计算机科学
COMPUTER SCIENCE

基于TCN-BiLSTM的入侵检测算法研究

白万荣, 魏峰, 郑广远, 王宝会

引用本文

白万荣, 魏峰, 郑广远, 王宝会. 基于TCN-BiLSTM的入侵检测算法研究[J]. 计算机科学, 2023, 50(11A): 230300142-8.

BAI Wanrong, WEI Feng, ZHENG Guangyuan, WANG Baohui. Study on Intrusion Detection Algorithm Based on TCN-BiLSTM[J]. Computer Science, 2023, 50(11A): 230300142-8.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.230300142

基于 TCN-BiLSTM 的入侵检测算法研究

白万荣[1]　魏　峰[1]　郑广远[2]　王宝会[2]
1 国网甘肃省电力公司电力科学研究院　兰州 730070
2 北京航空航天大学软件学院　北京 100191
(baiwanrong@yeah.net)

摘　要　网络安全直接关系到国家安全,如何准确高效地检测到电网中的网络威胁至关重要。针对传统 CNN 感受野较小以及未考虑数据时序特征的问题,结合网络流量数据的空间特征和时间特征,提出了一种基于时间卷积网络(TCN)和双向长短期记忆网络(BiLSTM)的注意力入侵检测算法。首先将网络流量特征进行特征编码,再使用森林优化特征筛选算法,减少数据的冗余性;然后进行重采样,解决数据不平衡问题;最后将数据输入到深度神经网络中,处理后的数据经过 TCN 和 BiLSTM 网络进行特征学习,通过自注意力机制进行权重分配,最终进行分类,实现入侵检测。在 NSL-KDD 数据集上进行对比实验,相比 CNN-BiLSTM 注意力模型,所提方法的准确率提升 4.3%,F1 值提升 1.8%,实验结果表明,该算法能有效地对网络入侵检测进行识别。

关键词:入侵检测;时间卷积网络;双向长短期记忆网络
中图法分类号　TN915.08

Study on Intrusion Detection Algorithm Based on TCN-BiLSTM

BAI Wanrong[1],WEI Feng[1],ZHENG Guangyuan[2] and WANG Baohui[2]
1 State Grid Gansu Electric Power Research Institute,Lanzhou 730070,China
2 School of Software,Beihang University,Beijing 100191,China

Abstract　Network security is directly related to national security. How to accurately and efficiently detect network threats in the power grid is very important. Aiming at the problems of small receptive field and no consideration of data timing characteristics of traditional CNN,combined with spatial and temporal characteristics of network traffic data,an attention intrusion detection algorithm based on time convolution network(TCN) and BiLSTM is proposed. First,feature coding is performed on network traffic characteristics. Then the forest optimization feature screening algorithm is used to reduce the redundancy of the data,and then resampling is carried out to solve the problem of data imbalance. Finally,the data is input into the deep neural network,and the processed data is extracted by the TCN and BiLSTM networks for feature learning. The self-attention mechanism is used for weight allocation,and finally the classification is carried out to realize the intrusion detection. The data set adopts NSL-KDD,and the experimental results show that the algorithm can identify network intrusion detection effectively.

Keywords　Intrusion detection,Temporal convolutional network,Bi-directional long short-term memory

1　引言

近年来,互联网迅速发展,给人们带来了诸多的便利,同时又带来了一系列网络安全问题。面对日新月异的网络攻击手段,安全系统面临着各种新型的威胁。

入侵检测技术大概可以分为基于签名的检测和基于流量特征的检测。基于签名,主要是依靠专家经验制定规则进行判断,该方法效率较高,但是需要一定的专家经验知识支撑,同时需要及时更新知识库[1],对于一些新型的攻击无能为力。基于流量特征的入侵检测,需要发掘流量中的异常部分,近年来,入侵检测引入了机器学习、深度学习的方法,通过学习流量中的隐藏知识,来发觉其中的异常,对这些流量进行分类,可以进一步提高了攻击的检测精度和应对未知攻击的能力[2]。

在传统机器学习方面,如贝叶斯网络[3]、决策树[4]、支持向量机[5](Support Vector Machine,SVM)等,在入侵检测的应用上效果显著。一些方法通过优化机器学习算法的参数选择来提高效率,如文献[6]使用 PSO 算法优化 SVM 的参数,使得算法在入侵检测上的效率提高。文献[7]使用 QPSO 算法优化 LightGBM 算法,并在 Spark 集群上进行训练,进一步提高了识别的效率。以上研究由于机器学习模型较为浅层,容易出现过拟合和泛化能力差的问题,近年来更多的是将深度学习与入侵检测进行结合,通过训练更深层的模型进行入侵检测。Yu 等[8]使用了双向长短期记忆网络进行入侵检测。

基金项目:基于后防护的全流程多源网络威胁溯源技术研究项目(52272222001B)
This work was supported by the Research Project of Multi-source Network Threat Tracing Technology Based on Post-protection (52272222001B).
通信作者:郑广远(zhengguangyuan@buaa.edu.cn)

将流量数据转为 NLP 序列，同时引用了注意力机制，利用双向长短期记忆网络进行异常检测，最终效果较好。Tan 等[9]提出一种基于注意力机制的入侵检测模型。首先在流量数据的特征中加入位置编码，并利用注意力机制完成对重要特征的关注，在实验中证明了该方法的有效性，但注意力机制可能会导致梯度消失或梯度爆炸问题。Ahsan 等[10]提出一种使用 CNN 和 LSTM 的入侵检测方法，该方法利用 CNN 进行空间特征提取，并使用 LSTM 挖掘特征间的时序关系。该方法从 NSL-KDD 数据集[11]中仅选取 10%的数据作为训练集，选取 5000 个样本作为测试集，获得了 99.7%的准确率。Hsu 等[12]同样利用 CNN 和 LSTM 的方法在 NSL-KDD 数据集上进行实验，并利用测试集进行测试，结果显示这种方法是有效的，但没有考虑到经过 CNN 提取的特征破坏了原有特征间的时序关系。

总结国内外学者的研究，目前入侵检测的算法研究主要还存在以下几点问题：

1）原始网络流量经过特征提取后，数据集的特征维度较大，特征较为冗余，会增加计算量，降低算法的效率，需要进行适当的特征筛选；

2）目前网络特征数据集多为不平衡数据集，正常流量较多，攻击样本数量较少，同时对于不同的攻击类型，样本数量也存在巨大差异，对算法的鲁棒性有影响；

3）网络流量特征既有空间特征和时序特征，需更有效地利用这两种特征，并训练出高效的算法。

本文基于以上问题提出了一种基于 TCN-BiLSTM（时间卷积和双向长短期记忆并联结合）的入侵检测算法，并基于 NSL-KDD 数据集进行了验证。主要创新点如下：

1）对于数据冗余问题，使用森林优化的特征选择算法进行特征筛选，相较于其他算法，森林优化特征选择算法具有更好的分类性能及维度缩减能力；

2）使用 Adaptive Synthetic（ADASYN）自适应合成算法，通过计算样本之间的不平衡度，生成样本点，平均各个类别的样本数量，解决数据的不平衡问题；

3）使用 TCN-BiLSTM 注意力机制模型进行训练，解决感受野问题，同时兼顾数据的空间特征和时序特征，为入侵检测提供了新思路。

2 方法与模型

本文整体方法有 3 个值得注意的点，如图 1 所示，分别为森林优化特征选取、ADASYN 算法和 TCN-BiLSTM 自注意力机制模型。

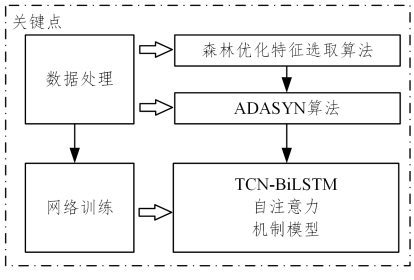

图 1 方法关键点

Fig. 1 Method key points

其中，森林优化特征选取和 ADASYN 算法解决数据不平衡的问题，在数据预处理阶段，对特征进行筛选，以及平衡样本。TCN-BiLSTM 自注意力机制模型解决算法分类不准确的问题，通过该模型提升网络流量入侵检测的准确性。接下来逐个介绍各个关键点。

2.1 森林优化特征选取

针对网络流量数据特征冗余的问题，需要用到特征筛选算法，将原始特征进行降维，以提升计算的效率。

森林优化算法（Forest Optimization Algorithm，FOA）是 Ghaemi 等[13]于 2014 年提出的一种仿生类进化算法，用于解决单目标非线性连续搜索空间问题。森林优化特征选择（Feature Selectionusing Forest Optimization Algorithm，FSFOA）算法将 FOA 算法用在特征选择上，并取得了不错的效果。

在 FSFOA 算法中，每棵树代表问题的一个可能解，即一个特征子集。树中的每个“1”表示相应的特征被选择参与机器学习过程，每个“0”表示在学习过程中相应特征被排除。FSFOA 算法包含 5 个部分：初始化森林、局部播种、形成候选森林、全局播种（global seeding）、更新最优树（update the best tree）。该算法以初始化森林为起始点，经过局部播种、形成候选区、全局播种这 3 个主要步骤产生新树，更新森林，经过 FSFOA 算法的多次迭代后得到最优特征子集。整体流程如图 2 所示。

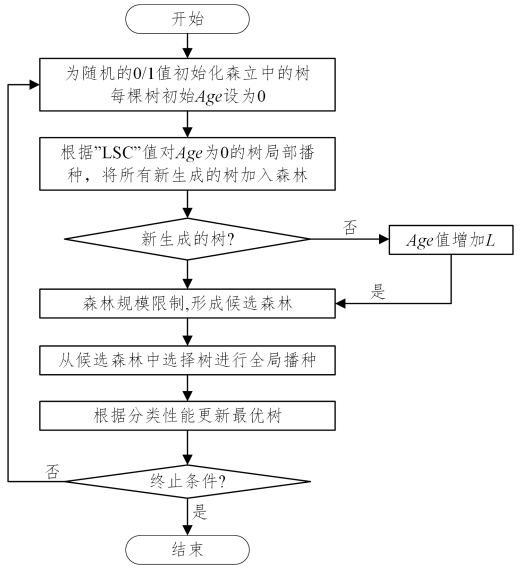

图 2 FSFOA 流程图

Fig. 2 FSFOA algorithm flowchart

但是在 FSFOA 算法的局部播种阶段，可能会有许多的劣质树加入到森林之中，进而影响最终的分类效果，所以 Chu 等[14]提出的 IFSFOA 算法在局部播种阶段采用极度贪婪策略，其主要思想是只将优质的新树加入到森林中，淘汰劣质树。但是考虑到贪婪算法容易陷入局部最优解，Chu 等[14]在保持全局播种阶段的树的数目不变的同时，将其中的侯选森林替换为由候选森林加上森林中所有 Age 为 0 的树，这样就使得同一棵 0-Age 树既可以局部播种，又可以全局播种，一定程度上解决了因极度贪婪策略带来的易陷入局部最优解的问题。

本文中使用 IFSFOA 算法，将算法应用于 NSL-KDD 数据集中，对原始流量特征进行特征筛选，最终保留了特征数为

20 的训练子集以便后续分类。

2.2 ADASYN 算法

在网络流量中，正常流量样本往往占有更大的比例，攻击样本占比较小，且不同的攻击类型样本数量的差距也往往过大，样本分布不均衡将导致模型较难学习到少数样本的特征规律。在少量的数据集上经过训练得到的分类模型，也容易产生过拟合的问题。当模型应用到新的数据上时，模型的准确性和健壮性将很差，所以往往需要对原始数据进行样本平衡。

ADASYN 关注的更多的是样本的不平衡度，将少数样本的样本数经过平衡后等于多数样本的样本数，算法的详细描述如算法 1 所示。

算法 1 ADASYN

输入：**Dtr** 为待训练样本，本文中为经过特征筛选后的数据集，包含 m 个样本$\{x_i, y_i\}$，i=1，2，…，m。其中x_i是 n 维特征空间 X 中的一个实例，在文中代表一条网络流量的特征；$y_i \in y$，代表x_i相关的类别标签，此处为网络攻击的类型；将m_s和m_l分别定义为少数类样本的数量和多数类样本的数量，因此$m_s \leqslant m_l$，$m_s + m_l = m$

1. 计算不平衡度

$$d = \frac{m_s}{m_l}, d \in (0, 1] \tag{1}$$

2. 如果 $d < d_{th}$，(d_{th}是类不平衡比率的预设阈值)

步骤 1 计算少数样本需要扩充的样本数：

$$G = m_l - m_s \tag{2}$$

此处经过平衡后将形成一个各类样本数完全平衡的数据集。

步骤 2 对于少数类中每个样本x_i，根据 n 维空间中的欧几里得距离找到 k 邻近，并计算r_i：

$$r_i = \frac{\Delta_i}{K}, i = 1, 2, \cdots, m_s \tag{3}$$

其中，r_i为x_i的 k 邻近中多数类的占比，$r_i \in [0, 1]$，Δ_i为多数类数量。

步骤 3 对r_i进行标准化处理，可得每个类别的占比。

$$\hat{r}_i = \frac{r_i}{\sum_{i=1}^{m_s} r_i} \tag{4}$$

步骤 4 计算少数类中每个样本x_i需要生成的合成样本的个数。

$$g_i = r_i \times G \tag{5}$$

步骤 5 根据 SMOTE 算法生成合成样本。

$$s_i = x_i + (x_{zi} - x_i) \times \lambda \tag{6}$$

其中，$(x_{zi} - x_i)$是 n 维空间中的差分向量；λ 是一个随机数，$\lambda \in [0, 1]$。重复合成直到满足需要步骤 5 合成的数目为止。

将特征筛选后的数据集输入到 ADASYN 算法中，最终形成多类样本数完全平衡的数据集，同时扩充了原始数据集，更加有利于后续的模型训练。

为了验证平衡后数据集的效果优劣，文中第 3 节进行了实验验证。

2.3 TCN-BiLSTM 自注意力机制模型

网络流量特征一般包含一定的空间特征和时序特征，要学习到网络流量特征之间的隐含关系，需要引入合适的深度神经网络模型。

在计算机视觉和自然语言处理中，卷积神经网络(CNN)对于特征的学习有着良好的效果。后续又出现了对于时序序列数据学习的长短期记忆网络等，但 CNN 效果受限于感受野，因此，Bai 等[15]提出了一种特殊的卷积神经网络——时间卷积网络(TCN)用于序列建模任务。

TCN 是将因果卷积和空洞卷积相结合的时间卷积作为卷积层，将其与一个 1×1 的卷积融合为一个残差模块，再由多个残差模块相堆叠组成的神经网络，其中还有权值归一化和 Dropout。TCN 的结构如图 3 所示。

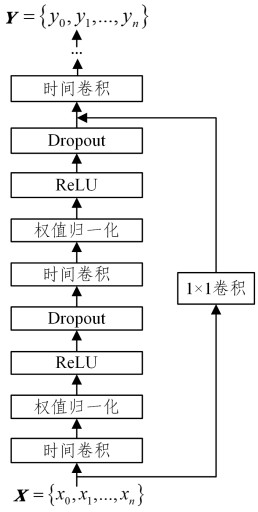

图 3 TCN 整体结构

Fig. 3 TCN overall structure

TCN 中的卷积包含因果卷积，能更好地处理序列问题；同时，膨胀卷积的使用扩大了卷积的感受野；残差模块的加入，可以使历史信息得到很好的保留。TCN 的详细介绍如下。

1)因果卷积

TCN 使用了更强大的因果卷积，防止了信息的遗漏。因果卷积只能看到未来信息，定义如下：

$$TCN = 1D\ FCN + casual\ convolutions \tag{7}$$

网络流量特征较多，通过因果卷积能防止信息的遗漏。但因果卷积也有传统卷积的问题，当需要许多历史信息时，需要进行层数的堆叠或扩大卷积核，这会带来较高的计算代价，因此需要使用空洞卷积来解决该问题。

2)空洞卷积

将空洞卷积应用于因果卷积中，不用增加计算量就可以获取到更多的历史信息。

空洞卷积主要通过“膨胀率”来扩大它的感受野，“膨胀率”定义了卷积在参数不变的基础上扩大的倍数。因此，卷积网络用比较少的空洞卷积层，就能够得到较大的感受野。

通过扩张卷积和因果卷积的融合，卷积能学到更大感受野的流量特征，保证流量的更多特征能在单个卷积核上更充分地学习。

3)残差链接

网络深度的增加，往往会引发梯度消失或梯度爆炸的问题。而且随着网络加深，模型性能逐渐增加至饱和，然后就会下降。这是由冗余的网络层学习了不是恒等映射的参数导致的。残差链接(Residual Connections)能够有效解决上述问题，即使网络很深也能保持良好的性能。

使用因果卷积，可以在保留更多原始信息的同时，学习到网络流量特征与特征之间的因果关系。通过扩张卷积，增加卷积的感受野，一个卷积能够学习到更多的特征；同时，残差模块的引入，可以加深 TCN 的层数，卷积能更充分地学习到

网络流量隐含特征，不容易出现梯度问题，模型性能能得到更好的提升。

基于以上优势，针对网络流量特征多、时序和空间特征并存的特点，TCN 非常适合对网络流量的特征学习。同时为了加强网络流量的时序特征学习，引入了 BiLSTM(双向长短期记忆)网络进行学习。

LSTM 网络层是由多个 cell 单元进行连接组成的，单元结构如图 4 所示。

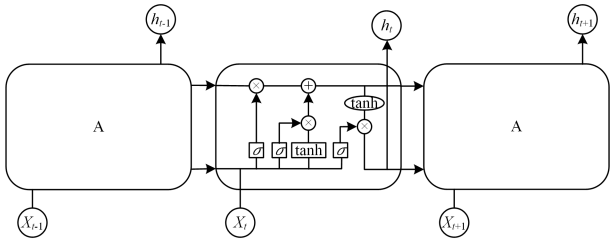

图 4 LSTM 结构

Fig. 4 LSTM structure

可以看到 LSTM 的重复结构 A 有 4 层。贯穿这些单元的主线为上面的一条线，每当经过 1 个重复结构 A 的时候，都会有相应的操作来决定舍弃什么旧的信息以及添加什么新的信息。对 cell state 的信息增减进行控制的结构称为门(gates)。一个 LSTM 单元中有 3 个这样的门，分别是遗忘门(forget gate)、输入门(input gate)、输出门(output gate)。

遗忘门决定了要从 cell state 中舍弃什么信息。其表示在入侵检测中，在网络流量数据输入到单元中时，一些信息通过遗忘门进行舍弃。

遗忘门公式为 $f_t=\sigma(\boldsymbol{W}_f\cdot[\boldsymbol{h}_{t-1},x_t]+\boldsymbol{b}_f)$，在连接 h_{t-1} 和 x_t 之后会乘以一个权值 $\boldsymbol{W}_f$ 并加上偏置 $\boldsymbol{b}_f$，这两个参数就是网络需要学习的参数。若 hidden state 的大小(也就是 size of hidden layer of neurons)为 $\boldsymbol{h}_{\text{size}}$，那么 $\boldsymbol{W}_f$ 的大小就为 $\boldsymbol{h}_{\text{size}}*\boldsymbol{h}_{\text{size}}$。$\boldsymbol{h}_{\text{size}}$ 的数值是人工设定的。

输入门决定了在网络流量数据的训练过程中，单元需要保留哪部分新的数据信息。其通过输入上一状态的输出 $\boldsymbol{h}_{t-1}$ 和当前状态输入信息 $\boldsymbol{x}_t$ 到一个 Sigmoid 函数中，产生一个介于 0 到 1 之间的数值 i_t 来确定需要保留多少新信息。同时，一个 tanh 函数层会通过上一状态的输出 $\boldsymbol{h}_{t-1}$ 和当前状态输入信息 X_t 来得到一个将要加入到 cell state 中的“候选新信息”$\widetilde{C}_t$。将刚才得到的数值 i_t 与“候选新信息”$\widetilde{C}_t$ 相乘得到真正要加入到 cell state 中的更新信息。

$$i_t=\sigma(\boldsymbol{W}_i\cdot[\boldsymbol{h}_{t-1},x_t]+\boldsymbol{b}_i) \tag{8}$$

$$\widetilde{C}_t=\tanh(\boldsymbol{W}_C\cdot[\boldsymbol{h}_{t-1},x_t]+\boldsymbol{b}_C) \tag{9}$$

输入门(一个 Sigmoid 函数层与一个 tanh 层)，两个神经网络层都会和之前遗忘门一样学习各自的参数。

输出门决定了在对网络流量数据的学习过程中，要从 cell state 中输出什么信息。与之前类似，首先有一个 Sigmoid 函数产生一个介于 0 到 1 之间的数值 o_t 来确定需要输出多少 cell state 中的信息。cell state 的信息在与 o_t 相乘时首先会经过一个 tanh 层进行“激活”(非线性变换)，之后得到的就是这个 LSTM block 的输出信息 h_t。

$$o_t=\sigma(\boldsymbol{W}_o\cdot[\boldsymbol{h}_{t-1},x_t]+\boldsymbol{b}_o) \tag{10}$$

$$h_t=o_t*\tanh(C_t) \tag{11}$$

输出门同样有自己的权值参数需要学习。整个一层神经网络由若干个 LSTM 单元组成，将经过前部分处理过后的网络流特征序列($x_1,x_2,\cdots,x_n$)送入网络，经过 LSTM 网络，最终由全连接层展成需要的维度。

前向的 LSTM 与后向的 LSTM 结合成为 BiLSTM。BiLSTM 可以有效地学习网络流量特征与特征之间的前后信息，利用当前位置的正反方向信息进行双向学习。

将 TCN 与 BiLSTM 相并联，形成 TCN-BiLSTM 模型。TCN 学习到网络流量数据的时序特征和空间特征，BiLSTM 加深网络流量时序特征的学习，通过注意力机制再次进行特征提取，最终给出分类结果。

注意力机制，顾名思义，通过专注某些部分，分给该部分更多的权重。注意力机制的本质是让模型学习权值，为输入加上权值，关注更多想要关注的数据，使得后续的数据训练受到这一部分的影响。

自注意力机制动态调节模型的连接权重，可以作为神经网络中的一层来使用。其计算过程如图 5 所示。

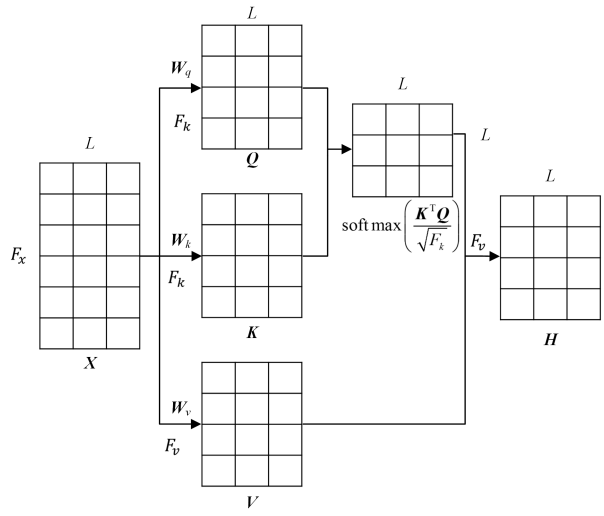

图 5 自注意力机制

Fig. 5 Self-Attention mechanism

输入序列 $\boldsymbol{X}=[x_1,x_2,\cdots,x_n]\in R^{F_x\times L}$，输出序列 $\boldsymbol{H}=[h_1,h_2,\cdots,h_n]\in R^{F_V\times L}$，自注意力模型的计算过程如下：

1)将每个 x_i 输入映射到 3 个不同的空间，得到查询向量 $q_i\in^{F_x\times L}$，键向量 $k_i\in^{F_x\times L}$，值向量 $v_i\in^{F_x\times L}$。输入序列线性映射过程如下：

$$\boldsymbol{Q}=\boldsymbol{W}_q\boldsymbol{X}\in^{F_x\times L} \tag{12}$$

$$\boldsymbol{K}=\boldsymbol{W}_k\boldsymbol{X}\in^{F_x\times L} \tag{13}$$

$$\boldsymbol{V}=\boldsymbol{W}_v\boldsymbol{X}\in^{F_x\times L} \tag{14}$$

其中，$\boldsymbol{W}_q\in R^{F_k\times F_x}$，$\boldsymbol{W}_k\in R^{F_k\times F_x}$，$\boldsymbol{W}_v\in R^{F_v\times F_x}$ 为 3 个映射参数矩阵，$\boldsymbol{Q}=[q_1,q_2,q_L]$，$\boldsymbol{K}=[k_1,k_2,k_L]$，$\boldsymbol{V}=[v_1,v_2,v_L]$分别是由查询向量、键向量、值向量组成的矩阵。

2)对每个查询向量，计算输出向量 $\boldsymbol{h}_l$。

$$\boldsymbol{h}_l=att((\boldsymbol{K},\boldsymbol{V}),q_l)=\sum_{j=1}^{L}a_{lj}v_j \tag{15}$$

其中，l 和 j 表示输出和输入向量序列的位置，a_{lj} 表示第 l 个输入关注到第 j 个输入的权重。

通过自注意力机制来调整模型连接权重，将特征乘以重要性因子，提高模型的精度。

本文 TCN-BiLSTM 自注意力机制模型如图 6 所示。

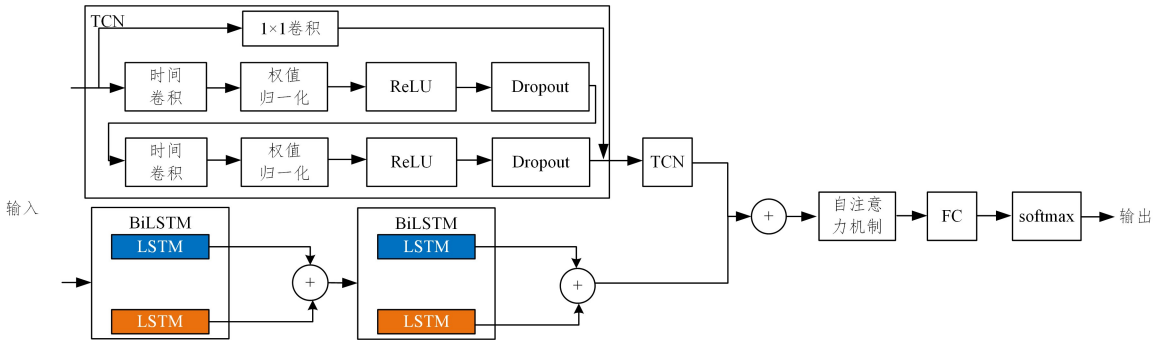

图 6　TCN-BiLSTM 网络结构

Fig. 6　TCN-BiLSTM network architecture

其中,TCN 部分由 2 个隐藏层构成,通过去权值归一化以及 dropout 来提高性能,BiLSTM 部分采用 2 层结构,最后经过一个全连接网络,实现分类。该模型具体算法如算法 2 所示。

算法 2　TCN-BiLSTM

输入:原始数据集

输出:样本类别

步骤 1　前向传播(模型提取特征进行分类)

1. 空间特征提取

1.1. 进行数据预处理,将结果输入 TCN;

1.2. 时空卷积提取特征,权重共享减少参数;

1.3. 用激活函数对时空卷积输出做非线性映射;

1.4. 通过 dropout,防止过拟合;

1.5. 将上述步骤重复两次,并将结果作为残差与原特征进行融合;

1.6. 重复 1.5 两次,得到空间特征,等待与 BiLSTM 层特征融合。

2. 时间特征提取

2.1　同空间特征提取,进行数据预处理,将结果送入 BiLSTM;

2.2　BiLSTM 模型通过更新门信息进行时间特征提取;

2.3　重复步骤 2.1 和 2.2 BiLSTM 层数的次数,输出得到时间特征。

3. 特征融合

将前两步得到的信息融合成"并联特征"。

4. 自注意力机制

将第三步得到的结果送入自注意力模型中,进行二次特征提取,选择重要的信息。

5. 分类

将上一步的输出通过全连接,再利用 softmax 函数进行分类。

步骤 2　反向传播

通过损失函数计算最后全连接层输出和真实值之间的误差,采用梯度下降的方法反向传播误差,逐层调整模型相关参数,减少真实值和模型输出值之间的误差。

重复步骤 1 和步骤 2,当训练轮次达到阈值时,停止训练。

模型复杂度分析通常使用 FLOPS,即每秒浮点运算次数,理解为计算速度。它是一个衡量硬件性能的指标。最终整体模型复杂度为 1 056 676FLOPS,参数量为 28 100。

本文提出基于 TCN-BiLSTM 的入侵检测模型的总体流程如图 7 所示。

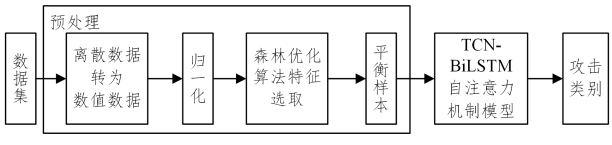

图 7　算法模型整体流程

Fig. 7　Overall process of algorithm model

本文的入侵检测方法整体流程主要分为两部分,第一部分为网络流量数据预处理,核心有两个,其一是通过森林优化算法进行特征选择,解决网络流量数据的冗余问题,其二是通过 ADASYN 算法进行重采样,解决网络流量数据攻击类别不平衡的问题;第二部分为模型训练部分,使用 TCN-BiLSTM 自注意力机制进行模型训练,最终将网络流量数据进行攻击类别的分类。

3　实验与结果分析

为了验证本文提出的入侵检测方法的准确性,在网络公开数据集 NSL-KDD 上进行了实验。本文设计了多组对比实验,对比了使用森林优化特征选取算法进行特征筛选以及未使用该算法的模型效果对比了使用 ADASYN 算法进行样本平衡以及未使用样本平衡的模型效果,对比了本文 TCN-BiLSTM 模型和随机森林以及 CNN-BiLSTM 模型的效果;同时对于 BiLSTM 层数进行了对比实验,选择了效果最好的 BiLSTM 层数,且对于模型训练轮次进行了实验,选择出了合适的训练轮次。接下来是对数据集的详细介绍。

3.1　数据集

NSL-KDD 数据集是对 KDD99 数据集的改进,其删除了原有 KDD99 数据集中的冗余数据,使数据集中的数据更加适合入侵检测研究。NSL-KDD 数据集被划分为训练集 KDDTrain+和测试集 KDDTest+,训练集和测试集的比例大约为 5∶1,详细数量如表 2 所列。NSL-KDD 数据集中包含了 39 种攻击类型和 1 种正常类型。其中,在训练集中只包含了 22 种攻击类型和 1 种正常类型;在测试集中共有 40 种类型,其中出现了训练集中不曾参与训练的 17 种新的攻击类型,这些不曾参与训练的攻击类型的存在,使得模型对它们的识别较为困难。本文将 NSL-KDD 数据集中的 39 种攻击类型划分为 DoS,Probe,R2L,U2R 这 4 种,如表 1 所列。

表 1　数据集中攻击类型分布

Table 1　Distribution of attack types in the dataset

攻击类型	攻击名称
Probe	Satan,Saint,Ipsweep,Portsweep,Nmap,Mscan
DoS	Apache2, Smurf, Neptune, Back, Teardrop, Pod, Land, Mailbomb,Processtable,UDPstorm
R2L	WarezClient,Guess_Password,WarezMaster,R2L,Imap, Ftp _ Write, Named, MultiHop, Phf, Spy, Sendmail, SnmpGetAttack,Worm,Xsnoop,Xlock,SnmpGuess
U2R	Buffer_Overflow, Httptuneel, Rootkit, Perl, Ps, Xterm, SQLattack,LoadModule

同时,该数据集也存在着分类任务中普遍存在的问题,即

数据不均衡问题，这将导致模型对少数类的分类准确率较低，降低了模型检测性能。NSL-KDD 数据集中五分类数据分布如表 2 所列。

表 2 数据集分布

Table 2 Dataset distribution

Dataset	Numbers of Records					
	Total	Normal	Dos	Probe	U2R	R2L
KDD Train+20%	25192	13449 (53%)	9234 (37%)	2289 (9.16%)	11 (0.04%)	209 (0.8%)
KDD Train+	125973	67343 (53%)	45927 (37%)	11656 (9.11%)	52 (0.04%)	995 (0.85%)
KDD Test+	22544	9711 (43%)	7458 (33%)	2421 (33%)	200 (0.9%)	2654 (12.1%)

从表 2 可以看出，数据类型的分布极度不平衡，Normal 类型数据占比高达 53%，而 U2L 占比却只有 0.04%，比率高达 1336∶1。这将导致模型在训练时更加倾向于 Normal 类型，从而限制了模型整体识别性能的提升和对于少数类的识别率。

数据集共有 43 列特征，前 41 列为网络流量特征，第 42 列为流量类别，第 43 列为能够正确标注给定记录的学习者数量。第 43 列与流量本身无关，故删除该特征。将流量类型 Normal，Dos，Probe，R2L 和 U2R 分别映射为 0，1，2，3，4。

数据中存在 3 个字符型离散特征：'protocol_type''service''flag'，首先对它们编码，将其转换成数字表示。此处采用硬编码方法。

同时，为了减少数据分布不平衡的问题，文中采用 ADASYN 生成少数类样本以解决数据不平衡问题。

最后将特征进行归一化，以便于收敛。归一化公式如式(6)所示：

$$x_i'=\frac{x_i-\min(\boldsymbol{x})}{\max(\boldsymbol{x})-\min(\boldsymbol{x})} \tag{16}$$

3.2 模型训练

采用 NSL-KDD 数据集中的 KDDTrain+进行训练，将 KDDTrain+20%作为验证集，KDDTest+作为测试集。

IFSFOA 算法的参数根据常用的经验值设置，将 lifetime 固定为 15、arealimit 为 50、transferrate 为 5%，LSC 和 GSC 根据流量数据的特征维数设置，分别设置成特征维数的 1/5，LSC 为 8，GSC 是 LSC 的二倍，为 16。

TCN 层数设置为 2 层时，拟合效果较好，当继续增加层数之后，模型的参数随之增加，更容易出现过拟合现象，其中各含有的 64 个隐层节点数能够使模型效果达到最优，过大的膨胀因子会容易导致丢失数据之间的连续性和完整性，所以膨胀因子为 2，dropout 设置为常用值 0.5。

BiLSTM 隐藏节点设置为 100，BiLSTM 层数即 BiLSTM 网络的深度，一般来说网络越深，拟合效果越强，越能学习到深层的特征，但过深的网络会引起过拟合以及梯度问题。为了获得合适的 BiLSTM 层数，本文进行了实验验证，结果如表 3 所列。

表 3 BiLSTM 层数模型对比

Table 3 Comparison of BiLSTM Layer models

层数	准确率/%
1	78.6
2	80.7
3	79.1

可以看出在 BiLSTM 层数设置为 2 层时拟合效果更好，所以本文算法采用的 BiLSTM 层数为 2。

训练时采用小样本随机抽样，batch_size 设置为 256，每次训练的迭代次数动态调整，根据效果来决定轮次。优化器为 Adam，学习率设置成常用经验值，为 1×10^{-4}，学习率衰减为 1×10^{-5}，此时模型收敛速度不会过快或者过慢，同时学习率衰减使得模型不易出现震荡问题，损失函数为多分类常用的损失函数交叉熵函数，计算式如下：

$$Loss(\hat{x},x)=-\sum_{i=1}^{n}x\log(\hat{x}) \tag{17}$$

其中，x 是真实标签，$\hat{x}$ 是预测的类分布。

3.3 结果分析

本文使用神经网络模型常用的评估指标对 TCN-BiLSTM 模型训练过程中得到的测试结果进行分析。只通过模型的准确率来评估模型的预测性能是不够的，当数据类别极为不平衡时，模型总体的准确率可能会有虚高的情况。而混淆矩阵从模型对各个类别的预测表现来评价模型的预测性能，这样能更加全面地对模型进行评估。混淆矩阵如表 4 所列。

表 4 混淆矩阵

Table 4 Confusion matrix

预测/真实	1(Postive)	0(Negative)
1(Postive)	TP(真阳)	FP(假阳)
0(Negative)	FN(假阴)	TN(真阴)

评价模型预测性能时最直观的标准是对准确率(Accuracy，简称为 Acc)的计算，其表示在所有样本中，通过模型预测得出正确结果的样本所占比例，表达式如式(18)所示：

$$Accuracy=\frac{(TP+TN)}{(TP+FP+FN+TN)} \tag{18}$$

其中，TN 是数据为异常且预测为异常的数量，TP 是数据为正常且预测也为正常的数量，FN 是数据为正常但预测为异常的数量，FP 是数据为异常但预测为正常的数量。

精准率(Precision)表示模型预测为真的样本，其确实为真的样本所占比例，表示方法如式(19)所示。召回率(Recall)表示所有确实为真的样本中，模型预测为真的样本占比，如式(20)所示：

$$Precision=\frac{TP}{TP+FP} \tag{19}$$

$$Recall=\frac{TP}{TP+FN} \tag{20}$$

但精准率和召回率有时候会出现矛盾的情况，这样就需要综合考虑，最常见的方法就是 F1 值，F1 值公式如下，其中 P，R 分别代表精准率和召回率。

$$F_1=\frac{2PR}{P+R} \tag{21}$$

为了更好地分析 TCN-BiLSTM 模型的性能，训练结束后记录损失函数的结果，如图 8 所示。

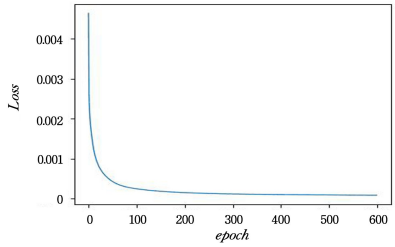

图 8 训练 loss 图

Fig. 8 Training loss

模型训练过程，训练集和验证集的准确率变化如图 9 所示。

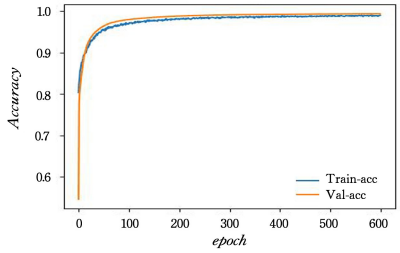

图 9　训练过程中训练集和验证集准确率

Fig. 9　Accuracy of training and validation sets during the training process

从图 8 和图 9 可看出，训练在 400 多个 epoch 时，loss 趋于收敛，同时训练集和验证集准确率也趋于收敛。记录训练集和验证集的准确率，如表 5 所列。

表 5　训练结果

Table 5　Training results

训练轮次	训练 Acc/%	验证 Acc/%
200	98.4	98.1
400	99.2	99.3
450	99.7	99.5
500	99.1	99.2

最终选择训练轮次设定为 450。

PR 曲线如图 10 所示，其中 AP(代表 PR 曲线下的面积)为 0.899，表明该算法分类效果较好。

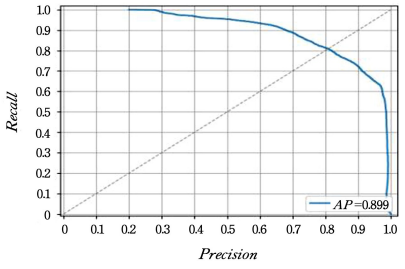

图 10　PR 曲线

Fig. 10　PR curve

为了更好查看模型效果，引入 ROC 曲线，其中 ROC 曲线的横坐标 FPR(False Positive Rate)$=FP/(F+TN)$，纵坐标 TPR(True Positive Rate)$=FP/(TP+FN)$，模型 ROC 曲线如图 11 所示，AUC(代表 ROC 曲线下面的面积)为 0.959，说明模型的整体表现良好。

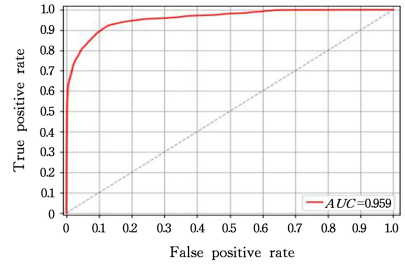

图 11　ROC 曲线

Fig. 11　ROC curve

将同时使用数据平衡算法和森林特征选取算法与未使用时在本文的 TCN-BiLSTM 模型上进行了对比实验，实验结果如表 6 所列。

表 6　实验效果对比

Table 6　Comparison of experimental effects

(单位：%)

算法名称	Acc	P	R	$F1$
未使用数据平衡算法和森林特征选取算法	76.4	78.3	76.2	77.3
使用 ADASYN 算法	77.2	78.4	80.4	79.4
使用森林特征选取算法	78.4	78.6	78.0	78.3
使用数据平衡算法和森林特征选取算法	80.7	83.2	82.8	82.9

可以看出使用森林优化特征选取算法对比未使用时准确率提升 0.8%，F1 值提升 1.9%。使用 ADASYN 算法对比未使用时准确率提升 2%，F1 值提升 2%，同时由于森林优化特征选取减少了数据冗余，ADASYN 算法进行了数据平衡，因此解决了数据的问题。通过结合两种算法，准确率最终提升 4.3%，F1 值最终提升 5.6%。可以推断，两种方法结合更能有效提升入侵检测方法的准确率。

在验证集上同时使用森林优化特征选取算法和 ADASYN 算法对各个模型进行对比，结果如表 7 所列。

表 7　各个模型对比结果

Table 7　Comparison results of various models

(单位：%)

模型名称	Acc	P	R	$F1$
随机森林	72.3	73.4	74.1	73.7
DNN+BiLSTM	73.2	75.1	71.8	73.4
CNN+BiLSTM	75.6	76.9	78.1	77.5
TCN+BiLSTM	78.1	78.9	79.3	79.1
CNN+BiLSTM 注意力机制模型	76.7	81.4	80.8	81.1
本文模型	80.7	83.2	82.8	82.9

可以看出本文使用的 TCN-BiLSTM 入侵检测模型相较于其他模型具有良好的效果。使用 TCN 相比于使用 DNN 和 CNN，准确率分别提升了 4.9%和 2.5%，F1 提升了 5.7%和 1.6%；使用本文模型相比使用 CNN-BiLSTM 注意力模型准确率提升 4.0%，F1 值提升 1.8%。这说明本文提出的模型对于入侵检测数据集的学习能力十分良好。

本文的整体算法模型，在网络流量特征冗余问题上，使用了森林优化特征选取算法，减少了数据集的维度，经实验结果表明，该方法在准确率上有明显的提升。对于特征不平衡问题，使用了 ADASYN 算法，使得模型在准确率上也有了明显的提升，同时引入了 TCN-BiLSTM 模型，进一步提高了入侵检测的准确性。

本文对入侵检测所面临的种种问题带来了新的解决方案，本文提出的检测模型对网络攻击数据的标准化预处理一定程度上降低实验带来的误差与噪声，使用重采样技术减小了样本不平衡带来的影响，使用 TCN-BiLSTM 模型有效学习到了数据的隐性关系。

结束语　目前，将深度学习模型应用于网络入侵检测已经成为一个大趋势，TCN-BiLSTM 模型充分发挥了神经网络技术在图像和自然语言处理上的优势，创新性地将 TCN-BiLSTM 模型应用于入侵检测方法。这种方法降低了数据的复杂度，减少了参数量，提高了模型训练的速率，与此同时，该方法还提高了模型的精度，相比 CNN-BiLSTM 注意力模型准确率提升 4.0%，F1 值提升 1.8%。本文的研究成果对网络安全入侵检测具有很高的实用价值，为入侵检

测拓展了新的研究思路。

本文提出的入侵检测方法还存在一些不足，首先是部分类型攻击样本的数量较少导致模型的准确率还有提升空间，且目前仅训练了 NSL-KDD 数据集，后续可以扩展更多的数据集，进一步提升模型的泛化能力。

TCN-BiLSTM 模型的进一步优化以及未来应用于入侵检测的可能性，可以从数据处理、模型结构优化两个方向来探究。首先是数据处理，可以研究如何将离散型数据进行更有效的表达，以及如何进行更有效的特征编码。其次是模型优化，未来，随着入侵检测研究的发展，我们将研究出更适用入侵检测的模型方案。在更多数据样本的支持下 TCN-BiLSTM 模型可以进一步优化其结构，实现更高的效率。

参 考 文 献

[1] NIKOLOVA E, JECHEVA V. Some similarity coefficients and application of data mining techniques to the anomalybased IDS [J]. Telecommunication Systems, 2012, 50(2): 127-135.

[2] ALAZAB A, ABAWAJY J, HOBBS M, et al. Crime toolkits: the productisation of cybercrime [C] // IEEE. IEEE, 2013: 1626-1632.

[3] XIAO L, CHEN Y, CHANG C K. Bayesian Model Averaging of Bayesian Network Classifiers for Intrusion Detection [C] // Computer Software & Applications Conference Workshops. IEEE, 2014.

[4] JING X Y, BI Y, DENG H. An innovative two-stage fuzzykNN-DST classifier for unknown intrusion detection [J]. International Arab Journal of Information Technology, 2016, 13(4): 359-366.

[5] OHKI T, GUPTA V, NISHIGAKI M. Efficient Spoofing Attack Detection against Unknown Sample using End-to-End Anomaly Detection [C] // Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). 2019.

[6] ALSAADI H I, ALMUTTAIRIR M, BAYAT O, et al. Computational Intelligence Algorithms to Handle Dimensionality Reduction for Enhancing Intrusion Detection System [J]. Journal of Information Science andEngineering 2020, 36: 293-308.

[7] TANG C F, BULI N, AI Z. Research on networkintrusion detection based on LightGBM [J]. Computer Applications and Software, 2022, 39(8): 298-311.

[8] YU Y, LIU G, YAN H, e t al. Attention-based BiLSTM model for anoma- lous HTTP traffic detection [C] // 15th International Conference on Service Systems and Service Management. 2018: 1-6.

[9] TAN M, IACOVAZZI A, CHEUNG N M M, et al. A neural attention model for real-time network intrusion detection [C] // 2019 IEEE 44th Conference on Local Computer Networks. 2019: 291-299.

[10] AHSAN M, NYGARD K E. Convolutional neural networks with LSTM for intrusion detection [C] // Proceeding of 35th International Conference on Computers and Their Applications. 2020: 69-79.

[11] GURUNG S, GHOSE M K, SUBEDI A. Deep learning approach on network intrusion detection system using NSL-KDD dataset [J]. International Journal of Computer Network and Information Security, 2019, 11(3): 8-14.

[12] HSU C M, HSIEH H Y, PRAKOSA S W, et al. Using long short term memory based convolutional neural networks for network intrusion detection [C] // International Wireless Internet Conference. 2018: 86-94.

[13] GHAEMI M, FEIZI-DERAKHSHI M R. Forest optimization algorithm [J]. Expert Systems with Applications, 2014, 41(15): 6676-6687.

[14] CHU B, LI Z S, ZHANG M L, et al. Research onImprovements ofFeature Selection Using Forest Optimization Algorithm [J]. Journal of Software, 2018, 29(9): 2545-2558.

[15] BAI S, KOLTER J Z, KOLTUN V. An empirical evaluation of generic convolutionalandrecurrent networks for sequence modeling [J]. arXiv: 1803.01271, 2018.

BAI Wanrong, born in 1985, postgraduate, senior engineer, is a member of China Computer Federation. His main research interests include network security and machine learning.

ZHENG Guangyuan, born in 1996, M.S. His main research interests include artificial intelligence and network security.