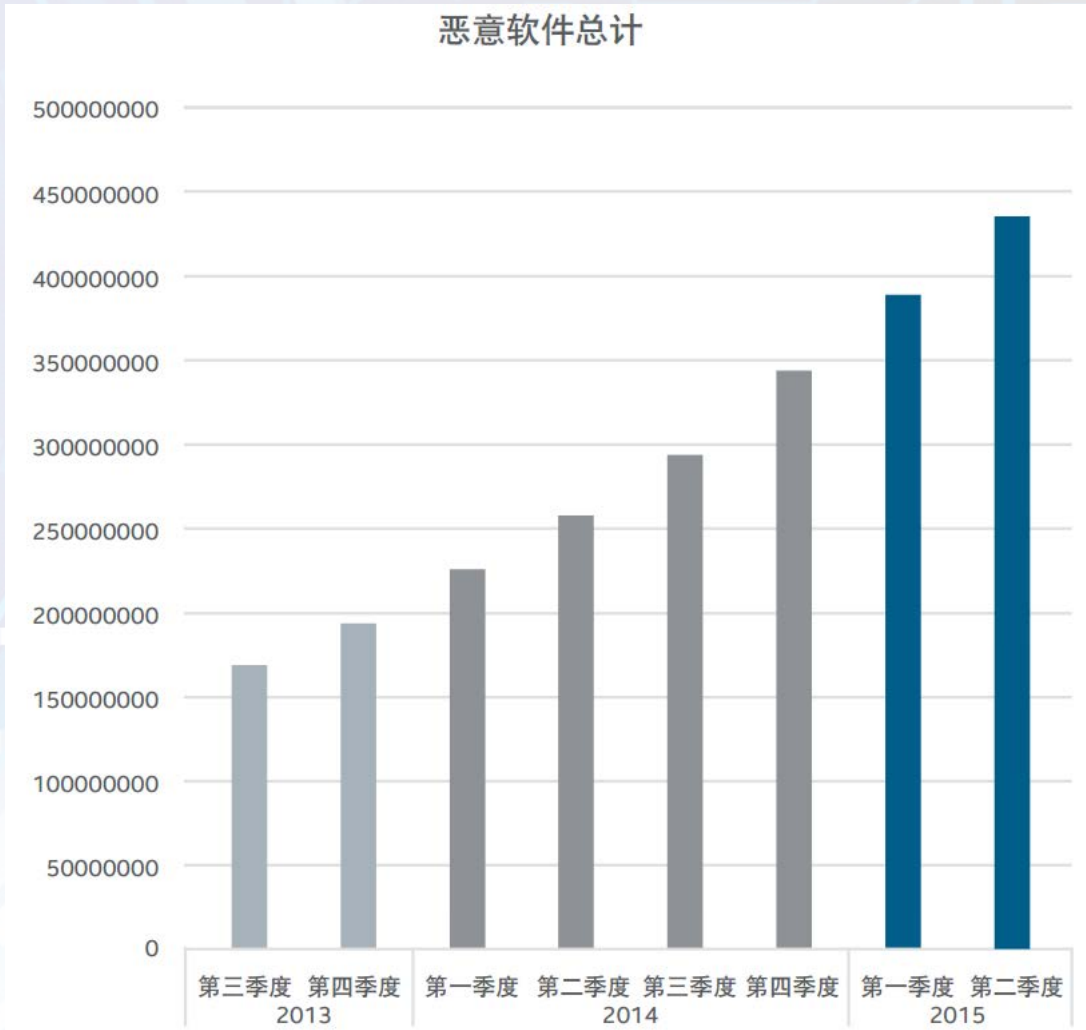


图2. 近2年恶意程序总量

来源：McAfee Labs威胁报告（2015年8月） http://www.mcafee.com/cn/resources/reports/rp-quarterly-threats-aug-2015.pdf?view=legacy

1. 引言

面临的严峻挑战

- 现在恶意程序越来越复杂，其制作者采用越来越高级的技术和技巧来逃避基于特征的检测。
- 恶意程序传播到目标后，自身代码和结构在空间与时间上都可能具有不同的变化。
- 由于计算机软件的脆弱性与互联网的开放性，我们将与恶意程序*长久共存*。

提纲

1. 引言
2. **恶意程序检测研究现状**
   - ✓ **2.1 常见恶意程序检测方法**
   - ✓ **2.2 基于机器学习的恶意程序检测方法**
   - ✓ **2.3 基于免疫原理的恶意程序检测方法**
     - **2.3.1 基于带有惩罚因子的阴性选择算法**
     - **2.3.2 基于危险特征的阴性选择算法**
     - **2.3.3 基于免疫浓度的特征提取方法**
     - **2.3.4 基于免疫协同机制的学习框架**
3. 类别向导信息增益(CIG)
4. 基于CIG的恶意程序检测方法
5. 实验
6. 总结

2.1 常见恶意程序检测方法

- 比较法
  - ✓ 用原始的正常备份与被检测内容进行比较（内容比较）。
- 校验和法
  - ✓ 对比原始文件和现有文件的校验和。
- 特征码法
  - ✓ 现在最主流的反恶意程序技术：大部分杀毒软件都是基于特征码的。
  - ✓ 特征码是一段在恶意程序中截取出来的，独一无二的二进制程序码，足以标识出一个特定的恶意程序。**特征码**与**恶意程序是一一对应的关系**。
- 行为监控法
  - ✓ 监控程序行为，根据恶意程序特定的行为特性进行检测。

2.1 常见恶意程序检测方法

- 智能检测法
- 通过分析恶意程序代码，获得统计的启发式知识，利用这些启发式知识进行恶意程序检测。例如：
  - ✓ 基于机器学习的启发式检测方法
  - ✓ 基于免疫原理的启发式检测方法
  - ✓ ……

2.2 基于机器学习的恶意程序检测方法

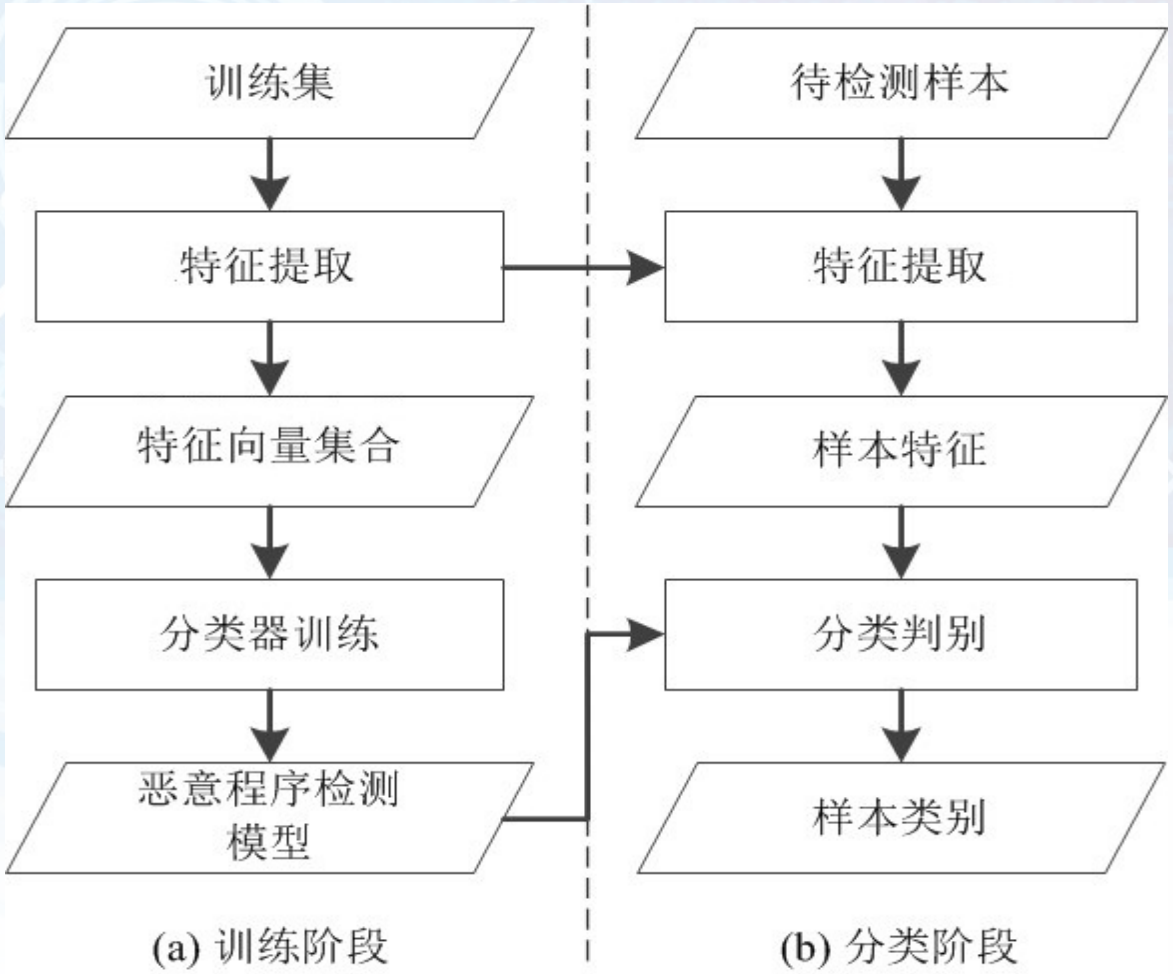


图3. 基于机器学习的恶意程序检测方法的框架

2.2 基于机器学习的恶意程序检测方法

研究简史

时间	工作
2004年	Kloter等人第一次将机器学习和数据挖掘的方法应用于恶意程序检测； 此后，很多研究者将机器学习与数据挖掘的方法应用于计算机病毒检测；
2006年	Henchiri等人采用数据挖掘的方法，提取出代表恶意程序的频繁模式进行计算机病毒检测； Kloter对其2004年提出的方法进行了扩展；
2007年	Reddy对Kloter的方法进行了改进，得到了更优的结果；
2009年	Tabish S M等提出了一个在字节层次上对文件内容进行统计分析的恶意程序检测模型，此模型是第一个完全不基于特征码的计算机病毒检测； M. Zubair Shafiq等人提出将程序的PE头中各个字段作为特征；
2013年	Dahl G.E.等人将随机投影和神经网络应用在大规模恶意程序检测上。
2014年	Zhang和Tan提出了类别向导信息增益，并成功用于恶意程序检测。

2.3 基于免疫原理的恶意程序检测方法

- 基于生物免疫原理，开展计算机恶意程序检测方法研究是非常合理和有价值的。因为：
  - ✓ 首先，生物免疫系统和计算机安全系统具有高度相似的功能
  - ✓ 其次，生物免疫系统的诸多优点都是计算机安全系统所需要的特性；
    - 这些优点包括噪声忍耐、无中心控制和强化记忆等。
  - ✓ 最后，前人研究表明基于免疫原理进行恶意程序检测是可行的、有效的。

2.3 基于免疫原理的恶意程序检测方法

研究简史

时间	工作
1990	Ishida等人提出了最早的**人工免疫网络算法**，Hunt和Cooke在1996年改进了这种算法，主要用于分布式诊断和优化计算；
1994	Forrest等提出了**阴性选择算法**，首先将生物免疫原理应用于计算机异常检测中，开启了免疫原理应用于计算机安全领域的先河；
2000	受Burnet克隆选择学说启发，Castro和Zuben提出了**克隆选择算法**；
2002	Matzinger提出了**危险理论**，对阴性选择算法进行了补充和完善，同年Aickelin等将其引入到计算机安全领域中；
2005	Greensmith提出了**树突状细胞算法**进行异常检测；
2009	Tan提出了**免疫浓度**，并成功用于病毒检测
至今	许多研究者提出了多种基于免疫原理的计算机恶意程序检测方法。

2. 恶意程序检测研究现状

- 2.1 常见恶意程序检测方法
- 2.2 基于机器学习的恶意程序检测方法
- 2.3 基于免疫原理的恶意程序检测方法
  - ✓ **2.3.1 基于带有惩罚因子的阴性选择算法**
  - ✓ **2.3.2 基于危险特征的阴性选择算法**
  - ✓ **2.3.3 基于免疫浓度的特征提取方法**
  - ✓ **2.3.4 基于免疫协同机制的学习框架**

2.3.1 基于带有惩罚因子的阴性选择算法

- 自体-异体有害性定义缺陷
  - ✓ 传统的阴性选择算法认为“自体”都是无害的，而“异体”都是有害的。
- 带有惩罚因子的阴性选择算法（NSAPF）
  - ✓ 摆脱了传统阴性选择算法中对“自体”和“异体”**有害性定义的缺陷**，关注**程序代码本身的危险性；**
  - ✓ 通过调整惩罚因子，充分挖掘和调节了**特征的表征性**，提高了相应模型的性能。

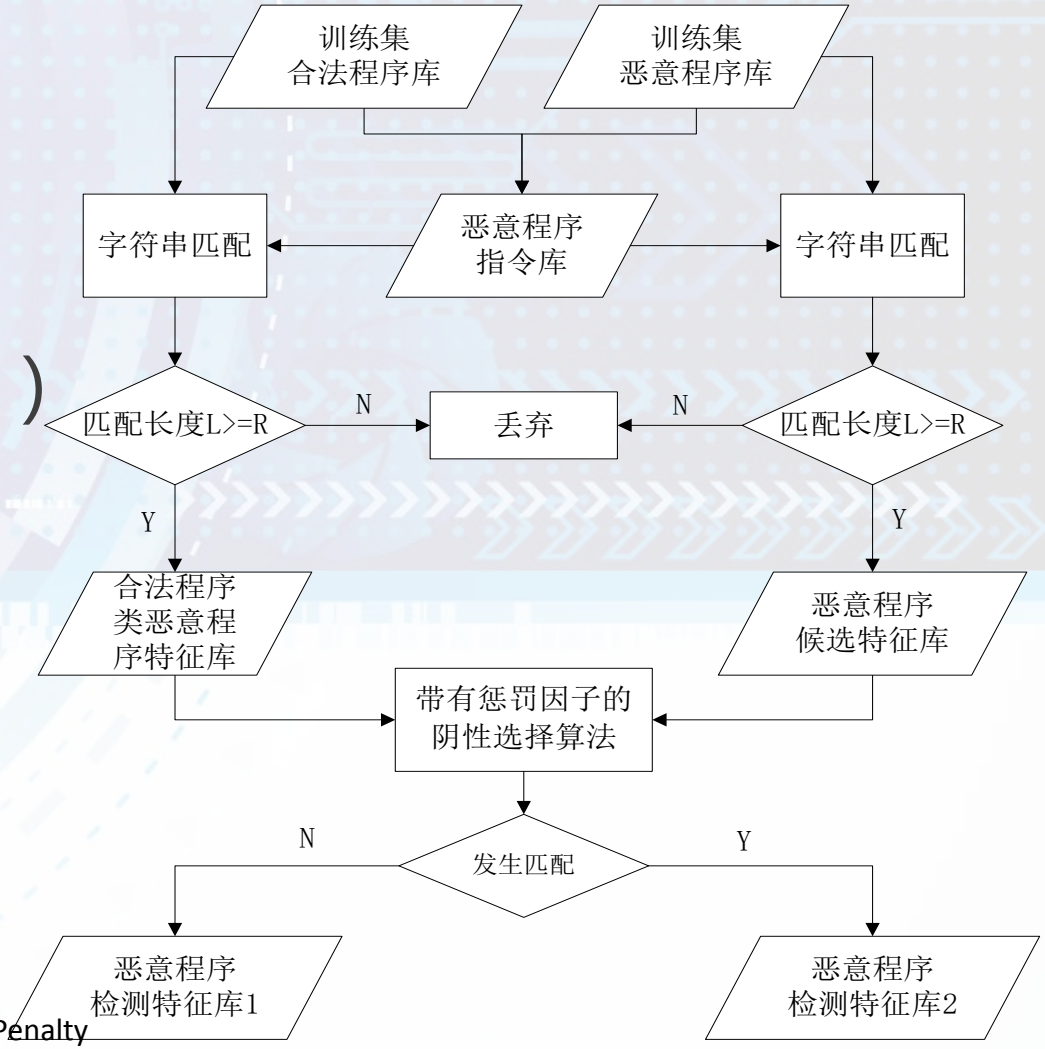
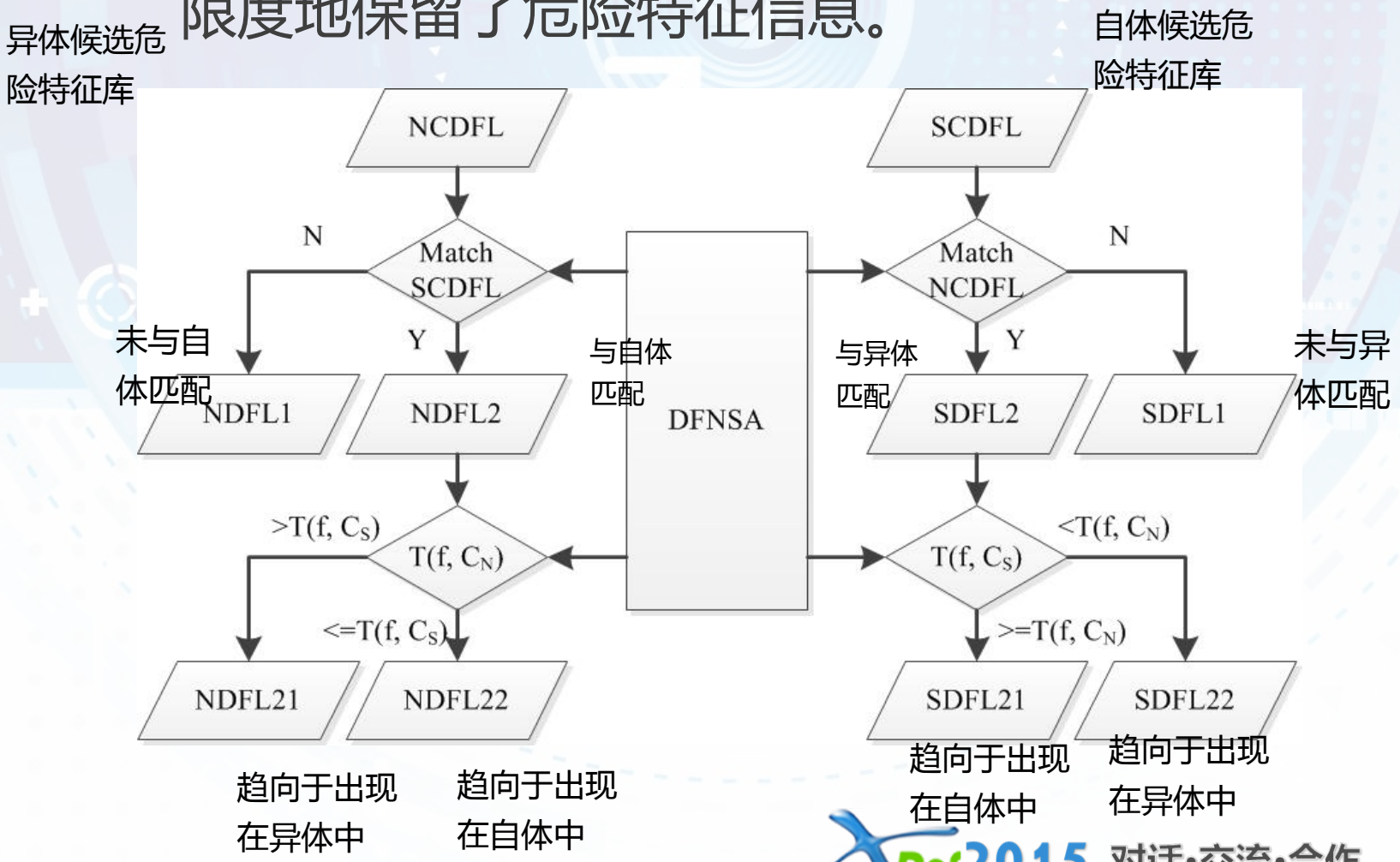


图4. 特征提取流程图

P.T. Zhang, W. Wang, Y. Tan, "A Malware Detection Model based on a Negative Selection Algorithm with Penalty Factor," ***Science in China Series F - Information Science***, Vol. 53, no. 12, pp. 2461-2471, 2010.

2.3.2 基于危险特征的阴性选择算法

基于危险特征的阴性选择算法（DFNSA）详细划分了**危险特征空间**，最大限度地保留了危险特征信息。



特征向量定义：

$$< \frac{M_{NDFL1}}{L_{NDFL1}}, \frac{M_{NDFL21}}{L_{NDFL21}}, \frac{M_{SDFL1}+M_{SDFL21}}{L_{SDFL1}+L_{SDFL21}} >$$

其中M_i表示样本与危险特征库i的匹配值，L_i表示特征库i所有特征的权值和，

$i \in \{NDFL1, NDFL21, SDFL1, SDFL21\}$

图5. 基于危险特征的阴性选择算法流程图

2.3.3 基于免疫浓度的特征提取方法

启发

- 生物免疫系统中存在一种免疫浓度机制，即只有抗原浓度超过某个特定阈值后，免疫系统才会对这些抗原进行免疫应答。

目的

- 利用免疫浓度的思想来构造恶意程序特征，解决现有方法中特征向量维度过高的问题，从而简化分类器的设计和实现。
- 相对现有工作中特征提取只使用恶意程序趋向信息一个维度，此项工作中增加了另一维正常文件的趋向信息。
- 运用局部浓度方法考虑特征的位置相关信息，提升模型效率。

2.3.3 基于免疫浓度的特征提取方法

- 借鉴免疫浓度机制，可以构造三种浓度特征：
  - ✓ 全局浓度特征向量(GC)：在整个样本上分别计算两个类别的特征浓度值。
  - ✓ 局部浓度特征向量(LC)：将样本划分为*N*个局部区域，在这些区域上分别计算浓度特征向量，连接构成LC
  - ✓ 混合浓度特征向量(HC)：$HC =< GC, LC >$

2.3.3 基于免疫浓度的特征提取方法

- 全局浓度
  - 全局浓度的特征是一个以（异体浓度、自体浓度）为形式的二维向量

$$VC_i = \frac{VN_i}{N_i}, BC_i = \frac{BN_i}{N_i}$$

  - 其中，BC_i和VC_i表示自体、异体浓度； BN_i指同时出现在文件i和自体基因库中的基因个数，VN_i 指同时出现在文件i和异体基因库中的基因个数，N_i指文件中不同的基因个数，选取方式也是半长窗口错位的窗口滑动覆盖法。

Y. Tan, C. Deng, G.C. Ruan, “Concentration Based Feature Construction Approach for Spam Detection,” *Proceedings of International Joint Conference on Neural Networks* , Atlanta, Georgia, USA, June 14-19, 2009, pp. 1344-1350.

2.3.3 基于免疫浓度的特征提取方法

局部浓度

- 使用不带错位覆盖的滑窗划分出固定个数N、固定长度W-bit的字 符串片断，在此基础上以全局浓度的方式提取这个片断内部的浓度 信息，形成局部浓度向量(VC_i, BC_i)，$i=0, 1,\ldots,N$，再将这些向量有序连接$<(VC_1, BC_1), (VC_2, BC_2),\ldots,(VC_N, BC_N)>$

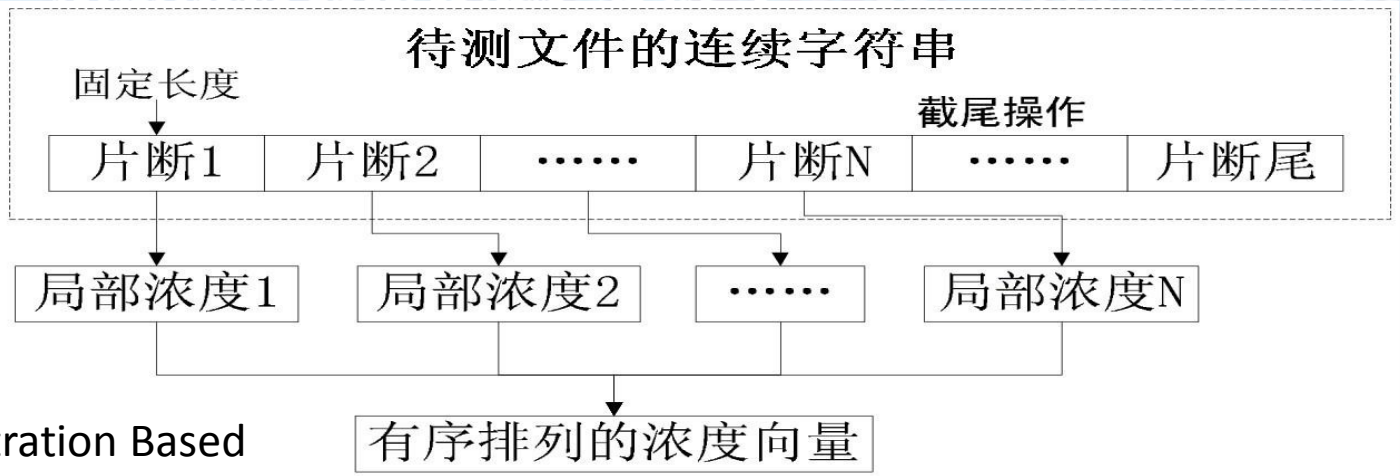


图10. 局部浓度的构造过程

Y.C. Zhu, Y. Tan, " A Local Concentration Based Feature Extraction Approach for Spam Filtering," ***IEEE Transactions on Information Forensics and Security***, Vol. 6, No. 2, June 2011, pp. 486-497.

2.3.4 基于免疫协同机制的学习框架

- 特征提取
  - ✓ 抗原**特异性特征向量**和**非特异性特征向量**分别模拟生物免疫系统中的抗原表位和危险特征；
    - 选出N_1个仅出现在恶意程序中的特征构成L_1；
    - 选出N_2个同时出现在合法程序和恶意程序中的特征构成L_2；
- 实值免疫信号
  - ✓ 分类器C_1和C_2将其对输入向量的评分作为实值信号输出给**协同分类器**C_3，分别模拟免疫应答中的第一信号和第二信号；
- 协同判别
  - ✓ **协同分类器**C_3在两种信号的协同作用下分类判别待检测样本。

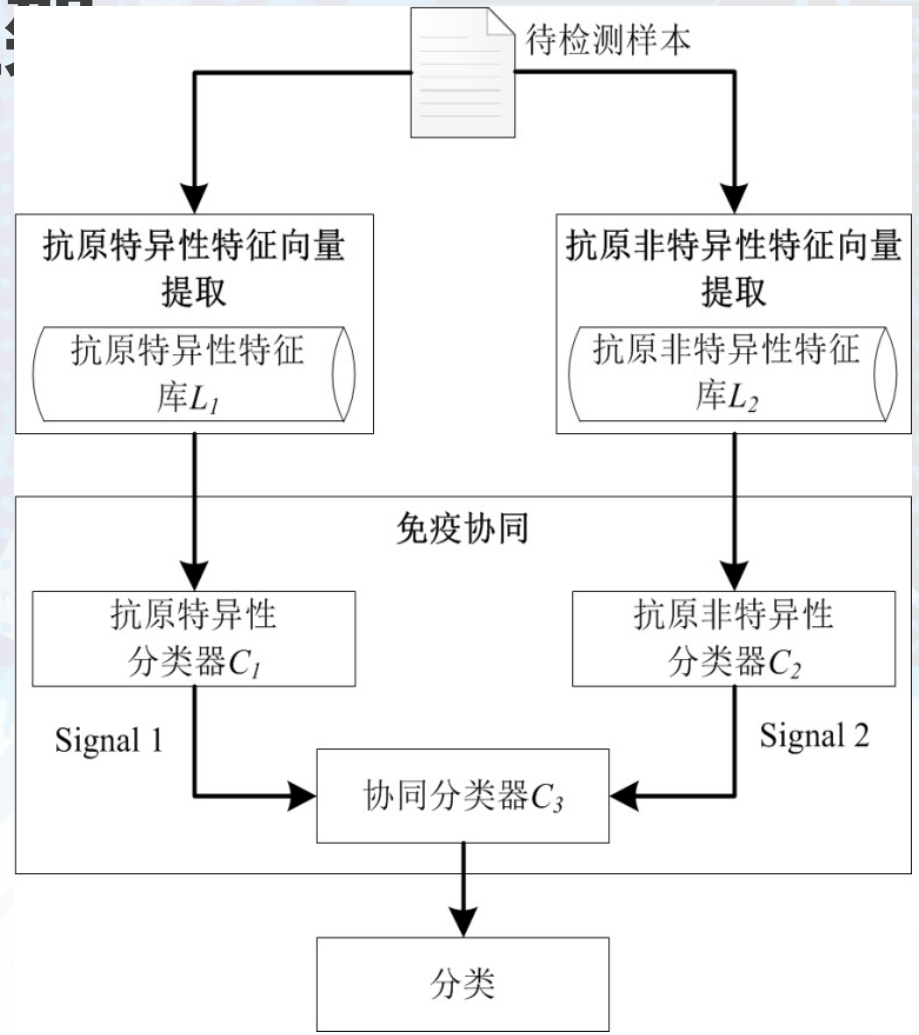


图6. 基于免疫协同机制的学习框架

More Details!

Ying Tan

Artificial Immune System: Applications in Computer Security , ***Wiley & IEEE Press***

January 26, 2016.

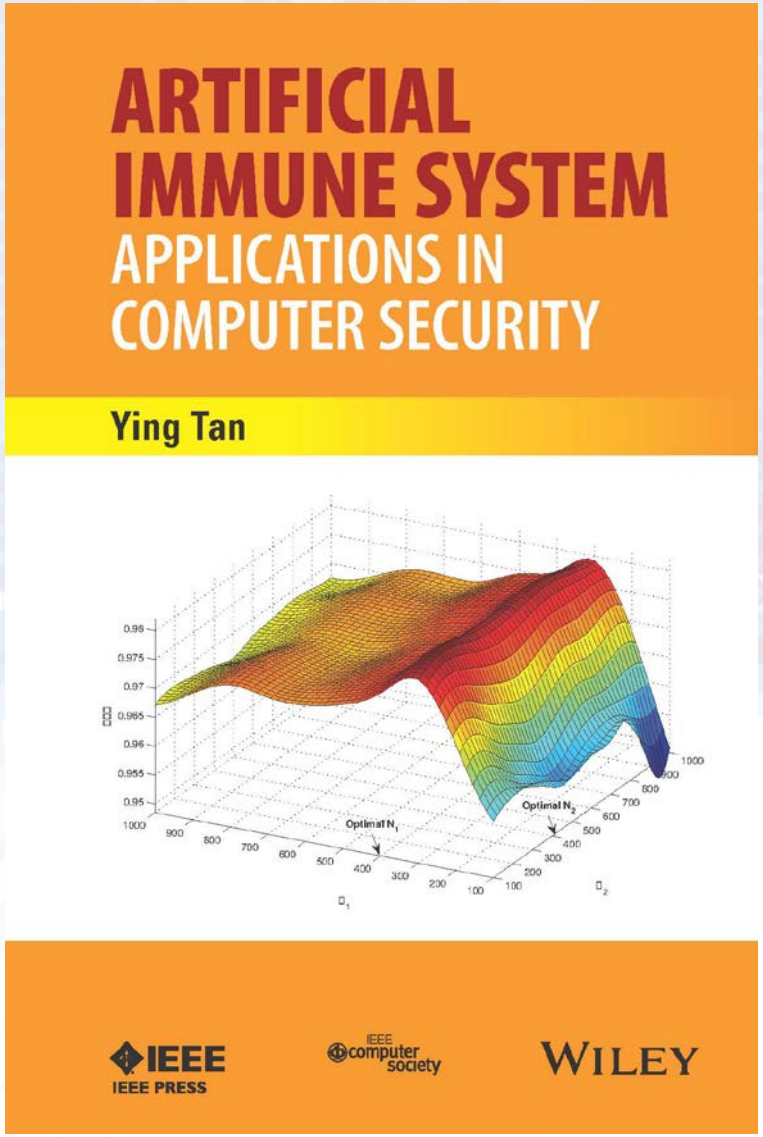
ISBN: 978-1119076285.

[TOC with samples]

[Book at Amazon.Com]

Email: ytan@pku.edu.cn

URL: http://www.cil.pku.edu.cn



提纲

1. 引言
2. 恶意程序检测研究现状
3. **类别向导信息增益(CIG)**
   - ✓ **3.1 CIG研究背景**
   - ✓ **3.2 CIG的定义**
   - ✓ **3.3 分析**
4. 基于CIG的恶意程序检测方法
5. 实验
6. 总结

3.1 CIG研究背景

- 在现实世界中，计算机恶意程序有两种存在形式：
  - ✓ **纯恶意程序**（Malware loader）：仅包含恶意程序的原始代码
  - ✓ **染毒程序**（Infected executable）：纯恶意程序和合法程序的结合体

- 染毒程序可以是被恶意程序感染的合法程序，也可以是由恶意程序作者将合法程序和纯恶意程序绑定后的产物。
- 染毒程序是恶意程序在现实世界中的主要存在形式。

P.T. Zhang and Y. Tan, “Class-wise Information Gain ” ***Proc of IEEE Third International Conference on Information Science and Technology (ICIST 2013)*** , Yangzhou, China, March 23-25, 2013. pp. 972-978.

3.1 CIG研究背景

- 从理论上讲，染毒程序可以包含所有的合法程序特征，因此合法程序特征不适合检测染毒程序，提取**恶意程序特征**成为恶意程序检测的关键。
- 很多工作将恶意程序检测问题看做合法程序与纯恶意程序的分类问题，经常将信息增益（IG）作为特征选择指标。由于IG没有考虑特征的类别信息，因此IG选择出的特征的类别是未知的。
  - ✓ 在检测纯恶意程序时，这些工作取得较好的结果；
  - ✓ 在检测染毒程序时，这些工作的性能会急剧下降。

3.1 CIG研究背景

- 信息增益(IG)是一种衡量某特征f对分类所带来的信息量的指标。

$$IG(f)=\sum_{v_f\in\{0,1\}}\sum_{C\in\{C_i\}}P(v_f,C)\log\frac{P(v_f,C)}{P(v_f)P(C)}$$

- 注：特征f带来的信息量越多，其信息增益越大。在分类问题中人们希望得到高信息增益的特征。

- **类别向导信息增益（CIG）**
  - ✓ CIG扩展了IG，比IG更加精细，能够度量" ***特征f为识别类别C带来的信息量*** " ；
  - ✓ CIG允许研究者自由调整检测特征库中不同类别特征的分布，具有建立更加稳定的模型的潜力。

3.2 CIG的定义

- CIG公式如下：

$$CIG(f, C_i) = P(v_f = 1, C_i) \log \frac{P(v_f = 1, C_i)}{P(v_f = 1)P(C_i)} + \sum_{C \in \{C_j\} \wedge i \neq j} P(v_f = 0, C_j) \log \frac{P(v_f = 0, C_j)}{P(v_f = 0)P(C_j)}$$

- ✓ CIG由两部分组成，其中第一部分表示特征f出现在类别C_i中的统计信息，第二部分表示特征*f*不出现在其他类别中的统计信息。
- ✓ 特征*f*在类别C_i和C_j中的取值隐式地表达了特征*f*的类别信息，建立了特征取值与类别信息的关联。

3.2 CIG的定义

在恶意程序检测场景下

$$CIG(f,C_B) = P(v_f=1,C_B)log\frac{P(v_f=1,C_B)}{P(v_f=1)P(C_B)} + P(v_f=0,C_M)log\frac{P(v_f=0,C_M)}{P(v_f=0)P(C_M)}$$

$$CIG(f,C_M) = P(v_f=0,C_B)log\frac{P(v_f=0,C_B)}{P(v_f=0)P(C_B)} + P(v_f=1,C_M)log\frac{P(v_f=1,C_M)}{P(v_f=1)P(C_M)}$$

B表示合法程序类别

M表示恶意程序类别

3.3 分析

$$P(v_f = 1) = 0.5,\ \ P(C_B) = 0.5$$

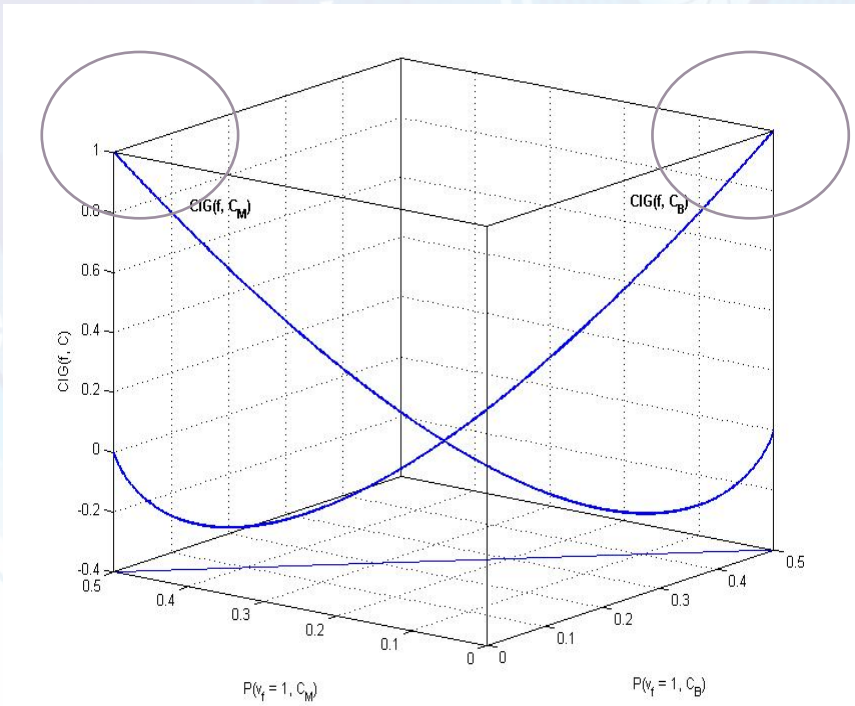


图13. 函数CIG(f, C)

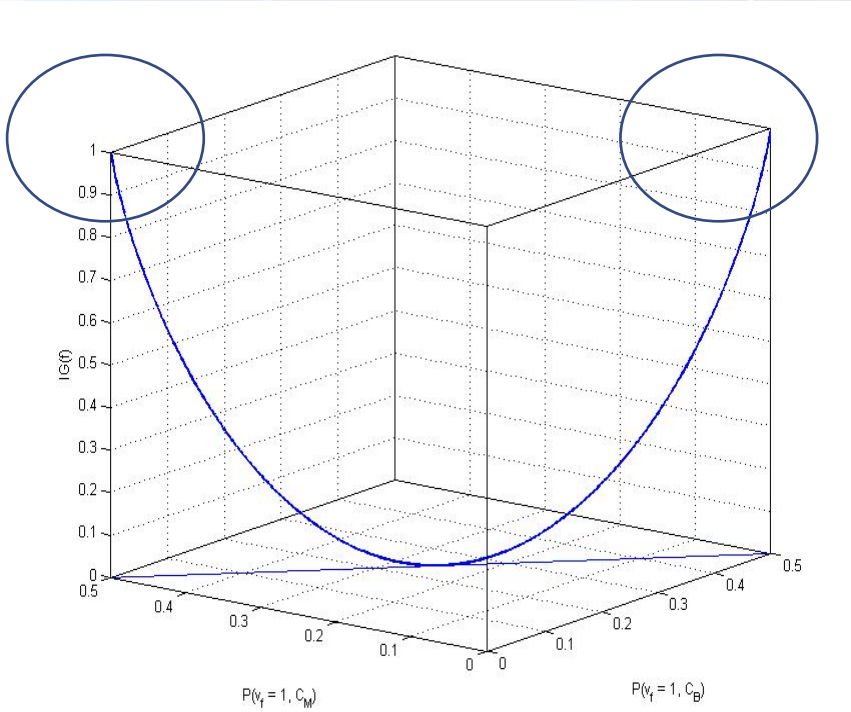


图14. 函数IG(f)

- 在两类分类问题中，CIG可以看做是IG在类别角度上的分解。

3.3 分析

- 具有较高$IG(f)$ 值的特征可能具有较高的$CIG(f,C_B)$与较低的$CIG(f,C_M)$，或者较高的$CIG(f,C_M)$与较低的$CIG(f,C_B)$。

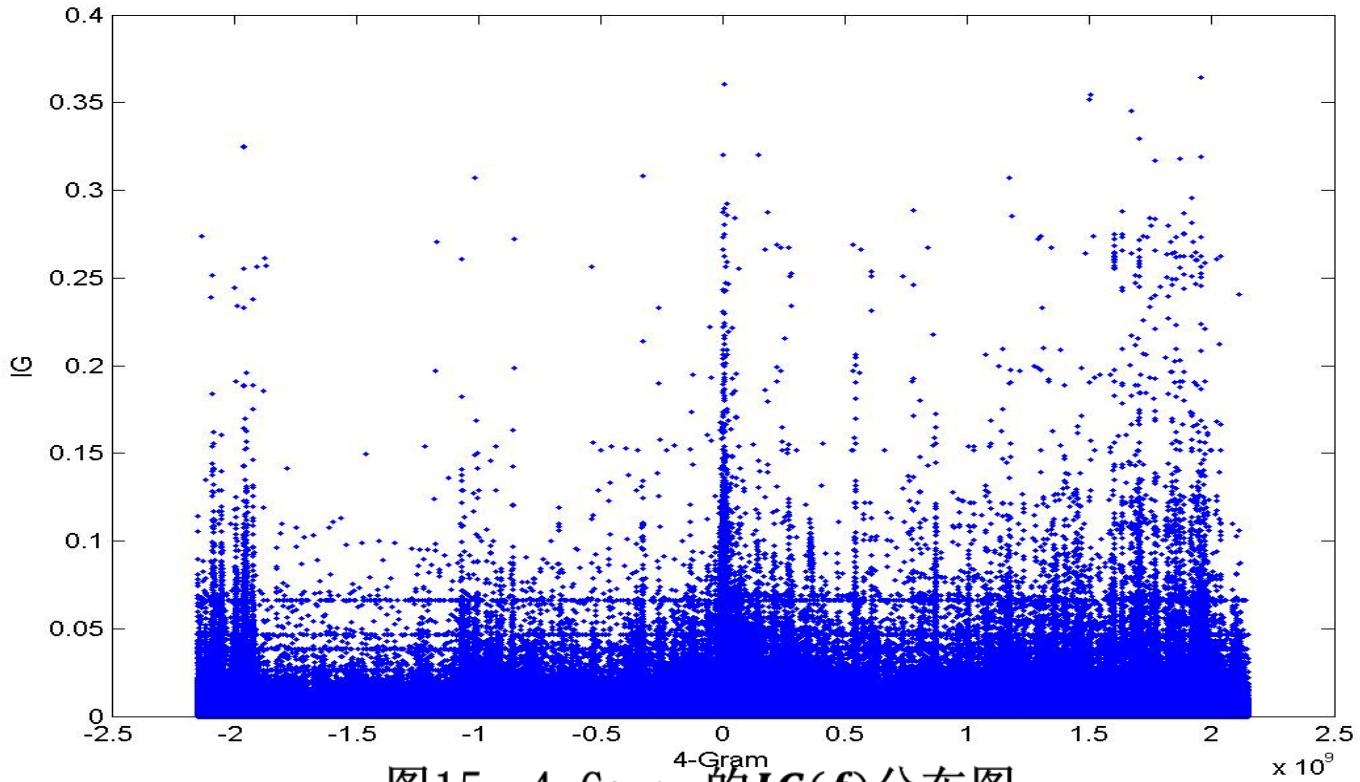


图15. 4-Grams的$IG(f)$分布图

3.3 分析

- 相对于恶意程序特征，合法程序特征往往具有较高的信息增益值。
  - ✓ 这是因为合法程序使用了更加一般的、经常使用的计算机指令，因此其中的特征分布更加集中，特征分布的不均衡性更加明显。

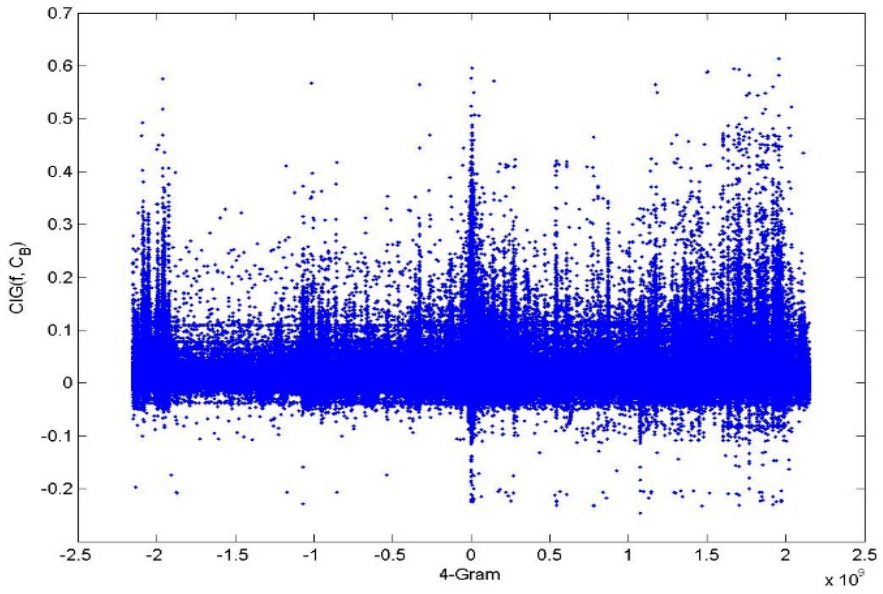


图16. 4-Gram的$CIG(f, C_B)$分布图

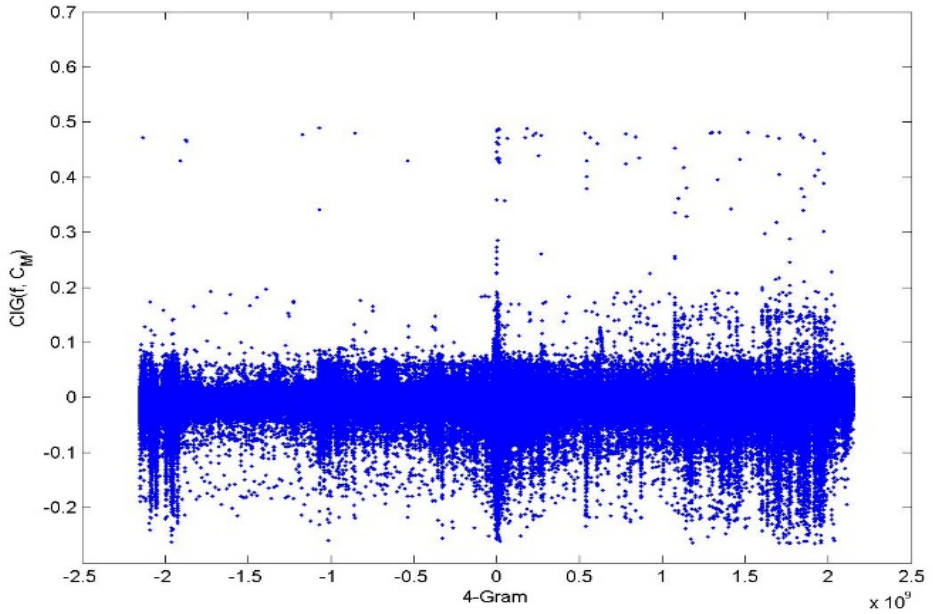


图17. 4-Gram的$CIG(f, C_M)$分布图

3.3 分析

- 与恶意程序特征相比，合法程序特征往往更加显著，能够为分类合法程序和纯恶意程序带来更多的信息量。
- 在检测染毒文件时合法程序特征是无效的，因此基于IG的恶意程序检测方法在检测染毒程序时性能将急剧下降。

提纲

1. 引言
2. 恶意程序检测研究现状
3. 类别向导信息增益(CIG)
4. **基于CIG的恶意程序检测方法**
   - ✓ **4.1 特征提取**
   - ✓ **4.2 分类**
5. 实验
6. 总结

4.1 特征提取

- 该方法将N元文法（N-Gram）作为候选特征。
- 由于IA-32(Intel Architecture–32) 指令集中指令的最小单位是1个字节，因此将一个文法（Gram）定义为长度为1 个字节的二进制比特串。
- N-Gram就是N个连续的Gram，即一个长度为N字节的二进制比特串。
  - N-Gram的特征空间大小为$(2^8)^N$。

4.1 特征提取

- 基于CIG的恶意程序检测(CIG-MD)方法利用恶意程序特征，有效检测纯恶意程序和染毒程序。

这里引入CIG-B仅仅是为了进行对比实验，我们真正要在现实世界中使用的检测特征库是CIG-M。

后面的实验结果说明，虽然利用CIG-B可以有效分类合法程序和纯恶意程序，但对于染毒程序检测，基于CIG-B的恶意程序检测方法几乎等价于随机分类。

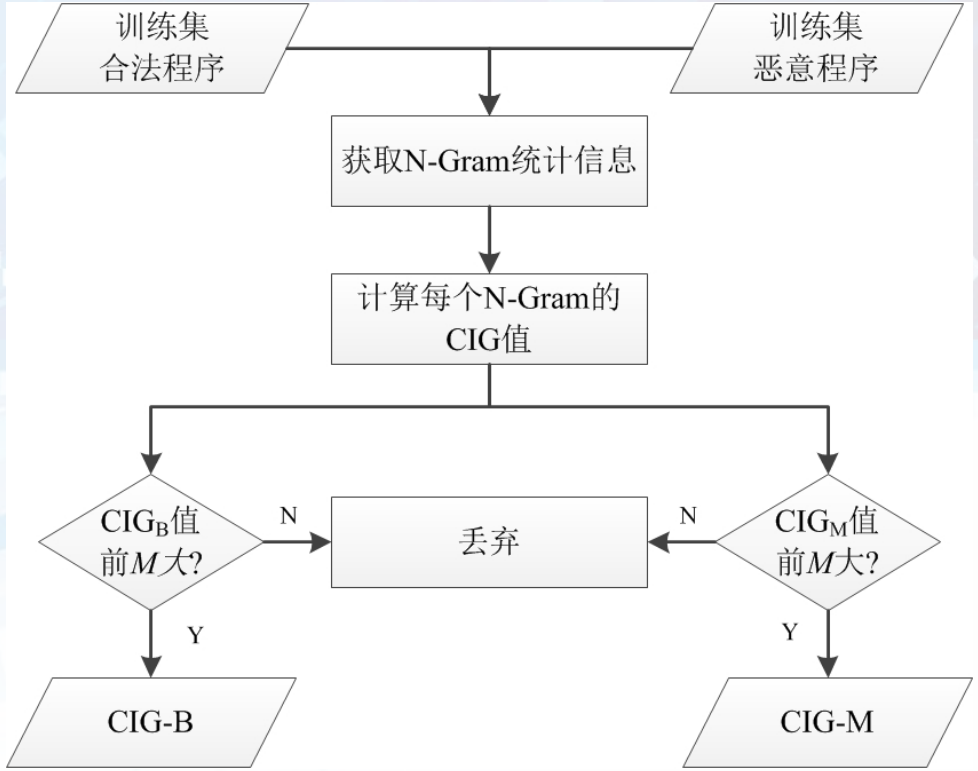


图18. 特征选择模块流程图

4.2 分类

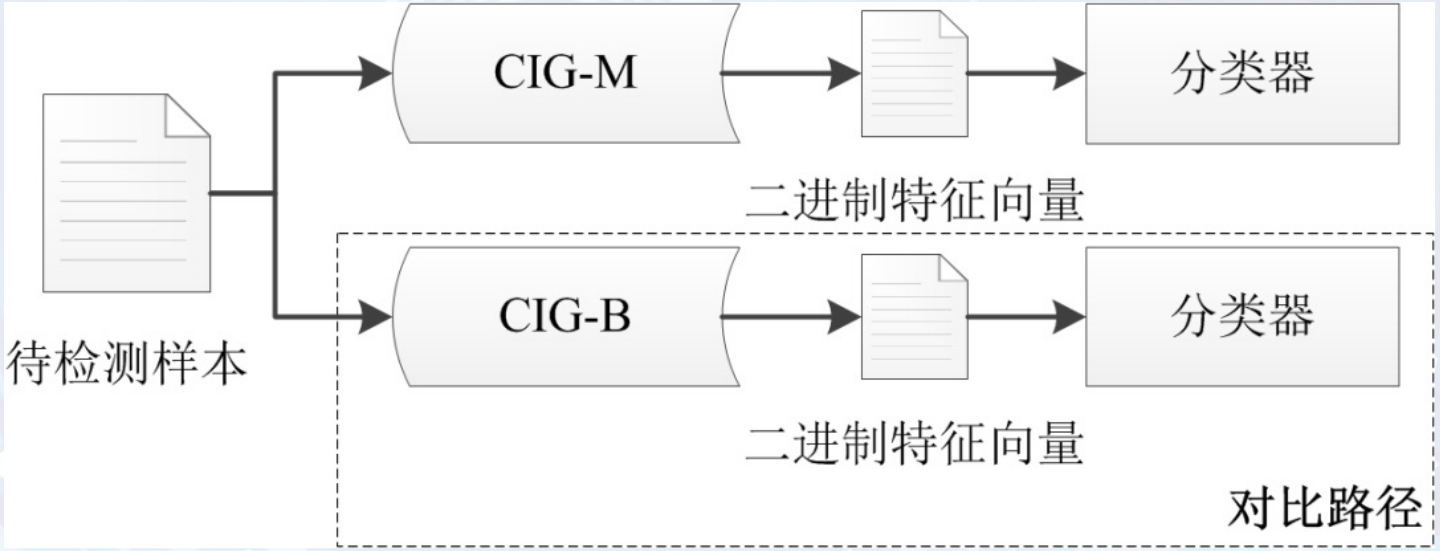


图19. 分类模块流程图

第一条路径利用CIG-M将待检测样本映射为二进制特征向量，这种方式将在现实世界中应用；
第二条路径则利用CIG-B提取待检测样本的二进制特征向量，仅用来开展对比实验。

4.2 分类

- 根据检测特征库（CIG-B或CIG-M）中的每个特征是否出现在一个待检测样本中，算法将此样本映射为一个长度为M的二进制特征向量$< x_1, x_2, \ldots, \sim x_M >$。 当特征库中的第i个特征出现在样本中时，$x_i = 1$，否则$x_i = 0$。
- 之后由训练好的分类器对待检测样本进行分类判别。

提纲

1. 引言
2. 恶意程序检测研究现状
3. 类别向导信息增益(CIG)
4. 基于CIG的恶意程序检测方法
5. **实验**
   - ✓ **5.1 数据集**
   - ✓ **5.2实验结果**
   - ✓ **5.3空间复杂度**
6. 总结

5.1 数据集

- 数据集（恶意程序来自VXHeavens数据集）

类型	合法程序	Backdoor	Constructor	Trojan	Virus	Worm	Miscellaneous
数量	1458	2200	172	2350	1048	351	1007

- 数据集设置
  - 测试集1(T_1)
    - 合法程序：原始合法程序
    - 恶意程序：原始恶意程序
  - 测试集2(T_2)
    - 合法程序：原始合法程序
    - 恶意程序：令T_1中的所有恶意程序采用在文件尾部插入恶意代码的方式感染T_1中的所有合法程序。生成的染毒程序是原始合法程序和恶意程序的结合体。

5.3 实验结果

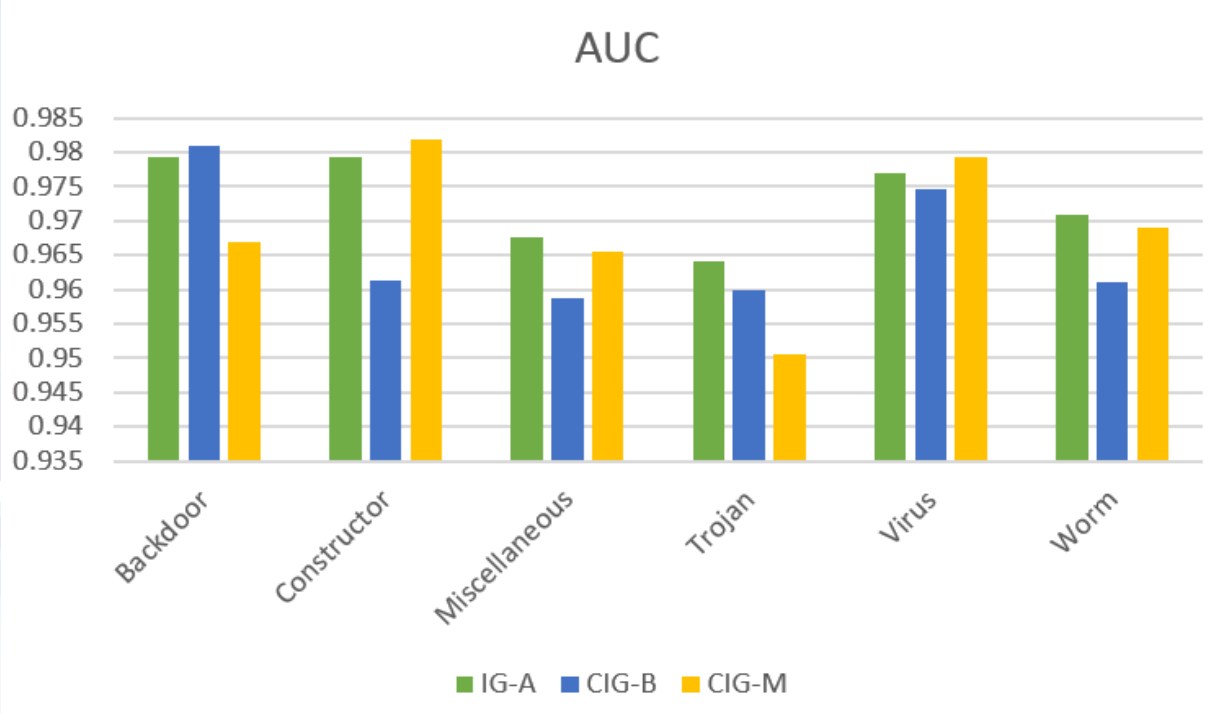


图21. 不同算法在T_1上的结果

IG-A为IG选择出的检测特征库。

- 在绝对数值上，IG-A、CIG-B和CIG-M的方法的实验结果相差不大

5.3 实验结果

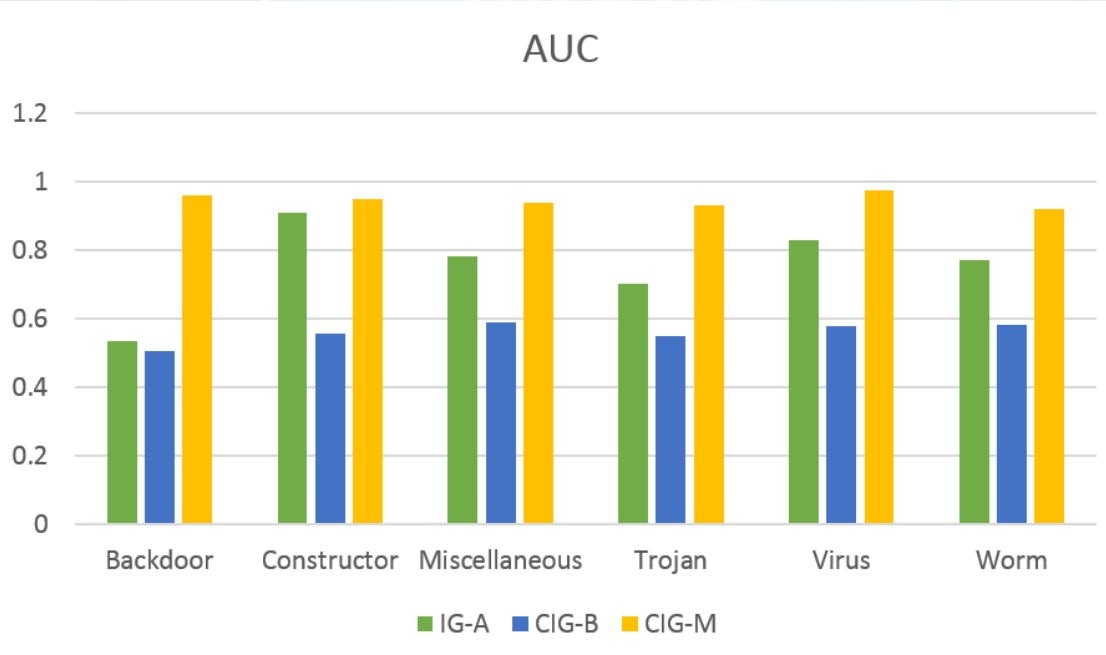


图22. 不同算法在T_2上的结果

- IG-A和CIG-B的性能急剧下降；
- CIG-M性能仍保持在高水平。

5.4 空间复杂度

- 特征选择阶段需要计算每个候选特征的CIG，这会耗费大量内存。
- 如果仅处理出现在恶意程序中的候选特征，那么那些仅出现在合法程序中的候选特征占用的内存就可以被削减掉。
- 利用这种方法，基于CIG的恶意程序检测方法空间复杂度大约降低**60%**。

提纲

1. 引言
2. 恶意程序检测研究现状
3. 类别向导信息增益(CIG)
4. 基于CIG的恶意程序检测方法
5. 实验
6. **总结**

6. 总结

- **类别向导的信息增益(CIG)**
- 在检测染毒程序时，合法程序特征是无效的。由于染毒程序是恶意程序在现实世界中的主要存在形式，因此基于合法程序特征的恶意程序检测方法是无法在现实世界中直接应用的。CIG可以将合法程序特征过滤掉；
- 基于CIG，提出了一种能够同时有效地检测纯恶意程序和染毒程序的通用恶意程序检测方法，取得了较好的检测效果；
- 实验结果表明：CIG能够为一个模式分类系统的每个类别分别选择重要特征，这些特征具有识别相应类别的能力，其比IG更加精细，极大地扩展了IG。

6. 总结

研究展望

- ✓ 结合计算机病毒的具体特点，充分挖掘和利用免疫机制，构建更加完善的恶意程序检测模型
- ✓ 结合程序结构化特征，实现更强的免疫协同效应
  - 程序结构化特征包括程序结构信息、应用程序编程接口调用信息等。
- ✓ 类别向导信息增益的应用扩展
  - 结合免疫原理和其他智能算法，提高CIG-MD方法对纯恶意程序和染毒程序的检测效果；
  - 探索将CIG应用其他相关领域，如垃圾电子邮件检测等。

感谢您的关注！

Thank you for your attention