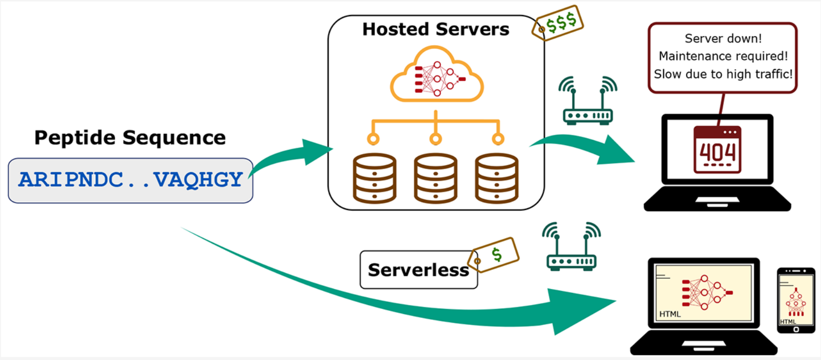
JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

Serverless Prediction of Peptide Properties with Recurrent Neural Networks

Mehrad Ansari and Andrew D. White*

Cite This: J. Chem. Inf. Model. 2023, 63, 2546−2553 Read Online

ACCESS | Metrics & More | Article Recommendations



ABSTRACT: We present three deep learning sequence-based prediction models for peptide properties including hemolysis, solubility, and resistance to nonspecific interactions that achieve comparable results to the state-of-the-art models. Our sequence-based solubility predictor, MahLooL, outperforms the current state-of-the-art methods for short peptides. These models are implemented as a static website without the use of a dedicated server or cloud computing. Web-based models like this allow for accessible and effective reproducibility. Most existing approaches rely on third-party servers that typically require upkeep and maintenance. Our predictive models do not require servers, require no installation of dependencies, and work across a range of devices. The specific architecture is bidirectional recurrent neural networks. This *serverless* approach is a demonstration of edge machine learning that removes the dependence on cloud providers. The code and models are accessible at https://github.com/ur-whitelab/peptide-dashboard.

1. INTRODUCTION

Deep learning models have been widely applied to extract information from big data in cheminformatics. Compared to machine learning algorithms, deep learning can perform feature extraction and learn patterns over various nonlinear layers of representations of the input data,[1] can explain the vanishing effects of gradients,[2] and can perform better with raw high-dimensional data.[3] There is a growing increase in the number of web-based implementations of deep learning frameworks that provide convenient public access and ease of use.[4−9] Notably, many web servers have been developed for sequence design tasks, like analysis of RNA, DNA, or proteins, for example, survival analysis based on mRNA data (GENT2,[10] PROGgeneV2,[11] SurvExpress,[12] MEXPRESS,[13] etc.), studying prognostic implications of noncoding RNA (PROGmiR,[14] SurvMicro,[15] OncoLnc,[16] TANRIC[17]), survival analysis based on protein (TCPAv3.0,[18] TRGAted[19]) and DNA (MethSurv,[20] cBioPortal[21]) data, and multiple areas of assessing cancer therapeutics.[22] These scientific web servers and web-based services allow for the availability of complex inference algorithms to a much broader user community and promote open science. This is especially important because of the disparities between lower and higher income nations, where there are disparities in the types of research activities that can be performed.[23] Cheminformatics-related research, the topic of this work, mostly takes place at those nations privileged with resource-rich institutions, where there are adequate funding resources. Yet, web-based implementations can broaden access to these methods.

Beyond disparities among institutions, web-based implementations are also a mechanism for reproducibility in science. In peptides specifically, Melo et al.[24] argue that deep learning sequence design should be accomplished by free public access to the (1) source code, (2) training and testing data, and (3) published findings. However, this is not often true; Littmann et

Received: October 20, 2022
Published: April 3, 2023

https://doi.org/10.1021/acs.jcim.2c01317

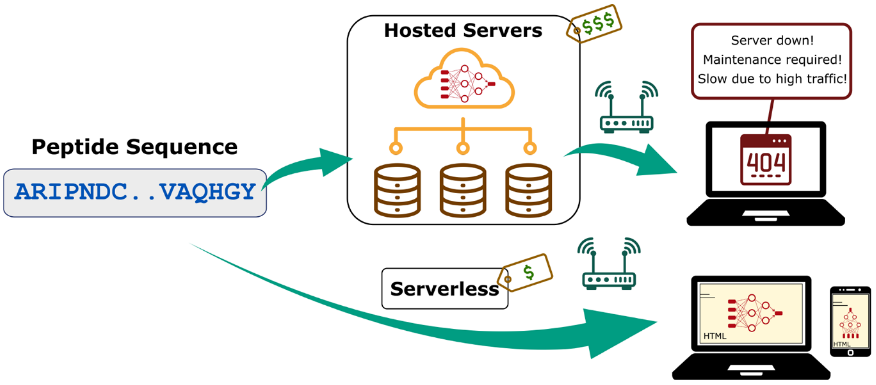


Figure 1. Conventional web-based cheminformatics frameworks vs the proposed serverless approach.

al.[25] found in an analysis of ML research articles in biomedicine and life sciences published between 2011 and 2016 that only 50% released software, while 64% released data. Web-based servers do not fit the exact definition of open science (due to lack of source code access), but they do accomplish the goal of enabling others with broader expertise to build on previous advances and are often more accessible and convenient than access to model and source code alone.

Thus, there is a compelling argument to continue web-based tools. There are, however, two major drawbacks: source-code can be inaccessible as discussed above and the reliance on third-party or self-hosted servers. Deep learning inference often requires GPUs, and this requires a specialized hosting service or a complex self-hosted setup. This creates difficult ongoing expenses, and many tools are thus only available for a limited time after publication. Additionally, there can be low incentives to increase capacity. Popular tools, like RoseTTAFold,[26] can have days-long queues. The expense and deployment problems also can create disparities in impact of research between resource-rich and low-resource institutions, because not all researchers can afford to create web-based implementations.

To address the challenges above, we demonstrate a *serverless* deep learning web-based server, https://peptide.bio, that predicts peptide properties using recurrent neural networks (RNN) via users' local devices. These trained models are implemented in JavaScript and are loaded to a user's web browser. Users make predictions by running these trained models on a web browser on their local machines, or even cell phones, without having to install any modules. They can be run locally as well, if desired[a]. The *serverless* computing describes a programming model and architecture, where small code snippets are executed in the cloud without any control over the resources on which the code runs.[27] This is by no means an indication that there are no servers. Simply, it means that the developer leaves most operational concerns such as resource provisioning and scalability to the cloud provider (the end-user in our case). Figure 1 provides a visual demonstration on how our approach contrasts with the existing chem-informatics frameworks. John et al.[28] proposed the idea of serverless computing for efficient model utilization on cloud resources without specific constraints on the cloud provider. However, in this work, we seek to fully remove the need for a cloud provider and bypass this conventional dependency. While this is indeed impractical for the resource-intensive training step of deep learning models, the trained models are typically cheap to evaluate; thus, inference is robustly feasible on even limited computing resources (i.e., commodity phones, laptops). This removes hosting costs and the conventional dependence on cloud providers or self-hosting of resource-rich academic institutions. Although we make some compromises here on model size and complexity, we expect the continued improvement of hardware (i.e., Moore's law[29]) to increase the type of models possible in JavaScript each year. This serverless approach should accelerate reproducible ML science, while also lowering the gap between resource-rich universities and the rest, as well as enabling a better dissemination of research from a broader community of chemists.

This paper is organized as follows: We start by providing a brief overview of some comparable predictive sequence-based models for the classification tasks in this work (hemolysis, solubility, and nonfouling) in Section 1.1. In Section 2, we describe the data sets, architecture of our deep learning models, and the choices for the hyperparameters, as well as a high level overview of the methods used in the previous comparable sequence-based models in the literature. This is followed by evaluating the model in a comparative setting with the state-of-the-art models in Section 3. Finally, we conclude the paper in Section 4, with a discussion of the implications of our findings.

1.1. Previous Work.

Quantitative structure−activity relationship (QSAR) modeling is a well-established field of research that aims at mapping sequence and structural properties of chemical compounds to their biological activities.[30] QSAR models have been successfully applied to angiotensin-converting enzyme(ACE)-inhibitory peptides,[31−33] antimicrobial peptides,[34−37] and antioxidant peptides.[38−40] For solubility predictions, DSResSol (1)[41] improved prediction accuracy (ACC) and AUROC to 75.1% and 0.84, respectively, by identifying long-range interaction information between amino acid k-mers with dilated convolutional neural networks and outperformed all existing models such as DeepSol,[42] PaRSnIP,[43] SoluProt,[44] Protein−Sol,[45] and PROSO II.[46] HAPPENN[47] forms the state-of-the-art (SOTA) model for hemolytic activity prediction with ACC of 85.7% and has better performance compared with HemoPI[48] and HemoPred.[49] Hasan et al.[50] developed a two-layer prediction framework, called HLPpred-Fuse, that can distinguish between hemolytic and nonhemolytic peptides, as well as their low and high activity, with an AUROC of 0.91 (averaged over two reported independent data sets). However, short peptides (<6 amino acid residues) are excluded from their data sets due to the difficulty in capturing meaningful sequence information from shorter peptides.

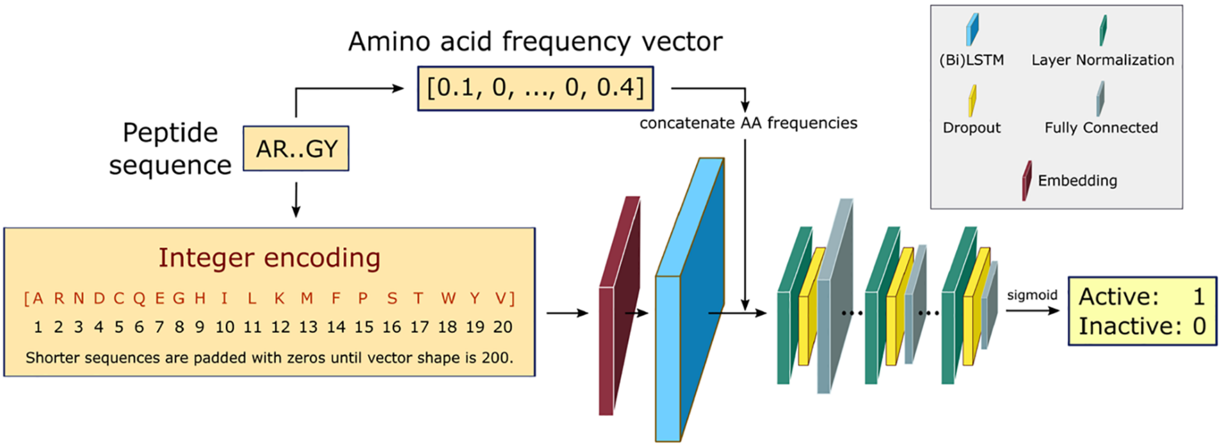


Figure 2. RNN architecture. Fixed-length integer-encoded sequences are first fed to a trainable embedding layer, yielding a semantically more compact representation of the input essential amino acids. The bidirectional LSTMS and direct inputs of amino acid frequencies prior to the fully connected layer improve the learning of bidirectional dependency between distant residues within a sequence. The fully connected layers are downsized in three consecutive steps with layer normalization and dropout regularization. The final layer uses a sigmoid activation to output a scalar that shows the probability of being active for the desired training task.

2. MATERIALS AND METHODS

2.1. Data Sets. *2.1.1. Hemolysis.* Hemolysis is defined as the disruption of erythrocyte membranes that decrease the life span of red blood cells and causes the release of hemoglobin. Identifying nonhemolytic antimicrobial is critical to their applications as nontoxic and safe measurements against bacterial infections. However, distinguishing between hemolytic and nonhemolytic peptides is complicated, as they primarily exert their activity at the charged surface of the bacterial plasma membrane. Timmons and Hewage[47] differentiate between the two whether they are active at the zwitterionic eukaryotic membrane, as well as the anionic prokaryotic membrane. In this work, the model for hemolytic prediction is trained using data from the Database of Antimicrobial Activity and Structure of Peptides (DBAASP v3[51]). The activity is defined by extrapolating a measurement assuming dose response curves to the point at which 50% of red blood cells (RBC) are lysed. If the activity is below 100 μg/mL, it is considered hemolytic. Each measurement is treated independently, so sequences can appear multiple times. The training data contains 9316 sequences (19.6% positives and 80.4% negatives) of only L- and canonical amino acids. Note that due to the inherent noise in the experimental data sets used, in some observations (∼40%), an identical sequence appears in both negative and positive class. As an example, sequence "RVKRVWPLVIRTVIAGYNLYRAIKKK", is found to be both hemolytic and nonhemolytic in two different lab experiments (i.e., two training examples).

2.1.2. Solubility. The training data contains 18,453 sequences (47.6% positives and 52.4% negatives) based on data from PROSO II.[46] Solubility was estimated by retrospective analysis of electronic laboratory notebooks. The notebooks were part of a large effort called the Protein Structure Initiative and consider sequences linearly through the following stages: Selected, Cloned, Expressed, Soluble, Purified, Crystallized, HSQC (heteronuclear single quantum coherence), Structure, and deposited in PDB.[52] The peptides were identified as soluble or insoluble as described in ref 46: "comparing the experimental status at two time points, September 2009 and May 2010; we were able to derive a set of insoluble proteins defined as those which were not soluble in September 2009 and still remained in that state 8 months later".

2.1.3. Nonfouling. Data for predicting resistance to nonspecific interactions (nonfouling) are obtained from ref 35. Positive data contains 3600 sequences. Negative examples are based on 13,585 sequences (20.9% positives and 79.1% negatives) coming from insoluble and hemolytic peptides, as well as the scrambled positives. The scrambled negatives are generated with lengths sampled from the same length range as their corresponding positive set, and residues were sampled from the frequency distribution of the soluble data set. Samples are weighted to account for the class imbalance caused by the data set size for negative examples. A nonfouling peptide (positive example) is defined using the mechanism proposed by White et al.[53] Briefly, White et al. showed that the exterior surfaces of proteins have a significantly different frequency of amino acids, and this increases in aggregation prone environments, like the cytoplasm. Synthesizing self-assembling peptides that follow this amino acid distribution and coating surfaces with the peptides creates nonfouling surfaces. This pattern was also found inside chaperone proteins, another area where resistance to nonspecific interactions is important.[54]

2.2. Model Architecture. To identify the position-invariant patterns in the peptide sequences, we build a recurrent neural network (RNN), using a sequential model from Keras framework[55] and the TensorFlow deep learning library back-end.[56] Specifically, the RNN employs bidirectional Long Short-term Memory (LSTM) networks to capture long-range sequence correlations. Compared to the conventional RNNs, LSTM networks with gate control units (input gate, forget gate, and output gate) can learn dependency information between distant residues within peptide sequences more effectively.[57−59] They can also partly overcome the problem of vanishing or exploding gradients in the back-propagation phase of training conventional RNNs.[60] We use a bidirectional LSTM (bi-LSTM) to enhance the capability of our model in learning bidirectional dependence between N-terminal and C-terminal amino acid residues. An overview of the RNN architecture is shown in Figure 2.

Peptide sequences are represented as integer-encoded vectors of shape 200, where the integer at each position in the vector corresponds to the index of the amino acid from the alphabet of the 20 essential amino acids: [A, R, N, D, C, Q, E, G, H, I, L, K, M, F, P, S, T, W, Y, V]. The maximum length of the peptide sequence is fixed at 200, and all sequences with higher lengths are excluded. For those sequences with shorter lengths, zeros are padded to the integer encoding representation to keep the shape fixed at 200 for all examples to allow input sequences with flexible lengths. Note that this is primarily applied to the training step for implementation considerations, and the trained model can make predictions on variable-length sequences as input. Every integer-encoded peptide sequence is first fed to an embedding layer. The embedding layer enables us to convert the indices of discrete symbols (i.e., essential amino acids) into a representation of a fixed-length vector of defined size. This is beneficial in the sense of creating a more compact representation of the input symbols, as well as yielding semantically similar symbols close to one another in the vector space. This embedding layer is trainable, and its weights can be updated during training along with the others layers of the RNN.

The output from the embedding layer either goes to a double stacked bi-LSTM layer or a single LSTM layer to identify patterns along a sequence that can be separated by large gaps. The former is used in predicting solubility and hemolysis, whereas the latter is for predicting a peptide's resistance to nonspecific interactions (nonfouling). The rationale behind this choice for the nonfouling model is that the bi-LSTM layer did not contribute to a better performance when compared with the LSTM layer (same ACC and AUROC of 82% and 0.93, respectively). The output from the LSTM layer is then concatenated with the relative frequency of each amino acid in the input sequences. This choice is partially based on our earlier work,[61] and helps with improving model performance. The concatenated output is then normalized and fed to a dropout layer with a rate of 10%, followed by a dense neural network with a ReLU activation function. This is repeated three times, and the final single-node dense layer uses a sigmoid activation function to force the final prediction as a value between 0 and 1. This scalar output shows the probability of the label being positive for the corresponding predicted peptide biological activity. We use this probability to evaluate the confidence of the model in making inferences on new sequences in our web-based implementation.

The hyperparameters are chosen based on a random search that resulted in the best model performance in terms of the Area Under the Receiver Operating Characteristic (AUROC) curve[62] and accuracy. The AUROC shows the model's ability to discriminate between positive and negative examples as the discrimination threshold is varied, and the accuracy is defined as the ratio of correct predictions to the total number of predictions made by the model. The embedding layer has the same input dimension of 21 (alphabet length added by one to account for the padded zeros) and output dimension of 32. The LSTM layer has 64 units, and the first, second, and third dense layers have 64, 16, and 1 units, respectively. We train with the Adam optimizer[63] of binary cross-entropy loss function, which is defined as

$$-\frac{1}{N}\sum_{i=1}^{N}[y_i\log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{1}$$

where y_i is the true value of the ith example, $\hat{y}_i$ the corresponding prediction, and N the size of the data set. The learning rate is adapted using a cosine decay schedule with an initial learning rate of 10^{-3}, decay steps of 50, and minimum of 10^{-6}. The data split for training, validation, and test is 81%, 9%, and 10%, respectively. To avoid overfitting, we add early stopping with patience of 5 that restores model weights from the epoch with the maximum AUROC on the validation set during training.

Previous models for peptide prediction tasks use a variety of deep learning and classical machine learning methods. The prediction server PROSO II employs a two-layered structure, where the output of a primary Parzen[64] window model for sequence similarity and a logistic regression classifier of amino acid k-mer composition are fed to a second-level logistic regression classifier. HAPPENN uses normalized features selected by SVM and ensemble of Random Forests, which are fed to a deep neural network with batch normalization and dropout regularization to prevent overfitting. DSResSol (1) takes advantage of the integration of Squeeze-and-Excitation (SE)[65] residual networks[66] with dilated convolutional neural networks.[67] Specifically, the model includes five architectural units, including a single embedding layer, nine parallel initial CNNs with different filter sizes, nine parallel SE-ResNet blocks, three parallel CNNs, and fully connected layers.

3. RESULTS

Table 1 shows the classification performance for all three tasks, along with a comparison between our RNN model and the state-of-the-art methods (see Section 1.1 for a brief overview). All models achieve the same result range as the state-of-the-art methods. We compare the feature extraction capability of our RNN with other unconditional protein language models that provide pretrained sequence representations that transfer well to supervised tasks. Specifically, we train two machine learning models on the hemolytic data set, using a UniRep[68,69] representation of the peptide sequences, followed by a logistic regression and a Random Forests[70] classifier. Our RNN architecture slightly outperforms both models in terms of AUROC. The one-hot representation of peptides followed by RNN results in the best hemolysis model in terms of AUROC in ref 71. The choice of one-hots requires training features specific to each position though, so we do not expect the model to generalize. In contrast, our model is length agnostic and will have a relatively smaller generalization error for

Table 1. Performance Comparison on the Testing Data Set[a]

Method	Task	ACC (%)	AUROC
Embedding + Bi-LSTM*	Hemolysis	84.0	0.84
UniRep + Logistic Regression	Hemolysis	82.0	0.81
UniRep + Random Forests	Hemolysis	84.0	0.78
HAPPENN[47]	Hemolysis	85.7	–
HLPpred-Fuse[50]	Hemolysis	–	0.91[b]
one-hots + RNN[71]	Hemolysis	76.0	0.87
Embedding + LSTM*	Nonfouling	82.0	0.93
Embedding + Bi-LSTM (MahLooL*)	Solubility	70.0	0.76
PROSO II[46]	Solubility	71.0[c]	0.78[c]
DSResSol (1)[41]	Solubility	75.1[d]	**0.84**[d]

[a]Best performing method for each task is in bold. Our approach is highlighted with an asterisk. [b]AUROC is averaged over two reported data sets excluding <6 amino acid residue peptides. [c]Based on sequence clustering at 90% identity. [d]Based on test set.[72]

sequences with lengths it has not observed before. Moreover, this removes the need for having training data at each position for each amino acid.

Our predictive model for the solubility task, MahLooL ("محلول"), has accuracy of 70.0%, and this is mostly attributed to the difficulty associated with solubility predictions in cheminformatics. Note that the solubility data set used contains a large distribution of sequence lengths (18−198). DSResSol (1) outperforms all existing solubility models on the testing set from ref 72. Readers are encouraged to refer to ref 41 for the comparison of DSResSol (1) with all the state-of-the-art sequence-based solubility predictors. For the sake of a better comparison with our approach, we evaluate DSResSol (1)'s performance on the same testing set used for MahLooL. We explore the changes in model performance by training on all the training set, but filtering the testing data based on the sequence lengths, as illustrated in Table 2. MahLooL has comparable performance with respect to DSResSol (1) on the entire testing set. For short length (18−50) peptides, surprisingly it outperforms DSResSol (1), with an AUROC of 0.95 and accuracy of 91.3%. With longer length peptide sequences, the property inference task becomes more difficult by only using the amino acid sequence information, as other experimental settings and conditions become important, adding more epistemic uncertainty to the predictions.

Table 2. Performance Comparison between MahLooL and DSResSol (1) by Training MahLooL on All the Training Set, While Filtering the Testing Set Based on Sequence Lengths[a]

		MahLooL		DSResSol (1)	
Length filter	# Test seqs	ACC (%)	AUROC	ACC (%)	AUROC
none	1845	70.0	0.76	71.0	0.78
(18−50)	23	**91.3**	**0.95**	87.0	0.92
(50−100)	272	76.5	0.80	75.7	0.82
(100−150)	715	70.2	0.76	70.6	0.76
(150−198)	806	70.5	0.74	71.6	0.77
(18−100)	295	78.0	0.82	76.9	0.83

[a]Best performing method for each task is in bold.

To allow for transparency between users and developers, details of the models' performance, training procedures, intended use, and ethical considerations have been incorporated as model cards[73] on https://peptide.bio/. Model cards present information about how the model is trained, its intended use, caveats about its use, and any ethical or practical concerns when using model predictions. A brief overview of the model cards are presented in Table 3.

To evaluate the contribution of different architectural components to the model's performance, we conducted a set of ablative experiments on the solubility model only. In each ablation trial, an architectural component is removed, and the corresponding test AUC and accuracy are reported via a 5-fold cross-validation on the solubility data set. We remove the effect of regularization techniques (see Materials and Methods in Section 2) in our ablation trials by disregarding the early stopping callback and fixing the number of training epochs to 50. The learning rate is also set to a fixed value of 10^{-3}. This is the reason for the lower performance of the "full model". The results from our ablation study are shown in Table 4, sorted by the highest AUROC. We point out that the AUROC of the

Table 3. Summary of Model Cards: Intended Use, Caveats, and Any Ethical or Practical Concerns with Three Developed Models[a]

	Hemolysis	Solubility	Nonfouling
Intended use	Peptides between 1 and 190 residues. L- and canonical amino acids.	Peptides or proteins expressed in *E. coli* that are less than 200 residues long. May provide solubility predictions more broadly applicable.	Short peptides between 2 and 20 residues.
Factors	Data set was from sequences thought to be antimicrobial or clinically relevant.	Solubility was defined in PROSO II[46] as sequence that was transfectable, expressible, secretable, separable, and soluble in *E. coli* system.	Data set was gathered based on the mechanism proposed in ref 53.
Ethical considerations	These predictions are not a substitute for laboratory experiments.	None noted.	None noted.
Caveats	Sequences tested were typically from biological sources.	These data are mostly long sequences and so may not be as applicable to solid-phase synthesized peptides. Model accuracy is low for long sequences.	These data are mostly short sequences. Mechanism is indirect. Negative examples have insoluble peptides overrepresented so that accuracy may be inflated if only comparing soluble peptides.

[a]For more details, refer to https://peptide.bio/.

Table 4. Ablation Trials to Evaluate the Contribution of Model's Architectural Components in Classification Performance on Solubility Data Set via 5-Fold Cross-Validation[a]

Change in architecture	ACC (%)	AUROC
None* (full model)	63.1 ± 4.1	0.683 ± 0.046
Removing AA count frequencies	62.2 ± 2.9	0.667 ± 0.027
Removing dropout	61.7 ± 2.4	0.667 ± 0.030
Removing layer normalization	62.2 ± 3.4	0.661 ± 0.030
Removing first and second dense layers	59.8 ± 2.0	0.637 ± 0.021
Removing bidirectionality of LSTM layer	56.1 ± 1.1	0.580 ± 0.015

[a] For comparison, the performance of the model with full architecture (as shown in Figure 2) is highlighted with an asterisk.

solubility model has a significant drop from 0.76 to 0.68 after removing the regularization callbacks and fixing the learning rate in our cross-validation analysis. Removing amino acid count frequencies, dropout, and layer normalization layer reduced AUROC by about 2%. The removal of the first and second dense layers decreased performance by about 5%. Finally, our ablation analysis shows that Bi-LSTM is the most contributing component of the architecture, as its removal decreased AUROC by about 10%. Indeed, the bidirectionality feature of Bi-LSTM layers boosts the performance by enabling additional learning of the dependence between N-terminal and C-terminal amino acid residues.

4. DISCUSSION

We present three sequence-based classifiers to predict hemolysis, solubility, and resistance to nonspecific interactions of peptides and achieve competitive results compared with state-of-the-art models. The hemolytic model predicts the ability for a peptide to lyse red blood cells and is intended to be applied to peptides between 1 and 190 residues, L- and canonical amino acids (AUROC and accuracy of 0.84 and 84.0%, respectively). The hemolysis training data set is from sequences thought to be antimicrobial or clinically relevant, so it may not generalize to all possible peptides. Our solubility model, MahLooL, is trained with data mostly containing long sequences; thus, it may not be as applicable to solid-phase synthesized peptides. MahLooL provides state-of-the-art sequence-based solubility predictions for short peptides (<50) with AUROC and accuracy of 0.95 and 91.3%, respectively. However, its accuracy is lower for long peptide sequences (>100). Its intended use is for peptides or proteins expressed in *E. coli* that are less than 200 residues long and may provide solubility predictions more broadly applicable. The nonfouling model predicts the ability for a peptide to resist nonspecific interactions and is intended to be applied to short peptides between 2 and 20 residues (AUROC and accuracy of 0.93 and 82.0%, respectively). The nonfouling training data mostly contain short sequences, where negative examples have insoluble peptides overrepresented, so the accuracy may be inflated if only comparing soluble peptides.

5. CONCLUSIONS

Our proposed RNN models allow for automatic extraction of features from peptide sequences and remove the reliance on domain experts for feature construction. Moreover, these models are implemented in JavaScript, so that they can run on a static website through a browser on users' phones or desktops. This *serverless* approach removes the conventional dependence of deep learning models in cheminformatics on third-party hosted servers, thus reduces cost, increases flexibility, accessibility, and promotes open science. Our work is impactful in two ways; First, we demonstrate a new paradigm for sharing methods seamlessly and without servers. This should enable better dissemination of research from a broader community of chemists. Second, we demonstrate on a solubility predictor that outperforms existing SOTA for short peptide sequences. Using the same architecture, we achieve competitive results for hemolysis and nonfouling predictions that should enable better design of peptide molecules, given information on their sequence only.

■ ASSOCIATED CONTENT

Data Availability Statement

All data and code used to produce results in this study are publicly available in the following GitHub repository: https://github.com/ur-whitelab/peptide-dashboard. The JavaScript implementation of the models is available at https://peptide.bio/.

■ AUTHOR INFORMATION

Corresponding Author

Andrew D. White − *Department of Chemical Engineering, University of Rochester, Rochester, New York 14627, United States;* orcid.org/0000-0002-6647-3965;
Email: andrew.white@rochester.edu

Author

Mehrad Ansari − *Department of Chemical Engineering, University of Rochester, Rochester, New York 14627, United States*

Complete contact information is available at:
https://pubs.acs.org/10.1021/acs.jcim.2c01317

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors thank Mohammad Madani at the University of Connecticut for providing support on the implementation of DSResSol (1). Research reported in this work was supported by the National Institute of General Medical Sciences of the National Institutes of Health under award number R35GM137966. We thank the Center for Integrated Research Computing (CIRC) at University of Rochester for providing computational resources and technical support.

■ ADDITIONAL NOTE

[a] https://github.com/ur-whitelab/peptide-dashboard/blob/master/examples/Quick_start.ipynb

REFERENCES

(1) Dara, S.; Tumma, P. Feature Extraction By Using Deep Learning: A Survey. In *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, 2018; pp 1795−1801.

(2) Hinton, G. E.; Osindero, S.; Teh, Y.-W. A fast learning algorithm for deep belief nets. *Neural computation* **2006**, *18*, 1527−1554.

(3) Zhang, L.; Tan, J.; Han, D.; Zhu, H. From machine learning to deep learning: progress in machine intelligence for rational drug discovery. *Drug discovery today* **2017**, *22*, 1680−1685.

(4) Hwang, S.; Gou, Z.; Kuznetsov, I. B. DP-Bind: a web server for sequence-based prediction of DNA-binding residues in DNA-binding proteins. *Bioinformatics* **2007**, *23*, 634−636.

(5) Zeng, S.; Mao, Z.; Ren, Y.; Wang, D.; Xu, D.; Joshi, T. G2PDeep: a web-based deep-learning framework for quantitative phenotype prediction and discovery of genomic markers. *Nucleic Acids Res.* **2021**, *49*, W228−W236.

(6) Savojardo, C.; Martelli, P. L.; Fariselli, P.; Casadio, R. DeepSig: deep learning improves signal peptide detection in proteins. *Bioinformatics* **2018**, *34*, 1690−1696.

(7) 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**, *20*, 882−894.

(8) Tsutsumi, K.; Goshtasbi, K.; Risbud, A.; Khosravi, P.; Pang, J. C.; Lin, H. W.; Djalilian, H. R.; Abouzari, M. A Web-Based Deep Learning Model for Automated Diagnosis of Otoscopic Images. *Otology & Neurotology* **2021**, *42*, e1382−e1388.

(9) Li, G.; Iyer, B.; Prasath, V. S.; Ni, Y.; Salomonis, N. DeepImmuno: deep learning-empowered prediction and generation of immunogenic peptides for T-cell immunity. *Briefings in bioinformatics* **2021**, *22*, bbab160.

(10) Park, S.-J.; Yoon, B.-H.; Kim, S.-K.; Kim, S.-Y. GENT2: an updated gene expression database for normal and tumor tissues. *BMC medical genomics* **2019**, *12*, 1−8.

(11) Goswami, C. P.; Nakshatri, H. PROGgeneV2: enhancements on the existing database. *BMC cancer* **2014**, *14*, 1−6.

(12) Aguirre-Gamboa, R.; Gomez-Rueda, H.; Martínez-Ledesma, E.; Martínez-Torteya, A.; Chacolla-Huaringa, R.; Rodriguez-Barrientos, A.; Tamez-Pena, J. G.; Trevino, V. SurvExpress: an online biomarker validation tool and database for cancer gene expression data using survival analysis. *PloS one* **2013**, *8*, No. e74250.

(13) Koch, A.; De Meyer, T.; Jeschke, J.; Van Criekinge, W. MEXPRESS: visualizing expression, DNA methylation and clinical TCGA data. *BMC genomics* **2015**, *16*, 1−6.

(14) Goswami, C. P.; Nakshatri, H. PROGmiR: a tool for identifying prognostic miRNA biomarkers in multiple cancers using publicly available data. *Journal of clinical bioinformatics* **2012**, *2*, 23.

(15) Aguirre-Gamboa, R.; Trevino, V. SurvMicro: assessment of miRNA-based prognostic signatures for cancer clinical outcomes by multivariate survival analysis. *Bioinformatics* **2014**, *30*, 1630−1632.

(16) Anaya, J. OncoLnc: linking TCGA survival data to mRNAs, miRNAs, and lncRNAs. *PeerJ. Computer Science* **2016**, *2*, No. e67.

(17) Li, J.; Han, L.; Roebuck, P.; Diao, L.; Liu, L.; Yuan, Y.; Weinstein, J. N.; Liang, H. TANRIC: an interactive open platform to explore the function of lncRNAs in cancer. *Cancer research* **2015**, *75*, 3728−3737.

(18) Chen, M.-J. M.; Li, J.; Wang, Y.; Akbani, R.; Lu, Y.; Mills, G. B.; Liang, H. TCPA v3. 0: an integrative platform to explore the pan-cancer analysis of functional proteomic data. *Molecular & Cellular Proteomics* **2019**, *18*, S15−S25.

(19) Borcherding, N.; Bormann, N. L.; Voigt, A. P.; Zhang, W. TRGAted: A web tool for survival analysis using protein data in the Cancer Genome Atlas. *F1000Research* **2018**, *7*, 1235.

(20) Modhukur, V.; Iljasenko, T.; Metsalu, T.; Lokk, K.; Laisk-Podar, T.; Vilo, J. MethSurv: a web tool to perform multivariable survival analysis using DNA methylation data. *Epigenomics* **2018**, *10*, 277−288.

(21) Gao, J.; Aksoy, B. A.; Dogrusoz, U.; Dresdner, G.; Gross, B.; Sumer, S. O.; Sun, Y.; Jacobsen, A.; Sinha, R.; Larsson, E.; Cerami, E.; Sander, C.; Schultz, N. Integrative analysis of complex cancer genomics and clinical profiles using the cBioPortal. *Science signaling* **2013**, *6*, pl1−pl1.

(22) Zheng, H.; Zhang, G.; Zhang, L.; Wang, Q.; Li, H.; Han, Y.; Xie, L.; Yan, Z.; Li, Y.; An, Y.; Dong, H.; Zhu, W.; Guo, X. Comprehensive review of web servers and bioinformatics tools for cancer prognosis analysis. *Frontiers in oncology* **2020**, *10*, 68.

(23) May, M.; Brody, H. Nature index 2015 global. *Nature* **2015**, *522*, S1−S1.

(24) Melo, M. C.; Maasch, J. R.; de la Fuente-Nunez, C. Accelerating antibiotic discovery through artificial intelligence. *Communications biology* **2021**, *4*, 1−13.

(25) Littmann, M.; Selig, K.; Cohen-Lavi, L.; Frank, Y.; Hönigschmid, P.; Kataka, E.; Mösch, A.; Qian, K.; Ron, A.; Schmid, S.; et al. Validity of machine learning in biology and medicine increased through collaborations across fields of expertise. *Nature Machine Intelligence* **2020**, *2*, 18−24.

(26) Baek, M.; DiMaio, F.; Anishchenko, I.; Dauparas, J.; Ovchinnikov, S.; Lee, G. R.; Wang, J.; Cong, Q.; Kinch, L. N.; Schaeffer, R. D.; et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **2021**, *373*, 871−876.

(27) Baldini, I.; Castro, P.; Chang, K.; Cheng, P.; Fink, S.; Ishakian, V.; Mitchell, N.; Muthusamy, V.; Rabbah, R.; Slominski, A., et al. *Research Advances in Cloud Computing*; Springer, 2017; pp 1−20.

(28) John, A.; Muenzen, K.; Ausmees, K. Evaluation of serverless computing for scalable execution of a joint variant calling workflow. *PLoS One* **2021**, *16*, No. e0254363.

(29) Mack, C. A. Fifty years of Moore's law. *IEEE Transactions on semiconductor manufacturing* **2011**, *24*, 202−207.

(30) Cherkasov, A.; Muratov, E. N.; Fourches, D.; Varnek, A.; Baskin, I. I.; Cronin, M.; Dearden, J.; Gramatica, P.; Martin, Y. C.; Todeschini, R.; et al. QSAR modeling: where have you been? Where are you going to? *Journal of medicinal chemistry* **2014**, *57*, 4977−5010.

(31) Deng, B.; Ni, X.; Zhai, Z.; Tang, T.; Tan, C.; Yan, Y.; Deng, J.; Yin, Y. New quantitative structure−activity relationship model for angiotensin-converting enzyme inhibitory dipeptides based on integrated descriptors. *Journal of agricultural and food chemistry* **2017**, *65*, 9774−9781.

(32) Wang, Y.-T.; Russo, D. P.; Liu, C.; Zhou, Q.; Zhu, H.; Zhang, Y.-H. Predictive modeling of angiotensin I-converting enzyme inhibitory peptides using various machine learning approaches. *Journal of agricultural and food chemistry* **2020**, *68*, 12132−12140.

(33) Guan, X.; Liu, J. QSAR study of angiotensin I-converting enzyme inhibitory peptides using SVHEHS descriptor and OSC-SVM. *International Journal of Peptide Research and Therapeutics* **2019**, *25*, 247−256.

(34) 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. *J. Chem. Inf. Model.* **2018**, *58*, 1141−1151.

(35) Barrett, R.; Jiang, S.; White, A. D. Classifying antimicrobial and multifunctional peptides with Bayesian network models. *Peptide Science* **2018**, *110*, No. e24079.

(36) Lu, Y.; Qiu, Q.; Kang, D.; Liu, J. QSAR Modeling of Antimicrobial Peptides Based on Their Structural and Physicochemical Properties. *Journal of Biology and Nature* **2018**, 120−126.

(37) Das, P.; Sercu, T.; Wadhawan, K.; Padhi, I.; Gehrmann, S.; Cipcigan, F.; Chenthamarakshan, V.; Strobelt, H.; Dos Santos, C.; Chen, P.-Y.; et al. Accelerated antimicrobial discovery via deep generative models and molecular dynamics simulations. *Nature Biomedical Engineering* **2021**, *5*, 613−623.

(38) Chen, N.; Chen, J.; Yao, B.; Li, Z. QSAR study on antioxidant tripeptides and the antioxidant activity of the designed tripeptides in free radical systems. *Molecules* **2018**, *23*, 1407.

(39) Deng, B.; Long, H.; Tang, T.; Ni, X.; Chen, J.; Yang, G.; Zhang, F.; Cao, R.; Cao, D.; Zeng, M.; et al. Quantitative structure-activity relationship study of antioxidant tripeptides based on model population analysis. *International journal of molecular sciences* **2019**, *20*, 995.

(40) Olsen, T. H.; Yesiltas, B.; Marin, F. I.; Pertseva, M.; Garcia-Moreno, P. J.; Gregersen, S.; Overgaard, M. T.; Jacobsen, C.; Lund, O.; Hansen, E. B.; Marcatili, P. AnOxPePred: using deep learning for the prediction of antioxidative properties of peptides. *Sci. Rep.* **2020**, *10*, 1−10.

(41) Madani, M.; Lin, K.; Tarakanova, A. DSResSol: A sequence-based solubility predictor created with Dilated Squeeze Excitation Residual Networks. *International Journal of Molecular Sciences* **2021**, *22*, 13555.

(42) Khurana, S.; Rawi, R.; Kunji, K.; Chuang, G.-Y.; Bensmail, H.; Mall, R. DeepSol: a deep learning framework for sequence-based protein solubility prediction. *Bioinformatics* **2018**, *34*, 2605−2613.

(43) Rawi, R.; Mall, R.; Kunji, K.; Shen, C.-H.; Kwong, P. D.; Chuang, G.-Y. PaRSnIP: sequence-based protein solubility prediction using gradient boosting machine. *Bioinformatics* **2018**, *34*, 1092−1098.

(44) Hon, J.; Marusiak, M.; Martinek, T.; Kunka, A.; Zendulka, J.; Bednar, D.; Damborsky, J. SoluProt: prediction of soluble protein expression in Escherichia coli. *Bioinformatics* **2021**, *37*, 23−28.

(45) Hebditch, M.; Carballo-Amador, M. A.; Charonis, S.; Curtis, R.; Warwicker, J. Protein−Sol: a web tool for predicting protein solubility from sequence. *Bioinformatics* **2017**, *33*, 3098−3100.

(46) Smialowski, P.; Doose, G.; Torkler, P.; Kaufmann, S.; Frishman, D. PROSO II−a new method for protein solubility prediction. *FEBS journal* **2012**, *279*, 2192−2200.

(47) Timmons, P. B.; Hewage, C. M. HAPPENN is a novel tool for hemolytic activity prediction for therapeutic peptides which employs neural networks. *Sci. Rep.* **2020**, *10*, 1−18.

(48) Chaudhary, K.; Kumar, R.; Singh, S.; Tuknait, A.; Gautam, A.; Mathur, D.; Anand, P.; Varshney, G. C.; Raghava, G. P. A web server and mobile app for computing hemolytic potency of peptides. *Sci. Rep.* **2016**, *6*, 1−13.

(49) Win, T. S.; Malik, A. A.; Prachayasittikul, V.; S Wikberg, J. E.; Nantasenamat, C.; Shoombuatong, W. HemoPred: a web server for predicting the hemolytic activity of peptides. *Future medicinal chemistry* **2017**, *9*, 275−291.

(50) Hasan, M. M.; Schaduangrat, N.; Basith, S.; Lee, G.; Shoombuatong, W.; Manavalan, B. HLPpred-Fuse: improved and robust prediction of hemolytic peptide and its activity by fusing multiple feature representation. *Bioinformatics* **2020**, *36*, 3350−3356.

(51) Pirtskhalava, M.; Amstrong, A. A.; Grigolava, M.; Chubinidze, M.; Alimbarashvili, E.; Vishnepolsky, B.; Gabrielian, A.; Rosenthal, A.; Hurt, D. E.; Tartakovsky, M. DBAASP v3: database of antimicrobial/cytotoxic activity and structure of peptides as a resource for development of new therapeutics. *Nucleic acids research* **2021**, *49*, D288−D297.

(52) Berman, H. M.; Westbrook, J. D.; Gabanyi, M. J.; Tao, W.; Shah, R.; Kouranov, A.; Schwede, T.; Arnold, K.; Kiefer, F.; Bordoli, L.; et al. The protein structure initiative structural genomics knowledgebase. *Nucleic acids research* **2009**, *37*, D365−D368.

(53) White, A. D.; Nowinski, A. K.; Huang, W.; Keefe, A. J.; Sun, F.; Jiang, S. Decoding nonspecific interactions from nature. *Chemical Science* **2012**, *3*, 3488−3494.

(54) White, A. D.; Huang, W.; Jiang, S. Role of nonspecific interactions in molecular chaperones through model-based bio-informatics. *Biophysical journal* **2012**, *103*, 2484−2491.

(55) Chollet, F. *Keras*, 2015.

(56) Abadi, M. et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, 2015. https://www.tensorflow.org/.

(57) Sutskever, I.; Martens, J.; Hinton, G. E. *Generating Text with Recurrent Neural Networks*; ICML, 2011.

(58) Segler, M. H.; Kogej, T.; Tyrchan, C.; Waller, M. P. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science* **2018**, *4*, 120−131.

(59) Ye, Y.; Wang, J.; Xu, Y.; Wang, Y.; Pan, Y.; Song, Q.; Liu, X.; Wan, J. MATHLA: a robust framework for HLA-peptide binding prediction integrating bidirectional LSTM and multiple head attention mechanism. *BMC bioinformatics* **2021**, *22*, 1−12.

(60) Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural computation* **1997**, *9*, 1735−1780.

(61) Barrett, R.; White, A. D. Investigating Active Learning and Meta-Learning for Iterative Peptide Design. *J. Chem. Inf. Model.* **2021**, *61*, 95−105.

(62) Hanley, J. A.; McNeil, B. J. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* **1982**, *143*, 29−36.

(63) Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. *arXiv Preprint*, arXiv:1412.6980, 2014.

(64) Parzen, E. On estimation of a probability density function and mode. *annals of mathematical statistics* **1962**, *33*, 1065−1076.

(65) Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018; pp 7132−7141.

(66) He, K.; Zhang, X.; Ren, S.; Sun, J. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, 2016; pp 630−645.

(67) Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv Preprint*, arXiv:1511.07122, 2016.

(68) Alley, E. C.; Khimulya, G.; Biswas, S.; AlQuraishi, M.; Church, G. M. Unified rational protein engineering with sequence-based deep representation learning. *Nat. Methods* **2019**, *16*, 1315−1322.

(69) Ma, E. J.; Kummer, A. Reimplementing Unirep in JAX. *bioRxiv*, 2020.

(70) Breiman, L. Random forests. *Machine learning* **2001**, *45*, 5−32.

(71) Capecchi, A.; Cai, X.; Personne, H.; Köhler, T.; van Delden, C.; Reymond, J.-L. Machine learning designs non-hemolytic antimicrobial peptides. *Chemical Science* **2021**, *12*, 9221−9232.

(72) Chang, C. C. H.; Song, J.; Tey, B. T.; Ramanan, R. N. Bioinformatics approaches for improved recombinant protein production in Escherichia coli: protein solubility prediction. *Briefings in bioinformatics* **2014**, *15*, 953−962.

(73) Mitchell, M.; Wu, S.; Zaldivar, A.; Barnes, P.; Vasserman, L.; Hutchinson, B.; Spitzer, E.; Raji, I. D.; Gebru, T. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, And Transparency*, 2019; pp 220−229.