采用卷积神经网络和注意力机制判断是否为恶意样本

随着互联网规模和软件数量爆炸式增长的场景下，恶意软件也出现了很多的变异版本，仅靠运行前的静态分析以及评估技术难以满足多变环境下的需求。因此，项目组将进一步研究程序运行时的动态执行序列，总结恶意程序在运行时的共同特征，比如调用了那些典型的dll（动态链接库）。在此基础上，采用深度学习的方法学习动态执行序列的先后顺序、dll的调用关系等特征，进而快速对程序进行自动化判断。

判断给定程序是恶意样本还是正常程序，主要是根据恶意样本的动态执行序列以及调用的dll的特征进行判断。恶意样本检测包括动态执行序列向图片的转换和CNN+自注意力机制对图片进行判断。

动态执行序列是程序运行时通过Intel PT记录程序的跳转地址和跳转方向。Intel PT（process trace）是英特尔架构上的新硬件功能，Intel PT允许以最小的开销（少于5％的运行时开销）捕获分支结果和有关在处理器上执行的代码的时序信息。通过执行每个分支的结果，开发人员可以离线重构程序执行的控制流，并使用该信息来辅助分析。PT跟踪由一系列数据包（具有不同的类型）组成。例如，为了表示条件分支的选择，Intel PT使用的TNT数据包有两种不同的大小：8位和64位。为了重构执行流程，还需要考虑其他一些事情，例如间接分支，函数返回或中断。为了对此建模，英特尔PT添加了更多数据包，例如TIP用于间接分支和函数返回，以及FUP用于异步事件位置。

最近，随着深度学习的发展，图像分类技术得到了改善。 现阶段，使用卷积神经网络（CNN）的方法具有比人类更好的分类性能。注意力是由我们如何视觉关注图像的不同区域或如何关联一个句子中的单词所产生的。深度学习中的注意力可以广义地解释为重要性的权重：为了预测或推断一个元素，例如图像中的像素或句子中的单词，我们使用注意力来估计其与其他元素相关联的程度。带有注意力机制的CNN工作流程如图1所示。

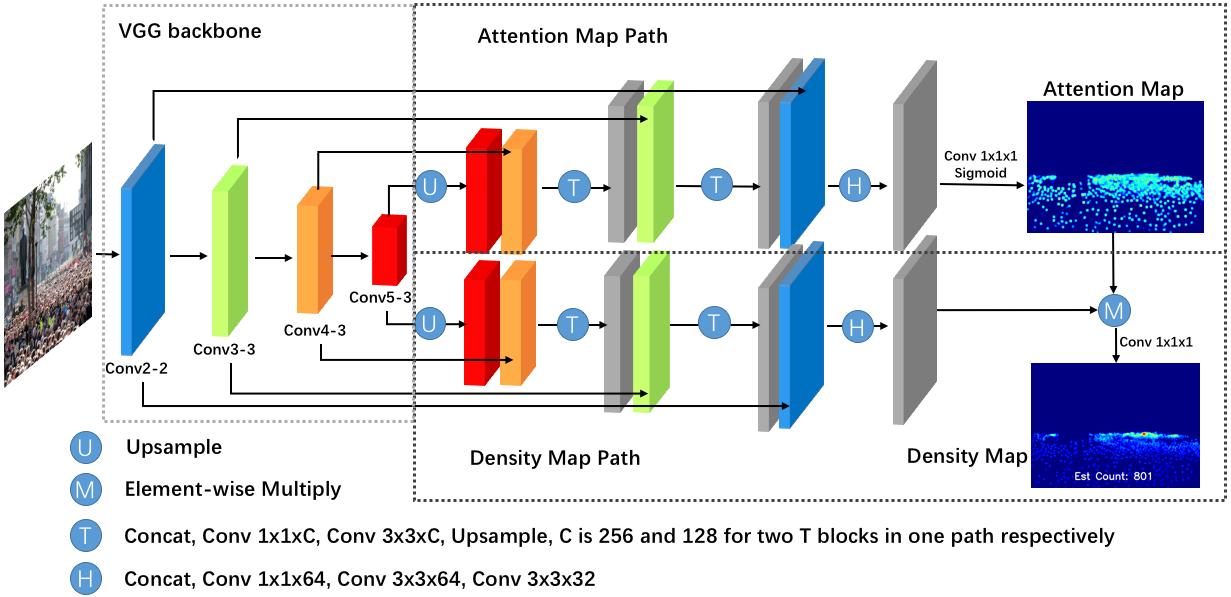
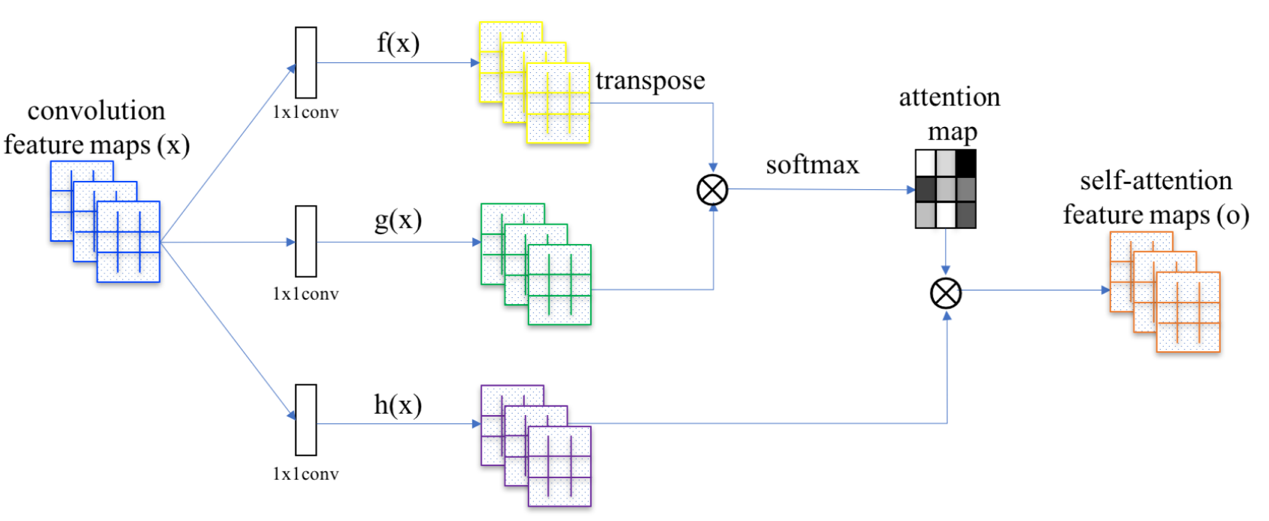


图1 带有注意力机制的CNN



针对程序执行时的动态指令序列，项目组设计了一种特殊的one-hot编码模式表示控制流序列。为此，项目组首先通过Intel PT对程序运行进行动态追踪，记录在CPU中产生的数据包，这些数据包的集合就是控制流序列。项目组首先执行所有的样本并收集所有样本的控制流序列，得到的控制流序列使用特殊的one-hot编码进行表示，再根据样本是否为恶意样本对控制流序列添加标签（Type），形成。这种one-hot编码和RGB编码是一一对应的，所以控制流序列可以转换为彩色图片表示，图片与标签形成。将图片放入带有注意力机制的卷积神经网络(CNN)中形成恶意程序的预测模型。对于需要检测的程序，只需要将程序动态执行并收集控制流序列，再将控制流序列使用特殊的one-hot编码表示，最终转换成图片。将图片输入预测模型就能得到程序是否为恶意的判断结果。具体的深度学习判别恶意代码的过程如图2所示。

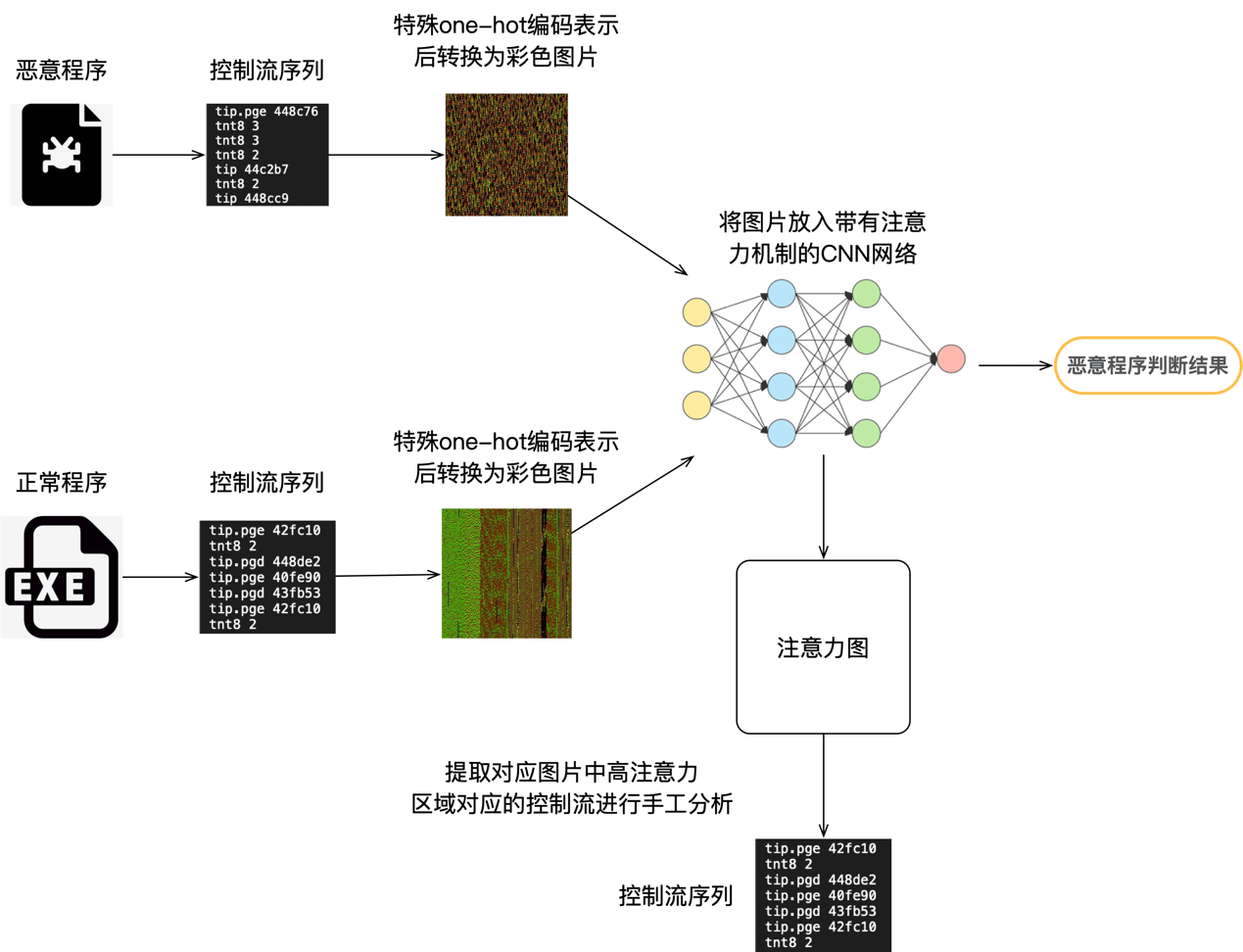


图2 深度学习判断恶意程序

1. 数据集中的恶意程序和正常程序在intel PT的追踪下进行执行，通过该工具动态收集执行过程中产生的数据包并记录。在追踪的同时记录程序的起始地址以及程序中调用的dll所在地址。
2. 设计一种特殊的one-hot编码，将记录的每一条控制流数据包转换为一个RGB值，将数据包中的地址与dll地址进行对照，将普通跳转指令和调用dll的指令进行区分，使用不同的RGB值进行表示。
3. 将整个序列裁剪为m\*m的段落，每一个段落都可以填充在图片中得到一个二维彩色图片。
4. 将转换后的图像通过注意力机制应用于CNN。使用已知的样本的标签和样本对应的图像来训练网络。然后，网络预测输入的样本是否为恶意样本并输出图像对应的注意力图。将注意力图中高注意力区域对应的控制流进行分析，使得预测模型具有可解释性。
5. 啦啦啦啦