Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue

Cite this: *Digital Discovery*, 2025, **4**, 2478

PepMSND: integrating multi-level feature engineering and comprehensive databases to enhance *in vitro*/*in vivo* peptide blood stability prediction†

Haomeng Hu,‡[a] Chengyun Zhang,‡[b] Zhenyu Xu,‡[b] Jingjing Guo,[c] An Su,[a] Chengxi Li[d] and Hongliang Duan*[c]

Received 24th March 2025
Accepted 16th July 2025

DOI: 10.1039/d5dd00118h

rsc.li/digitaldiscovery

Deep learning has emerged as a transformative tool for peptide drug discovery, yet predicting peptide blood stability—a critical determinant of bioavailability and therapeutic efficacy—remains a major challenge. While such a task can be accomplished through experiments, it requires much time and cost. Here, to address this challenge, we collect extensive experimental data on peptide stability in blood from public databases and the literature and construct a database of peptide blood stability that includes 635 samples. Based on this database, we develop a novel model called PepMSND, integrating KAN, Transformer, GAT, and SE(3)-Transformer to perform multi-level feature engineering for peptide blood stability prediction. Our model can achieve an ACC of 0.867 and an AUC of 0.912 on average and outperforms the baseline models. We also develop a user-friendly web interface for the PepMSND model, which is freely available at http://model.highslab.com/pepmsnd. This research is crucial for the development of novel peptides with strong blood stability, as the stability of peptide drugs directly determines their effectiveness and reliability in clinical applications.

Introduction

Peptides and proteins have gradually become a popular modality in the pharmaceutical industry. To date, over 120 peptide-based drugs have received regulatory approval worldwide, playing a crucial role in the treatment of cancers, metabolic disorders, cardiovascular diseases, and autoimmune conditions.[1] However, despite their growing clinical success, the broader application of peptide therapeutics is still limited. The limitations of peptide-based drugs can be attributed to multiple factors, among which instability remains a major hurdle.[2] Peptides are highly susceptible to enzymatic hydrolysis by proteases in the body, including those found in the plasma, gastrointestinal tract, liver, and immune cells. This often results in a very short half-life, severely limiting their oral bioavailability and overall therapeutic efficacy.[3–5] In addition to instability, challenges such as potential toxicity, high manufacturing costs, and reliance on intravenous administration further hinder their broader clinical translation.[1] To improve their stability, many modification strategies have been proposed: D-form or unnatural amino acid residues, *N*-methylation or formation of cyclic peptides, and conjugation with macromolecular carriers such as proteins, lipids, and polymers.[6–8]

Considering the importance of the blood stability of peptides in their clinical application, how to measure/predict this property becomes an appealing issue.[9] Traditionally, experimental methods such as blood stability and enzyme degradation tests have been universal methods. These methods allow for more accurate identification and assessment of the blood stability of peptides in different experimental settings. However, they necessitate high costs and a long time, which cannot satisfy the recommendation for high-throughput screening or large-scale studies.[10] To address such a problem, people turn to computational methods, which have attracted much attention in other fields. Take ProtParam as an example, this technique explores half-life and N-terminal residues based on the N-end rule[11–14] and combines with the experimental statistics-based rule[15] to measure the peptide stability. Additionally, a multi-variable regression model is also implemented to predict the half-life of peptides in blood.[16] With the advancement of deep learning in peptide development, Mathur *et al.*[17] predicted this property of

[a]*College of Pharmaceutical Sciences, Zhejiang University of Technology, Hangzhou, 310014, China*

[b]*AI Department, Shenzhen Highslab Therapeutics. Inc, Shenzhen, 518000, China*

[c]*Faculty of Applied Sciences, Macao Polytechnic University, Macao, 999078, China. E-mail: hduan@mpu.edu.mo*

[d]*College of Chemical and Biological Engineering, Zhejiang University, Hangzhou, 310027, China*

† Electronic supplementary information (ESI) available. See DOI: https://doi.org/10.1039/d5dd00118h

‡ These authors contributed equally to this work.

peptides by using an SVM model that was trained on a database consisting of the half-life of 261 peptides in mammalian blood.

Nevertheless, peptide blood stability prediction still faces essential challenges despite related developments in the past few decades. For example, in the blood stability test, the SUPR peptide showed a very different half-life under different experimental conditions: this peptide is susceptible to hydrolysis in mouse plasma, whereas more than 50% of the peptide remains unhydrolyzed in human plasma within 24 hours. However, such peptide blood stability differences are usually neglected due to the individual and fragmented data availability, which may lead to the model's misclassifications. Additionally, many methods prefer to adopt relatively simple low-dimensional representations to illustrate peptide features, which usually neglect their conformations, which are vital for distinguishing the stability difference.[18,19] Actually, linear and cyclic peptides share the same amino acid sequences but have entirely different blood stability.[20] To address the problem above, comprehensive systematized experimental data on peptide blood stability are necessary, which can accelerate the development of related research. Therefore, in this study, we collect experimental data from public databases and the literature as much as possible to build a specific peptide blood stability database. Furthermore, we perform comprehensive peptide feature engineering including basic molecular descriptors, SMILES, molecular structures and complex 3D conformations to illustrate the intrinsic characters of peptides, thereby developing a novel model called PepMSND tailored for predicting peptide stability in various blood environments (see Fig. 1). The combination of our database and multimodal model offers an opportunity to identify potential peptide candidates with strong blood stability, improving the peptide drug development. To facilitate the accessibility of PepMSND for a broader audience, particularly researchers without a deep learning background, we integrate it into a server environment and developed an intuitive online web service platform: **http://model.highslab.com/pepmsnd**.

Experimental

Data collection

In this study, as shown in Fig. 1A(a), we collect peptide blood stability data samples manually from various sources such as published studies, patents, and related databases. Based on the universal claim of Cavaco *et al.* that the experimental half-life value is a good choice to demonstrate the stability of peptides,[21] we adopt peptide [Title/Abstract] AND half-life[Title/Abstract] as the keyword to search for associated information in PubMed and find 1413 studies published in the range 2015–2024 year. In addition, we search public databases such as PEPlife,[22] DrugBank[23] and THPdb[24] to explore more data. To ensure the quality and quantity of data, we perform the following data cleaning: (1) removing peptides that lack or are missing stability information; (2) removing peptides for which no explicit sequence information is given; (3) excluding peptides for which experimental conditions are not explicitly given; (4) ignoring peptides that are not experimented on in human or murine blood; (5) excluding peptides with complex modifications (*e.g.*, polyethylene glycol modifications) because they are difficult to convert accurately to standard SMILES format. Finally, a total of 635 samples are collected.

Since the FASTA format does not allow for a perfect and accurate representation of unnatural and modified residues, SMILES was used in this study to characterize the dataset in one dimension. As shown in Fig. 1A(b), for standard sequences, an automated conversion tool was developed to generate SMILES representations. However, for particularly complex or non-standard structures, manual drawing was performed using ChemDraw. All SMILES representations were subsequently standardized using RDKit.

Peptide structure generation

Traditionally, experimental methods such as X-ray crystal diffraction, nuclear magnetic resonance spectroscopy, and electron microscopy are effective ways to investigate peptide structures.[25] With the advancement of technology, structure prediction models such as Alphafold,[26] RoseTTAFold,[27] ESM-Fold[28] and HighFold[29] can also provide plausible structures with high accuracy and efficiency. As displayed in Fig. 1A(c), first, we search the PDB database, and for peptides that could not be retrieved, we adopt different strategies to predict their structure. For the natural linear peptides, AlphaFold2 is implemented.[30] For the natural cyclic peptides, we use our proposed model, HighFold. For the peptides with complex modifications, RDKit (version 2023.3.2) is used. Based on this toolkit, 5000 conformations are generated for the peptide input and are optimized by the UFF force field. Ultimately, only the conformations with the lowest energy are selected for further experiments (detailed information can be found in the ESI Section 7†). Given the presence of both linear and cyclic peptide structures in these modified peptides, we employed RDKit's ETKDGv3 algorithm[31] for 3D structure generation. ETKDGv3 extends the applicability of previous ETKDG versions and demonstrates reliable performance across small molecules, linear peptides, and cyclic peptides.

The PepMSND model

Multimodal technology is a method that can efficiently integrate and process various data. It not only enhances the depth and breadth of data processing but also significantly improves the accuracy and generality of the model.[32] In this study, we apply this technology to perform comprehensive feature engineering that takes physicochemical properties, sequence information, molecular structure, and 3D conformation into consideration (Fig. 1B). Specifically, (1) molecular descriptors input as the 0D feature are processed by the Kolmogorov–Arnold Network (KAN)[33] to capture the physical and chemical properties of peptides; (2) SMILES input as the 1D feature is processed by the Transformer[34] to absorb the sequence relationship; (3) molecular structure as the 2D feature is processed by Graph Attention Networks (GAT)[35] to learn the interaction of atoms and bonds; (4) predicted 3D structure as the 3D feature is processed by the SE(3)-Transformer[36] to provide additional information. Subsequently, a series of learnable weights is

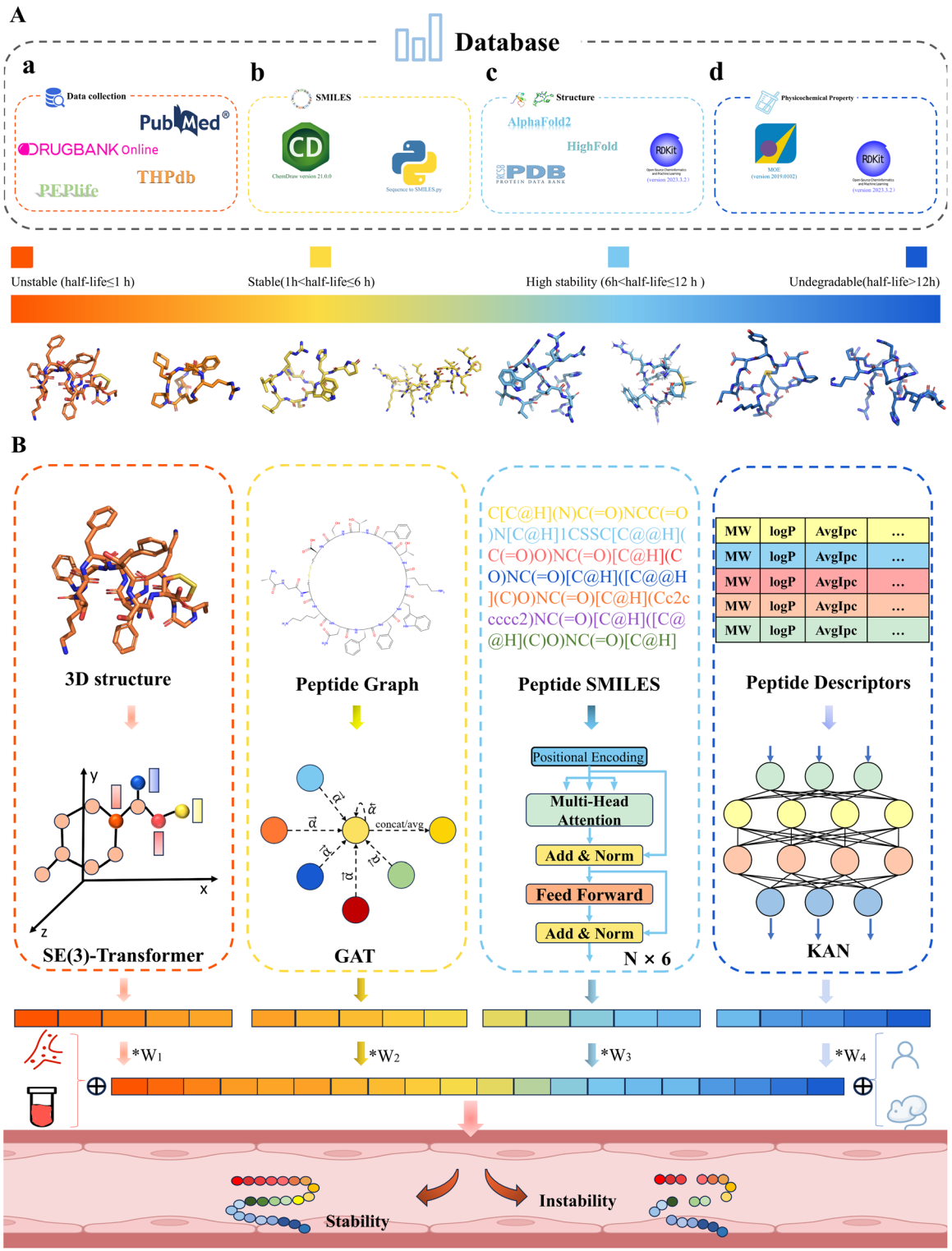

Fig. 1 The workflow of our database and PepMSND model. (A) The illustration of the database. (B) The architecture of the Pep-MSND model. This model includes four modules: the SE(3)-Transformer for the 3D peptide structure feature, the GAT model for the 2D peptide molecular structure, the Transformer for 1D peptide SMILES, and the KAN model for 0D peptide physicochemical properties.

employed to integrate these features, generating a joint feature vector. This vector is then fed into a shared layer for further dimensionality reduction. Notably, within this process, we explicitly encode the experimental conditions—including testing species and *in vivo*/*in vitro* environment—using a two-dimensional binary vector, which is then concatenated with the output of the main representation layers immediately before the final prediction layer. For example, [1, 0] denotes an *in vitro*

measurement in human blood, while [0, 0] denotes an *in vivo* measurement in human blood. This design ensures that critical contextual information is preserved with high representational weight and not diluted during earlier stages of representation learning. By incorporating these features near the output layer, the model can directly utilize them in its decision-making process. Further details regarding this encoding scheme and its implementation are provided in the following sections.

KAN model

In this study, we incorporate the Kolmogorov–Arnold Network (KAN), a recently proposed alternative to traditional Multi-Layer Perceptrons (MLPs) that has shown promising performance in modeling complex nonlinear relationships while maintaining parameter efficiency. Unlike conventional neural networks that use fixed activation functions on neurons, KAN replaces linear weights with learnable univariate functions parametrized *via* splines, enabling greater flexibility and adaptability in function approximation.

Recent studies have demonstrated KAN's superior performance across various domains. For example, Vaca-Rubio *et al.*[37] applied KAN to satellite traffic forecasting tasks and achieved higher accuracy than traditional MLPs. Abdulkadir *et al.*[38] utilized a quasi-Newton optimized KAN to predict wind power output and observed substantial error reduction. In the field of genomics, Cherednichenko and Poptsova[39] successfully integrated Linear KAN (LKAN) and Convolutional KAN (CKAN) layers into DNA sequence classification networks, outperforming traditional fully connected and CNN architectures on multiple benchmark datasets.

In this work, we apply the KAN model to learn representations of peptide physicochemical properties encoded by molecular descriptors. The use of KAN facilitates flexible functional mappings, which are particularly important given the heterogeneous and nonlinear nature of such molecular features. Specifically, we use RDKit (version 2023.3.2) to calculate the associated property values and remove the descriptors, including constant values, to avoid information redundancy. Then, these selected descriptors are evaluated using a random forest model, and the top 140 important features are retained as the input for the KAN model (detailed information can be found in the ESI Section 1†). Additionally, we introduce two features: species and experimental environment. By utilizing one-hot encoding, these features are incorporated as additional 0D features into the set of molecular descriptors, thereby enriching the input information for the model. The function of this technology is as follows:

$$f(x) = \sum_{q=1}^{2n+1} \Phi_q \left(\sum_{p=1}^{n} \phi_{q,p}(x_p) \right)$$

Transformer model

Compared to the SMILES representations of typical small-molecule compounds commonly found in molecular property prediction datasets, peptide SMILES are not only significantly longer in sequence length but also more structurally and syntactically complex. Peptides often contain cyclic structures, disulfide bonds, and extensive stereochemical information, leading to richer and more intricate SMILES encodings. To effectively capture the intrinsic correlation with such a sequence, we adopt the Transformer model. This model consists of three key layers: an embedding layer, attention layers and a feed-forward neural network (FNN). In the embedding layer, a positional encoder is introduced for precisely fusing the position information of each character in its sequence into the corresponding embedding vector. The function of this technology is as follows:

$$\mathrm{PE}_{(\mathrm{pos},2i)} = \sin(\mathrm{pos}/10\,000^{2i/d_{\mathrm{model}}})$$

$$\mathrm{PE}_{(\mathrm{pos},2i+1)} = \cos(\mathrm{pos}/10\,000^{2i/d_{\mathrm{model}}})$$

where pos is the position, i is the embedding vector dimension and d_{model} is the model dimension.

The attention mechanism is the core of the Transformer model. It can focus on the significant information among a vast amount of data and mitigate the impact of redundant information. Its calculation is as follows:

$$\mathrm{Attention}\ (Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

where Q is the query, K is the key, and V is the value; all are calculated based on inputted sequences.

GAT model

Graph Attention Networks (GATs) are neural networks specifically designed to handle graph data. During the message-passing process. GATs leverage the attention mechanism to dynamically learn the importance of neighbor nodes for each node. This allows the network to pay attention to the most relevant information while ignoring less important or redundant information. In our task, the atoms are represented by nodes, and bonds are represented by edges. With the implementation of GAT, the 2D feature can be absorbed. With its inherent ability to integrate graph structure information, GAT can comprehensively and deeply analyze the interrelationships and patterns among the atoms in the polypeptide. This feature enables us to extract meaningful features that provide strong support for subsequent classification tasks. Its calculation is shown below:

$$e_{ij} = a\left(\mathbf{W}\overrightarrow{h_i},\ \mathbf{W}\overrightarrow{h_j}\right)$$

$$\alpha_{ij} = \mathrm{softmax}_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k\in N_i}\exp(e_{ik})}$$

where $\mathbf{W}$ is a weight matrix. $\mathbf{W}\overrightarrow{h_i}$ and $\mathbf{W}\overrightarrow{h_j}$ are the feature vectors of nodes i and j after linear translation. N_i is the neighbourhood set of node i, and α_{ij} is the attention coefficient normalized by the softmax function.

SE(3)-Transformer model

The SE(3)-Transformer is a self-attention module variant for 3D point clouds and graphs. Under continuous 3D roto-translations, this model can remain equivariant. In the model, each atom is represented by its 3D coordinates (x, y, z). This model consists of three components: (1) edge-wise attention weight α_{ij}, which remains unchanged under SE(3) on each edge; (2) edge-wise SE(3)-equivariant value messages, which propagate information between nodes; and (3) a linear/attentive self-interaction layer. The attention for each structural node is calculated as follows:

$$f_{\mathrm{out},i}^{\ell} = \underbrace{W_V^{\ell\ell} f_{\mathrm{in},i}^{\ell}}_{(3)\ \text{self-interaction}} + \sum_{k \geq 0} \sum_{j \in N_i \setminus \{i\}} \underbrace{\alpha_{ij}}_{(1)\ \text{attention}} \underbrace{W_V^{\ell k}(x_j - x_i) f_{\mathrm{in},j}^{(k)}}_{(2)\ \text{value message}}$$

where $W_V^{\ell\ell} f_{\mathrm{in},i}^{\ell}$ is the self-interaction term, showing the result of the input feature $f_{\mathrm{in},i}^{\ell}$ of node i after undergoing a linear transformation. α_{ij} is the attention weight, representing the degree to which node i pays attention to or attends to its neighboring node j. $W_V^{\ell k}(x_j - x_i) f_{\mathrm{in},j}^{(k)}$ is the value message term, including the message sent from node j to node i. This includes the positional difference between nodes $(x_j - x_i)$ and the input feature $f_{in,j}^{(k)}$ of node j.

It should be noted that a simplified version of the SE(3)-Transformer model that contains equilateral layers and 3D coordinates is adopted in our study to deal with the peptide structure.

Baseline models

A comprehensive evaluation of PepMSND is conducted using a diverse set of baseline models that are commonly applied in both peptide-specific and general molecular property prediction tasks. These baselines include traditional machine learning methods such as Support Vector Machine (SVM), Random Forest (RF), and *K*-Nearest Neighbors (KNNs), all of which rely on 0D physicochemical descriptors as input. To enable a systematic comparison across different molecular representation paradigms, we also incorporate deep learning models: RNN for 1D SMILES sequences, GIN for 2D molecular graph representations, and KAN for 0D descriptors. These models are widely used in studies on cyclic peptide membrane permeability[40] and peptide toxicity prediction.[41]

Support vector machines (SVMs)

SVM[42] is a powerful and widely utilized machine learning model for classification and regression tasks. In the classification tasks, its target is to determine an optimal hyperplane that separates data points belonging to different classes in the feature space. To achieve this goal, SVM relies on specific training samples, known as support vectors, which are the points nearest to the decision boundary.

Random forest (RF)

RF[43] is a prominent ensemble learning method for tasks such as classification, regression, and feature selection. It achieves prediction accuracy and robustness by constructing multiple decision tree models and aggregating their prediction results. In this model, each decision tree is independently generated based on a random subset of the training data, and randomness is considered when selecting the features for splitting. This effectively reduces the risk of overfitting.

Extreme gradient boosting (XGBoost)

XGBoost[44] is a machine learning algorithm based on the gradient boosting framework, capable of efficiently handling various machine learning tasks such as regression, classification and ranking. Its goal is to achieve efficient, flexible, and portable distributed gradient boosting.

K-nearest neighbors (KNNs)

KNN[45] is generally implemented in data mining and image classification tasks. This model classifies data points by majority voting based on the nearest neighbors in the feature space.

Graph isomorphism network (GIN)

GIN[46] is an efficient graph neural network model for processing graph-structured data. By combining a flexible neighbor information aggregation mechanism with multi-layer perceptrons, GIN can not only accurately capture the complex relationships between nodes but also progressively construct comprehensive local and global structural representations as the network depth increases. In particular, when dealing with heterogeneous graphs containing multiple types of nodes and edges, GIN leverages a specific message-passing mechanism and type-aware aggregation functions to effectively integrate different node information. This significantly enhances the model's generality.

Evaluation metrics

In this study, we adopt several metrics: Accuracy (ACC), Precision (Pre), F1 score (F1_Score), Recall, Area Under the Curve (AUC) and Matthews Correlation Coefficient (MCC) to evaluate the performance of models in predicting the blood stability of peptides:

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$

$$\mathrm{Pre} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

$$\mathrm{F1} = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}}$$

t-distributed Stochastic Neighbor Embedding (*t*-SNE)

t-SNE (*t*-distributed Stochastic Neighbor Embedding) is an efficient nonlinear dimensionality reduction algorithm designed to map high-dimensional data points to a two-dimensional or three-dimensional space while preserving the similarities or differences between data points as much as possible.

Results and discussion

Database

In our database, there are a total of 635 peptide entries, among which the number of cyclic peptides is 107, and the number of linear peptides is 528 (Fig. 2C). Among cyclic peptides, many of them are cyclized by disulfide bonds. To be more stable, most peptides have been modified in different ways, including D-residue replacement or introduction of N-terminal capping. In addition, more complex modifications such as *N*-methylation and reductive amination of amide bonds can also be found in this dataset. Fig. 2A displays that the lengths of most peptides range from 6 to 50 amino acids, accounting for 73% of the total dataset. As shown in Fig. 2D, in this dataset, the number of peptides evaluated in *in vivo* human blood is 115, while the number evaluated in*in vitro* human blood is 222. Additionally, the number of peptides tested in *in vivo* mouse blood is 217, and the number evaluated in *in vitro* mouse blood is 81, showing the diversity of our data. To more clearly illustrate the peptide blood stability distribution, we divide peptides into four categories: unstable, stable, highly stable, and non-degradable. Specifically, the stability of peptides can be classified according to the following criteria: a peptide is considered unstable if the proportion of the original peptide in the blood drops to less than 50% after 1 hour; if at least 50% of the original peptide remains unhydrolyzed from 1 to 6 hours, the peptide is classified as a stable peptide; if the proportion remains above 50% from 6 to 12 hours, it is regarded as highly stable; and if the original peptide remains unhydrolyzed for more than 12 hours, it is classified as a non-degradable peptide. As shown in Fig. 2B, the majority of peptides (approximately 57% of the dataset) belong to the unstable class. As for the stable peptides, they are mostly either cyclic in structure or feature certain modifications along the peptide chain, such as the substitution of non-natural residues, which contribute to enhancing their stability. To provide a richer information context, we adopted two-dimensional *t*-SNE projections of peptide molecules based on Morgan fingerprints, with color coding according to peptide chain length and modification status. These visualizations aim to explore the distribution characteristics of peptide molecules in chemical space.

In Fig. 2E, the distribution of peptides shows a clear trend of length-based clustering: peptides of similar lengths tend to aggregate in similar regions of the *t*-SNE space, indicating that peptide chain length has a significant impact on the structural features captured by Morgan fingerprints. This phenomenon suggests that even without any supervised model learning, the chemical structure information of peptides partially exhibits regularities related to sequence length.

Fig. 2F illustrates the influence of modification status on structural distribution. It can be observed that unmodified peptides typically form relatively compact clusters, whereas modified peptides are more often distributed across different regions or embedded within clusters of unmodified peptides. This distribution discrepancy likely arises because chemical modifications substantially change peptide molecules' structural characteristics, thus modifying how they cluster by similarity in fingerprint space.

The above visualization results are based on the original Morgan fingerprint representation, without the use of deep learning models or supervised embedding extraction. Therefore, these observations directly reflect the fundamental chemical structural features of peptide molecules.

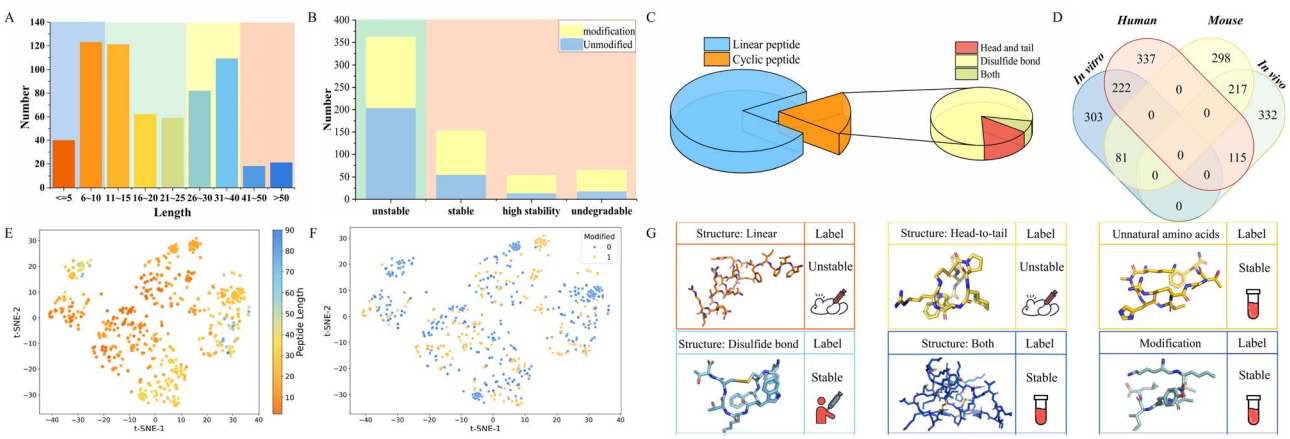

Fig. 2 (A) Overview of the peptide database. (B) Distribution of peptide stabilities in blood. (C) Structural classification of peptides. (D) The number of peptides across different blood environments. (E) Low-dimensional embedding of peptides colored by sequence length. (F) Low-dimensional embedding of peptides colored by the presence of chemical modifications. (G) Partial display of the database content.

Model performance comparison

We use a dataset comprising 635 peptide entries from our database and categorize them into two groups according to their stability in blood: those retaining more than 50% of their original structure after 1 hour are classified as stable peptides, while those that do not meet this criterion are designated as unstable peptides. To develop and evaluate predictive models, the dataset is partitioned into training and testing sets at a ratio of 9 : 1. Additionally, we employ 10-fold cross-validation on the training set to mitigate potential biases and ensure the robustness of our models.

Based on this dataset, we develop a novel model called PepMSND to predict blood stability. To comprehensively evaluate the performance of this model, we compare it to several different models (see Methods and ESI, Section 2†). In this comparison, all models are trained on the same datasets and evaluated by the 10-fold cross-validation. All results are average values from 10-fold cross-validation. Table 1 demonstrates that PepMSND exhibits excellent performance in these metrics, showing 0.867, 0.849, 0.836, 0.841, 0.912, and 0.726 in the Accuracy (ACC), Precision (Pre), Recall, F1 Score (F1_Score), Area Under the Curve (AUC), and Matthews Correlation Coefficient (MCC), respectively. Compared to other models, PepMSND demonstrates superiority in all metrics. In addition to AUC-ROC, we assessed PepMSND using AUC-PR, obtaining a score of 0.899 ± 0.041. In addition, we observe that the traditional descriptor-based models often perform better than the graph-based models. These results suggest that physicochemical properties provide an informative context that can improve performance on this task, and more training samples are required for the graph-based models to capture associated information. In Fig. 3, sections A1 to A4 demonstrate the performance of the model across various peptide chain structures. Whether on natural peptide chains or on more complex linear or cyclic peptides, the model can retain outstanding performance. This not only validates the broad applicability of this model but also highlights its stability and accuracy in handling peptide molecules with different chemical properties and structural complexities. Furthermore, Fig. 3B shows the model's robust discriminative ability. Specifically, the two peptide chains presented in Fig. 3B1 and 3B2 exhibit distinct differences in terms of their three-dimensional structures and stabilities, despite sharing completely the same amino acid sequences. By integrating input information from various modules, the model precisely captures the subtle feature differences between them and achieves accurate discrimination. Similarly, Fig. 3B3 and 3B4 exhibit only minor differences in amino acid configuration at the sequence level, yet the PepMSND model can realize this difference and make correct judgments.

The effect of different blood environments on model performance

We are also interested in the performance of our model in different experimental blood environments. Based on different species and experimental environments, we divide our test dataset into four classes. In most cases, PepMSND demonstrates satisfactory ability in predicting the stability of peptides. Take the prediction in the *in vivo* human blood condition as an example, PepMSND achieves 0.919, 0.894, 0.882, 0.867, 0.905, and 0.827 in the ACC, Pre, Recall, F1_score, AUC and MCC, which display the great power of this model to understand the peptide in this environment. The stable model performance indicates that our model delivers high accuracy with great generality. But we also notice that the change in the experimental environments also affects the model. The ACC gaps between Human/*In Vivo* and Human/*In Vitro* can reach 9.19%. However, such a phenomenon is not investigated in the experiments with Mouse/*In Vivo* and Mouse/*In Vitro*. Additionally, the species is also one factor affecting the model's ability. Compared to the ACC in the Mouse/*In Vitro* environment, the ACC in Human/*In Vitro* is lower (Table 2).

A prevalent issue in previous research practices is the frequent oversight by many researchers of the potential impact of species differences and experimental conditions on model predictive performance. While most peptides exhibit similar stability across different species and in both *in vivo* and *in vitro* blood stability tests, it cannot be overlooked that certain peptides demonstrate significant variations in blood stability under specific species and experimental conditions. This

Table 1 Performances of baseline models based on different peptide representation strategies (0D: physicochemical descriptors; 1D: SMILES sequences; 2D: molecular graphs) and the proposed multimodal model, evaluated using various metrics on the test set. All results are average values from 10-fold cross-validation[a]

Model	ACC	Pre	Recall	F1_Score	AUC-ROC	MCC
RF(0D)	0.784 ± 0.043	0.778 ± 0.061	0.690 ± 0.113	0.726 ± 0.074	0.863 ± 0.054	0.556 ± 0.087
SVM(RBF)(0D)	0.763 ± 0.056	0.788 ± 0.098	0.615 ± 0.116	0.682 ± 0.086	0.836 ± 0.036	0.516 ± 0.112
XGBoost(0D)	0.800 ± 0.047	0.780 ± 0.045	0.736 ± 0.108	0.753 ± 0.071	0.877 ± 0.052	0.587 ± 0.095
KNN(0D)	0.731 ± 0.058	0.721 ± 0.095	0.599 ± 0.104	0.652 ± 0.087	0.799 ± 0.057	0.444 ± 0.124
GIN(2D)	0.777 ± 0.031	0.785 ± 0.064	0.661 ± 0.088	0.712 ± 0.048	0.786 ± 0.033	0.537 ± 0.059
KAN(0D)	0.823 ± 0.057	0.845 ± 0.056	0.722 ± 0.124	0.771 ± 0.086	0.872 ± 0.057	0.635 ± 0.109
RNN(1D)	0.736 ± 0.060	0.739 ± 0.102	0.640 ± 0.237	0.639 ± 0.199	0.772 ± 0.085	0.456 ± 0.146
PeptideCLM(1D)	0.759 ± 0.053	0.762 ± 0.080	0.637 ± 0.143	0.682 ± 0.101	0.800 ± 0.064	0.500 ± 0.104
PepMSND	**0.867 ±0.043**	**0.849 ±0.071**	**0.836 ±0.063**	**0.841 ±0.053**	**0.912 ±0.037**	**0.726 ±0.086**

[a] Bold values represent the best.

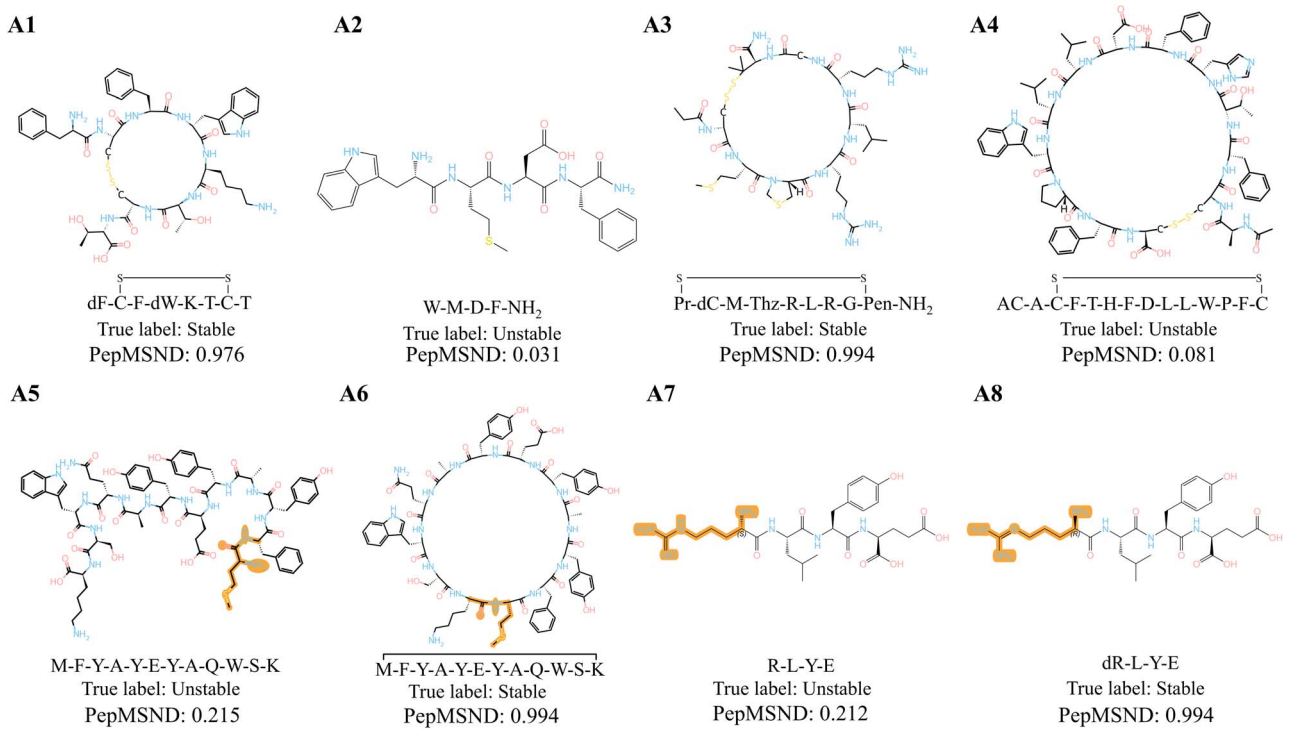

Fig. 3 The predictive performance of PepMSND for different sequences.

discrepancy often leads to the erroneous neglect of potentially promising peptides. Therefore, during the construction of the PepMSND model, we incorporated both as important features into the model and assigned them significant weights. To demonstrate the importance of these two pieces of information, we explore the model's performance when removing them. The results are displayed in Fig. 4. After the removal of these two conditions, significant drops appear in all evaluation indicators. The decrease in ACC, Recall, F1_Score, AUC, and MCC is 2.19%, 5.76%, 3.54%, 2.38%, and 4.32%, respectively, indicating that information about species and experimental conditions is beneficial to model performance.

The effect of peptide length on model performance

We are also interested in the influence of peptide length on model performance. Therefore, in this section, we conduct a detailed analysis of the model performance in peptides with different lengths. As shown in Table 3, the PepMSND model demonstrates its generality in these different subsets, achieving all ACC exceeding 0.800. Notably, in predicting the stability of peptides with lengths of 26–40, the ACC of our model can reach 0.935, demonstrating great predictive ability. In other metrics, this model also gains satisfactory results: 0.930 for Pre, 0.860 for Recall, 0.880 for F1_Score, 0.970 for AUC and 0.845 for MCC. This superior performance on peptides of length 26–40 may be partially attributed to their higher representation in the training dataset. However, such a trend could also be influenced by potential data leakage or sequence redundancy between training and test sets.

Ablation experiment

In this study, we use different modules to focus on different peptide features. Here, we conducted an ablation study to explore the effect of these different modules. Under the same experimental setup, we implement several variants of the

Table 2 Performances of different species and experimental environments with different metrics on the test set[a]

	ACC	Pre	Recall	F1_Score	AUC	MCC
Human/*In Vivo*	**0.919**	**0.894**	0.882	0.867	0.905	**0.827**
Human/*In Vitro*	0.827	0.858	0.783	0.811	0.867	0.648
Mouse/*In Vivo*	0.888	0.835	0.863	0.846	0.930	0.757
Mouse/*In Vitro*	0.894	0.875	**0.936**	**0.901**	**0.946**	0.797
PepMSND	0.867	0.849	0.836	0.841	0.912	0.726

[a] Bold values represent the optimal results.

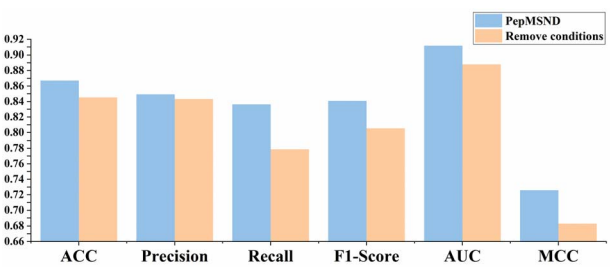

Fig. 4 Comparison of model performance before and after removing species and experimental environment information.

Table 3 Performances of different lengths with different metrics on the test set[a]

	ACC	Pre	Recall	F1_Score	AUC	MCC
5–25	0.818	0.812	0.808	0.806	0.865	0.638
26–40	**0.935**	**0.930**	**0.860**	**0.880**	**0.970**	**0.845**
5–40	0.857	0.838	0.831	0.833	0.905	0.706
PepMSND	0.867	0.849	0.836	0.841	0.912	0.726

[a] Bold values represent the optimal results.

PepMSND model for predicting peptide stability. These include the following:

(1) PepMSND_w/o_GAT: only containing Transformer, KAN and SE(3)-Transformer,

(2) PepMSND_w/o_Transformer: only containing GAT, KAN and SE(3)-Transformer,

(3) PepMSND_w/o_SE(3)-Transformer: only containing Transformer, GAT and KAN,

(4) PepMSND_w/o_KAN: only containing Transformer, GAT and SE(3)-Transformer.

Fig. 5A shows the performance comparisons of PepMSND and its variants. No surprise, the lack of one module can lead to a decline in PepMSND's performance. Compared to PepMSND_w/o_GAT, PepMSND_w/o_Transformer, PepMSND_w/o_SE(3)-Transformer, and PepMSND_w/o_KAN, our model displays a deeper understanding of this task, achieving the F1-Score improvement of 1.70%, 1.52%, 2.10%, and 11.72%, respectively. Such improvement can also be observed in the ACC metric, where our model reaches increases of 1.25%, 1.10%, 1.72%, and 9.53%. Furthermore, we conduct a detailed analysis to explore the effect of these modules on our model. As demonstrated in the figure, both PepMSND_w/o_GAT and PepMSND_w/o_Transformer exhibit comparable performance in predicting the blood stability of peptides. This indicates that both the Transformer and GAT architectures play equally significant roles in our model. It should be noted that Transformer focuses on the sequence feature, which is rather different from the graph-based GAT. The former contributes to finding the correlation between tokens in a sequence, and the latter captures the explicit information about atoms and bonds. Namely, the text feature (1D feature) is as same as the graph feature (2D feature). We also observed that the PepMSND model without the SE(3)-Transformer module (PepMSND_w/o_SE(3)-Transformer) performs less effectively than the two previously mentioned models, with an achieved accuracy (ACC) of 0.850 and an F1 score of 0.820. This underscores the significance of the 3D structure of peptides in our model. When the KAN module is removed from our model, we observe a more significant performance decline across all metrics, with a decrease of more than 0.100. This decrease is most pronounced in the MCC metric, which drops from 0.726 to 0.537. This suggests that the model without the KAN module has a worse ability to predict peptide blood stability. The inclusion of the KAN module appears to be crucial for the model's performance, likely due to its ability to capture important features and relationships involving different physicochemical properties. In other words, these peptide properties can assist the model in establishing a correlation between peptides and their blood stability. By incorporating these properties into the model, our model can better understand and predict how different peptides will behave in the blood, which is important for various applications such as drug design and peptide-based therapies. Fig. 5B shows that the absence of any one module not only reduces the confidence of our model but also leads to prediction errors. For instance, in Fig. 5B1, when the SE(3)-Transformer module is removed, it becomes difficult for this model to effectively distinguish linear and cyclic peptides based on low-dimensional feature information. It may be attributed to the absence of spatial structural information.

Feature visualization analysis

To comprehensively evaluate the effectiveness of PepMSND, we employ the *t*-SNE method for intuitive visualization analysis. Specifically, we first utilize *t*-SNE to visualize the embedding features before training. As shown in Fig. 6A, the pre-training data exhibits a disordered and random distribution in the feature space. Subsequently, after model training, we extract the embedding features from the last multi-source feature fusion module and use *t*-SNE to map them into a two-dimensional space for cluster analysis. As depicted in Fig. 6B, between positive and negative samples, a relatively clear boundary emerges after training. This indicates that the multimodal architecture of PepMSND successfully learns effective knowledge capable of distinguishing different samples.

The web server for PepMSND

To easily access the PepMSND model, we provide a user-friendly web interface. This web server is free and open to all users. This website does not acquire cookies and collect any personal information. It is compatible with most web browsers, including Microsoft Edge, Google Chrome, Apple Safari, and Mozilla Firefox on major operating systems such as Windows, macOS, and Linux.

Fig. 7A shows the homepage of the web interface, where users can click 'Services' and 'Database' to access the functional interface and database. Fig. 7C shows the interactive interface for predicting the stability of peptides in blood. Users can provide a peptide chain containing only natural amino acids in the Input Sequence input box, and then fill in and choose the remaining boxes for providing experimental information and binding sites for disulfide bonds. To further enhance the user experience, we have added 'Check' and 'Example' buttons. In this way, users can not only quickly grasp the input requirements of the web page, but also intuitively judge the correctness of the input through the examples. Upon completion of the prediction, the web page will display the percentage of the input peptide chain accounted for by each amino acid in the peptide chain, the confidence level and the blood stability classification in the results section. In addition, we offer the option to download HighFold to predict the structure of the input peptide.

Due to the limitations of the FASTA representation for peptides, many peptide sequences containing modifications

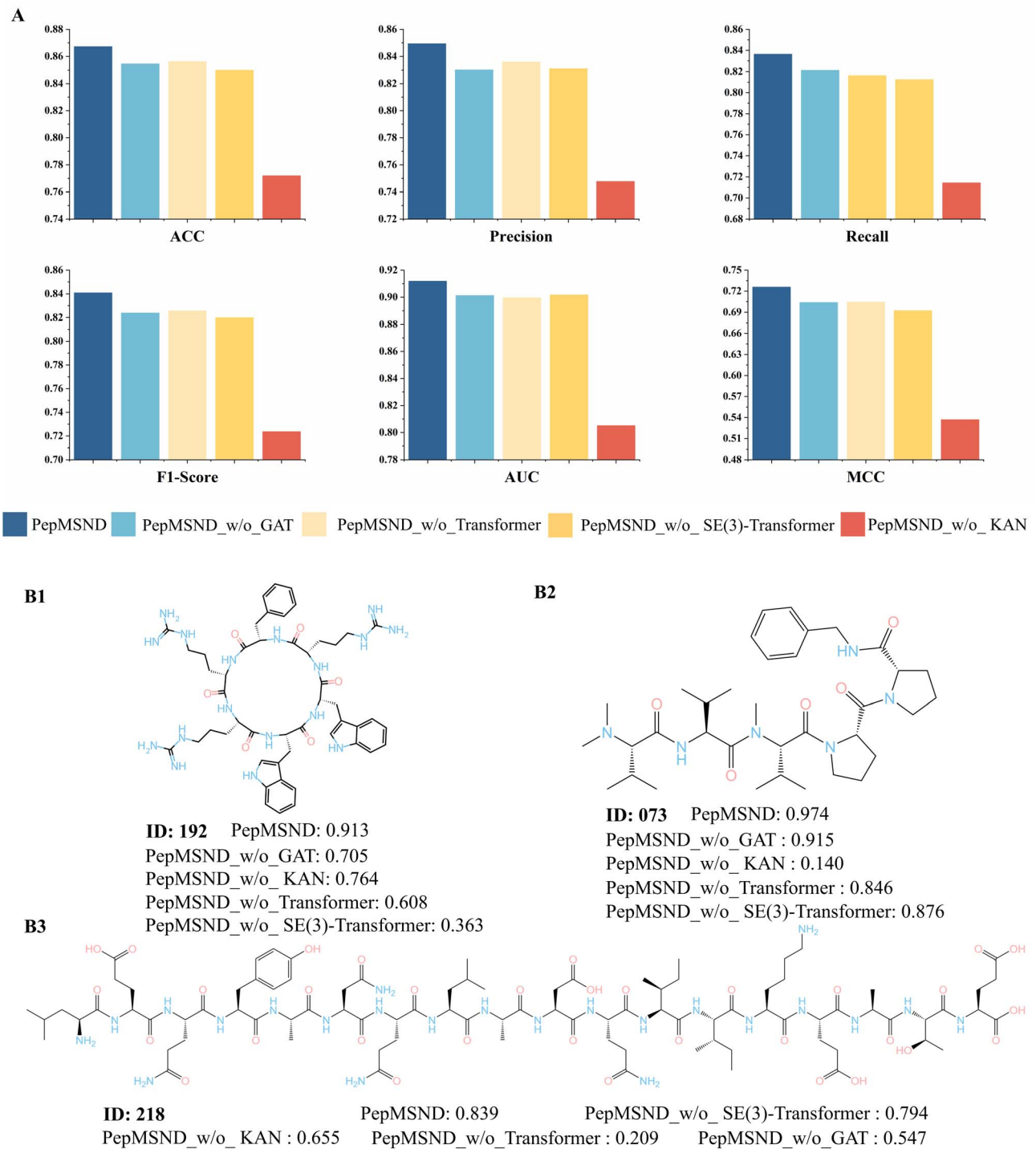

Fig. 5 The results of the ablation experiment. (A) The results across different metrics. (B) Representative examples of test datasets showing the confidence of PepMSND in predicting the blood stability of peptides with various feature combinations.

cannot be directly represented by FASTA. Compared to FASTA, SMILES is more suitable for our work, especially when calculating the physicochemical properties of peptides using RDKit or providing input for graph neural networks. To assist more users in conveniently obtaining the SMILES according to corresponding peptide sequences, we have developed a FASTA to SMILES functional interface on our webpage (Fig. 7B). When inputting the peptide sequence in the designated input box and selecting the type of cyclization from the dropdown menu, the users can obtain corresponding SMILES. If users choose disulfide bond cyclization, they also need to specify the binding sites of the disulfide bonds in another input box to ensure accurate SMILES. Similar to the peptide blood stability prediction interface, we provide "Check" and "Example" buttons to help

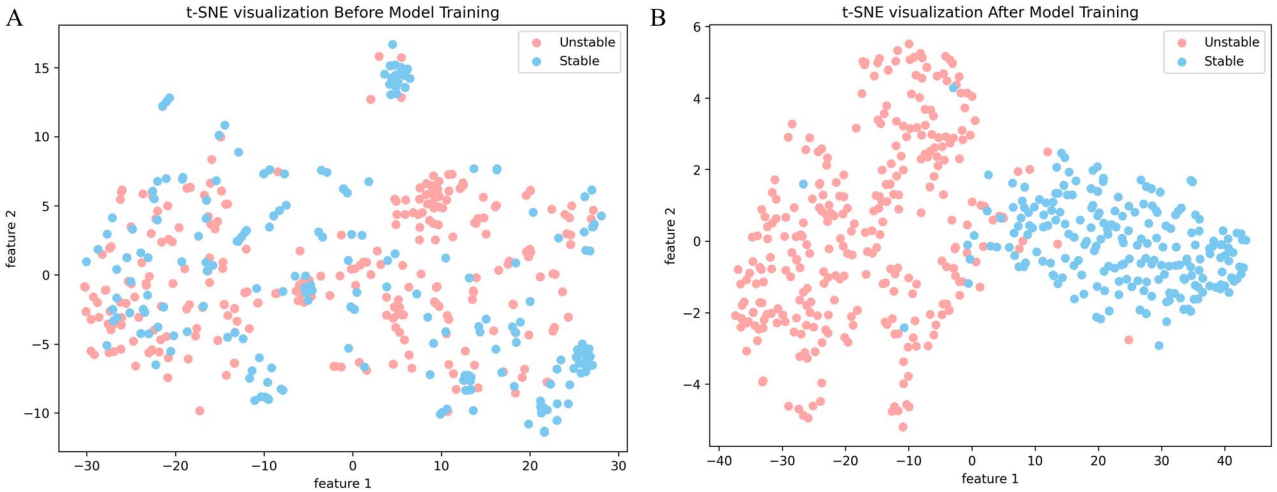

Fig. 6 Visualization of features. (A) *t*-SNE distribution of features before training. (B) *t*-SNE distribution of features after training.

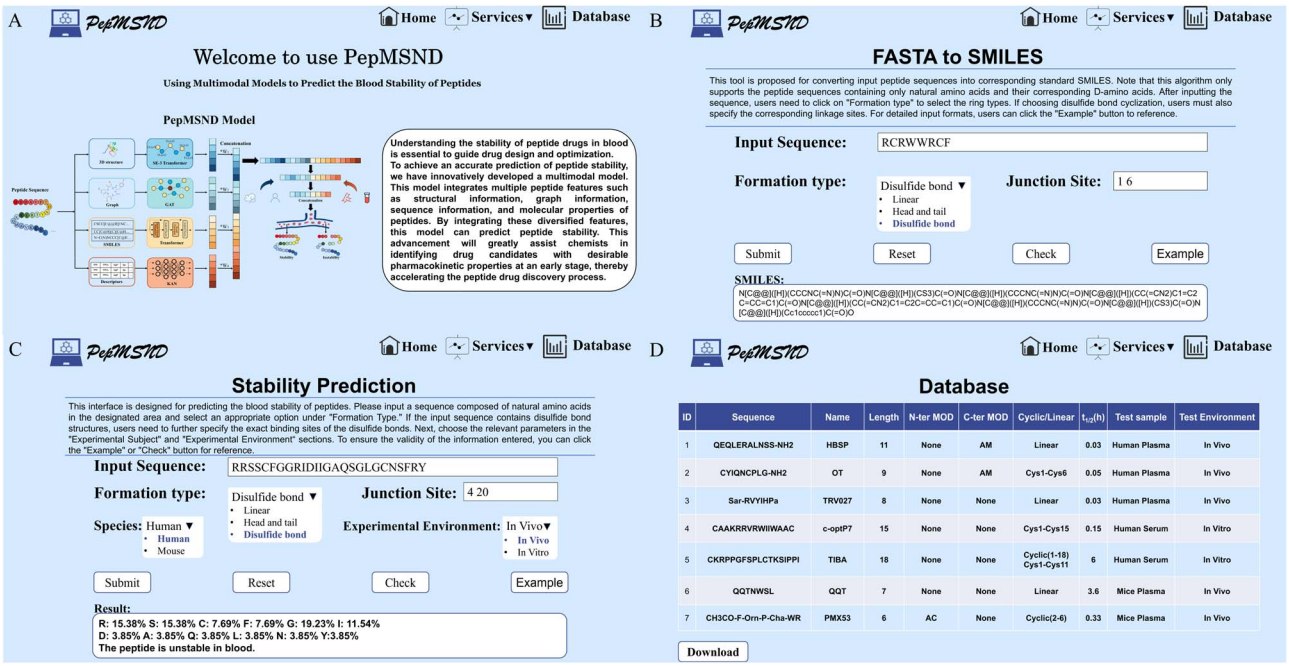

Fig. 7 PepMSND web server. (A) PepMSND's homepage. (B) The functional interface for converting FASTA into SMILES. (C) The functional interface for predicting the blood stability of polypeptides. (D) Presentation page of the database.

users verify their input and refer to the correct input format, thereby ensuring correct SMILES translation. Once the webpage completes the translation, the SMILES will be displayed in the results area for users to view and copy for their use.

Conclusions

In this study, we provide a comprehensive peptide blood stability database. This dataset has diverse blood experiments and peptides, comprising 635 samples. It represents the first open-source database that offers easy access to peptide stability in blood. Furthermore, we develop a multi-model model based on this dataset to predict peptide blood stability across various experimental settings. This model integrates multi-dimensional features, including 0D physicochemical properties, 1D sequence, 2D molecular structure, and 3D structure, to capture the correlation between peptides and their blood stability. At the same time, for the convenience of researchers' use, we not only integrate the model into an online server but also design a fully functional web interface to provide free access and use to researchers.

To evaluate the ability of our model, we compare it to baseline models such as GIN. In this performance comparison, our model exhibits an excellent ability to determine whether the peptide is stable in blood, achieving 0.867, 0.849, 0.836, 0.841, 0.912, and 0.726 in ACC, Pre, Recall, F1_Score, AUC, and MCC,

respectively. Additionally, our model shows improvements of 4.53% in ACC, 6.97% in F1_Score, 4.02% in AUC, and 9.05% in MCC, when compared to the best baseline model. This benchmark experiment demonstrates our model's superiority in predicting peptide blood stability.

To further explore the effect of different features in our model, we perform a series of ablation experiments. The results show that in this model, 1D sequence information and 2D structural information play equally important roles, while 3D structural information and 0D physicochemical property information play a more important role and can provide more useful information. However, although these variant models perform worse than our model, they still show superiority when compared to other baseline models such as SVM. The PepMSND model will accelerate the advancement of peptide drugs, enabling the screening of peptides with high blood stability at a lower cost in the early stages. It will also provide effective suggestions for peptide drug modification to researchers dedicated to designing stable and efficient peptide therapeutics.

It is important to note that the peptide blood stability dataset currently suffers from limitations in both quality and quantity, which pose substantial challenges to the real-world applicability of our model. In particular, our peptide stability database requires ongoing expansion and refinement. To address the issue of data scarcity, transfer learning emerges as a promising strategy, given its demonstrated effectiveness across various domains. Moreover, accurately obtaining high-resolution peptide structures remains difficult due to current limitations in structure prediction tools. Although we adopted a randomized 9 : 1 data split combined with 10-fold cross-validation to ensure robust evaluation, we acknowledge that this approach has inherent shortcomings when applied to peptide datasets. Specifically, the presence of highly similar or nearly identical sequences in both training and test sets can lead to overly optimistic estimates of generalization performance. To better evaluate the model's capability on truly novel peptides, future work will investigate more rigorous data partitioning strategies, such as scaffold-based or similarity-aware splitting methods (*e.g.*, clustering or Tanimoto similarity thresholds), which are better suited to mitigate data leakage and enhance the assessment of real-world performance.

Data availability

The code and dataset used in this study are publicly available in the GitHub repository at **https://github.com/hmenghu/PepMSND**. The dataset and code have also been deposited in the Zenodo repository and can be accessed *via* the following DOI: **10.5281/zenodo.15687572**. This repository contains: (i) curated and processed peptide blood stability datasets used for model training and evaluation; and (ii) code for data preprocessing and model training.

Author contributions

Haomeng Hu: writing – original draft, software, data curation, investigation, validation, visualization, formal analysis. Chengyun Zhang: writing – original draft, visualization. Zhenyu Xu: software. Jingjing Guo: writing – review & editing. An Su: writing – review & editing. Chengxi Li: writing – review & editing. Hongliang Duan: conceptualization, supervision.

Conflicts of interest

The authors declare no competing financial interest.

Acknowledgements

This project was supported by the Macao Science and Technology Development Fund (Grant No. 0151/2024/RIA2), the internal grant from Macao Polytechnic University (RP/FCA-07/2024), and the Natural Science Foundation of Zhejiang Province (LD22H300004).

Notes and references

1 Q. Yang, Z. Hu, H. Jiang, J. Wang, H. Han, W. Shi and H. Qian, *Chin. J. Nat. Med.*, 2025, **23**, 31–42.
2 L. Otvos and J. D. Wade, *Front. Chem.*, 2014, 2.
3 M. Erak, K. Bellmann-Sickert, S. Els-Heindl and A. G. Beck-Sickinger, *Bioorg. Med. Chem.*, 2018, **26**, 2759–2765.
4 N. Yin, M. A. Brimble, P. W. Harris and J. Wen, *Med. Chem.*, 2014, **4**, 763–769.
5 R. Böttger, R. Hoffmann and D. Knappe, *PLoS One*, 2017, **12**, e0178943.
6 M. T. Weinstock, J. N. Francis, J. S. Redman and M. S. Kay, *Biopolymers*, 2012, **98**, 431–442.
7 L. Pollaro and C. Heinis, *Med. Chem. Commun.*, 2010, **1**, 319–324.
8 S. T. Buckley, F. Hubálek and U. L. Rahbek, *Tissue Barriers*, 2016, **4**, e1156805.
9 L. M. Berezhkovskiy, *J. Pharmaceut. Sci.*, 2013, **102**, 2082–2084.
10 L. C. Burnett, A. A. Skowronski, R. Rausch, C. A. LeDuc and R. L. Leibel, *Int. J. Obes.*, 2017, **41**, 355–359.
11 A. Varshavsky, *Genes Cells*, 1997, **2**, 13–28.
12 A. Bachmair, D. Finley and A. Varshavsky, *Science*, 1986, **4**, 234.
13 D. K. Gonda, A. Bachmair, I. Wünning, J. W. Tobias, W. S. Lane and A. Varshavsky, *J. Biol. Chem.*, 1989, **264**, 16700–16712.
14 J. W. Tobias, T. E. Shrader, G. Rocap and A. Varshavsky, *Science*, 1991, **254**, 1374–1377.
15 K. Guruprasad, B. V. B. Reddy and M. W. Pandit, *Protein Eng. Des. Sel.*, 1990, **4**, 155–161.
16 M. Cavaco, J. Valle, I. Flores, D. Andreu and M. A. R. B. Castanho, *Clin. Transl. Sci.*, 2021, **14**, 1349–1358.
17 D. Mathur, S. Singh, A. Mehta, P. Agrawal and G. P. S. Raghava, *PLoS One*, 2018, **13**, e0196829.
18 F. Wang, N. Sangfuang, L. E. McCoubrey, V. Yadav, M. Elbadawi, M. Orlu, S. Gaisford and A. W. Basit, *Int. J. Pharm.*, 2023, **634**, 122643.
19 J. Shirian, A. Hockla, J. J. Gleba, M. Coban, N. Rotenberg, L. M. Strik, A. Alasonyalilar Demirer, M. L. Pawlush,

J. A. Copland, E. S. Radisky and J. M. Shifman, *Biomolecules*, 2024, **14**, 1187.
20 L. T. Nguyen, J. K. Chau, N. A. Perry, L. De Boer, S. A. J. Zaat and H. J. Vogel, *PLoS One*, 2010, **5**, e12684.
21 M. Cavaco, D. Andreu and M. A. R. B. Castanho, *Angew. Chem., Int. Ed.*, 2021, **60**, 1686–1688.
22 D. Mathur, S. Prakash, P. Anand, H. Kaur, P. Agrawal, A. Mehta, R. Kumar, S. Singh and G. P. S. Raghava, *Sci. Rep.*, 2016, **6**, 36617.
23 D. S. Wishart, Y. D. Feunang, A. C. Guo, E. J. Lo, A. Marcu, J. R. Grant, T. Sajed, D. Johnson, C. Li, Z. Sayeeda, N. Assempour, I. Iynkkaran, Y. Liu, A. Maciejewski, N. Gale, A. Wilson, L. Chin, R. Cummings, D. Le, A. Pon, C. Knox and M. Wilson, *Nucleic Acids Res.*, 2018, **46**, D1074–D1082.
24 S. S. Usmani, G. Bedi, J. S. Samuel, S. Singh, S. Kalra, P. Kumar, A. A. Ahuja, M. Sharma, A. Gautam and G. P. S. Raghava, *PLoS One*, 2017, **12**, e0181748.
25 C. A. Lutomski, T. J. El-Baba, C. V. Robinson, R. Riek, S. H. W. Scheres, N. Yan, M. AlQuraishi and L. Gan, *Cell*, 2022, **185**, 2617–2620.
26 J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli and D. Hassabis, *Nature*, 2021, **596**, 583–589.
27 M. Baek, F. DiMaio, I. Anishchenko, J. Dauparas, S. Ovchinnikov, G. R. Lee, J. Wang, Q. Cong, L. N. Kinch, R. D. Schaeffer, C. Millán, H. Park, C. Adams, C. R. Glassman, A. DeGiovanni, J. H. Pereira, A. V. Rodrigues, A. A. Van Dijk, A. C. Ebrecht, D. J. Opperman, T. Sagmeister, C. Buhlheller, T. Pavkov-Keller, M. K. Rathinaswamy, U. Dalwadi, C. K. Yip, J. E. Burke, K. C. Garcia, N. V. Grishin, P. D. Adams, R. J. Read and D. Baker, *Science*, 2021, **373**, 871–876.
28 Z. Lin, H. Akin, R. Rao, B. Hie, Z. Zhu, W. Lu, N. Smetanin, R. Verkuil, O. Kabeli, Y. Shmueli, M. Fazel-Zarandi, T. Sercu, S. Candido and A. Rives, *Science*, 2023, **379**, 1123–1130.
29 C. Zhang, C. Zhang, T. Shang, N. Zhu, X. Wu and H. Duan, *Briefings Bioinf.*, 2024, **25**, bbae215.
30 E. F. McDonald, T. Jones, L. Plate, J. Meiler and A. Gulsevin, *Structure*, 2023, **31**, 111–119.
31 S. Wang, J. Witek, G. A. Landrum and S. Riniker, *J. Chem. Inf. Model.*, 2020, **60**, 2044–2058.
32 Y. Huang, H. Zhao and L. Huang, *arXiv*, 2021, preprint, arXiv:2106.04538, DOI: **10.48550/arXiv.2106.04538**.
33 Z. Liu, Y. Wang, S. Vaidya, F. Ruehle, J. Halverson, M. Soljačić, T. Y. Hou and M. Tegmark, *arXiv*, 2024, preprint, arXiv:2404.19756, DOI: **10.48550/arXiv.2404.19756**.
34 A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser and I. Polosukhin, *arXiv*, 2021, preprint, arXiv:1706.03762, DOI: **10.48550/arXiv.1706.03762**.
35 P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò and Y. Bengio, *arXiv*, 2018, preprint, arXiv:1710.10903, DOI: **10.48550/arXiv.1710.10903**.
36 F. B. Fuchs, D. E. Worrall, V. Fischer and M. Welling, *arXiv*, 2020, preprint, arXiv:2006.10503, DOI: **10.48550/arXiv.2006.10503**.
37 C. J. Vaca-Rubio, L. Blanco, R. Pereira and M. Caus, *arXiv*, 2024, preprint, arXiv:2405.08790, DOI: **10.48550/arXiv.2405.08790**.
38 A. S. Mubarak, Z. S. Ameen, S. Mati, A. Lasisi, Q. N. Naveed and R. A. Abdulkadir, *Heliyon*, 2024, **10**, e40799.
39 O. Cherednichenko and M. Poptsova, *Briefings Bioinf.*, 2025, **26**, bbaf129.
40 J. Li, K. Yanagisawa and Y. Akiyama, *Briefings Bioinf.*, 2024, **25**, bbae417.
41 J.-H. Wang and T.-Y. Sung, *ACS Omega*, 2024, **9**, 32116–32123.
42 M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt and B. Scholkopf, *IEEE Intell. Syst. Their Appl.*, 1998, **13**, 18–28.
43 L. Breiman, *Mach. Learn.*, 2001, **45**, 5–32.
44 T. Chen and C. Guestrin, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
45 T. Cover and P. Hart, *IEEE Trans. Inform. Theory*, 1967, **13**, 21–27.
46 K. Xu, W. Hu, J. Leskovec and S. Jegelka, *arXiv*, 2019, preprint, arXiv:1810.00826, DOI: **10.48550/arXiv.1810.00826**.