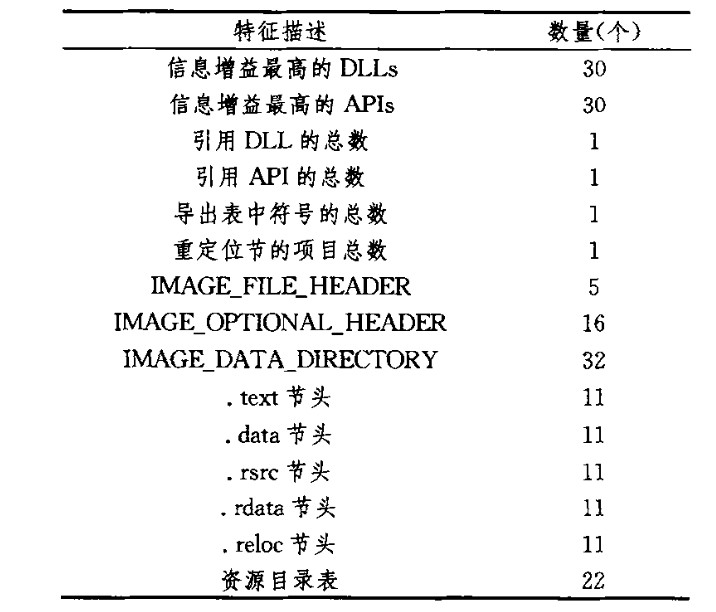
**论文总结**

**一.《基于PE静态结构特征的恶意软件检测方法》**

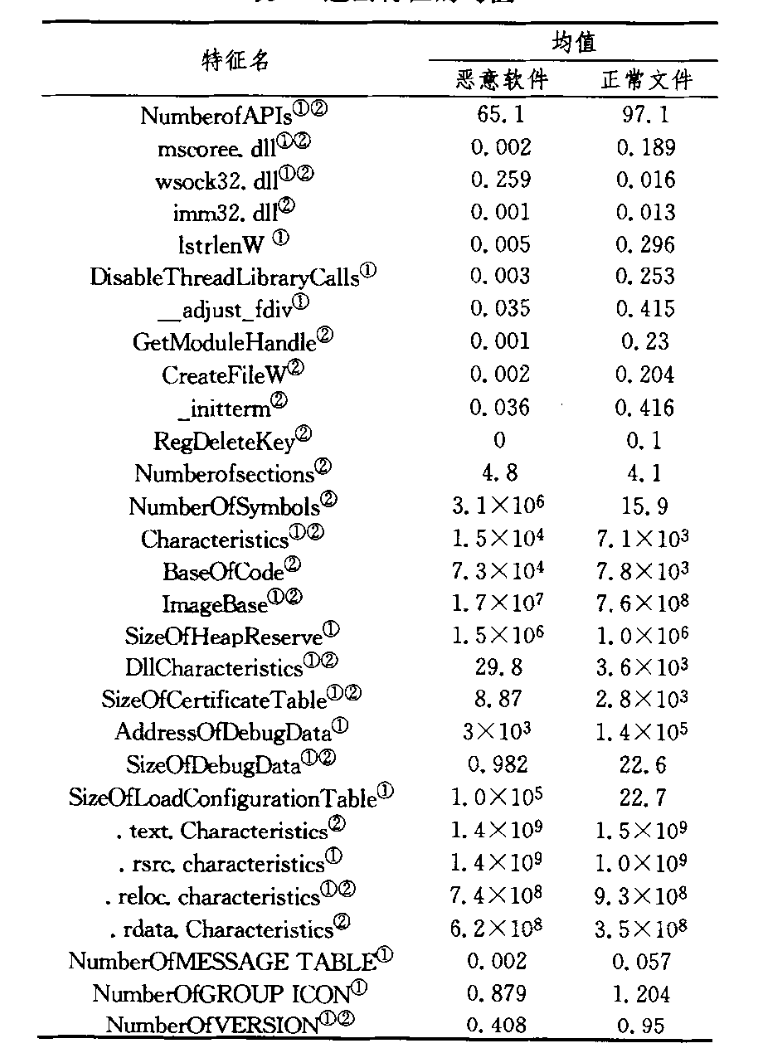
1.概述：通过分析静态PE文件的结构，提取出PE文件的初始特征约200个左右，采用了两种特征选择的方法筛除了大部分特征，一种是Filter特征选择方法（选择和目标函数关联度较大的特征子集）中的Cfs-SubsetEval，另一种是Wrapper特征选择方法（将学习算法的性能作为特征选择的指标）。使用了几种机器学习的分类算法，将80%的数据作为训练集，20%的数据作为测试集，利用C4.5、随即森林、集成学习的方法来进行分类。

2.特征选择：

初始特征集：



经过两种特征选择算法后选取的特征：



2.1恶意软件的DLL和API都比正常文件的少，为了隐藏功能，不在导入表中引用DLL和API，而是使用Loadlibrary显式地加载动态库。

2.2恶意软件 的调试数据也明显小于正常文件的，这是因为恶意软件为了增 加调试的难度，很少有调试信息

2.3恶意软件4个节．text、．rsrc、．reloc、．rdata节的Characteristics属性和正常文件的比也有明显差异，Characteristics属性通常指定该节是否可读、 可写、可执行等等，部分恶意软件的代码节存在可写异常，只读数据节和资源节存在可写、可执行异常等。

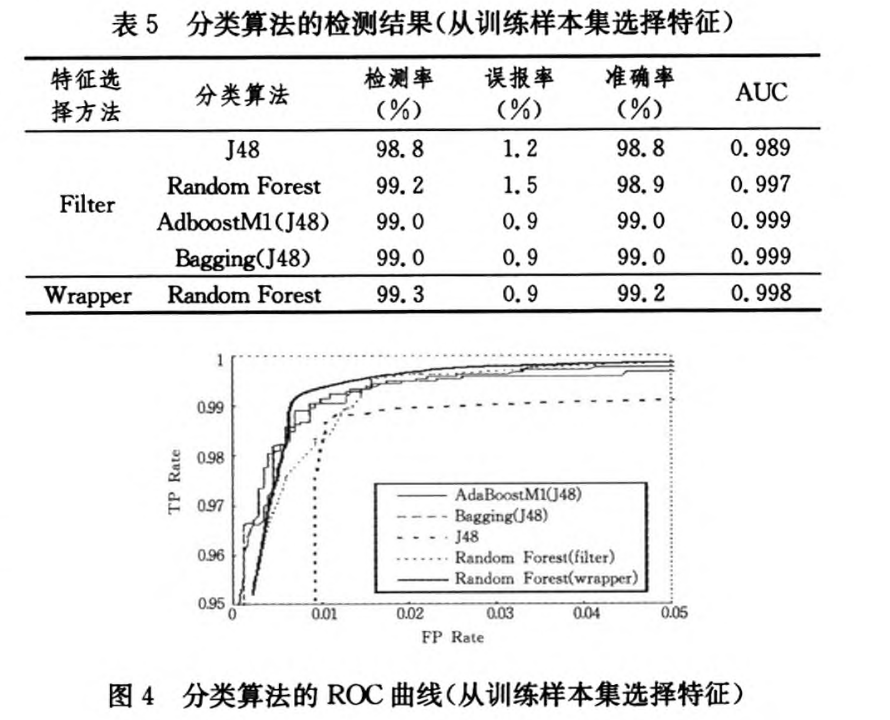
2.4恶意软件资源节的资源个数也明显小于正常文件的，如消息表、组图表、版本资源等，这是因为恶意软件很少使用图形界面资源，也很少有版本信息。

3.实验结果：

准确率Accuracy：所有被正确分类的数量在待测集合中所占比例。

检测率：被正确分类的恶意软件数目在待测集合的所有恶意软件中所占比例。

误报率：被错误分类成恶意软件的正常软件在待测集合的所有正常软件中所占比例。



**二.《Selecting Features to Classify Malware》**

1.概述：本文从静态PE文件中不同部分共选择了7个特征。因为从不同的部分选择，理论上说这些特征的相关性是最小的，也能最好地判别正常PE文件和恶意样本。首先初始有645个特征，利用对恶意软件分析的经验选择了100初始特征，利用随机森林算法对这100个特征都单个的跑一遍，并对单个特征的准确率从高到低排列。并对这些特征按照在PE文件中的位置分成七个Buckets，分别标注为Data Directory，Optional Header，Imports，Exports，Resources, Sections and File Header。每次从一个Bucket中选出单个准确率最高的特征并且未被选中的特征，不断加入特征集并观察多个特征同时使用的准确率。最终选择了7个特征（因为选到8个特征的时候准确率不增了或者下降了），得到了最小特征集。

2.特征选择：

初始特征集：最开始有645个特征（具体是什么没给）。再利用经验从PE header、10个sections、输入输出表中的各个字段中选择了100个特征（也没有给出）。

经过算法分析后选择的特征：Debug Size、Image Version、Iat RVA、Export Size、Resource Size、Virtual Size2、Number of Sections这7个特征。

Debug Size：记录了Debug目录表的大小。一般正常PE文件的Debug目录不为空。

Image Version：标识了PE文件的版本，一般来说正常PE文件该字段都不为0的，而许多恶意软件的该字段为0。

Iat RVA：输入地址表的相对地址。一般正常文件是4096，而恶意软件的该字段值为0或者一个很大的数。因为许多恶意软件不使用导入函数或者会混淆其输入表。

Export Size：记录了导出表的大小。通常只有DLL文件或者非可执行文件有导出表，恶意代码的该字段一般为0。

Resource Size：一般正常文件会有许多资源，而恶意软件为了隐藏身份，几乎没有资源，该字段都是0。

Virtual Size2:许多恶意代码只有一个Section，该字段一般为0。表示了第二Section（不清楚第二Section是什么）的大小。

Number Of Sections：标识了PE文件中Section的数量，正常文件以及恶意代码该字段各种值都有，暂不清为什么该特征能较好区分正常文件以及恶意代码（文献中原话）。

3.实验结果：

将含有100000个恶意代码和16000个正常PE文件的数据集均分成了5个部分。并利用之前选取的7个特征，分别使用IBk、J48、J48 Graft、PART、Random Forest、Ridor等机器学习算法进行分类操作，并且使用10倍交叉验证。

True Positive：恶意样本被判定成恶意。

True Negative：正常PE样本被判定为正常。

False Positive：正常PE文件被错判成恶意。

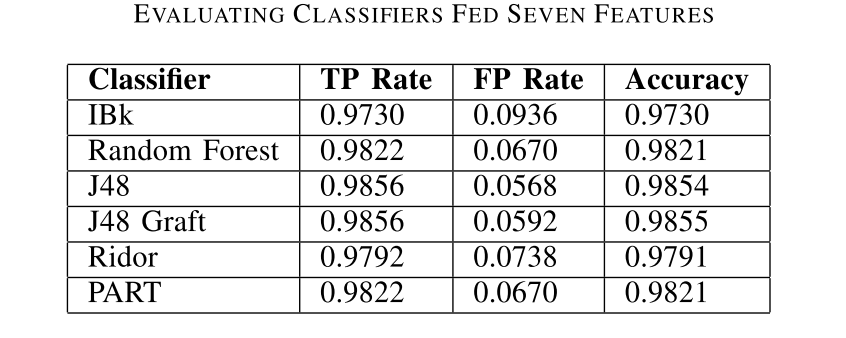
False Negative：恶意样本被错判为正常。

TP Rate：正阳/（正阳+假阴）

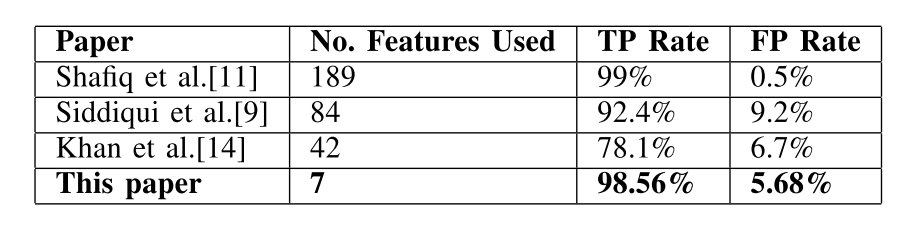
FP Rate：假阳/（正阴+假阳）

Accuracy：（TP+TN）/（TP+TN+FP+FN）

使用了以上六种机器学习算法，得到的几个率如下：



和其他类似的实验相比：



**三.《PE-Miner: Mining Structural Information to Detect Malicious Executables in Realtime》**

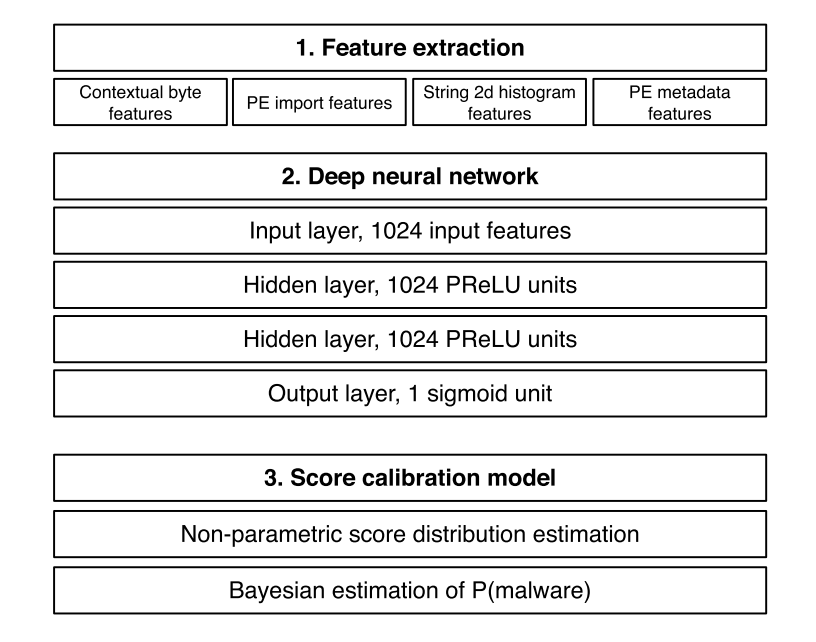
1.概述：

2.特征选择：

3.实验结果：

**四．《Deep Neural Network Based Malware Detection Using Two Dimensional Binary Program Features》**

1.概述：本文搭建了一个四层的神经网络对静态PE文件进行了分类。选取了4个部分的特征：第一部分，对二进制PE文件使用滑动窗口扫描，窗口大小为1024字节，步长为256字节，得到了字节熵直方图，计算得到的1X256的向量（没有源代码，细节还没弄明白）；第二部分，关于PE文件的导入特征，对导入的（DLL，函数名）对做哈希运算，映射到一个1X256的向量的某个位置上，得到一个1X256的向量；第三部分，直接从PE文件的某些字段获得（和我们的论文一样），也是得到一个1X256的向量。神经网络方面：输入维度是1024，同时含有两个隐藏层，均有1024个结点，输出层输出0或1。



2.特征选择：

3.实验结果：