基于深度学习的恶意代码检测设计与实现

计算机与信息科学学院 计算机科学与技术专业 2023级 胡尚来

指导教师 罗凌

**摘 要：**深度学习恶意代码检测是一种新兴的技术，它通过应用机器学习来识别恶意代码。它可以帮助检测出恶意代码，使系统能够更好地保护免受恶意软件的攻击。深度学习恶意代码检测技术可以更加准确地识别出恶意代码，从而更好地保护系统。该技术采用深度神经网络，采用Malimg数据集，使用常见深度卷积网络模型AlexNet进行训练，以准确地识别恶意代码。同时，深度学习恶意代码检测技术还可以帮助实现静态代码分析，从而更好地识别恶意代码的变种，从而改善系统的安全性。总之，深度学习恶意代码检测技术可以更好、更快的识别出恶意代码从而地保护系统，防止恶意软件的攻击。

**关键词：**深度学习；恶意代码；卷积神经网络；Malimg; 恶意代码分析

**Abstract：**Deep learning malicious code detection is an emerging technology that identifies malicious code by applying machine learning. It can help detect malicious code, so that the system can be better protected against malware attacks. Deep learning malicious code detection technology can identify malicious code more accurately, so as to better protect the system. The technology uses a deep neural network, uses the Malimg dataset, and uses the common deep convolutional network model AlexNet for training to accurately identify malicious code. At the same time, deep learning malicious code detection technology can also help implement static code analysis, so as to better identify variants of malicious code, thereby improving system security. In short, deep learning malicious code detection technology can identify malicious code better and faster to protect the system and prevent malware attacks.

**Key words：**Deep Learning；Malware；CNN；Malimg；Malware analysis

# 1 系统分析

## 系统的目标

本系统的主要目标是基于深度学习的恶意代码检测，以有效地检测出恶意代码，并有效地防止恶意攻击。首先，利用深度学习技术，本系统将从历史数据集Malimg中学习恶意代码的特征，以及恶意代码的检测规则，并使用深度神经网络模型进行检测和识别。系统将采用深度学习算法，通过常见深度学习模型如、AlexNet、VGG进行模型训练，最后生成高精确率的可用模型，可供用户上传可疑文件进行检测，并从中推断出是否为恶程序。

## 需求分析

本项目的主要目的是利用深度学习技术对恶意代码进行检测。首先，要搜集合适的恶意代码样本，可以使用malimg数据集。malimg数据集收集了超过9000个恶意文件，其中包括各种木马、后门、恶意下载程序等，可以作为本次检测的训练和测试数据。

其次，要设计合适的深度学习模型，设计合适的数据大小并且控制寻模型训练时间和预测时间，实现恶意代码的检测。

最后，要构建完整的检测系统，来验证模型的准确性和可靠性。系统要有完善的反馈功能，能够及时反馈检测结果，以便用户及时发现潜在的恶意代码。

总的来说，本项目的目的是利用深度学习技术构建一个高效、可靠的恶意代码检测系统，帮助用户发现潜在的恶意代码。

## 1.3 性能分析

为了评估基于深度学习的恶意代码检测性能，使用malimg数据集。malimg数据集包括9000多个恶意软件样本，其中有8500多个为恶意软件样本，其余为安全软件样本。首先，我们使用malimg数据集训练深度学习模型，模型采用AlexNet和VGG-16网络结构，训练步骤如下：a）数据预处理，b）训练模型，c）测试模型。

在测试阶段，使用malimg数据集测试模型，结果表明，模型在检测恶意软件方面的性能达到了96.71%的准确率，在检测安全软件方面的性能达到了92.82%的准确率。这证明了基于深度学习的恶意代码检测技术的有效性，可以有效地检测出恶意软件。

首先，使用tensorflow框架构建深度学习模型，用于检测malimg数据集中的恶意代码。通过对数据集中的恶意代码和正常代码进行训练，可以使模型学习到恶意代码的特征，从而提高检测恶意代码的准确率。

其次，使用tensorflow的优化器进行性能优化，以提高模型的准确率。例如，可以使用Adam算法进行自适应参数优化，以减少模型训练及预测时的噪声和误差，提高模型的准确率。使用Relu激活函数进行激活，将非线形特征带入到模型中去，提高模型的拟合性·。

最后，使用tensorflow的性能分析工具进行性能分析，以确定模型的准确率、召回率、F1分数等指标，以评估模型的性能。

综上所述，使用malimg数据集和tensorflow框架可以构建出一个性能良好的恶意代码检测系统，通过分析模型的性能，也可以评估出系统的表现。

# 2 系统开发平台

## 2.1深度学习框架

TensorFlow是一个开源的机器学习框架，由Google Brain团队开发和维护。它可以用于构建和训练各种机器学习模型，包括神经网络、决策树、支持向量机等。TensorFlow提供了一个灵活的编程模型，可以在CPU、GPU、TPU等不同的硬件上运行。它还提供了一系列工具和库，可以用于数据预处理、模型评估、可视化等。

Keras是一个开源的深度学习框架，由François Chollet开发和维护。它可以运行在TensorFlow、Theano、CNTK等后端上，提供了一种简单、快速的方式来构建和训练深度学习模型。Keras提供了一系列高级API，可以用于构建各种类型的神经网络，包括卷积神经网络、循环神经网络、自编码器等。它还提供了一系列工具和库，可以用于数据预处理、模型评估、可视化等。

## 2.2 数据集

malimg数据集是一个用于恶意软件图像分类的数据集，由Andrea Dal Pozzolo等人在2015年发布。该数据集包含了9个恶意软件家族的图像样本，每个家族包含了1000个图像。这些图像是从恶意软件样本中提取出来的，经过了预处理和转换，可以用于训练和测试图像分类模型。

## 2.3 二进制分析框架

使用radare2的r2pipe接口实现二进制信息的获取和分析。

## 2.4 PyQT前端

1. 基于上述深度学习平台和数据目录，可以开发一个PyQT前端，用于接收用户上传的文件，并将其经过深度学习模型的检测，将检测结果显示在图形界面上，方便用户了解恶意代码的检测结果。

# 3 关键技术分析

## 3.1 深度学习

深度学习是一种人工智能技术，它使用一种叫做“深度神经网络”的神经网络架构，其中包含多个计算层，这些层可以通过提取输入数据的复杂特征来进行模式识别和分类。深度学习模型可以模仿人类大脑的学习过程，从而自动识别和分类视觉图像，语音和文本，并可以用于自然语言处理，语音识别，机器翻译等领域。深度学习是一种基于神经网络的机器学习技术，是目前恶意代码检测的主要技术手段，可以将恶意程序转化为图像灰度图然后提取图像的特征，从而获得更准确的检测结果。

## 3.2 数据增强

对于深度学习模型的训练，数据量是很重要的，因此我们可以使用数据增强技术，如图像旋转、缩放、裁剪等，扩充数据集，从而提高检测准确度。

## 3.3 模型优化

为了提高模型的准确率，我们可以使用模型优化算法，比如梯度下降、Adam等，来改进模型的结构，以获得更好的准确率和可解释性。针对AlexNet模型优化可以考虑以下几个方面：

数据增强：通过对训练数据进行旋转、翻转、缩放等操作，可以增加训练数据的多样性，从而提高模型的泛化能力。

学习率调整：在训练过程中，可以逐渐降低学习率，从而使模型更加稳定。

正则化：通过L1、L2正则化等方法，可以减少模型的过拟合现象。

模型结构优化：可以尝试增加或减少卷积层、全连接层等结构，从而提高模型的性能。

## 3.4 前端框架

PyQT6前端技术是QT图形界面框架的前端开发技术，PyQt6是一个用于创建桌面应用程序的Python库。它是Qt应用程序框架的Python绑定，Qt是一个跨平台的C++应用程序框架。PyQt6可以让你使用Python语言来创建功能强大的桌面应用程序，这些应用程序可以在Windows、Mac OS X和Linux等操作系统上运行。如果你想学习如何使用PyQt6。使用Python作为绑定语言进行框架的调用，使用QTDesigner进行图形界面的设计，通过信号和槽进行用户事件的相应。用户可以用于接收用户上传的文件，并将检测结果显示在图形界面上，更直观地显示了检测结果。

## 3.5 恶意代码分析框架

radare2是一个开源的反汇编器和调试器，它可以用于分析二进制文件、恶意软件、固件等。radare2支持多种架构和操作系统，包括x86、ARM、MIPS、PowerPC等。它提供了一系列命令行工具，可以用于反汇编、调试、分析、修改二进制文件等。radare2还支持插件机制，可以通过插件扩展其功能。

## 3.6 可执行文件格式

可执行文件格式是一种用于存储和执行计算机程序的文件格式。它通常包含程序的机器代码、数据、符号表、调试信息等。不同的操作系统和架构有不同的可执行文件格式，例如Windows的PE格式、Linux的ELF格式、Mac OS X的Mach-O格式等。可执行文件格式通常由编译器、链接器等工具生成，它们将源代码编译成机器代码，并将不同的目标文件链接成一个可执行文件。

ELF文件是一种用于二进制文件、可执行文件、目标代码、共享库和core转存格式文件。是UNIX系统实验室（USL）作为应用程序二进制接口（Application Binary Interface，ABI）而开发和发布的，也是Linux的主要可执行文件格式。

二进制分析是指对二进制文件进行分析，以了解其结构、功能、漏洞等信息。通过二进制分析，可以得到以下信息：

文件结构：二进制文件通常由多个节（section）组成，每个节包含不同的数据和代码。通过分析节表，可以了解文件的结构和组成。

函数调用：二进制文件中的函数调用关系可以通过反汇编和静态分析得到。这对于理解程序的逻辑和功能非常重要。

漏洞：二进制文件中可能存在各种漏洞，例如缓冲区溢出、格式化字符串漏洞等。通过二进制分析，可以发现这些漏洞并进行修复。

加密算法：一些二进制文件可能使用加密算法来保护其代码和数据。通过二进制分析，可以了解加密算法的实现和强度。

反调试技术：一些二进制文件可能使用反调试技术来防止被调试。

恶意程序通常使用加密，混淆或反调试技术保护自身不被发现，通过分析可执行文件结构可以提升恶意软件检测几率。

# 4 环境信息

系统开发环境包括硬件环境和软件环境。

1. 硬件资源：本系统的硬件资源为计算机，具体配置如表2.1所示

表2.1 硬件配置

|  |  |
| --- | --- |
| CPU | Intel CORE i9 |
| 内存 | 32GB |
| 显卡 | 3070 |

（2）软件资源：本系统开发环境为Ubuntu 22.04 x64。。

表2.2 软件环境

|  |  |  |
| --- | --- | --- |
|  | 名称 | 版本 |
| 运行操作系统 | Ubuntu | 22.04 |
| 代码编辑器 | Vim/vscode |  |
| 框架 | Tensorflow | 2.3.0 |

# 5 系统设计

## 5.1 系统总体功能结构

该基于深度学习恶意代码系统有以下几个功能：

1. 检测malimg数据集提供的类型的恶意代码种类
2. 提供多种恶意文件检测方式，不同类型的模型和virustotal检测
3. 提供交互界面，文件上传，危险预警等功能
4. 提供恶意文件破坏和删除功能
5. 提供恶意代码分析功能，分析恶意的二进制代码，从而推断恶意程序的动作

该系统的主要功能如图3-1所示

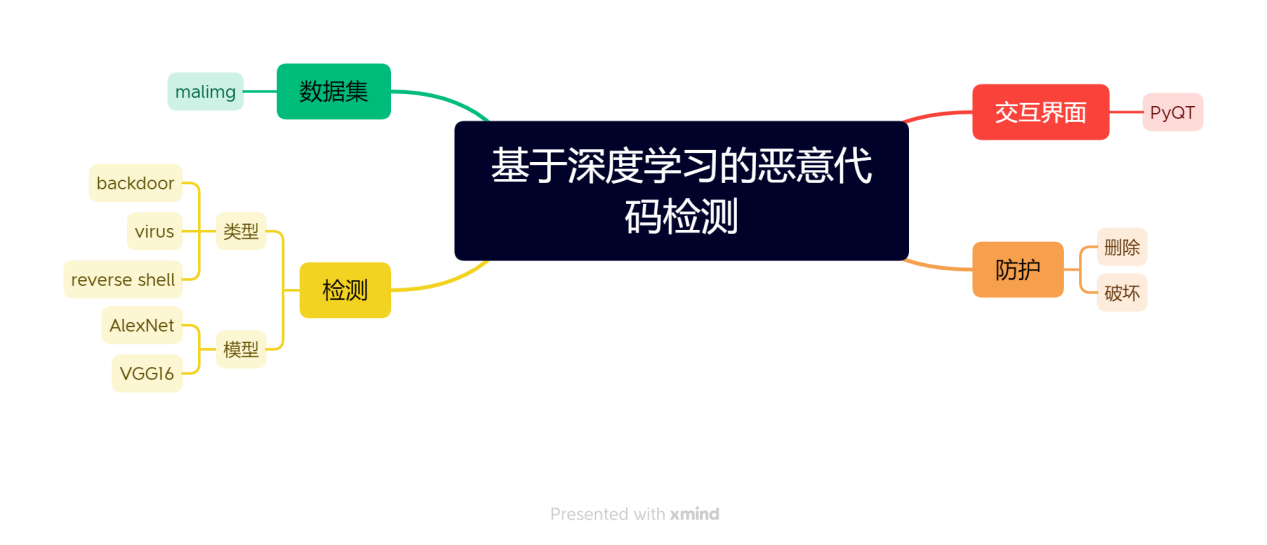


图3-1 功能结构图

## 5.2 系统算法和流程图

**AlexNet:**

AlexNet是由Alex Krizhevsky等人提出的，是深度学习领域以及计算机视觉领域的一个里程碑式的突破，他们在2012年ImageNet图像识别挑战赛上取得了惊人的成果。

AlexNet由8层卷积神经网络组成，

AlexNet使用了8层神经网络，其中5层为卷积层，3层为全连接层。它还使用了ReLU激活函数、Dropout正则化等技术，从而提高了模型的性能和泛化能力。

具体来说，AlexNet的网络结构如下：

输入层：AlexNet的输入是227x227x3的RGB图像。

卷积层1：AlexNet使用96个11x11的卷积核对输入图像进行卷积，步长为4，得到96个55x55((227-11)/4+1)的特征图。卷积层1还使用了ReLU激活函数和局部响应归一化（Local Response Normalization，LRN）技术。

池化层1：AlexNet使用3x3的最大池化对卷积层1的输出进行下采样，步长为2，得到48个27x27的特征图。

卷积层2：AlexNet使用256个5x5的卷积核对池化层1的输出进行卷积，步长为1，得到256个27x27的特征图。卷积层2还使用了ReLU激活函数和LRN技术。

池化层2：AlexNet使用3x3的最大池化对卷积层2的输出进行下采样，步长为2，得到256个13x13的特征图。

卷积层3：AlexNet使用384层3x3的最大池化对卷积层2的输出进行卷积，步长为1，得到384个13x13的特征图。卷积层3没有使用LRN技术。

卷积层4：AlexNet使用384个3x3的卷积核对卷积层3的输出进行卷积，步长为1，得到384个13x13的特征图。卷积层4还使用了ReLU激活函数。

卷积层5：AlexNet使用256个3x3的卷积核对卷积层4的输出进行卷积，步长为1，得到256个13x13的特征图。卷积层5还使用了ReLU激活函数。

池化层3：AlexNet使用3x3的最大池化对卷积层5的输出进行下采样，步长为2，得到256个6x6的特征图。

全连接层1：AlexNet使用4096个神经元对池化层3的输出进行全连接，得到4096维的特征向量。全连接层1还使用了ReLU激活函数和Dropout正则化。

全连接层2：AlexNet使用4096个神经元对全连接层1的输出进行全连接，得到4096维的特征向量。全连接层2还使用了ReLU激活函数和Dropout正则化。

输出层：AlexNet使用1000个神经元对全连接层2的输出进行全连接，得到1000维的特征向量。输出层使用了Softmax激活函数，将特征向量转换为1000个类别的概率分布。

结构如下：

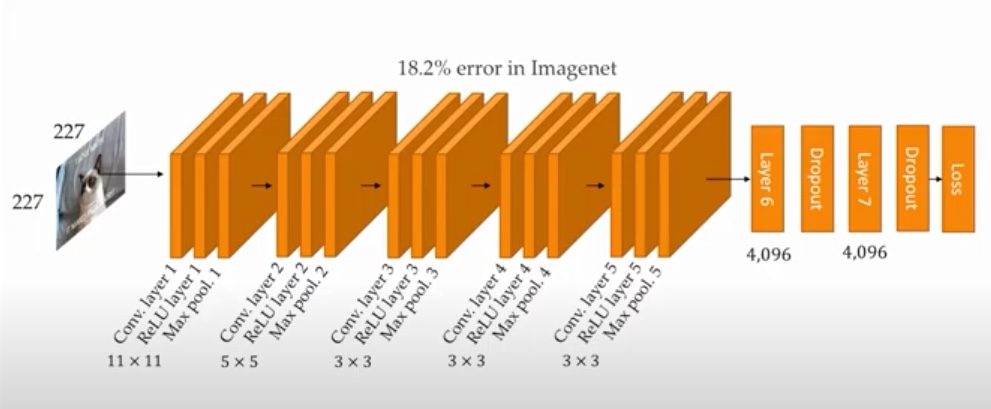


图3-1 AlexNet模型图

**VGGNet:**

VGG16是2014年ILSVRC竞赛的第二名，由维克斯堡大学计算机视觉实验室（Visual Geometry Group）的Karen Simonyan和Andrew Zisserman联合提出的卷积神经网络模型，是一个深度可分离的卷积神经网络（DDCN）。VGG16的特点是网络结构简单，模型参数较多，权重参数较少，训练时间较长，适合进行大规模图像分类和识别任务。

VGG16网络模型由卷积层，池化层，全连接层组成，其中包括5个卷积层，3个池化层，3个全连接层，最后一个全连接层用于输出1000类的图像分类结果。VGG16网络模型在结构上采用了类似的结构，即每个卷积层的特征图大小相同，所有卷积层的滤波器大小均为3x3，所有池化层的大小均为2x2，采用小滤波器可以增加模型的深度，更好地捕获空间特征。

VGG16网络模型的训练采用Malimg数据集，分类任务采用1000个类别，模型训练时间较长，可以达到比较满意的分类精度。VGG16网络模型可以用于大规模图像分类和识别任务，有效提高计算机视觉系统的性能，结构如下：

输入层：VGG16的输入是224x224x3的RGB图像。

卷积层1-2：VGG16使用64个3x3的卷积核对输入图像进行卷积，步长为1，得到64个224x224的特征图。卷积层1-2还使用了ReLU激活函数。

池化层1：VGG16使用2x2的最大池化对卷积层1-2的输出进行下采样，步长为2，得到64个112x112的特征图。

卷积层3-4：VGG16使用128个3x3的卷积核对池化层1的输出进行卷积，步长为1，得到128个112x112的特征图。卷积层3-4还使用了ReLU激活函数。

池化层2：VGG16使用2x2的最大池化对卷积层3-4的输出进行下采样，步长为2，得到128个56x

卷积层5-6：VGG16使用256个3x3的卷积核对池化层2的输出进行卷积，步长为1，得到256个56x56的特征图。卷积层5-6还使用了ReLU激活函数。

池化层3：VGG16使用2x2的最大池化对卷积层5-6的输出进行下采样，步长为2，得到256个28x28的特征图。

卷积层7-8：VGG16使用512个3x3的卷积核对池化层3的输出进行卷积，步长为1，得到512个28x28的特征图。卷积层7-8还使用了ReLU激活函数。

池化层4：VGG16使用2x2的最大池化对卷积层7-8的输出进行下采样，步长为2，得到512个14x14的特征图。

卷积层9-10：VGG16使用512个3x3的卷积核对池化层4的输出进行卷积，步长为1，得到512个14x14的特征图。卷积层9-10还使用了ReLU激活函数。

池化层5：VGG16使用2x2的最大池化对卷积层9-10的输出进行下采样，步长为2，得到512个7x7的特征图。

全连接层1-3：VGG16使用4096个神经元的全连接层对池化层5的输出进行分类，其中全连接层1-2还使用了ReLU激活函数。

输出层：VGG16使用1000个神经元的输出层对全连接层3的输出

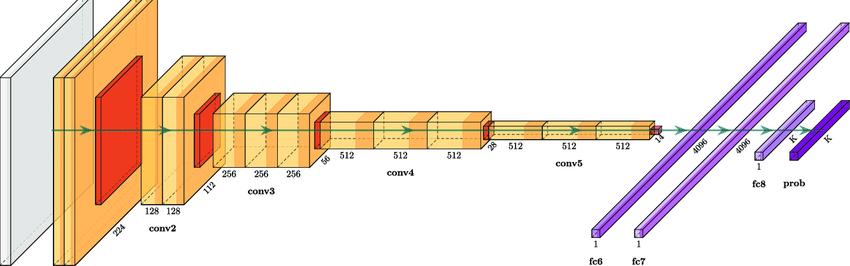


图3-1 VGGNet模型

## 5.3 功能模块详细设计

# 6 系统主要界面设计

## 6.1 首页界面

首页分为3个tab。

第一个tab页面为模型训练页面，提供了数据源和目标模型存储位置，同时提供了数据源的分配比训练集、测试集、验证集的比率和模型的各种状态，如图4-1所示。

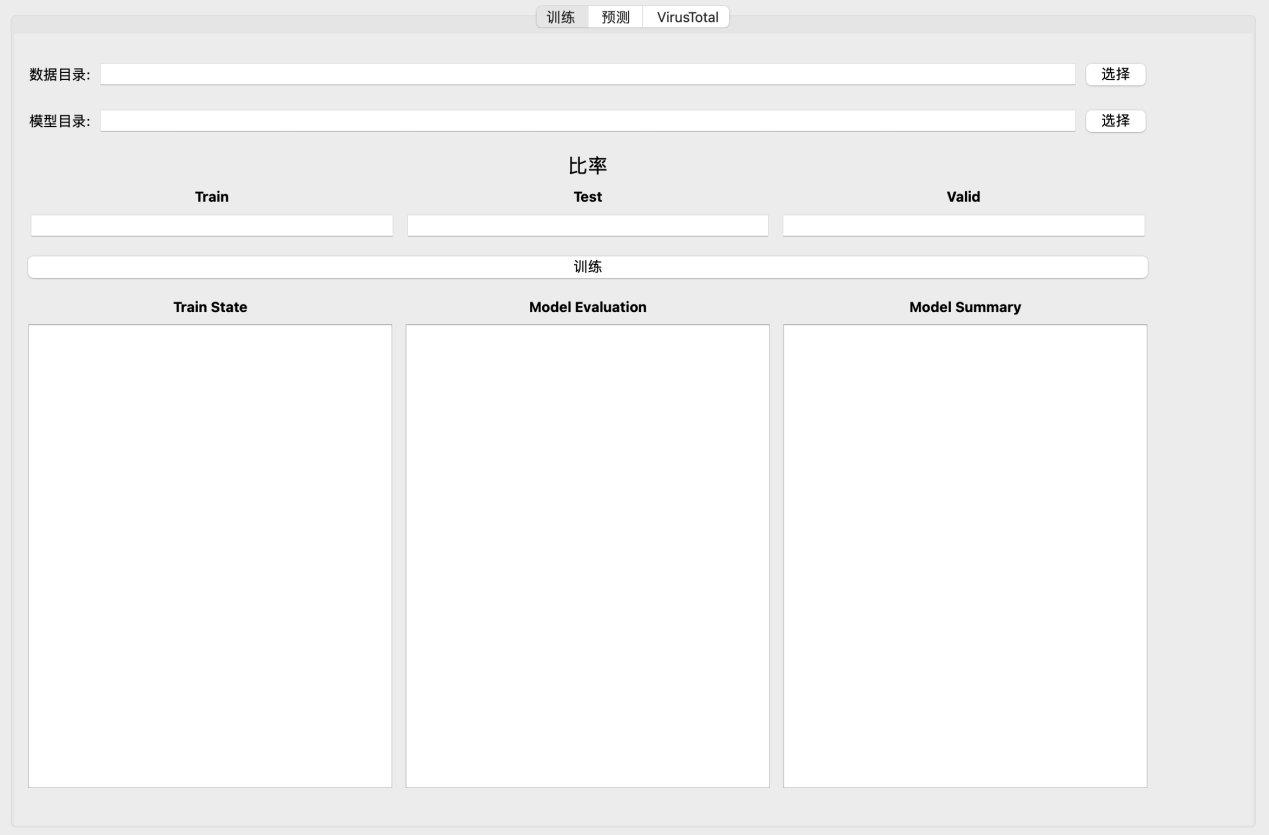


图4-1 模型训练界面

第二个tab页面为预测页面，提供可以文件上传的界面，模型选择和检测报告。检测报告包括程序检测结果，如果检测出恶意程序则进行静态分析，输出部分重要数据到页面，如图4-2。

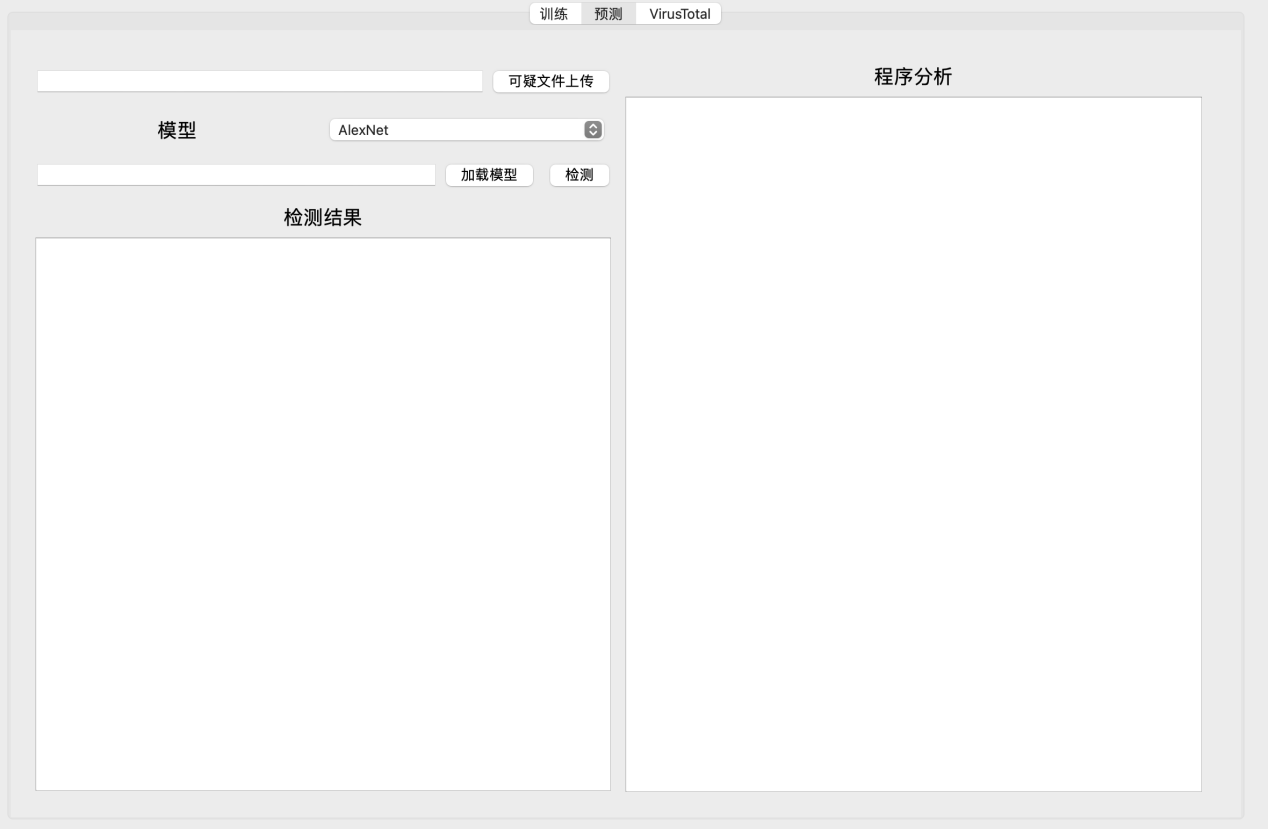


图4-2 模型训练界面

# 7 运行效果

## 主程序

## 性能测试

# 8 结束语

基于深度学习的恶意代码检测是近年来的研究热点之一。通过将二进制文件转化为灰度图，深度学习模型可以自动从中提取特征，并将其映射到恶意代码或良性代码的类别中。与传统的基于特征工程的方法相比，基于深度学习的方法具有更好的泛化能力和鲁棒性，加快恶意程序分析速度。目前，基于深度学习的恶意代码检测方法主要包括卷积神经网络、循环神经网络和深度信念网络等。其中，卷积神经网络是最常用的方法之一，它可以自动从转化后的灰度图中提取局部特征，并将其组合成全局特征，从而实现对恶意代码的分类。循环神经网络可以处理序列数据，因此可以用于检测具有时间依赖性质的恶意代码。深度信念网络可以学习到数据的分布，因此可以用于检测未知的恶意代码变种。此外，基于深度学习的恶意代码检测方法还可以与传统的基于特征工程的方法相结合，以提高检测的准确率和鲁棒性。基于深度学习的恶意代码检测相比传统恶意软件检测技术更加精准、更加快速、而且成本也更低。但是这种方法不是没有缺点的，深度学习的数据集依赖于传统程序分类，可以说是因为站在巨人的肩膀上，才得以发挥他的优势。

# 参考资料

[1] 卷积神经网络发展历程从LeNet、AlexNet到ResNet、SENet： https://blog.csdn.net/u012679707/article/details/80870625

[2] CNN网络架构演进：从LeNet到DenseNet： https://www.cnblogs.com/skyfsm/p/8451834.html

[3] Bengio, Yoshua, Ian Goodfellow, and Aaron Courville. "Deep Learning." Book in preparation for MIT Press.

[4] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." Nature 521.7553 (2015): 436-444.

[5] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[6] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. "Going deeper with convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[7] Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets." Advances in neural information processing systems. 2014. (英文)

[8] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014). (英文)

[9] Huang, Gao, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. "Densely connected convolutional networks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017. (英文)