

# 面向恶意 PDF 文档分类的对抗样本生成方法研究

刘超<sup>1</sup>, 娄尘哲<sup>1,2</sup>, 喻民<sup>1,2</sup>, 姜建国<sup>1</sup>, 黄伟庆<sup>1</sup>

<sup>1</sup>中国科学院信息工程研究所 北京 中国 100093

<sup>2</sup>中国科学院大学网络空间安全学院 北京 中国 100093

**摘要** 通过恶意文档来传播恶意软件在现代互联网中是非常普遍的,这也是众多机构面临的最高风险之一。PDF文档是全世界应用最广泛的文档类型,因此由其引发的攻击数不胜数。使用机器学习方法对恶意文档进行检测是流行且有效的途径,在面对攻击者精心设计的样本时,机器学习分类器的鲁棒性有可能暴露一定的问题。在计算机视觉领域中,对抗性学习已经在许多场景下被证明是一种有效的提升分类器鲁棒性的方法。对于恶意文档检测而言,我们仍然缺少一种用于针对各种攻击场景生成对抗样本的综合性方法。在本文中,我们介绍了PDF文件格式的基础知识,以及有效的恶意PDF文档检测器和对抗样本生成技术。我们提出了一种恶意文档检测领域的对抗性学习模型来生成对抗样本,并使用生成的对抗样本研究了多检测器假设场景的检测效果(及逃避有效性)。该模型的关键操作为关联特征提取和特征修改,其中关联特征提取用于找到不同特征空间之间的关联,特征修改用于维持样本的稳定性。最后攻击算法利用基于动量迭代梯度的思想来提高生成对抗样本的成功率和效率。我们结合一些具有信服力的数据集,严格设置了实验环境和指标,之后进行了对抗样本攻击和鲁棒性提升测试。实验结果证明,该模型可以保持较高的对抗样本生成率和攻击成功率。此外,该模型可以应用于其他恶意软件检测器,并有助于检测器鲁棒性的优化。

**关键词** 恶意PDF文档; 对抗样本; 文档分类; 样本生成; 鲁棒性

中图法分类号 TP181/TP393.0 DOI号 10.19363/J.cnki.cn10-1380/tn.2023.09.02

## Research on Adversarial Example Generation Method for Malicious PDF Document Classification

LIU Chao<sup>1</sup>, LOU Chenzhe<sup>1,2</sup>, YU Min<sup>1,2</sup>, JIANG Jianguo<sup>1</sup>, HUANG Weiqing<sup>1</sup>

<sup>1</sup> Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China

<sup>2</sup> School of Cyber Security, University of Chinese Academy of Sciences, Beijing 100093, China

**Abstract** Spreading of malware via malicious documents is very common in the modern Internet and is one of the highest risks faced by many organizations. PDF documents are the most widely used document type worldwide, and as a result, there are countless attacks caused by them. The use of machine learning methods for malicious document detection is a popular and effective approach, but the robustness of machine learning classifiers has the potential to expose certain problems in the face of well-designed samples from attackers. In the field of computer vision, adversarial learning has proven to be an effective method for improving the robustness of classifiers in many scenarios. For malicious document detection, we still lack a comprehensive approach for generating adversarial examples for various attack scenarios. In this paper, we introduce the basics of PDF file formats, as well as effective malicious PDF document detectors and adversarial sample generation techniques. We propose a model to generate adversarial examples for adversarial learning in the area of malicious documents detection, and use the generated adversarial examples study the detection effectiveness (and evasion effectiveness) for hypothetical scenarios with multiple detectors. The key operations of the model are association feature extraction and feature modification, where association feature extraction is used to find the associations between different feature spaces and feature modification is used to maintain the stability of the examples. The final attack algorithm leverages the idea of momentum-based iterative gradient to boost the success rate and efficiency of generating adversarial examples. We combined some convincing datasets and rigorously set up the experimental environment and metrics, followed by tests against example attacks and robustness enhancement. Experimental results confirmed that the proposed model can maintain a high level of generation rate and success rate. Moreover, this model can be applied to other malware detectors and contribute to robust optimization.

**Key words** malicious PDF document; adversarial example; document classification; example generation; robustness

通讯作者: 喻民, 博士, 高级工程师, Email: yumin@iie.ac.cn。

本课题得到中国科学院青年创新促进会(No. 2021155)资助。

收稿日期: 2020-03-22; 修改日期: 2020-04-11; 定稿日期: 2023-01-09

## 1 引言

恶意 PDF(Portable Document Format)文档是指在正常 PDF 电子文档中直接添加恶意代码, 或使用链接跳转、点击触发等方式远程下载恶意代码程序, 利用电子文档解析程序的漏洞使恶意代码得以执行, 实现攻击者的恶意目的。PDF 文档是全世界应用最广泛的文档类型, 兼容性高、体积小, 人们普遍将其看作高效、便捷与安全的信息交互载体, 攻击者正是利用这种普遍认知, 大肆利用恶意 PDF 文档进行攻击。PDF 文档不仅具有复杂的文档结构, 并且支持嵌入 JavaScript 等脚本语言, 在文档中还可以内嵌其他类型的文件。复杂的文档结构为恶意代码提供了隐藏和存储空间, 支持脚本语言则为恶意代码的执行提供了可能。2019 年以来, 恶意 PDF 文档已经成为实施高级持续威胁(Advanced Persistent Threat, APT)的重要载体<sup>[1]</sup>。同时可以被恶意代码利用的文档阅读器漏洞不断涌现, 对恶意 PDF 文档检测的研究具有重大的意义。

机器学习技术在许多应用中都取得了非常显著的成功, 如网络入侵检测, 面部识别和自动驾驶汽车。目前基于机器学习的检测器已经成为检测恶意 PDF 文档的主流方法<sup>[2]</sup>。然而大量的例子表明, 当基于机器学习的系统被部署到应用程序中时, 很容易被对抗攻击影响, 在恶意 PDF 文档检测领域的对抗攻击主要表现为利用精心构造的输入样本使得检测器在预测时输出错误的结果。例如, 攻击者可能会对一个恶意样本进行修改使它在测试时显示为良性, 从而逃避目标检测器, 这种修改后的样本称为对抗样本<sup>[3]</sup>。如果将对抗样本用于恶意 PDF 文档, 会使主流的机器学习检测器失效, APT 攻击就更容易实现, 给组织甚至个人带来更严重的危害。基于军备竞赛的思想<sup>[4]</sup>, 研究对抗样本的生成方法有助于提高检测器的鲁棒性。

在计算机视觉领域, 研究人员已经提出多种生成对抗样本的方法<sup>[5-8]</sup>。但 PDF 文档具有更复杂的结构和更多依赖性强的特征, 攻击者修改样本的能力更加灵活。早期逃避文档检测器的方法主要有畸形文档和模仿攻击, 利用文档格式规范的缺陷以及文档解析器的处理逻辑漏洞, 将与恶意行为相关的特征隐藏起来, 使检测器难以提取。Šrndić 等<sup>[9-10]</sup>利用一种简单的梯度下降攻击算法在特征空间中生成可以逃避检测器的恶意 PDF 对抗样本。Xu 等<sup>[11]</sup>提出了一种使用遗传算法的技术来评估检测器的鲁棒性<sup>[12]</sup>, 并生成相应的恶意 PDF 对抗样本。Dang 等<sup>[13]</sup>利用真

值评分机制实现了 EvadeHC 方法。尽管这些逃避攻击增加了更多的约束, 但他们考虑的场景仍然过于简单, 这些生成方法仅针对特定类型的检测器生成相应的对抗样本, 从而带来了通用性和可传递性的不确定性。

为了提高检测器的鲁棒性, 最直接的方法是寻找多种场景下可以逃避检测器的对抗样本, 利用生成的对抗样本进行对抗训练。本文构造了与以往使用的对抗攻击方法不同的攻击场景, 即多个不同检测器串联的联合检测器, 这也是在线检测引擎常用的场景。在这种场景下, 由于不同的检测器使用不同数量或类型的特征, 以往的针对性较强的对抗样本会失去作用。针对此攻击场景本文首先提出对于一个被判定为恶意的 PDF 文档, 它在不同特征空间中提取出的特征应该是具有关联性的。目前使用机器学习方法检测恶意 PDF 文档时, 基于内容的特征和基于结构的特征是两类主要特征, 它们也是分类的基础特征。通过使用 Apriori 算法<sup>[14]</sup>, 可以有效分析不同类型特征之间的关联, 并从关联规则中探索与恶意行为有关的特征。然后基于获得的关联规则为文档的修改提供了一种匹配方法, 这对基于特征空间的逃避攻击非常重要。最后将这些工作与基于动量迭代梯度的方法相结合<sup>[15]</sup>, 形成一个完整的攻击模型。实验表明, 本文的方法生成的对抗样本可以成功地逃避使用不同特征空间的恶意 PDF 文档检测器, 将其用于对抗训练可以有效提升检测器的鲁棒性。

本文主要贡献如下:

(1) 本文提出了针对恶意 PDF 文档分类的对抗样本生成方法 MissJoint, 并基于该方法实现了对抗样本生成模型, 该模型生成的对抗样本具有较高的攻击成功率, 同时也保留了原始样本的恶意性。

(2) 改进了基于动量迭代的梯度下降方法, 提高了文档型对抗样本生成的效率。

(3) 设计了针对参数调整、攻击效果及鲁棒性提升等多组实验, 实验结果表明本文提出的方法具有较高的对抗样本生成率和攻击成功率。

本文的其余部分安排如下: 第 2 章介绍了与研究相关的基础知识和相关工作; 第 3 章详细论述了恶意 PDF 文档对抗样本生成方法; 第 4 章报告了实验结果和分析; 最后第 5 章对本文进行总结并提出下一步工作。

## 2 背景知识和相关工作

本节介绍相关的工作, 包括 PDF 文件格式, 机器学习方法和用于支持本研究的目标检测器。最后,

将介绍对抗样本生成方法的理论。

## 2.1 PDF 文件格式

PDF 文档是一种开放标准格式,是为了在不同平台上显示统一格式的内容和布局而产生的<sup>[16]</sup>。PDF 文档结构由 4 个部分组成: 文件头, 文件体, 交叉引用表和文件尾, 如图 1 上部分所示。文件头包含 PDF 字样和格式版本。文件体是 PDF 文档中最重要的部分, 它包含一组构成文档内容的 PDF 对象。这些对象可以具有 8 种基本类型之一: 布尔值, 数字, 字符串, 流, 名称, 数组, 字典和 Null 对象。对象之间可以间接引用, 文档解析器在解析 PDF 文档时将对象的间接引用关系作为解析顺序, 从而构成 PDF 文档的逻辑结构, 如图 1 下部分所示。文档逻辑结构中的节点是对象或数组, 因为它们对应于各个元素的整数索引, 并且每条边与子对象在父对象中的名称相对应。交叉引用表为文件体中的对象索引, 文件尾则提供查找交叉引用表和特殊对象的方法。

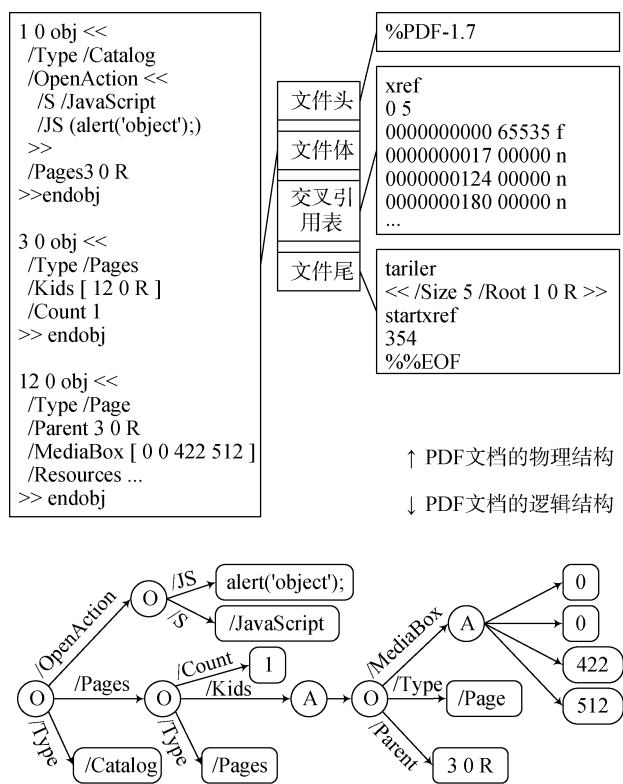


图 1 PDF 文档的物理和逻辑结构

Figure 1 The physical and logical structure of a PDF document

## 2.2 机器学习及分类

有监督机器学习在信息安全任务中十分流行, 它通常抽象地考虑问题。首先, 特征提取器从训练样本中提取特征, 然后指定选择规则以减少特征数量并形成特征空间  $D$ , 其中训练数据集  $D_t$  中的每个向

量化样本均表示为一个点。同时, 标签空间  $L$  被定义并分配给每个训练样本, 用于识别类别, 比如良性和恶性。分类器的目标是找到一个决策函数  $f$ , 该函数将  $D$  中的数据点映射到具有低预测误差的不同类别中。该函数在训练集上有良好的表现, 基于平稳性假设可以相信测试点与已知训练点的分布  $P$  是一致的。

机器学习技术已广泛用于恶意 PDF 文档检测: 预测未知的 PDF 文档是否被正确标记为恶意或良性。特征的提取至关重要, 因为特征的质量可能会不同程度地影响预测性能。2.3 节将讨论基于不同类型进行特征提取的两个代表分类器。

## 2.3 目标检测器

早期的工作已经提出了几种基于机器学习的恶意 PDF 文档检测器。例如, 某些工具提取 JavaScript 代码以进行静态或组合分析。由于 PDF 文档的复杂性和攻击技术的进步, 出现了各种混淆方法。之后研究人员一直在寻找提升机器学习分类器准确率的新型特征类型, 最新的检测器使用基于内容的特征和基于结构的特征。基于内容的特征与 PDF 内容相关, 主要关注特定关键字(名称, 位置, 数量和长度), 元数据(文件大小, 作者, 时间戳和创建日期)的存在与否, 以及文件中的间接对象或流对象(Javascript)。基于结构的特征也包括与 PDF 结构相关的特定关键字; 还有另一个特殊的特征, 称为逻辑结构路径。逻辑结构路径间接描述了 PDF 文档的解析过程, 是 PDF 文档逻辑结构中一系列边的序列, 这些序列以根对象开始, 以第 2.1 节中描述的对象结尾。

本文根据文献[17]中发布的基于机器学习的 PDF 检测器, 选择了 4 个检测器作为攻击目标: PJScan<sup>[18]</sup>, PDFRate<sup>[19]</sup>, Hidost<sup>[20]</sup>和 Slayer NEO<sup>[21]</sup>。其中, PJScan, PDFRate 和 Slayer NEO 使用关键字, 元数据或内容作为特征, 而 Hidost 使用逻辑结构路径作为特征。PDFRate 和 Hidost 是最前沿的高精度检测器, 而且与 2.4 节中描述的攻击中使用的目标检测器一致。本文将 4 个检测器串联形成最终的联合检测器, 注意这与集成分类器不同, 该检测器定义了更严格的输出标准, 只要其中一个的预测显示为恶意, 则整体输出即为恶意。

## 2.4 对抗样本生成方法

针对恶意 PDF 文档检测器的对抗攻击主要聚焦于逃避攻击, 因为在恶意软件检测任务中, 最直接的目的就是修改恶意样本, 产生具有恶意行为的对抗样本, 在测试时使得检测器无法识别, 从而破坏系统完整性。与图像识别任务中的对抗样本类似, 文

档型对抗样本的生成也有两类不同的方法: 基于内容的方法和基于特征空间的方法。

基于内容的方法是生成对抗样本的一种直观方法, 攻击者主要利用官方文档规范中的不严谨或机器学习系统的缺陷来修改文档的内容。畸形文档攻击、模仿攻击和反向模仿攻击是典型的基于内容的攻击方法。

基于特征空间的方法利用数据样本向量化特点, 在特征空间中修改特征向量从而生成对抗样本。这类方法在检测模型的决策边界对样本的特征向量进行迭代修改, 由此产生的对抗样本实际上逃避了机器学习系统的分类器组件。攻击情形通常描述如下: 给定一个检测器  $f(x)$ , 它将输出标签  $y$  作为一个特征向量  $x$  的预测; 攻击者使用精巧的手段修改样本  $s$ ,  $s$  可以被向量化表示为  $x(s)$ , 修改后的样本为  $s'$ , 同时产生一个新的向量  $x' = x(s')$ ; 攻击者的目的为使

$f(x')$  返回的标签是错误的, 同时  $s'$  保留了原始样本  $s$  的恶意功能。基于特征空间产生对抗样本可以大大降低复杂性, 典型的方法包括 EvadeML<sup>[11]</sup> 和 Mimicus<sup>[10]</sup>, 它们也是最先进的对抗样本生成方法, 本文的方法将与它们进行比较。

### 3 对抗样本生成

本节主要介绍对抗样本生成的思路、方法和实现。主要包括概述、关联规则提取、特征修改、攻击算法、鲁棒性提升等部分。

#### 3.1 模型概述

本文的对抗样本生成技术以梯度下降攻击为基础在特征空间中寻找对抗样本。关键在生成操作之前, 需要完成独有的特征关联分析和特征修改, 同时在生成方法上进行了适当改进, 采用基于动量迭代的梯度下降, 整体模型如图 2 所示。

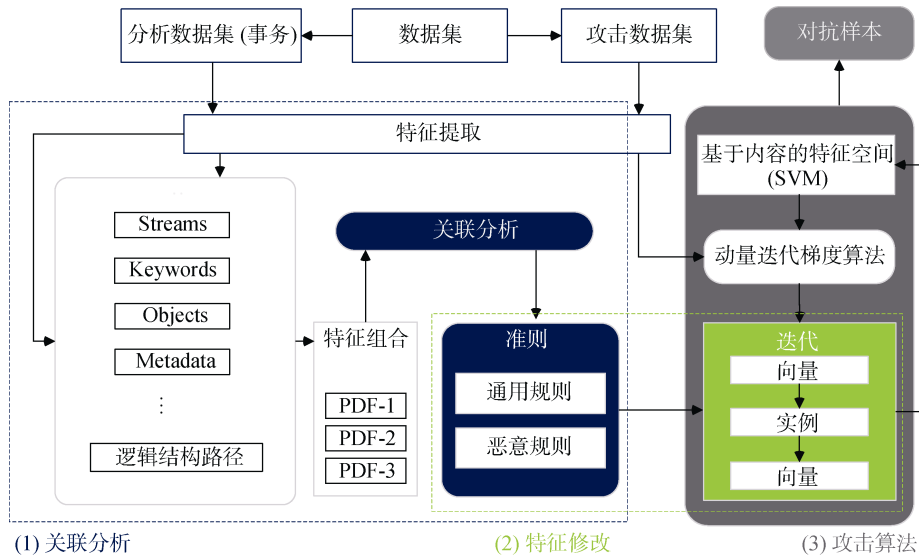


图 2 生成对抗样本技术模型

Figure 2 The model to generate adversarial examples

首先, 我们需要从数据集的样本中提取并选择所需要的特征, 将这些特征结合在一起形成关联分析集。通过将关联分析算法用于随机选择形成的恶意样本和良性样本, 获得一组包含特征关联的规则。

其次, 获得的规则分为 2 条准则: 1) 通用型关联规则可以执行同步修改; 2) 恶意型关联规则作为约束不可以进行修改。根据这 2 条准则在生成对抗样本的过程中修改实际样本, 称为特征修改算法。

最后, 我们将对选定的恶意 PDF 文档进行向量化处理, 并将该向量用作特征空间中的初始点。在迭代逼近目标点的过程中, 使用准则 2) 进行方向的微调。在将向量映射到实例的过程中则使用准则 1)。

直到找到对抗样本或达到生成方法设置的迭代阈值, 生成过程结束。

#### 3.2 关联规则提取

对特征进行关联分析即挖掘特征之间的关联规则, 本文采取 Apriori 算法, 利用逐层搜索的迭代方法找出数据库中项集的关系。项集的概念即为项的集合, 项集出现的频率是指包含该项集的事务数。如果某项集满足最小支持度, 则称它为频繁项集。因此 Apriori 算法将任务分解为 2 个主要的子任务:

##### (1) 频繁项集生成

将提取得到的特征全部放到一个集合中, 定义为关联项集  $R$ 。从数据集中提取每个样本的项集( $R_n$ )

用于关联分析, 从而建立一个包含所有样本项集的事务集  $N$ , 简单实现如下所示。

```

1 Combine ( $F_1, F_2, F_3, \dots, F_n$ )  $\rightarrow R$ 
2 For ( $i = 1; i \leq n; i++$ )
3    $R_i = \text{Extract } R \text{ from } S_i$ 
4 End for
5 Combine ( $R_1, R_2, R_3, \dots, R_n$ )  $\rightarrow N$ 

```

为了保证对比实验的公平性, 本文进行关联分析时主要使用两类特征: 第一类是 135 个基于内容的特征, 与 PDFRate 和 Mimicus 保持一致, 它们是一些对象关键字、对象属性及元数据的抽象统计, 表 1 中列出了部分特征及描述; 第二类是 1000 个基于结构的特征, 与 Hidost 和 EvadeML 保持一致, 它们是可以精确描述 PDF 结构的结构路径, 表 2 中列出了部分特征; 由于最新的 Hidost 更新提出结构路径特征会随时间更新, 因此本文最终提取的结构路径特征将与 EvadeML 所使用的特征稍有不同, 这将在结果分析中再次讨论。基于未来的研究添加更多新型、有效的特征可以有效扩展关联分析方法。

表 1 基于内容的特征

Table 1 Content-based features

author_len	元数据作者字段字符数量
author_num	元数据作者字段数字字符数量
creator_num	元数据创作者字段数字字符数量
size	PDF 文档大小(字节数)
count_acroform	Acroform 对象标记统计数
count_action	Action 对象(AA、OpenAction)标记统计数
count_endobj	对象结束标记统计数
count_image_total	image 对象统计数
count_javascript	JavaScript 对象标记统计数
count_page	page 标记统计数
image_totalpx	图像像素和统计数
keywords_num	关键字数字字符统计数
keywords_len	关键字字符统计数
len_stream_min	Stream 对象与下一个对象终止标记的最小距离
pos_acroform_avg	Acroform 对象标记的平均归一化位置
producer_num	元数据修改字段数字字符数量
pdfid1_lc	小写字符统计数
ratio_size_obj	对象统计数与 size 的比值
subject_mismatch	subject 值的差异统计数
version	从标题中提取的 PDF 版本值

我们的任务是从事务集  $N$  中生成所有的频繁项集  $F$ , 过程如算法 1 所示。我们让频繁项集的长度从  $k = 1$  开始, 并在循环语句中重复运算, 直到没有新的频繁项集被识别。其中的操作包括生成候选集

表 2 基于结构的特征

Table 2 Structure-based features

/Metadata/Length	/Metadata
/Metadata/Type	/Metadata/Subtype
/AcroForm/Fields	/AcroForm
/AcroForm/DR/Encoding/ PDFDocEncoding	/AcroForm/DR/Encoding/ PDFDocEncoding/Type
/AcroForm/DR/Encoding/ PDFDocEncoding/Differences	/OpenAction/MediaBox
/OpenAction/Contents/Filter	/OpenAction/Contents
/OpenAction/Contents/Length	/OpenAction/Resources
/OpenAction/Resources/ProcSet	/PageLayout
/Pages/MediaBox	/PageLabels
/Pages/Resources/ProcSet	/PageLabels/Nums
/Pages/Resources	/Names/EmbeddedFiles
/Names/JavaScript/Names/S	/PageLabels/Nums/S
/Threads	/Pages/Rotate
/ViewerPreferences/Direction	/AcroForm/DA
/Outlines/Type	/Outlines/Count
/Pages/Kids	/OpenAction/D/Resources/ProcSet et
/StructTreeRoot/RoleMap	/Type

(Candidate itemsets)、计算支持度(support)和排除不常见的候选。最后产生的每个频繁项集都是若干个(随最小支持度设定的不同而不同)特征的集合。

#### 算法 1: 频繁项集生成

输入: 事务集  $N$ , 关联项集  $R$ .

输出: 频繁项集  $F$ .

```

1  $C_k$ : Candidate itemsets of size  $k$ ;
2  $F_k$ : frequent itemsets of size  $k$ ;
3  $F_1 = \{\text{frequent 1-itemsets}\}$ ;
4 For ( $k=1; F_k \neq \emptyset; k++$ )
5    $C_{k+1} = \text{generateCandidates}(F_k)$ ;
6   For each  $t$  in  $R$  do
7     Increment count of candidates in  $C_{k+1}$  that are
8     contained in  $t$ ;
9   End for
10   $F_{k+1} = \text{candidates in } C_{k+1} (\text{support} \geq \text{min\_sup})$ ;
11 End for
12 Return  $U_k F_k$ ;

```

#### (2) 关联规则生成

得到所有的频繁项集后, 可以按照是否满足最小置信度从频繁项集及其真子集中提取关联规则。值得注意的是, 我们还需要进一步分析提取出的关联规则。首先使用提升度作为关联规则的第一层过滤, 主要是删除具有高可信度但实际上独立的特征关联。其次需要在真实的样本中验证关联规则, 形成第二层过滤, 主要是参考标准的 PDF 文档规范和实际解析文档的经验对关联规则进行分析, 最终从频繁项集中提取出有效的关联规则。

表 3 列出了部分生成的关联规则。使用第一层过滤可以删除一些关联规则, 如 `author_num => size`, 这是一个典型的独立特征关联, 因为 PDF 解析时往往只考虑文件中最后一个 Author 元数据字段的内容, 该内容的修改不需插入额外的字段, 所以对 `size` 的影响很小。第二层过滤可以删除如 `keywords_num => /Names/EmbeddedFiles`、`len_stream_min => /PageLabels/Nums` 等关联规则, 这是因为根据人工解析 PDF 文档的经验, 关键字数字字符统计数与嵌入文件没有统计意义上的关联, 嵌入文件与 `count_action`、`count_js` 等特征的关联性更强; `len_stream_min` 属于特殊的统计特征, 它的改动通常与 PDF 文档中流对象更为密切。

表 3 关联规则

Table 3 Association rules

<code>author_num =&gt; /Metadata/Length</code>
<code>author_num =&gt; size</code>
<code>creator_num =&gt; /Metadata/Type</code>
<code>count_acroform =&gt; /AcroForm/Fields</code>
<code>count_acroform =&gt; /AcroForm/DR/Encoding/PDFDocEncoding</code>
<code>count_action =&gt; /OpenAction/Contents/Filter</code>
<code>count_action =&gt; /OpenAction/Contents/Length</code>
<code>count_endobj =&gt; /Pages/Resources</code>
<code>count_image_total =&gt; /Pages/MediaBox</code>
<code>count_javascript =&gt; /Pages/Resources/ProcSet</code>
<code>count_javascript =&gt; /OpenAction/Resources/ProcSet</code>
<code>count_javascript =&gt; /Names/JavaScript/Names/S</code>
<code>count_page =&gt; /PageLayout</code>
<code>image_totalpx =&gt; count_image_total</code>
<code>keywords_num =&gt; /Names/EmbeddedFiles</code>
<code>keywords_len =&gt; /Outlines/Count</code>
<code>len_stream_min =&gt; /PageLabels/Nums</code>
<code>pos_acroform_avg =&gt; /AcroForm</code>
<code>producer_num =&gt; /Metadata</code>
<code>ratio_size_obj =&gt; /Pages/Resources</code>
<code>subject_num =&gt; /PageLabels/Nums/S</code>
<code>version =&gt; /Metadata/Type</code>

其他关联规则如 `count_image_total => /Pages/MediaBox`、`count_action => /OpenAction/Contents/Length`、`count_javascript => /Names/JavaScript/Names/S` 等可以有效地表示内容特征与结构路径特征存在的关联性: 文档中图片的总数量影响了页面的多媒体显示与文件嵌入, 文档中对嵌入脚本执行或脚本内容相关的统计数则与相应的解析路径密切相关。能够产生有效关联规则的主要原因在于频繁项集去

除了稀疏的结构路径特征, 通过对大量样本进行统计分析显示出了文档内容与文档解析过程之间不可分割的联系。

### 3.3 特征修改

我们采用的事务集包括从良性样本集中选择的  $N_b$  集和从恶意样本集中选择的  $N_m$  集, 根据上节的关联分析将形成两组强关联规则。两组强关联规则一方面保证了修改特征时可以实现同步修改, 这是跨特征空间的基础; 另一方面则可以形成约束规则, 计算两个集合的交集, 并用  $N_m$  减去交集, 就可以发现一些只存在于恶意样本中的关联规则。在使用基于特征空间的方法生成对抗样本时, 可以修改的特征是需要受限制的, 因为在特征空间中修改向量可能会改变 PDF 文档的结构, 并增加破坏恶意功能的的风险。使用约束规则我们就不必担心这种情况。

对抗攻击的迭代逼近操作需要由攻击算法实现, 下一小节将对此进行解释。然而, 每次迭代后新向量点的生成都需要特征修改算法的参与。为了使每次逼近的点不是错误的点, 使用约束规则对特征向量进行修改, 完成“向量-实例-向量”的二次转换, 接着进入下一个迭代过程。在算法 2 中描述了这个过程。

#### 算法 2: 特征修改

输入: 迭代生成向量  $x'_t$ , 恶意型关联规则  $M$ , 通用型关联规则  $G$ .

输出: 新向量  $x'_t$ .

```

1  $I_t$  = Iteratively generated vector set, with initial vector;
2  $S_m$  = Compare ( $x'_{t-1}$ ,  $x'_t$ ); /* View modified features */
3 For each  $t$  in  $S_m$  do
4     Find the corresponding association rule in  $G$ ;
5      $t$  remains unchanged;
6     If  $t$  not in  $G$  do
7         Find the corresponding association rule in  $M$ ;
8         Modify  $t$  to be consistent with  $x'_{t-1}$ ;
9 End for
10 Synchronously modify the content-based and structure-based features of  $x'_t$  according to  $S_m$  and form a new example  $s'_t$ ;
11  $x'_t$  = Vectorize( $s'_t$ );
12 Return  $x'_t$ ;
```

### 3.4 攻击算法

根据 2.4 节的描述, 可以将基于梯度的方法转换为求解约束优化问题, 如公式(1)。对于任何目标恶意样本  $x$ , 最佳攻击策略都会找到一个示例  $x'$  来最小化  $f$ , 同时要限制其与  $x$  的距离。为了使  $x'$  保留恶意功能, 公式中定义  $c(x, x')$  为成本函数, 它确保可以在  $L_p$  范数距离内找到  $x'$ 。也可以将公式(1)转换为公式(2)的形式, 即最大化损失函数  $J(x', y)$  以生成  $\|x' - x\|_\infty \leq \epsilon$  距离满足  $L_\infty$  范数的对抗样本。



$$x' = \operatorname{argmin}_x (f(x) + c(x, x')), \text{ s.t. } \|x' - x\|_\infty \leq \varepsilon \quad (1)$$

$$x' = x + \varepsilon \cdot \operatorname{sign}(\nabla_x J(x, y)), \text{ s.t. } \|x' - x\|_\infty \leq \varepsilon \quad (2)$$

本文使用基于改进后的动量迭代梯度的方法生成对抗样本, 动量迭代法生成的对抗样本在白盒攻击和黑盒攻击中均具有较高的成功率。该方法减轻了白盒攻击与可传递性之间的权衡, 成为一种较强大的攻击算法。通过在迭代过程中沿损失函数的梯度方向累积速度矢量来加速梯度下降。保留之前的梯度有助于快速通过较差的局部最大值或最小值, 在本文构造的攻击场景中可以明显降低特征依赖关系的影响。本文利用动量的概念, 与联合检测器场景结合来生成 PDF 对抗样本。

### 算法 3: 攻击算法

输入: 检测器  $f$  及其 loss function  $J$ , 初始向量  $x$ .

输出: 对抗样本  $x'$ .

```

1 Set the attack parameters of the method:  $\epsilon, \mu, T$ 
2  $\alpha = \epsilon / T$ ; /* The step size per iteration */
3  $g_0 = 0$ ;  $x'_0 = x$ ; /* Initialize  $g_t$  and  $x'_t$  */
4 For each  $t$  in  $T$  do
5   Input  $x'_t$  to  $f$  to obtain:
6    $g_t = \nabla_x J(x'_t, y)$ ;
7   Update  $g_{t+1}$  by accumulating the velocity vector in
8   the gradient direction:
9    $g_{t+1} = \mu \cdot g_t + g_t / \|g_t\|$ ; (3)
10  Update  $x'_{t+1}$  by:  $x'_{t+1} = x'_t + \alpha \cdot \operatorname{sign}(g_{t+1})$ ; (4)
11   $x'_{t+1}$  = Feature modification ( $x'_{t+1}, M, G$ );
12 End for
13 Return  $x' = x'_T$ ;
```

算法 3 中总结了动量迭代攻击方法, 其中在本文的实验部分中针对攻击方法进行了改进, 使其适合文档型的对抗样本生成。公式(3)中我们定义  $g_t$  收集第  $t$  次迭代之前的梯度, 每一次迭代的衰减因子为  $\mu$ ; 公式(4)中显示第  $t$  次迭代中由上一次迭代产生的向量点以  $\alpha$  为步长、 $g_t$  为方向进行更新, 直到产生成功的对抗样本。算法的第 11 行主要描述了特征修改算法的使用, 不使用特征修改算法会引起两方面的问题: 一方面是不能保证特征向量的跨特征空间和恶意功能保留。例如攻击算法可能在一次迭代中改变了特征 count\_image\_total, 特征修改算法将根据关联规则向 PDF 中添加等量的 MediaBox 片段, 使结构特征/Pages/MediaBox 同步修改, 并累积所有修改产生新的文件, 释放特征依赖, 最终由新文件产生新的向量; 不使用特征修改算法则会产生增量更新, 向 PDF 尾部加入填充了无意义字符的新文件体, 但新文件体中并不一定包含 MediaBox 片段, 导致新向量的结构特征发生未知的变化。另一方面是可能会产生特征空间中不存在或非正常的向量。例如当 size 的变化引起结构特征/Metadata/Length 的变化, ver-

sion 特征可能会产生未知的改变(负值或不存在的版本值), 这会使最终形成的对抗样本是损坏的或无法成功逃避检测器。使用特征修改算法则可以根据关联规则限制 version 特征值的改变, 从而修正迭代攻击的方向, 产生正常的向量。

### 3.5 鲁棒性提升测试

自从深度神经网络的对抗样本被发现以来, 相关文献中有一个普遍的共识, 即对抗训练可以提升神经网络对这些样本的鲁棒性, 因此大多数新型攻击都建议将对抗训练作为防御这些攻击的第一道防线。对抗训练本质上是一种数据增强的方法, 该方法的求解可以被归纳为 Min-Max 的过程, 即 Inner Maximization 和 Outer Minimization 两个步骤。Inner Maximization 用于通过最大化分类损失函数来生成对抗样本, Outer Minimization 用于使用 Inner Maximization 阶段生成的对抗本来训练模型, 使得输出结果最小化, 受 Inner Maximization 阶段的影响非常大。因此对抗训练是一种非自适应策略, 需要使用强大的攻击来执行训练。

上面章节已经研究了生成对抗样本的技术, 在实验部分将对生成的对抗样本进行对抗训练以完成鲁棒性提升测试。在对抗训练中有 3 个关键点, 如图 3 所示: 首先在构造恶意 PDF 文档检测模型的时候, 选择使用 SVM 算法, 因为它在此前的研究中是可靠有效的; 其次在训练分类器时, 使用多类型特征, 即基于内容的特征和基于结构的特征, 这样能够体现出本文生成方法具有跨特征类型的性质; 最后在训练时采取交叉验证的方法, 并将对抗样本添加到训练集中进行扩充, 使得最后模型的分平面较原始模型有轻微的偏离, 从而具有更好的鲁棒性。

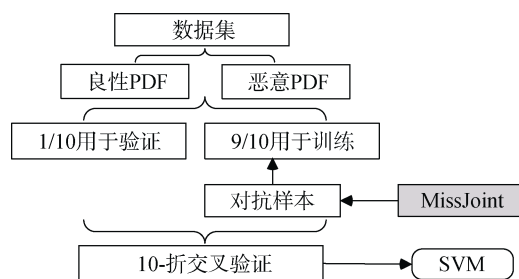


图 3 对抗训练设置

Figure 3 Setting of adversarial training

## 4 实验与分析

在本节中, 首先介绍实验中使用的数据集和相关实验设置。然后设置对抗样本生成实验和鲁棒性提升实验, 通过将本文的方法与其它两种攻击方法

进行比较评估本文方法的有效性。

#### 4.1 数据集

本文的实验采用收集了 11200 个恶意 PDF 文档和 10500 个良性 PDF 文档的总数据集。其中恶意 PDF 文档主要来自 Contagio archive<sup>[22]</sup>, Virus Share<sup>[23]</sup>和 Virus Total<sup>[24]</sup>, 以及部分自行采集的样本。良性 PDF 文档来自 Contagio archive, Google 搜索和一些研究机构, 包括公告, 操作手册等。这些数据集是当前恶意 PDF 文档检测研究的通用数据集, 具有代表性。

在对抗样本生成实验中使用了 3 个数据集: 2 个事务数据集  $N_b$  和  $N_m$  用于提取相关特征并分析关联规则, 而攻击数据集则由恶意 PDF 文档组成, 这些恶意 PDF 文档被用作生成针对 PDF 检测器的对抗样本的起点。其中从恶意数据集和良性数据集中随机抽取了 2000 个样本作为事务集, 攻击数据集中包含 500 个样本, 这是为了与用来对比的 EvadeML 方法一致。表 4 总结了攻击样本的选择过程。

表 4 数据集  
Table 4 Datasets

数据集	描述	数量
事务集	简单随机抽样	2 000
	恶意 PDF 文档总数	11 200
攻击数据集	由 Wepawet 检测到具有网络 API 调用行为的样本	9 766
	Cuckoo 观察到的有网络活动的样本	1 621
	由 pdfwr 重新正确打包的样本	1 412
	PDFRate 检测的确定恶意样本	1 366
	Hidost 检测的确定恶意样本	632
	Hidost 和 PDFRate 的交集	500

在鲁棒性提升实验中使用了 2 个数据集: 用于训练和测试的数据集  $S$  和攻击数据集  $M$ , 其中将总数据集作为  $S$ ,  $M$  则采用对抗样本生成实验中的攻击数据集。

#### 4.2 评估指标

在对抗样本生成实验中, 我们主要关注 2 个评估指标: 针对目标恶意 PDF 文档检测器攻击的对抗样本生成率和攻击成功率。其中, 对抗样本生成率指的是可以通过攻击算法生成对应对抗样本的原始恶意样本在整个攻击数据集中所占的比例。攻击成功率是指成功逃避检测器的样本在整个攻击数据集中所占的比例, 它也可以从目标检测器的准确率 (Accuracy)、召回率 (Recall) 和 F1 分数 (F1-Score) 中反映出来。

鲁棒性提升实验也将准确率、召回率和 F1 分数作为评估指标。一般将混淆矩阵用于上述指标的计

算, 如表 5 所示, TP 表示正确检测到恶意 PDF 文档的数量, FP 表明被误判为恶意的良性文档的数量, TN 表明正确检测到良性文档的数量, FN 表示被误判为良性的恶意 PDF 文档的数量。

表 5 混淆矩阵  
Table 5 Confusion matrix

	恶意(预测)	良性(预测)
恶意	TP	FN
良性	FP	TN

根据混淆矩阵可得各评价指标的计算方法:

准确率:  $\text{Accuracy} = \text{TP} / (\text{TP} + \text{FP})$

召回率:  $\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$

F1 分数:  $\text{F1-Score} = (2 \times \text{Recall} \times \text{Accuracy}) / (\text{Recall} + \text{Accuracy})$

如果一个生成的对抗样本可以实现原始的恶意功能, 就可以说它是一个成功的对抗样本。因此文本使用沙箱技术作为验证系统来确定对抗样本是否保留了恶意行为。即在装有 PDF 阅读器的虚拟沙箱中运行提交的样本, 并报告样本的行为, 包括调用的网络 API 及其参数等。在实验中, 检测器对样本的预测得分是在迭代过程中获得的; 迭代过程结束后, 被判定为良性的样本将提交到沙箱中。最后将验证系统的结果与目标检测器的预测结果进行比较, 得出最终的逃避结果。

#### 4.3 实验设置

##### (1) 实验环境

我们用来进行实验的计算机环境设置如下: 一台主机计算机(Intel Core i7-6300 CPU @ 3.40GHz, 运行 64 位 Windows 10 桌面版, 16GB 物理内存); 一台辅助计算机(Intel Core i5-7300HQ CPU @ 2.50GHz, 运行 64 位 Ubuntu 16.04 Server, 16GB 物理内存)。辅助机器主要用于部署沙箱验证系统。

##### (2) 检测器

Mimicus 攻击方法创建了一个代理 SVM 检测器来执行和验证它们的攻击, 同时 Hidost 检测器也使用 SVM 机器学习方法。因此, 在我们的实验中, 我们也将使用 SVM 的模型实现作为容器来生成对抗样本。

##### (3) 关联规则分析

我们首先探讨关联规则分析过程中最小支持度的阈值选择是否会影响生成关联规则的数量, 从而影响分析的效率。因此, 我们对一组随机选择的 2000 个样本( $N_b$  或  $N_m$ )进行了研究。我们使用 0.002 的刻度在范围为 0.2% 到 2.0% 的最小支持度下挖掘和分析



关联规则。在图 4 中展示了最终不同最小支持度下最终获得的关联规则的数量。我们可以看到由于特征之间存在明确的关系, 因此在经验范围内, 关联规则的数量不会受支持阈值的太大影响。为了消除偶然性并全面分析特征之间的关联规则, 我们在实验中使用 0.5% 的最小支持度重复了 10 次提取。

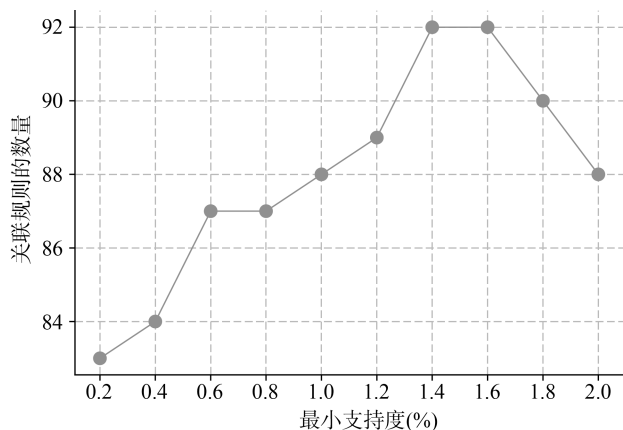


图 4 在不同最小支持度下生成对抗样本的数量

Figure 4 The number of association rules obtained at different minimum support

#### (4) 攻击参数

接下来我们需要研究攻击算法相关参数的设置效果。衰减因子  $\mu$  是提高攻击成功率的首要考虑因素, 因为它决定了梯度下降的速度, 因而我们对衰减因子的值进行测试。我们通过我们的攻击算法生成对抗样本, 其中设置距离  $\varepsilon = 25$ , 迭代次数为 15, 衰减因子范围为 0.0~2.0, 刻度为 0.1。图 5 显示了针对 PDFRate 和 Hidost 生成对抗样本的成功率。与针对图像数据集进行的实验相似, 其中攻击 Hidost 的成功率是单峰曲线, 其最大值在  $\mu = 1.0$  左右获

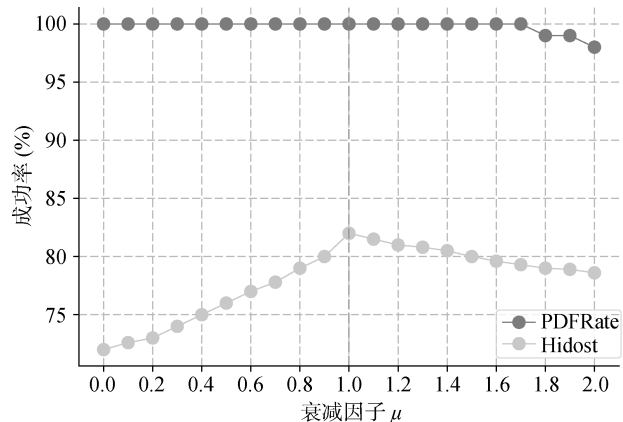


图 5  $\mu$  从 0.0 到 2.0 下对 PDFRate 和 Hidost 的攻击成功率

Figure 5 The success rates against PDFRate and Hidost with  $\mu$  from 0.0 to 2.0

得。这意味着积累过多或过少的之前的梯度都会使速度不稳定。

然后, 我们测试攻击两个检测器时距离  $\varepsilon$  的大小对成功率的影响。我们将距离的大小限制到 50 以下, 因为当距离太大时, 它将增加复杂性或导致特征空间中的位置跳跃。我们在实验中设置不同的距离进行攻击, 并在图 6 中给出了测试结果。提高攻击 PDFRate 的成功率所需的距离 ( $\varepsilon = 15$ ) 比 Hidost ( $\varepsilon = 20$ ) 更短。这是因为当距离不是很大时, 关联规则并不会被频繁使用。当距离  $\varepsilon = 30$  时, 两个检测器的攻击成功率均达到峰值。之后曲线趋于稳定, 原因是迭代过程中对特征的修改已基本完成。

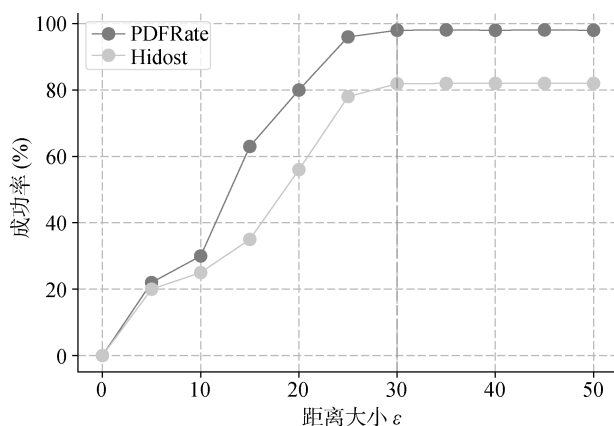


图 6 不同距离下对 PDFRate 和 Hidost 的攻击成功率

Figure 6 The success rates against PDFRate and Hidost with different distance

最后, 对于迭代次数对成功率的影响, 我们保持其他参数一致, 使迭代次数从 5 开始递增, 研究了攻击两个检测器的成功率, 并在图 7 中给出测试结果。可以观察到迭代次数与距离参数相似, 即攻击的成功率随着迭代次数的增加而增加。当迭代次数为

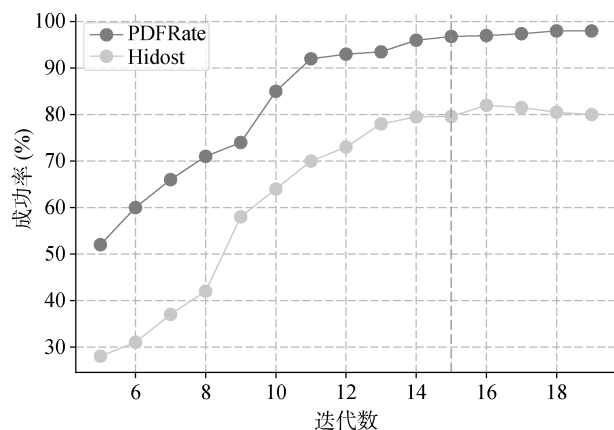


图 7 不同迭代数下对 PDFRate 和 Hidost 的攻击成功率

Figure 7 The success rates against PDFRate and Hidost with different iterations

15 时, 成功率达到峰值并保持稳定。该结果可以指导我们设置黑盒攻击的迭代次数, 从而提高攻击效率。

#### 4.4 结果分析

根据上一节中的实验设置测试, 我们的方法将使用以下参数: 最大距离  $\epsilon$  设置为 30, 迭代次数为 15, 衰减因子  $\mu$  为 1.0。我们将使用了关联规则的动量攻击方法叫作 MissJoint, 而未使用则为 MissJoint-norule; 同时, 用于攻击 PDFRate 的 EvadeML 被定义为 EvadeML-P, 而用于攻击 Hidost 的 EvadeML 被定义为 EvadeML-H。接下来在对抗样本生成实验中分别研究它们的攻击效果并进行比较。

##### (1) MissJoint 和 MissJoint-norule 攻击效果

从图 8 可以看出, 无论是 MissJoint 还是 MissJoint-norule, 针对 PDFRate 检测器生成的保留了恶意功能的对抗样本都很多, 成功率分别为 98% 和 96%。但是 MissJoint 在 Hidost 检测器上的表现要好于 MissJoint-norule: MissJoint 的成功率为 82%, 而 MissJoint-norule 的成功率仅为 36%。这表明尽管基于动量迭代梯度的方法在计算机视觉方面攻击黑盒系统时提供了良好的模型可移植性, 但在恶意软件领域, 特征空间的复杂性仍然受到限制。为了降低这种复杂性, 我们尝试进行特征关联, MissJoint 攻击的初步结果显示有效。

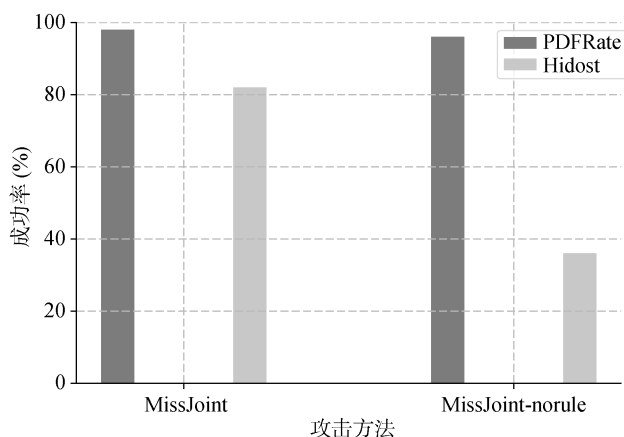


图 8 MissJoint 和 MissJoint-norule 的攻击成功率

Figure 8 The success rates of MissJoint and MissJoint-norule

##### (2) 对抗样本生成率

为了进一步验证 MissJoint 方法的有效性, 我们在包含 500 个恶意样本的同一数据集上实施了 MissJoint, Mimicus, EvadeML-P 和 EvadeML-H 攻击。表 6 中汇总了各自的对抗样本生成率, 可以看出这 4 种方法都具有较高的对抗样本生成率, 但 MissJoint

和 EvadeML-H 没有达到 100%。我们判断 MissJoint 是由于迭代过程中多次释放特征依赖关系导致最终没有得到对抗样本, EvadeML-H 则是因为 Hidost 的自我更新删除了许多鲁棒性较差的特征。

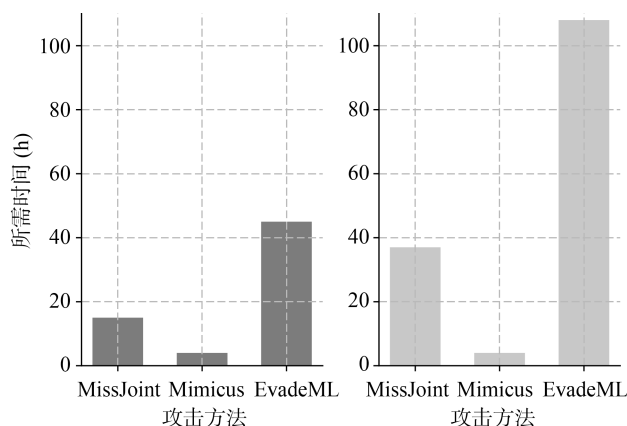


图 9 3 种攻击方法的执行效率

Figure 9 Execution efficiency of three methods

表 6 4 种攻击方法的对抗样本生成率

Table 6 The generation rates of four attack methods

攻击方法	对抗样本数量	对抗样本生成率(%)
MissJoint	498	99.6
Mimicus	500	100
EvadeML-P	500	100
EvadeML-H	496	99.2

##### (3) 对抗样本攻击成功率

实验还在单个检测器和目标联合检测器上观察了对抗样本的性能。其中每个检测器使用的训练集大小和比例略有不同, 因为我们需要根据其他作者提出的参数对其进行优化。但是我们选择了包含 1000 个恶意样本和 1000 个良性样本的相同测试集。将恶意样本替换为相同数量的对抗样本, 然后输出各检测器的准确性, 召回率和 F1 分数。结果记录在表 7 中。

首先可以观察到各个检测器的准确率稍有波动, 这是因为对抗样本对检测器的误报率影响较小, 这也进一步证明对抗攻击可以在不影响其他样本的情况下逃避检测器。对于联合探测器来说, 由于误报率在多个探测器之间进行了积累, 因此准确度会略有下降。其次从表的第 3 行和第 4 行可以明显看出, 由于 Mimicus 和 EvadeML-P 在攻击过程中没有特定的关联规则可遵循, 生成的对抗样本很难成功逃避 Hidost 检测器和联合检测器, 它们仍然保持较高的召回率和 F1 分数。此外 EvadeML 攻击方法的作者进行过交叉逃避实验, 使用有针对性的对抗样本攻

击非目标检测器, 他们的结果表明, 针对 Hidost 检测器产生的对抗样本在攻击 PDFRate 检测器时成功率可达 77.6%; 而当条件相反时, 针对 PDFRate 检测器产生的对抗样本攻击 Hidost 检测器则几乎无效。这与我们在对比实验中的实验结果保持一致, 从表中的第 5 行可以观察到 Evademl-H 不仅会降低 Hidost 检测器的性能, 对其他 3 个检测器的性能也有一定程度的影响。最后从表的最后一行可以看到, Miss-Joint 方法大大降低了所有检测器的性能。我们认为主要原因是方法中的特征关联分析增加了攻击范围, Evademl-H 方法的交叉实验能达到 77.6% 则是随机的,

因为它的特征修改是不确定的。同时可以发现联合检测器相对于 MissJoint 依然有 0.552 的召回率, 除去测试集中 500 个恶意样本, 还有数十个对抗样本被联合检测器标记为恶意, 这与我们在对比 MissJoint 和 MissJoint-norule 攻击效果的实验中结果是一致的, 原因在于不同检测器使用的解析工具和特征提取工具有差别, 当恶意样本中使用了未混淆的脚本代码或触发 API 时, 即使对其特征进行修改产生了相应的对抗样本, 保留下来的恶意功能仍然是未被混淆的, 容易被不同的解析工具发现从而给出恶意的预测结果。

表 7 对抗样本攻击成功率

Table 7 The results on detectors against adversarial examples

		PJScan	PDFRate	Slayer	Hidost	Joint
Original	准确率	0.886	0.977	0.985	0.991	0.904
	召回率	0.824	0.96	0.972	0.992	0.998
	F1-分数	0.854	0.968	0.978	0.991	0.949
Mimicus	准确率	0.821	0.955	0.971	0.991	0.902
	召回率	0.485	0.49	0.502	0.959	0.971
	F1-分数	0.61	0.648	0.662	0.975	0.935
EvadeML-P	准确率	0.817	0.954	0.971	0.991	0.9
	召回率	0.473	0.482	0.497	0.946	0.95
	F1-分数	0.6	0.64	0.657	0.968	0.924
EvadeML-H	准确率	0.828	0.962	0.976	0.983	0.884
	召回率	0.512	0.58	0.612	0.522	0.81
	F1-分数	0.633	0.724	0.752	0.682	0.845
MissJoint	准确率	0.821	0.955	0.971	0.983	0.839
	召回率	0.486	0.492	0.506	0.535	0.552
	F1-分数	0.611	0.649	0.665	0.693	0.666

#### (4) 攻击方法执行效率

最后, 我们在两个先进的检测器上比较了这 3 种攻击方法的时间复杂度, 如图 9 所示。Mimicus 使用的时间最少, 因为它在迭代过程中操作最少。由于使用遗传算法, EvadeML 花费的时间最长, 它需要大量的计算资源才可以减少时间。我们的方法无需花费太多时间和资源即可保持良好的性能。

#### (5) 鲁棒性提升

本文针对对抗样本的攻击强度进行了鲁棒性提升实验, 由上述攻击方法生成相应的对抗样本, 根据 3.5 提出的鲁棒性提升测试方法, 分别进行对抗训练, 然后测试剩下类型的对抗样本对训练后模型的影响, 实验结果如表 8 所示。

可以看出, 由于 Mimicus 和 EvadeML-P 产生的对抗样本主要基于内容特征空间, 因此使用其进行对抗训练后, 互相产生的抑制作用较强: F1 分数达到

了 0.692 和 0.724, 而这两种方法对 EvadeML-H 和 MissJoint 产生的对抗样本依然无法很好地检测, 使得训练后的检测器性能偏低。EvadeML-H 产生的对抗样本则基于结构特征空间, 将其用于对抗训练后产生的检测器拥有较高的性能, 可以有效抵御 Mimicus 和 EvadeML-P 产生的对抗样本, F1 分数达到了 0.907 和 0.898; 对于由 MissJoint 产生的对抗样本, 也有了性能提升。MissJoint 产生的对抗样本基于多种特征类型的关联, 由其进行对抗训练后产生的检测器鲁棒性得到很大的提升, 不论对于 Mimicus 和 EvadeML-P 攻击, 还是 EvadeML-H 攻击, 都保持很高的性能。分析可知, 当不同类型的特征同时得到修改时, 产生的对抗样本经过对抗训练后可以很好地平衡模型的决策平面, 从而对使用单个特征类型产生的对抗样本形成抑制作用, 提升检测器的鲁棒性。

表 8 鲁棒性提升测试结果

Table 8 Robustness improvement test results			
	准确率	召回率	F1-分数
Mimicus 产生对抗样本进行训练			
EvadeML-P	0.843	0.587	0.692
EvadeML-H	0.824	0.512	0.632
MissJoint	0.822	0.493	0.616
EvadeML-P 产生对抗样本进行训练			
Mimicus	0.885	0.613	0.724
EvadeML-H	0.845	0.589	0.694
MissJoint	0.83	0.504	0.627
EvadeML-H 产生对抗样本进行训练			
Mimicus	0.921	0.893	0.907
EvadeML-P	0.912	0.884	0.898
MissJoint	0.859	0.792	0.824
MissJoint 产生对抗样本进行训练			
Mimicus	0.984	0.952	0.968
EvadeML-P	0.965	0.937	0.951
EvadeML-H	0.958	0.924	0.941

## 5 结论

本文提出了针对恶意 PDF 文档分类的对抗样本生成方法 MissJoint, 并基于该方法实现了对抗样本生成模型, 该模型对抗基于多检测引擎的恶意 PDF 文档检测系统。实验结果表明, 本文提出方法保持了较高的样本生成率和攻击成功率, 且生成的对抗样本保留原始的恶意行为, 使用生成的对抗样本进行对抗训练可以有效提升检测器的鲁棒性。本文提出的方法和模型也同样适用于其他恶意软件检测工具, 如恶意安卓应用检测。

下一步工作准备将对抗样本生成和检测器的鲁棒性提升结合起来, 通过检测发现未知的对抗样本, 利用对抗训练完成系统的自升级, 提升检测系统的鲁棒性。

## 参考文献

- [1] 全球高级持续性威胁(APT)2019 年研究报告, Shenzhen Tencent Computer System Co., Ltd., <https://s.tencent.com/research/report/902.html>, Jun. 2018.
- [2] Šrndić N, Laskov P. Detection of Malicious PDF Files Based on Hierarchical Document Structure[J]. *Proceedings of the 20th Annual Network & Distributed Systems Symposium*, 2013: 1-16.
- [3] Szegedy C, Zaremba W, Sutskever I, et al. Intriguing Properties of Neural Networks[EB/OL]. 2013: arXiv: 1312.6199. <https://arxiv.org/abs/1312.6199>.
- [4] Biggio B, Roli F. Wild Patterns: Ten Years after the Rise of Adversarial Machine Learning[J]. *Pattern Recognition*, 2018, 84:

317-331.

- [5] Goodfellow I J, Shlens J, Szegedy C. Explaining and Harnessing Adversarial Examples[EB/OL]. 2014: arXiv: 1412.6572. <https://arxiv.org/abs/1412.6572>.
- [6] Papernot N, McDaniel P, Jha S, et al. The limitations of deep learning in adversarial settings[C]. *2016 IEEE European Symposium on Security and Privacy*, 2016: 372-387.
- [7] Papernot N, McDaniel P, Goodfellow I, et al. Practical Black-Box Attacks Against Machine Learning[C]. *The 2017 ACM on Asia Conference on Computer and Communications Security*, 2017: 506-519.
- [8] Madry A, Makelov A, Schmidt L, et al. Towards Deep Learning Models Resistant to Adversarial Attacks[EB/OL]. 2017: arXiv: 1706.06083. <https://arxiv.org/abs/1706.06083>.
- [9] Biggio B, Corona I, Maiorca D, et al. Evasion Attacks Against Machine Learning at Test Time[C]. *The 2013th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part III*, 2013: 387-402.
- [10] Šrndić N, Laskov P, Components C. Practical evasion of a learning-based classifier: A case study[C]. *2014 IEEE Symposium on Security and Privacy*, 2014: 197-211.
- [11] Xu W, Qi Y, Evans D. Automatically evading classifiers[C]. *Proceedings of the 2016 network and distributed systems symposium*, 2016, 10.
- [12] Forrest S. Genetic Algorithms: Principles of Natural Selection Applied to Computation[J]. *Science*, 1993, 261(5123): 872-878.
- [13] Dang H, Huang Y, Chang E C. Evading Classifiers by Morphing in the Dark[C]. *The 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017: 119-133.
- [14] Agrawal R, Srikant R. Fast algorithms for mining association rules[C]. *Proc. 20th int. conf. very large data bases, VLDB*, 1994, 1215: 487-499.
- [15] Dong Y P, Liao F Z, Pang T Y, et al. Boosting adversarial attacks with momentum[C]. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018: 9185-9193.
- [16] PDF Reference and Adobe Extensions to the PDF Specification, Adobe. Inc, [https://www.adobe.com/devnet/pdf/pdf\\_referenre\\_archive.html](https://www.adobe.com/devnet/pdf/pdf_referenre_archive.html), 2008.
- [17] Maiorca D, Biggio B, Giacinto G. Towards Adversarial Malware Detection: Lessons Learned from PDF-Based Attacks[J]. *ACM Computing Surveys*, 2020, 52(4): 78.
- [18] Laskov P, Šrndić N. Static Detection of Malicious JavaScript-Bearing PDF Documents[C]. *The 27th Annual Computer Security Applications Conference*, 2011: 373-382.
- [19] Smutz C, Stavrou A. Malicious PDF Detection Using Metadata and Structural Features[C]. *The 28th Annual Computer Security Applications Conference*, 2012: 239-248.
- [20] Maiorca D, Ariu D, Corona I, et al. A structural and content-based approach for a precise and robust detection of malicious PDF files[C]. *2015 International Conference on Information Systems Security and Privacy*, 2016: 27-36.
- [21] Šrndić N, Laskov P. Hidost: A Static Machine-Learning-Based Detector of Malicious Files[J]. *EURASIP Journal on Information Security*, 2016, 2016(1): 1-20.

- [22] Malicious Documents Archive for Signature Testing and Research -Contagio Malware Dump, S. Chenette, <http://contagiodump.blogspot.de/2010/08/malicious-documents-archive-for.html>, 2009.

- [23] Virus Share, J. M. Roberts, <https://virusshare.com>, 2011.

- [24] Virustotal-free online virus, malware and url scanner, V. Total, <https://www.virustotal.com>, 2012.



**刘超** 现任中国科学院信息工程研究所正研级高级工程师, 研究领域为网络空间安全。Email: liuchao@iie.ac.cn



**娄尘哲** 于 2017 年在宁夏大学计算机科学与技术专业获得学士学位。现在中国科学院信息工程研究所网络空间安全专业攻读硕士学位。研究领域为恶意软件检测。研究兴趣包括: 恶意文档检测、对抗性机器学习等。Email: louchenzhe@iie.ac.cn



**喻民** 现任中国科学院信息工程研究所高级工程师, 研究领域为恶意代码分析。Email: yumin@iie.ac.cn



**姜建国** 现任中国科学院信息工程研究所研究员, 研究领域为网络安全与保密技术。Email: jiangjianguo@iie.ac.cn



**黄伟庆** 现任中国科学院信息工程研究所正研级高级工程师, 研究领域为网络空间安全。Email: huangweiqing@iie.ac.cn