

# 基于决策树模型的恶意程序判定方法

王维<sup>[1]</sup>, 肖新光<sup>[1]</sup>, 胡永涛<sup>[2]</sup>

(1. 安天实验室, 哈尔滨 150001)

(2. 公安部第三研究所, 上海 200031)

**摘要:** 在反病毒企业中, 对未知样本的分析处理和病毒的判定, 往往依靠大量的人工判定, 而每日捕捉上来的样本数量是非常大的, 这会导致对危险级别较高的样本往往得不到最优先的处理, 本文建立了一种基于决策树的恶意代码判定方法, 从大量的已知数据挖掘过程中学习并建立了决策树模型, 应用于未知样本的预分拣流程中, 准确率达到了 91.2% 以上, 取得了较为显著的判定效果。

**关键词:** 决策树; 病毒; 反病毒; 恶意代码; 恶意程序; 样本处理

## The Detection of Malware Based on the Decision Tree Model

Wang Wei<sup>1</sup>, Xiao Xin-guang<sup>1</sup>, Hu Yong-tao<sup>2</sup>

(1. Antiy Labs, Harbin, 150001)

(2. The Third Research Institute Of The Ministry Of Public Security, 200031)

**Abstract:** In the Anti-Virus enterprises, who sorting unknown sample files and determined that the virus often need to rely on a lot of artificial determination, but there is a very large number of samples were captured every day, this will lead to some high-risk level of the handling of samples are often not being processed first. This article has established one kind based on the Decision Tree's malicious code determination method, build the Decision Tree model based on a large number of known sample files, and has get the more then 91.2% correctness, and now applied to unknown sample pre-sorting process in Antiy Labs.

**key words:** Decision Tree; Virus; Anti-Virus; Malware; Malcode; Sample File;

## 1 反病毒工程的样本危机与算法选择

### 1.1 传统反病毒工程在样本分拣过程中的困境

几年前, 传统的反病毒企业每天捕获到的样本数量不多, 依靠病毒分拣人员的手工判定即可完成, 而近几年, 计算机得到了广泛的应用, 新的软件层出不穷, 同时, 随着病毒种类的日益增多, 用户无法自行判定可疑程序, 这个责任交给了反病毒公司。据统计, 反病毒机构每日可获得的样本数量达近万个。依靠纯手工分拣是无法完成的。于是在安天近几年的探索过程中, 先后建立了两代样本自动预分拣流程。现在应用了环形流水线工作体制, 这包括, 黑, 白名单对照扫描预处理, 以及神经网络算法等参与预分拣流程。所以自动化分拣流程在整个的样本分拣流程中起了决定性的作用。基本解决了传统式分拣样本的危机。但是样本压力的不断上升, 使安

天仍在不断地探索更高效的方法。

## 1.2 决策树算法在分拣过程中优势

在未知分拣的过程中，以前使用的了神经网络算法对未知文件的静态信息进行了挖掘，但是基于神经网络的模型有其先天性的不足：

第一，他无法获知网络的输入层中哪些信息中起了决定性作用，导致我们在使用训练好的网络模型时，不得不采集全部的数据，供其运行，这不仅降低了处理的效率，而且对于已训练好的网络模型中蕴含的知识，我们无法获得，以至无法用于其他相关项目的研究。

第二，神经网络需要的样本数量不宜过大，并且输入结点的元数据，即我们所能从 PE 文件中提取的数量与隐层数两者之间的配合较为困难，虽然一些经验理论可以起到一定的指导作用，但是应用到“变化粒度”的多层次模型中，情况将更为复杂。

而决策树正好可以弥补神经网络的这些缺陷，它可以方法地使用统计的方法得到输入数据，并且在使用过程中，有判定的速度快，判断流程直观的优点。

# 2 算法模型

## 2.1 决策树算法理论简介

决策树生成算法需要以下两个步骤

### 1. 树的生成

开始，数据都在根节点递归的进行数据分片。

停止分割的条件：

- (1) 一个节点上的数据都是属于同一个类别
- (2) 没有属性可以再用于对数据进行分割。

分割流程见图 1

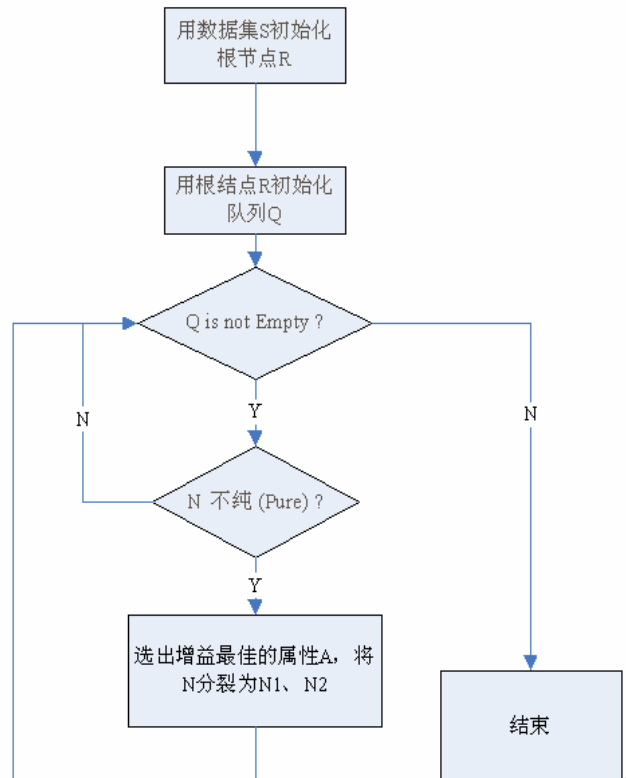


图 1 树分割流程

其中信息增益度量采用 (ID3/C4.5), 描述如下:

任意样本分类的期望信息:

$$I(s_1, s_2, \dots, s_m) = -\sum P_i \log_2(p_i) \quad (i=1..m)$$

其中, 数据集为 S, m 为 S 的分类数目,  $P_i \approx \frac{|S_i|}{|S|}$

$C_i$  为某分类标号,  $P_i$  为任意样本属于  $C_i$  的概率,  $s_i$  为分类  $C_i$  上的样本数  
由 A 划分为子集的熵:

$$E(A) = \sum (s_{1j} + \dots + s_{mj}) / s * I(s_{1j} + \dots + s_{mj})$$

A 为属性, 具有 V 个不同的取值

$$\text{信息增益: Gain}(A) = I(s_1, s_2, \dots, s_m) - E(A)$$

## 2. 树的修剪

去掉一些可能是噪音或者异常的数据, 由于篇幅关系, 这里不再详细介绍.

## 2.2 基于决策树算法样本分拣模型

### 2.2.1 准备训练数据:

我们首先构造了输入数据, 约 20 万条记录, 包括 12,326 个病毒文件和 91,856 个正常文件. 提取于可疑文件的来源社会属性和文件本身的静态属性. 总计属性约 150 项. 部分属性见表 1.

一级属性	二级属性	值	
获取渠道	是否用户上报	0/1	
	HoneyPot 捕获	0/1	
	诱饵信箱捕获	0/1	
	...		
PE 文件信息	MZ Dos Header	e_cblp e_cp e_crlc e_cparhdr e_minalloc e_maxalloc e_ss e_sp e_csum e_ip e_cs e_lfarlc e_ovno e_res_4 e_oemid e_oeminfo e_res2_10 e_lfanew	bin
	Coff PE Header	NumberOfSections TimeDateStamp PointerToSymbolTable NumberOfSymbols SizeOfOptionalHeader Characteristics	bin
	Optional Heade	MajorLinkerVersion MinorLinkerVersion SizeOfCode SizeOfInitializedData SizeOfUninitializedData AddressOfEntryPoint BaseOfCode BaseOfData ImageBase SectionAlignment	bin
	...		

数据二次挖掘	入口点在可执行节的外部		0/1
	入口点是否在节的首部		0/1
	节名是否是系统默认		0/1
	VC 编写		0/1
	Delphi 编写		0/1
	...		

表 1 训练属性部分列表

从决策树的算法可知，决策树模型要求了输入属性组与预测属性组的一维线性关系，所以适合于关系型数据库的分析统计。并且，对于决策过程的决大部分统计信息都可以采用 SQL 语法实现。

如我们中间运算中需要做如下统计：

(1) 在数据集中，有导出表的木马文件数比例：

```

Select Sum(IsTrojan) / Count(IsTrojan) As ExportRatio
From t_SampleFiles
Where IsTrojan = 1 And
      DataDirectory_0_VirtualAddress <> 0 And
      DataDirectory_0_Size <> 0

```

(2) 在数据集中有导出表的情况下，又有导入表的木马文件比例。

```

Select Sum(IsTrojan) / Count(IsTrojan) As ImportRatio
From t_SampleFiles
Where IsTrojan = 1 And
      DataDirectory_0_VirtualAddress <> 0 And
      DataDirectory_0_Size <> 0 And
      DataDirectory_1_VirtualAddress <> 0 And
      DataDirectory_1_Size <> 0

```

### 2.2.2 训练决策树：

根据上面介绍的算法，我们生成了如下图的决策树结果，由于结果数据庞大，见简化的结果图 2

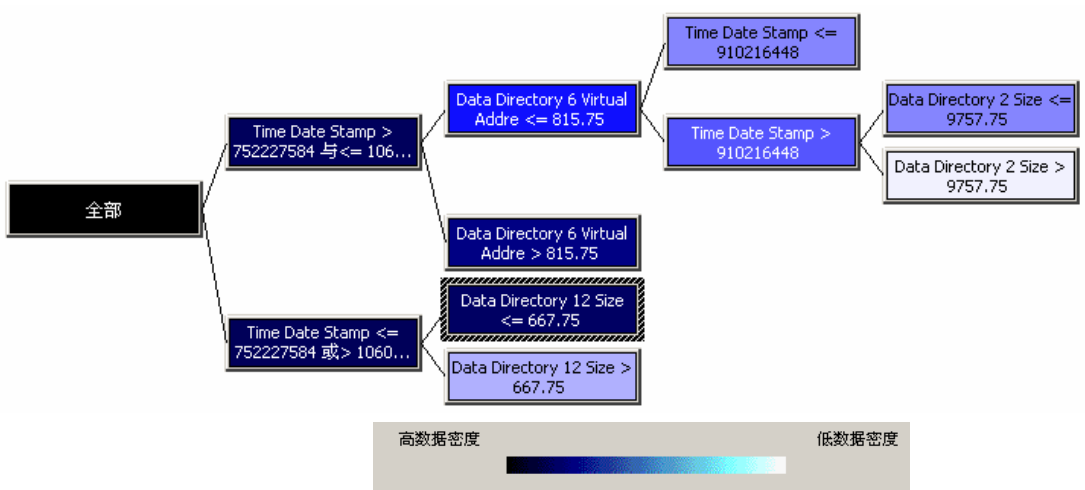


图 2 训练结果片段

叶结点结果见下表 2:

路径	木马文件	正常文件	木马比例
Data Directory 6 Virtual Address <= 815.75 与 Time Date Stamp > 752227584 与 <= 910216448	0	25	0%

Data Directory 2 Size <= 9757.75 与 Data Directory 6 Virtual Addre <= 815.75 与 Time Date Stamp > 910216448 与<= 1060230144	18	7	67.86%
Data Directory 2 Size > 9757.75 与 Data Directory 6 Virtual Addre <= 815.75 与 Time Date Stamp > 910216448 与<= 1060230144	0	11	0%
Data Directory 6 Virtual Addre > 815.75 与 Time Date Stamp > 752227584 与<= 1060230144	3	175	2.21%
Data Directory 12 Size <= 667.75 与 Time Date Stamp <= 752227584 或> 1060230144	219	7	96.07%
Data Directory 12 Size > 667.75 与 Time Date Stamp <= 752227584 或> 1060230144	2	16	19.29%
...	...	...	...

表 2 训练分析结果片断

### 2.2.3. 测试决策树:

我们准备了未参与学习过程的测试样本近 10 万条数据, 其中病毒样本 5,000 个, 正常样本 5,000 个. 最终的判定结果见表 3

综合判定结果	判定为病毒文件	判定为正常文件	准确率
病毒文件 5,000	4,791	209	95.8%
正常文件 5,000	668	4,332	86.6%
总计	5,459	4,541	91.2%

表 3 测试统计结果

最终我们得到了 91.2% 的准确率. 达到了 80% 的预期目标.

## 3 总结与展望

反病毒产品的作用是为了第一时间捕捉到样本, 并为用户提供解决方案. 决策树模型可以大大减轻传统的反病毒工程中大量的手工分拣任务负担, 并且让分析员优先分析更似病毒的样本, 提高分析处理的效率。

而我们在建模的过程中, 还有许多可提取属性尚未参与训练, 而随着正在进行的恶意代码行为自动分析项目的研究, 以后随着陆续地补充进去新的属性. 分析准确率会更精确, 并可以进一步提高效率等。”

### 参考文献:

- [1] Matt Bishop 计算机安全: 艺术与科学 (英文影印版). 清华大学出版社, 北京, 2004
- [2] (加)JIawei Han;MICHELINE Kamber, DATA MINING: CONCEPTS AND TECHNIQUES, SECOND EDITION ,MORGAN KAUFMANN, 2007
- [3] Willian H.Inmon数据仓库. 机械工业出版社, 北京, 2006