人工智能大作业

基于深度学习的DGA域名检测

SY1806214

陈鸿超

目录

[1 背景介绍 3](#_Toc533930151)

[1.1 DGA域名简介 3](#_Toc533930152)

[1.2 传统的检测方法 3](#_Toc533930153)

[1.3 最新进展 3](#_Toc533930154)

[2 数据与模型 3](#_Toc533930155)

[2.1 数据来源 3](#_Toc533930156)

[2.2 数据集构造 3](#_Toc533930157)

[2.3 模型设计 3](#_Toc533930158)

[3 代码展示 3](#_Toc533930159)

[3.1 数据处理 3](#_Toc533930160)

[3.2 模型部分代码 3](#_Toc533930161)

[4 实验结果 3](#_Toc533930162)

# 背景介绍

## DGA域名简介

如今，互联网上的很多恶意攻击行为开始借助由域名生成算法生成的域名来抵抗安防软件的检测。这些域名生成算法通常会借助一组随机种子，持续不断的生成大量随机域名。使用这些DGA域名进行攻击的流程如图1.1所示。

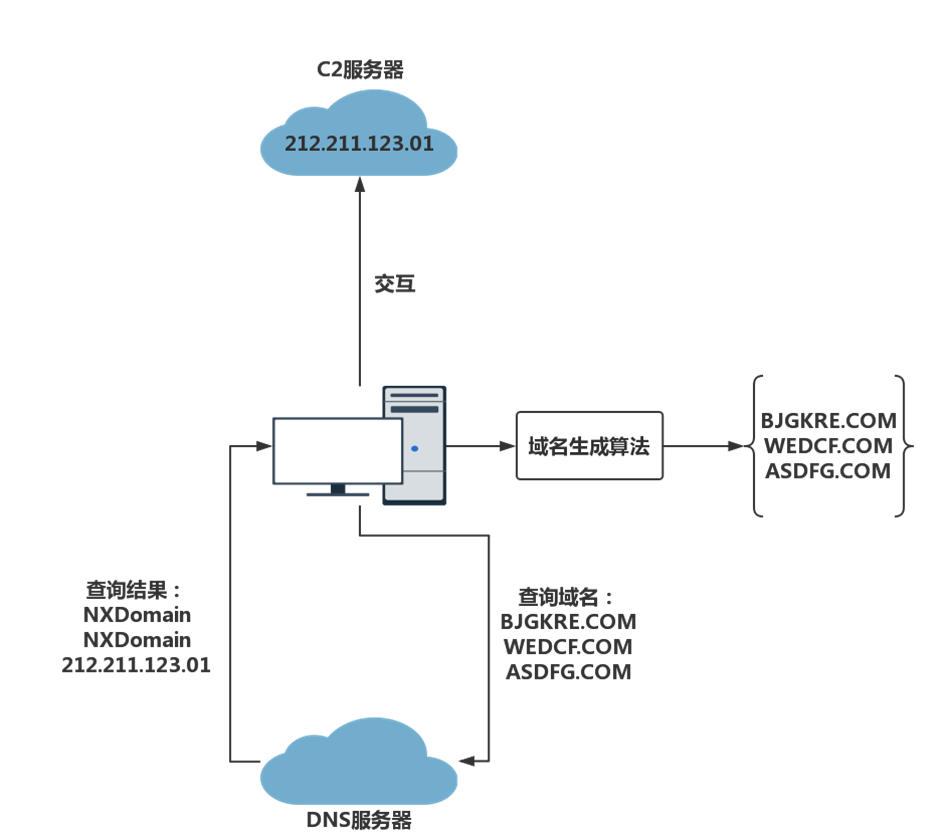


图 1.1 DGA类攻击流程

如图1.1 ，当某台主机或者服务器被恶意行为攻击成功之后，攻击者就会在被攻击者内部嵌入一段恶意代码。这段恶意代码会使用某种域名生成算法不断的生成DGA域名，并且利用被攻击者的主机去访问这些域名。同时，攻击者也会使用相同的算法生成大量DGA域名，并从中选取若干个进行注册。这样一来，当被攻击者成功访问某个DGA域名时，就说明这个域名是由攻击者注册的，比如图1.1中的ASDFG.COM这条域名。检测到这种被注册的DGA域名之后，嵌入到被攻击者主机内的恶意代码就会通过这条DGA域名查询得到的IP地址，与攻击者使用的C2服务器进行交互，使得攻击者能够持续的控制被攻击的主机并窃取数据。

## 传统的检测方法

DGA域名与正常的域名相比有很多区别，传统的研究方向主要分为两种，一是对域名的字符串特征进行分析，一是对DNS流量上进行分析。

域名字符串分析是一种实时检测的方式，它不依赖与任何的上下文信息，只是单纯的根据域名字符串进行分析，是一种比较难的检测方式。传统的方法主要还是需要人工去设计特征，然后将这些特征输入机器学习模型。常用的字符串特征进行了汇总，包括字符串长度、元音字母与辅音字母比例、数字字符比例、字符的信息熵、字符的词频分布以及有意义的字符串占域名字符串总长度的比值，然后使用随机森林模型进行训练和判别。

这种方式存在两个比较严重的问题。首先，人工设计的特征很容易被规避，黑客很容易设计一类新的DGA家族规避掉已有的检测特征。其次，人工设计特征的方式非常费时费力，当有新的DGA家族产生时，就必须重新设计一套相应的特征。这些问题都极大的限制了域名字符串分析方式的实用性。

在DNS流量方面，第一步通常是对比正常域名的DNS流量和DGA域名DNS流量，从中找出具有区分度的特征，常见的特征有域名的访问频率、访问规律性、共享IP、平均TTL、NXDomain流量等。提出特征之后，便使用随机森林、聚类等算法去训练模型，用训练好的模型去检测恶意域名。

但是，这种方法常常需要去监控整个网络，通过一段时间的信息收集整理之后才能获取到域名的各项特征，进而才能检测一个域名是否是恶意的。这种方式不仅难以实现，而且无法满足实时性的需求，无法在恶意域名出现的时刻就对其进行检测与拦截，也就无法有效的抑制恶意攻击的产生。

## 最新进展

2016年，论文Predicting domain generation algorithms with long short-term memory networks首次实现了使用深度学习方法进行DGA域名检测。在数据处理方面，该论文参考自然语言处理方面的研究内容，将域名字符串转换成了词向量。在模型构建方面，该论文使用了RNN的变种LSTM模型，该模型对于处理序列输入问题有很好的效果。

使用深度学习进行DGA域名检测不仅节省了人工设计特征所需要的开销，同时能够对域名进行实时检测，并且在准确率和泛化性上有了质的提升，对网络安全有着重大的促进意义。

# 数据与模型

## 数据来源

在本实验中，正常域名来自于Alexa网站的域名点击排行，总共收集了100万个正常域名。 DGA域名则来自于360网络实验室和osint公开的DGA域名种子网站，它们都属于公开的数据源。其中，从360网络实验室收集了34类不同家族的DGA域名，总共136万个DGA域名。从osint网站上收集了44类不同家族的DGA域名，总共144万个DGA域名。

## 数据集构造

为了增加实验结果的可信度，本实验从两个不同的来源收集了DGA域名。因此，也将构造两个数据集用于实验评估。

其中，第一个数据集由100万个正常域名加上来自于360网络实验室收集的34类不同家族的136万个DGA域名构成，第二个数据集由100万个正常域名加上来自于osint公开的DGA域名种子网站收集的44类不同家族的144万个DGA域名构成。

对于每一个数据集，为了使数据分布的更均匀些，本论文都从其中的正常域名以及每一类DGA家族域名中各随机抽取80%的域名样本作为训练集，剩下20%的域名样本作为测试集，总共构造了两组训练集与测试集，训练集的组成如表2.1所示。

**表 2.1 数据集构成**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 训练集 | | | 测试集 | | |
| 正常域名 | DGA域名 | DGA种类 | 正常域名 | DGA域名 | DGA种类 |
| 360数据集 | 800,000个 | 1090,000个 | 34类 | 200,000个 | 270,000个 | 34类 |
| osint数据集 | 800,000个 | 1150,000个 | 44类 | 200,000个 | 290,000个 | 44类 |

## 数据处理

在数据处理阶段，本实验首先根据单词列表将域名字符串转换成了一维整数矩阵，这种简单的线性转换并不能很好的体现出原始域名字符串中字符之间的关联信息，这也是自然语言处理中最基本的问题之一。因此，本实验参考自然语言处理方面的处理方法，对一维矩阵进一步处理，将其转换成携带更多原始数据信息的二维词向量，最终再输入到模型中。

为了降低实验的复杂度，提高可复现性，在生成词向量的过程中，本实验没有单独设计并训练一个语言模型，然后从中获取生成词向量所需要的参数矩阵，而是借助了Keras库中的Embedding层来生成词向量。

Embedding层的数据处理的流程如图2.1所示。

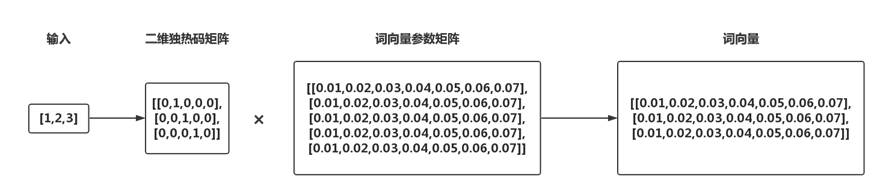


图2.1 Embedding层数据处理流程

为了方便显示，图2.1只简单的展示了一个3维整数数组转换成(3,7)的二维词向量的流程。

## 模型设计

在论文Predicting domain generation algorithms with long short-term memory networks中使用的LSTM网络的基础上，本实验加入最近比较流行的Attenton机制，以提高模型的性能。

原始LSTM网络的结构如图2.2所示，要由LSTM层、Dropout层、全连接层和激活层构成。

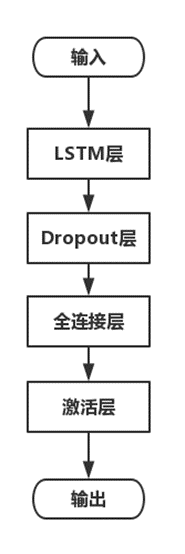


图2.2 LSTM网络结构

加入Attention机制后的网络结构如图2.3所示，其中Attention模块是权重计算模块，该模块接收LSTM层输出的二维特征矩阵作为输入，在模块内部经过学习计算产生一个权重矩阵，然后将输入的特征矩阵与计算得到的权重矩阵融合后输出。经过这一系列的运算处理，该模块的输出中不仅包含了词向量的特征信息，同时也包含了各个维度特征的权重信息，为后续分类模块提供了更多的参考信息。

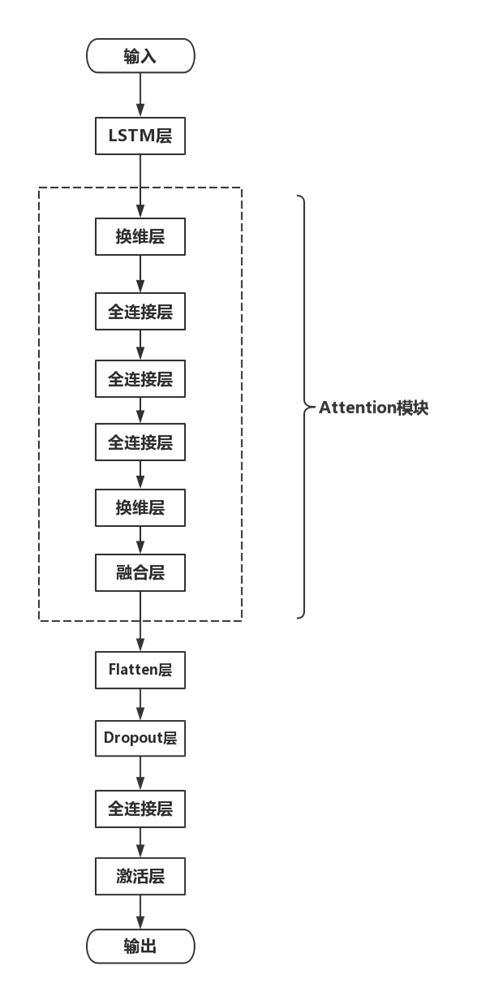
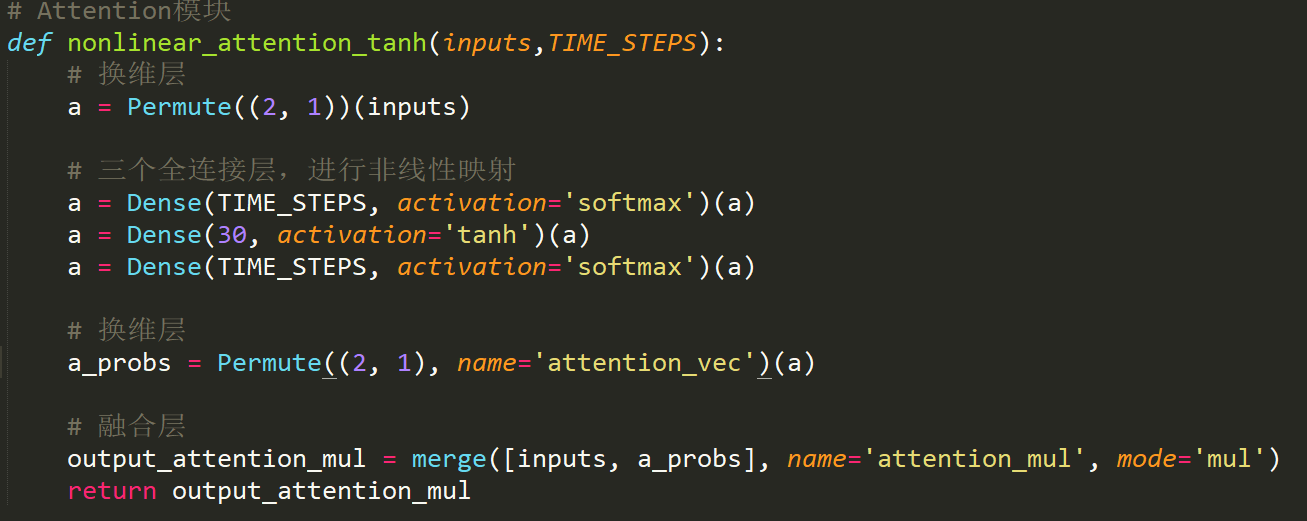


图2.3 LSTM+Attention网络结构

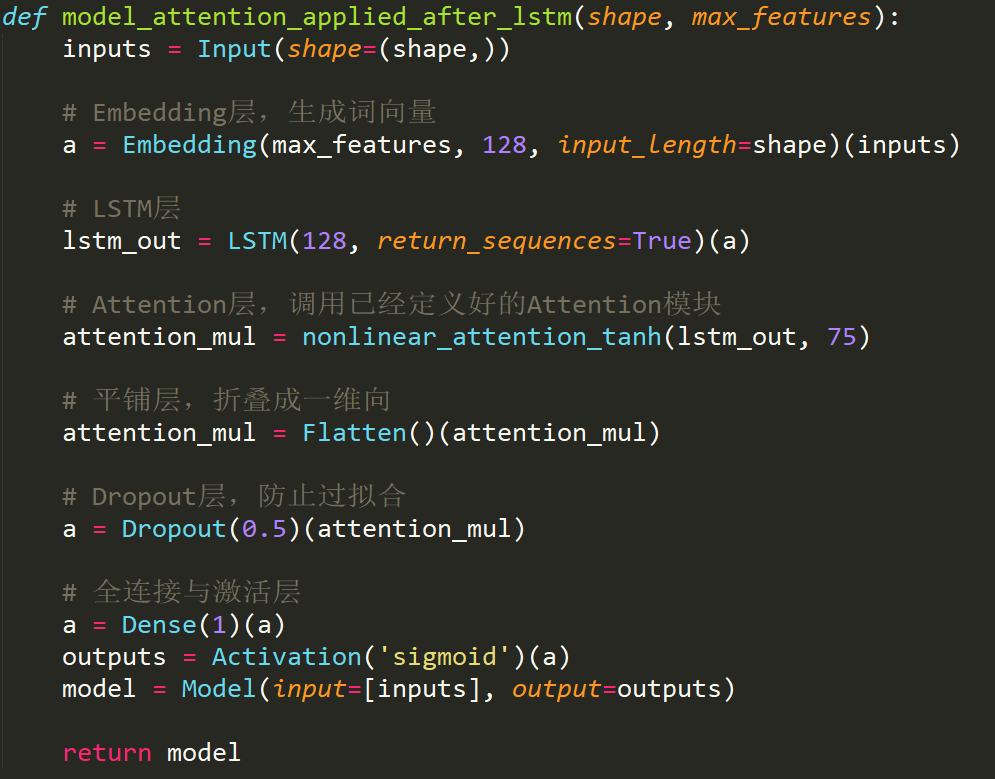
# 代码展示

本实验实验的是Keras框架，每部分的代码功能见注释。

## Attention模块代码



## 整体模型代码



## 训练与预测部分代码



# 实验结果

本实验分别在两个数据集上进行多次重复训练和测试，并统计测试结果的平均值，如表4.1所示。

**表 4.1测试结果**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据集 | Precision | Recall | score | 误报率 | 漏报率 |
| 360数据集 | 0.9891 | 0.9918 | 0.9905 | 0.0148 | 0.0082 |
| osint数据集 | 0.9940 | 0.9915 | 0.9928 | 0.0086 | 0.0085 |