OXFORD

Briefings in Bioinformatics, 2024, **25(6)**, bbae505
https://doi.org/10.1093/bib/bbae505
Problem Solving Protocol

MultiFeatVotPIP: a voting-based ensemble learning framework for predicting proinflammatory peptides

Chaorui Yan[1], Aoyun Geng[1], Zhuoyu Pan[2], Zilong Zhang[1], Feifei Cui[1,*]

[1]School of Computer Science and Technology, Hainan University, 58 Renmin Avenue, Meilan District, Haidian Campus, Haikou 570228, China
[2]International Business School, Hainan University, 58 Renmin Avenue, Meilan District, Haidian Campus, Haikou 570228, China
*Corresponding author. School of Computer Science and Technology, Hainan University, 58 Renmin Avenue, Meilan District, Haidian Campus, Haikou 570228, China. E-mail: feifeicui@hainanu.edu.cn

Abstract

Inflammatory responses may lead to tissue or organ damage, and proinflammatory peptides (PIPs) are signaling peptides that can induce such responses. Many diseases have been redefined as inflammatory diseases. To identify PIPs more efficiently, we expanded the dataset and designed an ensemble learning model with manually encoded features. Specifically, we adopted a more comprehensive feature encoding method and considered the actual impact of certain features to filter them. Identification and prediction of PIPs were performed using an ensemble learning model based on five different classifiers. The results show that the model's sensitivity, specificity, accuracy, and Matthews correlation coefficient are all higher than those of the state-of-the-art models. We named this model MultiFeatVotPIP, and both the model and the data can be accessed publicly at https://github.com/ChaoruiYan019/MultiFeatVotPIP. Additionally, we have developed a user-friendly web interface for users, which can be accessed at http://www.bioai-lab.com/MultiFeatVotPIP.

Keywords: proinflammatory peptide; inflammation; feature encoding; machine learning; ensemble learning

Introduction

Inflammation represents the host tissue or organ response to detrimental stimuli and is characterized by various outcomes, including functional impairment at the stimulation site [1], localized vasodilation, and fever [2]. Cytokines are secretory proteins that facilitate intercellular communication and regulate key physiological processes including immune responses, inflammation, and cell proliferation. Cytokines are produced by various cells, including immune, inflammatory, and other tissue cells [3]. By acting on specific receptors, they influence the function and behavior of target cells [4]. Cytokines can be classified into proinflammatory signals—proinflammatory cytokines that promote the inflammatory response process, and anti-inflammatory signals—anti-inflammatory cytokines that inhibit the process [2]. The release of proinflammatory cytokines in response to tissue damage or infection initiates an inflammatory response. These cytokines promote vasodilation, migration, and activation of white blood cells and the release of other inflammatory mediators, ultimately leading to inflammation. Ongoing research on inflammatory diseases has led researchers to categorize several conditions including depression [5], obesity [6], heart disease, and Alzheimer's disease [7] as inflammatory. In this context, an increasing number of researchers have realized that studying the mechanisms underlying inflammation is crucial.

Peptides that induce proinflammatory cytokines are referred to as proinflammatory peptides (PIPs) and are considered potential therapeutic candidates for alleviating and curing various diseases [8–10]. Current traditional experimental methods for identifying specific peptides have drawbacks, including time consumption, high costs, and challenges in high-throughput applications. Hence, researchers prefer sequence-based computational approaches to screen potential candidates before experimental verification to enhance efficiency and reduce costs [11, 12]. Tools for predicting the pro-inflammatory response to inducing peptides are limited, with only a few available tools, such as the ProInflam method proposed by Gupta *et al.* [10], the Proinflammatory Inducing Peptides - Ensemble Learning (PIP-EL) method by Manavalan *et al.* [13], and the ProIn-Fuse method by Khatun *et al.* [14]. The ProInflam and ProIn-Fuse methods employ the Support Vector Machine (SVM) [15, 11] and Random Forest (RF) [16, 17] classifiers, respectively, for PIP prediction, whereas PIP-EL utilizes an ensemble learning strategy. Ensemble learning, through the integration of multiple algorithms, improves prediction accuracy and stability, making it highly effective for complex bioinformatics challenges. These methods have been used extensively for peptide identification. For example, Hongwu *et al.* created an ensemble model combining the Light Gradient Boosting Machine (LightGBM) and logistic regression for Antimicrobial Peptide (AMP) prediction that yielded positive outcomes [18]. Ke *et al.* employed an ensemble of four classifiers for precise prediction of therapeutic peptides [19]. With the continual updating of the Immune Epitope Database (IEDB) [20] and further research, there is hope for more tools based on

Received: June 5, 2024. **Revised:** September 1, 2024. **Accepted:** September 30, 2024

new technologies and methodologies to be developed, improving and enriching the capability to predict the proinflammatory response to inducing peptides. These new tools may integrate more bioinformatics and machine learning algorithms, leveraging large datasets for training and validation to enhance the accuracy and reliability of predictions.

To achieve this goal, we developed a new computational method called MultiFeatVotPIP to assist in the preliminary prediction of PIPs using computational techniques. In this study, we created a nonredundant dataset that was divided into training and independent test sets. We then encoded the raw peptide sequence data using various encoding methods such as dipeptide deviation from expected mean (DDE), dipeptide composition (DPC), Amino Acid index (AAindex), amino acid composition (AAC), and grouped di-peptide composition (GDPC). We further applied the recursive feature elimination (RFE) strategy to select the most relevant features, which were subsequently used to train and predict the classification models. Finally, we integrated five classifiers for prediction: AdaBoost, XGBoost, RF, Gradient Boosting Decision Tree (GBDT), and LightGBM. Our model demonstrated superior performance in several key aspects compared to current state-of-the-art (SOTA) methods.

Methods and materials

Dataset

The IEDB [20] serves as an exhaustive resource specifically crafted for the aggregation, structuring, and dissemination of antigen and immune epitope information. The primary aim of the IEDB is to offer an integrated platform that assists researchers in accessing and sharing immune system-related data. Information in the database is pertinent to areas such as immunology, vaccine development, and immunotherapy.

Peptides triggering proinflammatory cytokines, known as PIPs, include cytokines like IL-1, IL-2, IL-12, IL-17, IL-18, IFN-γ, and TNF-α. In human and mouse T-cell experiments, peptides that induce proinflammatory cytokines are classified as having a positive proinflammatory effect. Based on previous studies, we found that different ranges of proinflammatory cytokines were used during dataset collection. The PronIn-Fuse and ProInflam methods limited the range of proinflammatory cytokines to IL-1α, IL-1β, TNF-α [21, 22], IL-12, IL-18 [23], and IL-23 [24]. In contrast, the PIP-EL method employs a broader range of cytokines, including IL-6, IL-8 [25], and IL-17 [26], which has been validated in various studies. Therefore, to ensure that our model is more representative of real-world conditions and possesses greater robustness, we adopted the same range of proinflammatory cytokines as the PIP-EL method. Additionally, considering that the dataset provided by the PIP-EL method was no longer accessible and that the IEDB database was updated in 2018, we recollected the dataset. We searched the IEDB database using the nine proinflammatory cytokines (IL-1α, IL-1β, IL-6, TNF-α, IL-12, IL-23, IL-8, IL-18, and IL-17) as keywords, from both human and mouse species. Peptides containing any of these nine proinflammatory factors were considered positive, whereas those not containing any of them were considered negative. Peptide sequences outside the 5–25 length range were deemed outliers and were excluded, focusing on sequences within this range in our candidate dataset [10]. We then applied CD-HIT to reduce redundancy and achieved a dataset sequence identity threshold of 0.60 [27]. The collected dataset was divided into training and test datasets. Given the necessity of a comparison with the current SOTA model, we excluded the data used for training the SOTA model from the independent test set. This step was performed to ensure the fairness and validity of the comparative experiment. Ultimately, we obtained 2872 training samples and 342 testing samples. The training dataset contained 1245 positive and 1627 negative samples, whereas the testing dataset included 171 positive and 171 negative samples. The detailed dataset information is presented in Table 1. The training and independent datasets used in this study are available at https://github.com/ChaoruiYan019/MultiFeatVotPIP.

Table 1. Dataset composition.

	PIP (train)	n-PIP (train)	PIP (test)	n-PIP (test)
ProIn-fuse (SOTA)	607	1098	134	156
MultiFeatVotPIP	1245	1627	171	171

Architecture of MultiFeatVotPIP

The workflow of MultiFeatVotPIP is shown in Fig. 1. Initially, data were collected from the IEDB database, followed by preprocessing and partitioning to acquire the training and test datasets used in our study. Subsequently, the data underwent feature encoding, in which they were manually encoded in five distinct ways using iLearnPlus [28, 29] and integrated into a single feature space, preliminarily encoding each peptide feature into a 1345-dimensional feature vector. Second, considering feature reduction, we utilized an RF to determine the importance ranking of features and performed RFE based on feature importance, ultimately selecting the top 490 features as the final feature space for peptide sequences [30]. Finally, we employed five base learners for learning and obtained the final prediction outcomes using a soft-voting ensemble strategy.

Feature representation

Our peptide sequences consist of 20 amino acids, represented by the abbreviations A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, and Y. To transform amino acid sequence data into numerical representations for machine learning classification, we utilized five peptide sequence feature-encoding methods: AAC, DPC, AAindex, DDE, and GDPC, creating a 1345-dimensional mixed feature space. Adopting various feature-encoding methods enriches the feature space. Table 2 lists all the feature-encoding methods used.

Amino acid composition

The AAC feature-encoding method produces a 20-dimensional feature vector. These dimensions correspond to the proportion of each of the 20 amino acids in a peptide. The formula is as follows:

$$AAC(i) = \frac{N(i)}{N} \tag{1}$$

where $N(i)$ refers to the count of the ith type of amino acid among the 20 amino acids, and N denotes the length of the peptide sequence.

Dipeptide composition

The DPC describes the composition of dipeptides, each consisting of two amino acids. This represents the distribution of dipeptides within the peptide sequence. The DPC was calculated as the ratio of specific dipeptides to the total number of possible dipeptides (400 combinations), normalized between 0 and 1. The formula for DPC is:

$$cDPC(i) = \frac{N(i)}{N_{total}} \tag{2}$$

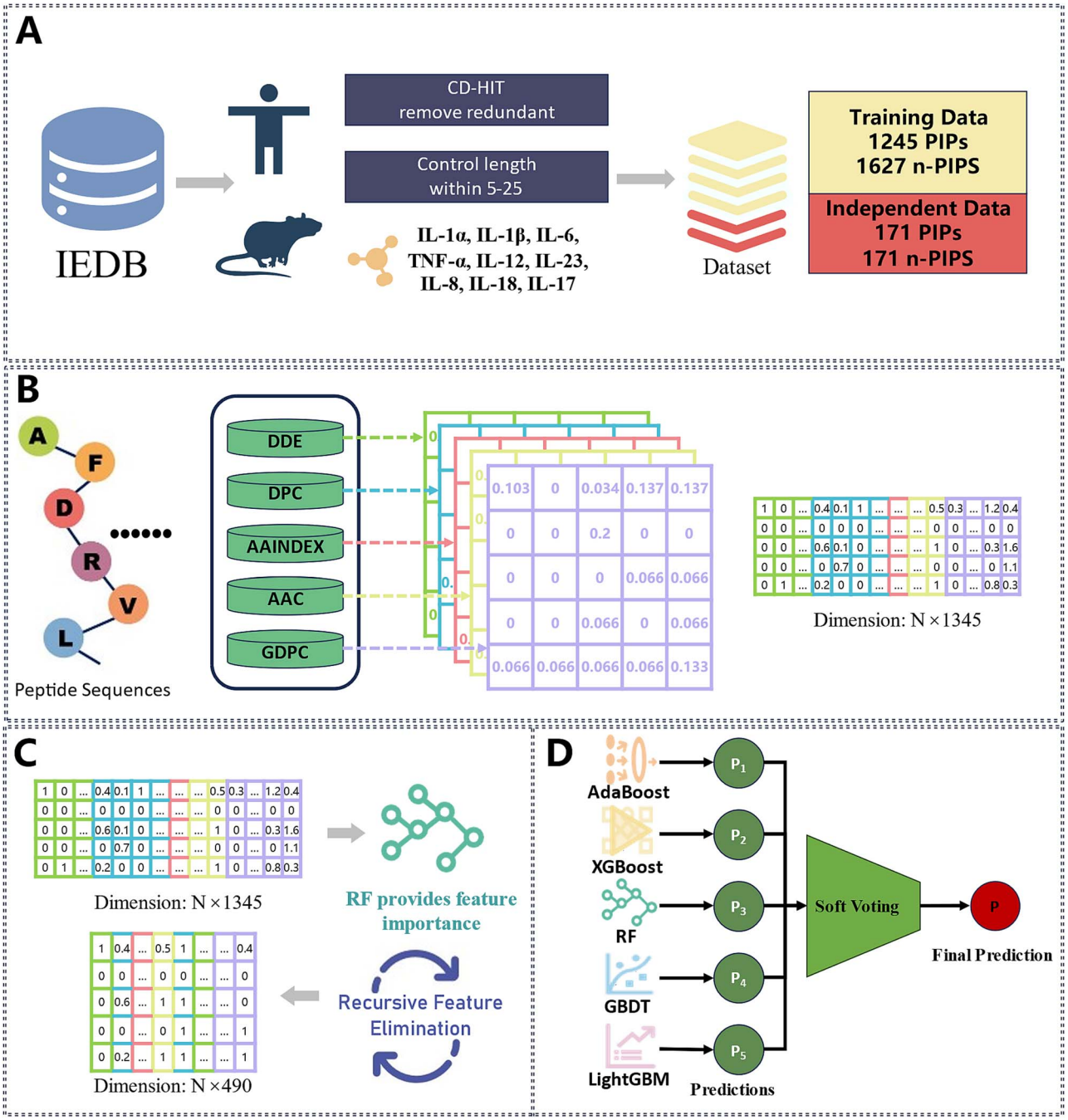

Figure 1. Workflow diagram of the MultiFeatVotPIP model. (A) Data collection and preprocessing. (B) Feature encoding. (C) Feature selection. (D) Model training and ensemble learning.

where $N(i)$ is the occurrence number of a specific dipeptide i in the sequence, and N_{total} is the total number of possible dipeptide occurrences in the sequence (for nonoverlapping dipeptides, N_{total} equals the sequence length minus 1). The DPC feature yielded a 400-dimensional vector, with each dimension representing a specific dipeptide.

Amino Acid index

The AAindex encoding method was based on the amino acid index database, which contains various physicochemical properties (PCPs) of amino acids. For each property, the database provided numerical values for 20 standard amino acids. To encode the peptide sequences, the AAindex database was queried to retrieve the relevant PCPs of each amino acid in the sequence. Properties with "NA" values are excluded, resulting in a feature vector. Ultimately, this method yielded a 500-dimensional feature vector representing the combined PCPs of the amino acids in the peptide [31].

Dipeptide deviation from expected mean

The DDE encoding method quantifies the compositional properties of dipeptides in protein sequences by measuring the deviation in the observed frequency of each dipeptide from its expected mean value. This method generates a 400-dimensional feature vector, as there are 400 possible dipeptides formed by 20 standard amino acids. The calculation process involved the following

Table 2. The five manual feature encoding methods used and their encoding dimensions.

Feature descriptor	Dimension	Description	Reference
Amino acid composition (AAC)	20	Describes the proportion of amino acids in peptides, focusing on capturing the overall compositional features of peptides.	[31]
Dipeptide composition (DPC)	400	Describes the composition of dipeptides, focusing on capturing the interactions of neighboring amino acids to predict functions.	[31]
Amino Acid Index Database (AAindex)	500	Based on the Amino Acid index database, this method encodes various physicochemical properties of amino acids, producing a feature vector that represents the combined physicochemical properties of the amino acids in the peptide.	[32]
Dipeptide Deviation from Expected Mean (DDE)	400	Measures how much the observed frequency of each dipeptide deviates from its expected mean value, generating a feature vector that captures the compositional properties of dipeptides.	[33]
Grouped dipeptide composition (GDPC)	25	Categorizes amino acids into five groups based on their physicochemical properties and calculates the frequency of dipeptides formed by these groups, yielding a 25-dimensional feature vector.	[34]

steps:

$$Dc(i) = \frac{n_i}{N} \tag{3}$$

$$TM(i) = \frac{C_{i_1}}{CN} \times \frac{C_{i_2}}{CN} \tag{4}$$

$$TV(i) = \frac{TM(i) \times (1 - TM(i))}{N} \tag{5}$$

$$DDE(i) = \frac{DC(i) - TM(i)}{\sqrt{TV(i)}} \tag{6}$$

where $n_i$is the frequency of dipeptide i in the sequence, and N is the total number of dipeptides in the sequence (the sequence length minus 1), C_{i_1} and C_{i_2} are the counts of codons for the first and second amino acids in dipeptide i, respectively, and CN is the total number of codons, excluding stop codons (61 codons). The resulting DDE feature vector for a peptide sequence PPP is represented as

$$DEE_p = \{DDE(1), DDE(2), \ldots, DDE(400)\} \tag{7}$$

Grouped Di-Peptide Composition

The GDPC encoding method categorizes the 20 standard amino acids into five distinct groups based on their PCPs [32]. These groups were defined as follows: aliphatic, aromatic, positively charged, negatively charged, and uncharged. The GDPC method involves calculating the frequency of dipeptides formed by these groups, resulting in 25 descriptors. The descriptors were computed using the following formula:

$$GDPC = \frac{N_{R_iR_j}}{L-1} \tag{8}$$

where $N_{R_iR_j}$ $(1 \le i, j \le 5)$ represents the number of occurrences of the residue pair $R_iR_j$in the sequence, where R_i and R_j are amino acids belonging to one of the five groups. L is the sequence length. The GDPC method ultimately yields a 25-dimensional feature vector because there are $5 \times 5 = 25$ possible pairs of five groups.

Feature optimization

In our model, we employed five diverse manual encoding methods to enhance the feature space, enabling a more thorough representation of the PIPs' characteristics. However, adopting these diverse feature-encoding methods significantly increases data dimensionality. High-dimensional feature spaces increase the computational complexity and may introduce noise, potentially diminishing the generalization ability of the model. To address this challenge, we implemented a feature selection strategy to refine the feature space. This approach not only simplifies the model and shortens the training times but also boosts the prediction accuracy.

In our study, we employed RF, a robust ensemble learning method, to evaluate feature importance. The RF model provided a contribution score for each feature, indicating its impact on the predictive performance. We then applied an RFE strategy, integrating parameter search and performance evaluation, to sequentially remove less critical features based on their importance scores [33]. This iterative process was terminated when ~490 features were retained, yielding the optimal performance of the model.

Machine learning method

The MultiFeatVotPIP model incorporates five core learners: AdaBoost [34], XGBoost [35, 36], RF [37], GBDT [38], and LightGBM. By integrating these learners, our goal is to harness their combined strengths and create a robust classifier for PIPs. For fine-tuning, we used a grid-search strategy to optimize the hyperparameters. Cross-validation ensures the generalizability of the hyperparameter tuning and minimizes overfitting risks [39].

To boost the predictive power of our model, we employ a voting ensemble strategy [38], integrating five core learners. In ensemble learning, the voting strategy is a meta-algorithm that combines predictions from multiple models to produce more accurate forecasts and is widely adopted in the field of bioinformatics [40]. We implemented soft voting, which differs from hard voting in that it aggregates probability estimates from each model and selects the class with the highest average probability as the final prediction. Soft voting leverages the confidence level of each learner's prediction, thereby providing a more nuanced and often more accurate final decision, particularly in the case of diverse models such as AdaBoost, XGBoost, RF, GBDT, and LightGBM.

Initially, we independently trained each learner and then aggregated their predictions for the final outcome using a soft voting strategy. This strategy enables our model to harness the diverse expertise of learners and ensure robust and reliable PIP classification. The observed performance improvement in the validation process validated this approach.

Performance evaluation

To assess the performance of our model, we utilized four key metrics [41]: sensitivity (SN), specificity (SP), accuracy (ACC), area under the ROC (AUC), and Matthews correlation coefficient (MCC) [42]. These metrics were derived from true positives (TPs) for correctly identified positive cases, false positives (FPs) for incorrect positive identifications, true negatives (TNs) for correct negative identifications, and false negatives (FNs) for incorrect negative cases. The formulae for these metrics are as follows:

$$SN = \frac{TP}{TP + FN} \tag{9}$$

$$SP = \frac{TN}{TP + FN} \tag{10}$$

$$ACC = \frac{TP + TN}{TP + FP + TN + FN} \tag{11}$$

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)\,(TP + FN)\,(TN + FP)\,(TN + FN)}} \tag{12}$$

These metrics were critical for evaluating the performance of the proposed model. SN measures how well the model identifies positive cases, SP assesses the accuracy in identifying negative cases, ACC denotes the overall correctness of the model's predictions, and the MCC offers a balanced evaluation and is particularly valuable for imbalanced datasets [10]; AUC gauges performance across all classification thresholds, with a high AUC indicating strong differentiation between classes [43].

Results

Compositional and positional information analysis

Amino acid enrichment at specific positions can help distinguish between PIPs and n-PIPs. Using Two Sample Logo [44] allows us to analyze the enrichment of amino acids at different positions within the sequences. This method generates two sets of visual information: the enrichment of positive samples and the depletion of positive samples. In this study, we used Two Sample Logo to analyze the amino acid enrichment at the first 15 positions of PIP and n-PIP, as shown in Fig. 2A. We found that PIP and n-PIP exhibit different amino acid preferences. PIPs were enriched with serine (S) at positions 1, 2, 7, and 13; leucine (L) at positions 5, 7, 8, 10, 11, 14, and 15; arginine (R) at positions 3 and 9; and phenylalanine (F) at position 4. In contrast, n-PIPs were enriched with aspartic acid (D) at positions 1, 5, 7, 8, and 13 and glycine (G) at positions 10 and 14.

Figure 2B presents the distribution of data lengths: the left shows the length distribution of all data, the middle illustrates the length distribution of the positive samples, and the right displays the length distribution of the negative samples. Most of the sequence lengths were between 15 and 20, and there were no sequence lengths below 10 bp. Therefore, based on the distribution pattern of the data, we can consider that controlling the data length within the range of 5–25 will not exclude important data.

Manual features and dimensionality refinement

To enable the model to learn more valuable features for distinguishing PIPs from n-PIPs, we tested different encoding methods and initially tested their performance in distinguishing PIPs from n-PIPs. Specifically, we used 10 different feature encoding methods commonly used in peptide classification tasks for the training set data: DDE, DPC, AAindex, AAC, GDPC, Composition of K-Spaced Amino Acid Group Pairs (CKSAAGP), Composition, Transition, and Distribution of Codons (CTDC), Generalized Topological Polar Coefficient (GTPC), Composition, Transition, and Distribution of Tripeptides (CTDT), and Pseudo Amino Acid Composition (PAAC). An RF trainer is used to train the models. We compared the performance of these 10 features through five-fold cross-validation. The results are shown in Fig. 3A. To enhance the informativeness of the feature space, we investigated the feature combination strategies. Based on the feature ranking results from Fig. 3A, we incrementally added different types of features starting from a single DDE feature, trained the RF classifiers, and recorded the ACC scores across a five-fold cross-validation. As shown in Fig. 3B, we selected a combination of five features: DDE, DPC, AA index, AAC, and GDPC.

Table 3. Comparison of performance before and after feature selection.

	SN	SP	ACC	AUC	MCC
Before selection	**0.508**	0.822	0.686	0.714	0.350
After selection	**0.508**	**0.830**	**0.691**	**0.718**	**0.360**

Bold values indicate the best performance.

To enhance feature efficiency, we employed an RF model as a feature selector using the training dataset to rank features based on their importance. Faced with the challenge of selecting the optimal number of features for efficiency without compromising feature space integrity, we utilized the RFE method [33]. The RFE method iteratively determined the optimal feature set size, resulting in a streamlined feature set for training. Specifically, because the number of dimensions to be retained for the best RFE performance was uncertain, we adopted a recursive approach. This involved gradually evaluating the ACC obtained using RFE while retaining different feature dimensions. The final results, shown in Fig. 3C, indicated that the optimal feature space was achieved when the feature dimensions were reduced from 1345 to 490. Table 3 presents the comparison results of the five-fold cross-validation of the training set before and after feature selection. Feature selection slightly improved the accuracy of PIP prediction. Although the improvement was not significant, reducing the dimensions not only impacted the model's accuracy but also helped us train the model faster and reduced computational costs.

Subsequently, we analyzed the composition of the retained 490-dimensional features. As shown in Fig. 4A, among the 490 features ultimately retained by the RF-RFE method, the AA index features accounted for the highest proportion, followed by DDE and GDPC. Given the different base quantities of the features, we analyzed the dimensions and proportions of each feature type before and after feature selection, as shown in Table 4. The highest retention ratio was observed for AAC, followed by GDPC and the AA index. This indicates that the initial AAC, GDPC, and AA index features contained a significant amount of useful information that helped us distinguish between PIP and n-PIP. We also performed a SHapley Additive exPlanations (SHAP) analysis on the retained 490-dimensional features. Figure 4B presents the SHAP summary plot for the top 10 features, showing that most of the selected features had a significant impact on the model predictions. We found that the features with the highest SHAP values were mostly the AA index and DDE types, suggesting that the deviation between the observed and expected frequencies of specific dipeptides, as well as the overall composition characteristics of amino acids, also contributed to better distinguishing between proinflammatory peptides and no proinflammatory peptides. Complete information on the selected features

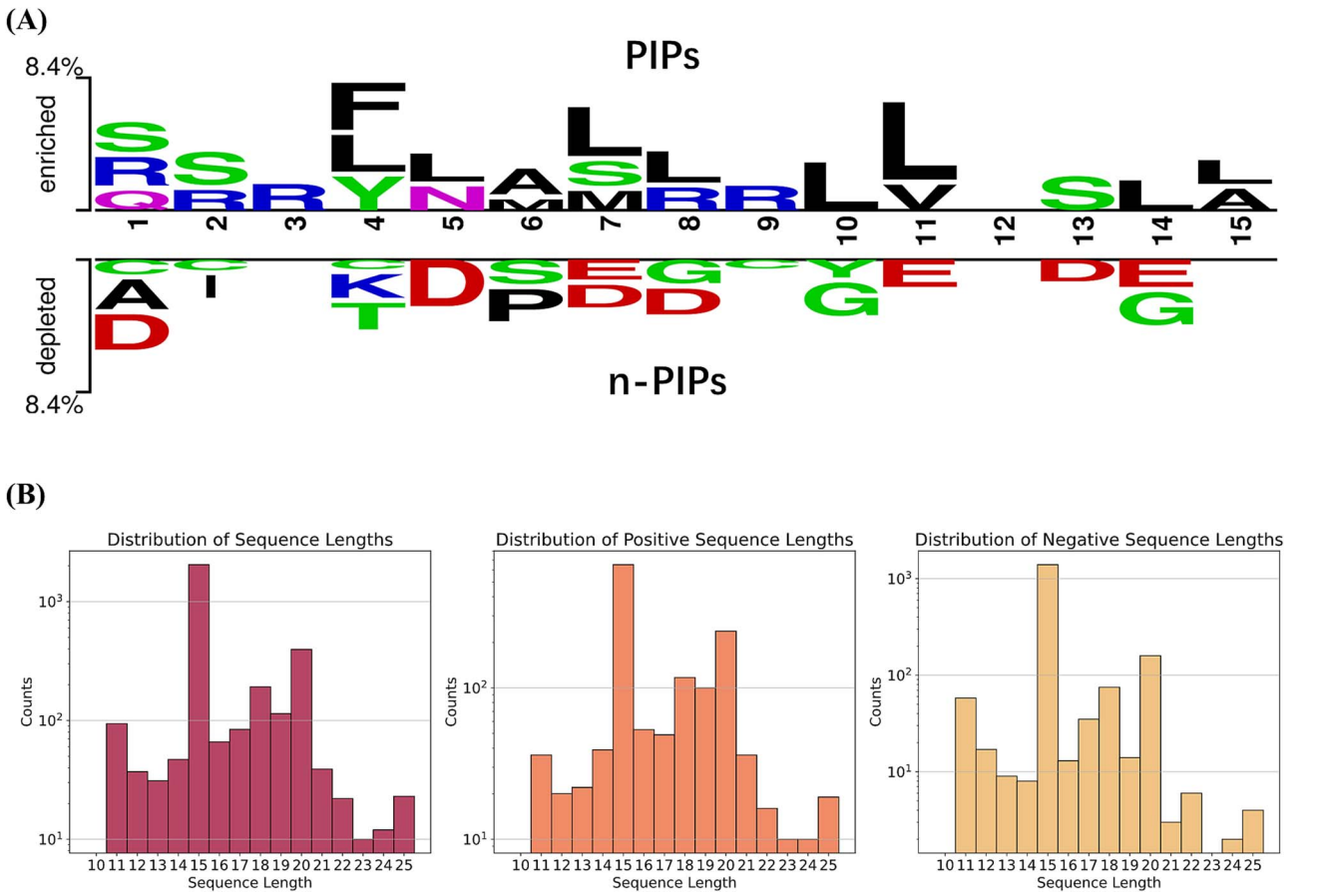

Figure 2. (A) Analysis of the amino acid enrichment of the 15 N-terminal amino acid sequences of PIPs and n-PIPs using Two Sample Logo. The statistical test method used was the *t*-test, with a significance threshold set at $P < .05$. (B) Distribution of sequence lengths (total samples, positive samples, and negative samples).

and full SHAP plots are available on GitHub (https://github.com/ChaoruiYan019/MultiFeatVotPIP).

Comparison of MultiFeatVotPIP with other methods

MultiFeatVotPIP employs a soft-voting algorithm from ensemble learning that integrates the outputs from five algorithms. Before selecting the baseline models, we compared several commonly used machine learning models, including SVM, K-Nearest Neighbors (KNN), Logistic Regression (LR), Stochastic Gradient Descent (SGD), Linear Discriminant Analysis (LDA), Decision Tree, RF, AdaBoost, XGBoost, GBDT, and LightGBM. These models were evaluated using five-fold cross-validation on the training set to compare their AUC performances. As shown in Fig. 5A, the results indicate that GBDT, RF, LightGBM, XGBoost, and AdaBoost performed notably better than the other models. Therefore, these five models were chosen as baseline models for the subsequent ensemble work. To validate the effectiveness of our ensemble strategy on these baseline models, we compared the performances of the five baseline models and the ensemble model using a five-fold cross-validation of the training set. The heat map in Fig. 5B shows a comparison of the results. Our ensemble model achieved the highest values across all evaluation metrics. This improvement can be attributed to the ability of the ensemble model to integrate diverse decision patterns from multiple algorithms, thereby enhancing the overall predictive performance and robustness compared to individual baseline models.

As our model employs a soft voting ensemble strategy, as shown in Fig. 6A, we evaluated different ensemble methods on an independent test set. Notably, the AUC for the hard-voting strategy could not be calculated directly; therefore, it was approximated using the soft-voting AUC. The results demonstrate that our soft voting method generally outperforms other ensemble methods, probably because it considers probabilistic predictions from each base model, leading to more accurate final predictions. By contrast, hard voting and stacking may not effectively capture this level of detail. Owing to the discontinuation of the ProInflam and PIP-EL servers, we compared our model with the ProIn-Fuse SOTA model to validate its performance. Figure 6B presents the results of the comparison between the ProIn-Fuse and MultiFeatVotPIP models, and Table 5 lists the specific performance metrics. Our model showed improvements in the SN, SP, ACC, and MCC, with sensitivity increasing by 1.9%, specificity by 5.4%, accuracy by 3.6%, and MCC by 7.6%.

Deep learning has demonstrated exceptional performance and robust data processing capabilities across multiple domains, and its application in bioinformatics has become increasingly common [45–47]. For example, the team led by Li *et al.* proposed a deep learning framework based on a dual transformer and dual Gated Recurrent Unit (GRU) architecture to predict small secreted peptides in plants [48]. In this study, we explored cutting-edge deep learning models for the analysis of biological sequences. Specifically, we trained six deep learning model frameworks, namely, Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM) [32], BiLSTM [49], Transformer [50], Bert_CNN [51], and

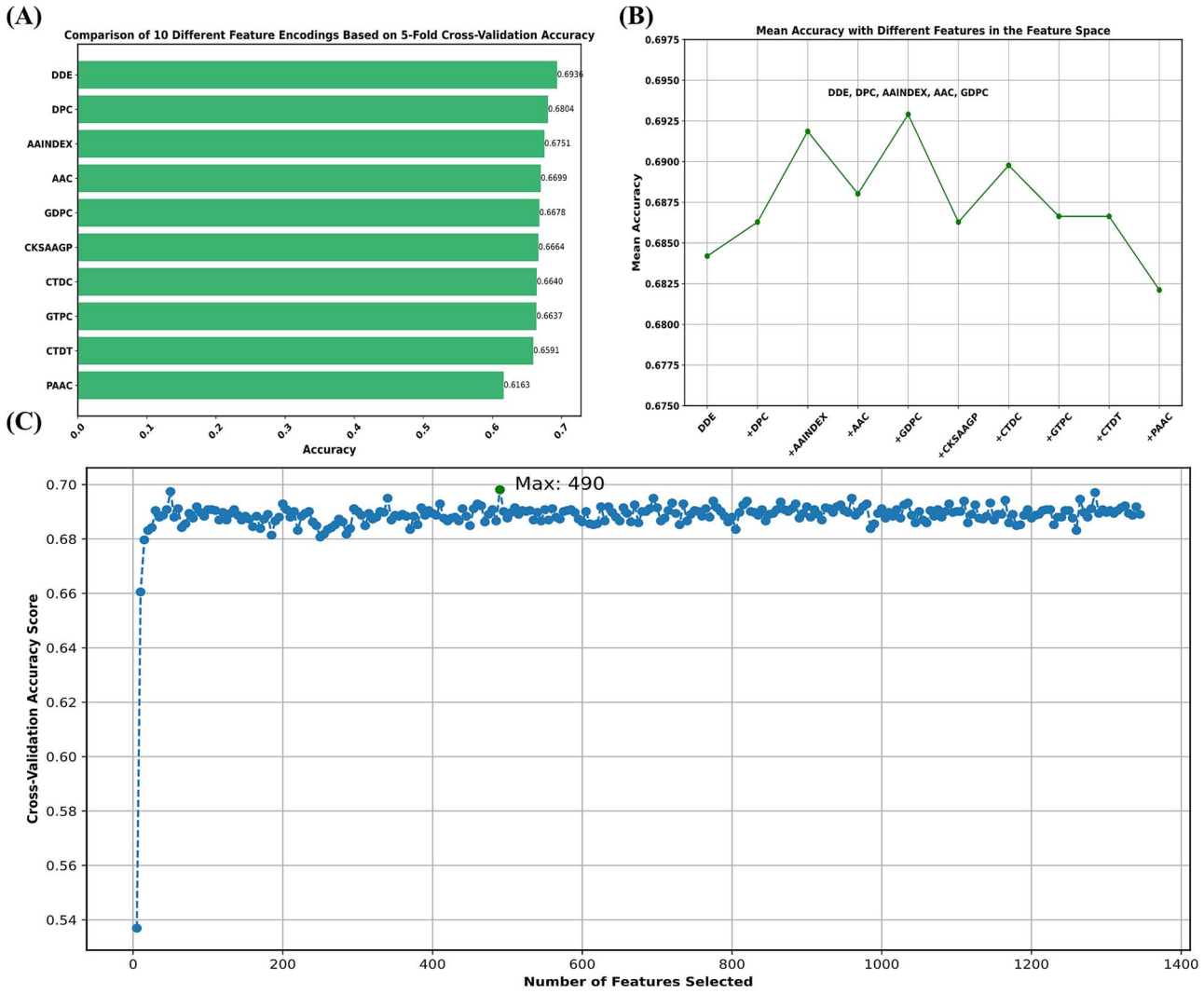

Figure 3. Evaluation of feature encoding methods and feature selection process. (A) Mean accuracy of 10 different feature encoding methods using five-fold cross-validation. (B) Comparison of five-fold cross-validation accuracy for different combinations of feature encodings. (C) Cross-validation accuracy scores for different numbers of retained features using RFE.

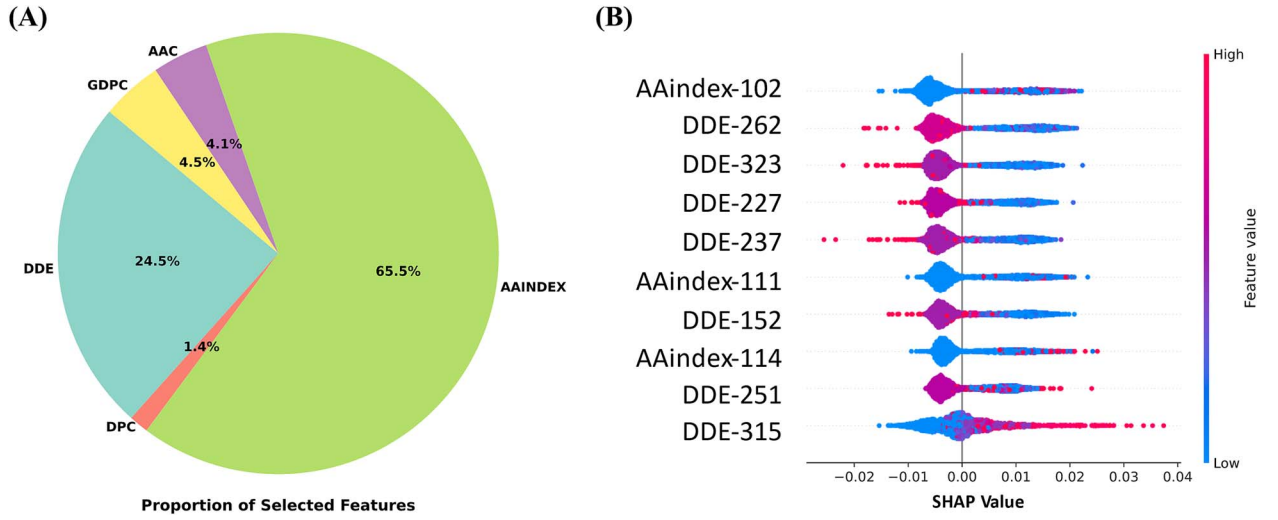

Figure 4. Composition and impact of retained features. (A) Proportion of feature types among the 490-dimensional features retained through RF-RFE. (B) SHAP beeswarm plot showing the impact of the top 20 features on model predictions.

Table 4. Analysis of feature retention.

Features	Original feature dimensions	Selected feature dimensions	Selected/Original feature ratio (%)	Selected/490-dimension space ratio (%)
AAindex	500	321	64.2	65.5
DDE	400	120	30.0	24.5
GDPC	25	22	88.0	4.5
AAC	20	20	100.0	4.1
DPC	400	7	1.75	1.4

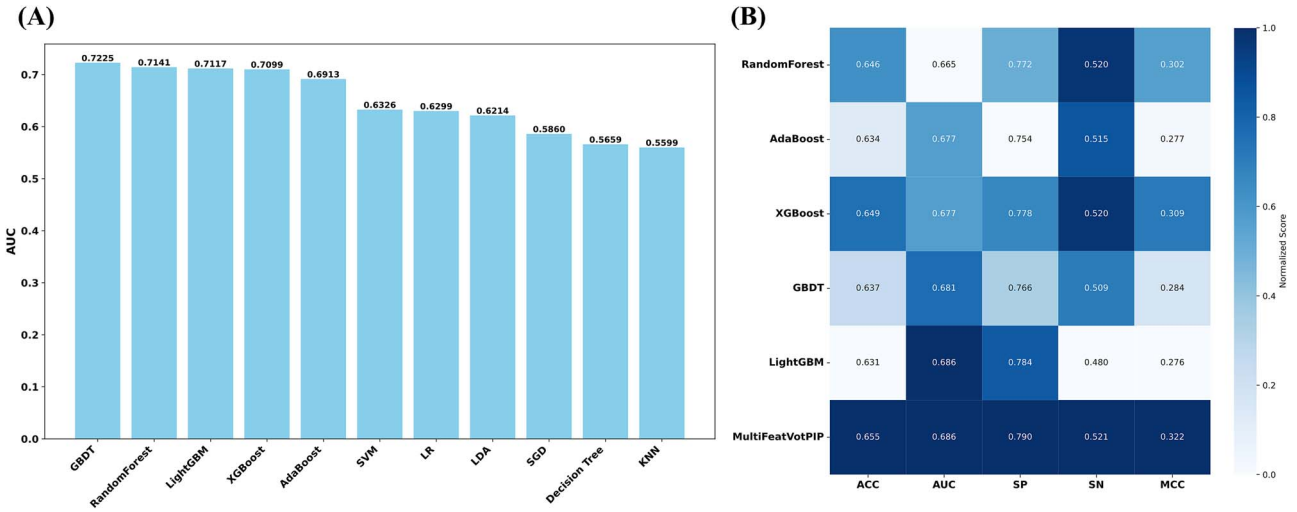

Figure 5. Comparison of baseline models and ensemble models. (A) Bar chart showing AUC performance of various machine learning models on the training set using five-fold cross-validation. The top five models were selected as baseline models. (B) Performance comparison of different ensemble methods on an independent test set. The soft voting method demonstrates superior overall performance.

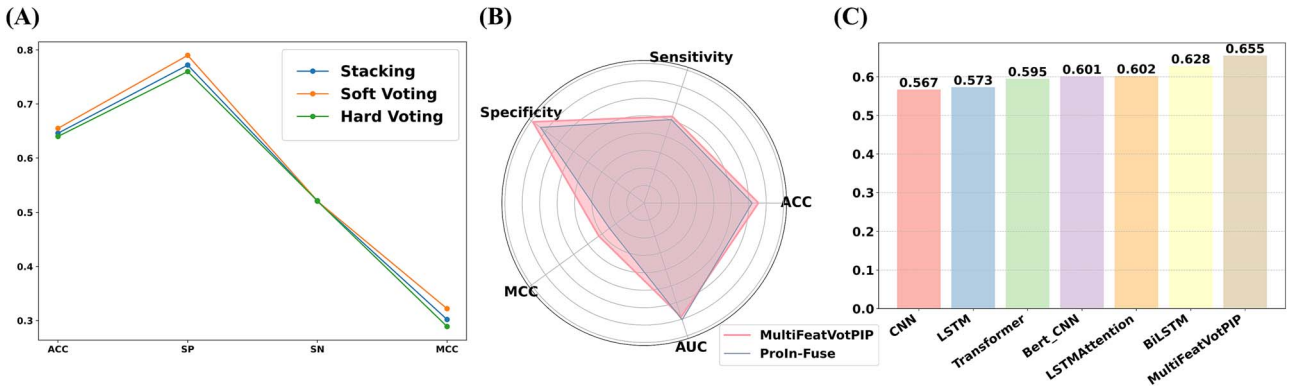

Figure 6. Comparison of ensemble methods and model performance. (A) Line plot comparing the performance of three ensemble methods across various metrics on an independent test set. (B) Radar chart comparing the performance of the MultiFeatVotPIP and ProIn-Fuse models. (C) Performance of six deep learning models on an independent test set.

Table 5. The performance of models PronIn-fuse and MultiFeatVotPIP on independent test sets.

	SN	SP	ACC	AUC	MCC
ProIn-fuse	0.502	0.736	0.619	**0.704**	0.246
MultiFeatVotPIP	**0.521**	**0.790**	**0.655**	0.686	**0.322**

Bold values indicate the best performance.

LSTMAttention [52] with training dataset data, and tested them using an independent test set, obtaining the evaluation results as shown in Fig. 6C. As shown in Fig. 6C, despite the potential of deep learning for high-dimensional data, our comparison of these six deep learning frameworks with traditional machine learning algorithms indicates no distinct advantage of deep learning in PIP classification. We attribute this to two possible reasons. First, deep learning models require larger datasets for training and fine-tuning, and, in cases where the dataset is limited, these models cannot be fine-tuned effectively. Secondly, for PIPs, manually curated features may capture key patterns and characteristics more accurately than automatically extracted features, thus providing a more precise description of the biological functions and characteristics of PIPs. Nonetheless, deep learning holds considerable promise for predicting PIP. First, the performance of the six commonly used deep learning models for sequence prediction is highly sensitive to hyperparameters such as the

number of layers and batch size. Exploring a broader range of hyperparameter configurations can potentially yield improved outcomes. Furthermore, several future research avenues could be pursued to enhance the model's performance. For instance, integrating traditional machine learning with deep learning techniques can be explored, where deep learning models are employed for advanced feature extraction, followed by the fusion of high-performing features, such as the AAindex, and subsequent classification using traditional machine learning models. Moreover, given that deep-learning models typically excel with larger datasets, the increasing availability of validated PIP data suggests that deep-learning approaches are likely to exhibit progressively superior performance in PIP prediction.

Web servers

To facilitate user access, we created a user-friendly web server interface based on MultiFeatVotPIP (http://www.bioai-lab.com/MultiFeatVotPIP). This server offers prediction functionality, allowing users to upload their data for prediction. Additionally, the data used in this study are available on GitHub (https://github.com/ChaoruiYan019/MultiFeatVotPIP), and we downloaded links to the web server for user convenience.

Discussion and conclusion

The onset of inflammation can lead to abnormalities in the body tissues or organs. In this study, datasets were collected from the IEDB database using specific processes and constraints. Subsequently, we propose MultiFeatVotPIP, a method dedicated to identifying PIPs using computational techniques to reduce research costs and accelerate progress. To enhance model performance and computation speed, we employed feature selection methods to reduce the feature dimensions. We compared several baseline models and selected the most suitable one for ensemble learning. We further validated the effectiveness of the MultiFeatVotPIP model integration strategy and compared MultiFeatVotPIP with SOTA models, finding that MultiFeatVotPIP outperformed SOTA models in several aspects, including higher precision and accuracy for positive and negative samples and achieving higher MCC values. In addition, we developed a user-friendly web service based on the MultiFeatVotPIP model, accessible at http://www.bioai-lab.com/MultiFeatVotPIP.

Although the MultiFeatVotPIP designed in this study has already surpassed the SOTA models, there are still some aspects that could be improved. First, the dataset used for training had a minor imbalance issue, with more negative than positive samples collected. However, because this issue was not very prominent, we did not adopt specific methods to resolve it. Second, as deep learning continues to evolve, it has demonstrated exceptional performance and robust data-processing capabilities across various fields, and its application in bioinformatics is becoming increasingly common. Although our preliminary experiments showed that some commonly used deep learning techniques may not be ideal for PIP prediction tasks, we believe that deep learning has great potential for predicting PIPs. Therefore, in future work, we will leverage deep-learning techniques to achieve more accurate PIP predictions.

Key Points

- A novel ensemble learning model, MultiFeatVotPIP, was developed to identify proinflammatory peptides.
- Five types of feature encoding were fused for proinflammatory peptide sequences using a recursive feature elimination strategy combined with RF feature importance to reduce the feature dimensions.
- Soft voting strategy was used to ensemble five different classifiers to enhance predictive performance.
- Achieved superior results compared to state-of-the-art models.
- Provides a user-friendly proinflammatory peptide identification website.

Funding

This study was supported by the National Natural Science Foundation of China (Grant Numbers 62101100 and 62262015).

Conflict of interest: None declared.

Data availability statement

The datasets used and/or analyzed during the current study are available at http://www.bioai-lab.com/MultiFeatVotPIP. Researchers can access the data through this link for further analysis and validation of the results presented in this study.

References

1. Nathan C, Ding AH. Nonresolving inflammation. *Cell* 2010;**140**:871–82. https://doi.org/10.1016/j.cell.2010.02.029.
2. Turner MD, Nedjai B, Hurst T. *et al.* Cytokines and chemokines: At the crossroads of cell signalling and inflammatory disease. *BBA-Mol Cell Res* 2014;**1843**:2563–82. https://doi.org/10.1016/j.bbamcr.2014.05.014.
3. Zhang Z, Cui F, Cao C. *et al.* Single-cell RNA analysis reveals the potential risk of organ-specific cell types vulnerable to SARS-CoV-2 infections. *Comput Biol Med* 2021;**140**:105092. https://doi.org/10.1016/j.compbiomed.2021.105092.
4. de Oliveira CMB, Sakata RK, Issy AM. *et al.* Cytokines and pain. *Braz J Anesthesiol* 2011;**61**:255–65. https://doi.org/10.1016/S0034-7094(11)70029-0.
5. Young JJ, Bruno D, Pomara N. A review of the relationship between proinflammatory cytokines and major depressive disorder. *J Affect Disord* 2014;**169**:15–20. https://doi.org/10.1016/j.jad.2014.07.032.
6. DeFuria J, Belkina AC, Jagannathan-Bogdan M. *et al.* B cells promote inflammation in obesity and type 2 diabetes through regulation of T-cell function and an inflammatory cytokine profile. *Proc Natl Acad Sci* 2013;**110**:5133–8. https://doi.org/10.1073/pnas.1215840110.
7. Xia S, Zhang X, Zheng S. *et al.* An update on Inflamm-aging: mechanisms, prevention, and treatment. *J Immunol Res* 2016;**2016**:1–12. https://doi.org/10.1155/2016/8426874.
8. Gupta S, Mittal P, Madhu MK. *et al.* IL17eScan: a tool for the identification of peptides inducing IL-17 response. *Front Immunol* 2017;**8**:1430. https://doi.org/10.3389/fimmu.2017.01430.
9. Liu R, Zhang Z, Fu X. *et al.* AIPPT: Predicts anti-inflammatory peptides using the most characteristic subset of bases and sequences by stacking ensemble learning strategies. In *Proceedings of the 2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Istanbul, Turkiye, IEEE, 2023, pp. 23–29, https://doi.org/10.1109/BIBM58861.2023.10385565.
10. Gupta S, Madhu MK, Sharma AK. *et al.* ProInflam: a webserver for the prediction of proinflammatory antigenicity of peptides

and proteins. *J Transl Med* 2016;**14**:178. https://doi.org/10.1186/s12967-016-0928-3.
11. Wang Y, Zhai Y, Ding Y. *et al.* SBSM-Pro: support bio-sequence machine for proteins. 2023. arXiv:2308.10275. https://doi.org/10.48550/arXiv.2308.10275.
12. Cui F, Zhang Z, Zou Q. Sequence representation approaches for sequence-based protein prediction tasks that use deep learning. *Brief Funct Genomics* 2021;**20**:61–73. https://doi.org/10.1093/bfgp/elaa030.
13. Manavalan B, Shin TH, Kim MO. *et al.* PIP-EL: a new ensemble learning method for improved proinflammatory peptide predictions. *Front Immunol* 2018;**9**:9. https://doi.org/10.3389/fimmu.2018.01783.
14. Khatun MS, Hasan MM, Shoombuatong W. *et al.* ProIn-Fuse: improved and robust prediction of proinflammatory peptides by fusing of multiple feature representations. *J Comput Aided Mol Des* 2020;**34**:1229–36. https://doi.org/10.1007/s10822-020-00343-9.
15. Zhang HY, Zou Q, Ju Y. *et al.* Distance-based support vector machine to predict DNA N6-methyladenine modification. *Curr Bioinform* 2022;**17**:473–82. https://doi.org/10.2174/1574893617666220404145517.
16. Zhou L, Wang H. A combined feature screening approach of random forest and filter-based methods for ultra-high dimensional data. *Curr Bioinform* 2022;**17**:344–57. https://doi.org/10.2174/1574893617666220221120618.
17. Jiao SH, Zou Q, Guo HN. *et al.* iTTCA-RF: a random forest predictor for tumor T cell antigens. *J Transl Med* 2021;**19**:449. https://doi.org/10.1186/s12967-021-03084-x.
18. Lv H, Yan K, Guo Y. *et al.* Hesham AE-L, Liu B: AMPpred-EL: An effective antimicrobial peptide prediction model based on ensemble learning. *Comput Biol Med* 2022;**146**:105577. https://doi.org/10.1016/j.compbiomed.2022.105577.
19. Yan K, Lv H, Wen J. *et al.* PreTP-stack: prediction of therapeutic peptide based on the stacked ensemble learning. *IEEE/ACM Trans Comput Biol Bioinform* 2023;**20**:1337–44. https://doi.org/10.1109/TCBB.2022.3183018.
20. Dhanda SK, Mahajan S, Paul S. *et al.* IEDB-AR: immune epitope database-analysis resource in 2019. *Nucleic Acids Res* 2019;**47**:W502–6. https://doi.org/10.1093/nar/gkz452.
21. Dinarello CA. Proinflammatory cytokines. *Chest* 2000;**118**:503–8. https://doi.org/10.1378/chest.118.2.503.
22. Zhang J-M, An J. Cytokines, inflammation, and pain. *Int Anesthesiol Clin* 2007;**45**:27–37. https://doi.org/10.1097/AIA.0b013e318034194e.
23. Dinarello CA. Interleukin-18, a proinflammatory cytokine. *Eur Cytokine Netw* 2000;**11**:483–6.
24. Ouyang W, Rutz S, Crellin NK. *et al.* Regulation and functions of the IL-10 family of cytokines in inflammation and disease. *Annu Rev Immunol* 2011;**29**:71–109. https://doi.org/10.1146/annurev-immunol-031210-101312.
25. Chou C-W, Lin F-C, Tsai H-C. *et al.* The importance of pro-inflammatory and anti-inflammatory cytokines in Pneumocystis jirovecii pneumonia. *Med Mycol* 2013;**51**:704–12. https://doi.org/10.3109/13693786.2013.772689.
26. Yan J-w, Wang Y-J, Wj P. *et al.* Therapeutic potential of interleukin-17 in inflammation and autoimmune diseases. *Expert Opin Ther Targets* 2014;**18**:29–41. https://doi.org/10.1517/14728222.2013.843669.
27. Huang Y, Niu B, Gao Y. *et al.* CD-HIT suite: a web server for clustering and comparing biological sequences. *Bioinformatics* 2010;**26**:680–2. https://doi.org/10.1093/bioinformatics/btq003.
28. Chen Z, Zhao P, Li F. *et al.* iLearn: An integrated platform and meta-learner for feature engineering, machine-learning analysis and modeling of DNA, RNA and protein sequence data. *Brief Bioinform* 2019;**21**:1047–57. https://doi.org/10.1093/bib/bbz041.
29. Chen Z, Zhao P, Li C. *et al.* iLearnPlus: a comprehensive and automated machine-learning platform for nucleic acid and protein sequence analysis, prediction and visualization. *Nucleic Acids Res* 2021;**49**:e60–0. https://doi.org/10.1093/nar/gkab122.
30. Fu X, duan H, Zang X. *et al.* Hyb_SEnc: An Antituberculosis peptide predictor based on a hybrid feature vector and stacked ensemble learning. *IEEE/ACM Trans Comput Biol Bioinform* 2024;**PP**:1–17. https://doi.org/10.1109/TCBB.2024.3425644.
31. Tung C-W, Ho S-Y. Computational identification of ubiquitylation sites from protein sequences. *BMC Bioinformatics* 2008;**9**:310. https://doi.org/10.1186/1471-2105-9-310.
32. Chen J, Zou Q, Li J. DeepM6ASeq-EL: prediction of human N6-Methyladenosine (m6A) sites with LSTM and ensemble learning. *Front Comp Sci* 2022;**16**:162302. https://doi.org/10.1007/s11704-020-0180-0.
33. Jiang X, Zhang Y, Li Y. *et al.* Forecast and analysis of aircraft passenger satisfaction based on RF-RFE-LR model. *Sci Rep* 2022;**12**:11174. https://doi.org/10.1038/s41598-022-14566-3.
34. Schapire RE. Explaining AdaBoost. In: *Empirical Inference: Festschrift in Honor of Vladimir N Vapnik*. Schölkopf B, Luo Z, Vovk V. Berlin, Heidelberg: Springer Berlin Heidelberg; 2013, 37–52. https://doi.org/10.1007/978-3-642-41136-6_5.
35. Chen T, Guestrin C: XGBoost: A Scalable Tree Boosting System. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; San Francisco, California, USA. Association for Computing Machinery 2016: 785–794. https://doi.org/10.1145/2939672.2939785.
36. Abbas Z, Rehman, Tayara H. *et al.* Rehman mu, Tayara H, Zou Q, Chong KT: XGBoost framework with feature selection for the prediction of RNA N5-methylcytosine sites. *Mol Ther* 2023;**31**:2543–51. https://doi.org/10.1016/j.ymthe.2023.05.016.
37. Xuan P, Sun C, Zhang T. *et al.* Gradient boosting decision tree-based method for predicting interactions between target genes and drugs. *Front Genet* 2019;**10**:10. https://doi.org/10.3389/fgene.2019.00459.
38. John Lu ZQ. The elements of statistical learning: data mining, inference, and prediction. *Journal of the Royal Statistical Society Series A: Statistics in Society* 2010;**173**:693–4. https://doi.org/10.1111/j.1467-985X.2010.00646_6.x.
39. Zhou Z, Xiao C, Yin J. *et al.* PSAC-6mA: 6mA site identifier using self-attention capsule network based on sequence-positioning. *Comput Biol Med* 2024;**171**:108129. https://doi.org/10.1016/j.compbiomed.2024.108129.
40. Atallah R, Al-Mousa A. Heart disease detection using machine learning majority voting ensemble method. In *Proceedings of the 2019 2nd International Conference on New Trends in Computing Sciences (ICTCS)*, Amman, Jordan, IEEE, 2019, pp. 1–6, https://doi.org/10.1109/ICTCS.2019.8923053.
41. Fu X, Yuan Y, Qiu H. *et al.* AGF-PPIS: a protein–protein interaction site predictor based on an attention mechanism and graph convolutional networks. *Methods* 2024;**222**:142–51. https://doi.org/10.1016/j.ymeth.2024.01.006.
42. Xiao C, Zhou Z, She J. *et al.* PEL-PVP: application of plant vacuolar protein discriminator based on PEFT ESM-2 and bilayer LSTM in an unbalanced dataset. *Int J Biol Macromol* 2024;**277**:134317. https://doi.org/10.1016/j.ijbiomac.2024.134317.
43. Centor RM. Signal detectability: the use of ROC curves and their analyses. *Med Decis Making* 1991;**11**:102–6. https://doi.org/10.1177/0272989X9101100205.

44. Vacic V, Iakoucheva LM, Radivojac P. Two sample logo: a graphical representation of the differences between two sets of sequence alignments. *Bioinformatics (Oxford, England)* 2006;**22**:1536–7. https://doi.org/10.1093/bioinformatics/btl151.
45. Li Y, Wei X, Yang Q. *et al.* msBERT-promoter: a multi-scale ensemble predictor based on BERT pre-trained model for the two-stage prediction of DNA promoters and their strengths. *BMC Biol* 2024;**22**:126. https://doi.org/10.1186/s12915-024-01923-z.
46. Cui F, Li S, Zhang Z. *et al.* DeepMC-iNABP: deep learning for multiclass identification and classification of nucleic acid-binding proteins. *Comput Struct Biotechnol J* 2022;**20**:2020–8. https://doi.org/10.1016/j.csbj.2022.04.029.
47. Jin Q, Cui H, Sun C. *et al.* Domain adaptation based self-correction model for COVID-19 infection segmentation in CT images. *Expert Syst Appl* 2021;**176**:114848. https://doi.org/10.1016/j.eswa.2021.114848.
48. Li Z, Jin J, Wang Y. *et al.* ExamPle: explainable deep learning framework for the prediction of plant small secreted peptides. *Bioinformatics* 2023;**39**:btad108. https://doi.org/10.1093/bioinformatics/btad108.
49. Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF models for sequence tagging. *ArXiv* 2015. https://arxiv.org/abs/1508.01991.
50. Cho K, van Merrienboer B, Bahdanau D, Bengio Y. On the properties of neural machine translation: Encoder-decoder approaches. In Wu D, Carpuat M, Carreras X, Vecchi EM. (eds.) *Proc. SSST@EMNLP 2014, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, Doha, Qatar, 25 October 2014*, 103–111 (Association for Computational Linguistics, 2014). https://aclanthology.org/W14-4012/.
51. Ma J, Cai X, Wei D. *et al.* Aspect-based attention LSTM for aspect-level sentiment analysis. In: *Proceedings of the 2021 3rd World Symposium on Artificial Intelligence (WSAI), 18–20 June 2021, Dalian, China*. IEEE, New York, NY, USA, 2021, pp. 46–50. https://ieeexplore.ieee.org/document/9486323.
52. Safaya A, Abdullatif M, Yuret D. KUISAIL at SemEval-2020 Task 12: BERT-CNN for offensive speech identification in social media. arXiv, 2020. arXiv:2007.13184. https://arxiv.org/abs/2007.13184.