International Journal of
Molecular Sciences

MDPI

Article

Prediction of Peptide Detectability Based on CapsNet and Convolutional Block Attention Module

Minzhe Yu [1], Yushuai Duan [1], Zhong Li [1,*] and Yang Zhang [2]

[1] Department of Mathematical Sciences, School of Science, Zhejiang Sci-Tech University, Xuelin St., Hangzhou 310018, China; 201930605036@mail.zstu.edu.cn (M.Y.); duanyushuai1022@163.com (Y.D.)
[2] Institutes of Biomedical Sciences, Fudan University, 138 Yixueyuan Road, Shanghai 200032, China; zhangyang@fudan.edu.cn
* Correspondence: lizhong@zstu.edu.cn

Abstract: According to proteomics technology, as impacted by the complexity of sampling in the experimental process, several problems remain with the reproducibility of mass spectrometry experiments, and the peptide identification and quantitative results continue to be random. Predicting the detectability exhibited by peptides can optimize the mentioned results to be more accurate, so such a prediction is of high research significance. This study builds a novel method to predict the detectability of peptides by complying with the capsule network (CapsNet) and the convolutional block attention module (CBAM). First, the residue conical coordinate (RCC), the amino acid composition (AAC), the dipeptide composition (DPC), and the sequence embedding code (SEC) are extracted as the peptide chain features. Subsequently, these features are divided into the biological feature and sequence feature, and separately inputted into the neural network of CapsNet. Moreover, the attention module CBAM is added to the network to assign weights to channels and spaces, as an attempt to enhance the feature learning and improve the network training effect. To verify the effectiveness of the proposed method, it is compared with some other popular methods. As revealed from the experimentally achieved results, the proposed method outperforms those methods in most performance assessments.

Keywords: peptide detectability; CapsNet; CBAM; physicochemical properties of residues; amino acid composition; dipeptide composition

Citation: Yu, M.; Duan, Y.; Li, Z.; Zhang, Y. Prediction of Peptide Detectability Based on CapsNet and Convolutional Block Attention Module. *Int. J. Mol. Sci.* **2021**, *22*, 12080. https://doi.org/10.3390/ijms222112080

Academic Editor: Wojciech Bal

Received: 18 October 2021
Accepted: 2 November 2021
Published: 8 November 2021

1. Introduction

Proteomics is a vital technology in the field of high-throughput experiments. To be specific, protein detection and quantification are vital for gaining more insights into cell biology and human disease [1]. The advancement of mass spectrometry (MS) analysis is of critical significance to provide reliable results at the proteomics level, whereas several problems remain with the existing technology (complex experimental procedures make data processing difficult) (e.g., peptide detectability). The detectability of a peptide is defined as the possibility of observing (or identifying) the peptide from a standard sample mixture. The predicted peptide detectability can be drawn upon to solve shotgun proteomics (e.g., protein inference [2] and label-free quantification [3]). Since there are considerable variables when a certain peptide chain or amino acid sequence is being detected with the mass spectrometer, the characteristics of a set peptide are difficult to quantify [4]. Recently, researchers have proposed a wide range of methods to assess the detectability of peptides by employing standard samples [5] or peptide groups identified in different biological samples [6]. Since there have been numerous positive and negative samples with detectable peptide chains over the past few years, some machine learning (deep learning) methods [7,8] can be exploited to predict the detectability of peptides.

Guruceaga et al. initially defined two types or classes of peptides according to detected and undetected peptides by MS [9]. They extracted features by the physicochemical

properties of peptides stored in the AAIndex resource [10], and subsequently used machine learning methods (e.g., SVM and random forest (RF)) for classification. Li et al. [5] proposed a novel algorithm for the iterative learning of peptide detectability from complex samples. Based on their method, 292 features were computed solely from peptide chains and the neighboring residues in proteins as the feature input. Next, a modular neural network was designed to estimate protein quantities and predict the peptide detectability. Zimmer et al. [11] developed an algorithm by complying with the deep fully connected feed-forward neural network, thereby achieving the informed selection of synthetic prototypic peptides to effectively design targeted proteomics quantification assays. They adopted a BioFSharp toolbox to convert a set peptide chain into a feature vector with 45 entries, representing a numerical footprint of physicochemical properties of peptides as the input of the neural network. Wei et al. [12] developed PEPred-Suite, a tool that introduces an adaptive feature representation strategy capable of learning the most representative features exhibited by different peptide types. Thus, it can predict a variety of peptide types simultaneously (e.g., anti-inflammatory peptides and antiviral peptides). Zhang et al. [13] proposed a therapeutic peptide prediction feature by complying with the physicochemical properties of residues, and then adopted the RF prediction method to solve various peptide prediction problems. These researchers finally achieved a satisfactory prediction result. Recently, deep learning methods have been applied for feature extraction and learning and have been extensively employed in biological data prediction and classification [14]. For instance, Guruceaga et al. proposed a deep learning method for the detectability of peptides [15]. They extracted the sequence coding feature and subsequently adopted CNN training to develop a DeepMSPeptide method. By this method, the test dataset in the GPMDB database achieved an accuracy of 79.53%. Cheng et al. [6] proposed PepFormer, a novel type of end-to-end conjoined network that couples with a hybrid architecture of Transformer and gated cyclic units, which is capable of predicting the detectability of peptides by only complying with the peptide chain.

This study introduces a novel integrated learning network framework based on the capsule network (CapsNet) and the convolutional block attention module (CBAM) to predict the detectability of peptides. First, it builds the residue conical coordinate (RCC) feature [16] by complying with the physicochemical properties of residues and combines the statistical information to improve feature extraction. In addition, the amino acid composition (AAC) and dipeptide composition (DPC) and the sequence embedding code (SEC) are fused as the feature input for the detectability prediction of peptides. Next, the mentioned features are split into biological features and sequence features and separately inputted into the neural network model to reduce the influence attributed to the mentioned two types of features. In the proposed neural network framework, it applies CapsNet [17], thereby reducing the impact of losing part of the convolutional and pooling layers of CNN. Furthermore, CBAM [18] is introduced to the capsule network to assign weights to channels and spaces to learn the vital features and optimize the prediction results. The experimentally achieved results verify the effectiveness of this method.

2. Results

CapsNet and CBAM are integrated to design a novel neural network model for predicting the peptide detectability. To assess the proposed model, a comparison is drawn for the performance with or without a CBAM module, the experiment is performed using different feature inputs, and different neural network frameworks are compared to verify the effectiveness of the proposed model. Subsequently, this study analyzes the accuracy of the test set with the GPMDB database [3] and compares it with some popular methods. Furthermore, the detectability of different peptides is predicted for additional benchmark tests. In addition, to explore the ability of this model to predict other types of peptides, a test is set to predict whether the predicted peptides are anti-angiogenic peptides or antibacterial peptides.

2.1. Comparison with or without CBAM Module and with or without Feature Separation Input

To verify the effect of the CBAM attention module and feature separation input on peptide detectability prediction, three experimental groups are set, i.e., CapsNet + CBAM + feature (biological and sequence feature) separation input, CapsNet + feature separation input, and CapsNet + CBAM + feature combination input. The experimentally achieved results are presented in Figure 1 (Figure 1a draws a comparison of the area under the ROC curve (AUC) and Figure 1b shows a comparison of accuracy (ACC)). When repeating the experiment ten times (the experiment is repeated through the ten-fold cross-validation, and the average and deviations of ten results are taken as the result), box plots are employed to compare AUC and ACC indicators of the mentioned three frameworks. AUC and ACC indicators can reveal the advantage of the attention module of this study and the role of the feature separation input here (AUC and ACC by CBAM module and feature separation input achieve the highest values of 0.862 and 0.801, respectively). This study finds that the feature area receives more attention after convolution with a large degree of discrimination because of the CBAM module, thereby improving the accuracy of the detectability prediction of peptides. In addition, the feature separation operation is exploited to separate the feature of embedding from the features by other biological calculations to reduce the noise influence between two types of features during convolution, which can improve the prediction results as well.

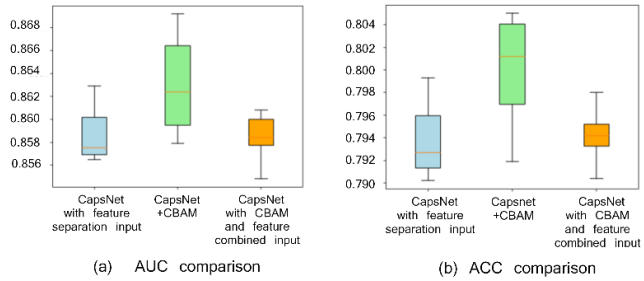

Figure 1. Comparison with or without CBAM and with or without feature separation input on the peptide detectability prediction for the test set from GPMDB. The blue, green, and orange groups are the results by CapsNet + feature separation input, CapsNet + CBAM + feature separation input, and CapsNet + CBAM + feature combined input, respectively.

2.2. Comparison with Different Input Features

Besides the CBAM module, various input features play different roles in the proposed prediction model. The ablation study is conducted by setting different feature inputs in the proposed model. To be specific, the predicting performance is compared by inputting the residue conical coordinate (RCC) feature, amino acid composition (AAC) and dipeptide composition (DPC), and the sequence embedding code (SEC), respectively. The RCC feature covers the effect of physicochemical properties exhibited by the respective amino acid. The AAC and DPC reflect the effect of the frequency of the respective amino acid and dipeptide on the detectability of peptides. The SEC contains the sequence information of residues in the peptide chain. The comparison results are shown in Table 1, and their ROC curves (the 10-fold cross-validation result is used to calculate the average value of true positive rate and false positive rate to plot the ROC curve) are shown in Figure 2a–c. In Table 1, the mean and standard deviations of ten-fold cross-validation are provided for all different feature inputs. According to the table, the addition of RCC and SEC features optimizes the results of peptide detectability prediction. The addition of AAC and DPC (frequency features) slightly improves the AUC result, whereas it improves the F-score result, which is shown in Figure 2d. Figure 2e provides the precision–recall (PR) curves between the proposed model and other models with different feature input combinations. The accuracy and recall rate of each model are obtained through the average of 10-fold cross-validation. The figure shows that the area under the PR curve by our method is larger than the areas under the PR curves by other models with different feature inputs, namely,

the performance of the proposed method is better than that of other models. In the ten-fold cross-validation, the statistically significant difference is analyzed between the proposed model and other models with different feature input combinations. The respective *p*-value in hypothesis testing is less than 0.05, indicating that the proposed model is significantly different compared to the other three models.

Table 1. Performance comparison by different feature inputs for peptide detectability prediction on the test set from GPMDB.

Model	AUC	Accuracy	Specificity	Sensitivity	F-Score
CapsNet + CBAM	0.8692 ± 0.0029	0.8063 ± 0.0027	0.8819 ± 0.0152	0.7308 ± 0.0140	0.7906 ± 0.0051
Without conical feature	0.8412 ± 0.0040	0.7748 ± 0.0040	0.8682 ± 0.0266	0.6814 ± 0.0219	0.7516 ± 0.0090
Without amino acid frequency	0.8615 ± 0.0045	0.7978 ± 0.0042	0.8860 ± 0.0225	0.7096 ± 0.0196	0.7782 ± 0.0084
Without embedding	0.8396 ± 0.0051	0.7755 ± 0.0053	0.8663 ± 0.0201	0.6847 ± 0.0197	0.7531 ± 0.0072

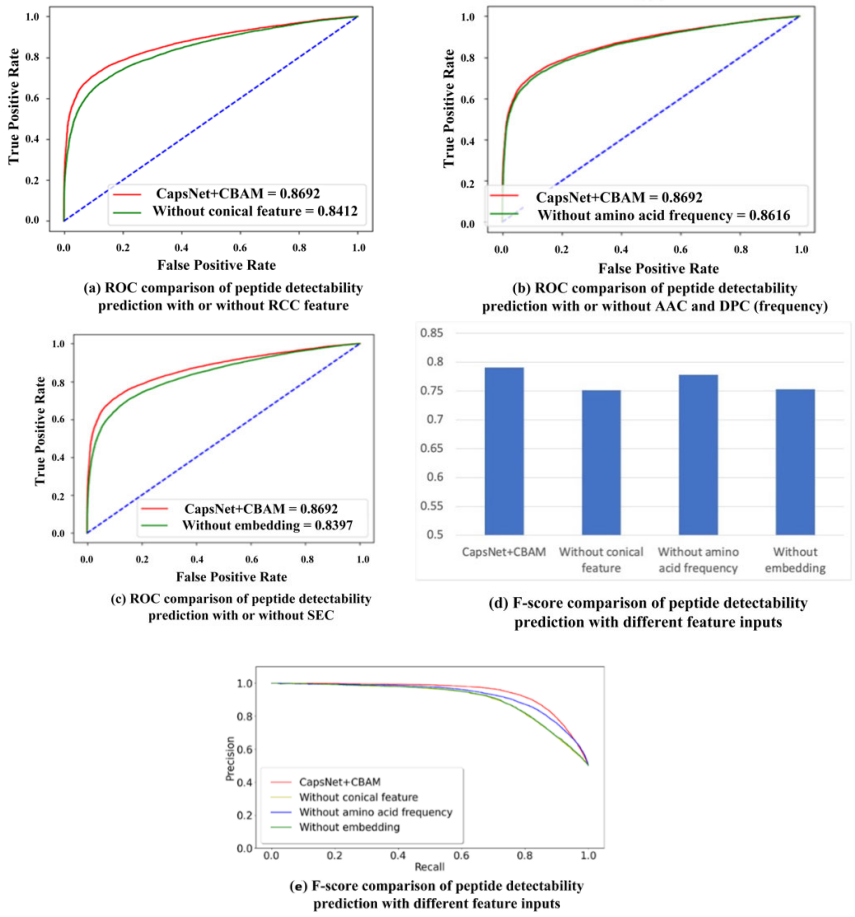

Figure 2. Performance comparison with different feature inputs for peptide detectability prediction on the test set from GPMDB.

2.3. Different Prediction Method Comparison for the GPMDB Dataset Test

The proposed model of integrating CapsNet and CBAM is also compared with other popular prediction methods for the detectability of peptides. The dataset is set up by referencing the 1D-2C-CNN method [15], and the 10,000 tryptic peptides are divided into 75% training set and 25% test set. During the training, the hyperparameters of this study are kept to be identical to those in the previous 10-fold cross-validation training model. Table 2 lists the comparison between the experimentally achieved results and other methods (e.g., the method of 1D-2C-CNN by Serrano et al. [15], RF, SvmR, C5, Pls, and Glm methods by

Guuruceaga et al. [9], the DNN method by Zimmer et al. [11], and the Gaussian method by Mallick et al. [16]). For the proposed method, the mean and standard deviations of 10-fold cross-validation are calculated as the result. For other methods, their trained model is adopted to acquire the prediction result. It is indicated that our AUC is 1.22% higher than that of the previous best model 1D-2C-CNN, and the other three indicators are better than other models (specificity slightly lower than 1D-2C-CNN). In addition, the statistically significant difference analysis is conducted between the proposed method and 1D-2C-CNN. In other words, the *p*-values of ten-fold cross-validations are determined between the proposed model and 1D-2C-CNN method. The results are presented in Figure 3, which suggest that the proposed model is significantly different from the 1D-2C-CNN method (*p*-value < 0.05). We also conduct a PR curve analysis between the proposed method and 1D-2C-CNN. The precision and recall of each model are obtained by averaging through the 10-fold cross-validation. The result is shown in Figure 4. The PR curve by the proposed method mostly covers the curve by 1D-2C-CNN, which indicates that the performance of the proposed model is better than that of 1D-2C-CNN.

Table 2. Performance comparison by different classifier algorithms for peptide detectability prediction on the test set from GPMDB.

Model	AUC	Accuracy	Specificity	Sensitivity	F-Score
CapsNet + CBAM	0.8692 ± 0.0027	0.8050 ± 0.0026	0.8823 ± 0.0147	0.7278 ± 0.0136	0.7887 ± 0.0049
1D-2C-CNN [15]	0.8570	0.7953	0.8880	0.7027	0.7744
RF [9]	0.7549	0.6924	0.7746	0.6103	0.6649
SvmR [9]	0.7384	0.6813	0.7830	0.5797	0.6453
DNN [11]	0.7360	0.6692	0.6813	0.6572	0.6659
C5 [9]	0.7312	0.6644	0.6513	0.6775	0.6687
Pls [9]	0.6350	0.6043	0.6396	0.5690	0.5898
Glm [9]	0.6349	0.6036	0.6426	0.5646	0.5875
Gaussian [16]	0.6342	0.5983	0.6121	0.5845	0.5927

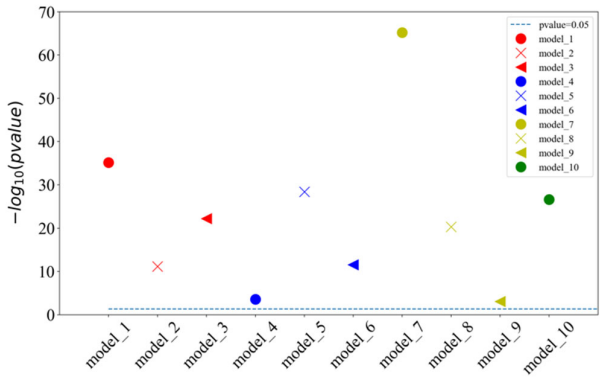

Figure 3. Statistically significant difference analysis of ten-fold cross-validation between our model and 1D-2C-CNN method. Model_1, model_2, . . . , model_10 are the models of CapsNet + CBAM by the ten-fold cross-validation. When *p*-value is less than 0.05, the difference between two models is significant. Since the *p*-values calculated by ten-fold cross-validation models are quite different, we provide the result of $-log_{10}$ (*p*-value) instead of *p*-value in this figure. When the *p*-value is higher than the baseline (*p*-value = 0.05), it indicates that the ten-fold cross-validation of our model is significantly different from the 1D-2C-CNN method.

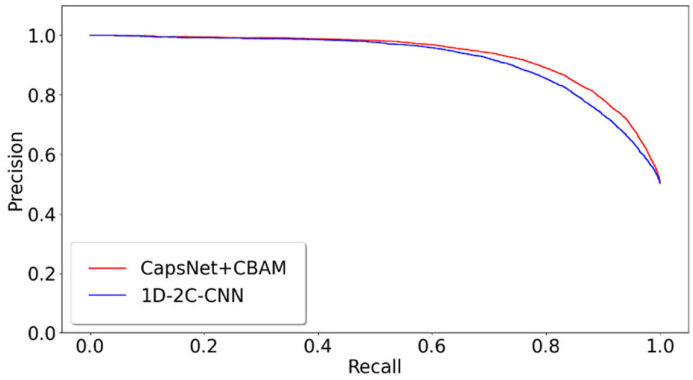

Figure 4. PR comparison of peptide detectability prediction between CapsNet + CBAM and 1D-2C-CNN.

2.4. Additional Benchmarks for Testing

The MS evidence of the human proteome offered by the HPP project is applied, and data are extracted from the neXtProt database [19] to build an additional benchmarking. The peptides of this database (also covered in GPMDB, but not related to the training and test sets of peptides according to Ref [15]) fall into three non-overlapping groups: (1) Proteins with MS evidence (PE1) in the existing version (2019-01-11); (2) PE1 proteins in the current release without MS evidence at the beginning of the HPP project (2011-08-23); (3) Missing Proteins (MPs) in the existing version. A total of 8000 peptide chains are selected from the respective group as three benchmark test sets.

Figure 5a,b show the detectability prediction results of peptides with different protein evidence from the three benchmark test sets. For the comparison, the proposed method on the first two datasets PE1 and Detected MPs shows the high detectability, which is consistent with the prediction performance of 1D-2C-CNN on the whole. However, the average probability values predicted by the proposed method on the PE1 and Detected MPs datasets are 0.7143 and 0.838, respectively, which are higher than those predicted by 1D-2C-CNN (0.7039 and 0.7052, respectively). For the Current MPs dataset, since the peptides in this dataset are considered difficult to detect, the lower the detectability value, the better the prediction method will be. As revealed from the comparison result, the proposed prediction model is better than the 1D-2C-CNN method (the proposed method achieves an average probability value of 0.3058 on the Current MPs dataset, while the prediction probability by 1D-2C-CNN is 0.3288).

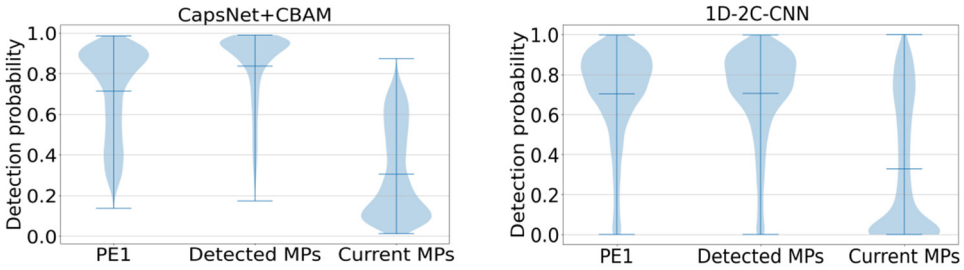

Figure 5. Detectability prediction comparison of peptides with different protein evidence from 3 benchmark test sets. (**a**) Additional benchmark result by our method. (**b**) Additional benchmark result by 1D-2C-CNN.

2.5. Additional Datasets for Testing

For the detectability prediction of peptides, it aims to examine the intrinsic characteristics in the peptide chain to distinguish the detectability of peptides. Lastly, an additional two datasets are applied to test the ability of the proposed neural network for searching the

mentioned features in the peptide chain. Two datasets include anti-angiogenic peptides (AAP) and antibacterial peptides (ABP). We also compared the proposed method with some popular methods (e.g., PEPred-Suite [12], AntiAngioPred [20], and AntiBP [21]), as shown in Figure 6. It is shown that the proposed method is better than the mentioned latest methods in the AUC indicator: the AUC by the proposed method reached 0.811 on the AAP dataset (PEPred-Suited and AntiAngioPred reached 0.804 and 0.742, respectively), and 0.979 on ABP (PEPred-Suite and AntiBP reached the same as 0.976). This verifies that our network model can also be applied for predicting other therapeutic peptides.

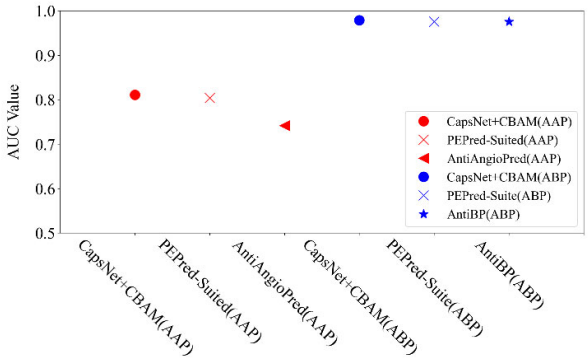

Figure 6. Comparison of AUC indicators by different methods on AAP and ABP datasets.

3. Discussion

This study proposes a neural network model integrated with CapsNet and CBAM to predict the detectability of peptides. It constructs the residue conical coordinate (RCC) feature, amino acid composition (ACC) and dipeptide composition (DPC), and sequence embedding code (SEC) to generate peptide chain features for the proposed network. For the mentioned features, they fall to biological feature and sequence feature by separate inputting to reduce the influence among the mentioned features. When using the CapsNet network, it reduces the impact of data loss in the pooling layer after convolution. In addition, a CBAM attention module is added to assign weights to channels and spaces to learn important features. The proposed model is compared with other extensively used deep learning frameworks (e.g., 1D-2C-CNN [15], RF, SvmR, DNN, C5, Pls, Glm, and Gaussian [9,11,16]). It is suggested to outperform other methods in main assessment indicators (AUC, accuracy, specificity, sensitivity, and F-score), which verifies the effectiveness of the proposed method. Consequently, it can act as a valid supplementary method for peptide detectability prediction and be applied in proteomics and other related fields.

Since the length of the peptide chain is short and its length is different, the proposed method selects a truncated fixed length for calculation. It will be our future work to find a feature extraction method that is more suitable for characteristics with different peptide chain lengths. In addition, this method selects two biological features and directly joins them together. We need to extract other biological features and explore a valid feature fusion method for peptide detectability prediction so the mentioned biological features can be more effectively integrated for training. With the growing popularity of neural networks, we also look for a more suitable network model for the classification of peptide chains. In addition, the CapsNet + CBAM model is an end-to-end model. It is expected that this model can be employed to predict other peptide chains (e.g., cell-penetrating peptides).

4. Materials and Methods

4.1. Dataset

The dataset applied for training and testing in this study originates from the GPMDB database [3], involving mass spectrometry data and detection frequencies for proteins identified by mass spectrometry. The method of generating the dataset was proposed

by Guruceaga et al. [9], capable of classifying trypsin-digested peptides according to the number of peptides observed in proteomics experiments and comparing the characteristics exhibited by the most observed peptides and the less observed peptides. A total of 100,000 tryptic peptides are selected as the dataset, and then the ten-fold cross-validation is performed for the experiment (i.e., it is divided into 10 parts, taking turns using 9 parts as the training data and 1 part as the test data for repeating the experiment). The respective test will obtain the corresponding prediction result. The average of the prediction results by 10 times is used as the final prediction result.

4.2. Feature Selection

Feature selection refers to a vital step in the deep learning of a neural network framework. The property of amino acids and sequence information are exploited to generate three features, i.e., the residue conical coordinate (RCC) feature, amino acid composition (AAC) and dipeptide composition (DPC), as well as the sequence embedding code (SEC) feature.

4.2.1. RCC Feature Based on Physicochemical Properties of Amino Acids

The respective amino acid has specific physicochemical properties, thereby affecting the characteristics of the peptide and being critical in determining the structure and function of the peptide. The 100 physicochemical properties of amino acids are drawn upon for feature extraction. Due to the significant numerical difference between the physicochemical properties, the standardized operation is exploited to process the data and improve the comparability of data. Furthermore, since some amino acids exhibit consistent characteristics, the mentioned amino acids are classified into 4 types [17] (Table 3).

Table 3. Classification of 20 amino acids.

Groups	Description	Amino Acids
Class I	Non-polar residues	A, V, L, I, P, F, W, M
Class II	Polar residues	G, S, T, C, Y, N, Q
Class III	Basic residues	K, R, H
Class IV	Acidic residues	D, E

Zhang et al. [16] proposed a residue conical coordinate (RCC) feature by mapping the respective amino acid to a point in a three-dimensional (3D) space. Subsequently, they integrated this spatial coordinate feature with the RF method to conduct a satisfying prediction for carbonylation site identification.

To be specific, to obtain the vital features of proteins with the use of a simple and effective mapping method, the following two hypotheses were proposed. (1). The amino acids in the same group are distributed on an identical conical surface since they exhibit consistent characteristics. The amino acids in the same group are distributed on the same conical surface because they show similar characteristics. For instance, as eight amino acids (A, V, L, I, P, F, W, and M) pertain to Class I, they are mapped on the identical cone surface ($\varphi = \varphi_1$) (Figure 7). Likewise, the other three types of amino acids are mapped on their respective cones. (2). To reveal the difference between amino acids, the radius vector length r of the conical surface is set in accordance with the molecular weight of the corresponding amino acid (http://lin-group.cn/server/iCarPS/download.html (accessed on 4 November 2021)). Following such a mapping rule, the similarity of similar amino acids and the difference between different amino acids can be indicated simply and effectively.

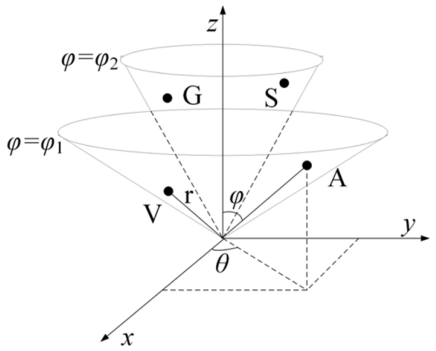

Figure 7. Schematic illustration of the 3-dimensional conical representation for characterizing amino acid residues. A, V, G, and S are the first two amino acids in the Non-polar residue group and Polar residue group. φ_1 and φ_2 represent the conical surface, which is formed by projection of amino acids in the corresponding group.

Therefore, by combining the physicochemical properties and the above mapping rules, each amino acid can be converted into a three-dimensional vector. The formula is as follows:

$$\begin{cases} x_{ij} = r_{ij} \times \sin\varphi_i \times \cos\theta_{ij} \\ y_{ij} = r_{ij} \times \sin\varphi_i \times \sin\theta_{ij} \\ z_{ij} = r_{ij} \times \cos\varphi_i \end{cases} \quad \varphi_i \in [0,\pi],\ \theta_{ij} \in [0,2\pi] \tag{1}$$

where r_{ij} expresses the molecular weight of the j-th residue $(j = 1, 2, \cdots, L_i)$ in the i-th category $(i = 1, 2, 3, 4)$ of the amino acid classification; L_i denotes the number of amino acids in the i-th group; and φ_i, θ_{ij} are defined below:

$$\varphi_i = \pi \times \left|\sin \frac{\overline{d}_i}{\left(\frac{1}{4}\sum_1^4 \overline{d}_i\right) \times \sqrt{\frac{1}{4}\sum_1^4 \left(\overline{d}_i - \frac{1}{4}\sum_1^4 \overline{d}_i\right)^2}}\right| \tag{2}$$

$$\theta_{ij} = \pi + 2 \times \tan^{-1} \frac{\sum_{m=1}^{9} PC_{jm} - \overline{d}_i}{\sqrt{\frac{1}{L_i}\sum_{j=1}^{L_i} \left|\sum_{m=1}^{9} PC_{jm} - \overline{d}_i\right|^2}} \tag{3}$$

where $\overline{d}_i = \frac{1}{L_i}\sum_{j=1}^{L_i}\sum_{m=1}^{9} PC_{jm}$; PC_{jm} represents the m-th $(m = 1, 2, \cdots, 9)$ physicochemical properties of the j-th $(j = 1, 2, \cdots, L_i)$ residue in the i-th category of amino acid classification. In the mentioned conversion process, since only 9 physicochemical properties of amino acids are applied for the calculation, the yielded feature matrix is relatively small for the proposed deep learning network model. This study improves the construction of the RCC feature that fully extracts the physicochemical properties of the residue. In other words, 100 physicochemical properties fall into 10 groups, each of which covers 10 physicochemical properties [13]. Subsequently, 10 sets of residue conical coordinate features are yielded by Equations (1)–(3). Such a measure is capable of increasing the feature dimension generated by the peptide chain and ensuring the feature proportion in the proposed neural network training. Furthermore, we combine statistical information (e.g., variance, skewness, and kurtosis) to extract features exhibited by peptide chains.

To be specific, the L-length peptide chain can be transformed as a $1 \times 3L$ matrix (e.g., $P = [x_1, y_1, z_1, \cdots, x_L, y_L, z_L]^{\mathrm{T}}$). The overall geometric center of the sequence $\overline{u}(\overline{x}, \overline{y}, \overline{z})$ is first determined as:

$$\overline{u} = \frac{1}{L}\sum_{k=1}^{L} u_k \tag{4}$$

where u_k represents the k-th 3D coordinate point $(u_k = x_k, y_k, z_k)$.

Subsequently, the residues in the peptide chain are classified into 4 types (Table 3), and the geometric center $\overline{u}_i$ of the i-th category is determined $(i = 1, 2, 3, 4)$:

$$\overline{u}_i = \frac{1}{v} \sum_{n=1}^{v} u_{in} \tag{5}$$

where u_{in} denotes the coordinates of the n-th residue pertaining to the i-th category of amino acid classification; v represents the total number of the i-th residue category.

Lastly, the overall geometric center of the peptide chain and the geometric center of the amino acid classification are adopted to determine the statistical characteristics of variance, skewness, and kurtosis [13] as:

$$\sigma^2 = \frac{1}{4} \sum_{i=1}^{4} (\overline{u}_i - \overline{u})^2 \tag{6}$$

$$g = \frac{\frac{1}{4} \sum_{i=1}^{4} (\overline{u}_i - \overline{u})^3}{\sigma^3} \tag{7}$$

$$h = \frac{\frac{1}{4} \sum_{i=1}^{4} (\overline{u}_i - \overline{u})^4}{\sigma^2} \tag{8}$$

For each peptide chain, Equation (5) is used to generate a 1×12 dimensional vector according to the amino acid classification (4 types), and Equation (4) and Equations (6)–(8) are adopted to generate a 1×12 dimensional vector, so the total combination is a 1×24 dimensional vector. Since there are 10 groups of physicochemical properties exhibited by amino acids, a 1×240 dimensional feature vector is lastly obtained.

4.2.2. Amino Acid Composition and Dipeptide Composition

Amino acid composition (AAC) acts as a vital feature of a single amino acid, and dipeptide composition (DPC) represents the proportion information of adjacent amino acids in the peptide chain. Both of the mentioned features can indicate the property of the entire peptide chain [22,23]. AAC denotes the percentage of a single amino acid in a set peptide chain and can be calculated by:

$$\mathrm{AAC}(i) = \frac{\text{Frequency of amino acid}(i)}{\text{Length of the peptide}} \quad i = (1, 2 \cdots 20) \tag{9}$$

where i represents the respective amino acid, and the length of the AAC generation vector is 1×20.

Because AAC does not consider the order of amino acids in the peptide chain, the DPC feature is added in our feature information. DPC denotes the probability of two adjacent amino acids appearing in the entire peptide chain, which has a fixed length of 400 features, as expressed below:

$$\mathrm{DPC}(j) = \frac{\text{Total number of dipeptides}(j)}{\text{Total number of all possible dipeptides}} \tag{10}$$

where j represents one of 400 dipeptide compositions.

4.2.3. Neural Network Embedding

For biological sequences, the one-hot encoding is the most used feature as input in the neural networks [24]. When processing the peptide chain, the one-hot encoding turns out to be a 20-dimensional vector composed of 0 and 1, which is sparse and fails to show the neighboring relationship of the respective residue. Here, the sequence embedding code (SEC) [15] is adopted to calculate the neighboring relationship of the residues and control the dimensionality of the vector. SEC acts as a method to convert discrete variables into a continuous vector representation. Since the SEC is learnable, the representations of more similar residues will be closer to each other in the embedding space during the continuous

training process. For this reason, the SEC is selected as the feature input to participate in training in the neural network, and the respective amino acid embedding is set as a 20-dimensional vector.

4.3. Neural Network Architecture

Convolutional Neural Network (CNN) refers to an extensively applied network framework. Its capacity of solving classification is achieved primarily by using the convolutional and pooling layers to reduce the spatial size of data passing through the network. On that basis, the field of view of the neurons in the neural network increases, and the high-level features of the input area are detected. However, there exist two main problems in such a process. First, the convolution is locally connected and parameter-sharing, while it does not consider the correlation and mutual positional relationship between different features. Second, it keeps only the most active neurons in the process of maximum pooling and pushes them to the next layer, thereby causing the loss of valuable spatial information.

This study uses the capsule neural network (CapsNet) [17] to solve the mentioned problems. The difference between this neural network and the general neural network is that its neuron is a vector (a set of values) (vector neuron) rather than a scalar (single value). The vector is capable of representing a wide range of characteristics of a protein, and the modulus length of the vector can be exploited to measure the probability of the respective category. The greater the modulus value, the greater the probability of pertaining to a set category will be. Compared with CNN neurons with scalar input and output, the input and output vectors of the capsule network cover more feature space information. Furthermore, the capsule network substitutes the maximum pooling with a dynamic routing mechanism. Compared with the popular maximum pooling, it can retain the weighted sum of the features of the previous layer, whereas the maximum pooling only retains the most active neurons.

The proposed neural network model comprises a CNN convolutional layer, a CBAM module, a convolutional capsule layer (PrimaryCaps), as well as a fully connected capsule layer (BindCaps) (Figure 8). First, the RCC, AAC, and DPC features of the peptide chain are employed as the biological features, and the SEC feature is inputted into the CNN layer as the sequence feature. The mentioned features are convolved via CNN to extract local feature information and then inputted to the CBAM module. CBAM combines the channel attention mechanism and the spatial attention mechanism [18]. In the CBAM module, the local feature information will be multiplied by the different weights trained, which can arouse the attention to vital feature information and suppress the unimportant feature information. The CBAM module is presented in Section 2.4. Moreover, the detailed feature transfer in the neural network is expressed in Section 4.5.

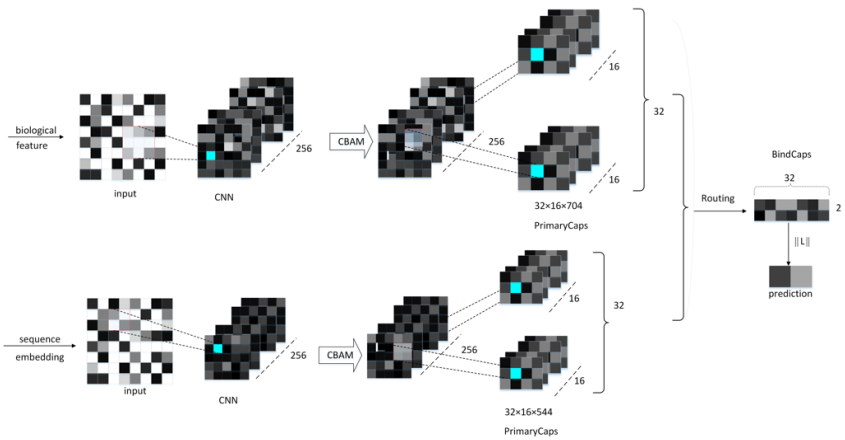

Figure 8. The neural network framework of the proposed method.

Subsequently, the biological and sequence features processed by CBAM are converted into 32 × 16 × 544, 32 × 16 × 704 feature maps and then inputted to the PrimaryCaps layer. To be specific, 32 represents the number of peptide chains input in one training, 16 indicates the number of channels in the PrimaryCaps layer, and 544 and 704, respectively, represent the size of the biological and sequence feature information obtained after CBAM. For the feature map corresponding to the respective peptide chain, the calculation of the iterative dynamic routing between the PrimaryCaps and BindCaps layers is illustrated in Figure 9. To be specific, the PrimaryCaps layer comprises 1248 (544 + 704) capsules u_i (each u_i denotes a 16-dimensional vector), and each capsule is multiplied by the weight matrix $W_{I,j}$ (the size 16 × 32, $i = 1, \ldots, 1248$, $j = 1,2$) to obtain $\hat{u}_{i|j}$. $\hat{u}_{i|j}$ is multiplied by the weighted sum of the parameters $c_{i,j}$ to yield s_j. Subsequently, through the nonlinear "Squeeze" function, the capsule vector in BindCaps is obtained after a set number of iterations (epoch). To be specific, the "squeeze" function [15] is adopted to scale the length of the output vector of each capsule to between [0, 1] and keep the direction unchanged, as expressed below:

$$v_j = \frac{||s_j||^2}{1 + ||s_j||} \cdot \frac{s_j}{||s_j||} \tag{11}$$

where s_j and v_j, respectively, represent the input and output of the j-th capsule.

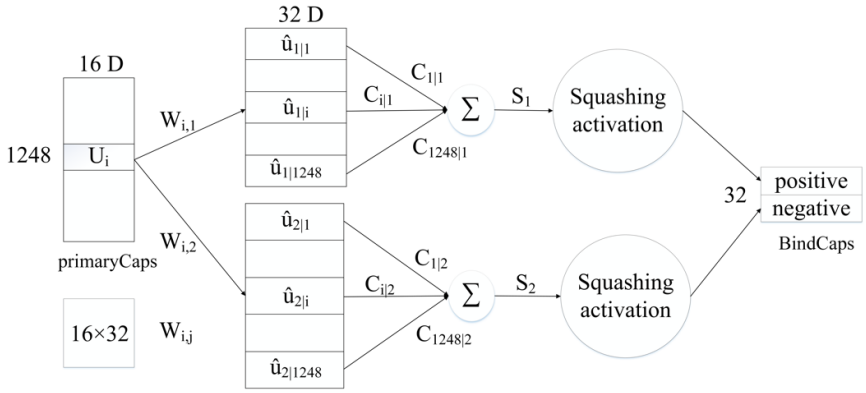

Figure 9. Routing calculation process between PrimaryCaps and BindCaps.

Lastly, the two output capsules calculated by using the routing algorithm are integrated to obtain two 32-dimensional BindCaps, corresponding to two possible markers of the input protein sequence, i.e., peptide detectability and undetectability. The modulus of the 32-dimensional vector represents the detectability and undetectability probability of the peptide. To be specific, the L2-norm of two 32-dimensional vectors is calculated, and the detectability of the peptide is predicted by comparing the magnitude of the two probability values.

In the proposed neural network framework, the loss function adopted to train the CapsNet model is the sum of two separate losses in BindCaps. For each BindCaps, the separate loss L_k is expressed as:

$$L_k = T_k \max(0, 0.9 - ||v_k||)^2 + 0.5(1 - T_k)\max(0, ||v_k|| - 0.1)^2 \tag{12}$$

where v_k denotes the vector output by BindCaps. If the peptide is detectable, $T_k = 1$; otherwise, it is 0.

4.4. CBAM Module

In the proposed neural network, to extract vital features distinguishing the detectability of peptides, the CBAM module is added after the first layer of convolution. CBAM refers to an attention mechanism module combining space and channel information. The spatial module assigns different weights to the identical dimension of biological features

and sequence features, while the channel attention assigns different weights to the feature map channels after convolution, which is inconsistent with SENet [25] that only focuses on the channel attention module. CBAM focuses on both space and channel information, so it can more effectively predict the detectability of peptides. The specific process of this module is that the first layer of CNN processes the feature map $F \in R^{C \times H \times W}$ as the input. Subsequently, the channel attention module $M_C(F)$ and the spatial attention module $M_S(F)$ of CBAM are employed to set different weights to the feature map (Figure 10). The complete process can be written as:

$$\begin{aligned} F' &= M_C(F) \times F \\ F'' &= M_S(F') \times F' \end{aligned} \tag{13}$$

where $\times$ denotes the element-wise multiplication.

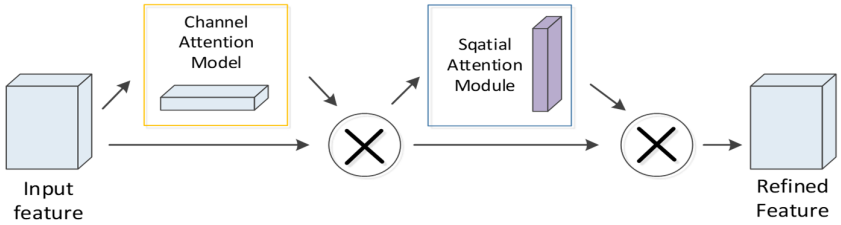

Figure 10. CBAM module.

According to the CBAM module, the channel attention module is adopted to compress the feature map of the peptide in the spatial dimension to yield a one-dimensional vector. Next, the vector is multiplied with the feature map of the initial input peptide element by element (Figure 11). During channel compressing in the spatial dimension, two pooling operations are considered, i.e., the average pooling F^c_{avg} and the maximum pooling F^c_{max}. They can be exploited to aggregate the spatial information of the feature-mapped peptides, and then the information is transmitted to a shared fully connected network. The size of the hidden layer is set to $R^{\frac{C}{r} \times 1 \times 1}$, where r denotes the reduction rate. The calculation formula is:

$$M_C(F) = \sigma(\mathrm{MLP}(\mathrm{AvgPool}(F)) + \mathrm{MLP}(\mathrm{MaxPool}(F))) \tag{14}$$

where σ denotes the sigmoid function and F represents the peptide feature after CNN processing. For the feature map after convolution, the channel places its stress on which channel is critical to the detectability of peptides, and average pooling has feedback for each pixel on the feature map. In addition, maximum pooling is applied for the gradient backpropagation calculation, and only the place with the largest response in the feature map achieves the gradient feedback.

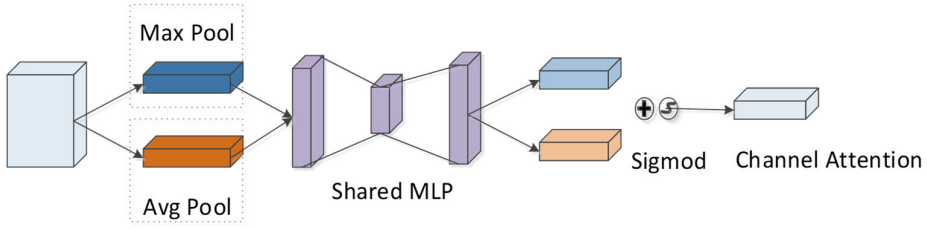

Figure 11. Channel attention module in CBAM.

The spatial attention module is exploited to compress the channel (Figure 12). Two pooling (average pooling and maximum pooling) operations are used to aggregate the channel information of the feature map and then merged to yield a 2-channel feature map. After the convolutional layer is passed through and activation function (sigmoid function)

operation is achieved, the weight coefficient is obtained and then multiplied with the first input feature to obtain the final output feature. The calculation formula is written below:

$$M_S(F) = \sigma\left(f^{7\times7}([\text{AvgPool}(F); \text{MaxPool}(F)])\right) \quad (15)$$

where σ denotes the sigmoid function, $f^{7\times7}$ represents the convolution operation, and the filter size is 7×7. For spatial attention, the focus is placed on which part of the corresponding convolution feature map of a peptide is critical to predicting the detectability of the peptide. Up-regulating the weight of the more important areas of the convolution feature map can optimize the detectability exhibited by the peptide.

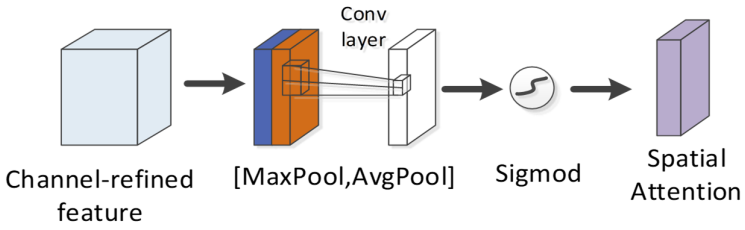

Figure 12. Spatial attention module in CBAM.

4.5. Network Parameter Setting and Feature Transfer

In the proposed framework, the corresponding parameter setting and the transfer process of features in the neural network are expressed below. The batch size input for training is set to 32 peptide chains. At the first network layer, the feature size of the biometric input is $32 \times 33 \times 20$, the convolution kernel is 9×9 in size, and the resulting feature input is $32 \times 256 \times 27 \times 14$. The sequence feature input has the feature size of $32 \times 50 \times 20$, the size of the convolution kernel is 7×7, and the resulting feature input is $32 \times 256 \times 42 \times 12$. At the second layer of the CBAM attention module, the reduction rate is set to 16, and the activation function is ReLU under the channel attention module, while the size of the first convolution kernel layer is 7×7 according to the spatial attention module. Based on the attention module, the feature input and output dimensions remain constant. At the third layer of PrimaryCaps, it is implemented by using 16 filters with the size of 9 and 7, respectively, and the stride is set to 2 in the respective capsule. The tensor generated by the sequence feature is $32 \times 256 \times 17 \times 2$, and the tensor generated by the biological feature is $32 \times 256 \times 11 \times 4$. The output feature sequence is changed to $32 \times 16 \times 544$, and the biological feature is altered to $32 \times 16 \times 704$, where 32 represents the training data, 16 denotes the attribute of each sequence trained by the capsule network, and 544 and 704 express the data trained by different weights. Next, at the fourth layer BindCaps, the features are calculated as two 32-dimensional vectors by using the routing algorithm, corresponding to the two classifications applied for prediction. The Adam optimizer [26] is used for the training, the number of epochs is set to 6, the initial learning rate is set to 0.001, and the loss function is set by complying with Equation (12). The ε hyperparameter of the capsule network is set to 1×10^{-8}, the routing parameter is set to 3, and the reduction parameter of the squeeze-and-excitation (SE) layer in CBAM module is set to 16.

4.6. Evaluation Index

The detectability prediction of peptides falls into 4 types, i.e., (1) True Positive (TP): the detectability of the detected peptide, consistent with the detectability of the actual peptide; (2) False Positive (FP): the detectability of the detected peptide, whereas the match set by the undetectability of the actual peptide is incorrect; (3) True Negative (TN): the undetectability of the detected peptide, complying with the undetectability of the actual peptide; (4) False Negative (FN): the undetectability of the detected peptide, as well as the detectability of the actual peptide.

The assessment indicators involved in this study include the area under the ROC curve (AUC), accuracy, specificity, sensitivity, and F-score [27]. To be specific, AUC is defined as the area under the ROC curve. The value of this area will not exceed 1. Since the ROC curve is generally above the line $y = x$, the value range of AUC is generally between 0.5 and 1. The AUC value acts as the assessment criterion since in many cases the ROC curve does not clearly indicate which classifier performs better; however, as a value, a classifier with a larger AUC is more effective. Sensitivity: The degree of a positive reaction to a real target. The higher the sensitivity, the easier it will be to identify the target. Specificity: The degree of negative reaction to false targets. The higher the specificity, the less likely there will be false positives, and there are only positive reactions for specific scenarios, i.e., strong screening ability or high pertinence. F-score refers to an index applied in statistics to measure the accuracy of a two-classification model. It considers the accuracy rate and recall rate of the classification model. F-score can be considered a harmonic average of model accuracy and recall that ranges from 0 to 1. The calculation formulas for the above assessment indicators are presented below:

$$\text{Accuracy} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{16}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{17}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}} \tag{18}$$

$$\text{F} - \text{score} = 2 \cdot \frac{\text{Accuracy} \cdot \text{Recall}}{\text{Accuracy} + \text{Recall}} \tag{19}$$

Author Contributions: Conceptualization, M.Y., Y.D. and Z.L.; methodology, Y.D., Z.L. and Y.Z.; software, M.Y.; validation, M.Y., Y.D. and Z.L.; formal analysis, Z.L.; writing—original draft preparation, M.Y. and Y.D.; writing—review and editing, Z.L. and Y.Z.; funding acquisition, Z.L. All authors contributed equally to the article. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under Grant No. 12171434, Zhejiang Provincial Natural Science Foundation of China under Grant No. LZ19A010002.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets and program used in this study are available at https://github.com/yuminzhe/yuminzhe-Prediction-of-peptide-detectability-based-on-CapsNet-and-CBAM-module (accessed on 4 November 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Nilsson, T.; Mann, M.; Aebersold, R.; Yates, J.R.; Bairoch, A.; Bergeron, J. Mass spectrometry in high-throughput proteomics: Ready for the big time. *Nat. Methods* **2010**, *7*, 681–685. [CrossRef]
2. Alves, P.; Arnold, R.J.; Novotny, M.V.; Radivojac, P.; Reilly, J.P.; Tang, H. Advancement in protein inference from shotgun proteomics using peptide detectability. *Biocomputing* **2007**, *12*, 409–420.
3. Craig, R.; Cortens, J.P.; Beavis, R.C. Open source system for analyzing, validating, and storing protein identification data. *J. Proteome Res.* **2004**, *3*, 1234–1242. [CrossRef] [PubMed]
4. Jarnuczak, A.F.; Lee, D.; Lawless, C.; Holman, S.W.; Eyers, C.E.; Hubbard, S.J. Analysis of intrinsic peptide detectability via integrated label-free and srm-based absolute quantitative proteomics. *J. Proteome Res.* **2016**, *15*, 2945. [CrossRef] [PubMed]
5. Li, Y.F.; Arnold, R.J.; Tang, H.; Radivojac, P. The importance of peptide detectability for protein identification, quantification, and experiment design in ms/ms proteomics. *J. Proteome Res.* **2010**, *9*, 6288–6297. [CrossRef] [PubMed]
6. Cheng, H.; Rao, B.; Liu, L.; Cui, L.; Xiao, G.; Su, R.; Wei, L. PepFormer: End-to-End Transformer-Based Siamese Network to Predict and Enhance Peptide Detectability Based on Sequence Only. *Anal. Chem.* **2021**, *93*, 6481–6490. [CrossRef]

7. Le, N.Q.K.; Huynh, T.T. Identifying SNAREs by incorporating deep learning architecture and amino acid embedding representation. *Front. Physiol.* **2019**, *10*, 1501. [CrossRef] [PubMed]
8. Le, N.Q.K.; Ho, Q.T.; Nguyen, T.T.D.; Ou, Y.Y. A transformer architecture based on BERT and 2D convolutional neural network to identify DNA enhancers from sequence information. *Briefings Bioinf.* **2021**, *5*, 5.
9. Guruceaga, E.; Garin-Muga, A.; Prieto, G.; Bejarano, B.; Marcilla, M.; Marín-Vicente, C.; Segura, V. Enhanced missing proteins detection in nci60 cell lines using an integrative search engine approach. *J. Proteome Res.* **2017**, *16*, 4374–4390. [CrossRef]
10. Shuichi, K.; Piotr, P.; Maria, P.; Andrzej, K.; Toshiaki, K.; Minoru, K. Aaindex: Amino acid index database, progress report 2008. *Nucleic Acids Res.* **2008**, *36*, D202–D205.
11. Zimmer, D.; Schneider, K.; Sommer, F.; Schroda, M.; Mühlhaus, T. Artificial intelligence understands peptide observability and assists with absolute protein quantification. *Front. Plant Sci.* **2018**, *9*, 1559. [CrossRef]
12. Wei, L.; Zhou, C.; Su, R.; Zou, Q. PEPred-Suite: Improved and robust prediction of therapeutic peptides using adaptive feature representation learning. *Bioinformatics* **2019**, *35*, 4272–4280. [CrossRef]
13. Zhang, Y.P.; Zou, Q. PPTPP: A novel therapeutic peptide prediction method using physicochemical property encoding and adaptive feature representation learning. *Bioinformatics* **2020**, *36*, 3982–3987. [CrossRef] [PubMed]
14. Hinton, G.E.; Salakhutdinov, R.R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507. [CrossRef]
15. Serrano, G.; Guruceaga, E.; Segura, V. DeepMSPeptide: Peptide detectability prediction using deep learning. *Bioinformatics* **2020**, *36*, 1279–1280. [CrossRef]
16. Zhang, D.; Xu, Z.C.; Su, W.; Yang, Y.H.; Lv, H.; Yang, H.; Lin, H. iCarPS: A computational tool for identifying protein carbonylation sites by novel encoded features. *Bioinformatics* **2021**, *37*, 171–177. [CrossRef]
17. Sabour, S.; Frosst, N.; Hinton, G.E. Dynamic routing between capsules. *arXiv* **2017**, arXiv:1710.09829.
18. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
19. Gaudet, P.; Michel, P.A.; Zahn-Zabal, M.; Britan, A.; Cusin, I.; Domagalski, M.; Bairoch, A. The neXtProt knowledgebase on human proteins: 2017 update. *Nucleic Acids Res.* **2017**, *45*, D177–D182. [CrossRef] [PubMed]
20. Ramaprasad, A.S.; Singh, S.; Gajendra, P.S.R.; Venkatesan, S. AntiAngioPred: A server for prediction of anti-angiogenic peptides. *PLoS ONE* **2015**, *10*, e0136990.
21. Lata, S.; Sharma, B.K.; Raghava, G.P. Analysis and prediction of antibacterial peptides. *BMC Bioinf.* **2007**, *8*, 263. [CrossRef]
22. Manavalan, B.; Subramaniyam, S.; Shin, T.H.; Kim, M.O.; Lee, G. Machine-learning-based prediction of cell-penetrating peptides and their uptake efficiency with improved accuracy. *J. Proteome Res.* **2018**, *17*, 2715–2726. [CrossRef]
23. Gautam, A.; Chaudhary, K.; Kumar, R.; Sharma, A.; Kapoor, P.; Tyagi, A.; Raghava, G.P. In silico approaches for designing highly effective cell penetrating peptides. *J. Transl. Med.* **2013**, *11*, 74. [CrossRef]
24. Rodriguez, P.; Bautista, M.; Gonzalez, J.; Escalera, S. Beyond one-hot encoding: Lower dimensional target embedding. *Image Vision Comput.* **2018**, *75*, 21–31. [CrossRef]
25. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
26. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
27. Le, N.Q.K.; Do, D.T.; Hung, T.N.K.; Lam, L.H.T.; Huynh, T.T.; Nguyen, N.T.K. A computational framework based on ensemble deep neural networks for essential genes identification. *Int. J. Mol. Sci.* **2020**, *21*, 9070. [CrossRef] [PubMed]