 **JIANGXINORMALUNIVERSITY**

本科生毕业设计（论文）

**基于决策树的钓鱼网站辨别系统**

**Phishing Website Identification**

**System Based on Decision Tree**

**学院名称： 计算机信息工程学院**

**专业名称： 计算机科学与技术**

**姓 名： 陈茂林**

**学 号： 201826303019**

**指导教师： 李云洪**

**完成日期： 2022.4.12**

## 摘要

网络带给人们便利的同时也催生着网络犯罪，网络诈骗和犯罪案件层出不穷，网络安全在互联网中的作用越来越突出；另一方面各类机器学习算法逐步应用于各种分类、识别问题中，并取得了良好的效果。本系统将采用**决策树**对于钓鱼网站进行识别。

首先，项目中采用的数据来自于ICU Machine Learning Repository，数据集中有充足的样本和特征，除了辨别钓鱼网站的必要的特征，还加入了辅助特征，这充分保证的系统所生成的**决策树**的可靠性。

其次，系统中采用了多种特征选择算法，包括**ID3**、**C4.5**、**CART**算法，对不同特点的数据集，可以采用与之相适应的算法生成决策树；例如如果输入系统的数据其特征具有多种取值时，可采用**C4.5算法**避免**ID3算法**对多取值特征的偏好。为了使决策树更加平衡，有更好的泛化性，本系统还增加了剪枝算法，**剪枝算法**与特征选择算法的组合极大达到提高了系统的可靠性。

最后，在构建**决策树**之后，为便于分析**其**的结构，还需绘制**决策树**，使其可视化。在各种组合中，**ID3算法**搭配后**剪枝算法**构建的决策树在测试集的预测准确率最高为95%左右。

关键字：ID3 C4.5 CART 剪枝 决策树

摘要 2

第1章 绪论 4

1.1 钓鱼网站辨别系统的开发背景 4

1.2 本设计的主要任务 5

第2章 数据及特征 5

2.1数据集介绍 6

2.2数据特征 6

2.2.1 URL特征 6

2.2.2 异常特征 7

2.2.3 HTML和JavaScript特征 7

2.2.4 域名特征 8

第3章 决策树分类系统 8

3.1 ID3算法 8

3.1.1信息增益 8

3.1.2 ID3算法构建决策树的过程 9

3.2 C4.5算法 10

3.2.1 信息增益比 10

3.2.2用C4.5算法构建决策树 10

3.3 CART算法 11

3.3.1 基尼指数 11

3.3.2 CART算法构建决策树 11

3.4 剪枝算法 12

第4章 系统的应用与预测结果分析 12

4.1钓鱼网站辨别系统的构建 12

4.1.1 决策树的表示方式 12

4.1.2 决策树的构建 13

4.2钓鱼网站辨别系统的绘制 13

4.3系统的预测准确率分析 14

4.3.1 剪枝算法及特征选择算法分析 15

4.3.2 决策树分析 16

第5章 总结和展望 17

5.1总结 17

5.1.2系统的优点 17

5.1.2系统的缺点 17

5.2 展望 18

参考文献 19

致谢 20

## 绪论

### 1.1 钓鱼网站辨别系统的开发背景

在互联网的发展过程中，http协议对于构建互联的网络世界至关重要，它提供的富文本使得网络在全世界范围内普及开来，其中的统一资源定位符（URL）使得人们能快捷的访问网络资源。然而，随着网络的进一步发展，与之有关的网络犯罪活动也更加频发，维护网络安全是互联网正常发展的重要课题。

在众多网络犯罪中，钓鱼网站是不易察觉的一种。钓鱼网站会伪造与目标网站相似的网站，使得访问者误以为该网站是正常网站，向它提交个人信息，例如账户、密码、验证码等，信息的暴露将导致访问者的账户陷入不安全的境地。钓鱼网站的攻击方式多样，例如网络钓鱼、偷渡式下载、垃圾邮件等[1]。

为应对钓鱼网站，各种检测技术不断发展，目前的检测方法主要有基于黑名单的过滤法、基于机器学习的分类算法。

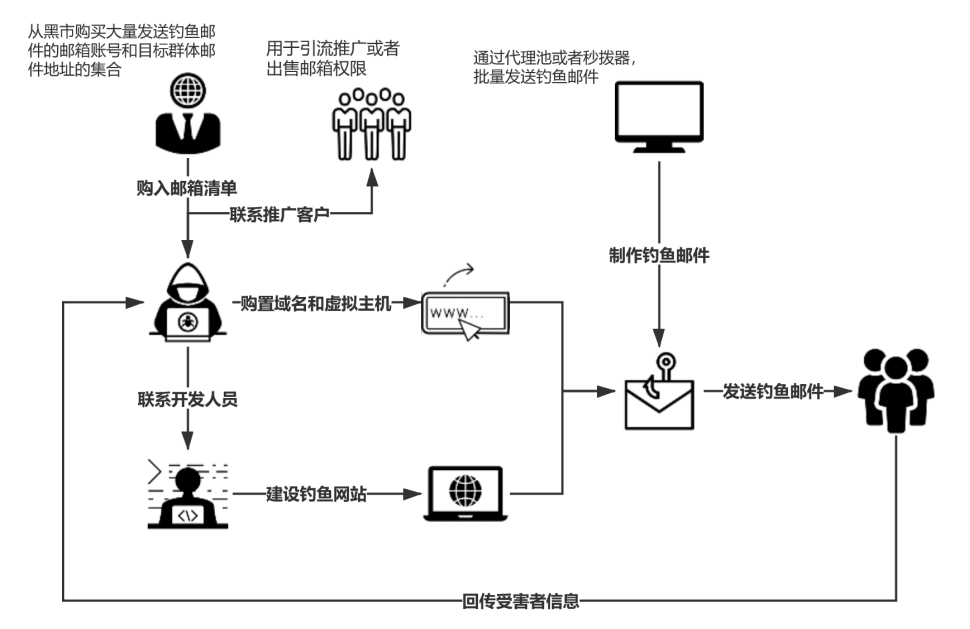
其中黑名单过滤算法将目标链接与黑名单中记录的链接进行比对，从而对目标链接进行判断，但由于网络中的钓鱼网站生成更新速度快，故该方法对于新近出现钓鱼网站辨别能力不高。本研究将采用机器学习算法，根据可靠数据集中提供的特征数据构建决策树，进而对目标链接进行辨别。

图1-1 钓鱼网站攻击流程和产业链

### 1.2 本设计的主要任务

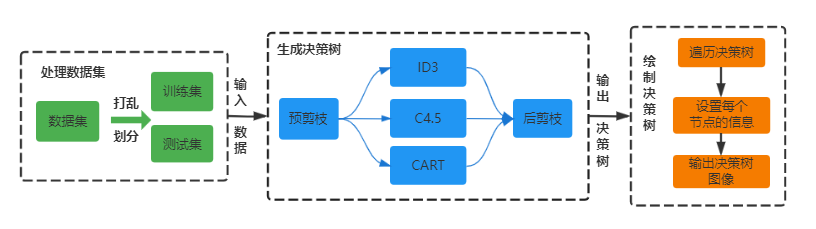
本设计的目的是综合采用决策树算法，来辨识钓鱼网站，本设计的任务包含三个部分：数据的采集和处理、 设计不同的决策树算法构建决策树、根据生成的决策树绘制具体图像。

图1-2 项目实现流程图

数据集的处理部分要解决的问题是收集有效的数据作为系统的输入，本系统所采集的数据格式为csv格式的字符串，在输入到系统中时为便于处理使用列表进行存储。特征名称也采用列表进行存储。Python的内置列表类型可以进行丰富的操作，便于数据的处理。

在生成决策树部分需要解决的问题有：如何存储决策树、采用哪些算法来实现决策树。本系统采用嵌套字典的方式存储决策树，无论是节点还是边，都采用字典进行存储，这种方式的好处是字典的结构简单易于调试。

在项目完成前不能确定各种算法对于钓鱼网站识别的效果，所以项目中同时实现了ID3、C4.5、CART、以及剪枝算法，最后运用于数据，根据效果来判断适合本项目的算法。

最后为便于分析生成的决策树，本项目实现了决策树绘制功能。通过递归的遍历决策树的节点和边，再采用graphiz进行绘制。为了区分各个节点，每个节点都会带上一个节点编号；对于中间节点，记录了其各个取值的样本数量，也记录了每个节点对应的特征名称；对于叶子节点，则记录了其对应的分类。树中每一条从根节点到叶节点的路径代表一种决策过程。

## 第2章 数据及特征

可靠的数据集是得到可靠研究结果的前提，非法的钓鱼网址可以从PhishTank、OpenPhish等网站中寻找，而合法的网址可以从Alexa、CommonCrawl等网站中获取。本研究中的决策树需要根据URL的相关特征进行构建，由于从上述网站中得到的是原始URL，其所含有的特征需要手动提取，所需的工作量大，不利于研究的推进。经查阅了解到ICU Machine Learning Repository中提供了合法网站与钓鱼网站的特征数据，且足够可靠，故本研究中将采用该数据集构建决策树。

### 2.1数据集介绍

本数据集在原始URL的基础上提取了30个特征，每个特征均有两到三个离散的取值，数据集中包含了11055个样本，其中2/3的样本将作为训练集，剩下的1/3作为测试集，以保证构建决策树过程的可靠性。本系统完成的是二分类任务，而数据集中钓鱼网站与合法网站的比例接近1：1，这能确保预测结果不会偏向某一分类。

数据集中的30个特征可分为不同的四个类别，分别为URL特征、异常特征、HTML和JS特征、以及域名特征。其中URL特征提取自URL本身，包括了URL的基本特征，而其余的多个特征则是通过URL所指向的网页和域名进行提取的。

### 2.2数据特征

#### 2.2.1 URL特征

这类特征提取自URL本身，当URL中包含IP地址、重定位符 “\\”、‘@’符号时，则这个URL很可能指向一个钓鱼网站，这是钓鱼网站地址的常见特征。有时候为了便于隐藏这些特征，钓鱼网站URL会使用重定位链接和长地址，使得用户不易发现这些特征。

此外，钓鱼网站一般创建时间短，攻击者不会特意的设计网页图标，而且其域名注册时间也往往少于一年，有时候为了便于攻击，钓鱼网站使用的端口也可能不是熟知端口。通过URL中包含的这些特征，我们简单的判别出一个URL是否指向钓鱼网站。

以下是这部分特征的符号和描述信息：

表2-1 URL特征及其描述信息

|  |  |  |
| --- | --- | --- |
| 特征分类 | 特征名称 | 特征描述 |
| URL特征 | having\_IP\_Address | URL中具有IP地址 |
| URL\_Length | URL的长度 |
| Shortining\_Service | 是否采用重定向缩短长度 |
| having\_At\_Symbol | 是否包含@符号 |
| double\_slash\_redirecting | "//"所在 的位置是否大于7 |
| Prefix\_Suffix | 是否采用了"-"分隔域名 |
| having\_Sub\_Domain | 是否具有子域名 |
| SSLfinal\_State | HTTPS的发布者是否可信以及证书年限 |
| Domain\_registeration\_length | 域名的注册时长 |
| Favicon | 网站图标是否加载自外部域名 |
| port | 是否使用熟知端口 |
| HTTPS\_token | 在域名部分是否存在HTTPS标记 |

#### 2.2.2 异常特征

异常特征是指网页的异常行为，例如钓鱼网站相对于一般网站的<a>标签占比更高；钓鱼网站中的链接在请求服务时可能转向空页面；在提交信息的时候钓鱼网站可能时通过邮件收集数据的。

这类特征的特征名称和描述信息如下表：

表2-2 异常特征及其描述信息

|  |  |  |
| --- | --- | --- |
| 特征分类 | 特征名称 | 特征描述 |
| 异常特征 | Request\_URL | 网页中包含的外部请求占比 |
| URL\_of\_Anchor | <a>标签的占比 |
| Links\_in\_tags | Meta,Script,Link标签的占比 |
| SFH | 请求服务的域名 |
| Submitting\_to\_email | 是否使用mail或mailto提交用户信息 |
| Abnormal\_URL | 在URL中是否含有主机名 |

#### 2.2.3 HTML和JavaScript特征

HTML和JavaScript特征提取自网页的源码。攻击者在假冒一个网站时，最简单的方式就是将对一个网站进行“复制”，通过使用Iframe可以快速的将原网站中的页面复制到钓鱼网站中，这将导致大量页面需要进行重定位。有些时候，攻击者可能会采用弹窗方式让用户提交信息，这种方式在是现实不需要过多的改变网页源码。为使钓鱼网站的这些改动不易被用户发现，攻击者有时会禁用右键单击功能。这类特征的符号及描述信息如下表：

表2-3 HEML和JS特征及其描述信息

|  |  |  |
| --- | --- | --- |
| 特征分类 | 特征名称 | 特征描述 |
| HTM和  JS特征 | Redirect | 重定向页面的次数 |
| on\_mouseover | onMouseOver是否改变状态栏 |
| RightClick | 右击是否被禁用 |
| popUpWidnow | 通过弹窗提交文本信息 |
| Iframe | 是否使用Iframe |

#### 2.2.4 域名特征

域名特征提取自URL所指向的网站，钓鱼网站的生命周期短，使得其域名年龄一般不会超过六个月，这样的短期链接使用的用户也比较少，所以网站访问量排名很低的网站有可能是钓鱼网站。

PageRank用于指示一个网站的重要程度，大部分钓鱼网站没有PageRank，少部分可以达到0.2。此外，如果URL位于钓鱼网站黑名单中，那这个URL一定是钓鱼网站，否则不一定为钓鱼网站。

这类特征的符号名称和描述信息见下表：

表2-4 域名特征及其描述信息

|  |  |  |
| --- | --- | --- |
| 特征分类 | 特征名称 | 特征描述 |
| 域名特征 | age\_of\_domain | 域名的年龄是否超过6个月 |
| DNSRecord | 在DNS中是否有域名记录 |
| web\_traffic | 网站的访问量排名 |
| Page\_Rank | PageRank是否大于0.2 |
| Google\_Index | 网页是否被Google编入索引 |
| Links\_pointing\_to\_page | 指向网页的链接数量 |
| Statistical\_report | 主机是否属于Top钓鱼域名或钓鱼IP |

## 第3章 决策树分类系统

决策树是一棵多叉树，树中的节点代表一个特征，该节点的分支代表该特征的一个决策选择，没有分支的叶节点代表决策结果。对于每一节点的不同分支，需要通过特征选择算法来确定其子节点的特征，特征选择的方式有信息增益、信息增益比、基尼指数，分别对应三种不同的特征选择算法：ID3、C4.5、CART算法。

### 3.1 ID3算法

#### 3.1.1信息增益

通过数据集样本计算得到的熵称为经验熵，数据集的不确定性可以通过经验熵进行度量，经验熵越大，则数据集的样本纯度越低。将数据集根据特征的取值进行分类，特征取值为的样本频率为，则特征的经验熵定义如下：

|  |  |
| --- | --- |
|  | (1) |

数据集的经验熵则通过数据集的分类结果进行计算，假设数据集的分类结果有种，其中分类结果为的频率即为，则数据集的经验熵计算公式如下：

|  |  |
| --- | --- |
|  | (2) |

经验条件熵是指在特征条件下数据集分类结果的经验熵，特征的前提下数据集分类结果的经验条件熵定义如下：

|  |  |
| --- | --- |
|  | (3) |

信息增益用于度量特征对数据集的信息不确定性的减少程度，公式如下：

|  |  |
| --- | --- |
|  | (4) |

信息增益越大，则表示特征对数据集的分类效果越好。在决策树的生成过程中，将选取对数据集信息增益最大的特征作为子节点的特征选择结果。

#### 3.1.2 ID3算法构建决策树的过程

Step1：假设当前训练集有个不同的特征，第个特征有个不同的取值，根据的取值将数据集划分为不同的子集，对于子集；

Step2：按照公式(4)计算该子集的信息增益；

Step3：取对应的特征作为当前节点的特征；

Step4：假设特征有种不同取值，每个取值对应一个分支节点。数据集中特征取值为的样本组成一个子集，其中；

Step5：若子集中为空集，则取消该分支节点；

Step6：若子集中的所有样本属于同一分类，则将该节点作为叶子节点，其分类结果为；

Step7：否则，将该节点作为中间节点，按照Step2~Step3步骤计算出该节点的特征。

Step8：重复Step1~Step7，直到构建出完整的决策树。

### 3.2 C4.5算法

#### 3.2.1 信息增益比

一个特征的取值个数越多，其信息增益也往往越大，这样得到的决策树往往不具有很好的泛化效果。为了使特征选择算法达到更佳的效果，通过信息增益率作为特征选择的方式，信息增益比的计算如下：

|  |  |
| --- | --- |
|  | (5) |

其中称为训练集关于特征的经验熵，计算方式如下：

|  |  |
| --- | --- |
|  | (6) |

其中表示的是数据集中特征取第个值的子数据集。当特征的取值越多时，数据集对特征的经验熵也会也会越大，从而避免在使用信息增益进行特征选择时容易选择特征取值多的特征。

|  |  |
| --- | --- |
|  | (7) |

#### 3.2.2用C4.5算法构建决策树

Step1：假设当前训练集有个不同的特征，第个特征有个不同的取值，根据的取值将数据集划分为不同的子集，对于子集；

Step2：按照公式(4)计算该子集的信息增益率；

Step3：取对应的特征作为当前节点的特征；

Step4：假设特征有种不同取值，每个取值对应一个分支节点。数据集中特征取值为的样本组成一个子集，其中；

Step5：若子集中为空集，则取消该分支节点；

Step6：若子集中的所有样本属于同一分类，则将该节点作为叶子节点，其分类结果为；

Step7：否则，将该节点作为中间节点，按照Step2~Step3步骤计算出该节点的特征。

Step8：重复Step1~Step7，直到构建出完整的决策树。

### 3.3 CART算法

#### 3.3.1 基尼指数

假设数据集有种分类结果，其中样本被归为第类样本数量为，数据集中的样本数量为，则该数据集的基尼指数计算方式为：

|  |  |
| --- | --- |
|  | (8) |

根据特征的不同取值将数据集划分为多个子集，假设取值为时对应的子集为，该子集的样本数量为，则特征条件下集合的基尼指数为：

|  |  |
| --- | --- |
|  | (9) |

特征条件下的基尼指数可以衡量样本集合的纯度，越小，表示通过特征对数据集进行分类的效果越好。

#### 3.3.2 CART算法构建决策树

Step1：假设当前训练集有个不同的特征，第个特征有个不同的取值，根据的取值将数据集划分为不同的子集，对于子集；

Step2：计算该子集的基尼指数；进而计算出特征条件下数据集的基尼指数；

Step3：取对应的特征作为当前节点的特征；

Step4：假设特征有种不同取值，每个取值对应一个分支节点。数据集中特征取值为的样本组成一个子集，其中；

Step5：若子集中为空集，则取消该分支节点；

Step6：若子集中的所有样本属于同一分类，则将该节点作为叶子节点，其分类结果为；

Step7：否则，将该节点作为中间节点，按照Step2~Step3步骤计算出该节点的特征。

Step8：重复Step1~Step7，直到构建出完整的决策树。

### 3.4 剪枝算法

在未剪枝的决策树中，可能出现过渡拟合的情况，例如当某一个节点的样本集中只有一个样本时，由于未进行剪枝，该节点下可能会形成一棵很长的单分支子树，而这颗子树的分类结果只有一个，在这种情况下，通过剪枝将该节点改为叶子节点，能减少决策树的长度，提高决策树的搜索效率。根据剪枝过程在决策树生成过程的不同阶段，可以将剪枝分为预剪枝和后剪枝。

多数投票法是指在一个数据集中有多种分类结果，如果每个分类在数据集中的样本数最多，则选择该分类作为数据集的分类结果。

预剪枝算法是指在节点生成前：对于某一个节点，①如果不进行划分，则令当前节点为叶节点，并在训练集上对当前节点采用多数投票法，得到当前节点的分类结果，再计算出不进行划分时在测试集预测的准确率。②如果进行划分，则根据当前节点的不同取值将训练集划分为多个子集，每个子集对应一个叶子节点，在每个子集上分别采用多数投票法得到对应叶子节点的分类结果，最后计算出进行划分后在测试集上的准确率。对比划分前后的准确率，来决定在节点生成前是否进行预剪枝。

后剪枝则是在节点生成后，对于某一个节点：①如果不进行划分，则对当前节点在训练集上采用多数投票法得到当前节点的分类结果，并在测试集上计算准确率。②如果进行划分，则当前节点下的子树已经通过特征选择算法进行生成，对于该子树在验证集上计算准确率。最后对比划分前后的准确率，来决定在节点生成后是否进行后剪枝。

## 第4章 系统的应用与预测结果分析

### 4.1钓鱼网站辨别系统的构建

#### 4.1.1 决策树的表示方式

构建一棵决策树需要有合适的逻辑结构，由于决策树是一颗多叉树，任一节点的子节点个数不是固定的，这就要求每个节点需要支持任意个数的子节点。在数据集中，一个特征最多有三种取值，如果采用孩子表示法，则对每个节点都需要设置三个孩子节点，当该节点的特征只有两个取值时，将会有一个孩子节点不能存储信息，造成空间的浪费；如果采用孩子兄弟表示法，则得到的是一颗二叉树，在绘制决策树时需要先转化为多叉的决策树，处理过程更加复杂。



**图4-1 用嵌套字典表示多叉树**

在本项目中，采用的是嵌套字典来表示决策树。字典中存放的是一组键-值对，在嵌套字典中，值可以是一个字典。当用嵌套字典存放多叉树时，可以将父节点作为键，而它的多棵子树合并为一个字典。由于决策树的每条边代表父节点对应特征的一种取值，所以在嵌套字典中还需存放边的信息。父节点特征对应的字典中，将边作为键，将子节点作为子字典。通过这样的方式，能够很清晰的表示出树的层级结构。

#### 4.1.2 决策树的构建

在第三章中论述了决策树特征选择和剪枝的相关算法，在构建决策树时，需要将两种算法进行组合，这两种算法构建的是特定节点，而我们的目标是构建整棵决策树。为此，可以采用递归的方式逐层的构建每一个节点。

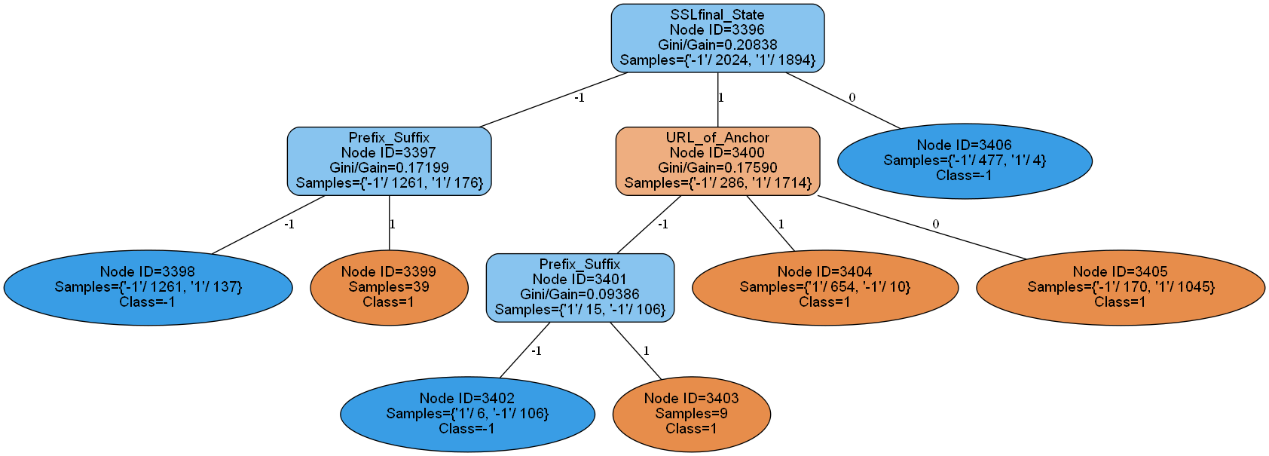
对于每一个节点，先进行预剪枝，来决定当前节点是否为叶子节点。如果剪枝为叶子节点，则直接向上层返回该叶子节点；否则进行特征选择，并将数据集传递到下一层，构建子树；构建完本节点的子树后再进行后剪枝；最后将本节点返回给上层作为父节点的子节点。

为得出最优的构建方式，可通过循环设置构建决策树的不同算法组合，得到不同决策树，选取其中在测试集上准确率最高的决策树最为最终的辨别系统。

### 4.2钓鱼网站辨别系统的绘制

生成决策树后为更好的分析决策树系统，还需对决策树进行绘制。本项目中采用Graphiz库绘制决策树，使用该库可以方便的绘制各种流程图，同样也可以方便的绘制决策树。

为更好的分析每一个节点，在绘制的决策树中，每个非叶节点将包含以下信息：节点编号、节点的信息增益/信息增益比/基尼值、样本统计情况；而每个叶节点则包含：节点编号、样本统计情况、分类结果。通过这些信息能够很方便的判断决策树是否正常构建，根据节点编号能快速定位错误节点并进行调试。



**图4-2 CART算法预剪枝构建的决策树**

上图是CART算法在预剪枝情况下生成的决策树，总共包含11个节点，每个，不同类别的节点颜色不同，设置了五种不同的颜色进行区分：对于非叶节点，其颜色由样本的分类比例决定，如果数据集中属于钓鱼网站的样本数量占比大，则节点为浅蓝色；如果钓鱼网站与非钓鱼网站的占比相同，则节点设置为白色；否则设置为浅橙色。对于叶节点，如果辨别结果为钓鱼网站，则叶节点设置为蓝色；否则设置为橙色。

### 4.3系统的预测准确率分析

将三种特征选择算法和两种剪枝算法进行组合，根据收集到的数据构建决策树。用构建好的决策树对训练集进行预测，不同算法组合下决策树的节点数量和预测的准确率如下表：

表4-1训练集结果比对表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 预剪枝 | 后剪枝 | ID3/节点数 | C45/节点数 | CART/节点数 |
| 否 | 否 | 0.9926/1109 | 0.9926/1015 | 0.9926/1242 |
| 是 | 否 | 0.9219/24 | 0.9165/11 | 0.9165/11 |
| 否 | 是 | 0.9587/223 | 0.9627/287 | 0.9592/225 |
| 是 | 是 | 0.9219/24 | 0.9165/11 | 0.9165/11 |

再用决策树对测试集进行预测，不同算法组合下决策树的节点数量和预测的准确率如下表：

表4-2 测试集结果比对表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 预剪枝 | 后剪枝 | ID3/节点数 | C45/节点数 | CART/节点数 |
| 否 | 否 | 0.9171/1109 | 0.9197/1015 | 0.9135/1242 |
| 是 | 否 | 0.9151/24 | 0.9089/11 | 0.9089/11 |
| 否 | 是 | 0.9534/223 | 0.9508/287 | 0.9524/225 |
| 是 | 是 | 0.9151/24 | 0.9089/11 | 0.9089/11 |

#### 4.3.1 剪枝算法及特征选择算法分析

由结果对比表可知，当不进行剪枝处理时，决策树的节点数在1000个以上，在训练集上的准确率普遍高于在测试集上的准确率，说明未使用剪枝算法的情况下决策树存在过拟合。其中如果不进行剪枝算法，则在训练集上的准确率可达到99%以上，但在测试集上的准确率却仅为91%左右，表明未剪枝的决策树过拟合严重。

当使用预剪枝算法时，构建得到的决策树在测试集上和训练集上的准确率均较低，出现这一情况有两方面的原因：

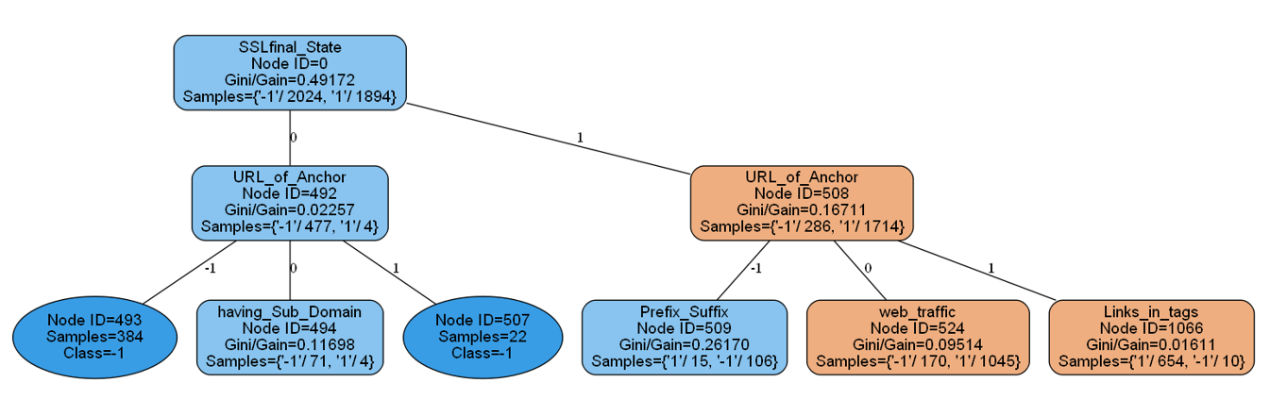
一是采用预剪枝算法主要采用的是多数投票法，在决策树构建过程中可能出现完全使用多数投票法构建子树的情况。相比于特征选择算法，多数投票法仅根据数据集中钓鱼网站样本的占比来决定分类结果，这种方式较为粗糙。

另一个原因是在本系统中，预剪枝算法会剪除大部分的节点。结合上表可知，使用预剪枝算法构建的决策树其节点数量少于25个，层数少于7层，而每个样本有30个特征。所以使用预剪枝算法构建的决策树并没有充分的利用数据的特征进行决策，构建的决策树虽然简洁但预测准确率相对较低。

当只使用后剪枝算法构建决策树时，得到的决策树节点在200到300之间，相较于预剪枝算法构建的少量节点的决策树，层数在12到24层，能更好的发挥各个特征的作用。相较于预剪枝算法，后剪枝是在子树构建完成后对子树的准确率进行评估，再决定是否进行剪枝，所以后剪枝在剪枝后的预测准确率必定会高于剪枝前，使得后剪枝算法构建的决策树预测效果更佳。

整体上来说三种特征选择算法构建的决策树的预测效果相差不大。由于数据集的每个特征都只含有两到三种取值结果，不会出现某个特征取值数量多导致其信息增益偏大的情况，所以采用信息增益率进行特征选择的C4.5算法构建的决策树，其预测效果和ID3算法接近。

#### 4.3.2 决策树分析

在测试集上，ID3算法与后剪枝算法结合所构建的决策树的预测准确率最高，预测准确率为95.34%。该决策树有绘制相应的决策树有223个节点，为便于分析，这里仅展示前三层结构（详细的决策树见附件ID3.png）：

**图4-3 ID3算法后剪枝决策树的部分结构**

决策树根节点特征为SSLfinal\_State，该特征表示的是否是应用HTTPS、发布者是否可信以及证书年限：若使用HTTPS、HTTPS可信且证书年限超过一年时，SSLfinal\_State取值为1，表示合法；证书不可信时SSLfinal\_State取值为0，表示可疑；否则取值为-1，表示非法。

决策树的采用该特征作为根节点表明：通过网站是否使用HTTPS以及HTTPS的SSL证书的使用时间，可以较好的判断该网站是否为钓鱼网站。结合实际，通过网站的SSL证书可以判别服务器的真实身份，进而辨别钓鱼网站。而大部分的钓鱼网站的存活时间较短，且大多采用不安全的HTTP，所以如果一个网站如果使用HTTP，或者使用HTTPS但证书年限较短，则很可能时钓鱼网站。

当SSLfinal\_state取不同值时其第二层节点的特征均为URL\_of\_Anchor，该特征表示的是a标签的占比，当一个网页的<a>标签占比超过31%时，可认定该网页为疑似钓鱼网站，当a标签的数量超过67%时，该网站有很大的可能是钓鱼网站。同样结合实际，钓鱼网站通常模仿常规网站对用户进行欺骗，通过a标签将常规网站的元素链接到钓鱼网站中，能快捷的完成网页的布局，所以一个钓鱼网站中可能会有大量的a标签。

由上述的分析可知，本项目构建的决策树符合实际，能较好的辨别钓鱼网站，具有实用性。

## 第5章 总结和展望

### 5.1总结

#### 5.1.2系统的优点

根据钓鱼网站辨别系统的运用效果得出本系统具有以下优点：

1. 本系统可以选择多种特征选择算法来构建决策树，综合运用了ID3、C4.5、CART算法，系统可以根据测试集的实验结果自由选择合适的决策树，从而得到最佳的判别效果。
2. 为减轻决策树过拟合问题，系统中采用了预剪枝和后剪枝算法，极大的改善了决策树的拟合问题，具体表现为在测试集中只采用后剪枝算法生成的决策树，其准确率比未剪枝的决策树提高了四个百分点。剪枝使本辨别系统具有更好的决策能力。
3. 本系统采用的数据集具有丰富而且必要的特征，这是决策树具有良好决策能力的重要前提，在运用于测试集中时，准确率达到95%，决策树的决策能力充分展现了出来。
4. 本系统设计的决策树具有通用性，不仅适用于钓鱼网站的辨别，还适用于一般的分类问题。只需将数据输入到系统，并能快速的构建和绘制出决策树。

#### 5.1.2系统的缺点

虽然本系统运用于测试集的最佳准确率可以达到95%，但还是存在以下缺点：

1. 系统所用到的数据发布于2015年，距今已有7年，而7年时间里互联网也在不断发展，其中的某些特征对于识别钓鱼网站的作用或许会减弱或增强，将本系统运用于现在的钓鱼网站识别其效果可能会有偏差。
2. 系统中采用的数据集是从URL中提取出来的一组特征，系统是基于这一组特征来构建决策树的。如果需要处理一个URL，则首先需要将相应的特征提取出来，再传入决策树中进行判别。而本系统暂未设计特征提取的功能，需要通过其他方式从URL中提取特征数据后才能传入系统。

### 5.2 展望

Sklearn库是python生态中流行的机器学习库，sklearn提供了丰富的API用于快速完成各种机器学习任务，其中也提供了决策树算法。将sklearn提供的决策树API应用于本项目中收集到的钓鱼网站数据，在测试集上的准确率可以到达97%，而本系统所实现的决策树算法在测试集上的最佳准确率为95%。这表明本系统的决策树算法还有提升空间。

通过分析剪枝算法后发现，本系统采用的剪枝算法基于多数投票法，这种方式简单快捷，要构建更佳的决策树，可采用CART剪枝算法。CART剪枝算法是一种后剪枝算法，当决策树构建完毕后，自底向上的对每个节点进行剪枝判断。这一算法由可分为两步：第一步在参数的控制下生成最优子树序列，第二步在子树序列中通过交叉验证法选择最优子树。这种方式产生的决策树具有更好的决策能力，是本系统的一个优化方向。

## 参考文献

[1]云雷,李丹,王欢欢.钓鱼网站检测技术研究综述[J].电子产品可靠性与环境试验,2021,39(05):114-119.

[2]陈鹏,李勇志,余肖生.基于特征选择模型的钓鱼网站快速识别方法[J].计算机技术与发展,2021,31(04):40-45.

[3]Dželila Mehanović,Jasmin Kevrić. Phishing Website Detection Using Machine Learning Classifiers Optimized by Feature Selection[J]. Traitement du Signal,2020,37(4):

[4]Routhu Srinivasa Rao,Tatti Vaishnavi,Alwyn Roshan Pais. CatchPhish: detection of phishing websites by inspecting URLs[J]. Journal of Ambient Intelligence and Humanized Computing,2020,11(2):

[5]Shafaizal Shabudin,Nor Samsiah Sani,Khairul Akram Zainal Ariffin,Mohd Aliff. Feature Selection for Phishing Website Classification[J]. International Journal of Advanced Computer Science and Applications (IJACSA),2020,11(4):

[6]王文腾,王传涛,袭薇,佟晖.钓鱼网站分类检测算法的比较性研究[J].北京建筑大学学报,2019,35(01):76-81.DOI:10.19740/j.1004-6011.2019.01.12.

[7]陈炜宏.基于决策树模型的构建方法及预测能力的分析——以钓鱼网站为例[J].通讯世界,2018,25(12):233-235.

[8]周志华著. 机器学习. 北京：清华大学出版社, 2016.01.

[9]李航著. 统计学习方法. 北京：清华大学出版社, 2012.03.

[10]莫勇. 基于决策树算法的Web客户端脚本安全检测研究[D].北京林业大学,2015.

## 致谢

在近一个学期的摸索与努力下，毕业设计的各项工作都已接近尾声。

在这个过程中，通过一步步的查阅资料，我对决策树算法的构建过程有了更深入的认识，从最初通过各类数据网站收集数据，在各类数据源中筛选和比对，得到适合本系统的数据；到后来通过手动实践设计决策树，并通过python语言实现决策树系统，对代码的规范和设计进行优化；再到后来通过可视化工具绘制决策树，使生成的决策树易于分析；最后完成这篇论文的撰写。整个过程花费了很多的时间，也有很多的收获。在此感谢李云洪副教授的指导。