基于注意力特征融合网络的 DGA 恶意域名检测方法

郝旭光
（山西省政务和公益域名注册管理中心，山西　太原　030024）

摘　要：僵尸网络借助 DGA 生成大量随机域名逃避安全防御系统监测。为解决已有 DGA 恶意域名检测方法准确性不高和泛化能力受限等问题，提出基于注意力特征融合网络。通过结合输入层、Embedding 层、卷积神经网络层、注意力模块和长短时记忆网络层，实现层次化特征提取使模型性能得到极大的改善。实验结果显示，该方法在各项指标上都有明显的提升，表现出优秀的 DGA 恶意域名检测能力。

关键词：DGA 域名；注意力机制；神经网络
中图分类号：TP393.4　　**文献标识码：**A　　**DOI：**10.19358/j.issn.2097－1788.2024.01.003
引用格式：郝旭光. 基于注意力特征融合网络的 DGA 恶意域名检测方法［J］. 网络安全与数据治理，2024，43(1)：19－27.

A DGA malicious domain detection method based on attention feature fusion network

Hao Xuguang
(Shanxi Organizational Name Administration Center, Taiyuan 030024, China)

Abstract: Botnets employ Domain Generation Algorithms (DGA) to generate numerous random domain names to evade detection by the security defense system. In order to solve the problems of low accuracy and limited generalization capabilities, this article proposes attentional feature fusion network. This model combines an input layer, an Embedding layer, a Convolutional Neural Network layer, an attention module, and a Long Short-Term Memory layer, achieving hierarchical feature extraction and substantially improving model's performance. Experimental results indicate that the approach exhibits significant improvements in various indicators, showcasing outstanding DGA malicious domain name detection capabilities.

Key words: DGA domain; attention mechanism; neural network

0　引言

域名服务系统（Domain Name System, DNS）是互联网最基础的应用系统，通过建立域名和 IP 地址的对应关系支撑服务其他业务应用，但其开放性和公平性也被恶意软件利用。僵尸网络借助域名生成算法（Domain Generation Algorithm, DGA）大量生成 DGA 域名，通过命令与控制（Command-and-Control, C&C）服务器操控受害者主机，达到逃避安全监控、提高生存和攻击能力的目的，从而进行大规模的分布式拒绝服务攻击、发送垃圾邮件、传播非法信息和钓鱼网站、运行勒索软件等恶意活动。其复杂性和隐蔽性导致传统的网络安全防御手段难以有效应对，追踪控制服务器位置变得更加困难。

如何高效检测和拦截 DGA 域名，是近年来网络安全防护技术研究的热点方向。纵观当前 DGA 恶意域名的检测方法主要包括基于特征提取的机器学习方法检测、基于无特征提取的深度学习方法检测和基于附加条件的深度学习方法检测[1]。基于特征提取的机器学习方法优势在于可以利用常见特征实现高效检测，比如借助于人工提取的诸如域名长度、元辅音占比、字符频率等，以及 DNS 请求和响应的频率、时序和地理分布等特征，使用分类器进行域名分类实现快速检测。基于无特征提取的深度学习方法借助深度学习的自动特征学习能力，既能缓解对人工提取特征的过度依赖又能发现传统统计方法无法发现的特征，很大程度上解决了特征检测法实时性差和易被绕开的缺点，提高了 DGA 恶意域名检测的准确性。基于附加条件的深度学习方法添加了某种附加条件以提高检测准确率，例如将注意力集中在域中更重要的子串并改善域的表达，增加域名的多字符随机性提取方法，通过词法分析和 Web 搜索来估计域名随机性等措施，以提高模型的检测性能，特别是针对新型 DGA 域名的

检测。

以上方法虽然在一定范围内取得了效果，但为了提高生存率，DGA 算法也在不断更新迭代，导致现有检测方法逐步失效。特征法依赖人工提取字符和流量特征，易受到复杂网络环境的干扰，攻击者可以重新设计 DGA 生成算法绕过检测，导致此类方法在面对新型 DGA 域名时，泛化能力和准确率受限。深度学习法在遇到如数据量少的 DGA 域名家族、新型 DGA 域名时，无法捕捉到某些关键信息，且易受到精心设计的对抗样本的欺骗，在应对更加智能的 DGA 域名上的表现不佳。附加条件法在不同的附加机制中，针对一些如短域名、高可读性域名存在误判和表现效果不佳的现象。

本文提出了一种注意力特征融合网络。通过 Embedding 层、卷积神经网络（Convolutional Neural Network，CNN）层、注意力模块和长短时记忆（Long Short Term Memory，LSTM）网络层集合了各种检测方法的优势和长处，显著提升了对 DGA 域名检测的能力。首先，Embedding 层使得网络能够学习输入数据的稠密向量表示，从而捕捉更丰富的信息。其次，CNN 层和 LSTM 网络层的组合实现了层次化特征提取，前者负责提取局部特征，后者捕捉长期依赖关系，增强了模型的泛化能力。第三，注意力模块的引入有助于关注域名字符间重要的局部特征，进一步解决长距离依赖关系难以捕捉的问题。实验表明，使用本文方法检测 DGA 域名，在准确率、精确率、召回率和综合性能上都有着明显的提升。

1　DGA 域名及其特征

1.1　DGA 域名

DGA 域名指通过 DGA 算法自动生成的域名，通常依赖于一个种子值（如当前日期、特定数值或者内置种子等）和一个预定义的算法。僵尸网络客户端通过 DGA 算法生成大量域名，并且进行查询，攻击者在控制端运行同一套 DGA 算法，生成相同的备选域名列表。当需要发动攻击的时候，从列表中选择少量的域名注册开通便可以建立通信，同时可以利用 IP 速变技术，实现 IP 和域名快速变化隐藏 C&C 服务器，逃避网络安全设备的监测跟踪，为僵尸网络提供一个持续且难以被追踪的通信连接。

以著名的 Conficker 僵尸网络为例，其 A/B 变种的 DGA 算法基于当前日期作为种子值，每天生成 250 个 . com 域名和 250 个 . net 域名。具体的，其使用一个基于时间的种子值，对 26 个字母进行置换，生成长度不同的域名。而 CryptoLocker 是一种勒索软件，其基于种子值和日期，通过一系列的数学运算和映射，生成不同的域名。Banjori 是一种恶意软件，其 DGA 具有多元递归关系，每次都会根据前一个域名生成下一个域名，用加减和取模运算得到下一个域名的前四个字母，同时保持后缀不变。表 1 展示了这三种恶意软件中使用 DGA 所生产的恶意域名以及部分正常的域名。

表 1　DGA 域名与正常域名示例

类别	域名
Conficker	spsiohj. info
	jszsbbev. biz
	hiecgl. org
	jnsdngcgso. cn
CryptoLocker	ugyrssqdfiec. com
	pmylvupllrmcx. ru
	ynqfwjnleebbpjt. com
	awpugolhvciitt. org
Banjori	goyyancorml. com
	xxscsikathrinezad. com
	fjkupartbulkyf. com
	blssbyplaywobb. com
正常域名	google. com
	youtube. com
	baidu. com
	bing. com

1.2　DGA 域名特征

DGA 域名由算法自动生成，无需人工干预。其具有语义不明确、结构不均衡、长度变化大、存活时间短等特征。可以将 DGA 域名与正常域名的差异总结为以下三点：

（1）语义性。正常域名通常具有较强的语义性，通常是为了表示实际的公司、组织或产品而创建的。正常域名往往包含有意义的单词、缩写或短语，以便用户能够轻松地识别和记住。相反，DGA 域名通常缺乏语义性，因为由算法自动生成，目的是让其难以被预测和追踪。

（2）结构和可读性。正常域名通常具有较好的结构和可读性，字符分布较为均衡，可能包含辅音和元音的组合，以及一定比例的数字和特殊字符。而 DGA 域名的结构和可读性通常较差，字符分布可能不均衡，字符组合可能显得更加随机和无规律。

（3）域名长度。正常域名的长度通常在一定范围内变化，具有较短的平均长度。而 DGA 域名的长度可能有很大差异，根据所使用的生成算法，长度可能非常短或

非常长。不过部分 DGA 可能会生成较短的域名，以模仿正常域名的外观。

虽然这三个方面的差异能够帮助区分正常域名与 DGA 域名，但也正因为 DGA 域名无语义、无规律的特点导致一般方法难以有效检测。

2 相关工作

基于 DGA 域名的特征，通过分析其不同的生成算法，研究者设计提出了不同的应对思路和检测方法，归纳起来主要为以下三类。

2.1 基于特征提取的机器学习方法的检测

利用合法域名与 DGA 域名在字符组合上的差异，Ma 等人[2]提出了一种轻量化的方法来检测 DGA 域名。该方法利用 URL 的词法构造特征，如 URL 的长度、中英文句号的数量和特殊字符的数量等判定 DGA 域名。Wang 和 Shirley[3]使用词语分割从域名中提取标记来检测恶意域名。所提出的特征空间包括字符数、数字和连字符的数量等。胡鹏程等人[4]从域名中提取了包括随机性、可读性、数字与字母分布情况、顶级域名、域名长度在内的多个特征，并使用机器学习算法进行测试。王红凯等人[5]通过人工提取域名长度、字符信息熵、多类字符比例等特征，使用随机森林实现 DGA 域名的检测。Agyepong 等人[6]则通过人工提取的 KL 散度、Jaccard 系数等特征用于训练模型完成检测。

通过网络流量分析并结合上下文特征，韩春雨等人[7]提出了一种基于 DNS 流量的 Fast-flux 域名检测方法，利用 DNS 流量中的域名语言特征和统计特征来区分 Fast-flux 域名和正常域名，并使用机器学习模型进行分类。其也引入了量化的地理广度、国家向量表和时间向量表特征，以加强对 Fast-flux 域名检测的针对性。Manasrah 等人[8]提出了一种基于 DNS 流量挖掘的 DGA 域名检测方法。该方法使用了多个相关的语言特征，如随机度、稀有度、打字难度等来衡量域名的特征，在不同类型的 DGA 域名上实现了高准确率和低误报率。Wang 等人[9]利用 DNS 流量中的域名统计特征和时间序列特征来区分 DGA 域名和正常域名，并使用聚类算法来划分不同类型的 DGA 域名。该方法使用如域名长度、元音比例、熵、请求频率和持续时间等特征来描述域名的特征，并在多种 DGA 家族上实现了良好的检测效果。Antonakakis 等人[10]提出称为 Pleiades 的检测系统。通过提取与 NXDOMAIN 字符串相关的统计特征，包括 n 元分布和字符频率，并使用机器学习算法将 NXDOMAIN 字符串分成 DGA 生成和合法两类，在大规模的 DNS 流量测试中表现出高检测率和低误报率。Silveira 等人[11]提出了一种使用被动 DNS 自动检测恶意域名的方法，从 DNS 流量中提取了 12 类不同的特征，并使用 XGboost 算法对特征进行学习，在数据集上的 AUC 达到了 0.976。

2.2 基于无特征提取的深度学习方法的检测

Highnam 等人[12]提出了一种混合神经网络 Bilbo，用于分析域名并评分其由字典 DGA 生成的可能性。该模型在跨不同字典 DGA 分类任务的泛化性能方面，在 AUC、F1 分数和准确性方面都能取得较好的成绩。Kumar 等人[13]提出了一种基于深度神经网络的增强 DGA 检测模型，该模型结合了额外提取的人工特征以及由深度学习模型提取的特征，在 DGA 域名分类方面的性能优于 SVM、RF 等现有方法。Yu 等人[14]通过将 LSTM 和 CNN 用于 DGA 域名检测，证明了深度学习方法相比于如随机森林等机器学习方法在检测时性能上的优越性。但其同时也发现，传统深度学习方法的性能容易受到数据不平衡的影响，导致在样本较少的 DGA 域名家族上的检测效果较差。申宋彦[15]通过卷积神经网络分别提取域名中的字符特征和词特征，并通过改进的卷积神经网络实现了对难度较大的恶意域名家族的识别效果提升。Vinayakumar 等人[16]比较了 RNN、CNN、LSTM 等深度学习方法在检测 DAG 恶意域名时的性能，发现递归神经网络的架构能够有效增强深度学习模型的整体检测能力。还有部分工作[17-19]采用了不同架构的 RNN 来检测恶意域名，包括门控循环单元（Gated Recurrent Units，GRU）和双向循环神经网络（Bi-directional Long Short-term Memory，Bi-LSTM）等，但这些方法在检测随机性较高的 DGA 域名时无法很好地捕捉到字符之间的序列关系，识别率较低。此外，生成对抗网络（Generative Adversarial Network，GAN）得益于其建立在博弈论上优秀的网络训练机制，在 DGA 域名识别任务中也得到了使用。如袁辰等人[20]和 Anderson 等人[21]通过在生成网络中不断生成真实度更高的恶意域名，同时在判别网络中对生成的恶意域名进行检测，使得判别网络的识别能力不断提高。

2.3 基于附加条件的深度学习方法的检测

随着 DGA 域名的算法越来越智能化，采用基于特征和深度学习的检测方法愈感力不从心。研究者又在此基础上增加一些附加条件来达到提高检测率的目标。Chen 等人[22]提出了一个结合注意机制的 LSTM 模型，将注意力集中在域中更重要的子串并改善域的表达，达到了更好的性能，在二元分类中，其误报率和假阴率分别低至 1.29% 和 0.76%。陈立皇等[23]也提出了一种基于注意力机制的深度学习模型，不同的是，他们采用一种域名的多字符随机性提取方法，提升了识别低随机 DGA 域名的有效性。Satoh 等[24]通过词法分析和 Web 搜索来估计域

名随机性，但该方法对域名长度较短时，无法区分，不包含在字典中的域名会被误判。

为了逃避神经网络的检测，恶意域名已升级为多个单词的组合。为此，Curtin 等[25]提出了用 smash 分数来评估 DGA 域名与英文单词的相似程度，并设计了递归神经网络架构与域注册信息的组合模型。虽然实验在对 matsnu 和 suppobox 像自然域名的家族的检测效果好，但是在那些不像自然域名 DGA 系列表现效果欠佳。

综合分析以上相关 DGA 域名检测方法，各种模型算法面对不同的 DGA 家族在一定时期达到了较高的检测准确率和较好的网络防御效果，但在面对不断升级的 DGA 算法和一些特殊的结构设计还存在着漏检和误检的情况。随着新技术的发展，特别是恶意算法对新技术的综合运用使得恶意域名特征更加难以捕捉，需要综合利用各种检测方法的优势，提升检测范围的覆盖率、准确率。因此，本文集合深度学习模型的优点，引入注意力模块，提高了 DGA 域名的检测能力。

3　注意力特征融合网络

本文所提出的注意力特征融合网络模型结构如图 1 所示，包括输入层、Embedding 层、卷积网络层、注意力模块、长短期记忆网络层和输出层。功能分别为：（1）输入层：负责接收原始的域名数据，作为神经网络的起始输入；（2）Embedding 层：负责将输入的离散域名字符映射为稠密向量表示，以便更好地捕捉字符间的相关性；（3）卷积网络层：负责提取域名序列中的局部特征，如字符的组合模式，有助于识别 DGA 恶意域名中的模式；（4）注意力模块：紧接在卷积网络层之后，负责在处理域名的局部特征时关注更具判别力的局部特征，以提高恶意域名检测的准确性；（5）长短期记忆网络层：依据得到的重要性不同的域名局部特征来捕捉域名序列中的长期依赖关系，以便更好地理解字符间的上下文关系；（6）输出层：负责将神经网络的预测结果转化为具体的分类标签，例如判断输入域名是正常域名还是 DGA 恶意域名。

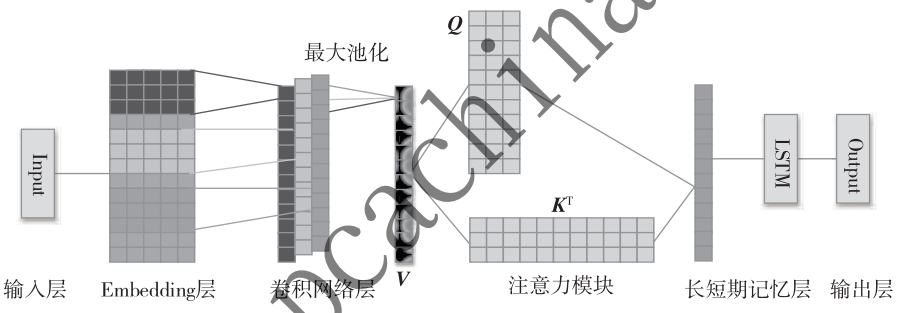

图 1　注意力特征融合网络整体结构

3.1　输入层

接收原始的域名数据，并将其转换为适合神经网络处理的格式。作为神经网络的起始部分，输入层对数据质量和格式的处理至关重要，因为它们会直接影响网络的学习效果和性能。

输入数据通常以域名序列的形式提供，每个域名由一系列字符组成，包括字母、数字和连字符等。为了使神经网络能够更好地处理这些离散字符，需要对输入数据进行预处理。预处理的主要步骤包括：（1）将域名转换为小写形式，以消除字符大小写的影响；（2）统一域名的长度，对较短的域名进行填充或截断较长的域名，以确保输入具有相同的维度；（3）将离散字符映射到整数编码，以便神经网络能够处理这些数据。

经过预处理后，输出的数据为整数编码的域名序列。例如，给定一个原始域名“example. com”，经过预处理后，输入层可能输出一个整数序列，如［5，24，1，13，16，12，5，28，3，15，13］。整数编码的序列可以被后续的神经网络层（如 Embedding 层）接收并处理，进一步提取有助于 DGA 恶意域名检测的特征。本文所使用的域名数据中，域名长度集中分布情况如图 2 所示。可见，域名长度分布在 4 ~ 73 之间，且集中分布在 8 ~ 30 之间，因此在进行预处理时，将所有域名长度超过 32 的部分进行截断，而对长度不足 32 的域名，则对其序列化后的表示进行补零，使得所有输入的序列长度都为 32。

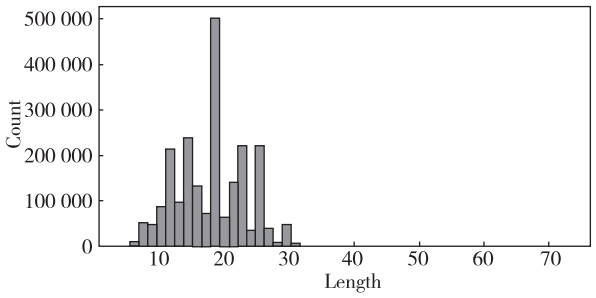

图 2　域名长度分布图

3.2 Embedding 层

Embedding 层将输入层提供的整数编码域名序列转换为稠密向量表示，帮助神经网络更好地捕捉输入数据中的相关性和特征，从而提高整体性能。具体的，是将每个整数编码的字符映射到一个固定长度的连续向量空间。映射过程可以理解为一个查找表操作，其中每个整数编码都对应一个预先定义的向量。在训练过程中，Embedding 层会通过反向传播算法更新这些向量，使其能够更好地捕捉字符之间的相关性。因为本文将以域名中的每个字符作为处理对象，所以 Embedding 的维度为 $\boldsymbol{R}^{v\times e}$，其中 v 指的是 vocabsize，即出现的所有字符的数量，而 e 指的是 embeddingsize，即每条字符向量的长度。每条域名在经过 embedding 层的映射之后，其维度会变成 $\boldsymbol{R}^{i\times e}$，其中 i 指的是 inputsize，即输入域名的长度。因为在预处理中将域名的长度都对齐为 32，所以 inputsize 为 32。

3.3 卷积网络层

卷积网络层负责提取输入序列中的局部特征。通过卷积操作，该层能够捕捉字符之间的邻近关系，从而识别 DGA 恶意域名中的特定模式。卷积操作可以被表示为一个滑动窗口在输入矩阵上按照一定的步长进行扫描。具体而言，给定一个输入矩阵 $\boldsymbol{X}$，一个卷积核 $\boldsymbol{T}$ 和一个偏置 $\boldsymbol{b}$，卷积操作可以通过下式计算：

$$Y_{ij} = \sum_{m}\sum_{n} X_{i+m,j+n}\cdot T_{mn} + \boldsymbol{b} \tag{1}$$

其中，Y_{ij} 是输出矩阵 $\boldsymbol{Y}$ 的第（i，j）个元素，（m，n）是卷积核 $\boldsymbol{T}$ 的索引。通过遍历输入矩阵上的所有可能位置，可以计算出完整的输出矩阵 $\boldsymbol{Y}$。在本层中采用了一维卷积（1D-CNN），因为这种形式的卷积能够更好地处理序列数据。具体的，一维卷积只沿着域名序列的长度方向进行，从而能有效地捕捉字符之间的局部模式。在模型中同时使用了大小为 3 的多个卷积核，以实现对多种局部特征的提取，从而增强模型的表征能力。在卷积层后，网络还使用了最大池化层来降低模型的参数，并去除作用不显著的冗余信息。

3.4 注意力模块

为输入序列中的每个元素分配不同的权重，以便在处理序列数据时关注更具判别力的部分。通过注意力机制，神经网络能够更好地捕捉长距离依赖关系，提高恶意域名检测的准确性。本模型中所使用的注意力模块采用自注意力机制（Self-Attention），其计算过程可以分为三个步骤：（1）计算查询（Query）、键（Key）和值（Value）矩阵；（2）计算注意力分数；（3）计算加权值和。假设输入矩阵 $\boldsymbol{X}$ 的维度为（t，d），其中 t 是序列长度，d 是特征维度。首先，计算查询矩阵 $\boldsymbol{Q}$、键矩阵 $\boldsymbol{K}$ 和值矩阵 $\boldsymbol{V}$：

$$\boldsymbol{Q}=\boldsymbol{X}\boldsymbol{W}_{Q},\ \boldsymbol{K}=\boldsymbol{X}\boldsymbol{W}_{K},\ \boldsymbol{V}=\boldsymbol{X}\boldsymbol{W}_{V} \tag{2}$$

其中，$\boldsymbol{W}_{Q}$、$\boldsymbol{W}_{K}$ 和 $\boldsymbol{W}_{V}$ 分别为查询、键和值矩阵的权重矩阵。然后，需要计算注意力分数。在自注意力机制中，注意力分数由查询矩阵 $\boldsymbol{Q}$ 和键矩阵 $\boldsymbol{K}$ 的点积得出，并通过缩放因子$\frac{1}{\sqrt{d_k}}$进行缩放，其中d_k是键矩阵的维度。计算注意力分数的公式如下：

$$\boldsymbol{S}=\mathrm{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right) \tag{3}$$

最后，计算加权值和：

$$\boldsymbol{Y}=\boldsymbol{S}\boldsymbol{V} \tag{4}$$

此时，输出矩阵 $\boldsymbol{Y}$ 的维度与输入矩阵 $\boldsymbol{X}$ 相同，但元素的权重经过重新分配，使得网络更加关注重要部分。

3.5 长短期记忆层

负责处理序列数据中的长期依赖关系。LSTM 是一种特殊的递归神经网络，通过引入门控单元来解决传统 RNN 中的梯度消失和梯度爆炸问题。LSTM 单元包含三个门控单元：输入门（input gate）、遗忘门（forget gate）和输出门（output gate），以及一个单元状态（cell state）。给定一个输入向量 $\boldsymbol{x}_t$ 和前一时刻的隐藏状态 $\boldsymbol{h}_{t-1}$，LSTM 单元中输入门、遗忘门、输出门的计算过程分别如下所示：

$$\boldsymbol{i}_t=\sigma\left(\boldsymbol{W}_{ix}\boldsymbol{x}_t+\boldsymbol{W}_{ih}\boldsymbol{h}_{t-1}+\boldsymbol{b}_i\right) \tag{5}$$

$$\boldsymbol{f}_t=\sigma\left(\boldsymbol{W}_{fx}\boldsymbol{x}_t+\boldsymbol{W}_{fh}\boldsymbol{h}_{t-1}+\boldsymbol{b}_f\right) \tag{6}$$

$$\boldsymbol{o}_t=\sigma\left(\boldsymbol{W}_{ox}\boldsymbol{x}_t+\boldsymbol{W}_{oh}\boldsymbol{h}_{t-1}+\boldsymbol{b}_o\right) \tag{7}$$

进而单元状态更新的过程可以表示为：

$$\boldsymbol{c}_{\tilde{Y}}=\tanh\left(\boldsymbol{W}_{cx}\boldsymbol{x}_t+\boldsymbol{W}_{ch}\boldsymbol{h}_{t-1}+\boldsymbol{b}_c\right) \tag{8}$$

$$\boldsymbol{c}_t=\boldsymbol{f}_t\odot c_{t-1}+\boldsymbol{i}_t\odot \tag{9}$$

而隐藏状态更新的过程可以表示为：

$$\boldsymbol{h}_t=\boldsymbol{o}_t\odot\tanh\left(\boldsymbol{c}_t\right) \tag{10}$$

其中，σ（·）是 Sigmoid 激活函数，⊙表示按位乘法，$\boldsymbol{W}$ 和 $\boldsymbol{b}$ 是权重矩阵和偏置向量。长短期记忆层位于卷积网络层和注意力模块之后，以处理经过局部特征提取和注意力分配的序列数据。通过对序列中的字符进行长期依赖关系建模，LSTM 层有助于捕捉 DGA 域名中的潜在模式，从而提高整个网络的性能。

3.6 输出层

将提取的特征映射到目标任务的预测结果。完成两个任务，判断一个域名是否为 DGA 域名（二分类任务）和判断一个域名为正常或来自特定算法家族的域名（多分类任务）。

在进行二分类任务时，输出层只包含一个神经元，该神经元使用 Sigmoid 激活函数将最后一层的输出映射到

（0，1）区间，得到域名为 DGA 域名的概率：

$$P(\boldsymbol{y}=1 \mid \boldsymbol{x}) = \sigma(\boldsymbol{W}_h \boldsymbol{h}_t + \boldsymbol{b}_o) \tag{11}$$

其中，$\boldsymbol{W}_h$ 和 $\boldsymbol{b}_o$ 是输出层的权重矩阵和偏置向量，$\boldsymbol{h}_t$ 是 LSTM 层的最终隐藏状态。

在进行多分类任务时，输出层的神经元数量与类别的数量相等，该层使用 softmax 函数将输出层神经元的输出映射为概率分布：

$$P(y_i \mid \boldsymbol{x}) = \frac{e\boldsymbol{W}_{hi}\boldsymbol{h}_t + \boldsymbol{b}_{oi}}{\sum_{j=1}^{N} e\boldsymbol{W}_{hj}\boldsymbol{h}_t + \boldsymbol{b}_{oj}}, \quad i=1, \cdots, N \tag{12}$$

其中，$\boldsymbol{W}_{hi}$ 和 $\boldsymbol{b}_{oi}$ 分别表示输出层第 i 个神经元的权重和偏置。模型最终的预测结果就为概率最大的那一类。

4 实验与分析

4.1 实验数据集

本文所使用的数据集由公开的合法域名数据集和 DGA 域名数据集组合而成。其中合法数据集为 Alexa 统计的 100 万个互联网中访问流量最高的网站的域名。DGA 域名数据集为 360 Netlab 发布的 42 类 DGA 家族共 1 147 770条域名。进一步地，对完整的数据集进行分层采样，即在每个 DGA 家族以及正常域名内部按比例进行采样，然后将采样的数据合并为训练集、测试集和验证集，三者的占比为 7∶2∶1。

4.2 实验指标

在本研究中，选用了以下四项评估指标：平均准确率（Accuracy）、精确率（Precision）、召回率（Recall）及 F_1 值。其中平均准确率的计算方式为：

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FP}+\text{TN}+\text{FN}} \tag{13}$$

精确率的计算方式为：

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \tag{14}$$

召回率的计算方式为：

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \tag{15}$$

F_1 值的计算方式为：

$$\text{F}_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{16}$$

在二分类和多分类两种情况下，四项评估指标中 TP、FP、TN、FN 的含义为：

（1）真阳性（True Positive，TP）。二分类：正确预测为 DGA 域名的实例数量。多分类：将属于该 DGA 家族的域名成功预测为该 DGA 家族的实例数。

（2）假阳性（False Positive，FP）。二分类：将正常域名误判为 DGA 域名的实例数量。多分类：将其他 DGA 家族的域名或正常域名错误地归类到该 DGA 家族的实例数。

（3）真阴性（True Negative，TN）。二分类：正确预测为正常域名的实例数量。多分类：将其他 DGA 家族的域名和正常域名成功预测为非该 DGA 家族的实例数。

（4）假阴性（False Negative，FN）。二分类：将 DGA 域名误判为正常域名的实例数量。多分类：未能将该 DGA 家族的域名成功预测为该 DGA 家族的实例数。

4.3 实验环境与参数设置

软件方面，本文的实验在 Windows 10 系统下进行，使用的 Python 版本为 3. 10，使用的深度学习库 TensorFlow 版本为 2. 10。硬件方面，实验设备的内存大小为 16 GB，CPU 为 Intel® 倕 Core™ i7-8700K。实验中的各项参数设置如表 2 所示。

表 2 实验所使用的具体参数

参数	设置
学习率	0. 005
训练 batchsize	512
训练 epoch	12
损失函数	交叉熵
优化器	Adam
embeddingsize	256
输入域名长度	32
长短期记忆层隐藏节点数	256
卷积网络层卷积核大小	3
卷积网络层隐藏节点数	256

4.4 实验结果

实验对比了传统的深度学习网络 CNN 和 LSTM 与本文所提出的自注意力特征融合网络在检测 DGA 域名上的效果。三种网络进行二分类任务时的结果如表 3 所示。

表 3 不同方法在判断域名是否为 DGA 域名时的效果

Method	Class	Precision	Recall	F_1	Accuracy/%
CNN	Benign	0. 994 8	0. 994 2	0. 994 5	99. 42
	DGA	0. 994 9	0. 995 5	0. 995 2	99. 55
	Average	0. 994 8	0. 994 9	0. 994 9	99. 49
LSTM	Benign	0. 995 1	0. 977 5	0. 986 2	97. 75
	DGA	0. 980 7	0. 995 8	0. 988 2	99. 58
	Average	0. 987 4	0. 987 3	0. 987 2	98. 73
本文方法	Benign	0. 994 0	0. 996 3	0. 995 1	99. 63
	DGA	0. 996 8	0. 994 8	0. 995 8	99. 48
	Average	**0. 995 6**	**0. 995 4**	**0. 995 5**	**99. 55**

实验结果表明，本文所提出的方法在所有评价指标上都取得了较高的分数，而且在大多数情况下超过了CNN和LSTM；在平均精度、平均召回率、平均F_1分数和平均准确率方面，表现优于其他两种方法；且在识别DGA域名的精度上相比于CNN和LSTM都有较大提升。尽管LSTM在识别DGA域名的召回率上达到了最高(0.995 8)，但在识别正常域名的召回率和精度上，LSTM的表现不如其他两种方法，导致其平均表现稍弱。结合前文对DGA域名与正常域名在形式上的差异分析，可以说明本文方法所采用的网络结构能够更好地学习到域名字符序列中局部特征与长距离依赖，有利于当域名中存在大量随机与无规律字符时学习到更加准确的域名表征，从而实现检测效果的提升。

为了直观体现不同架构的网络在泛化能力与学习能力上的差异，进一步绘制了二分类情况下训练过程中不同网络结构的学习曲线，如图3～图5所示。

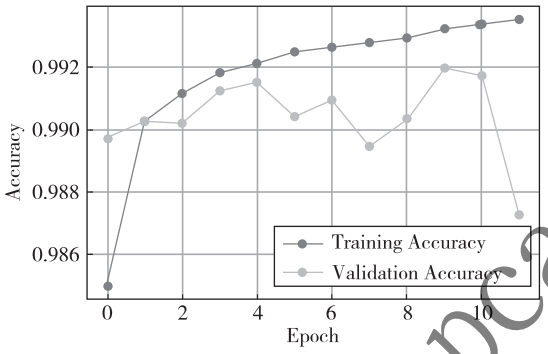

图3　CNN在二分类情况下训练时的学习曲线

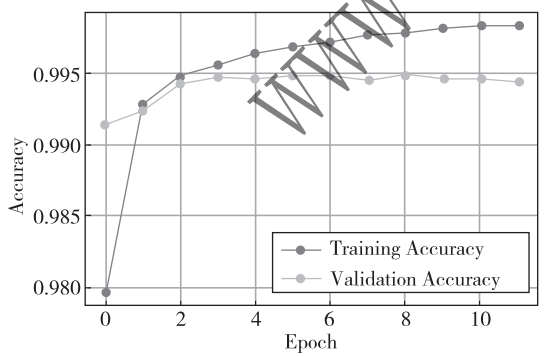

图4　LSTM在二分类情况下训练时的学习曲线

可见，随着训练的进行，LSTM在验证集上的准确率与在训练集上的准确率差距逐渐增大，且在第12轮训练时模型在验证集上的准确率已经开始有了下降的趋势，证明网络的性能不仅达到了上限，还即将过拟合，这也说明网络的泛化能力有限，不能较好地学习到域名序列间字符的局部关系。而CNN随着训练的进行，其在验证集上的准确率出现了大幅度的波动与下降，说明模型的泛化能力较差，且出现了较严重的过拟合现象，也说明模型无法学习到字符间的长期依赖关系将导致性能有着较大地下降。而本文所提出的方法随着训练的进行，其在训练集和验证集上的准确率变化都十分平稳，且在验证集上的准确率没有发生下降，说明本模型有着更好的学习能力，能够有效对域名序列中的特征进行学习。

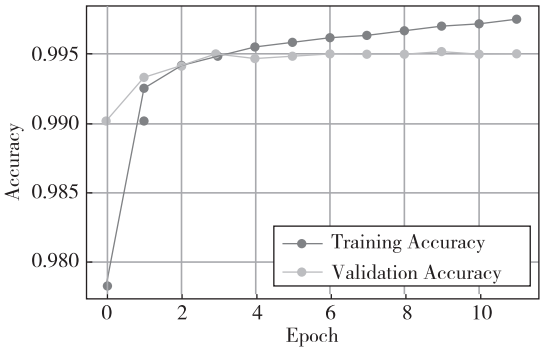

图5　本文方法在二分类情况下训练时的学习曲线

三种网络进行多分类任务时的效果如表4所示。需要说明的是，数据集在去除样本个数少于2的DGA类别后，共有38个不同的DGA类以及1个正常类。其中，表格中在对每一类域名分类时效果最好的方法的结果进行了加粗显示。

表4　不同方法在判断域名所属具体类别时的效果

Class	F_1			Class	F_1		
	CNN	LSTM	本文方法		CNN	LSTM	本文方法
Benign	0.99	0.99	**1.00**	Proslikefan	0.00	0.00	**0.18**
Bamital	**1.00**	**1.00**	**1.00**	Pykspa_v1	**1.00**	0.98	**1.00**
Banjori	**1.00**	**1.00**	**1.00**	Pykspa_v2_fake	**0.50**	0.29	0.33
Chinad	0.97	0.78	**0.99**	Pykspa_v2_real	0.08	0.05	**0.12**
Conficker	0.29	0.24	**0.31**	Qadars	**0.99**	0.98	0.98
CryptoLocker	**0.48**	0.37	0.42	Ramnit	0.82	0.65	**0.83**
Dircrypt	0.35	0.29	**0.41**	Ranbyus	0.89	0.79	**0.90**
Dyre	**1.00**	**1.00**	**1.00**	Rovnix	**1.00**	0.99	**1.00**
Emotet	**1.00**	**1.00**	**1.00**	Shifu	0.95	0.82	**0.96**
Fobber_v1	0.44	0.06	**0.45**	Simda	0.99	0.99	**1.00**
Fobber_v2	0.32	0.06	**0.35**	Suppobox	0.88	0.85	**0.95**
Gameover	**1.00**	0.96	**1.00**	Symmi	**0.99**	**0.99**	**0.99**
Gspy	**1.00**	0.95	**1.00**	Tempedreve	0.24	0.00	**0.27**
Locky	0.48	0.34	**0.50**	Tinba	**0.99**	0.92	**0.99**
Matsnu	0.06	0.00	**0.20**	Tinynuke	**1.00**	0.71	**1.00**
Murofet	0.90	0.78	**0.91**	Tofsee	**1.00**	**1.00**	**1.00**
Necurs	0.85	0.77	**0.87**	Vawtrak	**0.77**	0.75	0.72
Nymaim	**0.28**	0.22	0.19	Vidro	0.21	0.25	**0.26**
Omexo	**1.00**	0.77	**1.00**	Virut	0.78	0.61	**0.79**
Padcrypt	0.94	0.92	**0.99**	Average	0.990	0.981	**0.992**

通过对数据分析，发现本文所提出的方法在所有 39 类域名中的 21 类取得了最好的识别效果，在其中的 13 类上与其他方法同时取得了最好的效果，仅仅在其中 5 类上的效果落后于其他模型，由此可见本网络在同时检测多类 DGA 域名时的有效性。结果中的平均F_1值为根据各方法在各类域名上的F_1值与该类域名在数据集中的占比进行加权平均得来的，可以发现本文方法取得了最好的效果，以此也说明了本文方法在学习域名字符间的局部特征以及长期依赖关系上的有效性。特别需要注意的是，本文所提出的方法不仅能够在 CNN 与 LSTM 已有较好检测效果的特定 DGA 家族（如 Necurs、Suppobox 和 Padcrypt）上实现进一步的效果提升，还能够对 CNN 与 LSTM 几乎无法检测的特定 DGA 家族（如 Proslikefan、Matsnu）实现检测，说明本文方法不仅有着更好的泛化性能，还能够学习到传统网络无法学习到的特征。

更进一步地，绘制在多分类情况下不同网络结构的学习曲线，如图 6 ~ 图 8 所示。

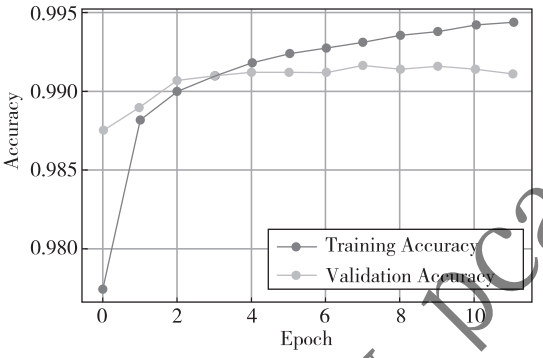

图 6　CNN 在多分类情况下训练时的学习曲线

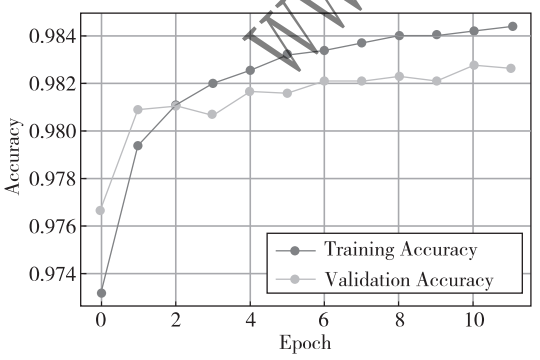

图 7　LSTM 在多分类情况下训练时的学习曲线

可见，在多分类情况下，随着训练的进行 CNN 在验证集上的准确性首先平缓上升，然后出现了下降的趋势，说明模型即将过拟合。而 LSTM 随着训练的进行，在验证集上的准确率上升并不平缓，而是有所波动。虽然其在训练终止时并没有出现过拟合现象，但因其收敛速度过于缓慢，其在验证集上的结果始终都低于另外两个模型。而本文方法兼具收敛快速与泛化能力强的特点，使其随着训练的进行，在验证集上的准确率稳步提升，且没有出现过拟合的现象。

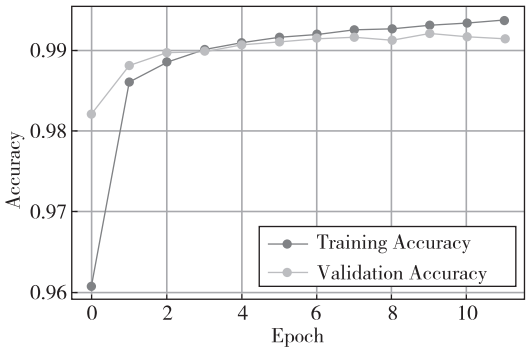

图 8　本文方法在多分类情况下训练时的学习曲线

5　结论

本文针对网络安全领域中的 DGA 恶意域名检测问题，提出了一种基于注意力机制的特征融合网络。该方法结合了 Embedding 层、卷积神经网络层、注意力模块和长短时记忆网络层，旨在实现更精确和高效的域名分类。实验结果表明，所提出的方法在各项评价指标上均优于传统深度学习方法，具有较强的泛化能力。未来研究将进一步探讨更高效的神经网络结构、结合多源信息以及在实际网络环境中的部署和应用，以实现更全面和实时的 DGA 恶意域名检测。

参考文献

[1] 王媛媛，吴春江，刘启和，等. 恶意域名检测研究与应用综述［J］. 计算机应用与软件，2019，36（9）：310 - 316.

[2] MA J，SAUL L K，SAVAGE S，et al. Beyond blacklists：learning to detect malicious web sites from suspicious URLs［C］// Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining，2009：1245 - 1254.

[3] WANG W，SHIRLEY K. Breaking bad：detecting malicious domains using word segmentation［J］. arXiv preprint arXiv：1506.04111，2015.

[4] 胡鹏程，刁力力，叶桦，等. 基于人工特征与深度特征的 DGA 域名检测算法［J］. 计算机科学，2020，47（9）：311 - 317.

[5] 王红凯，张旭东，杨维永，等. 基于随机森林的 DGA 域名检测方法：中国，CN105577660A［P］. 2016 - 05 - 11.

[6] AGYEPONG E，BUCHANAN W J，JONES K. Detection of algorithmically generated malicious domain［C］//International Conference of Advanced Computer Science & Information Technology，2018.

[7] 韩春雨，张永铮，张玉. Fast - flucos：基于 DNS 流量的 Fast

－flux 恶意域名检测方法［J］. 通信学报，2020，41（5）：37－47.

［8］MANASRAH A M，KHDOUR T，FREEHAT R. DGA-based botnets detection using DNS traffic mining［J］. Journal of King Saud University-Computer and Information Sciences，2022，34（5）：2045－2061.

［9］WANG T S，LIN H T，CHENG W T，et al. DBod：clustering and detecting DGA-based botnets using DNS traffic analysis［J］. Computers & Security，2017，64：1－15.

［10］ANTONAKAKIS M，PERDISCI R，NADJI Y，et al. From throw-away traffic to bots：detecting the rise of DGA-based malware［C］//The 21st USENIX Security Symposium USENIX Security 12，2012：491－506.

［11］SILVEIRA M R，CANSIAN A M，KOBAYASHI H K. Detection of malicious domains using passive DNS with XGBoost［C］//2020 IEEE International Conference on Intelligence and Security Informatics（ISI）. IEEE，2020：1－3.

［12］HIGHNAM K，PUZIO D，LUO S，et al. Real-time detection of dictionary DGA network traffic using deep learning［J］. SN Computer Science，2021，2（2）：110.

［13］KUMAR A D，THODUPUNOORI H，VINAYAKUMAR R，et al. Enhanced domain generating algorithm detection based on Deep Neural Networks［J］. Deep Learning Applications for Cyber Security，2019：151－173.

［14］YU B，GRAY D L，PAN J，et al. Inline DGA detection with deep networks［C］//IEEE International Conference on DataMining Workshops. IEEE，2017.

［15］申宋彦. 基于域名特征融合的恶意域名检测方法研究［D］. 兰州理工大学，2023.

［16］VINAYAKUMAR R，SOMAN K P，POORNACHANDRAN P，et al. Evaluating deep learning approaches to characterize and classify the DGAs at scale［J］. Journal of Intelligent & Fuzzy Systems，2018，34（3）：1265－1276.

［17］WOODBRIDGE J，ANDERSON H S，AHUJA A，et al. Predicting domain generation algorithms with Long Short-term Memory Networks［J］. arXiv preprint arXiv：1611.00791，2016.

［18］LISON P，MAVROEIDIS V. Automatic detection of malware-generated domains with recurrent neural models［J］. arXiv preprint arXiv：1709.07102，2017.

［19］盛振威，徐国天. 基于融合 CNN 与 GRU 的 DGA 恶意域名检测方法［J］. 网络安全技术与应用，2022（12）：29－32.

［20］袁辰. 基于对抗模型的恶意域名检测方法的研究与实现［D］. 北京：北京建筑大学，2018.

［21］ANDERSON H S，WOODBRIDGE J，FILAR B. DeepDGA：adversarially-tuned domain generation and detection［C］//ACM Workshop on Artificial Intelligence & Security. ACM，2016.

［22］CHEN Y，ZHANG S，LIU J，et al. Towards a deep learning approach for detecting malicious domains［C］//IEEE International Conference on Smart Cloud SmartCloud. IEEE，2018.

［23］陈立皇，程华，房一泉. 基于注意力机制的 DGA 域名检测算法［J］. 华东理工大学学报（自然科学版），2019，45（3）：478－485.

［24］SATOH A，NAKAMURA Y，NOBAYASHI D，et al. Estimating the randomness of domain names for DGA bot callbacks［J］. IEEE Communications Letters，2018，22（7）：1378－1381.

［25］CURTIN R R，GARDNER A B，GRZONKOWSKI S，et al. Detecting DGA domains with recurrent neural networks and side information［EB］. Eprint arXiv：1810.02023，2018.

（收稿日期：2023－11－10）

作者简介：

郝旭光（1974－），男，本科，高级工程师，主要研究方向：网络域名管理、政务信息化建设和网络安全。

版权声明

凡《网络安全与数据治理》录用的文章，如作者没有关于汇编权、翻译权、印刷权及电子版的复制权、信息网络传播权与发行权等版权的特殊声明，即视作该文章署名作者同意将该文章的汇编权、翻译权、印刷权及电子版的复制权、信息网络传播权与发行权授予本刊，本刊有权授权本刊合作数据库、合作媒体等合作伙伴使用。同时，本刊支付的稿酬已包含上述使用的费用，特此声明。

《网络安全与数据治理》编辑部