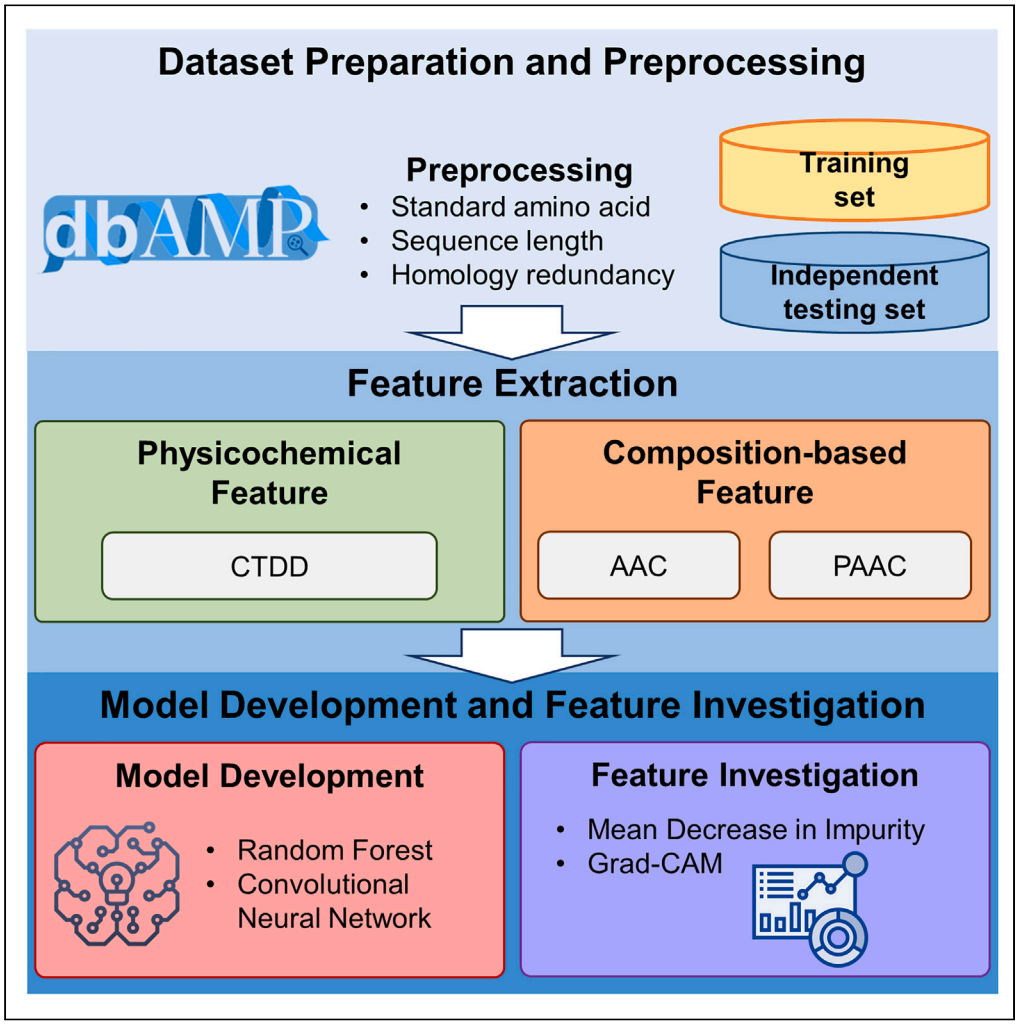
iScience

CellPress
OPEN ACCESS

Article

Multi-label classification and features investigation of antimicrobial peptides with various functional classes



Chia-Ru Chung, Jhen-Ting Liou, Li-Ching Wu, Jorng-Tzong Horng, Tzong-Yi Lee

horng@db.csie.ncu.edu.tw (J.-T.H.)
leetzongyi@nycu.edu.tw (T.-Y.L.)

Highlights

Developed classifiers for multifunction AMP identification

Achieved superior AUC scores in predicting different AMP classes

Identified essential features for classifying multifunction AMPs

Advanced AMP research through feature analysis for multiple functionalities

Chung et al., iScience *26*, 108250
December 15, 2023 © 2023
https://doi.org/10.1016/j.isci.2023.108250

iScience

CellPress
OPEN ACCESS

Article

Multi-label classification and features investigation of antimicrobial peptides with various functional classes

Chia-Ru Chung,[1] Jhen-Ting Liou,[1] Li-Ching Wu,[2] Jorng-Tzong Horng,[1,3,*] and Tzong-Yi Lee[4,5,6,*]

SUMMARY

The challenge of drug-resistant bacteria to global public health has led to increased attention on antimicrobial peptides (AMPs) as a targeted therapeutic alternative with a lower risk of resistance. However, high production costs and limitations in functional class prediction have hindered progress in this field. In this study, we used multi-label classifiers with binary relevance and algorithm adaptation techniques to predict different functions of AMPs across a wide range of pathogen categories, including bacteria, mammalian cells, fungi, viruses, and cancer cells. Our classifiers attained promising AUC scores varying from 0.8492 to 0.9126 on independent testing data. Forward feature selection identified sequence order and charge as critical, with specific amino acids (C and E) as discriminative. These findings provide valuable insights for the design of antimicrobial peptides (AMPs) with multiple functionalities, thus contributing to the broader effort to combat drug-resistant pathogens.

INTRODUCTION

The abuse of antibiotics leading to drug-resistant bacteria has created a global health crisis. The development of new antibiotics has become increasingly challenging, and there is an urgent need to explore alternative therapeutics.[1,2] Antimicrobial peptides (AMPs) are natural compounds that exhibit antimicrobial properties and are potential candidates for drug development. Unlike antibiotics, AMPs do not readily cause resistance.[3,4] They are produced by various living organisms, ranging from microorganisms to humans, and play a crucial role in innate immunity.[5,6] AMPs are typically positively charged, containing cationic and hydrophobic amino acids, and are usually helical polypeptides.[2,7,8] Their antimicrobial activity is rapid, either directly killing microorganisms by disrupting cell membranes or translocating across membranes to act on intracellular targets.[5,9] The electrostatic interaction between the positively charged AMPs and the negatively charged bacterial cells is the primary mechanism of antimicrobial activity.[10,11] AMPs possess various functional classes or activities. These include, but are not limited to, antimicrobial activities against bacteria, viruses, and fungi, as well as antitumor properties and effects on mammalian cells.[12–16] In addition, AMPs also play important roles in immunomodulation, including innate immune defense, inflammation regulation, chemokine induction, and wound healing.[2,11] Yet, the high cost of manufacturing AMPs limits their development.[17] Therefore, detailed investigation and accurate prediction of AMPs' functional classes can effectively reduce manufacturing costs and provide valuable information for developing new drugs.

Recently, considerable efforts have been made to predict the functional classes of AMPs.[18–32] These efforts can be divided into two categories. The first category consists of studies that have focused on the prediction of a single functional class, using what is commonly known as a binary classifier.[20,21,31,32] Such classifiers have advantages in simplifying understanding and illustrating the relationship between features and labels. In addition, a comprehensive review of the various current approaches to AMP identification and the differences among them is provided by Xu et al.[19] However, our primary interest is in the second category, which is the prediction of multiple functional classes. In this case, the classifiers used are known as multi-label classifiers. There are two standard methods for constructing multi-label classifiers: binary relevance and algorithm adaptation.[33] Binary relevance involves building a classifier for every label, which means that the number of classifiers equals the number of labels. This method is advantageous as it considers the relationship between features and labels, similar to binary classifiers. However, it can be relatively slow to build a large number of classifiers. Algorithm adaptation, on the other hand, involves adapting the algorithm to build a multi-label classifier directly or using original algorithms that support multi-label classification. Representative algorithms include multi-label k-nearest neighbor (ML-KNN),[34] collective multi-label classifier (CML),[35] neural network[36] and random forest (RF).[37] By using these methods, it is possible to predict multiple functional classes of AMPs, which can provide valuable information for developing new drugs.

[1]Department of Computer Science and Information Engineering, National Central University, Taoyuan, Taiwan
[2]Department of Biomedical Sciences and Engineering, National Central University, Taoyuan, Taiwan
[3]Department of Bioinformatics and Medical Engineering, Asia University, Taoyuan City, Taiwan
[4]Institute of Bioinformatics and Systems Biology, National Yang Ming Chiao Tung University, Hsinchu City, Taiwan
[5]Center for Intelligent Drug Systems and Smart Biodevices (IDS2B), National Yang Ming Chiao Tung University, Hsinchu City, Taiwan
[6]Lead contact
*Correspondence: horng@db.csie.ncu.edu.tw (J.-T.H.), leetzongyi@nycu.edu.tw (T.-Y.L.)
https://doi.org/10.1016/j.isci.2023.108250

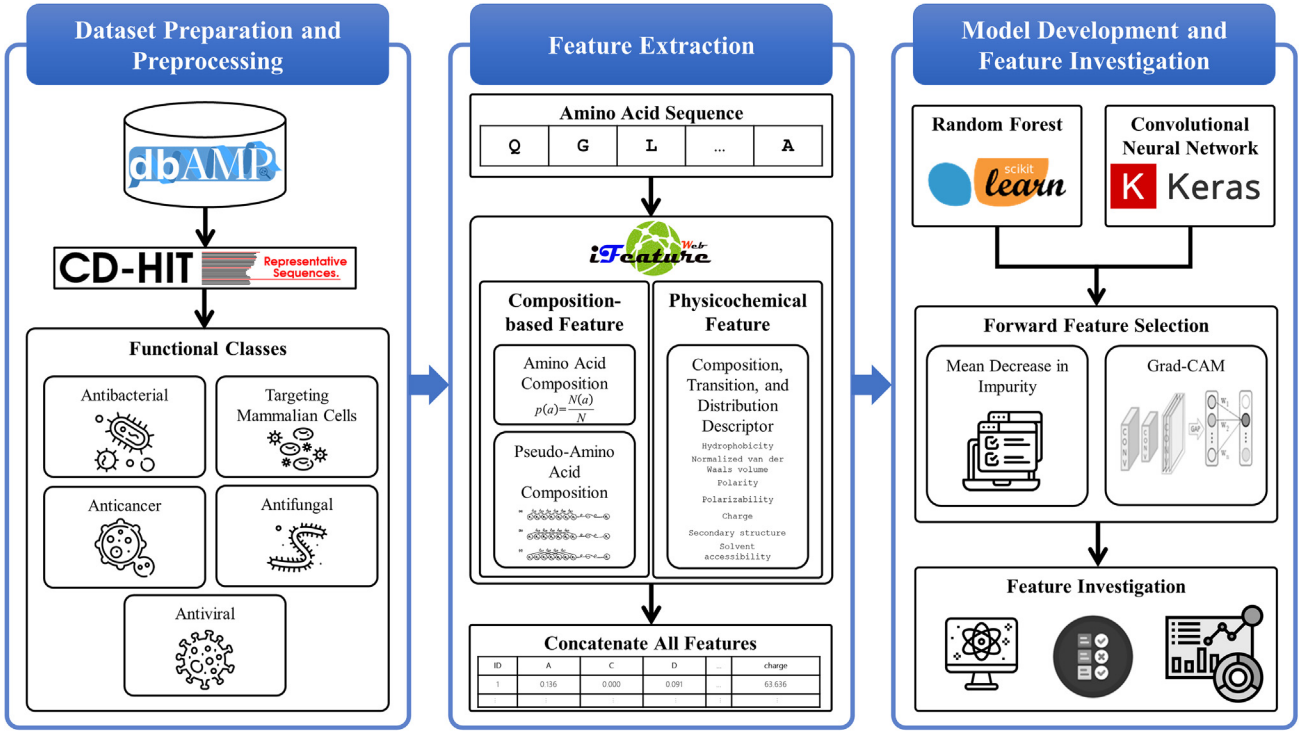


Figure 1. The workflow of this study

First, we retrieved peptide sequences from the reliable source, dbAMP, and subjected it to rigorous preprocessing steps to remove any erroneous or non-qualifying data. Secondly, we extracted essential features, and leveraged machine learning algorithms to build robust models capable of predicting the functional classes of AMPs. The final stage involved feature selection and analysis, aimed at identifying the most relevant and informative features that could be used to investigate the characteristics and properties of AMPs further.

While both binary relevance and algorithm adaptation methods offer distinct advantages, they also have certain limitations. One of the main challenges is the insufficient exploration of the physicochemical properties of AMPs. This aspect, if thoroughly analyzed, could provide significant insights into the identification of AMP functional classes and even enable the design and synthesis of novel AMPs. Several studies have ventured into using binary relevance to predict the functional classes of AMPs.[18,23,26–29,32] Zhang and Li presented Pep-CNN, a deep learning model for predicting therapeutic peptides.[29] This involved constructing eight different binary classification models, each assigned to a different functional class, such as anti-angiogenic, antibacterial, anticancer, anti-inflammatory, antiviral, cell-penetrating, quorum sensing, and surface-binding peptides. Similarly, Yan et al. proposed TPpred-ATMV, an adaptive multi-view model based on a tensor learning framework.[26] The model was developed to predict the same eight functional classes that were addressed in the study by Zhang and Li.[29] Zhang and Zou contributed to the field by proposing an RF prediction method, PPTPP, also designed to identify these eight functions.[23] Tools such as iAMPpred[32] and AMPfun[18] are major contributors to this effort through the incorporation of extensive collections of peptides categorized into different functional classes. They use a variety of peptide features, such as compositional, physicochemical, and structural aspects, as inputs to support vector machine (SVM) algorithms. MultiPep,[38] on the other hand, applies algorithm adaptation to predict AMP functional classes and builds a deep learning multi-label classifier capable of predicting 20 functional classes. The features derived from the convolutional layer help MultiPep outperform other leading peptide bioactivity classifiers in benchmarking multi-label datasets.

Despite these efforts, there is still a lack of research on identifying patterns or critical features for AMPs with several functional classes. Therefore, it is crucial to find effective features through different machine learning methods to understand the characteristics of AMPs and predict their functional classes. The main objective of this study is to investigate and analyze the important and effective features used in different methods, such as binary relevance and algorithm adaptation. By comprehensively analyzing these features, we aim to reconstruct models that outperform existing approaches using a selected set of crucial features that can effectively distinguish functional classes. To achieve our research goals, we followed a systematic workflow as illustrated in Figure 1. First, we applied rigorous preprocessing steps to ensure that only high-quality data were included by obtaining peptide sequence data from the dbAMP database.[39] We then extracted relevant features for each peptide sequence and applied various machine learning algorithms during the model development phase. To improve model performance and interpretability, we performed feature selection, carefully selecting a subset of features for further analysis. This comprehensive approach allowed us to accurately analyze the functional classes of AMPs and identify the critical features essential for this analysis. With this study, we aim to contribute to the advancement of AMP research by providing valuable insights into the characterization

Table 1. Number of peptides in each functional class

Functional class	Training set	Independent testing set
Antibacterial	4685	531
Targeting Mammalian Cells	2514	276
Antifungal	2007	223
Antiviral	977	102
Anticancer	814	90

of multifunctional AMPs. By identifying critical features and refining predictive models, we aim to improve our understanding of AMPs and facilitate the development of effective strategies for their classification and functional annotation.

RESULTS

Basic properties of AMPs

In total, we have collected 6,845 AMPs. Stratified random sampling was used to partition our dataset, which included 6,160 sequences for training and 685 sequences for independent testing. The stratified random sampling method was chosen to ensure that each functional activity was proportionately represented in both the training and independent test sets, effectively maintaining the overall distribution of classes found in the entire dataset. The similar proportions of each functional activity in the training and independent test sets, as demonstrated in

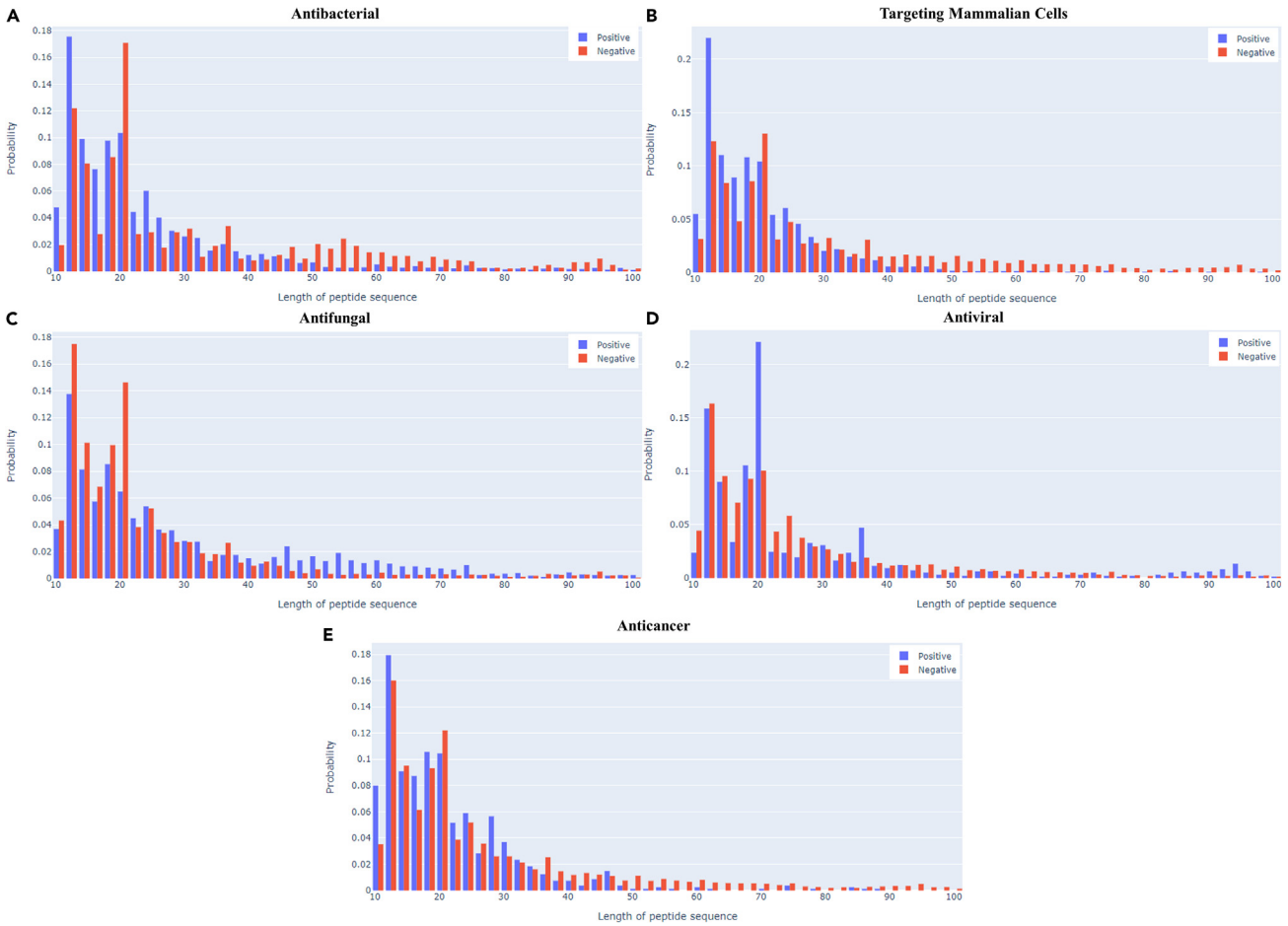


Figure 2. Distribution of sequence length for different functional classes of AMPs

An overview of sequence length distributions for AMPs with different functional classes is presented. The functional classes include (A) Antibacterial, (B) Targeting Mammalian Cells, (C) Antifungal, (D) Antiviral, and (E) Anticancer. Comparing these distributions can provide insights into the length requirements for each functional class, which would be significant for designing different functions of AMPs.

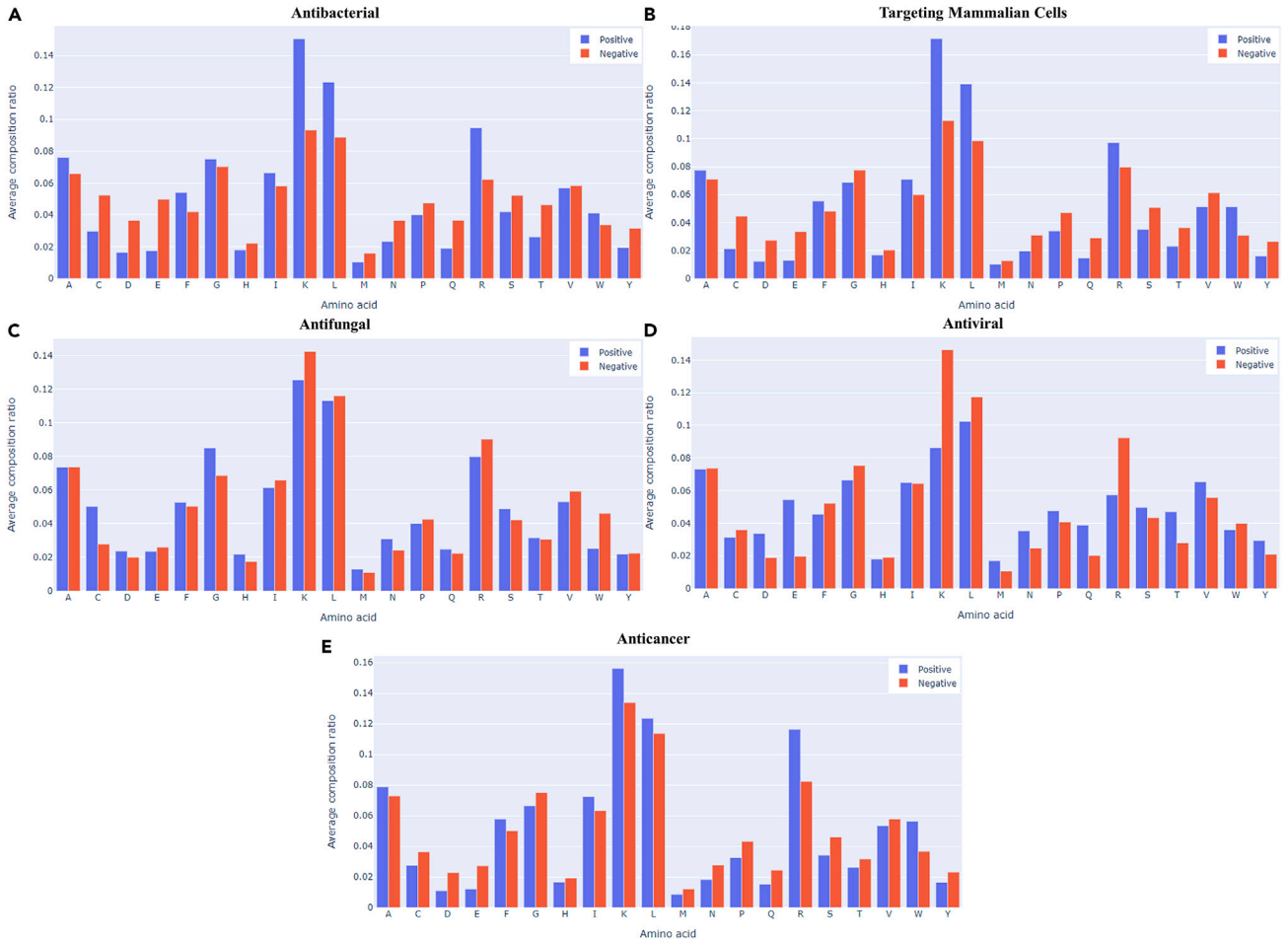


Figure 3. Averages of 20 natural amino acid composition ratios for different functional classes of AMPs

The average ratios of amino acid composition for AMPs with different functional classes are shown in this figure: (A) Antibacterial, (B) Targeting Mammalian Cells, (C) Antifungal, (D) Antiviral, and (E) Anticancer. Understanding these average compositions improves the predictive accuracy of computational models in classifying AMP functions, which is essential for the development of novel AMP-based therapeutics, especially in the fight against drug-resistant pathogens.

Table 1, attest to this strategy's success. The distribution of various properties of peptide sequences (training data) was investigated. Figure 2 depicts the distribution of peptide sequence length for different functional classes; most classes consisted of relatively short peptides, which is consistent with the finding that AMPs are generally short peptides. Considering the perspective of a wider peptide length distribution range, most functional classes showed a higher proportion of positives than negatives for shorter peptide length. In contrast, the proportion of negatives was higher than that of positives in the case of longer peptide length. However, the observation was exceptionally opposite for the antifungal class but less evident for the antiviral class. In addition, the difference between the negative and positive proportions of each class was observed to be higher for the peptide length of about 12 and 20 amino acids. The most significant difference was visible in the antifungal and antiviral classes; the percentage of negative proportions at a peptide length of 20 was increased by nearly 10% and more in antifungal and antiviral classes, respectively.

The average composition ratio of 20 natural amino acids is shown in Figure 3, with K, L, and R observed as the top three proportions in the composition distribution of each functional class. Moreover, the difference between positive and negative was also evident in almost all functional classes; however, the difference was relatively small in antifungal and anticancer classes. In addition, the antibacterial, targeting mammalian cells, and anticancer classes exhibited an enhanced proportion of positives than negatives for the amino acid K with the highest compositional ratio, whereas it was the opposite for antifungal and antiviral classes.

Performances of models and feature selection

Five RF and 5 CNN models with binary relevance, along with 1 RF and 1 CNN model with algorithm adaptation, were constructed with their respective classifiers named RF_binary, CNN_binary, RF_multi, and CNN_multi. Using the 245 features as input, the performance of each functional class is shown in Table 2. In addition, the performance of adaptive methods with the macro-averaged score is presented in

Table 2. 10-fold cross validation on training set

Functional Class	Classifier	Accuracy	Precision	AUC	MCC
Antibacterial	RF_binary	0.8698	0.8767	0.9080	0.6163
	RF_multi	0.8677	0.8688	0.9110	0.6078
	CNN_binary	0.8184	0.8580	0.8355	0.4696
	CNN_multi	0.8159	0.8534	0.8308	0.4587
Targeting Mammalian Cells	RF_binary	0.7795	0.7627	0.8568	0.5382
	RF_multi	0.7781	0.7598	0.8561	0.5348
	CNN_binary	0.7078	0.6527	0.7713	0.3907
	CNN_multi	0.7157	0.6504	0.7781	0.4128
Antifungal	RF_binary	0.7745	0.7736	0.8239	0.4530
	RF_multi	0.7713	0.7826	0.8249	0.4441
	CNN_binary	0.7083	0.5782	0.7162	0.2964
	CNN_multi	0.7018	0.5654	0.7097	0.2719
Antiviral	RF_binary	0.8969	0.8833	0.8988	0.5545
	RF_multi	0.8969	0.9445	0.9018	0.5554
	CNN_binary	0.8753	0.6778	0.8307	0.4709
	CNN_multi	0.8716	0.6654	0.8355	0.4393
Anticancer	RF_binary	0.8964	0.8662	0.8215	0.4364
	RF_multi	0.8938	0.8848	0.8292	0.4132
	CNN_binary	0.8747	0.6871	0.7150	0.2242
	CNN_multi	0.8735	0.6098	0.7136	0.2558

AUC, area under the receiver operating characteristic curve; MCC, Matthew's correlation coefficient.

Table S1. The 6 RF classifiers displayed improved performance than the 6 CNN classifiers in each label; however, the CNN classifiers still showed an acceptable performance in distinguishing functional classes. The reason for this difference in the performance of RF and CNN classifiers could be the high dependency of CNN models on spatial features, with sequential input being the amino acid composition (AAC), pseudo amino acid composition (PAAC), and composition, transition, and distribution descriptor (CTDD)-related features. This indicates that AAC features are mainly affected by the AAC features only, also applicable to PAAC and CTDD features. Still, a specific ACC feature may have a strong correlation with a particular CTDD feature, which has been ignored in the current study. This implies that a meaningful or usefully arranged input for CNN may likely improve the performance.

After building classifiers using the 245 features, the Gini index and gradient-weighted class activation mapping (Grad-CAM) were adopted for computing the importance of features for RF and CNN models, respectively. Because the convolutional layer in the CNN model selects features through the training process itself, feature selection using another method is not necessary for the performance of the CNN model. Therefore, mainly RF classifiers were focused on in this study. The feature importance, also called Gini importance or MDI, was obtained through the RF classifier. To analyze the correlation between features and functional classes directly, only the feature importance of RF_binary classifiers was considered. The top 10 important features for each functional class are shown in Table S2. The difference between antifungal and anticancer classes was relatively small in the top 10 important features, similar to the amino acid composition distribution. Moreover, the amino acid K was not observed in the top 10 important features of antifungal and anticancer classes, whereas it was the third most important feature of other functional classes. Similarly, another amino acid (E) could be noted as the second and most important feature in the antibacterial and antiviral classes, respectively. This result corroborates with the amino acid composition distribution showing a relatively higher proportion of amino acid E in the antibacterial and antiviral classes. Additionally, the percentage (positive or negative) of amino acid E was more than twice compared to other amino acids in all functional classes except the antifungal class. This indicates that although amino acid E appeared only in two functional classes, it is the most important amino acid. Other amino acids in the top 10 important features included L, C, and Q, which appeared in targeting mammalian cells, and antifungal and antiviral classes, respectively. These important features of specific classes could also be observed in the amino acid composition distribution.

In addition to AAC-related features, other feature types provided informative features for each functional class. For the antibacterial class, all the features were charge-based except the PAAC-related amino acids, indicating charge as an important feature for this class. This may also be related to the physicochemical properties of AMPs and their mechanism with bacteria. The two most important features for the class of targeting mammalian cells were PAAC lambda related to the sequence order. In addition, 3 features were charge-related, and a single feature was related to the secondary structure. Thus, the sequence order information might be required for this class to classify effectively compared with the antibacterial class. Besides, the charge feature could also play an essential role, followed by the secondary structure feature. The

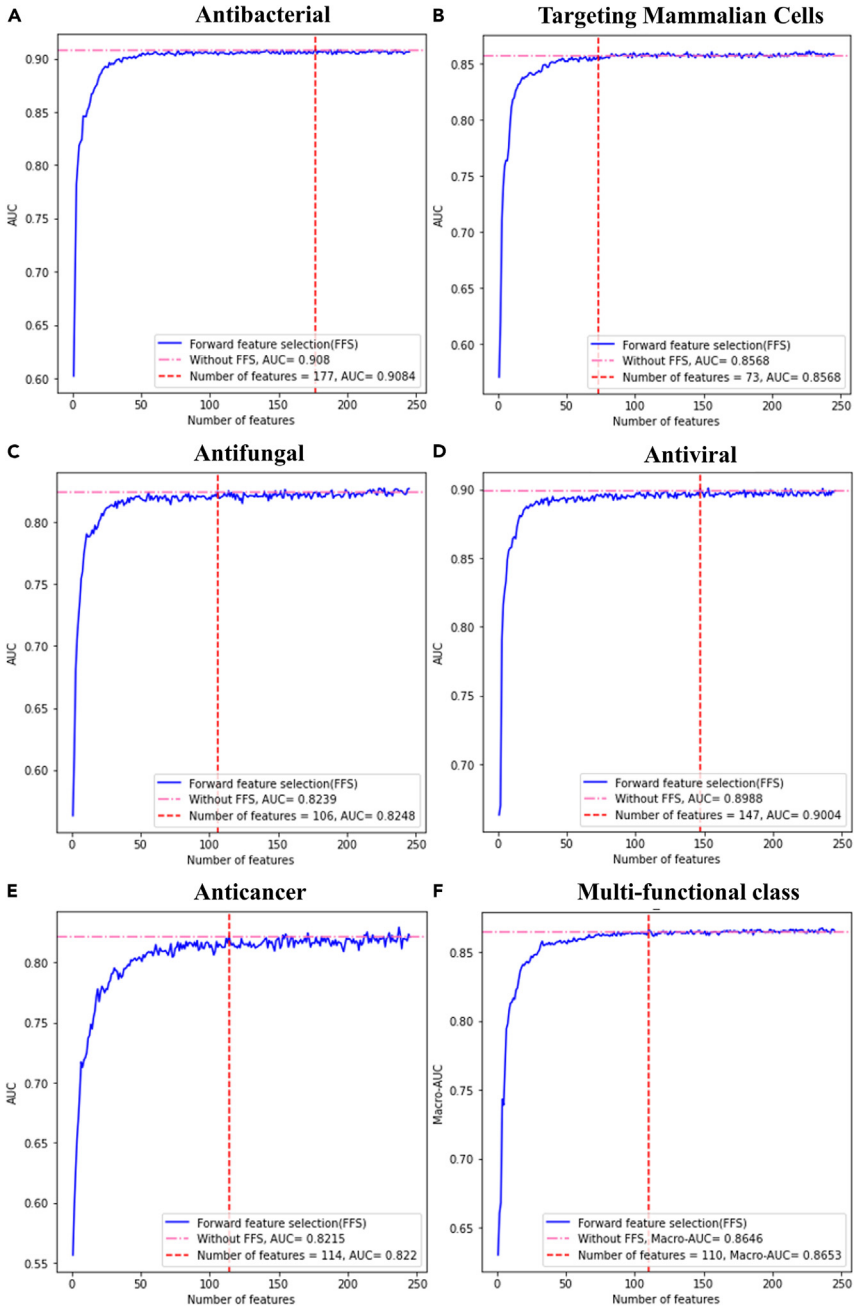


Figure 4. 10-fold cross validation on forward feature selection for different functional classes of AMPs

This illustration shows the AMP classification optimization process using forward feature selection from the first feature to the 245th. Classifiers are selected based on achieving an AUC that is comparable to or exceeds the AUC obtained when all features are used. The functional classes represented are (A) Antibacterial, (B) Mammalian cell targeting, (C) Antifungal, (D) Antiviral, (E) Anticancer, and (F) Multi-functional class.

antifungal class was observed to be significantly different, perhaps having an exceptional existence, from other functional classes with top features, including hydrophobicity and sequence order. However, the charge feature was relatively unimportant among the top 10 features. Similarly, the anticancer class was a comparatively different functional class with the most special composition of the top 10 features comprising 8 sequence-related and 2 hydrophobicity features. Notably, features related to AAC, AAC-PAAC, and charge were not observed. Further, in this study, there are only 10 features related to sequence order, and the lowest importance ranking of these 10 features for anticancer 14, this result shows that anticancer class is strongly related to sequence order. Meanwhile, the composition of the top 10 important features of the antiviral class was similar to the antibacterial class, mainly including AAC, PAAC-related amino acids, and charge.

Table 3. Performance on independent testing data after feature selection

Functional Class	Classifier	Accuracy	Precision	AUC	MCC
Antibacterial	RF_binary	0.8803	0.8904	0.8950	0.6312
	RF_multi	0.8745	0.8790	0.9066	0.6089
Targeting Mammalian Cells	RF_binary	0.7796	0.7551	0.8571	0.5358
	RF_multi	0.7854	0.7529	0.8568	0.5495
Antifungal	RF_binary	0.7927	0.8240	0.8436	0.5026
	RF_multi	0.7927	0.8293	0.8492	0.5029
Antiviral	RF_binary	0.9007	0.8696	0.9065	0.5431
	RF_multi	0.8993	0.9231	0.9126	0.5343
Anticancer	RF_binary	0.8905	0.7419	0.8733	0.3935
	RF_multi	0.8993	0.9200	0.8639	0.4543

AUC, area under the receiver operating characteristic curve; MCC, Matthew's correlation coefficient.

Furthermore, the features were arranged in increasing ranking order from 1 to 245 using forward feature selection to reconstruct classifiers. Note that we chose forward selection for its straightforward nature, simplicity, and ability to sequentially add features that contribute most to the model's predictive performance. It has the advantage of adding features to the model in a sequential manner, which minimizes redundancy and ensures that each feature that is selected makes a significant contribution to the performance of the model. Due to its simplicity and effectiveness, this method has been widely adopted by the bioinformatics community. After feature selection, the classifier was chosen depending on AUC (or macro-AUC), which was equal to or more than the AUC obtained from the classifier constructed using all 245 features. The forward feature selection process for each classifier is shown in Figure 4, and the performance on training data is demonstrated in Tables S3 and S4. Additionally, the performance on independent testing data after feature selection was also determined, as shown in Table 3. The macro-average scores for accuracy, precision, AUC, and MCC for algorithm adaptation methods using RF_multi classifier on independent testing data after feature selection were 0.8502, 0.8609, 0.8778, and 0.5300, respectively. The performance of RF_binary and RF_multi classifiers on training and independent data was observed to be similar and almost the same as before feature selection. Thus, the difference between the performance of binary relevance and algorithm adaptation methods in RF models is insignificant. However, the reconstructed classifiers exhibited similar or enhanced performance after feature selection using these fewer but specific important features to distinguish functional classes effectively. Moreover, the selected features make the classifiers better and more stable, which can be used for further analysis.

Analysis of informative features

In order to gain a comprehensive understanding of the selected features, we were subjected to a thorough analysis based on various aspects. One of the key findings from this analysis was that the majority of the selected features belonged to the CTDD category for all RF classifiers as clearly demonstrated in Table 4. This was mainly due to the fact that CTDD had the highest number of dimensions among all the feature categories which are shown in Table S5. Note that "dimension" refers to the total number of features within each feature type. The CTDD category has 195 features which is the most compared to other feature types, and therefore has the highest number of dimensions. However, it is important to note that the CTDD features should not be underestimated solely based on their dimensions, as they were initially selected as

Table 4. The distribution of features after feature selection

		Types of features		
Functional class	Number of selected features	AAC	PAAC	CTDD
Antibacterial	177	14(8%, 70%)	24(14%, 80%)	139(79%, 71%)
Targeting Mammalian Cells	73	7(10%, 35%)	21(29%, 70%)	45(62%, 23%)
Antifungal	106	6(6%, 30%)	17(16%, 57%)	83(78%, 43%)
Antiviral	147	12(8%, 60%)	24(16%, 80%)	111(76%, 57%)
Anticancer	114	5(4%, 25%)	16(14%, 53%)	93(82%, 48%)
all	110	8(7%, 40%)	24(22%, 80%)	78(71%, 40%)

The first percentage represents the ratio between the number of selected features and the total number of features being considered. The second one refers to the ratio between the number of selected features and the total number of features of a particular type.
AAC, Amino Acid Composition; PAAC, Pseudo Amino Acid Composition; CTDD, Composition, Transition, and Distribution Descriptor.

Table 5. The distribution of common features

Functional class	Number of common features	Number of AAC	Number of PAAC	Number of CTDD
Antibacterial	100(177, 110)	6(6%, 30%)	21(21%, 70%)	73(73%, 37%)
Targeting Mammalian Cells	59(73, 110)	5(8%, 25%)	19(32%, 63%)	35(59%, 18%)
Antifungal	73(106, 110)	5(7%, 25%)	15(21%, 50%)	53(73%, 27%)
Antiviral	94(147, 110)	8(9%, 40%)	21(22%, 70%)	65(69%, 33%)
Anticancer	69(114, 110)	4(6%, 20%)	15(22%, 50%)	50(72%, 26%)

The first value indicates the number of selected features from the RF_binary method that are used to predict a particular functional class, while the second value represents the number of selected features from the RF_multi method. Additionally, two percentages are provided to offer further insight into the significance of these features. The first percentage shows the ratio of the number of selected features to the total number of common features, while the second percentage demonstrates the ratio of the number of selected features to the total number of features of that specific type.
AAC, Amino Acid Composition; PAAC, Pseudo Amino Acid Composition; CTDD, Composition, Transition, and Distribution Descriptor.

important features. In fact, In addition to the CTDD category, the PAAC-related features exhibited the second highest proportion for RF classifiers in terms of specific feature type. This clearly indicates the significance of most of the PAAC-related features, highlighting their potential importance in the overall analysis.

We then conducted an analysis of the composition of selected features by comparing RF_binary and RF_multi classifiers in Table 5. This analysis revealed that the RF_multi classifier comprised some degree of important features selected in each RF_binary classifier. The percentage of commonly selected features divided by the number of selected features for RF_binary classifiers in each functional class was found to be as low as 56% (100/177), indicating that there was a considerable degree of difference between the two classifiers. However, the number of common features appearing in all six RF classifiers was 26, including 2 (8%, 10%) in the AAC category, 11 (42%, 37%) in the PAAC category, and 13 (50%, 7%) in the CTDD category. These 26 common features would be considered as core features since they appeared in all RF classifiers. Among these features, the percentage of PAAC-related features was significantly higher at 37% than other feature types, implying the importance of one-third of PAAC features in all RF classifiers. Interestingly, only 1 feature selected by the RF_multi classifier was not selected by any RF_binary classifiers in the CTDD category, indicating that RF_multi classifiers share many features with RF_binary classifiers. However, the RF_multi classifier may have a different perspective on features based on the association between classes that RF_binary classifiers do not consider. Overall, the study's findings suggest that the RF_multi classifier is a more comprehensive tool for predicting protein-protein interactions.

We conducted an analysis on individual features in addition to the different feature types analyzed earlier. Specifically, we focused on the top 10 important features and evaluated their value distributions in the training dataset across 5 RF_binary classifiers and 1 RF_multi classifier in Figures 4 and 5, respectively. Figure 5 displays the distribution of values for positive and negative samples in RF_binary classifiers, while Figure 6 shows the distribution for subsets in the RF_multi classifier. Our analysis found that the value distributions for some features varied between positive and negative samples in specific functional classes. For example, in the antibacterial class, there were notable differences in the value distributions for PAAC.E, PAAC.C, and PAAC.D. Similarly, in the targeting mammalian cells class, there were differences in the value distributions for charge_C3_100, PAAC.E, and PAAC.Q. In the antifungal class, differences were observed for PAAC.C and amino acid C, while in the antiviral class, amino acid E and PAAC.E showed differences. Interestingly, there was no clear difference between positive and negative data in the anticancer class.

For the RF_multi classifier, we mainly considered subsets with larger samples since subsets with smaller samples may not provide enough information. The distribution of the subset with only antifungal function differed from other subsets, similar to the RF_binary classifiers. Specifically, there were differences in the value distributions for PAAC.C and amino acid C in Figure 6. However, the difference was more evident between subsets, as observed for the subset with antiviral function (amino acid E, PAAC.E, and charge_C1_075). Most samples of the subset having antibacterial and function with targeting mammalian cells were distributed at 0 for amino acids C and E, and PAAC.C and PAAC.E. Furthermore, the distributions of subsets with combinations of antibacterial, targeting mammalian cells, and anticancer function or antibacterial, targeting mammalian cells and antifungal function, or antibacterial, targeting mammalian cells, antifungal and anticancer function were almost the same, except for the charge_C3_001 feature. Notably, the subset acquiring all functions had a relatively higher value for the charge_C1_075 feature. Overall, our analysis suggests that different functional classes may have different value distributions for individual features, and the RF_multi classifier considers the association between classes, which may result in different perspectives on feature importance compared to RF_binary classifiers.

Comparisons with other studies

In this study, we conducted a comprehensive comparison of the RF_multi classifier with the recently studied multi-label classifiers iAMPpred, AMPfun, and MultiPep, for predicting functional classes of AMPs using the independent testing data. To ensure a fair comparison, we excluded the peptide sequences present in MultiPep from our independent testing data. Detailed comparative information on previous studies is provided in Table S6. The performance of the RF_multi classifier was evaluated using the AUC metric, which was the primary focus

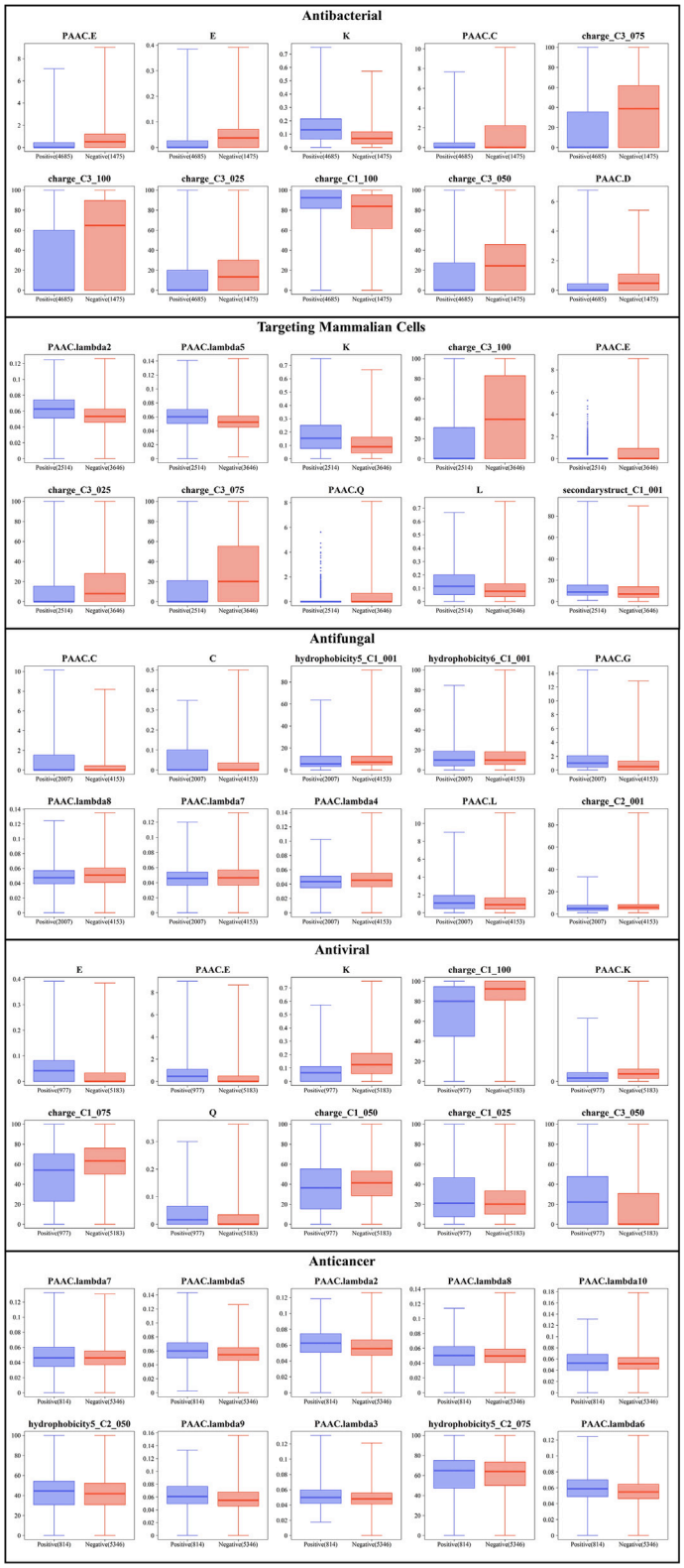


Figure 5. Boxplots of top 10 important features for different functional classes of AMPs

The key features that are most predictive for each functional class of AMPs are highlighted, providing insight into the underlying biology. The identification of these critical features can be used as a basis for the design of more effective AMPs and the improvement of predictive models.

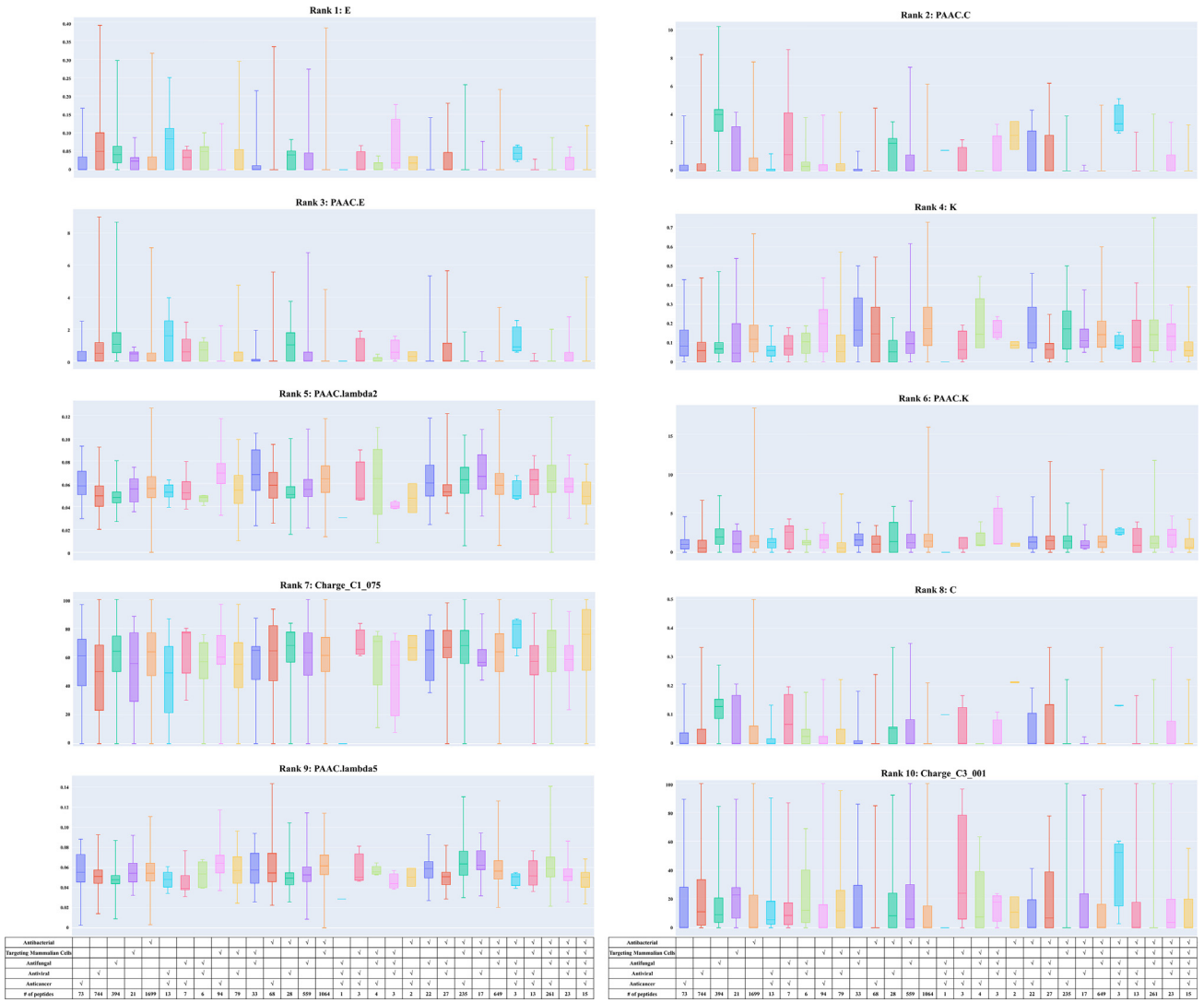


Figure 6. Boxplots of top 10 important features for the multi-functional prediction model

Characteristic distributions for different AMP functional classes are shown in the figure. Distinct distributions can be seen for the antiviral subset and for the amino acids C and E. The all-function subset has uniquely elevated values for charge_C1_075. Compared to binary classifiers, the RF_multi classifier captures nuanced feature importance.

of this study, and the results are presented in Table 6. The RF_multi classifier showed decent performance compared to the previously reported iAMPpred, AMPfun, and MultiPep classifiers for each functional class. Moreover, we compared the performance of the RF_multi classifier with subsets based on the SA metric, which was calculated based on the corresponding functional classes of the compared classifiers, and the results are presented in Table 7. To define each subset, we first considered the specific functional classes targeted by the prediction tools we were comparing, such as iAMPpred. For example, since iAMPpred primarily identifies antibacterial, antiviral, and antifungal peptides, we limited our comparison to these labels, effectively creating a subset. The subsets were defined by the intersection of the functional activities considered by our study and those of the comparison tool. Considering three binary classes (antibacterial, anti-viral, and anti-fungal), the total number of subsets used for comparison is 8. This approach allowed us to make a fair and focused comparison, ensuring that the classes considered were relevant to both our classifier and the one we were benchmarking against. To further adapt the RF_multi classifier with other classifiers, we calculated SA for each functional class. We found that the RF_multi classifier performed better than other classifiers in subsets with large samples, as shown in Figure 7. Notably, all classifiers failed to predict accurately when the number of samples for a specific subset was less than 3. Overall, our results demonstrate the superiority of the RF_multi classifier in predicting functional classes of AMPs, particularly in subsets with a large number of samples.

Table 6. Comparisons of RF_multi (proposed in this study) and previous studies

Functional Class	Classifier	Accuracy	Precision	AUC	MCC
Antibacterial	iAMPpred	0.7328	0.8234	0.6180	0.2211
	AMPfun	0.8277	0.8817	0.8391	0.4946
	MultiPep	0.7755	0.6667	0.8799	0.5377
	RF_multi	0.8745	0.8790	0.9066	0.6089
Targeting Mammalian Cells	AMPfun	0.6511	0.5801	0.6822	0.2577
	RF_multi	0.7854	0.7529	0.8568	0.5495
Antifungal	iAMPpred	0.5358	0.3937	0.6734	0.1994
	AMPfun	0.6526	0.4790	0.7325	0.3416
	MultiPep	0.3061	0	0.3181	−0.3351
	RF_multi	0.7927	0.8293	0.8492	0.5029
Antiviral	iAMPpred	0.3518	0.1166	0.3827	−0.1240
	AMPfun	0.7022	0.3068	0.7945	0.3513
	MultiPep	0.9388	–	0.3478	–
	RF_multi	0.8993	0.9231	0.9126	0.5343
Anticancer	AMPfun	0.6380	0.2238	0.7161	0.2315
	MultiPep	0.9184	0	0.7754	−0.0369
	RF_multi	0.8993	0.9200	0.8639	0.4543

AUC, area under the receiver operating characteristic curve; MCC, Matthew's correlation coefficient. "–" means that the tool did not provide the valid results.

DISCUSSION

In this study, we aimed to create a diverse set of classifiers to predict five different functional classes of AMPs, including antibacterial, antifungal, antiviral, targeting mammalian cells, and anticancer. Both RF and CNN models were used, along with binary relevance and algorithm adaptation methods. A major contribution of this study is our careful forward feature selection process. This process identified critical features such as AAC, PAAC, and CTDD. These features not only improved the performance and stability of our classifiers but also provided valuable biological insights. Importantly, a previous study[19] confirmed the importance of these features in tree-based classifiers, supporting the validity of our findings.

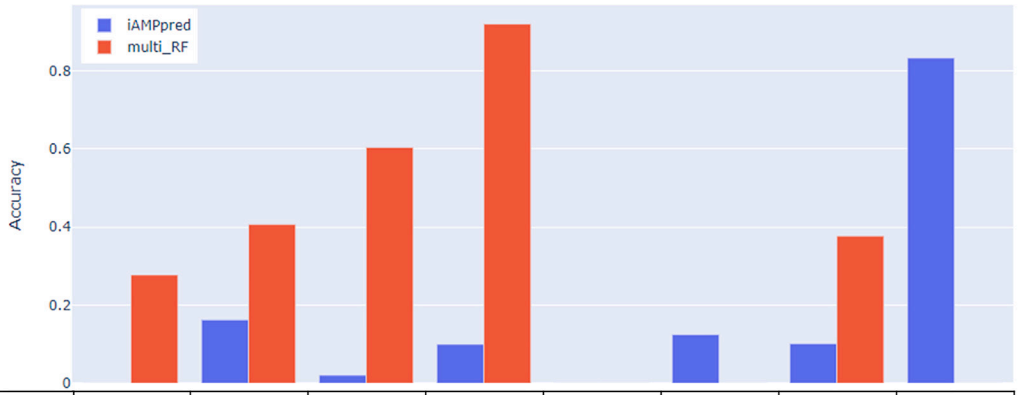
It is important to interpret these results in a nuanced context, although our RF_multi classifier showed superior performance in certain functional subsets. We would caution against viewing our performance metrics as being universally superior to all of the existing models. Direct comparisons are complicated by factors such as the specificity of our test set and the inherent differences in datasets between studies. Our analysis delved into the nuanced behavior of classifiers beyond simple performance metrics. For example, other classifiers we studied also achieved up to 80% accuracy in some subsets. Several factors, including the richness of the dbAMP database and the diverse patterns of newer peptide sequences, which specifically benefited the RF_multi classifier, may account for the difference in performance.

The features that were selected for our study contributed to the performance of the classifiers and provided a nuanced understanding of the mechanisms of AMP. Characteristics associated with AACs, PAACs, and CTDDs, and their distribution among different subgroups, suggest that the biological functions of AMPs are tightly coupled with specific amino acid properties or sequence patterns. These findings can be used to inform the design of new AMPs and to serve as potential targets for further experimental validation.

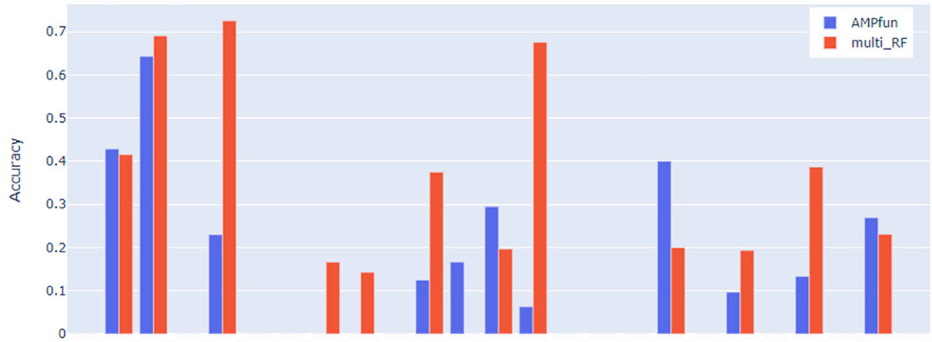
Table 7. Information and performance of subset-related results comparing with previous studies

Functional classes	Number of subsets	Classifier	SA
(Antibacterial, Antifungal, Antiviral)	8	iAMPpred	0.1066
		RF_multi	0.6628
(Antibacterial, Targeting Mammalian Cells, Antifungal, Antiviral, Anticancer)	25	AMPfun	0.2263
		RF_multi	0.4978
(Antibacterial, Antifungal, Antiviral, Anticancer)	7	MultiPep	0.1633
		RF_multi	0.7347

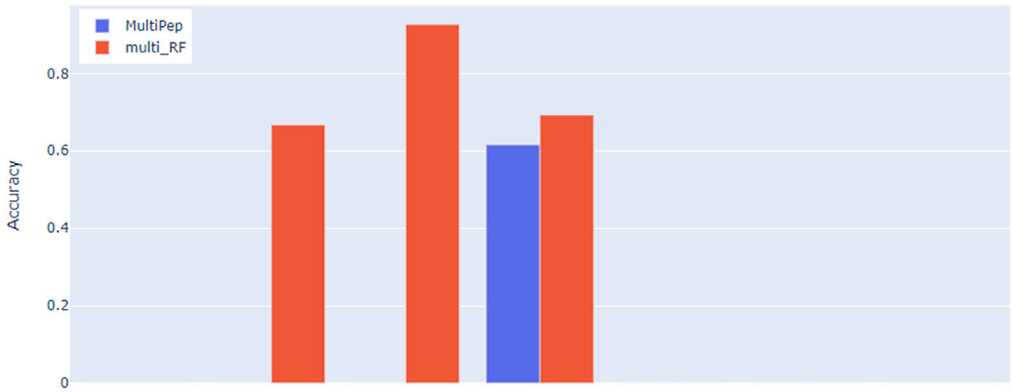
SA, Subset Accuracy.



Antibacterial				√		√	√	√
Antifungal			√		√		√	√
Antiviral		√			√	√		√
# of peptides	18	86	48	350	2	8	167	6



Antibacterial					√						√	√	√	√				√	√	√	√	√	√	√
Targeting Mammalian Cells				√				√	√	√				√	√	√	√			√	√	√		√	√
Antifungal			√				√			√			√			√	√	√	√			√	√	√	√
Antiviral		√				√	√		√			√			√		√		√		√		√		√
Anticancer	√					√		√			√				√	√		√		√			√	√	
# of peptides	3	77	42	3	200	1	1	12	7	4	8	6	61	111	1	2	1	5	4	31	2	75	1	26	1



Antibacterial				√	√	√	√
Antifungal			√			√	√
Antiviral		√					
Anticancer	√				√		√
# of peptides	1	3	27	13	1	3	1

Figure 7. Performance of subset-related results compared with iAMPpred (top), AMPfun (middle), and MultiPep (bottom)
This illustration shows a comparative analysis of our RF_multi classifier with iAMPpred, AMPfun, and MultiPep across 8 defined subsets of functional classes such as antibacterial, antiviral and antifungal. For each functional class, the performance is evaluated based on the SA. RF_multi performs better on larger subsets, while all the others struggle with subsets smaller than 3.

In conclusion, our work makes a significant contribution to the classification of AMPs. Our classifiers showed robust performance within our specific study parameters; however, they are only a small part of a larger spectrum of predictive models. Future research can address limitations and expand the applicability of our models across various conditions.

Limitation of the study

The limitations of the available multi-label classification data for AMPs are acknowledged in our study. Limitations such as incomplete annotation and inherent species biases present challenges that need to be addressed in future research. However, our methodology has been developed to identify generalizable patterns, which are adaptable to new data availability. Comparing our results with other studies such as AMPfun and MultiPep is critical. However, there are several challenges. Direct comparisons are difficult due to differences in functional categories, availability of trained models, and dataset composition between studies. In spite of these limitations, we believe that careful comparative analyses can provide valuable insights and help to direct future research efforts.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead contact
  - Materials availability
  - Data and code availability
- METHOD DETAILS
  - Data source and preprocessing
  - Feature extraction
  - Machine learning models
  - Evaluation metrics
- QUANTIFICATION AND STATISTICAL ANALYSIS

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.isci.2023.108250.

ACKNOWLEDGMENTS

This work was supported by the National Science and Technology Council, Taiwan (NSTC112-2221-E-008-045 and NSTC112-2321-B-A49-016). This work was also financially supported by the Center for Intelligent Drug Systems and Smart Biodevices (IDS2B) from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project and Yushan Young Fellow Program (112C1N084C) by the Ministry of Education (MOE).

AUTHOR CONTRIBUTIONS

J.T.L. carried out the data collection and curation. C.R.C. and J.T.L. participated in the data analyses, model construction, and drafted the manuscript. C.R.C., J.T.L., L.C.W., and T.Y.L. participated in the design of the study and performed the draft revision. J.T.H. and T.Y.L. conceived of the study, and participated in its design and coordination and helped to revise the manuscript. All authors read and approved the final manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: March 4, 2023
Revised: July 15, 2023
Accepted: October 16, 2023
Published: October 18, 2023

REFERENCES

1. Bahar, A.A., and Ren, D. (2013). Antimicrobial peptides. Pharmaceuticals *6*, 1543–1575. https://doi.org/10.3390/ph6121543.
2. Lei, J., Sun, L., Huang, S., Zhu, C., Li, P., He, J., Mackey, V., Coy, D.H., and He, Q. (2019). The antimicrobial peptides and their potential clinical applications. Am. J. Transl. Res. *11*, 3919–3931.
3. Magana, M., Pushpanathan, M., Santos, A.L., Leanse, L., Fernandez, M., Ioannidis, A., Giulianotti, M.A., Apidianakis, Y., Bradfute, S., Ferguson, A.L., et al. (2020). The value of antimicrobial peptides in the age of resistance. Lancet Infect. Dis. *20*, e216–e230. https://doi.org/10.1016/S1473-3099(20)30327-3.
4. Wang, S., Zeng, X., Yang, Q., and Qiao, S. (2016). Antimicrobial Peptides as Potential Alternatives to Antibiotics in Food Animal Industry. Int. J. Mol. Sci. *17*, 603. https://doi.org/10.3390/ijms17050603.
5. Hancock, R.E.W., and Sahl, H.G. (2006). Antimicrobial and host-defense peptides as new anti-infective therapeutic strategies. Nat. Biotechnol. *24*, 1551–1557. https://doi.org/10.1038/nbt1267.
6. Steinstraesser, L., Kraneburg, U., Jacobsen, F., and Al-Benna, S. (2011). Host defense peptides and their antimicrobial-immunomodulatory duality. Immunobiology *216*, 322–333. https://doi.org/10.1016/j.imbio.2010.07.003.
7. Jenssen, H., Hamill, P., and Hancock, R.E.W. (2006). Peptide antimicrobial agents. Clin. Microbiol. Rev. *19*, 491–511. https://doi.org/10.1128/CMR.00056-05.
8. Pasupuleti, M., Schmidtchen, A., and Malmsten, M. (2012). Antimicrobial peptides: key components of the innate immune system. Crit. Rev. Biotechnol. *32*, 143–171. https://doi.org/10.3109/07388551.2011.594423.
9. Schneider, V.A.F., Coorens, M., Ordonez, S.R., Tjeerdsma-van Bokhoven, J.L.M., Posthuma, G., van Dijk, A., Haagsman, H.P., and Veldhuizen, E.J.A. (2016). Imaging the antimicrobial mechanism(s) of cathelicidin-2. Sci. Rep. *6*, 32948. https://doi.org/10.1038/srep32948.
10. Ebenhan, T., Gheysens, O., Kruger, H.G., Zeevaart, J.R., and Sathekge, M.M. (2014). Antimicrobial peptides: their role as infection-selective tracers for molecular imaging. BioMed Res. Int. *2014*, 867381. https://doi.org/10.1155/2014/867381.
11. Yeung, A.T.Y., Gellatly, S.L., and Hancock, R.E.W. (2011). Multifunctional cationic host defence peptides and their clinical applications. Cell. Mol. Life Sci. *68*, 2161–2176. https://doi.org/10.1007/s00018-011-0710-x.
12. Boman, H.G. (2003). Antibacterial peptides: basic facts and emerging concepts. J. Intern. Med. *254*, 197–215. https://doi.org/10.1046/j.1365-2796.2003.01228.x.
13. Ling, R., Dai, Y., Huang, B., Huang, W., Yu, J., Lu, X., and Jiang, Y. (2020). In silico design of antiviral peptides targeting the spike protein of SARS-CoV-2. Peptides *130*, 170328. https://doi.org/10.1016/j.peptides.2020.170328.
14. van der Weerden, N.L., Bleackley, M.R., and Anderson, M.A. (2013). Properties and mechanisms of action of naturally occurring antifungal peptides. Cell. Mol. Life Sci. *70*, 3545–3570. https://doi.org/10.1007/s00018-013-1260-1.
15. Chiangjong, W., Chutipongtanate, S., and Hongeng, S. (2020). Anticancer peptide: Physicochemical property, functional aspect and trend in clinical application (Review). Int. J. Oncol. *57*, 678–696. https://doi.org/10.3892/ijo.2020.5099.
16. Javadpour, M.M., Juban, M.M., Lo, W.C., Bishop, S.M., Alberty, J.B., Cowell, S.M., Becker, C.L., and McLaughlin, M.L. (1996). De novo antimicrobial peptides with low mammalian cell toxicity. J. Med. Chem. *39*, 3107–3113. https://doi.org/10.1021/jm9509410.
17. Moretta, A., Scieuzo, C., Petrone, A.M., Salvia, R., Manniello, M.D., Franco, A., Lucchetti, D., Vassallo, A., Vogel, H., Sgambato, A., and Falabella, P. (2021). Antimicrobial Peptides: A New Hope in Biomedical and Pharmaceutical Fields. Front. Cell. Infect. Microbiol. *11*, 668632. https://doi.org/10.3389/fcimb.2021.668632.
18. Chung, C.R., Kuo, T.R., Wu, L.C., Lee, T.Y., and Horng, J.T. (2019). Characterization and identification of antimicrobial peptides with different functional activities. Brief. Bioinform. *21*, 1098–1114. https://doi.org/10.1093/bib/bbz043.
19. Xu, J., Li, F., Leier, A., Xiang, D., Shen, H.H., Marquez Lago, T.T., Li, J., Yu, D.J., and Song, J. (2021). Comprehensive assessment of machine learning-based methods for predicting antimicrobial peptides. Brief. Bioinform. *22*, bbab083. https://doi.org/10.1093/bib/bbab083.
20. Pang, Y., Yao, L., Jhong, J.H., Wang, Z., and Lee, T.Y. (2021). AVPIden: a new scheme for identification and functional prediction of antiviral peptides based on machine learning approaches. Brief. Bioinform. *22*, bbab263. https://doi.org/10.1093/bib/bbab263.
21. Pang, Y., Wang, Z., Jhong, J.H., and Lee, T.Y. (2021). Identifying anti-coronavirus peptides by incorporating different negative datasets and imbalanced learning strategies. Brief. Bioinform. *22*, 1085–1095. https://doi.org/10.1093/bib/bbaa423.
22. Pang, Y., Yao, L., Xu, J., Wang, Z., and Lee, T.Y. (2022). Integrating transformer and imbalanced multi-label learning to identify antimicrobial peptides and their functional activities. Bioinformatics *38*, 5368–5374. https://doi.org/10.1093/bioinformatics/btac711.
23. Zhang, Y.P., and Zou, Q. (2020). PPTPP: a novel therapeutic peptide prediction method using physicochemical property encoding and adaptive feature representation learning. Bioinformatics *36*, 3982–3987. https://doi.org/10.1093/bioinformatics/btaa275.
24. Xiao, X., Shao, Y.-T., Cheng, X., and Stamatovic, B. (2021). iAMP-CA2L: a new CNN-BiLSTM-SVM classifier based on cellular automata image for identifying antimicrobial peptides and their functional types. Brief. Bioinform. *22*, bbab209. https://doi.org/10.1093/bib/bbab209.
25. Tang, W., Dai, R., Yan, W., Zhang, W., Bin, Y., Xia, E., and Xia, J. (2022). Identifying multi-functional bioactive peptide functions using multi-label deep learning. Brief. Bioinform. *23*, bbab414. https://doi.org/10.1093/bib/bbab414.
26. Yan, K., Lv, H., Guo, Y., Chen, Y., Wu, H., and Liu, B. (2022). TPpred-ATMV: therapeutic peptide prediction by adaptive multi-view tensor learning model. Bioinformatics *38*, 2712–2718. https://doi.org/10.1093/bioinformatics/btac200.
27. Xu, J., Li, F., Li, C., Guo, X., Landersdorfer, C., Shen, H.-H., Peleg, A.Y., Li, J., Imoto, S., Yao, J., et al. (2023). iAMPCN: a deep-learning approach for identifying antimicrobial peptides and their functional activities. Brief. Bioinform. *24*, bbad240. https://doi.org/10.1093/bib/bbad240.
28. Du, Z., Ding, X., Xu, Y., and Li, Y. (2023). UniDL4BioPep: a universal deep learning architecture for binary classification in peptide bioactivity. Brief. Bioinform. *24*, bbad135. https://doi.org/10.1093/bib/bbad135.
29. Zhang, S., and Li, X. (2022). Pep-CNN: An improved convolutional neural network for predicting therapeutic peptides. Chemometr. Intell. Lab. Syst. *221*, 104490. https://doi.org/10.1016/j.chemolab.2022.104490.
30. Lin, Y., Cai, Y., Liu, J., Lin, C., and Liu, X. (2019). An advanced approach to identify antimicrobial peptides and their function types for penaeus through machine learning strategies. BMC Bioinf. *20*, 291. https://doi.org/10.1186/s12859-019-2766-9.
31. Boopathi, V., Subramaniyam, S., Malik, A., Lee, G., Manavalan, B., and Yang, D.C. (2019). mACPpred: A Support Vector Machine-Based Meta-Predictor for Identification of Anticancer Peptides. Int. J. Mol. Sci. *20*, 1964. https://doi.org/10.3390/ijms20081964.
32. Meher, P.K., Sahu, T.K., Saini, V., and Rao, A.R. (2017). Predicting antimicrobial peptides with improved accuracy by incorporating the compositional, physico-chemical and structural features into Chou's general PseAAC. Sci. Rep. *7*, 42362. https://doi.org/10.1038/srep42362.
33. Zhang, M.-L., and Zhou, Z.-H. (2014). A review on multi-label learning algorithms. IEEE Trans. Knowl. Data Eng. *26*, 1819–1837.
34. Zhang, M.-L., and Zhou, Z.-H. (2007). ML-KNN: A lazy learning approach to multi-label learning. Pattern Recogn. *40*, 2038–2048. https://doi.org/10.1016/j.patcog.2006.12.019.
35. Ghamrawi, N., and McCallum, A. (2005). Collective Multi-Label Classification, pp. 195–200.
36. Dayhoff, J.E. (1990). Neural Network Architectures: An Introduction (Van Nostrand Reinhold Co).
37. Biau, G., and Scornet, E. (2016). A random forest guided tour. Test *25*, 197–227.
38. Grønning, A.G.B., Kacprowski, T., and Schéele, C. (2021). MultiPep: a hierarchical deep learning approach for multi-label classification of peptide bioactivities. Biol. Methods Protoc. *6*, bpab021. https://doi.org/10.1093/biomethods/bpab021.
39. Jhong, J.H., Yao, L., Pang, Y., Li, Z., Chung, C.R., Wang, R., Li, S., Li, W., Luo, M., Ma, R., et al. (2022). dbAMP 2.0: updated resource for antimicrobial peptides with an enhanced scanning method for genomic and proteomic data. Nucleic Acids Res. *50*, D460–D470. https://doi.org/10.1093/nar/gkab1080.
40. Chen, Z., Zhao, P., Li, F., Leier, A., Marquez-Lago, T.T., Wang, Y., Webb, G.I., Smith, A.I., Daly, R.J., Chou, K.C., and Song, J. (2018). iFeature: a Python package and web server for features extraction and selection from protein and peptide sequences. Bioinformatics *34*, 2499–2502. https://doi.org/10.1093/bioinformatics/bty140.

41. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res. *12*, 2825–2830.
42. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., and Isard, M. (2016). {TensorFlow}: A System for {Large-Scale} Machine Learning, pp. 265–283.
43. Wang, G., Li, X., and Wang, Z. (2016). APD3: the antimicrobial peptide database as a tool for research and education. Nucleic Acids Res. *44*, D1087–D1093. https://doi.org/10.1093/nar/gkv1278.
44. Pirtskhalava, M., Amstrong, A.A., Grigolava, M., Chubinidze, M., Alimbarashvili, E., Vishnepolsky, B., Gabrielian, A., Rosenthal, A., Hurt, D.E., and Tartakovsky, M. (2021). DBAASP v3: database of antimicrobial/cytotoxic activity and structure of peptides as a resource for development of new therapeutics. Nucleic Acids Res. *49*, D288–D297. https://doi.org/10.1093/nar/gkaa991.
45. Thomas, S., Karnik, S., Barai, R.S., Jayaraman, V.K., and Idicula-Thomas, S. (2010). CAMP: a useful resource for research on antimicrobial peptides. Nucleic Acids Res. *38*, D774–D780. https://doi.org/10.1093/nar/gkp1021.
46. Zhao, X., Wu, H., Lu, H., Li, G., and Huang, Q. (2013). LAMP: A Database Linking Antimicrobial Peptides. PLoS One *8*, e66557. https://doi.org/10.1371/journal.pone.0066557.
47. Lee, H.T., Lee, C.C., Yang, J.R., Lai, J.Z.C., and Chang, K.Y. (2015). A large-scale structural classification of antimicrobial peptides. BioMed Res. Int. *2015*, 475062. https://doi.org/10.1155/2015/475062.
48. UniProt Consortium (2021). UniProt: the universal protein knowledgebase in 2021. Nucleic Acids Res. *49*, D480–D489. https://doi.org/10.1093/nar/gkaa1100.
49. Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C.L., Ma, J., and Fergus, R. (2021). Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proc. Natl. Acad. Sci. USA *118*, e2016239118. https://doi.org/10.1073/pnas.2016239118.
50. Suzek, B.E., Huang, H., McGarvey, P., Mazumder, R., and Wu, C.H. (2007). UniRef: comprehensive and non-redundant UniProt reference clusters. Bioinformatics *23*, 1282–1288. https://doi.org/10.1093/bioinformatics/btm098.
51. Fu, L., Niu, B., Zhu, Z., Wu, S., and Li, W. (2012). CD-HIT: accelerated for clustering the next-generation sequencing data. Bioinformatics *28*, 3150–3152. https://doi.org/10.1093/bioinformatics/bts565.
52. Bhasin, M., and Raghava, G.P.S. (2004). Classification of nuclear receptors based on amino acid composition and dipeptide composition. J. Biol. Chem. *279*, 23262–23266. https://doi.org/10.1074/jbc.M401932200.
53. Chou, K.C. (2005). Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. Bioinformatics *21*, 10–19. https://doi.org/10.1093/bioinformatics/bth466.
54. Chou, K.C. (2001). Prediction of protein cellular attributes using pseudo-amino acid composition. Proteins *43*, 246–255. https://doi.org/10.1002/prot.1035.
55. Cai, C.Z., Han, L.Y., Ji, Z.L., Chen, X., and Chen, Y.Z. (2003). SVM-Prot: Web-based support vector machine software for functional classification of a protein from its primary sequence. Nucleic Acids Res. *31*, 3692–3697. https://doi.org/10.1093/nar/gkg600.
56. Cai, C.Z., Han, L.Y., Ji, Z.L., and Chen, Y.Z. (2004). Enzyme family classification by support vector machines. Proteins *55*, 66–76. https://doi.org/10.1002/prot.20045.
57. Dubchak, I., Muchnik, I., Holbrook, S.R., and Kim, S.H. (1995). Prediction of protein folding class using global description of amino acid sequence. Proc. Natl. Acad. Sci. USA *92*, 8700–8704. https://doi.org/10.1073/pnas.92.19.8700.
58. Dubchak, I., Muchnik, I., Mayor, C., Dralyuk, I., and Kim, S.H. (1999). Recognition of a protein fold in the context of the Structural Classification of Proteins (SCOP) classification. Proteins *35*, 401–407.
59. Han, L.Y., Cai, C.Z., Lo, S.L., Chung, M.C.M., and Chen, Y.Z. (2004). Prediction of RNA-binding proteins from primary sequence by a support vector machine approach. RNA *10*, 355–368. https://doi.org/10.1261/rna.5890304.
60. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization, pp. 618–626.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited data		
Dataset used in this study	Jhong et al.[39]	https://awi.cuhk.edu.cn/dbAMP/
Software and algorithms		
Python version 3.8.12	Python Software Foundation	https://www.python.org
iFeature	Chen et al.[40]	https://ifeature.erc.monash.edu/
scikit-learn version 1.0.1	Pedregosa et al.[41]	https://scikit-learn.org/stable/
Tensorflow version 2.3.0	Abadi et al.[42]	https://pypi.org/project/tensorflow/

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Prof. Tzong-Yi Lee (leetzongyi@nycu.edu.tw).

Materials availability

No new unique reagents were generated in this study.

Data and code availability

The datasets used in this study are available on Github: https://github.com/chungcr/multiAMP. We have also shared the key codes. Additional information to reanalyze the data reported in this paper is available from the lead contact.

METHOD DETAILS

Data source and preprocessing

Various AMP-related databases have been reported recently, providing detailed information, such as physicochemical properties and functional classes of AMPs. In this study, we chose dbAMP[39] as the data source because this database is comparatively newer, with data collected from other standard databases also, such as APD3[43], DBAASP,[44] CAMP,[45] LAMP,[46] ADAM,[47] UniprotKB/Swiss-Prot[48] and several other integrated resources. Thus, the dbAMP database possesses more sufficient and diverse data with a collection of peptides for different functional classes of AMPs (Table S7).

In this study, our research focused on the identification and analysis of peptide sequences obtained from dbAMP. Note that all data were retrieved from the dbAMPv2 on March 23, 2022. Specifically, we selected only those peptides consisting of the 20 basic amino acids commonly found in biological systems. These amino acids are Alanine (A), Arginine (R), Asparagine (N), Aspartic Acid (D), Cysteine (C), Glutamine (Q), Glutamic Acid (E), Glycine (G), Histidine (H), isoleucine (I), leucine (L), lysine (K), methionine (M), phenylalanine (F), proline (P), serine (S), threonine (T), tryptophan (W), tyrosine (Y) and valine (V). To ensure the accuracy and relevance of our analysis, peptides containing amino acids B, J, O, U, X and Z were excluded from our investigation. By focusing only on the 20 canonical amino acids, we aimed to increase the reliability of our findings and provide a comprehensive understanding of the peptide sequences obtained. Additionally, peptides containing more than 100 or less than 11 amino acids were not selected because AMPs are usually short peptides with the chosen range of length commonly reported in previous studies.[49,50] Next, CD-HIT[51] was used to reduce homology bias and redundancy between peptides. It should be noted that CD-HIT uses a hierarchical clustering algorithm to group similar sequences based on a user-defined sequence identity threshold. Within each cluster, CD-HIT selects a representative sequence, known as the centroid, to represent the cluster. The centroid is typically the longest sequence in the cluster or the first sequence in the input file. In our study, we used CD-HIT with a sequence identity threshold of 90% to eliminate redundant peptides. This means that any two peptides with a sequence identity of 90% or higher were grouped into the same cluster, and one representative (the centroid) was selected to represent that cluster. Consequently, a total of 6845 quality peptides were obtained, and 5 functional classes (antibacterial, targeting mammalian cells, antifungal, antiviral, and anticancer) were chosen for prediction because the number of peptides in these classes is relatively enough and usually more prevalent in this field.

It is important to clarify that each peptide in our dataset is inherently an AMP with at least one known function, since our study focuses on multifunctional AMPs. When we refer to "negative" instances, we are in fact referring to AMPs that do not have the specific function under consideration in a particular binary classification task. For example, if we are training a binary classifier to predict a specific function, such as anticancer, the positive instances will be those peptides known to be anticancer peptides, while the negative instances will be those AMPs

known to perform other functions but not anticancer. This approach allows us to delineate the intricate differences between different functionalities of AMPs.

Feature extraction

The peptide sequences were first converted into numerical vectors as features and used as inputs for machine learning models. Therefore, the platform iFeature[40] was used to generate features related to composition and physicochemical properties (Table S5), and the definition of these features are described below. Note that iFeature is a Python toolkit. It is widely used in the bioinformatics community for extracting a wide range of features from proteins and peptides. Compared to other feature extraction tools, iFeature offers a wider range of feature types and is particularly effective in dealing with peptide sequences, making it well suited for the scope and purpose of our study.

Amino acid composition (AAC)[52] represents the ratio of each amino acid in a protein or a peptide sequence. The proportions of all 20 natural amino acids can be calculated as

$$p(a) = \frac{N(a)}{N}$$

where $N(a)$ is the number of a specific natural amino acid, and N is the total length of the protein or peptide sequence. For example, given a peptide sequence "AAPAACQGVL" comprising a total of 10 amino acids, the calculated proportion of "A" in this peptide is 0.4; accordingly, the proportions of other amino acids can be determined

Pseudo amino acid composition (PAAC)[53,54] mainly uses a matrix of amino acid frequencies similar to AAC to characterize the protein. Moreover, PAAC includes additional physicochemical-related factors, such as hydrophobicity values, hydrophilicity, side chain masses, and sequence order information. Therefore, PAAC represents a set of more than 20 features, where the first 20 are similar to AAC with additional information, and the others are mainly related to sequence order, also called sequence order-correlated factors.

Composition, transition, and distribution (CTD) represent the amino acid distribution patterns of a specific structural or physicochemical property in a protein or peptide sequence.[55–59] In the present study, the distribution descriptor was included as part of our feature set, considering the following seven physicochemical properties: hydrophobicity, normalized van der Waals volume, polarity, polarizability, charge, secondary structure, and solvent accessibility. The 20 amino acids were divided into three groups for each of the 13 attributes derived from the different physicochemical properties, as shown in Table S8. For each group, the occurrence (25/50/75/100%) and position of a residue was divided by the length of the sequence.

Machine learning models

The models used in this study to determine the importance of features for further analysis should be compatible with binary relevance and algorithm adaptation methods. Therefore, RF and convolutional neural network (CNN) were selected as compatible models. For all training processes, 10-fold cross-validation was applied to decrease the probability of overfitting or selection bias.

RF is an ensemble learning method consisting of many decision trees that assign the training data randomly and combines the votes from decision trees to produce the final prediction. RF usually provides a better generalization performance because of combining weak learners, which are models slightly better than random guessing. The core concept of a decision tree is to gain detailed information at every step, and the amount of information can be calculated using the formula of entropy or Gini impurity. Gini impurity can be calculated as:

$$\text{Gini impurity} = 1 - \sum_{i=1}^{c} (p_i)^2$$

where c is the number of classes and p represents the ratio of a specific class at the node. Gini impurity at every node in a decision tree can be calculated using the formula with simultaneous determination of the weighted Gini impurity for splitting the decision trees. Every splitting accompanies a decrease in impurity value for selected features. After completion of decision tree splitting, the mean decrease in impurity (MDI) can be calculated to evaluate the importance of each feature. In this study, we used a fixed number of 100 trees for all RF models. This was done to ensure the robustness of our results and to avoid overfitting. This parameter was chosen based on the integration of several literature guidelines, which indicate a sufficient number of trees to allow stable and reliable prediction performance. In addition to the number of trees, the RF models include several other hyperparameters that, if not specified, take default values as set in scikit-learn. These include, for example, the split quality criterion ('gini' in our case), the maximum tree depth (unbounded in our case), and the minimum number of samples required to split an internal node (default 2). These parameters remained at their default values throughout our study, as our extensive preliminary testing showed that the default setup provided us with satisfactory and robust results for our specific problem and dataset.

We built binary relevance and algorithm adaptation-based RF models using scikit-learn,[41] a python module for machine learning. Accordingly, we constructed 5 binary relevance models for 5 functional classes and a single algorithm adaptation model to predict 5 functional classes at once. The 245 features mentioned in Table S7 were used as inputs for all 6 models, and the feature importance could be determined by calculating MDI. Following the binary relevance approach, we constructed five different models, each dedicated to one of the five functional classes of AMPs identified in our study. By treating each class label as an independent entity, this method essentially transformed our multi-label problem into five separate binary classification tasks. In contrast, the algorithmic adaptation method produced a single model that could predict all five functional classes simultaneously. This approach did not require separating the problem into individual binary classification tasks, but instead directly addressed the multi-label nature of our problem. Both approaches used the same set of 245 features, as listed

in Table S7, to train the models. Each feature serves as an input to the models, and their relative importance was quantified using MDI, which provides an indicator of the contribution of each feature to the prediction of the functional classes.

CNN is a specific type of deep neural network that mainly uses a convolutional layer instead of a fully connected layer. Unlike the fully connected layer, the convolutional layer connects locally and shares the weight of all neurons in a particular feature map, which helps to reduce the number of parameters for easy optimization and avoiding overfitting. The typical CNN architecture comprises three types of layers: convolutional, pooling, and fully connected layer. The convolutional layer contains a set of filters, the parameters of which are being learned, and after learning, the feature maps can be observed as extracted features. Meanwhile, the pooling layer reduces the size of images or feature maps to reserve the most important part of feature maps and also lessens the parameters for the next layer, making the neural network learn efficiently. The fully connected layer plays the role of "classifier" for CNN and can map the feature space, learned using the convolutional layer, to label space. Therefore, the entire neural network learns to classify in a directed manner.

Gradient-weighted class activation mapping (Grad-CAM)[60] is applied to evaluate the importance of features in CNN. Grad-CAM uses the gradient of any target label (antivirus in this study) flowing into the final convolutional layer to produce a heatmap that can be mapped to the original image or features. A more positive number in a specific pixel or feature implies a proportional significance for the target label. Conversely, a bigger negative number is significant for other targets. Therefore, the absolute value of the heatmap can be regarded as a feature contribution, which can be used to calculate the importance of each feature.

To develop our CNN models, we used TensorFlow,[42] a well-established open-source library for high-performance numerical computation, specifically tailored for machine learning applications. The structure of these models consisted of two convolutional layers, followed by a fully connected layer. The convolutional layers were responsible for feature extraction, while the fully connected layer performed the final classification based on the extracted features. The Adam optimizer was used in our models due to its effective and computationally efficient gradient descent optimization, which is suitable for dealing with large-scale problems. We used binary cross-entropy as the loss function, a common choice for binary classification problems, as it quantifies the dissimilarity between the predicted and actual results. To address overfitting, a common machine learning problem where the model performs well on the training data but poorly on unseen data, we used early stopping. This technique stops training if the model's performance does not improve on the validation set after a certain number of epochs. For the five identified functional classes, we constructed five individual binary relevance models and one algorithm adaptation model capable of predicting all five classes simultaneously, as in the case of our RF models. The same 245 features mentioned above were used as inputs for these six models. To understand the contribution of each feature to the prediction, we used Grad-CAM, a method that provides visual explanations of CNN models without requiring architectural changes. This helps us identify the regions that are most relevant to a particular prediction, and thus sheds light on how the model makes its decisions.

Evaluation metrics

Four metrics were used to evaluate the performance of the present prediction models: accuracy, subset accuracy (SA), precision, area under the receiver operating characteristic curve (AUC), and Matthew's correlation coefficient (MCC). The formulae for accuracy, precision, and MCC are listed below:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN}$$

$$\text{Precision} = \frac{TP}{TP+FP}$$

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

$$SA = \frac{1}{n}\sum_{i=1}^{n} 1(\hat{y}_i = y_i)$$

where TP is true positive denoting the number of positive samples predicted correctly; FP is false positive, depicting the number of negative samples predicted wrongly; TN is true negative, meaning the number of negative samples predicted correctly; FN is false negative representing the number of positive samples predicted wrongly; n is the number of samples; $1(\hat{y}_i = y_i)$ with a value of 1 signifies exact match of the predicted labels of *i*th sample with the true labels, whereas a value 0 indicates mismatch of predicted labels; SA is the exact match ratio, i.e., the number of samples with exact match labels divided by the total number of samples.

AUC is the area under the receiver operating characteristic (ROC) curve. The ROC curve is a graphical plot that illustrates the diagnostic ability of a classifier system with variable threshold values; the x axis of the ROC curve denotes a false positive rate, and the y axis depicts a true positive rate. Considering that the threshold values of 0 and 1 represent positive or negative predictions, the point on the plot would be (1,1) and (0,0). Likewise, all the possible points can be obtained on the plot by changing the threshold value, and the area under the curve can be calculated. Unlike metrics using the confusion matrix values (TP, FP, TN, FN), AUC is constant and not affected by changing the threshold. Therefore, AUC was chosen as the metric for feature selection in this study.

For adaptive algorithms, the macro-averaged method was also employed, which computes the metric value independently for each class and subsequently takes an unweighted average to represent comprehensive performance.

QUANTIFICATION AND STATISTICAL ANALYSIS

All computations were performed in the Python programming language. The graphic abstract and Figure 1 were generated by Microsoft PowerPoint, other plots appearing in this study were generated by the Python package.