iAMP-2L: A two-level multi-label classifier for identifying antimicrobial peptides and their functional types

Xuan Xiao\textsuperscript{a,c,*}, Pu Wang\textsuperscript{a}, Wei-Zhong Lin\textsuperscript{a}, Jian-Hua Jia\textsuperscript{a}, Kuo-Chen Chou\textsuperscript{b,c,*}

\textsuperscript{a}Computer Department, Jing-De-Zhen Ceramic Institute, Jing-De-Zhen 333403, China
\textsuperscript{b}King Abdulaziz University, Jeddah, Saudi Arabia
\textsuperscript{c}Gordon Life Science Institute, 53 South Cottage Road, Belmont, Massachusetts 02478, USA

\textbf{A R T I C L E   I N F O}

Article history:
Received 30 October 2012
Received in revised form 10 January 2013
Accepted 21 January 2013
Available online 6 February 2013

Keywords:
Antimicrobial peptide
Pseudo amino acid composition
Physicochemical properties
Fuzzy K-nearest neighbor
Multi-label classification

\textbf{A B S T R A C T}

Antimicrobial peptides (AMPs), also called host defense peptides, are an evolutionarily conserved component of the innate immune response and are found among all classes of life. According to their special functions, AMPs are generally classified into ten categories: Antibacterial Peptides, Anticancer/tumor Peptides, Antifungal Peptides, Anti-HIV Peptides, Antiviral Peptides, Antiparasitidal Peptides, Anti-protozoal Peptides, AMPs with Chemotactic Activity, Insecticidal Peptides, and Spermicidal Peptides. Given a query peptide, how can we identify whether it is an AMP or non-AMP? If it is, can we identify which functional type or types it belong to? Particularly, how can we deal with the multi-type problem since an AMP may belong to two or more functional types? To address these problems, which are obviously very important to both basic research and drug development, a multi-label classifier was developed based on the pseudo amino acid composition (PseAAC) and fuzzy K-nearest neighbor (FKNN) algorithm, where the components of PseAAC were featured by incorporating five physicochemical properties. The novel classifier is called \textit{iAMP-2L}, where “2L” means that it is a 2-level predictor. The 1st-level is to answer the 1st question above, while the 2nd-level is to answer the 2nd and 3rd questions that are beyond the reach of any existing methods in this area. For the conveniences of users, a user-friendly web-server for \textit{iAMP-2L} was established at http://www.jci-bioinfo.cn/iAMP-2L.

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

Found among all classes of life, antimicrobial peptides, also called host defense peptides, are an evolutionarily conserved component of the innate immune response. These peptides are generally between 12 and 50 amino acids, including two or more positively charged residues provided by arginine, lysine or, in acidic environments, histidine, and a large proportion (generally >50\%) of hydrophobic residues\cite{1,2}. It has a special meaning for acidic environments, histidine, and a large proportion (generally positively charged residues provided by arginine, lysine or, in acidic environments, histidine) of activity and the short contact time required for inducing killing. AMPs have been considered as excellent candidates for developing novel therapeutic agents\cite{3,4}. With the growing microbial resistance to conventional antimicrobial agents\cite{5} as well as the avalanche of protein sequences generated in the postgenomic age, it is highly desirable to develop sequence-based computational tools for rapidly and accurately identifying AMPs and their types for helping design new and more effective antimicrobial agents, it is highly desirable to develop computational tools for rapidly and accurately identifying AMPs and their types for helping design new and more effective antimicrobial agents.

Actually, considerable efforts have been made in this regard. For instances, Wang et al. constructed the antimicrobial peptide database (APD)\cite{6} and the updated antimicrobial peptide database (APD2)\cite{7}, accessible at http://aps.unmc.edu/AP/main.php and aimed to be a useful tool for nomenclature, classification, information search, statistical analysis, prediction, and design of antimicrobial peptides. Their prediction interface allows users to input a query peptide sequence for predicting whether it has the potential to be antimicrobial. In 2007, by means of the hidden Markov models (HMMs), Fjell et al. proposed the AMPPer method for identifying AMPs. Meanwhile, Lata et al. successively developed the AntiBP predictor\cite{9} and AntiBP2 predictor\cite{10} for...
identifying antibacterial peptide, one of the subtypes of AMPs according to the amino acid sequence information. Thomas et al. [11] established a useful resource called CAMP (Collection of Anti-Microbial Peptides) for studying AMPs. Based on the experimentally validated data in CAMP, these authors further used various machine-learning algorithms such as Random Forests (RF), Support Vector Machines (SVM) and Discriminant Analysis (DA) to identify AMPs [11]. Subsequently, Wang et al. [12] proposed a new method for predicting AMPs by integrating the sequence alignment method with the feature selection method. Recently, Mohabatkar and coworkers proposed a new method for predicting AMPs peptides based on the concept of Chou’s pseudo-amino acid composition and machine learning methods [13].

Although the aforementioned methods each have their own advantages and did play a role in stimulating the development of this area, they were only focused on identifying whether a query peptide was AMP, or limited at identifying one of its subtypes, without considering various possible different functional types of AMPs. In fact many AMPs have different functions or belong to two or more functional types. It can be seen by a comparison of the sequences in APD database [6] that a same sequence may occur in different subclasses; e.g., the antimicrobial peptide with the code “AMP AP00012” is not only an antibacterial peptide but also an anticancer/tumor peptide and antifungal peptide. Actually, this kind of phenomenon is very common, as can be seen through a statistic analysis conducted on the APD entries. Accordingly, the AMP prediction should be a task of two-level multi-label classification. In view of this, the present study was initiated in an attempt to develop a two-level multi-label predictor for AMP, in which the 1st level is to identify whether a query peptide is AMP, and the 2nd-level is to identify which functional type(s) the peptide belongs to if it turns out to be an AMP in the 1st level prediction.

To establish a really useful prediction method for a biological system based on the sequence information, we need to accomplish the following procedures [14]: (1) construct or select a valid benchmark dataset to train and test the predictor; (2) formulate the biological sequences with an effective mathematical expression that can truly reflect the intrinsic correlation with the target to be predicted; (3) introduce or develop a powerful algorithm to operate the prediction; (4) properly perform a cross-validation test to objectively evaluate the anticipated accuracy; (5) establish a user-friendly web-server for the predictor that can be easily used by most experimental scientists. Below, let us describe how to realize these procedures one by one.

2. Materials and Methods

2.1. Benchmark Dataset

For the convenience of later description, the benchmark dataset is expressed by

\[ S = S_{\text{AMP}} \cup S_{\text{non-AMP}} \]  

(1)

where \( S_{\text{AMP}} \) is the AMP dataset consisting of AMP sequences only, \( S_{\text{non-AMP}} \) the non-AMP dataset with non-AMP sequences only, and \( \cup \) is the symbol for union in the set theory. The peptide sequences in \( S_{\text{AMP}} \) were fetched from the APD database [6,7]. According to their different functional types, the AMP sequences can be further classified into ten categories; i.e.,

\[ S_{\text{AMP}} = S_{\text{AMP}}^1 \cup S_{\text{AMP}}^2 \cup S_{\text{AMP}}^3 \cup S_{\text{AMP}}^4 \cup S_{\text{AMP}}^5 \cup \cdots \cup S_{\text{AMP}}^{10} \]  

(2)

where the subscripts 1, 2, 3, 4, 5, ..., 10 represent “Antibacterial”, “Anticancer/tumor”, “Antifungal”, “Anti-HIV”, “Antiviral”, “Antiparasital”, “Anti-protist”, “AMPs with chemotactic activity”, “Insecticidal” and “Spermicidal” peptides, respectively. As shown in Fig. 1, the lengths of AMPs are varying within the region from 5 to 100 amino acids. Of the aforementioned 10 subsets, the subsets for “Antiparasital”, “Anti-protist”, “AMPs with chemotactic activity”, “Insecticidal” and “Spermicidal” contained too few peptides (less than 50) to have statistical significance, and hence were not further considered in this study. Thus, Eq. (2) can be reduced to

\[ S_{\text{AMP}} = S_{\text{AMP}}^1 \cup S_{\text{AMP}}^2 \cup S_{\text{AMP}}^3 \cup S_{\text{AMP}}^4 \cup S_{\text{AMP}}^5 \]  

(3)

Furthermore, to reduce homology bias and redundancy, the program CD-HIT [15] was utilized to winnow those sequences that have \( \geq 40\% \) pairwise sequence identity to any other in a same subset. However, to ensure each of the subsets have sufficient samples for statistical treatment, the cutoff procedure was only imposed to those subsets that contained more than 150 samples. Finally, we obtained 878 AMPs, of which 454 belong to one functional attribute, 296 to two different functional attributes, 85 to three different functional attributes, 30 to four different functional attributes, and 13 to five different functional attributes. Because some AMPs may belong to two or more functional attributes [6,7], it is instructive to introduce the concept of “virtual AMP” as done in [16,17] when dealing with proteins with multiple location sites. The concept of virtual AMP can be briefed as follows. If an AMP possesses two different attributes of function, it will be counted as two virtual AMPs; if it possesses three attributes, it will be counted as three virtual AMPs; and so forth. Thus, the number of total virtual AMPs can be expressed as [17]

\[ N(\text{vir}) = N(\text{seq}) + \sum_{m=1}^{M} (m-1)N(m) \]  

(4)

where \( N(\text{vir}) \) is the number of total virtual AMPs, \( N(\text{seq}) \) the number of total different AMP sequences, \( N(1) \) the number of AMPs with one functional type, \( N(2) \) the number of AMPs with two functional types, and so forth; while \( M \) is the number of total functional types investigated. Substituting the aforementioned data into Eq. (4), we obtained

\[ N(\text{vir}) = N(\text{seq}) + (1 - 1) \times 454 + (2 - 1) \times 296 + (3 - 1) \times 85 + (4 - 1) \times 30 + (5 - 1) \times 13 \]

\[ = 878 + 0 + 296 + 170 + 90 + 52 = 1,486 \]  

(5)

meaning that the current benchmark dataset \( S_{\text{AMP}} \) contains 1,486 virtual AMPs, of which 770 belong to “Antibacterial”, 140 to “Anticancer/tumor”, 366 to “Antifungal”, 86 to ant-HIV, and 124 to “Antiviral” (see Table 1).

The peptide sequences in \( S_{\text{non-AMP}} \) were constructed according to the following procedures.

\[ N(\text{vir}) = N(\text{seq}) + \sum_{m=1}^{\text{M}} (m-1)N(m) \]

(4)

where \( N(\text{vir}) \) is the number of total virtual AMPs, \( \text{N(\text{seq})} \) the number of total different AMP sequences, \( \text{N(1)} \) the number of AMPs with one functional type, \( \text{N(2)} \) the number of AMPs with two functional types, and so forth; while \( \text{M} \) is the number of total functional types investigated. Substituting the aforementioned data into Eq. (4), we obtained

\[ N(\text{vir}) = N(\text{seq}) + (1 - 1) \times 454 + (2 - 1) \times 296 + (3 - 1) \times 85 + (4 - 1) \times 30 + (5 - 1) \times 13 \]

\[ = 878 + 0 + 296 + 170 + 90 + 52 = 1,486 \]  

(5)

meaning that the current benchmark dataset \( S_{\text{AMP}} \) contains 1,486 virtual AMPs, of which 770 belong to “Antibacterial”, 140 to “Anticancer/tumor”, 366 to “Antifungal”, 86 to ant-HIV, and 124 to “Antiviral” (see Table 1).

The peptide sequences in \( S_{\text{non-AMP}} \) were constructed according to the following procedures.

\[ N(\text{vir}) = N(\text{seq}) + \sum_{m=1}^{\text{M}} (m-1)N(m) \]  

(4)

where \( N(\text{vir}) \) is the number of total virtual AMPs, \( \text{N(\text{seq})} \) the number of total different AMP sequences, \( \text{N(1)} \) the number of AMPs with one functional type, \( \text{N(2)} \) the number of AMPs with two functional types, and so forth; while \( \text{M} \) is the number of total functional types investigated. Substituting the aforementioned data into Eq. (4), we obtained

\[ N(\text{vir}) = N(\text{seq}) + (1 - 1) \times 454 + (2 - 1) \times 296 + (3 - 1) \times 85 + (4 - 1) \times 30 + (5 - 1) \times 13 \]

\[ = 878 + 0 + 296 + 170 + 90 + 52 = 1,486 \]  

(5)

meaning that the current benchmark dataset \( S_{\text{AMP}} \) contains 1,486 virtual AMPs, of which 770 belong to “Antibacterial”, 140 to “Anticancer/tumor”, 366 to “Antifungal”, 86 to ant-HIV, and 124 to “Antiviral” (see Table 1).

The peptide sequences in \( S_{\text{non-AMP}} \) were constructed according to the following procedures.
According to the AAC-discrete model, the peptide $\mathbf{P}$ of Eq. (6) can be formulated by [21]

$$
\mathbf{P} = [f_1 \ f_2 \ \cdots \ f_{L}]^T
$$

(7)

where $f_i$ ($i = 1, 2, \cdots, 20$) are the normalized occurrence frequencies of the 20 native amino acids in peptide $\mathbf{P}$, and $\mathbf{T}$ the transposing operator. Many methods for predicting protein attributes were based on the AAC-discrete model [22–25]. However, as we can see from Eq. (7), if using the ACC model to represent the peptide $\mathbf{P}$, all its sequence-order effects would be lost, and hence the prediction quality might be considerably limited.

To avoid completely losing the sequence-order information, the pseudo amino acid composition (PseAAC) was proposed, as formulated by [26]

$$
\mathbf{P} = [p_1 \ p_2 \ \cdots \ p_{20} \ p_{20-1} \ \cdots \ p_{20+2}]^T
$$

(8)

where

$$
p_u = \frac{\sum_{j=1}^{20} w_{j-u} \ h_{R_j}}{\sum_{j=1}^{20} w_{20-j}} \quad (1 \leq u \leq 20)
$$

(9)

where

$$
\begin{align*}
\theta_1 &= \frac{1}{L} \sum_{i=1}^{L} \Theta(R_i, R_{i+1}) \\
\theta_2 &= \frac{1}{L} \sum_{i=1}^{L-2} \Theta(R_i, R_{i+2}) \\
&\vdots \\
\theta_L &= \frac{1}{L} \sum_{i=1}^{L-12} \Theta(R_i, R_{i+12})
\end{align*}
$$

(10)

where $\theta_i$ is the first-tier correlation factor, $\theta_2$ the 2nd-tier correlation factor, and so forth, while the correlation function is

$$
\Theta(R_i, R_j) = H(R_i) \cdot H(R_j)
$$

(11)

where $H(R_i)$ is the physicochemical property score of the amino acid $R_i$ while $H(R_i)$ the corresponding value for the amino acid $R_j$.

In this study, the following five physical-chemical properties were taken into account: (1) hydrophobicity [27]; (2) pK1 ($\text{C}^a$-COOH) [28]; (3) pK2 (NH3) [28]; (4) PI (25°C) [29]; (5) molecular weight. It is instructive to point out that many preliminary tests had been performed for a series of other physicochemical properties, but better outcomes were observed for the current case by using the aforementioned five properties (see Online Supporting Information S2 for the details).

The numerical values of the five physical-chemical properties for each of the 20 native amino acids can be obtained from [27-29] and most biochemistry text books (see, e.g., [30]). Note that before submitting these physicochemical quantities into Eq. (11), they were each subject to a standard conversion according to the following equation:

$$
y_i = \frac{x_i - \text{mean}(x)}{\text{std}(x)}
$$

(12)

where $x_i$ ($i = 1, 2, \cdots, 20$) is the original physicochemical score of the $i$th amino acid, mean($x$) the average of such score over the 20 native amino acids, and std($x$) the corresponding standard deviation. The converted values thus obtained will have a zero mean value over the 20 amino acids, and will remain unchanged if they go through the same conversion procedure again [31].

As we can see from the above equations, the first 20 elements in Eq. (8) actually reflect the conventional amino acid composition

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Dataset</th>
<th>Functional attribute</th>
<th>Subset</th>
<th>Number of sequences</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMP</td>
<td>$S^{\text{AMP}}$</td>
<td>Antibacterial</td>
<td>$S^{\text{AMP}}$</td>
<td>770</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Anticancer/tumor</td>
<td>$S^{\text{AMP}}$</td>
<td>140</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Antifungal</td>
<td>$S^{\text{AMP}}$</td>
<td>366</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Anti-HIV</td>
<td>$S^{\text{AMP}}$</td>
<td>86</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Antiviral</td>
<td>$S^{\text{AMP}}$</td>
<td>124</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total virtual AMPs</td>
<td>$S^{\text{AMP}}$</td>
<td>1,486</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total different AMPs</td>
<td>$S^{\text{AMP}}$</td>
<td>878</td>
</tr>
<tr>
<td>Non-AMP</td>
<td>$S^{\text{non-AMP}}$</td>
<td>N/A</td>
<td>N/A</td>
<td>2,405</td>
</tr>
</tbody>
</table>

* See Eqs. (4), (5).
AAC; while the additional $\lambda$ factors reflect some sequence-order information via a series of rank-different correlation factors (cf. Eq. (10)).

Since in the current study there are five physical-chemical properties to be taken into consideration, similar to the dimension-augmenting approach in [32], we should use a $(20 + 5 \times 2)$-D feature vector to represent a sample; i.e., instead of Eq. (8), we should have

$$P = [p_1, p_2, \ldots, p_{20}, p_{20.1}, \ldots, p_{20.5}]^T$$  \hspace{0.5cm} (13)

In this study, we choose $w = 0.1$ (cf. Eq. (9)) and $\lambda = 4$ for getting the optimal results.

It is instructive to note that, since the concept of PseAAC was introduced in 2001 [26], it has penetrated into almost all the fields of protein attribute predictions, such as predicting metalloproteinase family [33], predicting GABA(A) receptor proteins [34], predicting enzyme subfamily classes [35], predicting allergenic proteins [36], predicting cyclin proteins [37], predicting protein structural class [38], identifying bacterial virulent proteins [39], predicting DNA-binding proteins [40], predicting protein subcellular location [41], identifying protein submitochondrial localization [42], predicting apoptosis protein subcellular localization [43], predicting outer membrane proteins [44], predicting protein quaternary structure attribute [45,46], classifying amino acids [47], predicting G-protein-coupled receptor classes [48], predicting risk type of human papillomaviruses [49], predicting cyclin proteins [37], predicting protein folding rates [50], predicting protein supersecondary structure [51], among many others. Actually, the concept of PseAAC was not only limited for protein and peptide sequences; recently it was also extended to represent the feature vectors of DNA and nucleotides [52,53], as well as other biological samples (see, e.g., [54,55]). Because it has been widely used, in 2012 a powerful soft-ware called PseAAC-Builder [56] was established for generating various special Chou's pseudo-amino acid compositions, in addition to the web-server PseAAC [57] built in 2008.

2.3. Prediction Engine

An improved fuzzy $K$-nearest neighbor (FKNN) algorithm was used in this study. The FKNN classification method [58] is a variation of the KNN classifier. The latter is quite popular in pattern recognition community owing to its good performance and simple-to-use feature. According to the KNN rule [59,60], named also as the "voting KNN rule", a query sample should be assigned to the subset represented by a majority of its $K$ nearest neighbors, as illustrated in Fig. 5 of [14]. However, in the FKNN classifier, it was the membership values that would be used to determine which class the query sample should belong to, as formulated below.

Suppose $\mathbb{S}(N) = \{P_1, P_2, \ldots, P_n\}$ is a set of vectors representing $N$ peptides in a training set classified into $M$ classes $\{C_1, C_2, \ldots, C_M\}$, where $C_i$ denotes the $i$-th class; $S(P) = \{P_1, P_2, \ldots, P_n\} \subset \mathbb{S}(N)$ is the subset of the $K$ nearest neighbor peptides to the query peptide $P$. Thus, the fuzzy membership value for the query peptide $P$ in the $i$-th class of $\mathbb{S}(N)$ is given by [61]

$$\mu_i(P) = \frac{1 + \sum_{j=1}^{K} \mu_j(P)}{\sum_{j=1}^{K} \text{d}(P, P_j)^{-2/(\lambda-1)}}$$  \hspace{0.5cm} (14)

where $K$ is the number of the nearest neighbors counted for the query peptide $P$; $\mu_j(P)$, the fuzzy membership value of the training sample $P_j$ to the $i$-th class as will be further defined below; $\text{d}(P, P_j)$, the Euclidean distance between $P$ and its $j$th nearest peptide $P_j$ in the training dataset $\mathbb{S}(N)$; $\lambda > 1$, the fuzzy coefficient for determining how heavily the distance is weighted when calculating each nearest neighbor's contribution to the membership value. Note that the parameters $K$ and $\lambda$ will affect the computation result of Eq. (14), and they will be optimized by a grid-search as will be described later.

For the 1st-level prediction in identifying a query peptide $P$ as an AMP or non-AMP, a task of single-label classification, the quantitative definition for the aforementioned $\mu_i(P)$ in Eq. (14) is given by

$$\begin{cases} 
\mu_i(P) = 1, & \text{if } P \in C_i \\
\mu_i(P) = 0, & \text{otherwise}
\end{cases}$$  \hspace{0.5cm} (15)

After calculating all the memberships for a query peptide via Eqs. (15) and (14), it is assigned to the class with which it has the highest membership value; i.e., the predicted class for $P$ should be

$$C_u = \text{argmax}_{i} (\mu_i(P))$$  \hspace{0.5cm} (16)

where $u$ is the argument of $i$ that maximizes $\mu_i(P)$.

For the 2nd-level prediction in identifying which functional type(s) the query AMP peptide belongs to, a task of multi-label classification, an ingenious scheme will be used to replace Eq. (15); i.e.,

$$\begin{cases} 
\mu_i(P_j) = \frac{1}{n(\text{hit})}, & \text{if } P_j \in C_i \\
\mu_i(P_j) = 0, & \text{otherwise}
\end{cases}$$  \hspace{0.5cm} (17)

where $n(\text{hit})$ is the number of different classes that were hit by $P_j$ during the prediction. For instance: if only $C_1$ was hit by $P_j$, then we have $n(\text{hit}) = 1$ and $\mu_1(P_j) = 1$ and $\mu_{i \neq 1}(P_j) = 0$; if only $C_1$ and $C_2$ were hit by $P_j$, then $n(\text{hit}) = 2$, $\mu_1(P_j) = \mu_2(P_j) = 0.5$, and $\mu_{i \neq 1}(P_j) = 0$; and so forth.

After calculating all the memberships for a query peptide via Eqs. (17) and (14), its functional type(s) will be predicted by

$$C_u = \{C_i | \mu_i(P) \geq \Psi\}$$  \hspace{0.5cm} (18)

where $u$ is the argument(s) of $i$ satisfying the condition $\mu_i(P) \geq \Psi$, where $\Psi$ is a threshold. For example, if both $\mu_i(P)$ and $\mu_j(P)$ are equal to or greater than $\Psi$ but $\mu_{i \neq j}(P) \in \Psi$, then $u = \{1,3\}$ and $P$ will be predicted as "antibacterial" peptide and "antifungal" peptide (cf. Table 1), and so forth. The value of $\Psi$ will also be determined by an optimal procedure via the grid-search.

The classifier thus established is called iAMP-2L, where "1" means identifying, and "2L" means the identification consisting of two layers. The 1st layer is to identify a query peptide as AMP or not; if it is an AMP, the 2nd layer will be automatically continued to further identifying the AMP among the five functional attributes (cf. Table 1). To provide an intuitive picture, a flowchart to show the process of how the classifier works is given in Fig. 2.

2.4. Web Server

For practical applications, a user-friendly web-server for iAMP-2L was established at http://www.jci-bioinfo.cn/iAMP-2L, by which users can easily obtain their desired results without the need to follow the complicated mathematical equations involved for developing the predictor.

2.5. Performance Metrics

The ways to calculate the success rates for the single-label and multi-label classification should not be the same [31].
The 1st-level prediction in identifying a query peptide as an AMP or non-AMP belongs to the case of single-label classification. The following equation is often used in literature for examining the prediction quality of a single-label predictor

\[
\begin{align*}
S_n & = \frac{TP}{TP + FN} \\
Sp & = \frac{TN}{FP + TN} \\
Acc & = \frac{TP + TN}{TP + TN + FP + FN} \\
MCC & = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}
\end{align*}
\]

where \(TP\) represents the true positive; \(TN\), the true negative; \(FP\), the false positive; \(FN\), the false negative; \(Sn\), the sensitivity; \(Sp\), the specificity; \(Acc\), the accuracy; \(MCC\), the Mathew’s correlation coefficient. For an intuitive and easy-to-understand explanation of Eq. (19), see a recent paper [52].

The 2nd-level prediction in identifying a query AMP among its five functional types (cf. Table 1) belongs to the case of multi-label classification, and its quality should be evaluated as follows. For a multi-label system consisting of \(N\) samples, suppose \(M\) is the number of all possible different categories, i.e., the label set that contains the labels for all the possible categories concerned. Thus, the \(i\)-th peptide sample \(P_i\), and the category or categories it belongs to can be expressed by

\[
\{P_i, \{L_i\} (i = 1, 2, \ldots, N)
\]

where \(L_i\) is the subset that contains all the functional type label(s) for the \(i\)-th peptide sample. Obviously, we have

\[
\sum_{i=1}^{N} L_i \subseteq L \label{eq:21}
\]

where \(L_i (i = 1, 2, \ldots, M)\) is the label for the \(i\)-th functional type. For the current study, \(N = 878\) and \(M = 5\) (cf. Table 1). Suppose \(L_i\) represents the subset that contains all the predicted functional type label(s) for the \(i\)-th peptide. Thus, we have the following three metrics to measure the prediction quality for the multi-label system [17,62].

\[
\begin{align*}
\text{Hamming loss} & = \frac{1}{N} \sum_{i=1}^{N} \left( \frac{||L_i \cap L_i||}{|L_i|} \right) \\
\text{Accuracy} & = \frac{1}{N} \sum_{i=1}^{N} \left( \frac{||L_i \cap L_i||}{|L_i|} \right) \\
\text{Precision} & = \frac{1}{N} \sum_{i=1}^{N} \left( \frac{||L_i \cap L_i||}{|L_i|} \right) \\
\text{Recall} & = \frac{1}{N} \sum_{i=1}^{N} \left( \frac{||L_i \cap L_i||}{|L_i|} \right) \\
\text{Absolute} - \text{True} & = \frac{1}{N} \sum_{i=1}^{N} \Delta (L_i, L_i)
\end{align*}
\]

where \(M = 5\) (cf. Table 1) is the total number of AMPs functional types covered by the current benchmark dataset \(\mathcal{S}_{\text{AMP}}\); \(\cup\) the symbol of union in the set theory, \(\cap\) the intersection symbol, \(\Delta\) the operator acting on the set therein to count the number of its elements, and

\[
\Delta (L_i, L_i) = \begin{cases} 
1, & \text{if all the labels in } L_i \text{ are identical to those in } L_i \\
0, & \text{otherwise}
\end{cases}
\]

Among the above five metrics, the rate for “Hamming loss” [62] reflects the rate of absolute false, which is opposite to those of the other four. As can be easily seen from Eq. (22), when the multi-labels for all the samples are correctly predicted, i.e., \(L_i \equiv L_i\) or \(||L_i \cup L_i|| = ||L_i \cap L_i|| \quad (i = 1, 2, \ldots, N)\), the rate of Hamming loss is equal to 0. When each of \(P_i\) \( (i = 1, 2, \ldots, N)\) is predicted completely wrong, i.e., belonging to all the possible categories except its own true category or categories; i.e., \(L_i \cup L_i = L\) and \(L_i \cap L_i = \emptyset\), or \(||L_i \cup L_i|| = M\) and \(||L_i \cap L_i|| = 0\), the rate of Hamming loss is equal to 1. Therefore, the lower the Hamming loss is, the better the prediction quality will be. However, for the other four metrics, the meanings of their rates are just opposite; i.e., the higher their rates are, the better the prediction quality will be. As we can see from the above, it is much more complicated to evaluate the quality of a classifier on a multi-label system, just like in predicting protein subcellular localization for a system containing both single-location and multiple-location proteins, as elaborated in [31] and described by Eqs. (43)–(48) and Fig. 4 therein.

3. Results and Discussion

To validate a predictor, the following three cross-validation methods are often used in literatures: independent dataset test, subsampling test, and jackknife test [63]. However, as elaborated in [64] and demonstrated by Eqs. (28)–(30) in [14], considerable arbitrariness exists in the independent dataset test and subsampling test (or K-fold cross-over), and only the jackknife test is the least arbitrary that can always yield a unique result for a given benchmark dataset. Therefore, the jackknife test, also called Leave-One-Out (LOO) cross-validation [65], has been widely recognized and increasingly adopted by investigators to examine the quality of various predictors (see, e.g., [17,49,51,53,66–69]). In view of this, the jackknife test was also adopted in this study to examine the prediction quality of iAMP-2L.
For the 1st-level prediction, the values of parameter \( K \) and \( \varphi \) in Eq. (14) were determined by maximizing Acc of Eq. (19) using the jackknife test on the dataset \( S \) of Eq. (1) thru a 2-D grid search. For the 2nd-level prediction, the values of parameter \( K \) and \( \varphi \) in Eq. (14) and \( \Psi \) in Eq. (18) were determined by minimizing the Hamming loss of Eq. (22) with the jackknife validation on the dataset \( S_{AMP} \) of Eq. (2) thru a 3-D grid search. The optimal parameter values thus determined for \( K \) and \( \varphi \) in Eq. (14) and \( \Psi \) in Eq. (18) are summarized as follows

\[
\begin{align*}
K & = 19, \varphi = 1.8 \quad (\text{for 1st-level operation}) \\
K & = 21, \varphi = 1.5, \Psi = 0.31 \quad (\text{for 2nd-level operation})
\end{align*}
\]

The single-label prediction quality achieved by the 1st-level of \textit{iAMP-2L} for identifying AMPs and non-AMPs in the benchmark dataset \( S \) is measured by the four metrics as defined in Eq. (19). And their rates are given in Table 2, from which we can see the overall success rate achieved by \textit{iAMP-2L} in identifying AMP or non-AMP is over 86%.

The multi-label prediction quality achieved by the 2nd-level of \textit{iAMP-2L} is measured by the five metrics as defined in Eq. (22), and their outcomes for the benchmark dataset \( S_{AMP} \) (cf. Eq. (3)) are given below

\[
\begin{align*}
\text{Hamming loss} & = 0.1640 \\
\text{Accuracy} & = 0.6687 \\
\text{Precision} & = 0.8331 \\
\text{Recall} & = 0.7570 \\
\text{Absolute true} & = 0.4305
\end{align*}
\]

from which we can see the overall absolute-false or Hamming-loss (or absolute false) rate is very low (16.40%), while the absolute-true rate is much higher (43.05%), indicating the \textit{iAMP-2L} is quite a promising multi-label predictor in identifying the functional types of AMPs as elucidated below.

It is instructive to point out that, for a multi-label system like the current one, the absolute-true success rate for each of the individual AMP functional types is meaningless and misleading. And their rates are given in Table 2, from which we can see the overall success rate achieved by \textit{iAMP-2L} in identifying AMP or non-AMP is over 86%.

The multi-label prediction quality achieved by the 2nd-level of \textit{iAMP-2L} is measured by the five metrics as defined in Eq. (22), and their outcomes for the benchmark dataset \( S_{AMP} \) are given below

\[
\begin{align*}
\text{Hamming loss} &= 0.1640 \\
\text{Accuracy} &= 0.6687 \\
\text{Precision} &= 0.8331 \\
\text{Recall} &= 0.7570 \\
\text{Absolute true} &= 0.4305
\end{align*}
\]

The completely random guess and weighted random guess, as defined below.

The completely random guess (CRG) rates were calculated according to the following equation

\[
P(CRG) = \frac{1}{M} \frac{1}{C(M, m)} = \frac{1}{M} \frac{M!}{(m!)(M-m)!} (m \leq M)
\]

where \( M \) is the total number of all the AMP functional types investigated that is equal to 5 for the current benchmark dataset \( S_{AMP} \). \( m \) has the same meaning as in Eq. (3), and the symbol \( C(M, m) \) represents the number of combinations of \( M \) distinct things (or functional types) taken \( m \) at a time.

The weighted random guess (WRG) rates were calculated according to the following equation

\[
P(WRG) = \frac{N(m)}{N(seq)} \cdot \frac{1}{C(M, m)} = \frac{N(m)}{\sum_{m' = 1}^{M} N(m')} (m \leq M)
\]

where \( N(seq) \) and \( N(m) \) have the same meanings as in Eq. (3).

From Table 3 we can see the following: (1) absolute true rates for the AMPs with only one functional type is much higher than those with multiple functional types, indicating that it is much more difficult to predict the latter functional types exactly without any over-or under-prediction; (2) although for the small numbers of AMPs with 4 and 5 functional types the absolute-true rates by the \textit{iAMP-2L} are about the same or slightly higher than those by the completely random guess, for most AMPs with 1 to 3 functional types the absolute-true rates achieved by \textit{iAMP-2L} are about 12–15 times higher than those by the completely random guess; (3) particularly, for all the five cases, the absolute true rates by \textit{iAMP-2L} are about 5–100 times higher than those by the weighted random guess.

To further demonstrate the power of the \textit{iAMP-2L} predictor, let us compare it with some existing methods. As mentioned in the Introduction section, all the existing methods can only be used to identify a query peptide as an AMP or non-AMP, i.e., the 1st-level job by \textit{iAMP-2L}; none of the existing methods can be used to deal with the 2nd-level job of \textit{iAMP-2L}. Accordingly the comparison was limited in identifying AMPs or non-AMPs only. Also, the methods proposed in [12] and [13] did not provide any web-server, while the method in [9,10] were limited for antibacterial peptides only. To make it feasible and meaningful, the comparison was performed with the \textit{CAMP} method [11], which contained three different algorithms or operation engines: the Support Vector Machine, Random Forests, and Discriminant Analysis.

Table 2 Performance metrics (see Eq. (19)) achieved by \textit{iAMP-2L} in identifying AMP and non-AMP.

<table>
<thead>
<tr>
<th>Sn</th>
<th>Sp</th>
<th>Acc</th>
<th>MCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>87.13%</td>
<td>86.03%</td>
<td>86.32%</td>
<td>0.7265</td>
</tr>
</tbody>
</table>

* The rates reported here were based on the jackknife test on the benchmark dataset \( S \) of Eq. (1) and Online Supporting Information S1.

Table 3 A comparison of the absolute true success rates by different methods for the AMPs with different numbers of functional types.

<table>
<thead>
<tr>
<th>Number of functional types or labels</th>
<th>Number of AMPs</th>
<th>Absolute true rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>iAMP-2L</td>
<td></td>
<td>Completely random guessa</td>
</tr>
<tr>
<td>1</td>
<td>454</td>
<td>60.79%</td>
</tr>
<tr>
<td>2</td>
<td>296</td>
<td>23.99%</td>
</tr>
<tr>
<td>3</td>
<td>85</td>
<td>31.76%</td>
</tr>
<tr>
<td>4</td>
<td>30</td>
<td>33.33%</td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>23.08%</td>
</tr>
</tbody>
</table>

* The completely random guess was calculated according to Eq. (26).
* The weighted random guess was calculated according to Eq. (27).
iAMP-2L are remarkably higher than those by CAMP regardless of which of its three operation engines was used for the prediction. Why could the overall success rate be improved so remarkably by introducing the PseAAC? To address this problem, let us carry out a graphical analysis. Using graphic approaches to study biological systems can provide an intuitive vision and useful insights for helping analyze complicated relations therein, as indicated by many previous studies on a series of important biological topics, such as enzyme-catalyzed reactions [71–73], protein folding kinetics and folding rates [74], inhibition of HIV-1 reverse transcriptase [75], inhibition kinetics of processive nucleic acid polymerases and nucleases [76], protein sequence evolution [77], drug metabolism systems [78], and recently using wenxiang diagrams or graphs [79] to analyze protein-protein interactions [80].

To perform graphic analysis for the current case, let us consider the standard vectors [81,82] or norms [83] for AMPs and non-AMPs that were originally introduced for studying protein structural classification [81–83]. According to Eq. (13), when \( K = 4 \) the standard vector \( \bar{\mathbf{P}}^{\text{AMP}} \) for AMPs and the standard vector \( \bar{\mathbf{P}}^{\text{non-AMP}} \) for non-AMPs can be respectively formulated as

\[
\bar{\mathbf{P}}^{\text{AMP}} = \begin{bmatrix} p_1^{\text{AMP}} & p_2^{\text{AMP}} & \cdots & p_u^{\text{AMP}} & \cdots & p_{40}^{\text{AMP}} \end{bmatrix}
\]

and

\[
\bar{\mathbf{P}}^{\text{non-AMP}} = \begin{bmatrix} p_1^{\text{non-AMP}} & p_2^{\text{non-AMP}} & \cdots & p_u^{\text{non-AMP}} & \cdots & p_{40}^{\text{non-AMP}} \end{bmatrix}
\]

The components in Eq. (28) are given by

\[
p_u^{\text{AMP}} = \frac{1}{40} \sum_{k=1}^{N^{\text{AMP}}} p_{u,k}^{\text{AMP}} \quad (u = 1, 2, \ldots, 40)
\]

where \( p_{u,k}^{\text{AMP}} \) is the \( u \)-th PseAAC component of the \( k \)-th peptide in the training dataset \( S^{\text{AMP}} \), while \( N^{\text{AMP}} \) the total number of the AMP samples in \( S^{\text{AMP}} \).

The components in Eq. (29) are given by

\[
p_u^{\text{non-AMP}} = \frac{1}{40} \sum_{k=1}^{N^{\text{non-AMP}}} p_{u,k}^{\text{non-AMP}} \quad (u = 1, 2, \ldots, 40)
\]

where \( p_{u,k}^{\text{non-AMP}} \) is the \( u \)-th PseAAC component of the \( k \)-th peptide in the training dataset \( S^{\text{non-AMP}} \), while \( N^{\text{non-AMP}} \) the total number of the non-AMP samples in \( S^{\text{non-AMP}} \). According to Eqs. (28) and (29), the standard vectors for AMPs and non-AMPs are two 40-D vectors.

To provide an intuitive picture, let us project the 40 components in each of the two standard vectors onto a 2-D radar graph or diagram [84]. The radar diagrams thus obtained are shown in Fig. 3, where panel A is the standard vector for AMPs, and panel B for non-AMPs. As we can see from the figure, the radar diagram for the PseAAC of AMPs is remarkably different with that of non-AMPs.

In other words, incorporating the aforementioned five physical-chemical properties into PseAAC can significantly enhance the distinction between AMPs and non-AMPs. This is the key why the success rate achieved by iAMP-2L in identifying AMPs and non-AMPs was so high. This is the essence why we choose to use the five physical-chemical properties.

By following the similar procedures, the radar diagrams for the five different functional types of AMPs can also be generated, as shown in Panels A, B, C, D, and E of Fig. 4. Compared with Fig. 3, the distinction among the five radar graphs in Fig. 4 is less remarkable. This is because the classification of the AMP functional types is actually a multi-label problem since an AMP may belong to two or more functional types. That is also why the overall absolute-true success rates achieved by iAMP-2L for all the five cases listed in Table 3 are about 5–100 times higher than those by the weighted random guess.

4. Conclusion

The ability of AMPs to kill multidrug-resistant microorganisms has gained them considerable attention and clinical interest. With the growing microbial resistance to conventional antimicrobial agents, the demand for unconventional and efficient AMPs has become urgent. The results reported in this study indicate that the new predictor iAMP-2L holds very high potential to become a use-

---

**Table 4**

Comparison of iAMP-2L with CAMP [11] on the independent dataset \( S^{\text{test}} \) containing 920 AMPs and 920 non-AMPs verified by experiments that were outside the dataset used to train the predictors. The detailed peptide sequences in \( S^{\text{test}} \) are given in Online Supporting Information S3.

<table>
<thead>
<tr>
<th>Method</th>
<th>Algorithm</th>
<th>Sn</th>
<th>Sp</th>
<th>Acc</th>
<th>MCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAMP</td>
<td>Support vector</td>
<td>88.37%</td>
<td>66.63%</td>
<td>77.50%</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>machine</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Random forest</td>
<td>89.67%</td>
<td>25.98%</td>
<td>57.83%</td>
<td>0.1565</td>
</tr>
<tr>
<td>Discriminant analysis</td>
<td>86.63%</td>
<td>64.13%</td>
<td>75.38%</td>
<td>0.5076</td>
<td></td>
</tr>
<tr>
<td>iAMP-2L</td>
<td>Fuzzy K-nearest neighbor</td>
<td>97.72%</td>
<td>86.74%</td>
<td>92.23%</td>
<td>0.8446</td>
</tr>
</tbody>
</table>

where \( S^{\text{test}} \) is the training dataset for AMPs and non-AMPs. According to Eqs. (28) and (29), the absolute true success rates achieved by iAMP-2L for all the five cases listed in Table 3 are about 5–100 times higher than those by the weighted random guess.

**Table 4**

Comparison of iAMP-2L with CAMP [11] on the independent dataset \( S^{\text{test}} \) containing 920 AMPs and 920 non-AMPs verified by experiments that were outside the dataset used to train the predictors. The detailed peptide sequences in \( S^{\text{test}} \) are given in Online Supporting Information S3.

<table>
<thead>
<tr>
<th>Method</th>
<th>Algorithm</th>
<th>Sn</th>
<th>Sp</th>
<th>Acc</th>
<th>MCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAMP</td>
<td>Support vector</td>
<td>88.37%</td>
<td>66.63%</td>
<td>77.50%</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>machine</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Random forest</td>
<td>89.67%</td>
<td>25.98%</td>
<td>57.83%</td>
<td>0.1565</td>
</tr>
<tr>
<td>Discriminant analysis</td>
<td>86.63%</td>
<td>64.13%</td>
<td>75.38%</td>
<td>0.5076</td>
<td></td>
</tr>
<tr>
<td>iAMP-2L</td>
<td>Fuzzy K-nearest neighbor</td>
<td>97.72%</td>
<td>86.74%</td>
<td>92.23%</td>
<td>0.8446</td>
</tr>
</tbody>
</table>

where \( S^{\text{test}} \) is the training dataset for AMPs and non-AMPs. According to Eqs. (28) and (29), the absolute true success rates achieved by iAMP-2L for all the five cases listed in Table 3 are about 5–100 times higher than those by the weighted random guess.

**Table 4**

Comparison of iAMP-2L with CAMP [11] on the independent dataset \( S^{\text{test}} \) containing 920 AMPs and 920 non-AMPs verified by experiments that were outside the dataset used to train the predictors. The detailed peptide sequences in \( S^{\text{test}} \) are given in Online Supporting Information S3.

<table>
<thead>
<tr>
<th>Method</th>
<th>Algorithm</th>
<th>Sn</th>
<th>Sp</th>
<th>Acc</th>
<th>MCC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAMP</td>
<td>Support vector</td>
<td>88.37%</td>
<td>66.63%</td>
<td>77.50%</td>
<td>0.55</td>
</tr>
<tr>
<td></td>
<td>machine</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Random forest</td>
<td>89.67%</td>
<td>25.98%</td>
<td>57.83%</td>
<td>0.1565</td>
</tr>
<tr>
<td>Discriminant analysis</td>
<td>86.63%</td>
<td>64.13%</td>
<td>75.38%</td>
<td>0.5076</td>
<td></td>
</tr>
<tr>
<td>iAMP-2L</td>
<td>Fuzzy K-nearest neighbor</td>
<td>97.72%</td>
<td>86.74%</td>
<td>92.23%</td>
<td>0.8446</td>
</tr>
</tbody>
</table>

where \( S^{\text{test}} \) is the training dataset for AMPs and non-AMPs. According to Eqs. (28) and (29), the absolute true success rates achieved by iAMP-2L for all the five cases listed in Table 3 are about 5–100 times higher than those by the weighted random guess.

---

**Fig. 3.** The radar diagram (or graph) to show the difference between (A) AMPs and (B) non-AMPs via their 40-D PseAAC standard vectors as defined by Eqs. (28) and (29), respectively.
ful high throughput tool for identifying AMPs and its functional types. Or at the very least, it may play an important complementary role to the existing predictors in this area. It has not escaped our notice that, with more data available for "Antiparasital", "Anti-protist", "AMPs with chemotactic activity", "Insecticidal" and "Spermicidal" in future, the current method can straightforwardly extended to also cover these five AMP functional types. By that time, **iAMP-2L** will be able to identify AMPs and all their ten possible functional types as well. An announcement will be made either by a new publication or by the webpage of **iAMP-2L** when its coverage scope has been significantly enhanced.

**Acknowledgments**

This work was supported by the grants from the National Natural Science Foundation of China (No.60961003, No.6121027 and No.31260273), the Key Project of Chinese Ministry of Education (No. 210116), the Province National Natural Science Foundation of JiangXi (No.2010GZS0122, No.20114BAB211013 and No. 20122BAB201020), the LuoDi plan of the Department of Education of JiangXi Province (KJLD12083), the Jiangxi Provincial Foreign Scientific and Technological Cooperation Project (No.20120 BDH80023), the Department of Education of JiangXi Province (GJJ12490), and the JiangXi Provincial Foundation for Leaders of Disciplines in Science (201138CB220008).

**Fig.4.** The radar diagram to show the difference among the following five different functional types of AMPs: (A) Antibacterial, (B) Anticancer/tumor, (C) Antifungal, (D) Anti-HIV, and (E) Antiviral. See the legend of Fig. 3 for more explanation.


