Bioinformatics, 36(4), 2020, 1279–1280
doi: 10.1093/bioinformatics/btz708
Advance Access Publication Date: 14 September 2019
Applications Note

Sequence analysis

DeepMSPeptide: peptide detectability prediction using deep learning

Guillermo Serrano[1,†], Elizabeth Guruceaga[1,2,†] and Victor Segura [1,2,*]

[1]Bioinformatics Platform, Center for Applied Medical Research, University of Navarra, Pamplona 31008 and [2]IdiSNA, Navarra Institute for Health Research, Pamplona 31008, Spain

*To whom correspondence should be addressed.

[†]The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.

Associate Editor: John Hancock

Received on June 6, 2019; revised on September 2, 2019; editorial decision on September 10, 2019; accepted on September 11, 2019

Abstract

Summary: The protein detection and quantification using high-throughput proteomic technologies is still challenging due to the stochastic nature of the peptide selection in the mass spectrometer, the difficulties in the statistical analysis of the results and the presence of degenerated peptides. However, considering in the analysis only those peptides that could be detected by mass spectrometry, also called proteotypic peptides, increases the accuracy of the results. Several approaches have been applied to predict peptide detectability based on the physicochemical properties of the peptides. In this manuscript, we present DeepMSPeptide, a bioinformatic tool that uses a deep learning method to predict proteotypic peptides exclusively based on the peptide amino acid sequences.

Availability and implementation: DeepMSPeptide is available at https://github.com/vsegurar/DeepMSPeptide.

Contact: vsegura@unav.es

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Proteomics is one of the most important technologies in the field of the high throughput experiments (Nilsson *et al.*, 2010). In fact, protein detection and quantification are key to increase our understanding about the cell biology and the human diseases. In this context, the advances in mass spectrometry (MS) have been essential to provide the high-resolution instruments needed to obtain robust results at proteome level. Peptide detectability is also an important issue in Proteomics for protein identification, protein quantification and differentially expression studies (Li *et al.*, 2010). However, in the majority of the current bioinformatic pipelines for proteomic experiment analysis this feature is not considered. This peptide feature is difficult to quantify due to the great number of variables that contribute to the detection of a certain peptide or amino acid sequence by a mass spectrometer (Jamuczak *et al.*, 2016). Several methods have been previously reported to evaluate the detectability of a peptide using standard samples (Tang *et al.*, 2006) or using sets of identified peptides in different biological samples (Li *et al.*, 2010). In the last case, more complex models and best training approaches were applied due to the high number of examples available.

Our first approach to evaluate the peptide detectability was based on a classifier (Guruceaga *et al.*, 2017). We defined two peptide groups or classes: the peptides detected by MS and the non-detected peptides. The input features were the physicochemical properties of the peptides stored in AAIndex resource (Kawashima *et al.*, 2008) and other characteristics that can be calculated from their amino acid sequence. However, we considered that a method based on deep learning (DL) could be a better approach. Feature engineering is one of the most important steps when machine learning algorithms are applied to study and classify massive amounts of data (Hinton and Salakhutdinov, 2006). DL includes a variety of innovative methods with very high performance to carry out the feature learning extraction (LeCun *et al.*, 2015). These methods have dramatically improved the state-of-the-art in many domains such as drug discovery and Genomics (Aliper *et al.*, 2016; Miotto *et al.*, 2018). Examples of structures that implement these algorithms are the Restricted Boltzman Machines, the Deep Neural Networks (DNNs) and the Convolutional Neural Networks (CNNs).

In this manuscript, we present DeepMSPeptide, a bioinformatic tool written in Python to predict peptide detectability considering exclusively the peptide amino acid sequence. We have compared the obtained performance with previously used approaches (Guruceaga *et al.* 2017; Mallick *et al.*, 2007; Zimmer *et al.*, 2018) outperforming them.

2 Materials and methods

The input data used to train and test the different classifier approaches were obtained from the GPMDB database (Graig *et al.*, 2004), which includes thousands of peptides detected in hundreds of MS/MS experiments and the frequency of detection of each of them (Supplementary Material S1). The method to generate the input datasets is described elsewhere (Guruceaga *et al.*, 2017) and are

Table 1. Performance of the different classifier algorithms for peptide detectability prediction with the test dataset from the GPMDB database

Model	AUC	Accuracy	Specificity	Sensitivity	F-score
1D-2C-CNN	0.8570	0.7953	0.8880	0.7027	0.7744
1D-1C-CNN	0.8568	0.7917	0.9097	0.6737	0.7638
RF[a]	0.7549	0.6924	0.7746	0.6103	0.6649
SvmR[a]	0.7384	0.6813	0.7830	0.5797	0.6453
DNN[b]	0.7360	0.6692	0.6813	0.6572	0.6659
C5[a]	0.7312	0.6644	0.6513	0.6775	0.6687
Nnet[a]	0.7148	0.6723	0.8329	0.5118	0.6097
Rpart[a]	0.6893	0.6527	0.7467	0.5587	0.6167
Nb[a]	0.6456	0.5997	0.7280	0.4714	0.5408
Pls[a]	0.6350	0.6043	0.6396	0.5690	0. 5898
Glm[a]	0.6349	0.6036	0.6426	0.5646	0.5875
GlmStepAIC[a]	0.6349	0.6034	0.6424	0.5645	0.5874
Gaussian[c]	0.6342	0.5983	0.6121	0.5845	0.5927
Jrip[a]	0.6238	0.6240	0.6549	0.5930	0.6119

Note: Methods previously used in [a]Guruceaga *et al.* (2017); [b]Zimmer *et al.* (2018) and [c]Mallick *et al.* (2007) are compared with CNNs.

available in the GitHub repository. We implemented peptide detectability classifiers using different algorithms available in the R/Bioconductor package *caret* as previously described (Guruceaga *et al.*, 2017; Mallick *et al.*, 2007) to compare their performance with our DL approach. We also included in the comparison a simple DL method consisting in a DNN of four layers (Zimmer *et al.*, 2018).

The input vector of the DL classifier implemented using a CNN in TensorFlow was the integer conversion of the peptide sequence using a dictionary that assigned an integer to each amino acid. We standardized the input to feed the CNN performing zero-padding due to the different lengths of the peptides (from 4 to 81 amino acids). In this way, each vector that represented a peptide contained 81 elements. In addition, we used an *embedding layer* to process this input tensor as the first layer of the network. The second layer was a *dropout layer* that randomly set a fraction of the input units (0.2) to 0 at each epoch of the training, which prevented over-fitting. Next, we added 1 or 2 1D convolutional layers applying 64 filters with a sliding window of 1 and a width of 3. Finally, the model included two additional hidden layers. The output of the classifier was a probability (a single-unit layer with a sigmoid activation function) and the loss function selected for training was the binary cross entropy. Each model was trained for 200 epochs in batches of 100 samples. The accuracy was measured during the training with 1/4 of the samples as the validation set and a call-back function to stop the training when the loss function stabilized during five consecutive epochs.

3 Results

Table 1 summarizes the results obtained with the different peptide detectability prediction methods and the test dataset from the GPMDB database. The use of this database overcomes the limitations of other studies in which standard samples or a limited number of shotgun experiments were analyzed to generate the training dataset. In addition, we are predicting the intrinsic detectability of each peptide without the bias of the MS instrument used due to the representativeness of each type of mass spectrometer in GPMDB database. The best machine learning classifier was Random Forest (RF) using as input a vector of selected physicochemical characteristics of the peptides (57 features). The DNN is a simple DL structure in which more layers are added to a standard NN, so the complexity of the prediction function is not high enough to improve machine learning techniques. However, the DL approaches using CNNs outperformed significantly the results of RF and the other approaches using as input exclusively the amino acid sequence of each peptide (Supplementary Material S2). In particular, the 1D-2C-CNN provided the best results and using GPUs instead of CPUs its execution is 46% faster. As an additional benchmarking, we used the MS evidence for the human proteome provided by the HPP project (Supplementary Material S1). The results confirmed the improvement in performance of our DL algorithm.

The case of modified peptides was not considered but DeepMSPeptide is a versatile approach and including modified peptides as new integers in the dictionary in addition to a training dataset that contained this kind of amino acids would enable their detectability prediction.

4 Conclusions

In this manuscript, we present a DL approach to predict the peptide detectability in a MS-based experiment. For the first time only the amino acid sequence is required without the need of calculating the physicochemical properties of the peptides. Considering peptide detectability in high-throughput studies of protein identification and quantification can improve the results obtained. *DeepMSPeptide*, implemented in TensorFlow as a 1D-2C-CNN, is made available in a GitHub repository to predict the detectability of a set of input peptides.

Funding

This work was supported by PRBB-ISCIII (PT13/0001/0002), PRB3-ISCIII (PT17/0019/0013), Ministerio de Economía y Competitividad (DPI2015-68982-R) and MCIU/AEI/FEDER, UE (RTI2018-101481-B-100) co-financed by FEDER funds. The Bioinformatics Platform of CIMA is member of the ProteoRed-ISCIII platform. The authors gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan V GPU used for this research.

Conflict of Interest: none declared.

References

Aliper,A. *et al.* (2016) Deep learning applications for predicting pharmacological properties of drugs and drug repurposing using transcriptomic data. *Mol. Pharm.*, **13**, 1445–1454.

Graig,R. *et al.* (2004) Open source system for analyzing, validating, and storing protein identification data. *Mol. Syst. Biol.*, **3**, 1234–1242.

Guruceaga,E. *et al.* (2017) Enhanced missing proteins detection in NCI60 cell lines using an integrative search engine approach. *J. Proteome Res.*, **16**, 4374–4390.

Hinton,G.E. and Salakhutdinov,R.R. (2006) Reducing the dimensionality of data with neural networks. *Science*, **313**, 504–507.

Jamuczak,A.F. *et al.* (2016) Analysis of intrinsic peptide detectability via integrated label-free and SRM-based absolute quantitative proteomics. *J. Proteome Res.*, **15**, 2945–2959.

Kawashima,S. *et al.* (2008) AAindex: amino acid index database, progress report 2008. *Nucleic Acids Res.*, **36**, D202–D205.

LeCun,Y. *et al.* (2015) Deep learning. *Nature*, **521**, 436–444.

Li,Y.F. *et al.* (2010) The importance of peptide detectability for protein identification, quantification, and experiment design in MS/MS proteomics. *J. Proteome Res.*, **9**, 6288–6297.

Mallick,P. *et al.* (2007) Computational prediction of proteotypic peptides for quantitative proteomics. *Nat. Biotechnol.*, **25**, 125–131.

Miotto,R. *et al.* (2018) Deep learning for healthcare: review, opportunities and challenges. *Brief. Bioinform.*, **19**, 1236–1246.

Nilsson,T. *et al.* (2010) Mass spectrometry in high-throughput proteomics: ready for the big time. *Nat. Methods*, **7**, 681–685.

Tang,H. *et al.* (2006) A computational approach toward label-free protein quantification using predicted peptide detectability. *Bioinformatics*, **22**, e481–e488.

Zimmer,D. *et al.* (2018) Artificial intelligence understands peptide observability and assists with absolute protein quantification. *Front. Plant Sci.*, **9**, 1559.