TCN-HBP: A Deep Learning Method for Identifying Hormone-Binding Proteins from Amino Acid Sequences Based on a Temporal Convolution Neural Network

Jing Guo
School of Electrical and Information Engineering, Tianjin University, Tianjin, China
Email: guojing9667@tju.edu.cn

Abstract. Hormone-binding proteins (HBPs) are carrier proteins that specifically bind to targeted hormones. Some evidence suggests that the abnormal expression of HBPs causes various diseases. Therefore, it is significant to accurately identify HBPs to study these diseases. Recently, many researchers have proposed traditional machine learning methods to complete this work, but these methods are neither suitable for training on large-scale datasets nor take into account the contextual features of HBPs. In this paper, I propose a new deep learning method, TCN-HBP, to distinguish HBPs. TCN-HBP consists of a coding layer, embedding layer, convolutional neural network (CNN) layer and temporal convolutional network (TCN) layer. The coding and embedding layers extend the protein sequences into two-dimensional matrix data. The CNN layer convolves the matrix data to form feature maps. The TCN layer captures the contextual features present in the feature maps. Experiments show that the data generalization capabilities and recognition accuracy (99.15%) of TCN-HBP on large datasets perform better than previous methods.

Keywords. Hormone-binding proteins; amino acid sequences; deep learning; temporal convolutional network.

1. Introduction
Hormone-binding proteins (HBPs) produce specific expression in cells because they can bind specifically to targeted hormones [1, 2], as shown in figure 1. Hormone binding is divided into many types, and they take on different tasks in the human body. For example, abnormalities of thyroid hormones in serum are closely related to the functional expression of some thyroid HBPs [3], and sex hormone-binding globulins control the steroid levels in plasma [4]. However, some evidence suggests that multifarious diseases in the human body arise due to the deviant expression of HBPs [5]. Therefore, it is significant to accurately identify HBPs to study these diseases and hormone regulation.

Due to the high cost of identifying protein types based on biological experiments, many researchers have proposed computational methods to complete this work in recent years. For example, Tang et al. introduced a predictor called HBPred, which extracts the dipeptide composition from the protein residue sequences and predicts HBPs through an incremental feature selection strategy and support vector machine (SVM) [6]. Wang et al. used feature rating techniques and an SVM to predict HBPs, which can remove redundant information in the feature set and improve accuracy [7]. Basith et al. developed a method to identify HBPs using an extremely randomized tree, called iGHBP [8]. Akbar et al. used location-specific scoring matrices and SVM to identify HBPs [9].

The computational methods proposed by these scholars belong to the category of machine learning, and their experiments were carried out on hundreds of protein sequences. Traditional machine learning
methods show some unique advantages in training small-scale datasets [10, 11]. However, as the research time increases, the number of proteins also increases rapidly, and these methods are not suitable for large-scale dataset research [12].

Figure 1. Hormones specifically bind to HBPs [13].

Facing the explosive growth in the number of newly discovered proteins and the dilemma of machine learning methods, some scholars have used deep learning methods to identify protein types [14, 15]. Deep learning methods consist of complex neural networks suitable for large-scale data training and have achieved great success in image recognition, intelligent speech and natural language processing [16-18]. The temporal convolutional network (TCN) is an innovative model for natural language processing proposed by Bai et al. in 2018 that can efficiently excavate contextual features in sequence information [19]. Studies by some scholars have shown that the contextual features of primary protein sequences (amino acid sequences) have an important influence on functional expression [20-22].

In this treatise, I introduce a neoteric approach, TCN-HBP, to identify HBPs. TCN-HBP comprises a coding layer, embedding layer, convolutional neural network (CNN) layer and TCN layer. The coding and embedding layers extend the protein sequences into two-dimensional matrix data. The CNN layer convolves the matrix data to form feature maps. The TCN layer captures the contextual features present in the feature maps. I collected 27,682 protein sequences in the Universal Protein Resource (UniProt) and National Center for Biotechnology Information (NCBI) database [24]. Our use of the data complies with the provisions of the above database. After searching with the keyword “hormone binding”, a total of 13,749 hormone-binding proteins were collected. There are 123 sequences from reference [7] also added to the dataset. After redundant processing was performed, 13,841 sequences were used as positive samples. Balanced datasets are very helpful in model training and do not bias the judgement of positive samples. To collect negative samples marked by experts, the keyword “not hormone binding” was selected from the UniProt database in this article and 13,841 data points were collected as negative samples.

2. Materials
Special attention should be given to the data collection process when conducting biological data analysis. Since this article is aimed at protein identification under big data, it is necessary to collect as much data as possible. The data are from the UniProt database [23] and the NCBI database [24]. Our use of the data complies with the provisions of the above database. After searching with the keyword “hormone binding”, a total of 13,749 hormone-binding proteins were collected. There are 123 sequences from reference [7] also added to the dataset. After redundant processing was performed, 13,841 sequences were used as positive samples. Balanced datasets are very helpful in model training and do not bias the judgement of positive samples. To collect negative samples marked by experts, the keyword “not hormone binding” was selected from the UniProt database in this article and 13,841 data points were collected as negative samples.
I selected 2,000 amino acid sequences from the total dataset as an independent sample set to test our model. The remaining samples were used for training, of which 90% formed the training set and 10% formed the cross-validation set (see table 1 for details).

Table 1. Composition of experimental datasets.

| Data           | Positive | Negative | Total  |
|----------------|----------|----------|--------|
| Train_set      | 11,557   | 11,557   | 23,114 |
| Val_set        | 1,284    | 1,284    | 2,568  |
| Indepent-test_set | 1,000    | 1,000    | 2,000  |

3. Methods
The TCN-HBP comprises coding layer, embedding layer, CNN layer and TCN layer.

3.1. Coding Layer
In this process, the protein sequences consisting of amino acids were encoded into a list of numbers. The amino acid sequences converted into number lists could be easily manipulated mathematically, and they also laid the foundation for the next step in the embedding layer (see figure 2 for details).

3.2. Embedding Layer
Word embedding is a distributed acceptance expression that can map words to low-dimensional vectors. The superiority of applying word embedding is that it can capture the word semantics or the relationships between words. In the embedding stage, each number in the list is embedded in a consecutive vector space. After this process, the different vectors represent each input amino acid, and the protein sequences are transformed into a digital matrix (see figure 3 for details).

The calculation process is shown in equation (2).
\[ S_z = \text{Embedding}(S_z) = \begin{bmatrix} 0.7 & -0.7 & 0.2 & 0.1 & 0.4 & 0.7 & 0.9 & 0.7 \\ 0.1 & 0.6 & 0.3 & -0.5 & 0.6 & 0.9 & 0.8 & -0.3 \\ 0.9 & -0.3 & 0.1 & 0.4 & -0.3 & 0.1 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.9 & -0.3 & 0.3 & -0.9 & 0.1 & -0.1 \\ 0.2 & 0.6 & -0.2 & 0.9 & 0.4 & 0.1 & -0.9 & 0.6 \\ -0.6 & -0.2 & 0.1 & 0.1 & -0.1 & -0.7 & 0.2 & 0.9 \\ 0.9 & -0.2 & 0.5 & 0.5 & -0.2 & -0.1 & 0.3 & -0.1 \\ 0.2 & -0.9 & 0.8 & -0.6 & 0.6 & 0.9 & -0.9 & 0.8 \end{bmatrix} \] (2)

3.3. CNN Layer

The CNN extracts potential features in two-dimensional data through convolutional mapping operations [25]. Recently, CNNs have been applied to natural language processing, using filters to obtain the characteristics of natural statements expressed in a two-dimensional matrix [26]. In this paper, the amino acids in protein sequences are treated as words in natural statements (see figure 4 for details).

\[ \begin{align*} S_z &= \text{MACATIAN} \\ S_z &= \text{Embedding}(S_z) \\ S_i &= \text{Conv}(S_z) \\ S_i &= \text{Filter}(S_i) \end{align*} \]

**Figure 3.** Embedding layer.

\[ \begin{align*} S_z &= \text{MACATIAN} \\ S_z &= \text{Embedding}(S_z) \\ S_i &= \text{Conv}(S_z) \\ S_i &= \text{Filter}(S_i) \end{align*} \]

**Figure 4.** CNN layer.

The filter in the above figure contains the weight parameters, and the calculation results for \( S_z \) are as follows.

\[ S_i = \text{CNN}(S_z) = \begin{bmatrix} 0.9 & -0.2 & 0.9 & -0.5 \\ 0.1 & 0.7 & -0.1 & 0.3 \\ -0.2 & -0.3 & 0.6 & 0.2 \\ -0.2 & -0.9 & 0.9 & -0.9 \end{bmatrix} \] (3)
The CNN captures the features of the two-dimensional data through convolutional kernel operations, obtains new feature maps, and simplifies the calculation while acquiring the sequence features.

3.4. TCN Layer
The recurrent neural network is a classical algorithm that captures sequence features before and after statement dependence. However, it often encounters the problem of vanishing or exploding gradients, influenced by the sequence length [27, 28]. TCN provides a different method for calculating the hidden state, which overcomes the above problems by introducing gating and storage mechanisms [19], as shown in figure 5.

![Figure 5. TCN layer.](image)

As shown in the figure above, sequence $X_0, ..., X_T$ is the TCN input, and sequence $Y_0, ..., Y_T$ is the TCN output. The output of a single node in the network, such as $Y_T$, is associated with the entire input sequence. A TCN can process the sequence in parallel without processing in order as a recurrent neural network [19].

After $S_3$ passes TCN, I rate the following function to determine whether the amino acid sequence is an HBP.

$$F = TCN(CNN(Embedding(Encoding(S))))$$ (4)

3.5. TCN-HBP Model
The composition of TCN-HBP is shown in figure 6. The coding layer uses 1-20 different numbers to encode 20 amino acids, and the purpose of the embedding layer is to convert the coded sequence into a feature matrix with a higher dimension to enter the CNN layer. The CNN layer has two steps: a convolution operation and a pooling operation. In the CNN layer, the matrix output in the embedding layer is scanned by the filters to obtain new feature maps, and the subsequent feature maps are pooled to obtain the main feature information. The TCN layer can perform another deeper extraction of the feature output from the convolutional layer to mine contextual features in HBPs.

![Figure 6. The overall structure of TCN-HBP model.](image)
3.6. Model Parameters
The adjustable parameters in the model will have a great effect on the training process and results of the model. The parameter selection is shown below.

- The parameter selection in the embedding layer is 26.
- In the three CNN layers, the filter numbers are 64, 64, and 64, the kernel sizes are 13, 7, and 5, and the activation selection is ReLU.
- The pooling-size parameters in the three-layer pooling layer are all 2.
- In the two TCN layers, the filter selections are 64, 64, the kernel sizes are 13, 5, the stack selection is 1, the drop-rate is 0.1, and the activation is ReLU.
- The batch size is 64, the epoch size is 50, and the learning rate is 0.01.

4. Results and Discussion

4.1. Training Results
To avoid overfitting the model training and to find a more suitable model for HBP recognition, I set up 10 k cross-validation, which is applied to both the balanced experiment and the unbalanced experiment. The number of negative samples in the unbalanced experiment is three times that of the positive samples. The numbers of positive and negative samples in the balanced experiment are equivalent. The number of epochs for the two training models is set to 50. Table 2 shows the training result.

| Dataset        | Unbalanced experiment accuracy | Balanced experiment accuracy |
|----------------|-------------------------------|-----------------------------|
| Training set   | 97.90%                        | 99.08%                      |
| Validation set | 97.07%                        | 98.03%                      |

From the experimental results, it can be seen that the training accuracy of the balanced dataset can reach 99.08%, and the accuracy of the verification set can reach 98.03%.

During training, the model loss declines rapidly, and the model accuracy improves steeply before 5 epochs. As training continues, both training loss and accuracy are stable with specific intervals (figure 7).

4.2. Performance in Independent Samples Data Set and Comparison with Previous Models
I used the sensitivity (SN), specificity (SP), accuracy (ACC) and Matthews correlation coefficient (MCC) to evaluate the performance of a model on an independent test set. They are defined as follows:

\[ Acc = \frac{T_p + T_n}{T_p + T_n + F_p + F_n} \]  
\[ S_N = \frac{T_p}{T_p + F_n} \]  
\[ S_P = \frac{T_n}{T_n + F_p} \]  
\[ MCC = \frac{TT - FP}{\sqrt{(TP + FP)(TN + FN)(TP + FN)(TN + FP)}} \]

The four variables \(T_P, T_N, F_P\) and \(F_N\) in the above formula represent the number of true positive, true negative, false positive and false negative samples, respectively.
I compared TCN-HBP with several previous models identifying HBPs using independent test samples in reference [9]. The models are HBPred [7], iGHBP [9], HBpred2.0 [29] and iHBP-DeepPSSM [10]. I use the above four standards to estimate the representation of various models on independent test sets. The results show that the TCN-HBP outperforms HBpred, iGHBP and HBpred2.0 testing on independent datasets from the literature [9] and is close to that of iHBP-DeepPSSM. However, unlike the other four approaches, TCN-HBP is a method based on big data training. Therefore, I used an independent test dataset consisting of 1,000 positive samples and 1,000 negative samples from UniProt and NCBI to test the TCN-HBP (see table 3 for details).

### Table 3. Performance in independent datasets.

| Method          | Samples number | ACC     | SN      | SP      | MCC     |
|-----------------|----------------|---------|---------|---------|---------|
| HBpred          | 62             | 68.48%  | 80.43   | 56.52   | -       |
| iGHBP           | 82.31%         | 80.71   | 83.90   | 0.650   |         |
| HBpred2.0       | 84.78%         | 89.18   | 80.43   | 0.698   |         |
| iHBP-DeepPSSM   | 92.31%         | 93.89   | 82.01   | 0.860   |         |
| TCN-HBP         | 91.93%         | 83.87   | 100.00  | 0.850   | 0.983   |
| (2000 from UniProt and NCBI) | 99.15%   | 98.80   | 99.50   | 0.983   |

As shown in table 3, TCN-HBP has ACC, SN, SP and MCC values of 99.15%, 98.80, 99.50 and 0.983, respectively, on independent test sets containing 2,000 samples, which outperforms the other four
models. The test results of TCN-HBP on thousands of datasets are more convincing compared to the other models tested with dozens of datasets.

4.3. Comparison to Other Deep Learning Models
To further investigate TCN-HBP, I trained a pure convolutional network model (CNN model) and a CNN-LSTM model on the same dataset [30]. The training and test comparison results are shown in the table below.

| Model          | Train accuracy | Test accuracy |
|----------------|----------------|---------------|
| CNN Model      | 63.76%         | 60.66%        |
| CNN-LSTM Model | 98.03%         | 98.25%        |
| TCN-HBP        | 99.08%         | 99.15%        |

Table 4. Comparison with other models.

Obviously, compared with the CNN model, our model significantly improves training accuracy and test accuracy. Compared with the CNN+LSTM model, the test accuracy of TCN-HBP is 0.9% higher, which proves that our model has higher applicability than the CNN-LSTM model in the problem of HBP recognition.

I visualized the training process of the above three models in figure 5 so that I can compare the three models more clearly to see the differences. The training accuracy of the CNN Model cannot be improved with 10 epochs, and it is stable at approximately 63%. Although the training accuracy of the CNN-LSTM model is only slightly lower than that of TCN-HBP, its model accuracy and loss show large fluctuations in the later stage of the training process. Therefore, TCN-HBP is more robust in the field of protein sequence recognition than the CNN and CNN-LSTM models (figure 8).

![Figure 8. Comparison of the training process of the three models.](image)

5. Conclusions
The identification of HBPs is one of the leading issues currently studied by biologists. In this article, I introduced a novel deep learning technique to help biologists reduce preliminary work and improve the
efficiency of identifying HBP s. For potential contextual features in protein sequences, our method excavates them by combining CNN and TCN to improve recognition accuracy. Experimental results show that the data generalization capabilities and recognition accuracy of our method on large datasets are better than those of previous methods.

Acknowledgments
I would like to extend my deep gratitude to all those who have offered me practical, cordial and selfless support in writing this thesis.

References
[1] Baumann G 2002 Growth hormone binding protein: The soluble growth hormone receptor *Minerva Endocrinol* **27** 265-76.
[2] Dhiraviam K N, Balasubramanian S and Jayavel S 2018 Indole alkaloids as new leads for the design and development of novel DPP-IV inhibitors for the treatment of diabetes *Curr. Bioinform.* **13** 157-69.
[3] Mimoto M S and Refetoff S 2020 Clinical recognition and evaluation of patients with inherited serum thyroid hormone-binding protein mutations *Endocrinol. Invest* **43** 31-41.
[4] da Silva A J and Dos Santos E S 2018 Aqueous solution interactions with sex hormone-binding globulin and estradiol: A theoretical investigation *J. Biol. Phys.* **44** 539-56
[5] Kraut J A and Madias N E 2017 Adverse effects of the metabolic acidosis of chronic kidney disease *Adv. Chronic Kidney Dis.* **24** 289-97.
[6] Tang H, Zhao Y W, Zou P, Zhang C M, Chen R, Huang P and Lin H 2018 HBPred: A tool to identify growth hormone-binding proteins *Int. J. Biol. Sci.* **14** 957-64.
[7] Wang K, Li S, Wang Q and Hou C 2019 Identification of hormone-binding proteins using a novel ensemble classifier *Computing* **101** 693-703.
[8] Basith S, Manavalan B, Shin T H and Lee G 2018 iGHBP: computational identification of growth hormone binding proteins from sequences using extremely randomised tree *Comput. Struct. Biotechnol. J.* **16** 412-20.
[9] Akbar S, Khan S, Ali F, Hayat M, Qasim M and Gul S 2020 iHBP-DeepPSSM: identifying hormone binding proteins using PsePSSM based evolutionary features and deep learning approach *Chemom. Intell. Lab. Syst.* **204** 104103.
[10] Yu X, Cao J, Cai Y, Shi T and Li Y 2006 Predicting rRNA-, DNA-, and RNA-binding proteins from primary structure with support vector machines *J. Theor. Biol.* **240** 175-84.
[11] Bhardwaj N, Langlois R E, Zhao G and Lu H 2005 Kernel-based machine learning protocol for predicting DNA-binding proteins *Nucleic Acids Res.* **33** 6486-93.
[12] Qiu J, Wu Q, Ding G, Xu Y and Feng S 2016 A survey of machine learning for big data processing *EURASIP J. Adv. Signal Process.* **2016** 1-16.
[13] Sundström M, Lundqvist T, Rödin J, Giebel L B, Milligan D and Norstedt G 1996 Crystal structure of an antagonist mutant of human growth hormone, G120R, in complex with its receptor at 2.9 A resolution *J. Biol. Chem.* **271** 32197-203.
[14] Alipanahi B, Delong A, Weirauch M T and Frey B J 2015 Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning *Nat. Biotechnol.* **33** 831-8.
[15] Zeng H, Edwards M D, Liu G and Gifford D K 2016 Convolutional neural network architectures for predicting DNA-protein binding *Bioinformatics* **32** i121-7.
[16] Krizhevsky A, Sutskever I and Hinton G E 2012 Imagenet classification with deep convolutional neural networks *Adv. Neural Inf. Process. Syst.* **25** 1097-1105.
[17] Graves A, Mohamed A and Hinton G 2013 Speech recognition with deep recurrent neural networks *IEEE International Conference on Acoustics, Speech and Signal Processing* (Vancouver, BC, Canada: IEEE) pp 6645-9.
[18] Sutskever I, Vinyals O and Le Q V 2014 Sequence to sequence learning with neural networks arXiv preprint arXiv:1409.3215.
[19] Bai S, Kolter J Z and Koltun V 2018 An empirical evaluation of generic convolutional and recurrent networks for sequence modeling *arXiv preprint arXiv:1803.01271*.

[20] Yaseen A and Li Y 2014 Context-based features enhance protein secondary structure prediction accuracy *J. Chem. Inf. Model.* 54 992-1002.

[21] Garnier J, Gibart J F and Robson B 1996 GOR method for predicting protein secondary structure from amino acid sequence *Methods Enzymol.* 266 540-53.

[22] Starosta A L, Lassak J, Peil L, Atkinson G C, Virumäe K, Tenson T, Remme J, Jung K and Wilson D N 2014 Translational stalling at polyproline stretches is modulated by the sequence context upstream of the stall site *Nucleic Acids Res.* 42 10711-9.

[23] Pichler K, Warner K, Magrane M and Consortium U 2018 SPIN: Submitting sequences determined at protein level to UniProt *Curr. Protoc. Bioinform.* 62 e52.

[24] Pruitt K D, Tatusova T and Maglott D R 2005 NCBI Reference Sequence (RefSeq): A curated non-redundant sequence database of genomes, transcripts and proteins *Nucleic Acids Res.* 33 D501-4.

[25] Goodfellow I, Bengio Y and Courville A 2016 *Deep Learning* (Cambridge, MA, USA: MIT Press).

[26] Kalchbrenner N, Grefenstette E and Blunsom P 2014 A convolutional neural network for modelling sentences *arXiv preprint arXiv:1404.2188*.

[27] Graves A 2013 Generating sequences with recurrent neural networks *arXiv preprint arXiv:1308.0850*.

[28] Pascanu R, Mikolov T and Bengio Y 2012 Understanding the exploding gradient problem *CoRR* abs/1211.5063 21.

[29] Tan J X, Li S H, Zhang Z M, Chen C X, Chen W, Tang H and Lin H 2019 Identification of hormone binding proteins based on machine learning methods *Math. Biosci. Eng.* 16 2466-80.

[30] Qu Y H, Yu H, Gong X J, Xu J H and Lee H S 2017 On the prediction of DNA-binding proteins only from primary sequences: A deep learning approach *PLoS One* 12 e0188129.