Research Article

ACNNT3: Attention-CNN Framework for Prediction of Sequence-Based Bacterial Type III Secreted Effectors

Jie Li,1 Zhong Li,1,2 Jiesi Luo,3 and Yuhua Yao4

1School of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China
2School of Science, Zhejiang Sci-Tech University, Hangzhou 310018, China
3Key Laboratory for Aging and Regenerative Medicine, Department of Pharmacology, School of Pharmacy, Southwest Medical University, Luzhou 646000, China
4School of Mathematics and Statistics, Hainan Normal University, Haikou 571158, China

Correspondence should be addressed to Zhong Li; lizhong@zstu.edu.cn and Yuhua Yao; yaoyuhua2288@163.com

Received 6 February 2020; Revised 9 March 2020; Accepted 17 March 2020; Published 2 April 2020

1. Introduction

Gram-negative bacteria can secrete proteins into host cells through a variety of secretion systems which affect the cell and its external environment. This process can be mediated by a variety of secretory systems, which can be divided into eight categories: type I to VIII secretory systems (T1SS-T8SS) [1]. Type I and III secretory systems are independent of signal sequences (sec), while types II and IV depend on signal sequences. The proteins secreted by the secretory system have a signal peptide sequence mainly composed of N-terminal hydrophobic amino acids which can guide the protein through the cell membrane.

When the protein reaches the periplasm, the signal peptide is cut off. Type II and IV secretion systems remove the N-terminal part of the secreted protein in the periplasm. The difference between systems is that proteins pass through the outer membrane in different ways. When protein secreted by the type II secretion system passes through the outer membrane, an additional set of inner membrane and outer membrane proteins is needed to assist, while the type IV secretion system includes a series of autotransporters which form a hole in the outer membrane to make the protein pass through, autolyticly cut, and then release the protein. Neither the type I nor III secretion system processes the terminal amino acid of the secreted protein, nor does it appear that...
2 Computational and Mathematical Methods in Medicine

We collected a comprehensive dataset from multiple bacterial species known as T3SEs and non-T3SEs from previous studies by Yang et al. [10], Wang et al. [22], and Tay et al. [23]. CD-HIT [24] with the sequence identity cutoff of 30% was used for sequence alignment to remove proteins with high similarity, and by skipping proteins with less than 100 amino acids, we obtained a balance dataset containing 283 T3 proteins and 311 non-T3 proteins.

We established our negative sample set by selecting type I to VIII secreted proteins of Gram-negative bacteria and removing type III secreted proteins and their homologues. In order to establish a 1:3 ratio of positive to negative [11], we randomly selected negative samples from the previous work of Dong et al. [8] and eliminated protein sequences with high similarity, resulting in a total of 835 negative samples.

There are two test sets used to evaluate our method. The independent dataset collected from Li et al. [11] contains 35 type III effectors and 86 non-type III effectors. The other test dataset is from the plant pathogen P. syringae. 85 type III effectors and 14 non-type III effectors that were not included in all models were collected from Baltrus et al. [25].

At present, most tools are based on the full sequence information of proteins or only 100 C-terminal residues [26]. In previous studies, N-terminal residues have been shown to also provide targeted information for protein transport [27–29], and the target information of T3SEs is usually located in the 50-100 N-terminal residues in different bacteria [30, 31]. Therefore, we have only used the N-terminal sequences in all the following calculations.

2.2. Feature Extraction. The feature input of our model is the combination of one-hot encoding and the PSSM of a protein. Each sequence is transformed into a one-hot matrix with 100 rows and 20 columns and a PSSM matrix with 100 rows and 20 columns, which are integrated into a combination matrix with 200 rows and 20 columns as the feature input. The 20 columns in the one-hot matrix correspond to 20 amino acids. One-hot encoding solves the problem of the classifier not effectively processing attribute data and expands the features to a certain extent. However, compared to PSSM, one-hot encoding is weaker with regard to protein feature extraction. Here, the introduction of PSSM enables the network model to better learn the characteristic preference of proteins, because PSSM features consider the position weight, number, and other parameters of each amino acid in the protein. PSSM also considers evolutionary information, so even the same residue may generate different characteristics, and it can effectively extract information from amino acid sequences. We used the PSI-BLAST [32] search database from UniprotKB/Swiss-Prot [33] to obtain the PSSM of the target protein. The matrix is an L × 20 matrix, where L represents the total number of residues in the target protein’s amino acid sequence. At the same time, we use 1, 2, 3, ..., 20 to represent the individual characters of the ordered 20 basic amino acids and get the corresponding number of columns. In summary, Uᵢᵣ, indicates the possibility that the i position of the amino acid sequence of the target protein is encoded as the basic amino acid j during the evolution process.

2. Materials and Methods

2.1. Dataset. We collected a comprehensive dataset from multiple bacterial species known as T3SEs and non-T3SEs...
2.3. Overview of Attention-CNN Model. The traditional CNN model includes convolution, pooling, and full connection layers, and it can be used to extract the sequence features of proteins. However, the sequence of a protein is more like a piece of text composed of amino acids, and since, when one amino acid may be related to the amino acids around it or those even farther away, it is not enough to extract these features using only the CNN mechanism. We also need to consider the information before and after the protein sequence and the correlation between discontinuous amino acids. Intuitively, an amino acid or a segment of amino acids may have a great impact on the protein sequence, so we can set a higher weight to this or this part of amino acids and have thus introduced the attention layer into the network.

Attention is a network structure model layer based on encode-decode, which has achieved satisfying prediction results compared to other traditional models in many fields including machine translation, picture description, and speech recognition. This implementation of the attention mechanism retains the intermediate output results of the input sequence of an LSTM encoder, then trains a model to selectively learn these inputs and associate the output sequence with them when the model outputs.

We added the attention and full connection layers in parallel after the convolution and pooling layers, so that the model can not only take advantage of the mechanism of attention to learn the front and back features of the sequence, but also use the advantages of CNN feature extraction.

Our framework, ACNNT3, first uses multiple convolution and pooling layers to automatically learn protein sequence features, then takes the output feature vector as the input of the attention layer to calculate the score showing whether the neural network pays attention to the sequence features of the location. We define the output after convolution and pooling as a matrix \( M^o (d \times q) \), where \( d \) is the number of convolution kernels, \( q \) is the whole position after sequence pooling, and the column \( j \) of the feature map matrix \( M^o \) can be viewed as a feature vector (denoted by \( V_j \)). \( W_j \) is the normalized importance score that is used to further average the columns of the feature map matrix \( M^o \). The dense matrix output through the attention network is \( M^a \), i.e.,

\[
M^a = \sum_{j=1}^{q} w_j v_j, \tag{2}
\]

\[
w_j = \frac{\exp (e_j)}{\sum_{j=1}^{q} \exp (e_j)}, \tag{3}
\]

where \( e_j \) is the importance score of the shared network and \( W_j \) is the relevant standardization score.

In order to integrate the features after convolution-pooling and the feature output by the attention layer, we first connect all the values in \( M^o \) and project them linearly to a value that represents the contribution of the whole sequence, represented by \( S^o \), then we concatenate it with the dense representation \( M^a \) and input it into the logistic regression classifier to obtain the prediction score, namely,

\[
\text{Pred}(s) = \text{sigm}(\text{concat}(M^a, S^o)), \tag{4}
\]

where \( s \) represents a position in the integrated sequence.

\[
S^o = \text{dense}(\text{pool}(\text{conv}(\text{encode}(s)))), \tag{5}
\]

where \( \text{encode}(\cdot), \text{conv}(\cdot), \text{pool}(\cdot), \text{concat}(\cdot), \text{dense}(\cdot), \text{and sigmoid}(\cdot) \) represent the unification of one-hot and PSSM encoding, convolution, maximum pooling, concatenation, dense connection, and sigmoid operation, respectively. At the same time, for a specific sequence, we can also output a weight vector, i.e.,

\[
\text{AttMap}(s) = (w_1, \ldots, w_q). \tag{6}
\]

This formula is used to express the attention of the model to each position of the input sequence.

2.4. Model Training. ACNNT3 is composed of a series of modules which use the fusion features of 100 amino acids at the N-terminal of the protein as input to predict T3SEs (Figure 1). The ACNNT3 model consists of convolution, pooling, attention, and fully connected layers. We use cross-validation to train our model and improve the generalization ability. The loss function uses a binary cross entropy loss function, and the optimizer uses the Adam algorithm. In Figure 2, we give the accuracy (ACC) comparison on the independent datasets under different epochs and batches. Since the dataset is not very large, the number of training epochs is set as 50 and the best batch value on the crossvalidation set is 10 as the optimal setting.

2.5. Performance Evaluation. We used 5-fold cross-validation to estimate the classification performance of our model. Namely, we repeated the process five times and recorded the training parameters and average performance parameters for each time. The commonly used evaluation indexes for two-class classification are precision (PRE), sensitivity (SN), specificity (SP), F1 score, accuracy (ACC), and Matthew’s correlation coefficient (MCC):

\[
\text{PRE} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{7}
\]

\[
\text{SN} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{8}
\]

\[
\text{SP} = \frac{\text{TN}}{\text{TN} + \text{FP}}, \tag{9}
\]

\[
\text{F1 score} = 2 \times \frac{\text{TP}}{2 \text{TP} + \text{FP} + \text{FN}}, \tag{10}
\]
\[
\text{ACC} = \frac{TP + TN}{TP + FP + TN + FN}, \quad (11)
\]
\[
\text{MCC} = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP + FN) \times (TN + FP) \times (TP + FP) \times (TN + FN)}}, \quad (12)
\]

where TP, TN, FP, and FN represent the number of true positive, true negative, false positive, and false negative protein datasets, respectively.

The ROC curve is the relationship between the true positive and false positive rates, which is used to measure the comprehensive performance of different methods. The area under the ROC curve (AUC) is commonly used as a summary measure of diagnostic accuracy. In the ROC curve, the horizontal axis is the FPR (false positive rate, i.e., the ratio of wrongly predicted pairs over the total number of negative pairs), and the vertical axis is the TPR (true positive rate, i.e., the ratio of correctly predicted pairs over the total number of positive pairs). The maximum AUC is 1, which means a perfect prediction, and the AUC obtained by a random guess is 0.5.

3. Results

We have constructed a new prediction model to identify T3SEs by using a neural network that combines attention with CNN. In order to study the influence of the negative sample set on performance, we divided the training set into two parts. The positive to negative ratio of training set 1 is 1:1, and the positive to negative ratio of training set 2 is 1:3. The ACNNT3 model was trained using training sets 1 and 2, respectively. To evaluate the classification performance of our ACNNT3 model, we use ROC and AUC as the evaluation criterion. The ROC charts of 5-fold cross-validation curves under training sets 1 and 2 are shown in Figures 3(a) and 3(b). We can see that the ACNNT3 model achieved a good performance on the ROC chart. The mean AUC of the model is 0.95 on training set 1 and 0.98 on training set 2. These results show that our ACNNT3 model can accurately classify T3SEs and non-T3SEs on both training sets.

3.1. Comparison of Different Features on the Same Network.

We take the one-hot single feature and the fused feature containing the one-hot matrix and PSSM as inputs, respectively, using ACNNT3 as the training model, and use the independent dataset to evaluate the two models. The results show that in all evaluation indexes, the model with the fusion feature is superior to the one with single feature training, thus verifying the proposed fusion feature’s effectiveness (Figure 4). Compared to the one-hot single feature, the fusion feature is more comprehensive for the extraction of protein sequence information, and it can be seen from the
experimental results that two types of features have good compatibility with each other.

### 3.2. Comparison of Different Deep-Learning Methods

We compared the results from different popular network models using the independent dataset with the same feature input, as shown in Table 1. For a class of sequential processing problems, the addition of an attention layer makes the network model strengthen the connection before and after the amino

---

**Table 1: Comparison with mainstream deep-learning methods.**

<table>
<thead>
<tr>
<th>Method</th>
<th>PRE</th>
<th>F1 score</th>
<th>ACC</th>
<th>MCC</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACNNT3</td>
<td><strong>0.919</strong></td>
<td><strong>0.944</strong></td>
<td><strong>0.967</strong></td>
<td><strong>0.922</strong></td>
<td><strong>0.968</strong></td>
</tr>
<tr>
<td>DenseNet</td>
<td>0.850</td>
<td>0.907</td>
<td>0.942</td>
<td>0.870</td>
<td>0.951</td>
</tr>
<tr>
<td>VGG16</td>
<td>0.846</td>
<td>0.892</td>
<td>0.934</td>
<td>0.847</td>
<td>0.937</td>
</tr>
<tr>
<td>ResNet</td>
<td>0.609</td>
<td>0.691</td>
<td>0.838</td>
<td>0.552</td>
<td>0.795</td>
</tr>
<tr>
<td>CNN</td>
<td>0.780</td>
<td>0.842</td>
<td>0.901</td>
<td>0.776</td>
<td>0.904</td>
</tr>
<tr>
<td>LSTM</td>
<td>0.875</td>
<td>0.933</td>
<td>0.959</td>
<td>0.909</td>
<td>0.961</td>
</tr>
</tbody>
</table>

The bold values indicate the best prediction results.

---

**Table 2: Comparison of ACNNT3 and DeepT3, Effective T3, BPBAac, and BEAN2 on an independent dataset.**

<table>
<thead>
<tr>
<th>Method</th>
<th>PRE</th>
<th>SN</th>
<th>SP</th>
<th>F1 score</th>
<th>ACC</th>
<th>MCC</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACNNT3-1</td>
<td>0.919</td>
<td><strong>0.971</strong></td>
<td>0.965</td>
<td><strong>0.944</strong></td>
<td><strong>0.967</strong></td>
<td><strong>0.922</strong></td>
<td><strong>0.968</strong></td>
</tr>
<tr>
<td>ACNNT3-2</td>
<td>0.711</td>
<td>0.914</td>
<td>0.849</td>
<td>0.800</td>
<td>0.868</td>
<td>0.716</td>
<td>0.882</td>
</tr>
<tr>
<td>DeepT3-1</td>
<td>0.825</td>
<td>0.943</td>
<td>0.919</td>
<td>0.880</td>
<td>0.926</td>
<td>0.830</td>
<td><strong>0.974</strong></td>
</tr>
<tr>
<td>DeepT3-2</td>
<td>0.643</td>
<td>0.771</td>
<td>0.825</td>
<td>0.701</td>
<td>0.810</td>
<td>0.569</td>
<td>0.896</td>
</tr>
<tr>
<td>Effective T3</td>
<td>0.542</td>
<td>0.839</td>
<td>0.741</td>
<td>0.658</td>
<td>0.767</td>
<td>0.521</td>
<td>0.803</td>
</tr>
<tr>
<td>BPBAac</td>
<td><strong>0.944</strong></td>
<td>0.548</td>
<td><strong>0.988</strong></td>
<td>0.694</td>
<td>0.871</td>
<td>0.656</td>
<td>0.902</td>
</tr>
<tr>
<td>BEAN2</td>
<td>0.674</td>
<td>0.935</td>
<td>0.835</td>
<td>0.784</td>
<td>0.862</td>
<td>0.706</td>
<td>0.865</td>
</tr>
</tbody>
</table>

The bold values indicate the best prediction results.

---

**Table 3: Comparison of ACNNT3 and DeepT3, Effective T3, BPBAac, and BEAN2 on a P. syringae dataset.**

<table>
<thead>
<tr>
<th>Method</th>
<th>PRE</th>
<th>SN</th>
<th>SP</th>
<th>F1 score</th>
<th>ACC</th>
<th>MCC</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACNNT3-1</td>
<td>0.900</td>
<td>0.976</td>
<td>0.357</td>
<td>0.936</td>
<td><strong>0.887</strong></td>
<td>0.452</td>
<td>0.667</td>
</tr>
<tr>
<td>ACNNT3-2</td>
<td><strong>0.872</strong></td>
<td><strong>0.988</strong></td>
<td>0.143</td>
<td>0.926</td>
<td>0.866</td>
<td>0.265</td>
<td>0.565</td>
</tr>
<tr>
<td>DeepT3-1</td>
<td>0.905</td>
<td>0.962</td>
<td>0.429</td>
<td>0.932</td>
<td>0.884</td>
<td><strong>0.472</strong></td>
<td><strong>0.838</strong></td>
</tr>
<tr>
<td>DeepT3-2</td>
<td><strong>0.913</strong></td>
<td>0.924</td>
<td><strong>0.500</strong></td>
<td>0.918</td>
<td>0.860</td>
<td>0.437</td>
<td>0.763</td>
</tr>
<tr>
<td>Effective T3</td>
<td>0.906</td>
<td>0.906</td>
<td>0.428</td>
<td>0.906</td>
<td>0.838</td>
<td>0.334</td>
<td>0.810</td>
</tr>
<tr>
<td>BPBAac</td>
<td>0.875</td>
<td>0.494</td>
<td><strong>0.571</strong></td>
<td>0.631</td>
<td>0.505</td>
<td>0.046</td>
<td>0.562</td>
</tr>
<tr>
<td>BEAN2</td>
<td>0.883</td>
<td>0.988</td>
<td>0.083</td>
<td><strong>0.938</strong></td>
<td>0.884</td>
<td>0.271</td>
<td>0.607</td>
</tr>
</tbody>
</table>

The bold values indicate the best prediction results.
acid and the attention of important information in the sequence. From the experimental results, it can be seen that our network model ACNNT3 is better than the existing deep-learning framework for predicting T3SEs in many indicators.

3.3. Comparison with Existing Methods. In order to evaluate the effectiveness of our method, we compared the ACNNT3 performance with four popular methods, DeepT3 [11], BPBAac [22], Effective T3 [6], and BEAN2 [34], on the same independent dataset. The parameter settings of these methods are the same as those used by Li et al. [11]. We found that our ACNNT3-1 model has a higher SN, F1 score, ACC, and MCC than the other four methods (Table 2). The results also show that our method achieved satisfactory performance in almost all indicators. For the important index of ACC, the accuracy of ACNNT3-1 is 0.967, which is 9.9%, 4.1%, 15.7%, 20.0%, 9.6%, and 10.5% higher than ACNNT3-2, DeepT3-1, DeepT3-2, Effective T3, BPBAac, and BEAN2, respectively. In another P. syringae dataset, our model still performed better than the existing methods on the index of ACC (Table 3). The accuracy of ACNNT3-1 is 0.887. We selected the best model in the fivefold crossvalidation and used the independent and P. syringae datasets to test it. We also obtained the ROC curves of the model on two test sets (Figure 5). Overall, our method has been shown to be superior to all the latest methods in T3SE prediction and is reliably stable.

4. Conclusion

We have proposed a new prediction model for Gram-negative bacteria type III secreted proteins based on a deep neural network. In order to better learn the feature preference of type III secreted proteins, we integrated the one-hot encoding and PSSM extracted from the protein primary sequence as the feature input and embedded the attention layer into CNN to improve the model’s prediction ability. This method outperforms other existing methods on most indicators, and using feature and network model comparisons, we have shown its advantages. In comparison with other popular methods, ACNNT3 is more accurate at predicting and recognizing T3SEs in the independent test set, which reflects its advantages and effectiveness. However, we found that ACNNT3’s performance using the P. syringae dataset is not particularly obvious and was only slightly higher than the previous methods in terms of ACC and MCC. Our work in the future will focus on achieving better results in other experimental indicators and on applying this model for prediction using other large-scale datasets.

For easy implementation, all data used in this work and the source code for feature computing can be accessible at https://github.com/Lijiesky/ACNNT3.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant Nos. 11671009 and 61762035, Zhejiang Provincial Natural Science Foundation of China under Grant Nos. LZ19A010002 and LY18F020027.

References
