Liu *et al. J. Mater. Inf.* **2025**, *5*, 17
DOI: 10.20517/jmi.2024.91

Journal of Materials Informatics

Research Article

Open Access

Check for updates

Integrating sequence and chemical insights: a co-modeling AI prediction framework for peptides

Zihan Liu[1,#], Meiru Yan[2,3,#], Zhihui Zhu[2,3,#], Yongfu Guo[4], Mouzheng Xu[4], Jiaqi Wang[2,3,*]

[1]Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China.
[2]Wisdom Lake Academy of Pharmacy, Xi'an Jiaotong-Liverpool University, Suzhou 215123, Jiangsu, China.
[3]Jiangsu Province Higher Education Key Laboratory of Cell Therapy Nanoformulation, Xi'an Jiaotong-Liverpool University, Suzhou 215123, Jiangsu, China.
[4]XJTLU High Performance Computing Platform, Management Information Technology and System Office, Xi'an Jiaotong-Liverpool University, Suzhou 215123, Jiangsu, China.
[#]Authors contributed equally.

[*]**Correspondence to:** Dr. Jiaqi Wang, Wisdom Lake Academy of Pharmacy, Xi'an Jiaotong-Liverpool University, No 111, Renai Road, Suzhou 215123, Jiangsu, China. E-mail: Jiaqi.Wang02@xjtlu.edu.cn

How to cite this article: Liu, Z.; Yan, M.; Zhu, Z.; Guo, Y.; Xu, M.; Wang, J. Integrating sequence and chemical insights: a co-modeling AI prediction framework for peptides. *J. Mater. Inf.* **2025**, *5*, 17. https://dx.doi.org/10.20517/jmi.2024.91

Received: 23 Dec 2024 **First Decision:** 17 Jan 2025 **Revised:** 24 Jan 2025 **Accepted:** 6 Feb 2025 **Published:** 27 Feb 2025

Academic Editors: Hao Li, Lingyan Feng **Copy Editor:** Pei-Yun Wang **Production Editor:** Pei-Yun Wang

Abstract

Understanding the impact of the primary structure of peptides on a range of physicochemical properties is crucial for the development of various applications. Peptides can be conceptualized as sequences of amino acids in their biological representation and as molecular architectures composed of atoms and chemical bonds in their chemical representation. This study examines the influence of different biological and chemical representations of peptides on the local interpretability and accuracy of their respective prediction models and has developed "feature attribution" methodologies based on these representations. The effectiveness of these methodologies is validated through physicochemical analyses, specifically within the context of peptide aggregation propensity (AP) prediction, with training datasets derived from high-throughput molecular dynamics (MD) simulations. Our findings reveal significant discrepancies in the attribution extracted from sequence-based and chemical structure-based representations, which has led to the proposal of a co-modeling framework that integrates insights from both perspectives. Empirical comparisons have demonstrated that the contrastive learning-based co-modeling framework excels in terms of effectiveness and efficiency. This research not only extends the applicability of the attribution method but also lays the groundwork for elucidating the intrinsic mechanisms governing peptide activities and functions with the aid of domain-specific knowledge. Moreover, the co-modeling strategy is poised to

© The Author(s) 2025. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, sharing, adaptation, distribution and reproduction in any medium or format, for any purpose, even commercially, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

enhance the precision of downstream applications and facilitate future endeavors in drug discovery and protein engineering.

Keywords: Deep learning, molecular dynamics, peptide, aggregation propensity, feature attribution

INTRODUCTION

Peptides, which are short chains of amino acids[1], have become a major research focus in the field of "AI for Science" due to their substantial potential across a variety of fields, including drug discovery[2,3], tissue engineering[4], and the development of semiconducting materials[5,6], catalysts[7,8], *etc*. By leveraging the predictive capabilities of AI, researchers have been able to analyze and predict secondary and tertiary structures[9], molecular interactions[10], and physicochemical properties[11-13] of peptides in the vast chemical space (over 20^{50} sequences). These scientific advancements have led to transformative advances in high-throughput screening techniques, paradigm-shifting the development of peptide-based functional materials and thereby initiating a new era across multiple industries, including healthcare, semiconductors, nanotechnology, *etc*.

Characterized by their short length (typically no more than 50 amino acids[14]), peptides inherently possess simpler secondary structures[15] and less stable tertiary structures than proteins[16]. This intrinsic simplicity has led AI-driven peptide research to focus on predicting correlations between primary structures (e.g., sequence) and properties. The primary structure of a peptide can be represented biologically as a sequence of amino acids or chemically as a molecular graph consisting of atoms and chemical bonds. Machine learning models related to amino acid sequences include support vector machines (SVM)[17], random forest (RF)[18], and multilayer perceptron (MLP)[19], which encode the primary structure data into one-dimensional vectors[11,20]. In recent years, sequential models, such as recurrent neural networks (RNN)[21], long short-term memory (LSTM)[22], and transformers[23] have been extensively utilized to process sequential data[24-26]. Simultaneously, geometric deep learning models[27] that address molecular structures involving atoms and chemical bonds primarily employ graph neural networks (GNNs)[28], which are designed to learn from graph-structured data[29,30].

In this study, we delve into the impact of distinct representations of the same peptide - from both biological and chemical perspectives - on the local interpretability of their respective sequence-based and graph-based prediction models. Inspired by the up-to-date work on dual-modality models for molecular representations[31,32] and the successful application of attribution methods for small molecules[33], we have developed gradient-based feature attribution techniques tailored to sequence- and graph-based peptide prediction models. The goal of this research is to identify the specific amino-acid level features that contribute to peptide aggregation, a prerequisite of self-assembling. The reliability of these attributions is validated by analyses focused on the prediction of peptide aggregation propensity (AP) and by comparison with established aggregation principles[34-37]. This research extends the application of the attribution method to a broader range of peptide prediction models and has the capacity to reveal the intrinsic mechanisms governing peptide activities and functions by integrating additional domain-specific knowledge.

The analysis of the attribution results aligns with current domain knowledge, yet it also reveals intriguing discrepancies between the attributions derived from sequence-based and molecular graph-based prediction models, suggesting that different levels of representation of the prediction models would capture inconsistent information from the same peptide. These findings prompt us to propose a co-modeling framework that integrates both biological and chemical representations for more comprehensive analysis of peptides. To implement this co-modeling framework, we propose several fundamental feature integration

methods and have demonstrated through extensive comparisons that the contrastive learning-based co-modeling framework outperforms other methods [i.e., weighted sum (WS)[38], concatenation (Concat)[39], cross-attention (CA)[23], and compact bilinear pooling (CBP)[40]] in terms of efficacy and efficiency. In this implementation, we incorporate contrastive learning loss as a regularization term to enhance the correlation between the hidden features extracted from two representations of the same peptide modality, aiming to enrich the chemical information within sequence-based models. To validate the reliability of the proposed co-modeling approach, we have performed a comparative analysis of the accuracy between co-modeling and non-co-modeling methods across four datasets containing both regression and classification tasks.

In summary, the main contributions of this work are as follows: (1) We have developed gradient-based feature attribution methods for peptide prediction models, which are employed to evaluate the significance of amino acids at each position within the peptide sequence to various physicochemical properties. The reliability of the attribution was confirmed using the AP dataset generated by high-throughput molecular dynamics (MD) simulations; (2) Our research has revealed intriguing discrepancies between the models trained on sequence data and those that are trained on molecular data, highlighting the substantial impact of the peptide's primary structure on prediction results; (3) We propose a novel co-modeling framework that leverages contrastive learning to integrate biological and chemical representations for a more comprehensive peptide analysis. This research holds the potential to revolutionize our understanding of deep learning applications in peptide-related tasks and to streamline future studies in drug discovery and protein engineering.

MATERIALS AND METHODS

Coarse-grained MD simulations for obtaining AP

The coarse-grained MD (CGMD) simulations provide low-noise, consistent, and highly reproducible AP data within aqueous environment with pH = 7, covering samples from pentapeptides to decapeptides. Specifically, the AP values in our study were measured using GROMACS open-sourced package and the Martini 2.2 force field[41,42] to efficiently model peptide aggregation. For simulation preparation, all-atom peptide structures were first generated based on the CHARMM36 force field[43,44] and subsequently coarse-grained via the martinize.py script[41]. The coarse-graining significantly reduced computational cost by encoding four atoms into a single bead, each characterized by specific properties such as polarity, charge, and hydrogen bonding capacity. Approximately 150 pentapeptides to 81 decapeptides of the same type were placed in a simulation box of 15 nm × 15 nm × 15 nm, solvated with 28,400 water beads, achieving solvent concentrations of 0.074 mol/L for pentapeptides to 0.040 mol/L for decapeptides[34]. The system was neutralized with Na^+ or Cl^- ions as needed, followed by energy minimization and thermodynamic equilibration at 300 K and 1 bar using the Berendsen algorithm. All simulations are performed for 125 ns (equivalent to 500 ns due to coarse-graining effect) to ensure that the AP values were fully converged[35].

To quantify AP, we defined it as the ratio of the solvent-accessible surface area (SASA) at the beginning of the simulation ($SASA_{t=beg}$) to that at the end ($SASA_{t=end}$), i.e., $AP = SASA_{t=beg}/SASA_{t=end}$. Initially, AP equals to 1. For peptides undergoing aggregation, the SASA decreases as clustering occurs, causing AP to rise above 1. In contrast, for peptides that do not aggregate, the SASA remains relatively constant, with fluctuations around 1 due to minor computational noise.

To generate the training dataset, we performed simulations for a total of 62,159 peptide types, encompassing pentapeptides to decapeptides, thereby ensuring comprehensive coverage of various peptide lengths and compositions. The resulting AP values constituted a robust and high-throughput dataset, suitable for training and validating machine learning models.

Amino acid-level feature attribution

Peptides, whether represented by sequences or molecular graphs, are characterized by discrete variables. Consequently, attributing features to the primary structure of peptides necessitates addressing the challenge of assessing the activation of non-differentiable, discrete input variables.

Notably, discrete variables are transformed into differentiable vectors upon embedding in a continuous, high-dimensional space. The principle of attribution for predictive models entails extracting the gradient saliency in the continuous feature space after embedding and subsequently integrating this information back into the discrete input space. For a peptide object X, $X = \{x_1, x_2, ..., x_n\}$ represents the primary structure of the peptide composed of n discrete components. In the context of sequential models, n corresponds to the number of amino acids; for graphical models, n represents the number of nodes after coarse-graining in molecular simulations, with the coarse-graining rules taken from reference[42]. The feature vectors after the embedding layer are denoted as $H = \{h_1, h_2, ..., h_n\}$, where the feature vector of the i-th component is denoted as h_i = Embedding(x_i). The process of determining the saliency of each component x_i based on H is as follows:

$$h_i^{saliency} = h_i \circ \sum_{k=1}^{m} \frac{\partial \mathrm{L}_{attr}\left(\frac{k}{m} H\right)}{\partial h_i} \tag{1}$$

$$x_i^{saliency} = \frac{\left\| h_i^{saliency} \right\|_1}{\sum_{j=1}^{n} \left\| h_i^{saliency} \right\|_1} \tag{2}$$

Equation (1) calculates the gradient saliency on the feature vectors H, which are the output of the embedding layer. In this context, the loss function L_{attr} depends on the label and the specific task. For classification tasks, the loss function can be a cross-entropy loss, while for regression tasks, it may be directly equivalent to the predicted output of the model. Equation (1) gives the backpropagation process to derive the gradient with respect to the embedding layer H, where $\circ$ represents the element-wise product, i.e., the multiplication of the feature value and the gradient value, and m represents the number of steps used in the Riemann approximation of the integral. Equation (1) uses the integrated gradient method[45] to circumvent issues of gradient saturation, asymptoting the features in the embedding layer from an initial zero matrix to H. The product of the gradient value and the feature value represents the saliency of that particular feature.

Equation (2) first integrates the saliency of the i-th feature vector $h_i^{saliency}$ into the saliency of the discrete variable x_i. The saliency of each component x_i is calculated as the sum of the saliency of all feature vectors in their respective embedding layer. Subsequently, the saliency values for $x_1, x_2, ..., x_n$ are then normalized to ensure that the sum of the saliency of all components within each peptide X is equal to 1.

Sequence-based models can directly calculate the importance at the amino acid level through Equations (1) and (2). In contrast, graph-based models operate at the atomic level, necessitating additional integration to evaluate the amino acid contributions to peptide's physicochemical properties (i.e., AP). The sum of the saliencies of the nodes belonging to an amino acid in the molecular graph represents the saliency of that amino acid.

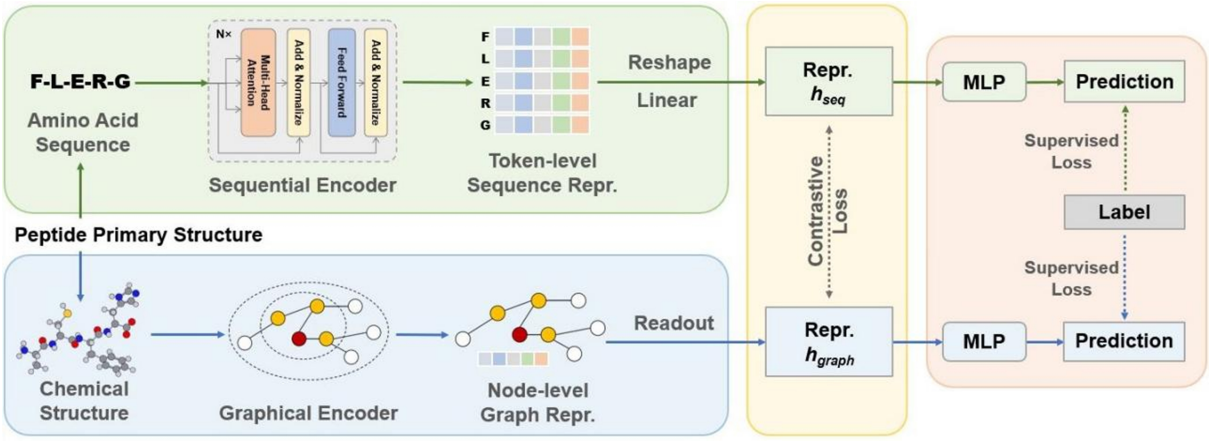

Figure 1. Schematic representation of the contrastive learning-based co-modeling framework for sequences and molecular graphs. This framework consists of a sequence-based encoder module (top left), a graph-based encoder module (bottom left), a fusion module (middle), and a predictor (right).

This approach can be broadly generalized to other tasks. Specifically, it can be tailored to discriminative tasks that take amino acid sequences or molecular graphs as input and predict classification or regression labels for peptides. By incorporating additional domain-specific knowledge, it becomes feasible to elucidate the underlying mechanisms governing the activities and functions of peptides.

Co-modeling of sequence and molecular graph

To fully exploit the diverse information extracted, we propose a co-modeling framework designed for concurrently modeling peptides using both sequence and molecular graphs (i.e., chemical information). As shown in Figure 1, the proposed co-modeling framework includes a sequence-based and a graph-based encoder module, a fusion module responsible for integrating the representations (implemented via contrastive learning as shown in Figure 1) and an MLP predictor. The fusion module, which is designed to combine the representations learned by the sequence-based and graph-based encoders, can be implemented using various principles. A number of fusion methods are tested in this research, including WS, Concat, CBP, CA, and contrastive learning [Supplementary Materials].

The WS, a conventional ensemble learning technique, suffers from information loss due to the substantial divergence between the representations extracted from sequences and molecular graphs. Concat, on the other hand, avoids this problem by directly merging the representations into a new vector. However, both WS and Concat fail to capture the correlations between representations. CBP, CA, and contrastive learning resolve this issue by effectively merging representations based on input correlations. Both CBP and CA explicitly integrate features so that the noisier of the two input representations affects the fused representation. In contrast, contrastive learning employs an implicit fusion of representations. By treating the representations from the different representations of the same peptide as positive pairs and those from different peptides as negative pairs, contrastive learning endeavors to improve the correlation between the representations extracted from two depictions. Since both representations capture primary structure information, there is a higher degree of similarity between them compared to representations from multimodalities. Therefore, we incorporate a contrastive loss-based regularization term during the training phase. The pseudocode of the training and testing of the co-modeling framework is presented in Figure 2.

Algorithm 1 Training and Testing Phases of Contrastive-based Co-Modelling Framework

Require: Training and testing datasets, hyperparameter λ, learning rate γ
--Training Phase--
  Initialize learnable parameters randomly
  while the model has not yet reached convergence **do**
    Sample a batch B from the training set
    for each peptide $(x, y) \in B$ **do**
      Extract the sequence information x_{seq} and the chemical structure x_{graph}
      Compute supervised loss by Equation S1
      Compute unsupervised InfoNCE loss by Equation S2
    end for
    Calculate the gradients on learnable parameters by backpropagation
    Update learnable parameters with learning rate γ
  end while
--Testing Phase--
  Load the sequence-based representation extractor $f_e(\cdot)$ and the MLP predictor $f_p(\cdot)$
  for each peptide x in test set **do**
    Forward the model and get the prediction $f_p(f_e(x_{seq}))$
  end for

Figure 2. Pseudocode of the training and testing protocols of the contrastive-based co-modeling framework.

AI experiments

This section details the experimental setup, including the datasets, comparative baselines, and configuration parameters, to validate and elucidate the efficacy of the co-modeling framework and compare the performance of the various fusion methods described previously.

Datasets

Dataset statistics are presented in Table 1. The datasets for the regression tasks consist of the AP dataset from CGMD simulations[35] and retention time (RT) prediction dataset from the previous project PXD006109[46]. The MD simulations offer low-noise, consistent, and highly reproducible AP data, covering samples from pentapeptides to decapeptides, characterizing the extent of peptide aggregation in aqueous environments. The AP dataset contains approximately 10,000 peptide samples of each length category, totaling over 60,000 entries. Notably, the AP values for peptide sequences in our dataset are fully dependent on the Martini 2.2 force field[41,42], offering a comprehensive understanding of the underlying principles governing aggregation. Therefore, the reliability of the proposed feature attributions can be substantiated by evaluating the consistency of the attributed importance of amino acids with the physicochemical principles prescribed in the Martini 2.2 force field.

For the classification tasks, the datasets include antimicrobial peptides (AMPs)[12] and peptide families from the PeptideDB (the data is deposited as Supplementary Materials)[47]. These datasets have been refined to include only peptide samples composed of the 20 standard natural amino acids, with a sequence length not exceeding 50 amino acids. In the PepDB dataset, class 0 denotes AMPs, 1 denotes peptide hormones, and 2 denotes toxins and venom peptides. In the AMP dataset, class 1 represents AMPs, while class 0 represents non-AMPs. The RT, PepDB, and AMP datasets are randomly divided into training, validation, and testing sets with an approximate ratio of 8:1:1.

Baselines

We aim to evaluate the performance difference between the co-modeling framework and non-co-modeling baselines and evaluate the implementations of different fusion modules. We follow the non-co-modeling

Table 1. Statistics of datasets

Dataset	Task	Samples	Classes	Length
AP	Regression	62,159	-	10
RT	Regression	121,215	-	50
AMP	Classification	9,321	2	50
PepDB	Classification	7,016	3	50

AP: Aggregation propensity; RT: retention time; AMP: anti-microbial peptide; PepDB: Peptide DataBase.

baselines established for the AP benchmark[35], which include sequence-based models such as RNN[21], LSTM[22], Bi-LSTM[22] and Transformer[23], and graph-based models such as graph convolutional networks (GCN)[48], graph attention networks (GAT)[49] and graph sampling and aggregation networks (GraphSAGE)[50]. Within the co-modeling framework, the sequence component employs multi-head attention blocks, while the molecular graph component leverages the graph convolution layers of GraphSAGE. The baseline implementations of the fusion module include WS, Concat, CA, and CBP. The method we employ for representation fusion via contrastive learning is termed Regularization by representation contrasting, or RepCon for short.

Implementation detail

The dimensions of the hidden and output layers of the sequence and graph encoders are both set to 64, and the feedforward network layer in the multi-head attention is set to 2048. The sequence encoder contains six multi-head attention blocks, with the number of attention heads set to 8. The number of graph convolutional layers in the graph encoder is set to 2. In the MLP predictors, the MLP contains four linear layers, utilizes the LeakyReLU activation function, and employs dropout for regularization. RepCon includes a hyperparameter λ for weighting loss terms. The λ controls the relative importance of the contrastive loss compared to the supervised loss. Generally, as a regularization term, the weight of the contrastive loss should be set so that its effect on the model weights is smaller than that of the supervised loss. Given the distinct nature of the mean squared error (MSE) loss and cross-entropy loss in their gradients when updating the model, λ is set to 2E-5 for regression tasks and 1E-2 for classification tasks through empirical experimentation. Note that RepCon is the combination of two end-to-end models. When the contrastive loss component is removed from RepCon, it effectively becomes two independent end-to-end networks, meaning that the non-co-modeling baselines inherently include an ablation study of RepCon. To evaluate the performance of the baseline models in regression tasks, we use mean absolute error (MAE), MSE, and the coefficient of determination (R^2). In classification tasks, classification accuracy (Acc.) serves as the metric of choice.

RESULTS AND DISCUSSION

Evaluating reliability via peptide aggregation attribution

The attribution analysis provides a quantitative evaluation of the influence of each amino acid on the AP of peptides. Preliminary validation of the amino acid attribution to aggregation was performed using decapeptides[37]. The findings suggest that highly polar amino acids (e.g., N and Q) and charged amino acids (e.g., D and K), significantly contribute to mitigating the aggregation tendency. In contrast, aromatic amino acids (e.g., F, W, and Y) have a substantial effect in promoting aggregation. These attribution results are consistent with previously reported aggregation rules[11,34,37,51].

Inconsistencies between sequence- and graph-based attribution

Sequence- and graph-based models, which are trained on sequence data and molecular graphs, respectively, achieve high levels of accuracy but provide inconsistent explanations for their prediction results. By

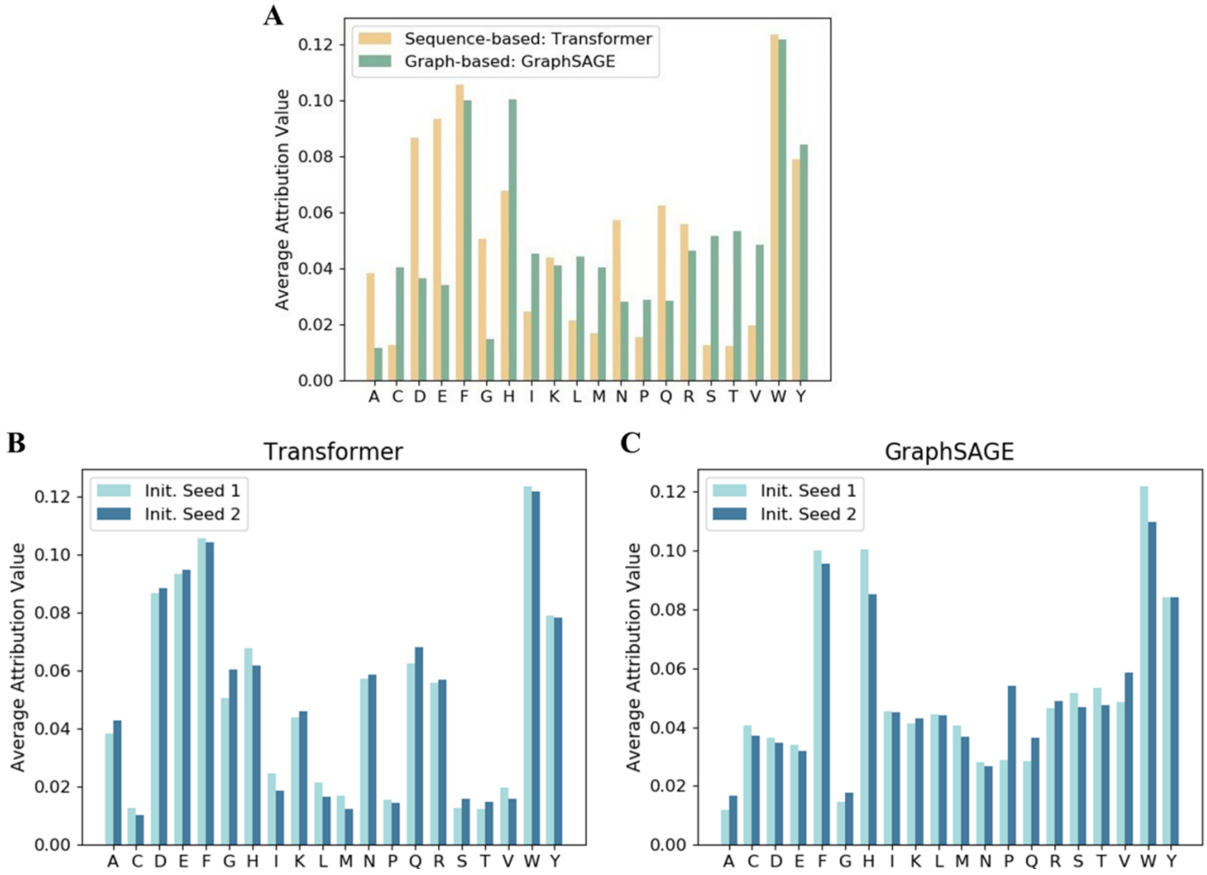

Figure 3. (A) Average attribution values of each amino acid in the AP prediction task. The *x*-axis represents the types of amino acids and y-axis shows the average attribution scores from the sequence-based (in yellow) and graph-based (in green) models; (B and C) Variation in attribution results for models with different random seed initializations for the Transformer (B) and the GraphSAGE (C) architectures. The results indicate that different initializations of the learnable parameters have minimal impact on the consistency of the attribution results. AP: Aggregation propensity; GraphSAGE: graph sampling and aggregation networks.

applying peptide attribution techniques to the Transformer model for sequence analysis and GraphSAGE for molecular graph analysis, we have identified significant discrepancies in attribution between these two types of models, as shown in Figure 3A. These discrepancies exceed the variation that can be attributed to the randomness inherent in model initialization, as detailed in Figure 3B and C.

Figure 3A presents the average attribution values for each amino acid category within the testing dataset, highlighting the importance of each amino acid in the AP prediction. Both models consistently assign high significance to aromatic amino acids F, W, and Y, and the charged amino acids K and R. However, there are notable discrepancies in the attribution of other amino acids. The sequence-based model prioritizes [A and G], which have simple side chains, [D and E] with negative charges, and [Q and N], which have polar and hydrophilic side chains. In contrast, the graph-based model highlights [C and M], which contain sulfur in their side chains, [S and T] with polar side chains, and [V, L, and I] with hydrophobic side chains. Providing a physical rationale for the observed discrepancies is challenging; however, compared with the established aggregation rules[37], the Transformer model is more consistent with published findings[37]. Specifically, D, E, R, and K are known to have significant negative effects on aggregation due to hydrophilicity and electrostatic repulsion, while F, W, and Y contribute positively to aggregation due to hydrophobicity and π-π interactions.

Figure 3B and C presents a comparative analysis of the attribution results for the sequence-based Transformer model across various initializations [Figure 3B] and for the graph-based GraphSAGE model [Figure 3C], both trained on the AP prediction task. The attribution results for the Transformer model exhibit minimal variance with respect to model initialization, indicating a robust and consistent attribution pattern across all amino acids. For GraphSAGE, while there are minor variations in the mean attribution values for certain amino acids, such as P, a predominantly consistent trend in attribution values is maintained for the majority of amino acids.

Based on this comparison, both the sequence-based Transformer and the graph-based GraphSAGE are minimally affected by model initialization. This experiment rules out the possibility that model initialization is the cause of the significant attribution differences between Transformer and GraphSAGE, and further proves that it is the representations of peptide primary structures and the mechanisms of the corresponding neural networks that lead to their attribution differences.

This attribution study highlights that different models can extract inconsistent features from the same peptide data, depending on whether the primary structure of the peptides is characterized biologically or chemically. This discrepancy suggests that the models relying solely on a single modality may fail to capture the full complexity of peptides, along with the potential benefits of a model that integrates both sequence and structure information to achieve more comprehensive feature extraction. Therefore, we propose a unified co-modeling framework that utilizes both sequence and graph representations of peptide primary structures, aiming to enhance the peptide feature extraction across multiple modalities.

Evaluation on regression tasks of co-modeling framework

Table 2 details the performance of a number of models on the AP and RT datasets, divided into three categories: sequence-based, graph-based, and co-modeling frameworks. The metrics used to evaluate performance include MAE, MSE, and R^2.

Among the sequence-based models, the Transformer model demonstrates superior performance on the AP dataset, achieving an MAE of 3.81E-2, an MSE of 2.33E-3, and an R^2 of 0.947 on the AP dataset. On the RT dataset, the Transformer model again shows the best performance, with an MAE of 1.57, an MSE of 5.02, and an R^2 of 0.991.

Among the graph-based models, the GraphSAGE model exhibits the highest performance. On the AP dataset, it achieves an MAE of 3.89E-2, an MSE of 2.44E-3, and an R^2 of 0.945. When applied to the RT dataset, the GraphSAGE model achieves an MAE of 2.57, an MSE of 12.9, and an R^2 of 0.977.

Among the co-modeling frameworks, the RepCon model demonstrates superior performance. On the AP dataset, it achieves the lowest MAE of 3.62E-2, the lowest MSE of 2.12E-3, and the highest R^2 of 0.953. Consistently, on the RT dataset, the RepCon model also exhibits the lowest MAE of 1.41, the lowest MSE of 4.40, and the highest R^2 of 0.993.

Summarizing the results above, we found that although R^2 values provide an overall assessment of model fit, they tend to plateau at high values (i.e., show minimal difference), especially in well-performing models. Metrics such as MAE and MSE, on the other hand, directly measure prediction errors and provide a clearer view of how accurately the models perform. For example, in the AP dataset, the co-modeling framework (RepCon) achieved an MAE of 3.62E-2 and an MSE of 2.12E-3, outperforming both sequence- and graph-based models. These lower error values highlight the superior accuracy of RepCon in predicting AP, even

Table 2. Sequence-based baselines, graph-based baselines, and co-modeling frameworks with different fusion module implementations on AP and RT datasets

Input	Model	AP			RT			Inference	
		MAE (× 10^{-2})	MSE (× 10^{-3})	R^2	MAE	MSE	R^2	Seq	Graph
Sequence	RNN	4.52 ± 0.10	3.23 ± 0.02	0.927 ± 0.013	1.95 ± 0.12	7.84 ± 0.44	0.986 ± 0.013	√	
	LSTM	4.28 ± 0.03	2.88 ± 0.04	0.935 ± 0.013	1.79 ± 0.04	6.59 ± 0.52	0.988 ± 0.014	√	
	Bi-LSTM	4.25 ± 0.06	2.87 ± 0.04	0.936 ± 0.011	1.54 ± 0.06	4.93 ± 0.33	0.991 ± 0.015	√	
	Transformer	**3.81** ± 0.06	**2.33** ± 0.03	**0.947** ± 0.014	**1.57** ± 0.09	**5.02** ± 0.21	**0.991** ± 0.014	√	
Graph	GCN	4.11 ± 0.03	2.76 ± 0.04	0.938 ± 0.017	3.34 ± 0.14	21.4 ± 1.32	0.962 ± 0.016		√
	GAT	4.09 ± 0.04	2.72 ± 0.06	0.939 ± 0.016	2.99 ± 0.16	17.3 ± 1.12	0.970 ± 0.017		√
	GraphSAGE	**3.89** ± 0.06	**2.44** ± 0.04	**0.945** ± 0.012	**2.57** ± 0.11	**12.9** ± 1.07	**0.977** ± 0.015		√
Co-modelin	WS	4.05 ± 0.06	2.68 ± 0.11	0.940 ± 0.011	1.92 ± 0.07	7.75 ± 0.98	0.986 ± 0.013	√	√
	Concat	3.75 ± 0.11	2.27 ± 0.06	0.949 ± 0.012	1.45 ± 0.08	4.67 ± 0.33	0.992 ± 0.014	√	√
	CA	3.79 ± 0.07	2.32 ± 0.08	0.948 ± 0.019	1.44 ± 0.05	4.52 ± 0.67	0.992 ± 0.016	√	√
	CBP	3.76 ± 0.06	2.31 ± 0.05	0.948 ± 0.014	1.48 ± 0.03	4.82 ± 0.56	0.992 ± 0.013	√	√
	RepCon	**3.62** ± 0.06	**2.12** ± 0.04	**0.953** ± 0.016	**1.41** ± 0.03	**4.40** ± 0.63	**0.993** ± 0.014	√	

The best-performing results are shown in bold. The column "Inference" specifies the required input for each model during the inference phase with the corresponding feature extraction modules. All the results presented are averages of ten outcomes obtained from ten random seeds. AP: Aggregation propensity; RT: retention time; MAE: mean absolute error; MSE: mean squared error; R^2: the coefficient of determination; RNN: recurrent neural networks; LSTM: long short-term memory; GCN: graph convolutional networks; GAT: graph attention networks; GraphSAGE: graph sampling and aggregation networks; WS: weighted sum; CA: cross-attention; CBP: compact bilinear pooling; RepCon: representation contrasting.

when the R^2 value shows only a marginal improvement. Similarly, for the RT dataset, RepCon's MAE of 1.41 and MSE of 4.40 represent substantial reductions in prediction error compared to the Transformer (MAE = 1.57, MSE = 5.02) and GraphSAGE (MAE = 2.57, MSE = 12.9), underscoring the advantage of integrating sequence and graph representations through contrastive learning.

Based on these results, it is evident that the co-modeling frameworks tend to outperform models that rely solely on sequence or graph data. This suggests that the combination of sequence and chemical information can improve the predictive accuracy of the models. Among the co-modeling frameworks evaluated, RepCon not only exhibits superior performance but also demonstrates greater efficiency in the inference process. By leveraging the complementary strengths of both representations, our approach not only significantly improves predictive performance but also offers richer insights into peptide structure-function relationships. This integration is crucial for advancing our understanding of peptide mechanisms and accelerating the development of peptide-based functional materials and drugs.

Evaluation on regression tasks of co-modeling framework

The boxplots presented in Figure 4 provide a comparative analysis of the classification accuracy of five co-modeling implementations (WS, Concat, CBP, CA, RepCon) and two non-co-modeling baselines (T: Transformer, G: GraphSAGE) on the AMP and PepDB datasets.

The co-modeling method utilizing the RepCon framework shows the highest accuracy among the evaluated methods. The median accuracy of RepCon exceeds that of the other methods, indicating reliable and robust performance. The interquartile range (IQR), represented by the height of the box, and the whiskers of RepCon, while not the smallest, are comparatively modest, suggesting a moderate level of variability in the accuracy of RepCon.

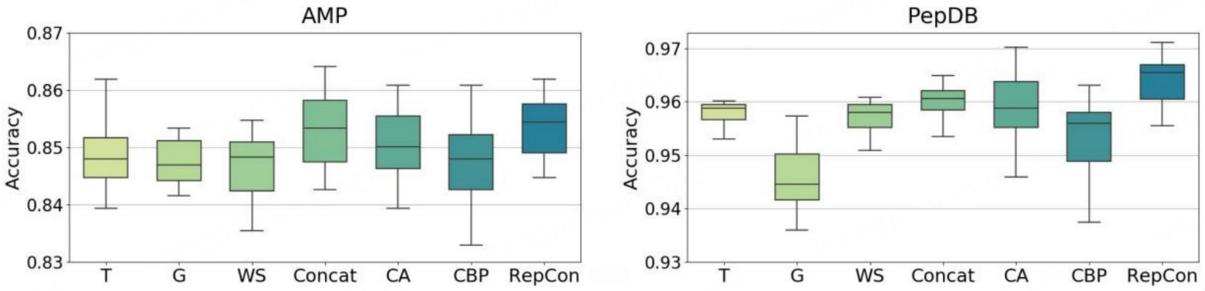

Figure 4. Comparison of classification accuracy between two non-co-modeling baselines (T: Transformer; G: GraphSAGE) and co-modeling implementations (WS: weighted sum; Concat: concatenation; CA: cross-attention; CBP: compact bilinear pooling; RepCon: representation contrasting) on the datasets AMP and PepDB. Each box represents the distribution of accuracy for a method, including the minimum, first quartile (bottom of the box), median (line within the box), third quartile (top of the box), and maximum. GraphSAGE: Graph sampling and aggregation networks; AMP: anti-microbial peptide; PepDB: Peptide DataBase.

The Concat method ranks second in performance after RepCon, with a median accuracy lower than RepCon, but higher than the other methods. In the AMP dataset, the IQR and whiskers of Concat are slightly larger than those of RepCon, but smaller in the PepDB dataset, suggesting a comparable level of performance variability. The median accuracies of the other co-modeling methods - WS, CBP, and CA - are all below that of Concat. Notably, the median accuracy of CBP is close to the non-co-modeling baselines, and it has the largest IQR and whiskers, indicating the highest level of variability and thus the least reliability among the co-modeling methods. The accuracy and reliability of WS and CA also do not significantly exceed those of the baseline models.

In summary, the co-modeling approaches generally demonstrate superior performance compared to the non-co-modeling baselines. In particular, models based on contrastive learning and Concat outperform the other co-modeling frameworks. Notably, even though RepCon does not require the integration of a GNN during the inference phase, it maintains a performance advantage over Concat. This suggests that RepCon's efficiency and effectiveness in feature extraction is competitive without the additional computational overhead associated with graph-based components.

Attribution study: attribution shifts in contrastive learning-based co-modeling

Table 3 provides a comparison of the amino acid level attribution value similarities between non-co-modeling baselines and the RepCon model in the context of AP prediction. The models being compared include the Transformer (T), a Transformer with different initialization (T'), GraphSAGE (G), and RepCon (R). We use four metrics to evaluate these similarities: Kendall's Tau, Spearman's Footrule, Jensen-Shannon Divergence (JS Divergence), and Cosine Similarity. The notation "↑" indicates that higher values imply a stronger correlation, while "↓" indicates that higher values imply a weaker correlation.

When comparing the Transformer model with different initializations (T & T'), the metrics reveal a high degree of similarity. Kendall's Tau and Spearman's Footrule, which evaluate the correlation between the rankings, yield values of 0.860 (± 0.142) and 0.945 (± 0.107), respectively. These high scores indicate a robust correlation in the rankings between the two models with different initializations. The JS Divergence records a low value of 0.099 (± 0.043), indicating a high degree of similarity between the distributions of the two models. Finally, the Cosine Similarity, another measure of similarity between value distributions, presents an extremely high value of 0.992 (± 0.010), further confirming the high similarity between the two Transformer models.

Table 3. Similarities of amino acid level attributions on AP prediction between non-co-modeling baselines and the RepCon model

Metric	T *vs.* T′	T *vs.* R	G *vs.* T	G *vs.* R
Kendall's Tau↑	0.860 ± 0.142	0.795 ± 0.154	0.165 ± 0.301	0.183 ± 0.313
Spearman's Footrule↑	0.945 ± 0.107	0.886 ± 0.115	0.227 ± 0.379	0.258 ± 0.340
JS Divergence↓	0.099 ± 0.043	0.119 ± 0.049	0.366 ± 0.107	0.329 ± 0.101
Cosine Similarity↑	0.992 ± 0.010	0.984 ± 0.014	0.782 ± 0.140	0.817 ± 0.125

T: Transformer, T′: Transformer with a different initialization, G: GraphSAGE, and R: RepCon. "A *vs.* B" represents a comparison of the attribution similarities between model A and model B. The results are presented as mean ± standard deviation. AP: Aggregation propensity; GraphSAGE: graph sampling and aggregation networks; RepCon: representation contrasting.

Comparing the T and R models, the correlation metrics show a moderate decrease compared with those of transformer models with different initializations (T *vs.* T'), with Kendall's Tau at 0.795 (± 0.154) and Spearman's Footrule at 0.886 (± 0.115). The JS Divergence increases to 0.119 (± 0.049), and the Cosine Similarity decreases to 0.984 (± 0.014), reflecting a slightly lower similarity between the Transformer and RepCon models compared to the comparisons between the Transformer models with different initializations. These results suggest that although the Transformer and RepCon models exhibit a high degree of correlation, this correlation is somewhat less pronounced than that observed between the two Transformer models, indicating a distinction in their attribution patterns.

When comparing GraphSAGE with both the Transformer (G *vs.* T) and RepCon (G *vs.* R), the correlation metrics are notably lower than those observed when comparing Transformer models with different initializations (T *vs.* T') and Transformer and RepCon models (T *vs.* R). This indicates a weaker correlation between models that use different representations and architectures for the input peptides. However, the similarities are slightly higher in the comparison between GraphSAGE and RepCon (G *vs.* R) than in the comparison between GraphSAGE and the Transformer (G *vs.* T), suggesting that the GraphSAGE's predictions are closer to those of RepCon than to those of the Transformer model.

In summary, the RepCon's model's explanation differs from that of the Transformer model despite using the same architecture during inference, as is also evidenced by that the GraphSAGE's prediction explanation is closer to RepCon than that of the Transformer model, indicating a shift caused by integrating the chemical information in RepCon.

Comparisons with previous state-of-the-art

This study significantly advances peptide prediction models by proposing a co-modeling framework that integrates both sequence and chemical representations, outperforming previous approaches. Compared to the work by Wang *et al.*, which demonstrated the potential of deep learning for predicting self-assembling peptides but relied solely on sequence-based models, our co-modeling framework incorporates both sequence and graph-based encoders[34]. This dual-modality approach achieves superior performance, as evidenced by lower prediction errors (e.g., MAE of 3.62E-2 and MSE of 2.12E-3 on the AP dataset) compared to sequence-only models such as the Transformer (MAE of 3.81E-2; MSE of 2.33E-3). Similarly, Liu *et al.* explored sequential and graphical encoding but did not integrate them[35]. Our contrastive learning-based co-modeling framework (RepCon) not only enhances prediction accuracy but also improves efficiency by leveraging complementary strengths of both representations. For instance, on the RT dataset, RepCon achieved an MAE of 1.41 and MSE of 4.40, outperforming both sequence-based Transformer and graph-based GraphSAGE models. This demonstrates the ability of the co-modeling framework to capture more comprehensive features from peptide data, leading to enhanced model performance.

In terms of model interpretability, our co-modeling framework addresses limitations observed in previous studies. Batra *et al.* highlighted the importance of overcoming human biases in peptide discovery but focused primarily on the discovery aspect without detailed attribution analysis[11]. Our study provides a robust attribution methodology, revealing discrepancies between sequence-based and graph-based models and demonstrating how the co-modeling framework can reconcile these differences. The attribution results show that graph-based models' explanations are closer to RepCon than sequence-only models, indicating a shift in feature importance due to the integration of chemical information. This detailed analysis offers deeper insights into peptide behavior and enhances the reliability of predictions, setting a new benchmark for model performance and interpretability in peptide research.

CONCLUSIONS

In this research, we investigate the impact of two peptide representations (i.e., sequence and graph) on the interpretability of predictive models. We have developed and validated gradient-based feature attribution techniques for both sequence-based and molecular graph-based models, revealing substantial differences between these model types. Our results have led to the proposal of a co-modeling framework that fuses information from both representations, offering a comprehensive analysis of peptides. The co-modeling framework not only improves prediction accuracy but also enhances the interpretability of models through attribution analysis. Future work could explore the application of this framework to other peptide properties and biomolecules, expanding its potential impact on drug discovery and protein engineering. By building on previous studies and addressing their limitations, this work sets a new standard for integrating diverse data types in AI models, paving the way for more effective and comprehensive peptide research, which could enhance efficiency in subsequent applications of peptide drug design and functional materials development.

DECLARATIONS

Acknowledgments

We acknowledge the high-performance computing platform at Xi'an Jiaotong-Liverpool University.

Authors' contributions

Conceptualization, methodology, software, data curation, visualization, writing-original draft preparation: Liu, Z.; Yan, M.; Zhu, Z.

Conceptualization, writing and revision, supervision, project administration, funding acquisition: Wang, J.

High-performance computing support, data processing and analysis: Guo, Y.; Xu, M.

Availability of data and materials

The data and code used in this study are publicly available to facilitate verification and replication of the results. The code used in this research can be accessed via the following link: https://github.com/Zihan-Liu-00/RepCon.

Financial support and sponsorship

This work was supported by the National Natural Science Foundation of China (No. 52101023) and Fundamental Research Plan (Natural Science Foundation) - General Programme of Jiangsu Provincial Department of Science and Technology (No. BK20241816).

Conflicts of interest

All authors declared that there are no conflicts of interest.

Ethical approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Copyright

© The Author(s) 2025.

REFERENCES

1. Langel, U.; Cravatt, B. F.; Graslund, A.; et al. Introduction to peptides and proteins. 1st Edition. CRC press: 2009. DOI
2. Muttenthaler, M.; King, G. F.; Adams, D. J.; Alewood, P. F. Trends in peptide drug discovery. *Nat. Rev. Drug. Discov.* **2021**, *20*, 309-25. DOI PubMed
3. Bhinder, B.; Gilvary, C.; Madhukar, N. S.; Elemento, O. Artificial intelligence in cancer research and precision medicine. *Cancer. Discov.* **2021**, *11*, 900-15. DOI PubMed PMC
4. Mohapatra, S.; Hartrampf, N.; Poskus, M.; Loas, A.; Gómez-Bombarelli, R.; Pentelute, B. L. Deep learning for prediction and optimization of fast-flow peptide synthesis. *ACS. Cent. Sci.* **2020**, *6*, 2277-86. DOI PubMed PMC
5. Tao, K.; Makam, P.; Aizen, R.; Gazit, E. Self-assembling peptide semiconductors. *Science* **2017**, *358*, eaam9756. DOI PubMed PMC
6. Kim, S. H.; Parquette, J. R. A model for the controlled assembly of semiconductor peptides. *Nanoscale* **2012**, *4*, 6940-7. DOI
7. Yang, Y.; Wang, X.; Wu, X.; et al. Computation-driven rational design of self-assembled short peptides for catalytic hydrogen production. *J. Am. Chem. Soc.* **2024**, *146*, 13488-98. DOI
8. Stone, E. A.; Hosseinzadeh, P.; Craven, T. W.; et al. Isolating conformers to assess dynamics of peptidic catalysts using computationally designed macrocyclic peptides. *ACS. Catal.* **2021**, *11*, 4395-400. DOI PubMed PMC
9. McDonald, E. F.; Jones, T.; Plate, L.; Meiler, J.; Gulsevin, A. Benchmarking AlphaFold2 on peptide structure prediction. *Structure* **2023**, *31*, 111-9.e2. DOI PubMed PMC
10. Lei, Y.; Li, S.; Liu, Z.; et al. A deep-learning framework for multi-level peptide-protein interaction prediction. *Nat. Commun.* **2021**, *12*, 5465. DOI PubMed PMC
11. Batra, R.; Loeffler, T. D.; Chan, H.; et al. Machine learning overcomes human bias in the discovery of self-assembling peptides. *Nat. Chem.* **2022**, *14*, 1427-35. DOI PubMed PMC
12. Bhadra, P.; Yan, J.; Li, J.; Fong, S.; Siu, S. W. I. AmPEP: sequence-based prediction of antimicrobial peptides using distribution patterns of amino acid properties and random forest. *Sci. Rep.* **2018**, *8*, 1697. DOI PubMed PMC
13. Veltri, D.; Kamath, U.; Shehu, A. Deep learning improves antimicrobial peptide recognition. *Bioinformatics* **2018**, *34*, 2740-7. DOI PubMed PMC
14. Hellinger, R.; Sigurdsson, A.; Wu, W.; et al. Peptidomics. *Nat. Rev. Methods. Primers.* **2023**, *3*, 25. DOI PubMed PMC
15. Seebach, D.; Hook, D. F.; Glättli, A. Helices and other secondary structures of beta- and gamma-peptides. *Biopolymers* **2006**, *84*, 23-37. DOI PubMed
16. Mittal, J.; Yoo, T. H.; Georgiou, G.; Truskett, T. M. Structural ensemble of an intrinsically disordered polypeptide. *J. Phys. Chem. B.* **2013**, *117*, 118-24. DOI PubMed
17. Hearst, M.; Dumais, S.; Osuna, E.; Platt, J.; Scholkopf, B. Support vector machines. *IEEE. Intell. Syst. Their. Appl.* **1998**, *13*, 18-28. DOI
18. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5-32. DOI
19. Murtagh, F. Multilayer perceptrons for classification and regression. *Neurocomputing* **1991**, *2*, 183-97. DOI
20. Almagro, A. J. J.; Salvatore, M.; Emanuelsson, O.; et al. Detecting sequence signals in targeting peptides using deep learning. *Life. Sci. Alliance.* **2019**, *2*, e201900429. DOI PubMed PMC
21. Medsker, L.; Jain, L. C. Recurrent neural networks: design and applications. 1st Edition. CRC Press: 1999. DOI
22. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural. Comput.* **1997**, *9*, 1735-80. DOI PubMed
23. Vaswani, A.; Shazeer, N.; Parmar, N. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762. Available online: https://arxiv.org/abs/1706.03762. (accessed 21 Feb 2025)
24. Charoenkwan, P.; Nantasenamat, C.; Hasan, M. M.; Manavalan, B.; Shoombuatong, W. BERT4Bitter: a bidirectional encoder representations from transformers (BERT)-based model for improving the prediction of bitter peptides. *Bioinformatics* **2021**, *37*, 2556-62. DOI PubMed
25. Chu, Y.; Zhang, Y.; Wang, Q.; et al. A transformer-based model to predict peptide–HLA class I binding and optimize mutated peptides for vaccine design. *Nat. Mach. Intell.* **2022**, *4*, 300-11. DOI
26. Wang, J.; Li, C.; Shin, S.; Qi, H. Accelerated atomic data production in *ab initio* molecular dynamics with recurrent neural network for materials research. *J. Phys. Chem. C.* **2020**, *124*, 14838-46. DOI
27. Bronstein, M. M.; Bruna, J.; Lecun, Y.; Szlam, A.; Vandergheynst, P. Geometric deep learning: going beyond euclidean data. *IEEE. Signal. Process. Mag.* **2017**, *34*, 18-42. DOI

28. Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Yu, P. S. A comprehensive survey on graph neural networks. *IEEE. Trans. Neural. Netw. Learn. Syst.* **2021**, *32*, 4-24. DOI
29. Yan, K.; Lv, H.; Guo, Y.; Peng, W.; Liu, B. sAMPpred-GAT: prediction of antimicrobial peptide by graph attention network and predicted peptide structure. *Bioinformatics* **2023**, *39*, btac715. DOI PubMed PMC
30. Wei, L.; Ye, X.; Xue, Y.; Sakurai, T.; Wei, L. ATSE: a peptide toxicity predictor by exploiting structural and evolutionary information based on graph neural network and attention mechanism. *Brief. Bioinform.* **2021**, *22*, bbab041. DOI
31. Boadu, F.; Cao, H.; Cheng, J. Combining protein sequences and structures with transformers and equivariant graph neural networks to predict protein function. *Bioinformatics* **2023**, *39*, i318-25. DOI PubMed PMC
32. Zhao, A.; Chen, Z.; Fang, Z.; Zhang, X.; Li, J. Dual-modality representation learning for molecular property prediction. *arXiv* **2025**, arXiv:2501.06608. Available online: https://doi.org/10.48550/arXiv.2501.06608. (accessed 21 Feb 2025)
33. McCloskey, K.; Taly, A.; Monti, F.; Brenner, M. P.; Colwell, L. J. Using attribution to decode binding mechanism in neural network models for chemistry. *Proc. Natl. Acad. Sci. U. S. A.* **2019**, *116*, 11624-9. DOI PubMed PMC
34. Wang, J.; Liu, Z.; Zhao, S.; et al. Deep learning empowers the discovery of self-assembling peptides with over 10 trillion sequences. *Adv. Sci.* **2023**, *10*, e2301544. DOI PubMed PMC
35. Liu, Z.; Wang, J.; Luo, Y.; Zhao, S.; Li, W.; Li, S. Z. Efficient prediction of peptide self-assembly through sequential and graphical encoding. *Brief. Bioinform.* **2023**, *24*, bbad409. DOI
36. Xu, T.; Wang, J.; Zhao, S.; et al. Accelerating the prediction and discovery of peptide hydrogels with human-in-the-loop. *Nat. Commun.* **2023**, *14*, 3880. DOI PubMed PMC
37. Wang, J.; Liu, Z.; Zhao, S.; et al. Aggregation Rules of Short Peptides. *JACS. Au.* **2024**, *4*, 3567-80. DOI PubMed PMC
38. Marler, R. T.; Arora, J. S. The weighted sum method for multi-objective optimization: new insights. *Struct. Multidisc. Optim.* **2010**, *41*, 853-62. DOI
39. Shang, W.; Sohn, K.; Almeida, D.; Lee, H. Understanding and improving convolutional neural networks via concatenated rectified linear units. *arXiv* **2016**, arXiv:1603.05201. Available online: https://doi.org/10.48550/arXiv.1603.05201. (accessed 21 Feb 2025)
40. Gao, Y.; Beijbom, O.; Zhang, N.; Darrell, T. Compact bilinear pooling. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, USA, Jun 27-30, 2016; IEEE, 2016; pp. 317-26. DOI
41. Marrink, S. J.; Risselada, H. J.; Yefimov, S.; Tieleman, D. P.; de, V. A. H. The MARTINI force field: coarse grained model for biomolecular simulations. *J. Phys. Chem. B.* **2007**, *111*, 7812-24. DOI PubMed
42. Monticelli, L.; Kandasamy, S. K.; Periole, X.; Larson, R. G.; Tieleman, D. P.; Marrink, S. J. The MARTINI coarse-grained force field: extension to proteins. *J. Chem. Theory. Comput.* **2008**, *4*, 819-34. DOI PubMed
43. Brooks, B. R.; Brooks, C. L.; Mackerell, A. D. J.; et al CHARMM: the biomolecular simulation program. *J. Comput. Chem.* **2009**, *30*, 1545-614. DOI PubMed PMC
44. Brooks, B. R.; Bruccoleri, R. E.; Olafson, B. D.; States, D. J.; Swaminathan, S.; Karplus, M. CHARMM: a program for macromolecular energy, minimization, and dynamics calculations. *J. Comput. Chem.* **1983**, *4*, 187-217. DOI
45. Sundararajan, M.; Taly, A.; Yan, Q. Axiomatic attribution for deep networks. *arXiv* **2017**, arXiv:1703.01365. Available online: https://doi.org/10.48550/arXiv.1703.01365. (accessed 21 Feb 2025)
46. Meier, F.; Geyer, P. E.; Virreira, W. S.; Cox, J.; Mann, M. BoxCar acquisition method enables single-shot proteomics at a depth of 10,000 proteins in 100 minutes. *Nat. Methods.* **2018**, *15*, 440-8. DOI
47. Liu, F.; Baggerman, G.; Schoofs, L.; Wets, G. The construction of a bioactive peptide database in Metazoa. *J. Proteome. Res.* **2008**, *7*, 4119-31. DOI PubMed
48. Jiang, W. Graph-based deep learning for communication networks: a survey. *Comput. Commun.* **2022**, *185*, 40-54. DOI
49. Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; Bengio, Y. Graph attention networks. *arXiv* **2017**, arXiv:1710.10903. Available online: https://doi.org/10.48550/arXiv.1710.10903. (accessed 21 Feb 2025)
50. Hamilton, W. L.; Ying, R.; Leskovec, J. Inductive representation learning on large graphs. *arXiv* 2017, arXiv:1706.02216. Available online: https://doi.org/10.48550/arXiv.1706.02216. (accessed 21 Feb 2025)
51. Frederix, P. W. J. M.; Scott, G. G.; Abul-Haija, Y. M.; et al. Exploring the sequence space for (tri-)peptide self-assembly to design and discover new hydrogels. *Nat. Chem.* **2015**, *7*, 30-7. DOI