Letter to the Editor

An Improved Method for Predicting Linear B-cell Epitope Using Deep Maxout Networks*

LIAN Yao[1], HUANG Ze Chi[2], GE Meng[3], and PAN Xian Ming[1,#]

To establish a relation between an protein amino acid sequence and its tendencies to generate antibody response, and to investigate an improved *in silico* method for linear B-cell epitope (LBE) prediction. We present a sequence-based LBE predictor developed using deep maxout network (DMN) with dropout training techniques. A graphics processing unit (GPU) was used to reduce the training time of the model. A 10-fold cross-validation test on a large, non-redundant and experimentally verified dataset (Lbtope_Fixed_ non_redundant) was performed to evaluate the performance. DMN-LBE achieved an accuracy of 68.33% and an area under the receiver operating characteristic curve (AUC) of 0.743, outperforming other prediction methods in the field. A web server, DMN-LBE, of the improved prediction model has been provided for public free use. We anticipate that DMN-LBE will be beneficial to vaccine development, antibody production, disease diagnosis, and therapy.

The humoral immune response is based on the amazing ability of antibodies to recognize and bind to antigens of intruding organisms, such as bacteria and viruses. Antibodies bind specifically to either a contiguous amino acid sequence of a protein known as the linear B-cell epitope (LBE), or to a folded structure formed by discontinuous amino acids known as the conformational B-cell epitope[1]. Predicting LBEs is difficult but important for immunological applications. Specifically, predicted LBEs can be synthesized and substituted for intact antigen molecules for detecting anti-protein antibodies in immunoassays, as immunogens for raising anti-peptide antibodies to cross-react with proteins of interest, or in the development of synthetic peptide vaccines.

The trailblazing propensity-based LBE prediction models were fairly simple in which a single or combined multiple physicochemical properties (for example, flexibility, solvent accessibility) were utilized to profile epitope propensity over antigen's primary sequence. Predictive quality of these methods was questioned in 2005 in a study by Blythe and Flower. They analyzed the predictive performance of 484 amino acid propensity scales on 50 antigens and determined that these propensity profiling methods performed only slightly better than random. The more sophisticated knowledge-based methods were explored to improve the prediction performance[2]. Such models included recurrent neural network, hidden Markov model, and naïve Bayes. In recent years, support vector machine (SVM) method was widely applied. These methods differ in the features extracted from the input epitope sequence, the size of datasets that were used to train the SVM model, and the type of SVM kernel function used. These knowledge-based methods have two major limitations including relatively small dataset (-1000 LBEs and non-LBEs) and inaccurate dataset (non-LBEs were random peptides instead of experimentally verified non-LBEs). LBtope method exploited the availability of several thousand experimentally verified LBEs and non-LBEs. Based on the large and experimentally verified dataset of Lbtope_Fixed_non_redundant (LFNR), LBtope model achieved an area under the receiver operating characteristic curve (AUC) of 0.688 (≈0.69). Using a multiple linear regression (MLR) method on the same LFNR dataset, the EPMLR model reported an AUC of 0.616 (≈0.62). Thus, utilizing a large and accurate dataset is critical for future LBE prediction method development. The modest levels of predictive performance (AUCs less than 0.70) also indicates a need for improved new predictive models and further research in the field.

doi: 10.3967/bes2015.065

*This study was supported by grant 2009CB918801 from the Ministry of Science and Technology of China

1. The Key Laboratory of Bioinformatics, Ministry of Education, School of Life Sciences, Tsinghua University, Beijing 100084, China; 2. Key Laboratory of Protein and Peptide Pharmaceutical, Institute of Biophysics, Chinese Academy of Sciences, Beijing 100101, China; 3. CAS Key Laboratory of Genome Sciences and Information, Beijing Institute of Genomics, Chinese Academy of Sciences, Beijing 100101, China

In this research, we investigated the deep architecture of a deep maxout network (DMN), DMN-LBE (Deep Maxout Networks for Linear B-cell Epitope prediction), combined with deep learning training technique of dropout. DMN is a modification of the deep neural network (DNN) that employs a new activation function called maxout[3]. DMN has been shown to improve the accuracy of dropout's fast approximate model averaging technique. We exploited a graphics processing unit (GPU) to accelerate the DMN training. The dipeptide composition features extracted from the primary amino acid sequence information were used to characterize and differentiate LBEs from non-LBEs. We evaluated the effectiveness of DMN-LBE using 10-fold cross-validation on a large, non-redundant and experimentally verified dataset, the LFNR dataset. The results demonstrated an accuracy of 68.33% and an AUC of 0.743, showing that DMN-LBE outperforms previous reported models.

The Immune Epitope Database (IEDB)[4] contains a large number of up-to-date experimentally verified epitopes and non-epitopes. It is the most commonly used and most authoritative database for epitope prediction. Currently, there are three large LBE datasets derived from the IEDB: the BEOracle dataset, the SVMTriP dataset and the LBtope dataset. New LBEs are continuously added to the IEDB, leading to an increasing number of LBEs contained in this database. The most recently curated dataset (LBtope) not only covers the LBE/non-LBE patterns of previous datasets (BEOracle and SVMTriP) but also includes new patterns discovered through subsequent biological experiments. The LBtope dataset is used for this study.

Within the LBtope dataset, five LBE sub-datasets have been created: Lbtope_Fixed (LF), Lbtope_ Fixed_non_redundant (LFNR), Lbtope_Variable (LV), Lbtope_Variable_non_redundant (LVNR) and Lbtope_Confirm (LC). The LF sub-dataset contains 12,063 LBEs and 20,589 non-LBEs, but does not have a redundancy reduction process. Similar sequences can exist in the LF dataset, lead to unreliable prediction models[5]. The LFNR sub-dataset contains 7,824 LBEs and 7,853 non-LBEs of a fixed length (20-mer). It was created after an 80% non-redundant process on the LF sub-dataset. The LV, LVNR, and LC sub-datasets do not have a fixed length, whereas most LBE prediction methods require patterns of a fixed length. Furthermore, the LV and LC sub-datasets do not have redundancy reduction processes. Given the above considerations, the LFNR sub-dataset is used for building our LBE prediction model.

The LFNR sub-dataset was randomly partitioned into 10 equal-sized subsets: 8 subsets were used as a training set to train the model, 1 subset was used as a validation set to select the best model, and the remaining 1 subset was used as an independent testing set. This train-validate-test process was repeated 10 times; each of the 10 LFNR subsets was used exactly once as the independent testing set. The 10 results from the independent testing set were averaged to produce the final estimation.

The dipeptide composition features derived from the primary protein sequence were used to characterize and differentiate LBEs from non-LBEs. The dipeptide composition provides the global information on the LBE sequences in the form of a 400 D (20×20) fixed-length vector. This vector encapsulates the information about the fractions of two consecutive amino acids. The dipeptide composition of each sequence was calculated using the following Equation:

$$\text{Fraction of dipeptide } (i) = \frac{\text{Total number of dipeptide } (i)}{\text{Total number of all possible dipeptides}} \quad (1)$$

where dipeptide (*i*) is one dipeptide i out of 400 dipeptides.

A DNN is an artificial neural network with several layers of hidden nodes between the input and output layers. As a feed-forward architecture, a standard DNN can be computed as follows:

$$\mathbf{h}^{l+1} = \sigma(\mathbf{W}^l \mathbf{h}^l + \mathbf{b}^l), \quad 1 \le l \le L \quad (2)$$

where $\mathbf{h}^{l+1}$ is the vector of inputs to the l+1 layer and σ is the activation function. L is the total number of hidden layers, $\mathbf{h}^l$ is the output vector of the hidden layer l, and w^l and $\mathbf{b}^l$ are the weight matrix and bias vector of layer l, respectively.

A DMN is a modification of a DNN architecture where the maxout activation function is used for σ in Equation 1. In a DMN, each hidden node takes the maximum value over the k nodes of a group. The output of the hidden node i of layer l+1 can be computed as follows:

$$h_i^{(l+1)} = \max_{j \in 1,\ldots k} z_{ij}^{(l+1)}, \quad 1 \le l \le L \quad (3)$$

where the values of $z^{(l+1)}_{ij}$ are the lineal pre-activation values from the layer l:

$$\mathbf{z}^{(l+1)} = \mathbf{W}^{(l)} \mathbf{h}^{(l)} + \mathbf{b}^{(l)} \quad (4)$$

The max-pooling operation is applied over the $z^{(l+1)}$ vector. DMNs reduce the number of parameters compared with DNNs, as the weight matrix $W^{(l)}$ of each layer in the DMN is $1/k$ of the size of its equivalent DNN weight matrix. An illustration of a DMN with 2 hidden layers and a group size of k=3 is shown in Figure 1.

The trained DMN-LBE consists of an input layer, two maxout hidden layers, and a softmax output layer. The two maxout hidden layers have 1000 and 500 randomly initialized nodes with a group size k of 100 and 5, respectively. The final softmax layer has two classes corresponding to LBE or non-LBE.

The DMN was trained using the back propagation (BP) algorithm and stochastic gradient descent (SGD) algorithm with dropout. Dropout prevents overfitting by randomly omitting a fraction of nodes for each training; a dropout rate of 0.5 was used. Hyperparameters were determined on the validation set. DMN-LBE uses a learning rate of 0.001, a momentum rate that increases linearly from 0.5 to 0.99 for the first 10 epochs and then remains at 0.99, a weight decay of 0.00005 and a minibatch size of 100. The validation set was evaluated at each epoch of the 1000 total training epochs to select the best model using the optimized training epoch that maximizes the classification accuracy of the validation set.

We used Pylearn2[6] to implement the model and exploited GPU to accelerate computation using NVIDIA Titan Black card.

The sensitivity (Sn), specificity (Sp), accuracy (Acc) and Matthew's correlation coefficient (MCC) of the method's performance were calculated using the following equations:

$$Sn = \frac{TP}{TP + FN} \times 100\% \quad (5)$$

$$Sp = \frac{TN}{TN + FP} \times 100\% \quad (6)$$

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (7)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (8)$$

where TP, TN, FP, and FN represent the numbers of true positive, true negative, false positive, and false negative cases, respectively.

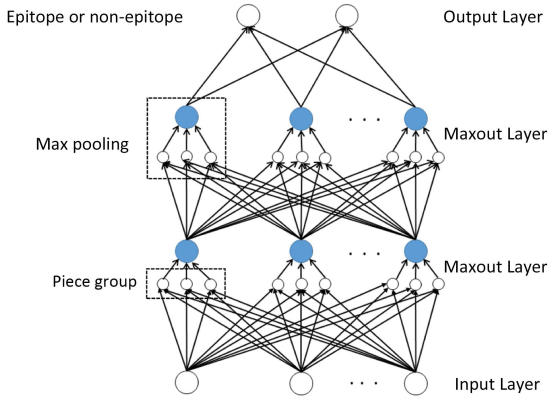

Figure 1. A deep maxout network with 2 hidden layers and a group size of k=3. The hidden nodes in blue perform the max operation.

The model was trained and tested on the LFNR dataset, which consists of 7824 experimentally verified LBEs and 7853 experimentally verified non-LBEs (20-mer). Based on the dipeptide composition features, the trained DMN-LBE model had a sensitivity of 68.72, a specificity of 67.94, an accuracy of 68.33 and an AUC of 0.743 using 10-fold cross-validation.

For comparison, the published models of ABCpred[7], AAP[5,8] and BCPred[5] servers were applied to the same LFNR dataset (Implementation of AAP method can be found at: http://ailab.ist.psu.edu/bcpreds/predict.html). The obtained accuracies were 52.33%, 60.20% and 54.29%, and AUCs were 0.533, 0.535 and 0.561, respectively. The performances of ABCpred, AAP and BCPred were significantly inferior compared to the current DMN-LBE model (accuracy=68.33%, AUC=0.743). The poor performance of these published models is due to the use of small datasets. We compare the DMN-LBE model with two recently developed large dataset models: the EPMLR[9] and LBtope models[10]. Using the same LFNR dataset, the EPMLR model achieved an accuracy of 58.45% and an AUC of 0.616. The LBtope model achieved accuracy of 64.86% and AUC of 0.688. The DMN-LBE model out-performed both the EPMLR model and LBtope model, by improving 9.83% accuracy and 12.7% AUC relative compared to the EPMLR model and 3.47% accuracy and 5.5% AUC compared to the LBtope model.

A detailed comparison of the DMN-LBE model with other reported models are presented in Table 1. The ROC plots for performances of ABCpred, AAP, BCPred, EPMLR, Lbtope, and DMN-LBE are shown in Figure 2.

Table 1. Performance of Different Methods on the LFNR Dataset

Method	Sensitivity	Specificity	Accuracy	MCC	AUC (95% CI)
ABCpred	57.40	47.28	52.33	0.05	0.533 (0.524-0.542)
AAP	65.94	54.48	60.20	0.21	0.535 (0.527-0.544)
BCPred	67.24	41.39	54.29	0.09	0.561 (0.552-0.570)
EPMLR	60.76	56.14	58.45	0.17	0.616 (0.608-0.625)
LBtope	65.88	63.97	64.86	0.30	0.688 (0.680-0.696)
DMN-LBE	68.72	67.94	68.33	0.37	0.743 (0.739-0.755)

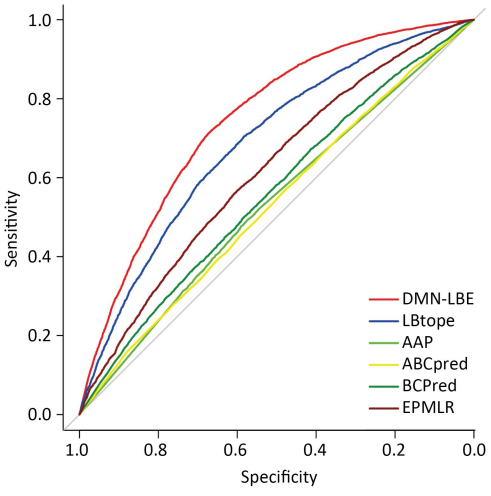

Figure 2. ROC curves for ABCpred, AAP, BCPred, EPMLR, Lbtope, and DMN-LBE.

In conclusion, we reported an improved LBE prediction model named DMN-LBE. This model is based on a deep architecture of DMN with dropout training technique. When applied to a large, non-redundant and experimentally verified dataset, the model achieved an AUC of 0.743, the best performing model in the field. We implemented our classification model as a free, user-friendly web server that is available at http://www.bioinfo.tsinghua.edu.cn/epitope/DMNLBE/ .

#Correspondence should be addressed to PAN Xian Ming. Tel: 86-10-62792827; E-mail: pan-xm@mail.tsinghua.edu.cn

Biographcal note of first author: LIAN Yao, female, born in 1988, PhD, majoring in bioinformatics.

Received: April 22, 2015;
Accepted: June 24, 2015

REFERENCES

1. Barlow DJ, Edwards MS, Thornton JM. Continuous and discontinuous protein antigenic determinants. Nature, 1986; 322, 747-8.
2. Gao J, Kurgan L. Computational prediction of B cell epitopes from antigen sequences. Methods Mol Biol, 2014; 1184, 197-215.
3. Goodfellow IJ, Warde-Farley D, Mirza M, et al. Maxout Networks. In International conference on machine learning, 2013.
4. Vita R, Zarebski L, Greenbaum JA, et al. The immune epitope database 2.0. Nucleic Acids Res, 2010; 38, D854-62.
5. El-Manzalawy Y, Dobbs D, Honavar V. Predicting linear B-cell epitopes using string kernels. J Mol Recognit, 2008; 21, 243-55.
6. Goodfellow IJ, Warde-Farley D, Lamblin P, et al. Pylearn2: a machine learning research library. arXiv preprint, 2013; arXiv:1308.4214.
7. Saha S, Raghava GPS. Prediction of continuous B-cell epitopes in an antigen using recurrent neural network. Proteins, 2006; 65, 40-8.
8. Chen J, Liu H, Yang J, et al. Prediction of linear B-cell epitopes using amino acid pair antigenicity scale. Amino Acids, 2007; 33, 423-8.
9. Lian Y, Ge M, Pan X. EPMLR: Sequence-based linear B-cell epitope prediction method using multiple linear regression. BMC Bioinformatics, 2014; 15, 414.
10. Singh H, Ansari HR, Raghava GP. Improved method for linear B-cell epitope prediction using antigen's primary sequence. PLoS One, 2013; 8, e62216.