Author’s response to reviews

Title: Deep learning methods improve linear B-cell epitope prediction

Authors:

Tao Liu (liutao_working@126.com)
Kaiwen Shi (15501107592@163.com)
Wuju Li (wujuchina@126.com)

Version: 1 Date: 28 Feb 2020

Author’s response to reviews:

Reviewer reports:

Reviewer #1: This paper describes the application of a deep learning r package for the purpose of epitope prediction. The manuscript is generally well written, although as I will describe below in my detailed comments, there is a need for further details in some sections. This is not my primary area of expertise, so I learned a lot from reading this paper. However, that is why I felt some more details are important to add. Perhaps some of the descriptions are more typical for an expert in this topic. But I would prefer to see the paper include some more details to make it accessible to a broader audience. One more primary issue that I have with the paper is that the results in Figure 3 show that this deep learning method outperforms existing methods in a huge way, but then in the additional dataset used in Figure 4, the results are much more limited in difference. Perhaps this is only due to the size of the datasets. But if definitely left me wondering if there was something else. Overfitting in the first set of experiments? Some bias in the datasets that would lead to this difference? It would be helpful if the authors could address this in the discussion.

Here are more detailed comments:

The authors describe a number of different machine learning methods in the introduction that have been applied to this problem. Have they been applied on the same datasets? For a non-expert in this field, it is difficult to assess how the methods compare to one another. It would be helpful if you could show a table that compares the performance of the existing methods so that it is a bit easier to see what the state of the art is currently that you are trying to achieve better performance.

Response: We added a table to compare the performance of various methods when applied on the same datasets.
In the methods, the authors indicate that they used the IEDB dataset to develop training and testing datasets. What is the outcome variable that is being evaluated in the training? Is it peptide length? Is it linear epitope or non-epitope? This is not clear from the description provided. I recognize that this is not my subject area, but many readers may use this manuscript to learn more about this topic. So this level of detail would be helpful.

Response: We call B-cell linear epitopes as positive samples, and the other peptides as negative samples. Therefore, the outcome variable that is being evaluated in the model training is B-cell linear epitope or non-epitope. An IEDBx dataset containing peptides of length x was used to build model predicting epitopes of length x. A series of IEDBx (x=11,12,…,50) datasets were used to build models predicting epitopes of length from 11 to 50, and thus determine the optimal length for epitope prediction. In the revised manuscript, we have made corresponding modifications.

If the independent dataset (Lbtope) is also available from IEDB, did you ensure that it was not included among all of the data used in training from IEDB20? Or is this what was removed in this step "To provide an objective performance comparison, the intersection set of 20 original Lbtope_Fixed and IEDB20 was deleted from the original Lbtope_Fixed". I think that is what you are saying here but want to confirm. This should be made more clear.

Response: The Lbtope_Fixed dataset consisted of samples of 20 AAs in length. We removed the intersection of Lbtope_Fixed and IEDB20 from Lbtope_Fixed. Therefore, the rest of the Lbtope_Fixed dataset had not been seen by our models and could be used fairly to compare our method with other methods. In the revised manuscript, we have made corresponding modifications.

Similarly, the ABCpred16 also had any overlap with IEDB20 removed, correct?

Response: The ABCpred16 dataset consisted of samples of 16 AAs in length. We removed the intersection of ABCpred16 and IEDB16 from ABCpred16. Therefore, the rest of the ABCpred16 dataset had not been seen by our models and could be used fairly to compare our method with other methods. In the revised manuscript, we have made corresponding modifications.

The feature extraction section is not clear. How do you take a vector of 400 elements that are binary and get a sum of 1? Please explain this more clearly.

Response: There are 20 amino acids that make up peptides. Dipeptide means two amino acids joined by a single peptide bond. So, there are 400 dipeptides. For a peptide of length n, it can be divided into n-1 dipeptide. The fractions of all 400 dipeptides in a peptide form a vector of 400 elements named dipeptide composition, whose elements should add up to 1. In the revised manuscript, we have made corresponding modifications in the “Feature extraction” section.
The authors say "Deep learning methods" in the abstract, intro, and the methods. Deep learning methods refer to a number of different machine learning methods. According to the methods, the authors used "keras", which is a neural network package. Why do the authors keep calling it "deep learning methods"? Could you just call it a neural network? Or is it something more than a neural network?

Response: In the revised manuscript, we have changed it to “feedforward deep neural network” where needed.

How many samples in the training and testing data were linear epitopes (pos) and non-epitopes (neg)? From my understanding that is the output that you are trying to learn. If this is the case, is this balanced in number? Accuracy as a training metric is known to have issues if the number of samples is unbalanced in a binary output. As indicated in the first sentences of the Results, this is confusing. Also, it is not clear to me why this is in the results? This is another detail that should be included in the methods. Once I looked at Figure 1B, this became more clear. But this should be in the methods, not results.

Response: In IEDBx (x=11,12,…,50) datasets, the numbers of positive samples approximately ranged from 21 to 24 thousands, and the numbers of negative samples approximately ranged from 196 to 211 thousands. We adopted ensemble learning to deal with the imbalance in the number of negative and positive samples. In the revised manuscript, we have given the corresponding description in the section of methods.

In addition to the downsampling approach, couldn't you have used a balanced accuracy metric? This way you could have compared the 20,000 epitopes to all of the non-epitopes at the same time?

Response: In each IEDBx dataset, the negative samples was about 10 times the positive samples. Therefore, we adopted ensemble learning to deal with the imbalance. By repeated subsampling the negative samples of an IEDBx dataset, we made multiple child datasets with a balance of negative and positive samples, from which multiple classifiers were generated by the feedforward deep neural network. These classifiers were combined as an ensemble to achieve much stronger generalization ability than that of a single classifier. Generally, to get a good ensemble, the classifiers should be as more accurate as possible, and as more diverse as possible. In this work, we used subsampling to introduce diversity. In addition, as the output of each classifier is binary, ensemble learning make it possible to compare performance of different methods using AUC values. In the revised manuscript, we have made corresponding explanation.

On page 13, the authors indicate that the "wrote a local R program" to submit the peptides to the web server and fetch predictions. Is this code being made available with the paper so that someone could reproduce this process?

Response: This code has been submitted as supplementary material.
In Figure 3, we see the results comparing DLBEpitope with the other methods and the difference is striking. This is not just a minor performance improvement. Can the authors provide more explanation for this in the text? Why is this method so dramatically different from the others? The difference is more subtle in the comparisons shown in Figure 4, which they explain why that might be. But Figure 3 differences are quite profound.

Response: Figure 3 shows how the different methods performed on the test dataset derived from the 2018 IEDB database. These test datasets were randomly selected and were not used for training our models. Our method performed far better than other methods, while overfitted models performed poorly in predicting test dataset they had not seen before. The performance improvement may be due to the large training dataset with experimental evidences and the combination of ensemble learning and deep neural network. The large training dataset provided more comprehensive recognition information. Multiple deep neural networks can learn more complex and accurate distinguishing information from multiple aspects. And ensemble learning can combine these networks to achieve much stronger generalization ability and make more accurate predictions.

Figure 4 shows how the different methods behave on two existing public datasets, the Lbtope_Fixed and ABCpred16 datasets. The Lbtope_Fixed dataset was derived from the 2012 IEDB database. It consisted of samples of 20 AAs in length. We have removed the intersection of Lbtope_Fixed and IEDB20 datasets to ensure that the rest of the Lbtope_Fixed has not been seen by our models. On the remaining Lbtope_Fixed dataset, our method performed far better than other methods. However, our method did not perform as good as on the IEDB20 test dataset. The Lbtope_Fixed was derived from peptides not less than 5 residues in the 2012 edition of IEDB, whereas the IEDB20 was derived from epitopes with 10 to 50 residues in the 2018 edition of IEDB. Removing the overlap with IEDB20, the remained Lbtope_Fixed data should be derived only from epitopes of length 5 to 9 or greater than 50 in the 2012 edition of IEDB. This may lead to some bias, and thus make our method perform worse than when tested with IEDB20 test dataset.

The ABCpred16 dataset consisted of samples of 16 AAs in length. We have also removed the intersection of ABCpred16 and IEDB16 to make sure that the rest of the ABCpred16 has not been seen by our models. On the remaining ABCpred16 dataset, our method did not significantly outperformed the other methods. The reason may be that the remained ABCpred16 dataset is too small (107 positive and 196 negative samples), which likely lead to some bias.

In the revised manuscript, we have made corresponding explanation in the discussion section.

Reviewer #2: This is a well-written paper that applies deep learning to the prediction of B-cell epitopes. The following are some suggestions for improving the paper.

1) How were the hidden layers of the deep learning neural network (DLNN) selected? Please provide a rationale. Were other DLNN architectures evaluated? If so, please provide the details in the paper. If not, why not?
Response: To prevent overfitting, it’s usually best to start with relatively few layers and parameters, then begin increasing the size of the layers or adding new layers. Accordingly, we started with one hidden layer and end up with four hidden layers with evaluation accuracy of about 0.8. In the revised manuscript, we have provided the details in the methods section.

2) Please provide details in the methods section how the DLNN models were evaluated. What were the criteria? How did you control for overfitting? Did you do a parameter sweep? Did you do hyperparameter tuning?

Response: The DLNN models were evaluated on a hold-out test set. We used the “tfruns” package to tune the hyperparameters for an evaluation accuracy as high as possible. And we used dropout to control for overfitting. In the revised manuscript, we have provided the corresponding details in the methods section.

3) An ensemble approach is mentioned in the results section but there are no methods described for this in the methods section. Please provide the details of the ensemble approach in the methods section.

Response: In the revised manuscript, we have provided the details of the ensemble learning approach in the methods section.

4) The split of the data into training and testing sets is mentioned in the results but only briefly in the methods. Please provide full details in the methods on how this was done and the rationale for the sample sizes picked.

Response: In IEDBx (x=11,12,…,50) datasets, the numbers of positive samples approximately ranged from 21 to 24 thousands, and the numbers of negative samples approximately ranged from 196 to 211 thousands. We adopted ensemble learning to deal with the imbalance in the number of negative and positive samples. For each IEDBx dataset, we randomly selected 20000 positive samples into the training set. The rest of the positive samples were left into the test set. Then the same amount of negative samples were randomly selected and left into the test set. Subsample of 20000 samples were randomly selected from the rest of the negative samples to enter the training set. This can make up a child dataset with positive and negative samples balanced. Then we obtained 11 child datasets from each IEDBx dataset and trained ensemble model accordingly. Repeated experiments showed that the performance was stable with the sample sizes picked. In the revised manuscript, we have provided the details of the dataset splitting in the methods section.

5) A stability analysis is mentioned in the results but not the methods. Please explain in the methods.
Response: In the revised manuscript, we have made the corresponding description of the stability analysis in the methods section.