Proceedings

Sequenced-based Discovery of Antibacterial Peptides Using Ensemble Gradient Boosting†

Ehdieh Khaledian [1,*], and Shira L. Broschat [1,2,3]

1 School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA P.O. Box 642752, USA; ehdieh.khaledian@wsu.edu (EK); shira@wsu.edu (SLB)
2 Department of Veterinary Microbiology and Pathology, Washington State University, Pullman, WA P.O. Box 647040, USA
3 Paul G. Allen School for Global Animal Health, Washington State University, Pullman, WA P.O. Box 647090, USA
* Correspondence: ehdieh.khaledian@wsu.edu;
† Presented at the 1st International Electronic Conference on Microbiology, 02–30 November 2020; Available online: https://sciforum.net/conference/ECM2020/

Published: 14 November 2020

Abstract: Antimicrobial resistance is driving pharmaceutical companies to investigate different therapeutic approaches. One approach that has garnered growing consideration in drug development is the use of antimicrobial peptides (AMPs). Antibacterial peptides (ABPs), which occur naturally as part of the immune response, can serve as powerful, broad-spectrum antibiotics. However, conventional laboratory procedures for screening and discovering ABPs are expensive and time-consuming. Identification of ABPs can be significantly improved using computational methods. In this paper, we introduce a machine learning method for the fast and accurate prediction of ABPs. We gathered more than 6000 peptides from publicly available datasets and extracted 1209 features (peptide characteristics) from these sequences. We selected the set of optimal features by applying correlation-based and random forest feature selection techniques. Finally, we designed an ensemble gradient boosting model (GBM) to predict putative ABPs. We evaluated our model using ROC curves, calculating the area under the curve (AUC) for several different models for comparison, including a recurrent neural network, a support vector machine, and iAMPpred. The AUC for the GBM was ∼0.98, more than 3% better than any of the other models.

Keywords: Antibacterial peptides; ensemble gradient boosting; drug discovery

1. Introduction

Antimicrobial resistance poses a severe health threat because it compromises treatment of a wide range of infections caused by bacteria, viruses, or fungi [1,2]. Moreover, bacterial infections can be resistant to multiple drugs, and resistance can be readily transferred to other bacteria over a short period. As a result, pharmaceutical companies have started investigating different therapeutic approaches. One approach that has generated much interest is the use of antimicrobial peptides (AMPs) [3]. Antibacterial peptides (ABPs) occur naturally as part of the immune response to combating microbial pathogens, and they can serve as powerful, broad-spectrum antibiotics. However, the conventional wet laboratory procedures used to screen peptides for the detection of ABPs are time-consuming and expensive [4]. In addition, ABPs evolve rapidly, and their small size, generally between 20 and 50 amino acids [5], makes them difficult to find via alignment-based detection approaches. Identification of ABPs can be significantly improved using computational methods. A promising computational approach is the use of a supervised

machine learning model that uses the physicochemical properties of ABPs. For example, in [6], the authors used pseudo amino acid composition and fuzzy k-nearest neighbors to develop a tool called iAMP-2L to predict AMPs. Later in [4], the authors developed an online prediction server called iAMPpred which increases the prediction performance by integrating compositional and physicochemical properties with structural features (features are peptide characteristics) and using a support vector machine (SVM) model. A semi-supervised machine learning approach was proposed by researchers in [7]. The authors used the DBSCAN algorithm for clustering and physicochemical properties as features. A recent work utilized an ensemble algorithm which integrates SVM, k-nearest neighbors, and four tree-based models [8]. In this work, extremely randomized tree algorithms were used for feature selection, and the resulting feature set was used to predict antihypertensive peptides (AHTPs). The features were used with an ensemble learner created by integrating all seven of the models. In another study, researchers proposed a tool called Deep-AmPEP30 to improve short AMP prediction using deep learning [9]. They developed the tool using an optimal feature set based on reduced amino acid composition together with a convolutional neural network.

In this paper, we introduce a machine learning method which predicts ABPs quickly and accurately. We gathered more than 6000 peptides from publicly available datasets and extracted 1209 features from these sequences. We selected the optimal features by applying correlation-based and random forest feature selection techniques. Finally, we designed an ensemble gradient boosting model (GBM) to predict putative ABPs. We evaluated our model using ROC curves, calculating the area under the curve (AUC) for several different models for comparison, including a recurrent neural network (RNN), a support vector machine (SVM), and iAMPpred. The AUC for the GBM approach was $\sim$0.98, more than 3% better than any of the other models.

2. Methods

In this section, we explain the overall pipeline for predicting ABPs using the gradient boosting model in detail.

2.1. Data collection and feature selection

We selected antibacterial peptides and non-antibacterial peptides from the data sets available through AmPEP [10] and DRAMP, the latter of which is a data repository of antimicrobial peptides [11]. Removal of duplicates resulted in a total of 6661 peptides of which 3423 were ABPs and 3238 were non-ABPs. Next, we used iFeature [12], a Python package, to extract features from the peptide sequences. iFeature provides a comprehensive toolkit for generating numerical feature representation schemes from peptide sequences. It is capable of calculating and extracting a broad range of important sequence encodings quickly. For the $>$6000 peptides in our data set, iFeature calculated and extracted 1209 features in just a few minutes. Table 1 summarizes these features. Amino acid composition (AAC) indicates the occurrence frequency of amino acids in sequences. Dipeptide composition (DPC) is the frequency of all possible permutations of two amino acids adjacent to each other. Dipeptide deviation from the expected mean (DDE) shows the dipeptide composition variance for a theoretical mean. Grouped amino acid composition (GAAC) categorizes the amino acids into five classes based on their physicochemical properties, e.g., hydropathy, charge, and molecular size. Grouped dipeptide Composition (GDPC) encoding is another variation of the DPC descriptor, and grouped tripeptide composition (GTPC) encoding is a tripeptide composition descriptor variation, which generates 125 descriptors. The composition, transition, and distribution (CTD) features represent the amino acid distribution patterns of a specific structural or physicochemical property in a protein or peptide sequence. The composition descriptor (CTDC) consists of values of the global compositions (percentages) of a protein's polar, neutral, and hydrophobic residues. Finally, the distribution

Table 1. List of peptide features.

Feature	Description	Dimension
AAC	Amino acid composition	20
DPC	Dipeptide composition	400
DDE	Dipeptide deviation from expected mean	400
GAAC	Grouped amino acid composition	5
GDPC	Grouped dipeptide composition	25
GTPC	Grouped tripeptide composition	125
CTDC	Composition	39
CTDD	Distribution	195
	Total Number of Features	1209

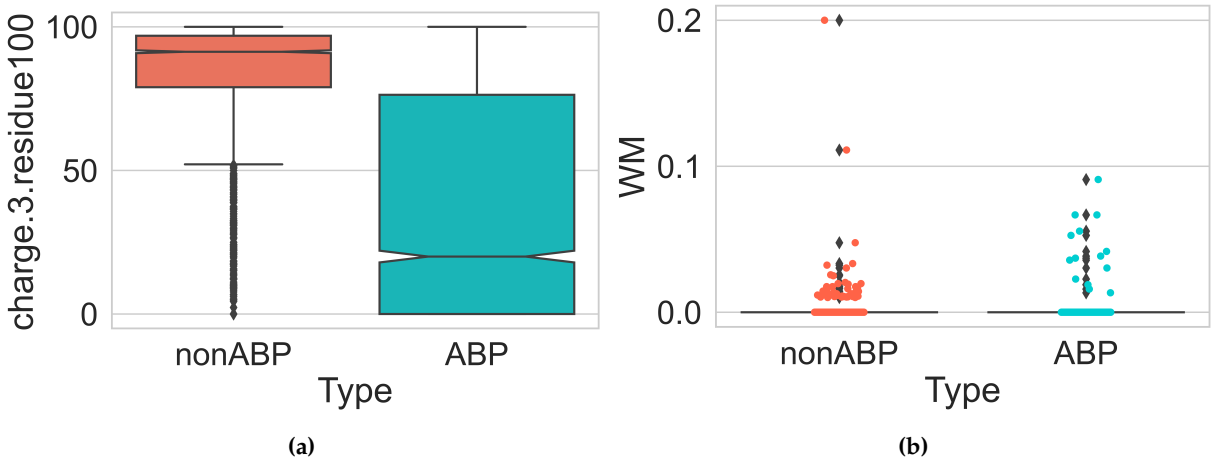

Figure 1. a) Two box plots showing values for an important feature for ABPs and non-ABPs. The median value (indentations in box) for non-ABPs is ∼90 while for ABPs it is is close to 25. Importantly, the boxes do not overlap, demonstrating that for >50% of the peptides, the feature values differ. **b)** For a feature of low importance, the distributions are similar.

descriptor (CTDD) comprises five values for each of the three groups (polar, neutral, and hydrophobic), namely the corresponding fraction of the entire sequence, where the first residue of a given group is located, and where 25, 50, 75 and 100% of occurrences are contained.

Not all features shown in Table 1 are informative, i.e., they don't discriminate between ABPs and non-ABPs, and some of the features provide redundant information. To determine an optimal feature set and to reduce the number of features, first we considered features with high correlation. We calculated Pearson's correlation coefficient for all features and removed one of each pair of features with a correlation >0.95. This reduced the number of features to 561. Next we used a random forest approach to determine the most relevant features. The random forest approach ranks the importance of features by determining how much the performance of the random forest decreases when a feature is removed. The more a feature lowers the impurity, the more influential the feature is. Figure 1 illustrates the distribution of two different peptide features. Figure 1a shows that the interquartile ranges are distinct for an important feature for ABPs and non-ABPs, while for features of low importance (Fig. 1b), the distribution is similar. In random forests, each feature's impurity decrease can be averaged across trees to determine the final importance of the feature [13].

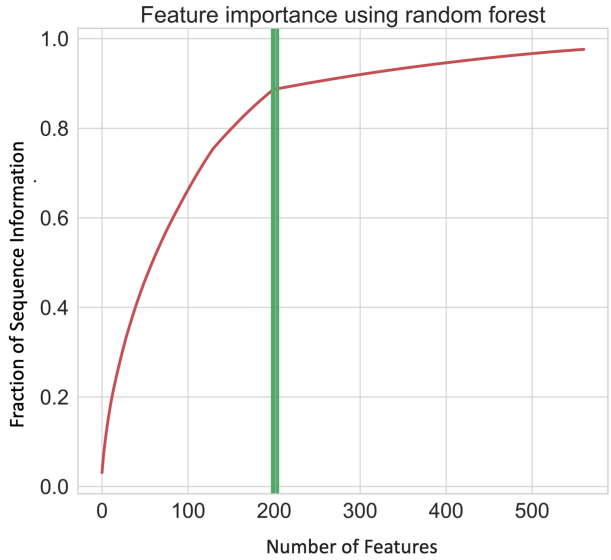

Figure 2. The fraction of peptide information available in 561 features is shown as a function of the number of features. The top 200 features were used in this work, and they represent ∼90% of the available information. The addition of 100 more features only adds 5%.

We retained the 200 most informative features which account for ∼ 90% of all the sequence information available in the 561 features. Figure 2 displays the cumulative contributions of features; an additional 100 features increases sequence information by only 5%.

2.2. Ensemble Gradient Boosting Model

Gradient boosting [14] is an ensemble boosting model that learns from previous mistakes, learning directly from the residual error. First, it trains a decision tree. Then the decision tree that is lately trained is used to predict. Next, this decision tree's pseudo residual error is computed and saved as the new y (labels). The lattar two steps are iterated until the number of trees that set to train is reached. Algorithm 1 demonstrates the GBM for a training set $\{(x_i, y_i)\}_{i=1}^{n}$, where x_i is a data point and y_i is its label, ABP or non-ABP for our work.

Algorithm 1: Gradient Boosting Algorithm

Data: Training data $\{(x_i, y_i)\}_{i=1}^{n}$, where x_i is a datapoint and y_i is the label of x_i
Input: Number of iterations M, logarithmic loss function, and decision tree base learner ($h(x)$)
Output: Final decision function F_M

Initialize the model $F_0(x) := argmin_\gamma \sum_{i=0}^{n} L(y_i, \gamma)$, $m := 0$
while $m \neq M$ **do**
  Calculate the pseudo residual error $r_{im} := -[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}]_{F(x)=F_{m-1}(x)}$ $for\ i = 1, ..., n$
  Fit a new base learner $h_m(x)$, using the new training set $\{(x_i, r_{im})\}_{i=1}^{n}$
  Find best gradient descent step size γ_m
  Update the model $F_m(x) := F_{m-1}(x) + \gamma_m h_m(x)$
  $m := m + 1$
return F_M

3. Results and Discussion

We divided our data sets into two subsets in order to perform cross-validation. The training subset, 75% of the data, was used to train the model. The remaining 25% of the data formed the validation, or test, subset which was used to analyze the performance of the model. Receiver operating characteristic (ROC) curves [15] were used to evaluate our model and to compare it to the performance of other models. An ROC curve gives the relationship between the false positive rate (FPR) and the true positive rate (TPR) at several threshold settings. The TPR (Eq. 1) is the ratio of true positive (TP) results and all possible positive data points, i.e., the sum of the true positive and false negative (FN) values. Similarly, the FPR (Eq. 2) is the ratio of false positive (FP) results and all possible negative data points, i.e., the sum of the false positive and true negative (TN) values. For our work, TPs are peptides correctly predicted as ABPs, TNs are peptides correctly predicted as non-ABPs, FPs are non-ABP peptides incorrectly predicted to be ABPs, and finally, FNs are ABPs incorrectly identified as non-ABPs.

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN} \tag{1}$$

$$FPR = \frac{FP}{N} = \frac{FP}{FP + TN} \tag{2}$$

Figure 3 gives the ROC curves for our GBM method, an SVM model, a recurrent neural network (RNN) using long short-term memory (LSTM) [16], and iAMPpred which is an online prediction server that increases the prediction performance by integrating compositional and physicochemical properties with structural features and uses a support vector machine (SVM) model. The area under the curve (AUC) for our GBM model is 98.5% which is approximately 3.5% more than those of the other models.

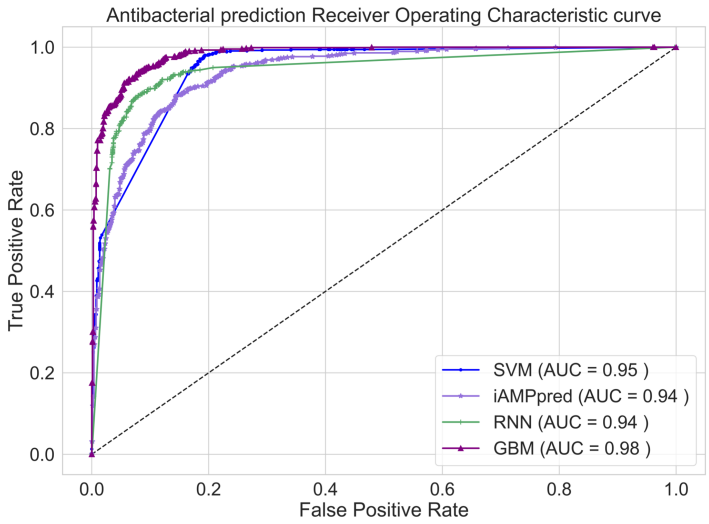

Figure 3. ROC curves for our GBM approach and three other approaches. Areas under the curve (AUCs) are shown for each model in the legend

4. Conclusions

Antibacterial peptides are a promising new approach for treating infections caused by bacteria, especially when antibacterial resistance is a problem. However, the laboratory work required to identify ABPs is both time-consuming and expensive. A machine learning model could assist with this work by

predicting ABPs that can then be verified experimentally. In this work, we have proposed such a model, an ensemble gradient boosting model, and we have shown that it gives more accurate results than other models.

Conflicts of Interest: The authors declare no conflict of interest

Abbreviations

AMP	Antimicrobial peptides
ABP	Antibacterial peptides
GBM	Gradient Boosting Machine

References

1. Alcock, B.P.; Raphenya, A.R.; Lau, T.T.; Tsang, K.K.; Bouchard, M.; Edalatmand, A.; Huynh, W.; Nguyen, A.L.V.; Cheng, A.A.; Liu, S.; others. CARD 2020: antibiotic resistome surveillance with the comprehensive antibiotic resistance database. *Nucleic acids research* **2020**, *48*, D517–D525.
2. Khaledian, E.; Brayton, K.A.; Broschat, S.L. A Systematic Approach to Bacterial Phylogeny Using Order Level Sampling and Identification of HGT Using Network Science. *Microorganisms* **2020**, *8*, 312.
3. Fu, H.; Cao, Z.; Li, M.; Wang, S. ACEP: improving antimicrobial peptides recognition through automatic feature fusion and amino acid embedding. *BMC genomics* **2020**, *21*, 1–14.
4. Meher, P.K.; Sahu, T.K.; Saini, V.; Rao, A.R. Predicting antimicrobial peptides with improved accuracy by incorporating the compositional, physico-chemical and structural features into Chou's general PseAAC. *Scientific reports* **2017**, *7*, 1–12.
5. Wu, Q.; Patočka, J.; Kuča, K. Insect antimicrobial peptides, a mini review. *Toxins* **2018**, *10*, 461.
6. Xiao, X.; Wang, P.; Lin, W.Z.; Jia, J.H.; Chou, K.C. iAMP-2L: a two-level multi-label classifier for identifying antimicrobial peptides and their functional types. *Analytical biochemistry* **2013**, *436*, 168–177.
7. Vishnepolsky, B.; Gabrielian, A.; Rosenthal, A.; Hurt, D.E.; Tartakovsky, M.; Managadze, G.; Grigolava, M.; Makhatadze, G.I.; Pirtskhalava, M. Predictive model of linear antimicrobial peptides active against gram-negative bacteria. *Journal of chemical information and modeling* **2018**, *58*, 1141–1151.
8. Manavalan, B.; Basith, S.; Shin, T.H.; Wei, L.; Lee, G. mAHTPred: a sequence-based meta-predictor for improving the prediction of anti-hypertensive peptides using effective feature representation. *Bioinformatics* **2019**, *35*, 2757–2765.
9. Yan, J.; Bhadra, P.; Li, A.; Sethiya, P.; Qin, L.; Tai, H.K.; Wong, K.H.; Siu, S.W. Deep-AmPEP30: Improve short antimicrobial peptides prediction with deep learning. *Molecular Therapy-Nucleic Acids* **2020**.
10. Bhadra, P.; Yan, J.; Li, J.; Fong, S.; Siu, S.W. AmPEP: Sequence-based prediction of antimicrobial peptides using distribution patterns of amino acid properties and random forest. *Scientific reports* **2018**, *8*, 1–10.
11. Kang, X.; Dong, F.; Shi, C.; Liu, S.; Sun, J.; Chen, J.; Li, H.; Xu, H.; Lao, X.; Zheng, H. DRAMP 2.0, an updated data repository of antimicrobial peptides. *Scientific data* **2019**, *6*, 1–10.
12. Chen, Z.; Zhao, P.; Li, F.; Leier, A.; Marquez-Lago, T.T.; Wang, Y.; Webb, G.I.; Smith, A.I.; Daly, R.J.; Chou, K.C.; others. iFeature: a python package and web server for features extraction and selection from protein and peptide sequences. *Bioinformatics* **2018**, *34*, 2499–2502.
13. Qi, Y. Random forest for bioinformatics. In *Ensemble machine learning*; Springer, 2012; pp. 307–323.
14. Friedman, J.H. Greedy function approximation: a gradient boosting machine. *Annals of statistics* **2001**, pp. 1189–1232.
15. McClish, D.K. Analyzing a portion of the ROC curve. *Medical Decision Making* **1989**, *9*, 190–195.
16. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural computation* **1997**, *9*, 1735–1780.