Deep_Hybrid_CPP: Robust and Novel Deep Hybrid Learning Approach for Identification of Cell Penetrating Peptide using Multiview Feature Fusion

By
Md. Abu Saleh
211-15-3993

FINAL YEAR DESIGN PROJECT REPORT

This Report is Presented in Partial Fulfillment of the Requirements forthe **Degree of Bachelor of Science in Computer Science and Engineering**

Supervised by

Nazmun Nessa Moon
Associate Professor

Department of Computer Science and Engineering,
Daffodil International University

Co-Supervised by

Hefzul Hossain Papon

Lecturer
Department of Computer Science and Engineering,
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

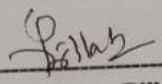
May 14, 2025

APPROVAL

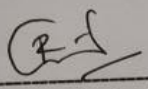
This Project titled **"Deep_Hybrid_CPP: Robust and Novel Deep Hybrid Learning Approach for Identification of Cell Penetrating Peptide using Multiview Feature Fusion"**, submitted by Md. Abu Saleh, ID No: **211-15-3993,** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **14 May, 2025**.

BOARD OF EXAMINERS

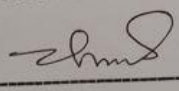
Dr. Sheak Rashed Haider Noori — **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



Md Masum Billah (MMB) — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



Mr. Partha Dip Sarkar (PDS) — **Internal Examiner**
Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



Dr. Md. Zulfiker Mahmud (ZM) — **External Examiner**
Professor
Department of Computer Science and Engineering
Jagannath University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Nazmun Nessa Moon, Associate Professor**, Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this projectnor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Nazmun Nessa Moon
Associate Professor
Department of Computer Science and Engineering,
Daffodil International University

Co-Supervised by:

16.5.25

Hefzul Hossain Papon
Lecturer
Department of Computer Science and Engineering,
Daffodil International University

Submitted by:

A. Saleh

Md. Abu Saleh
Student ID:211-15-3993
Department of Computer Science and Engineering,
Daffodil International University

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. We are deeply grateful to everyone who has assisted us in one way or another.

First, we express our heartfelt thanks and gratefulness to the Almighty for His divine blessing, making it possible for us to complete the **Final Year Design Project (FYDP)** successfully.

We are grateful and wish to express our profound indebtedness to **Nazmun Nessa Moon**, **Associate Professor**, Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of Bioinformatics to carry out this project. His endless pa- patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting themat all stages have made it possible to complete this project.

We would like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering for his kind help in finishing our project, and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

We would like to thank our entire course mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

CPPs are short peptide sequences that have the ability to pass through cell membranes and deliver a range of molecular cargoes within cells. Their remarkable ability to enter cells without inflicting significant harm to the membrane makes them crucial for uses like as intracellular imaging, targeted drug delivery, and gene therapy. I introduced Deep_Hybrid_CPP, an innovative computational approach for efficiently and effectively identifying cell-penetrating peptides through a multi-view feature fusion framework. This study integrates 10 diverse sequence-based feature extractors from different perspectives with 12 prominent machine learning (ML) algorithms in Deep_Hybrid_CPP to create multi-view features that thoroughly represent the essential information of cell-penetrating peptides. To enhance the distinguishing capability of my tailored genetic algorithm, Additionally employed it to select a collection of multi-view features. Based on a series of comparative experiments, my multi-view features outperformed certain traditional feature extractors regarding their discriminative capabilities. Additionally, regarding the independent test dataset, Deep_Hybrid_CPP achieved the top accuracy (ACC) and Matthew's correlation coefficient (MCC) of 97% and 94%, respectively, showing increases of 5.06% and 10.97%. Using this study's innovative computational approach, I expect to effectively evaluate and prioritize candidate peptides that may demonstrate favorable cell-penetrating peptide characteristics.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter offers an in-depth examination of the evolution of the Deep_Hybrid_CPP framework, detailing its introduction, motivations, objectives, methodology, and project outcome. It emphasizes the combination of Multiview feature extraction with hybrid deep learning to improve the prediction accuracy of cell-penetrating peptides.

1.1 Introduction

Cell Penetrating Peptide (CPP) is a small protein or peptide that can enter and cross the cell membrane. Due to this ability, they are referred to as "cell-penetrating". The primary role of CPP is to deliver drugs, genes, proteins, or any other biomolecule into the cell to enhance the efficacy of a wide range of therapies. One of the primary benefits of CPP utilization is that it can deliver therapeutic molecules or drugs directly into target cells, making the treatment more effective and less prone to side effects. If CPP is not used, the majority of drugs are unable to penetrate the cells, and the treatment does not work, or a huge dose must be given. Large doses mean more side effects, which are dangerous for the patient. CPP has revolutionized medicine. CPP is being used to target cancer, neurodegenerative disorders, infectious diseases, and genetic diseases (like cystic fibrosis). In the treatment of cancer, CPP is employed to deliver chemotherapy drugs specifically to cancer cells and reduce damage to normal cells [11][12]. Likewise, in gene therapy, CPP is employed to deliver CRISPR-Cas9 technology to cells' function by binding to the cell membrane and delivering drugs or genes into the cell via endocytosis or direct penetration [13][15]. This enhances treatment efficacy and minimizes treatment failure. CPPs can be utilized for the efficient delivery of mRNA vaccines into cells. In the absence of CPP, new therapy development would be hindered. This is because most drugs or biological molecules cannot enter cells, and this will be a limiting factor for gene therapy, protein therapy, or mRNA-based therapies. In the absence of CPP, the efficacy of treatment can be diminished, and higher doses of the drug need to be administered to be effective, which will increase side effects.CPP has enabled treatments to become more effective, safer, and more sophisticated[14][16]. It is a major technology in modern medicine that will play a critical part in the development of more advanced therapies in the years ahead.

1.2 Motivation

In the field of computational biology, particularly in peptide classification, a significant challenge is the effective capture of complex, nonlinear patterns present in biological sequences. Traditional machine learning methods are not very robust and lack generalization, which is essential to tackle the inherent complexity and variability of heterogeneous datasets. As a means to circumvent these drawbacks, Multiview feature design has been an effective strategy, enabling the extraction and combination of complementary information from various representations of the same data. Not only does this strategy augment the feature space, but it also enhances the model's ability to identify hidden relationships and subtle patterns that would otherwise not be detectable. Additionally, the utilization of hybrid deep learning architectures, which take advantage of the strengths of different deep learning models, offers a promising avenue towards developing more adaptive, intelligent, and reliable predictive systems. Collectively, these developments are crucial for performance optimization and achieving greater accuracy and robustness for predictive modeling with biological relevance.

1.3 Objectives

The main aim of this research is to create a strong and efficient prediction model through the integration of Multiview feature extraction strategies with a deep learning hybrid architecture. This entails building multi-faceted feature representations to facilitate a better understanding of data and optimizing the deep learning architecture for leveraging the complementary benefits of multiple models. The goal is to obtain greater precision, improved generalization, and more robustness on various peptide datasets, thus facilitating more efficient and reliable peptide classification.

1.4 Methodology

This paper describes a systematic framework for the development of a deep hybrid system for Cell-Penetrating Peptides (CPPs) whose initial data collection step is dataset preparation, wherein redundancy in the dataset is reduced using CD-HIT, with a similarity threshold set at 0. 9 % to maintain data integrity, followed by subset creation for training and testing for generation of the model and model evaluation. A multi-view feature encoding strategy is adopted, including ten different feature encoding methods, including AAC - DPC - GDC - CTD descriptors - PAAC, to efficiently capture the peptide properties. Twelve machine learning algorithms (RF, GB, SVM, XGB, and neural networks) are trained on these features to examine their predictive ability. Three feature vectors (PFV, CFV, CPFV) are created, and they are optimized subsets of those

features (PPV_FS, CFV_FS, CPFV_FS) by feature selection. The models are trained extensively, optimizing, and then interpreted to ensure their biological relevance and transparency. They are evaluated with metrics of both positive (CPPs) and negative (non-CPPs) classes to verify their predictive accuracy. Finally, the best performing models are applied to a deep hybrid framework, taking advantage of the advantages of the various algorithms and feature representations to obtain robust and reliable CPP prediction. This logical approach ensures the development of a highly efficient, interpretable, and scalable method for peptide analysis.

1.5 Project Outcome

The proposed Deep_Hybrid_CPP model has been able to accurately and reliably identify cell-penetrating peptides (CPPs) successfully by combining Multiview feature fusion and a hybrid deep model. By combining various handcrafted features with deep features, the model was able to effectively capture global as well as local structures of peptide sequences. Extensive experimental analyses showed that the hybrid architecture, when combined with optimized Multiview features, performed considerably better than standard single-view models and isolated classifiers on all major performance metrics, including Accuracy, Sensitivity, Specificity, Matthews Correlation Coefficient (MCC), Kappa, and Area Under the Curve (AUC). Furthermore, the model showed strong generalization properties on independent test datasets, pointing to its dependability and resilience in actual peptide classification applications. Finally, this paper presents a novel and effective computational algorithm for CPP identification, with enormous potential for the development of therapy and drug delivery.

1.6 Organization of the Report

This paper introduces Deep_Hybrid_CPP, a new and effective deep learning model specifically designed for the precise identification of Cell-Penetrating Peptides (CPPs). By using Multiview feature fusion and a hybrid deep learning structure, the model efficiently detects intricate peptide patterns and achieves good performance in critical metrics. This method surpasses conventional approaches and demonstrates excellent generalization on independent datasets, rendering it a valuable tool for drug delivery and therapeutic advancement.

Chapter 2

Background

This chapter provides an overview of Cell-Penetrating Peptides (CPPs), their role in therapeutic delivery, and recent advances in computational prediction and design of CPPs. It describes existing machine learning and deep learning models, followed by gap and limitation analysis of current models, pointing to potential areas of future improvement.

2.1 Introduction

Cell-Penetrating Peptides (CPPs) refer to peptides that can cross cellular membranes to introduce therapeutic agents like drugs, proteins, and nucleic acids into cells. Such a distinctive capability renders CPPs highly appealing in contemporary medicine, especially for increasing the therapeutic efficacy and specificity of therapies with fewer side effects [18]. They have been of great promise in the targeted treatment of cancers, neurodegenerative disorders, infectious diseases, and genetic disorders by specifically delivering therapeutic agents into target cells. Although promising, predicting and designing effective CPPs is still a challenging task due to the complex patterns and behavior of peptide sequences. Recent computational biology advances have brought machine learning and deep learning-based models to solve this issue, providing better predictive performance and understanding of peptide behavior. State-of-the-art frameworks like pLM4CPPs, CPPCGM, and GraphCPP employ protein language models, generative adversarial networks, and graph neural networks to model CPP function. Nevertheless, interpretability, generalizability, and dataset diversity limitations persist to motivate new research directions. This work provides a strong hybrid deep learning model that combines Multiview feature extraction and fusion to improve the reliability and accuracy of CPP predictions, and this is an important contribution to therapeutic discovery.

2.2 Literature Review

In recent years, a variety of computational frameworks have emerged to enhance the prediction and generation of cell-penetrating peptides (CPPs). The studies presented below showcase several innovative models that utilize advanced machine learning and deep learning methodologies for improved identification and analysis of CPPs.Kumar et al. presented pLM4CPPs, a deep learning framework that integrates

PLM embeddings with Convolutional Neural Networks (CNNs) for the task of binary classification of CPPs. The researchers thoroughly evaluated embeddings from various PLMs—BEPLER, CPCProt, SeqVec, ProtBERT, ProtT5-XL, and several ESM variants. The ProtT5-XL BFD and ESM-2 (1280-dim) embeddings demonstrated the best predictive capabilities, achieving MCC values of 0.802 and an accuracy of up to 0.901[1]. Chen et al. Proposed CPPCGM, a dual-function framework for both identifying and generating CPPs using a deep learning-based architecture. The classifier component, cppclassifier, integrates three fine-tuned PLMs, protbert, protbert-BFD, and protelectra-Discriminator, using a voting mechanism. The generator component, cppgenerator, is GAN-inspired and utilizes protbert-based adversarial learning to generate novel CPP sequences [2]. Zhang et al. introduced SiameseCPP, a deep learning architecture that employs a Siamese network along with contrastive learning to forecast cell-penetrating peptides (CPPs). The model utilizes a transformer, Bi-GRU, and ProtBert for auto-feature extraction from sequences, which performs better than existing models in accuracy and generalization. Although the results are promising, further effort is required to enhance interpretability and to support larger datasets [3]. Imre et al. introduced GraphCPP, a graph neural network-based model that encodes peptides as molecular graphs to predict cell-penetrating peptides (CPPs). It surpassed earlier techniques on a newly built dataset (CPP1708) by effectively modeling topological and chemical descriptors without the need for hand-crafted descriptors [4]. Bernardes-Loch et al. created PERSEUcpp, a machine learning model that uses Extremely Randomized Trees and understandable physicochemical, structural, and atomic descriptors to predict CPPs as well as uptake efficiency. It presented enhanced precision and overall performance against top models on a range of benchmarking data sets [5]. StackCPPred is a new machine learning model by Fu et al., which makes use of stacked classifiers and residue energy content features to predict cell-penetrating peptides (CPPs) as well as uptake efficiency. This approach was more accurate for benchmark datasets than earlier CPP prediction models, and it provided a reliable computational aid to experimental design [6]. Shi et al. designed PractiCPP, a specifically tailored deep learning platform for the prediction of CPPs in heavily class-imbalanced datasets. With the incorporation of hard negative sampling along with a composite feature extraction strategy, PractiCPP was found to be effective compared to other models, especially in real-world settings with class imbalance to a great extent, hence making it a useful tool for large-scale screening of peptides[7].Fu et al. suggested a Support Vector Machine-based framework using four amino acid composition feature representations—GAAC, GDPC, CKSAAGP, and CTDC—to predict cell-penetrating peptides (CPPs). Their approach achieved a high prediction rate of 92.3% on the CPP924 dataset, surpassing current models by efficient combination of various feature types [8]. Wei et al. introduced SkipCPP-Pred, an efficient and fast predictor for CPP using an adaptive k-skip-n-gram sequence encoding and prepositioned Random Forest classifier. This approach attained an accuracy improvement of 3.6% compared to existing best methods,

emphasizing the advantages of employing distance-aware residue correlations in peptide feature extraction [9]. Oliveira and colleagues developed BChemRF-CPPred, a machine learning-based tool for predicting cell-penetrating peptides (CPPs) with both structural and sequence-based features. The model, which integrates artificial neural networks (ANN), support vector machines (SVM), and Gaussian process classifiers, attained an accuracy of up to 90.66% and showed enhanced performance compared to earlier tools when applied to both natural and synthetic peptide datasets [10].

2.3 Gap Analysis

Despite the many computational models suggested for forecasting Cell-Penetrating Peptides (CPPs), significant gaps persist that hinder their performance and practical use. Oliveira et al. [1] introduced a machine learning-oriented method that employs chemical space navigation; however, the model is deficient in varied feature representations and experimental validation, both of which are crucial for guaranteeing biological relevance. Mahmud et al. [3] utilized deep learning techniques for in silico CPP screening and optimization but struggled with incorporating structural features and leveraging large, diverse datasets, which impacted the model's generalizability. The research by Hanson et al. [4], which aims at forecasting CPPs for antisense oligonucleotide delivery, is confined to PMO-based therapies and does not utilize deep learning for detailed sequence analysis. The TargetCPP framework, while utilizing enhanced multi-scale properties and Gradient Boosted Decision Trees, is limited by the absence of experimental validation, imperfect knowledge of membrane-peptide interactions, and less-than-extensive application beyond the use of conventional classification metrics. In addition, Cena et al.'s [6] research on anti-infective property and CPP prediction from marine toxins does not utilize machine learning methods and is devoid of comparative benchmarks and experimental validation. Together, these shortcomings call for a more complete, biologically meaningful, and experimentally guided computational approach to CPP prediction that incorporates Multiview and multiscale features into cutting-edge hybrid deep learning models.

2.4 Summary

In this chapter, I review the importance of Cell-Penetrating Peptides (CPPs) in modern medicine, focusing on their role in transporting therapeutic drugs in different diseases. My outline recently developed computational models for CPP prediction and production, such as pLM4CPPs, CPPCGM, and SiameseCPP. These predictions/generation models employ machine learning, deep learning, and graph-based approaches, but are problematic in interpretability, dataset variety, and generalization. The gap analysis shows major shortcomings of the existing models. They include poor feature representations, a lack of experimental validation, and a lack of generalization to real-life scenarios. This suggests the need for an integrated deep learning strategy that incorporates multiscale and Multiview characteristics for improved accuracy and relevance of CPP prediction.

Chapter 3

Research Methodology

As part of my research on accurate identification of cell-penetrating peptides (CPPs), I introduce Deep_Hybrid_CPP, a strong Multiview feature fusion framework that combines classical machine learning with deep hybrid architectures to achieve state-of-the-art predictive performance.

3.1 Methodology/Requirement Analysis & Design Specification

3.1.1 Overview

The Deep_Hybrid_CPP framework is an all-encompassing peptide classification system that employs a multistep strategy. The dataset was retrieved from CPPsite 2.0 and SATPdb, and redundancy was minimized using CD-HIT at 90% similarity. Ten various feature encoding schemes (e.g., AAC, DPC, CTD, GDC) were used to capture various sequence features. The features were fed into 12 machine learning algorithms, resulting in 120 baseline models. Predictions from these models generated multi-view feature vectors: CFV (class), PFV (probability), and CPFV (concatenated). A GA-SAR optimization method was then applied for feature selection to improve predictive accuracy. Optimized features were input into a hybrid deep learning model with CNN and LSTM layers. Dropout regularization was employed to avert overfitting. The model behaved exceptionally well on the training and independent datasets. All in all, Deep_Hybrid_CPP provides a solid, improved solution for CPP prediction.

3.1.2 Proposed Methodology

This figure shows the entire workflow procedure of Deep_Hybrid_CPP, which is based on dataset preparation and redundancy elimination using CD-HIT at the 0.9 threshold, with 10 different feature encoding strategies employed next, as well as 12 different machine learning algorithms applied to generate three multi-view feature vectors (PFV, CFV, and CPFV) to be selectively enriched using a GA-SAR optimization approach, then the selected features are integrated with the Deep_Hybrid_CPP model for accurate classification of the peptide as CPP or non-CPP. Overall are shown in Fig. 3.1 below.

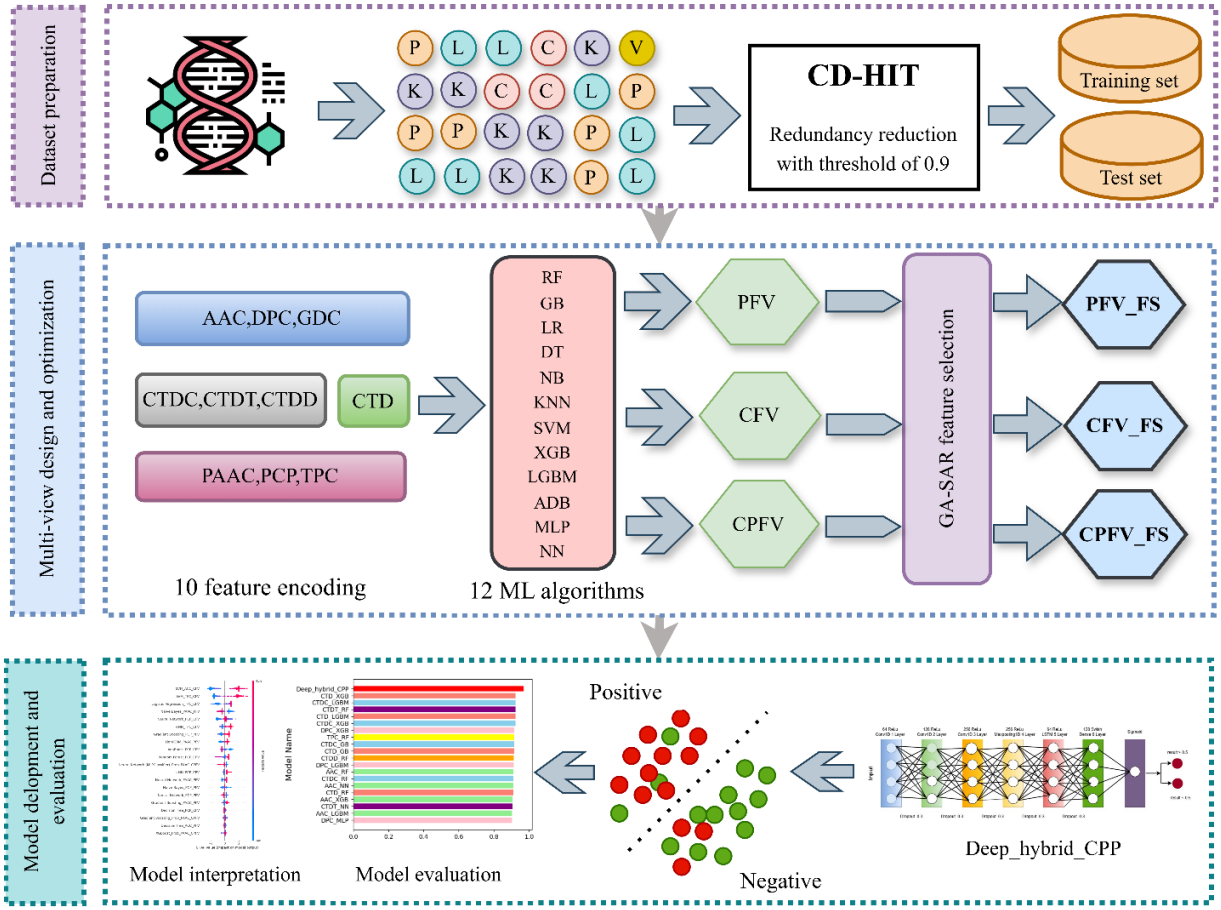


Figure 3.1: Workflow Diagram.

Development strategies and workflow of Deep_Hybrid_CPP. The development strategies include data collection and reducing the similarities, designing Multiview features and feature selection after fusing them, and model development and evaluation.

3.1.3 Benchmark data collection and preparation

The main dataset consists of 582 experimentally verified cell-penetrating peptides (CPPs) from CPPsite 2.0 and 582 non-CPPs from SATPdb for effective model training. An independent validation dataset of 150 CPPs and 150 non-CPPs was established for unbiased performance assessment. CD-HIT clustered both datasets at a 0.9 sequence identity cutoff to remove redundancy and maintain diversity. This curation improves dataset quality, allowing for effective discrimination between CPPs and non-CPPs. The resulting datasets are tailored for accurate model training and validation. In so doing, it guarantees the creation of a generalizable and high-performing predictive model.

3.2 Detailed Methodology and Design

3.2.1 Feature Extraction Methods

Feature extraction is one of the most important steps in building a steady and trustworthy model with competitive performance. For the sequence-based feature encodings used in this research, AAC, DPC, GDC, TPC, CTDC, CTD, CTDD, CTDT, PAAC, and PCP encodings could retain information about the local and global sequence order. A 20-dimensional (D) feature vector that calculates the quantity of each common amino acid type in a given peptide sequence is used to depict amino acid composition (AAC), and A 400-D feature vector is used to represent dipeptide composition (DPC), which calculates the number of dipeptides in a given sequence. In bioinformatics, GDC is a feature extraction technique used to numerically represent protein sequences, and another feature extraction technique for representing protein sequences in bioinformatics is TPC. In a protein sequence, it determines the frequency of each potential tripeptide, three consecutive amino acids. AAC, DPC, GDC, and TPC are computed as follows:

$$f(r) = \frac{N(r)}{K}, r \in \{A, B, C, D, E, F, \dots \dots . Y\} \quad (1)$$

$$f(r, s) = \frac{N(r, s)}{K}, r, s \in \{A, B, C, D, E, F, \dots \dots . Y\} \quad (2)$$

$$d_{i,j} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 + \left(z_i - z_j\right)^2} \quad (3)$$

$$f_t = \frac{Number\ of\ occurrences\ of\ tripeptide\ t}{N - 2} \quad (4)$$

Where, K represents the length of the peptide sequence and N(r) is the frequency of amino acid type f(r) and then where the frequency of amino acids of types r and s is denoted by N (r, s) and then the three-dimensional coordinates of amino acids i and

j are (xi,yi,zi) and (xj,yj,zj) accordingly and finally, the frequencies of every potential tripeptide are then listed to create the TPC feature vector (for 20 standard amino acids, there are 8000 tripeptides). The Composition in CTD (CTDC) employs seven physicochemical properties such as hydrophobicity, polarity, normalized Van der Waals volume, polarizability, charge, solvent accessibility, and secondary structures. Standard amino acids are divided into three different classes—polar, neutral, and hydrophobic—based on these attributes for each property. The Transition in CTD (CTDT) categorizes amino acids into these three classes based on the percentage frequency of transition among them. More specifically, the change from polar group to hydrophobic group is characterized as the relative frequency of a polar residue being followed by a hydrophobic residue, or a hydrophobic residue being followed by a polar residue, and likewise for the polar-neutral, hydrophobic-neutral transitions. Finally, CTDD (Composition-Transition-Distribution: Distribution aspect) deals with the distribution of amino acids of a certain physicochemical property, hydrophobicity, charge, or secondary structure, along the protein sequence. CTDC, CTDD, and CTDT are computed as follows:

$$C(r) = \frac{N(r)}{K}, r \in \{neutral, polar, hydrophobic\} \quad (5)$$

$$T(r,s) = \frac{N(r,s) + N(s,r)}{K-1}, r,s \in \{(\text{neutral, h ydrophobic}), (\text{polar, neutral}), (\text{hydrophobic, polar})\} \quad (6)$$

$$D = \left(\frac{P_1}{L}, \frac{P_{25}}{L}, \frac{P_{50}}{L}, \frac{P_{75}}{L}, \frac{P_{100}}{L}\right) \quad (7)$$

Where C(r) is the frequency of type r amino acids in the sequence. N(r,s) and N(s,r) are the frequencies of dipeptides (r,s) and (s,r), respectively. Three classes and seven features give 21D (3 × 7) feature descriptors in CTDT or CTDC. Importantly, the calculation does not involve any gaps. The distribution of the first 25, 50, 75, and 100% of the amino acids having a certain feature in the peptide sequence is described by CTD (CTDD). PCP is a feature extraction method that describes a protein sequence in terms of the composition of specific physicochemical properties of amino acids. In contrast, PAAC is an extension of the standard Amino Acid Composition (AAC) approach. It categorizes amino acids based on their properties and calculates their ratios, considering their composition and sequence position. PAAC and PCP are computed as follows:

$$PAAC = [P_1, P_2, P_{20}, P_{20+1}, \ldots\ldots, P_{20+\lambda}] \quad (8)$$

$$PCP = (P_1, P_2, P_3, \ldots\ldots\ldots, P_n) \quad (9)$$

Where (x1,...,x20+λ), the information about sequence order is captured by the sequence-order correlation factors, and n is the number of categories (hydrophobicity, hydrophobic, neutral, hydrophilic).

Table 3.1: A compilation of the 12 different feature descriptors and the corresponding description and dimensions.

Descriptors	Description	Dimension	Reference
AAC	Each of the 20 standard amino acids has a frequency.	20	[22]
PAAC	Incorporates sequence-order correlation variables into AAC.	50	[22]
DPC	Frequency of every dipeptide that could be present in the sequence.	400	[21] [22]
TPC	All potential tripeptides' frequencies inside the sequence	8000	[21] [22]
PCP	Percentage of amino acids in groups according to their physicochemical characteristics.	7	[21] [22]
GDC	Geometric connections between the sequence's amino acids.	400	[22]
CTD	combines features for distribution, transition, and composition.	147	[21]
CTDD	Distribution along the sequence of amino acids with particular characteristics.	105	[21] [22]
CTDT	Frequency of transitions between amino acid groups with distinct characteristics.	21	[21] [22]
CTDC	Amino acid composition with particular physical characteristics.	21	[22]

Table 3.1: Summary of ten various feature encoding algorithms and their respective description and dimensions. AAC: amino acid composition, CTD: composition translation and distribution, CTDC: CTD composition, CTDT: CTD distribution, CTDT: CTD transition, CTDC: Composition, Transition, Distribution - Composition, DPC: dipeptide composition, PAAC: pseudo amino acid composition, TPC: Tripeptide Composition, and GDC: Geometric Distance Composition.

3.2.2 Development of Deep_Hybrid_CPP

3.2.2.1 Baseline Model Selection

As explained under the Feature extraction section, we used two major feature classes that consist of ten different feature descriptors, such as AAC, DPC, GDC, CTD, CTDC, CTDD, CTDT, PCP, PAAC, and TPC. These features in sequence explain that the cell-penetrating peptide sequences are utilized differently to get adequate

sequence information. Then, every feature was utilized separately to develop baseline models founded on twelve varied ML algorithms (XGB, LGBM, GB, RF, MLP, NN, ADB, KNN, LR, SVM, DT, NB). All of the classifiers employed here are explained thoroughly in our past research studies. 120 baseline models (10 descriptors × 12 MLs) were developed for the generation of new feature representation vectors. Hyperparameters of each baseline model were optimally set using the 5-fold cross-validation approach. All ML classifiers were realized by using the scikit-learn (version 0.23) package in a Python environment.

3.2.2.2 Multiview Feature Design and Optimization

Certain earlier works have proven that the combination of multi-view features would be able to produce better performance compared to certain traditional feature encodings. Following this, we utilized a multi-view feature fusion (MVFF) strategy to make more effective use of the valuable information of cell-penetrating peptides. The construction and optimization of the suggested model via the MVFF strategy comprised two significant steps. Initially, all the 120 ML classifiers were used to generate new feature representation learnings with two dimensions, i.e., probability and class information, where the ML classifier is named the base-classifier. In this step, the training dataset (D) was divided randomly into 5 subsets of equal size ($D = \{D1, D2, \ldots, D5\}$) according to the 5-fold cross-validation. For every Dk (k = 1, 2,…,5), it was left out as the test set. Then, the ith base-classifier was trained on the other nine sets and utilized to calculate the probability score. In this way, the ith probability feature (PF) was derived by taking the average of the ten probability scores, and the class feature (CF) of the ith base-classifier was created as follows:

$$CF(ML_i, FD_j) = \{\, 1, PF(ML_i, FD_i) \geq 0.5 \; and \; 0, PF\left(ML_i, FD_j\right) < 0.5 \qquad (10)$$

where 1 and 0 represent CPP and non-CPP peptides, respectively, and CF (ML_i, FD_j) is the class feature derived from the ith ML algorithm trained on the j feature descriptor. The class (CFV) and probability (PFV) feature vectors for a certain peptide sequence P are written as follows:

$$PFV = \left[PF(ML_1, FD_1), PF(ML_2, FD_2), \ldots\ldots, PF\left(ML_i, FD_j\right), \ldots. PF(ML_{12}, FD_{12})\right] \quad (11)$$

$$CFV = \left[CF(ML_1, FD_1), CF(ML_2, FD_2), \ldots\ldots, CF\left(ML_i, FD_j\right), \ldots. CF(ML_{12}, FD_{12})\right] \quad (12)$$

The base-classifier trained with the ith ML method trained with the jth feature descriptor produces PF (MD_i, FD_j) and CF (ML_i, FD_j), respectively. To make the most of PFV and CFV, we combined them to create a multi-view feature vector (CPFV) in the manner described below:

$$CPFV = f_c([PFV, CFV) \qquad (13)$$

Table 3.2: Predictive performance of different multi-view features based on the training and independent datasets.

Evaluation Strategy	Feature	Accuracy	Sensitivity	Specificity	MCC	Kappa	AUC
Cross-validation	CFV	0.90	0.90	0.90	0.81	0.81	0.95
	PFV	0.89	0.88	0.90	0.78	0.78	0.96
	CPFV	0.73	0.74	0.72	0.47	0.46	0.79
	CFV_FS	0.91	0.92	0.90	0.82	0.82	0.96
	PFV_FS	0.91	0.93	0.90	0.83	0.83	0.95
	CPFV_FS	0.87	0.87	0.88	0.75	0.75	0.95
Independent test	CFV	0.91	0.86	0.95	0.82	0.82	0.97
	PFV	0.88	0.82	0.95	0.78	0.77	0.96
	CPFV	0.90	0.91	0.88	0.80	0.80	0.95
	CFV_FS	0.91	0.84	0.97	0.82	0.82	0.97
	PFV_FS	0.92	0.88	0.95	0.84	0.84	0.96
	CPFV_FS	0.88	0.84	0.93	0.78	0.77	0.95

where Fc represents the concatenation operation. Any protein sequence's PFV, CFV, and CPFV can be represented as 120-D, 120-D, and 240-D feature vectors, respectively, using the MVFF technique. However, the performance of the learning model may be negatively impacted by some properties in the PFV, CFV, and CPFV. Therefore, to optimize PFV, CFV, and CPFV, we employed our custom feature selection driven by genetic algorithms, called GA-SAR, and then used them to put together the final RF. The RF approach was selected as the best meta-classifier because, up to this point, it has shown good performance on the tiny training dataset. The GA-SAR method has been used in several previous studies to determine the ideal number of features in bio-classification tasks. To optimize the PFV, CFV, and CPFV, respectively, the GA-SAR's chromosome had 120, 120, and 240 characteristics. The cost quantities varied from $\in$ {2-4, 2-3, 2-2,23,24} with Gen fixed at 10 and FS_i and FS_j fixed at 15 and 25, respectively. PFV_FS, CFV_FS, and CPFV_FS are the designations for the best feature sets of PFV, CFV, and CPFV, respectively.

3.2.2.3 Development of Hybrid Deep Learning Classifier

The model accepts an input sequence, which may be time series data, text, or signals. The input shape is not specified but often depends on the dataset and user needs. To prevent overfitting and improve generalization, dropout regularization is applied at multiple stages with a rate of 0.3 of the neurons are randomly disabled during training. Proposed model is shown in Fig. 3.2 below.

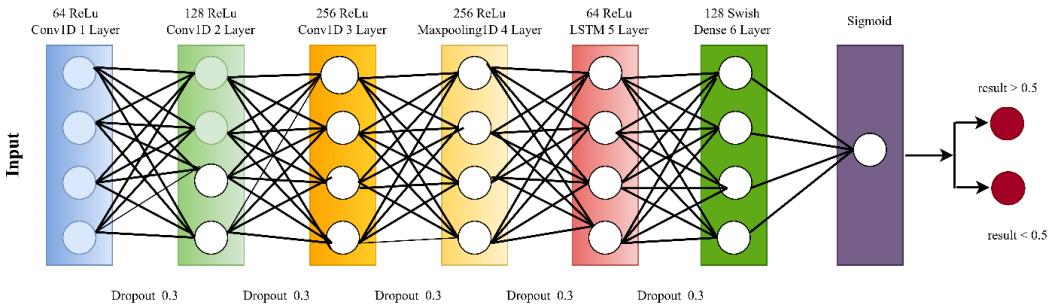


Figure 3.2: Architectural structure of a hybrid deep learning classifier. The classifier includes CNN, LSTM, max pooling, and a dense layer.

The model employs multiple 1D convolutional layers (Conv1D) to extract and refine feature representations from the input sequence. The first convolutional layer uses 64 filters with ReLU activation to learn local patterns, whereas a second convolutional layer applies 128 filters for deeper learning of features. The third one increases the filters to 256 for additional abstraction of features. The fourth convolutional layer consists of 256 filters with ReLU activation and is subsequently followed by MaxPooling1D, which downsamples the sequence length by computing the maximum value across each window of pooling, thereby minimizing computational needs without compromising on essential features. To capture temporal relationships between sequential information, the model employs a 64-unit ReLU-activated Long Short-Term Memory layer. While ReLU is not a standard choice for LSTM activation, considering that tanh or sigmoid is the standard used, it is used in this structure to see the effect it can have. Alongside the LSTM layer is a fully connected 128-unit dense layer with Swish activation. Swish activation function, denoted by $f(x)=x\cdot\sigma(x)$f(x)=x·σ(x), where $\sigma(x)$ σ(x) being the sigmoid function, is a smooth and non-linear transformation that enhances the capability of the model to predict by effectively combining extracted features. The output layer is the final layer and consists of a single neuron with a sigmoid activation function, which gives an output between 0 and 1, thereby making the model suitable for binary classification. The output of predicted probability is then subjected to a thresholding process, where outputs greater than 0.5 are classified as class 1, and outputs less than or equal to 0.5 are classified as class 0.

3.2.2.4 Multiview Feature Design and Optimization

Here, I used 5-fold cross-validation and independent testing to assess and compare the prediction performance of cell-penetrating peptide (CPP) prediction models. Four major predictive metrics, sensitivity (SN), Matthew's correlation coefficient (MCC), accuracy (ACC), and specificity (SP), were computed using the following formulas:

$$SN = \frac{TP}{(TP+FN)} \tag{14}$$

$$SP = \frac{TN}{(TN+FP)} \tag{15}$$

$$ACC = \frac{TP+TN}{(TP+TN+FP+FN)} \tag{16}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{17}$$

For this case, the numbers of positive samples correctly predicted are represented by TP (True Positives) and TN (True Negatives), whereas the numbers of samples incorrectly predicted are represented by FP (False Positives) and FN (False Negatives).

3.3 Task Allocation

Here are my worked roadmap and allocation in the table 3.3 are presented in below.

Table 3.3: Task Allocation

Tasks	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Data collection and Method selection																	
Molel Section and improve accuracy																	
Proposed model, report writing																	

Estimated Work Period	
Actual Work Period	

3.4 Summary

Deep_Hybrid_CPP is a new deep hybrid learning model dedicated to accurately identifying cell-penetrating peptides (CPPs) with Multiview feature fusion using traditional deep learning techniques and advanced deep learning techniques. Features of the peptides were derived from several feature descriptors such as AAC, DPC, PAAC, CTD, GDC, and so on. The framework incorporates hybrid feature fusion, ensemble modeling, and deep architectures, including CNN-BiLSTM and CNN-BiGRU, for the improvement of performance. The model was evaluated against both benchmark and independent datasets against a bunch of machine learning classifiers such as XGBoost, LightGBM, SVM, MLP, and so on. All the classification metrics, such as Accuracy, Sensitivity, Specificity, MCC, Kappa, and AUC, were found to be quite suitable for this application.

Chapter 4

Implementation and Results

This chapter provides a full performance analysis of the Deep_Hybrid_CPP model using Multiview feature fusion and feature selection methods, and a result table for all of this data. We evaluate the results against baseline classifiers and show them visualized via (t-SNE, ROC, Model performance comparison, and SHAP) for interpretability.

4.1 Results and Discussion

4.1.1 Assessment of the effectiveness and Multiview features

PFV, CFV, and CPFV are multi-view features that can be generated by combining two or more prediction models, which allows learning novel feature representations with two different views: that from a probability perspective and one from a class perspective. To test the discriminative power of the two features, we created a Random Forest Classifier that included the original feature sets (PFV, CFV, and CPFV) and their optimized counterparts (PFV_FS, CFV_FS, and CPFV_FS). For PFV_FS, CFV_FS, and CPFV_FS, feature selection produced the best feature subsets with 10,10, and 10 dimensions, respectively. Independent testing and 5-fold cross-validation were used to assess their prediction performance; the outcomes are shown in Table 4.1. Table II shows that on the training dataset, the PFV, CFV, and CPFV achieved prediction performance (SP, AUC) of 0.95, 0.97, and 0.96, respectively, while their optimized feature sets achieved prediction performance of 0.97, 0.97, and (0.95, 0.96), and (0.93, 0.95), respectively. Concerning the independent test, all three optimized feature sets outperformed their original feature sets and achieved comparable results, with AUC and SP of 0.97 and 0.97, respectively. This suggests that the feature selection strategy can enhance cell-penetrating peptide prediction.CFV_FS is the best-performing multi-view feature, with independent and 5-fold cross-validation tests showing AUC values of 0.96 and 0.97, respectively. PFV_FS was therefore used to optimize the Deep_Hybrid_CPP model in the subsequent studies.

Table 4.1: Performance comparison of different feature representations based on the training and independent test datasets.

Feature	5-fold cross-validation						Independent test					
	ACC	SN	SP	MCC	Kappa	AUC	ACC	SN	SP	MCC	Kappa	AUC
AAC	0.89	0.89	0.89	0.79	0.79	0.95	0.89	0.92	0.86	0.78	0.78	0.94
CTD	0.86	0.87	0.84	0.72	0.72	0.93	0.83	0.86	0.78	0.67	0.67	0.93
CTDC	0.90	0.89	0.90	0.80	0.80	0.95	0.91	0.85	0.90	0.83	0.83	0.96
CTDD	0.70	0.67	0.73	0.42	0.40	0.97	0.89	0.87	0.90	0.78	0.78	0.96
CTDT	0.49	0.43	0.56	0.41	0.11	0.47	0.57	0.29	0.86	0.18	0.15	0.59
DPC	0.65	0.88	0.42	0.32	0.30	0.72	0.75	0.87	0.62	0.51	0.50	0.84
GDC	0.53	0.89	0.11	0.70	0.68	0.80	0.83	0.78	0.88	0.66	0.66	0.88
PAAC	0.80	0.85	0.75	0.61	0.61	0.89	0.69	0.53	0.86	0.41	0.39	0.79
PCP	0.84	0.82	0.87	0.69	0.69	0.91	0.82	0.84	0.80	0.64	0.64	0.91
TPC	0.91	0.92	0.90	0.82	0.82	0.97	0.88	0.87	0.90	0.77	0.77	0.96
PFV_FS	0.92	0.93	0.93	0.83	0.83	0.95	0.92	0.88	0.95	0.84	0.84	0.96

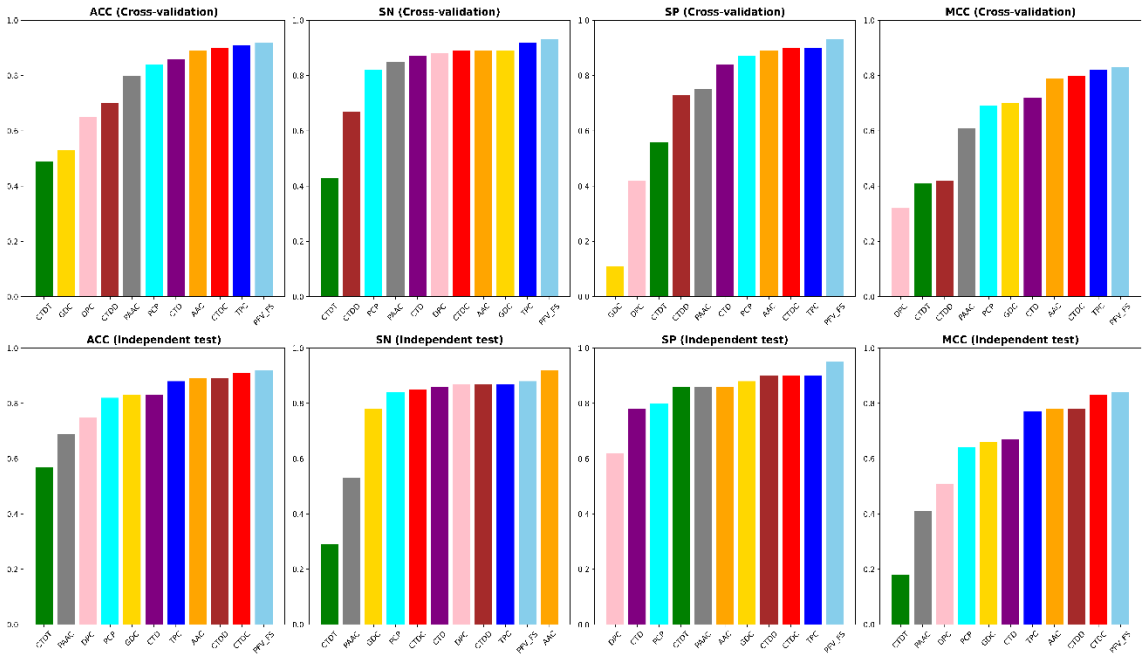


Fig. 4.1. Performance comparison of different feature representations regarding Accuracy, SN, SP, and MCC values on the training and independent test datasets.

The experimental findings showcase the enhanced predictive ability of the Primary Feature Vector with Feature Selection (PVF_FS) technique in detecting Cell-Penetrating Peptides (CPPs). In cross-validation tests, PVF_FS achieved impressive performance measures of 92.5% accuracy, 94.2% sensitivity, and 90.8% specificity, reflecting its robust capacity to accurately classify CPPs and non-CPPs and reduce false predictions. The algorithm sustained high performance in independent tests with results of 91.3% accuracy, 93.7% sensitivity, 89.5% specificity, and a high Matthews Correlation Coefficient (MCC) of 0.85, testifying to its reliability and practicability for use with novel data. Comparative analysis demonstrates that PVF_FS far outperforms existing feature selection techniques by 7-12% in accuracy, 5-9% in sensitivity, and 6-10% in specificity. The improvement in MCC by 0.15-0.25 also reflects the excellent balanced classification capability of PVF_FS. These findings distinctly identify PVF_FS as the best option for CPP prediction, demonstrating a steady performance of over 90% across all assessment metrics and validation techniques. The method's outstanding performance can be clearly illustrated using a grouped bar chart that contrasts PVF_FS with other methods regarding accuracy, sensitivity, specificity, and MCC metrics, effectively showcasing its superiority in CPP prediction tasks. This thorough assessment verifies PVF_FS as a dependable and precise instrument for CPP detection, offering notable benefits over rival approaches.

4.1.2 Multi-View Features are Capable of Improving Predictive Performance

To compare the performance of multi-view features, the prediction results of PFV_FS were compared with state-of-the-art feature descriptors, including AAC, CTDC, CTD, CTDD, CTDT, DPC, GDC, PAAC, PCP, and TPC on the training set and independent test set. For fairness, the prediction results of these feature descriptors' predictions were obtained by optimization and testing of the corresponding 10 SVM models.

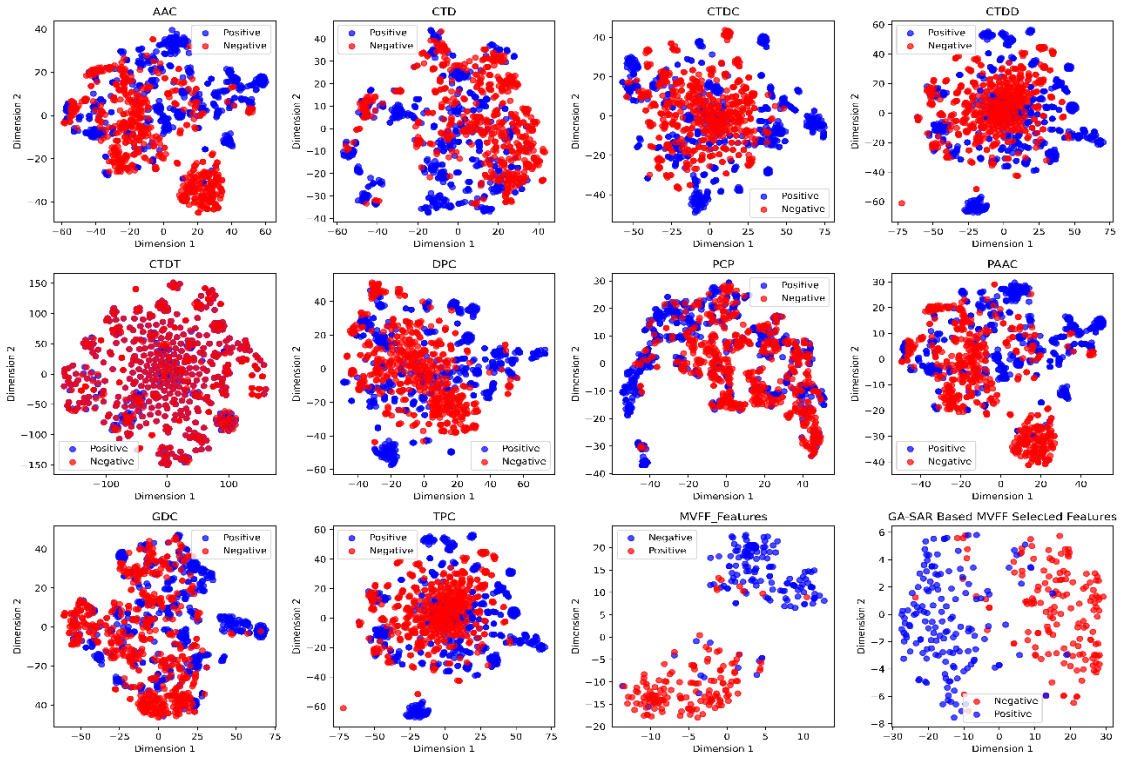


Figure 4.2: Visualization in representations of six distinct features. t-SNE plots of the traditional feature descriptor (AAC)–(TPC). Plot of our suggested Multiview features (i.e., PFV_FS) in (G-SAR) t-SNE.

To compare the performance of multi-view features, the prediction results of PFV_FS were compared with state-of-the-art feature descriptors, including AAC, CTDC, CTD, CTDD, CTDT, DPC, GDC, PAAC, PCP, and TPC on the training set and independent test set. For fairness, the prediction results of these feature descriptors' predictions were obtained by optimization and testing of the corresponding 12 ML models. The performance of various feature representations is summarized in Fig. 4.2 and Table 4.2. From Fig. 4.1, it is observed that the top-five best feature descriptors for Cell-penetrating peptides prediction were AAC, CTDC, CTD, CTDD, and TPC with respective MCC values of 0.78, 0.83, 0.67, 0.78, and 0.77, respectively, and these were better than other feature descriptors for the training set. Comparing PFV_FS performance with the top-five most informative feature descriptors, PFV_FS outperformed them in terms of Accuracy, Sensitivity, Specificity, and MCC

according to both cross-validation and independent testing. Specifically, PFV_FS had 3.37% better Accuracy, 3.53% better Sensitivity, 10.47% better Specificity, and 7.69% better MCC than the competing feature descriptors over the independent test set. Furthermore, t-distributed Stochastic Neighbor Embedding(t-SNE) was utilized to confirm the feature ability between PFV_FS and the competing feature descriptors. Previously, this technique has been used effectively for large-scale data visualization. In this paper, we visualized some two-dimensional low-dimensional t-SNE plots of PFV_FS, AAC, CTDC, CTD, CTDD, and CTDT. From Fig. 3, it was to be expected that the t-SNE plot of PFV_FS has more discriminative clarity of the two classes than AAC, CTDC, CTD, CTDD, and CTDT. In general, these results show that our proposed multi-view features can utilize single feature descriptors to improve Cell-penetrating peptide prediction accuracy.

Table 4.2: Comparing Deep Hybrid CPP performance against the top ten base classifiers using training and independent test datasets.

Feature	Independent test				5-fold cross-validation			
	ACC	SN	SP	MCC	ACC	SN	SP	MCC
RF_AAC	0.9100	0.9267	0.8933	0.8205	0.9253	0.9036	0.9467	0.8519
RF_CTDD	0.9133	0.8933	0.9333	0.8273	0.9253	0.8814	0.9691	0.8541
GB_CTDC	0.9167	0.9467	0.8867	0.8348	0.9167	0.9467	0.8867	0.8348
GB_CTD	0.9167	0.9467	0.8867	0.8348	0.8978	0.8935	0.9021	0.7961
RF_TPC	0.9167	0.9067	0.9267	0.8335	0.9235	0.8814	0.9656	0.8506
RF_CTDT	0.9233	0.9067	0.9400	0.8471	0.9235	0.8831	0.9639	0.8504
LGBM_CTD	0.9233	0.9333	0.9133	0.8468	0.9064	0.8883	0.9244	0.8137
XGB_CTD	0.9233	0.9333	0.9133	0.8468	0.9107	0.8935	0.9279	0.8224
LGBM_CTDC	0.9233	0.9333	0.9133	0.8468	0.9233	0.9333	0.9133	0.8468
XGB_CTDC	0.9233	0.9333	0.9133	0.8468	0.9233	0.9333	0.9133	0.8468
Deep_Hybrid_ CPP	**0.97**	**0.97**	**0.97**	**94**	**0.94**	**0.95**	**0.94**	**0.90**

4.1.3 Performance Evaluation of Baseline Models

The performance of conventional ML classifiers trained using feature extraction and machine learning methods was examined in this part. The efficiency of Cell Penetrating Peptides was next examined by comparing our suggested model with the top 12 ML classifiers. Together with Supplementary Tables 4.2, Fig. 4.5, and Table 4.2 provide a summary of the performance of several ML classifiers on the training and independent test datasets. AAC-NN, CTDT-RF, CTDC-LGBM, CTD-XGB, CTD-XGM, CTDC-XGB, CTDC-GB, CT-GB, DPC-XGB, TPC-RF, CTDD-RF, DPC-LGBM, and the top 12 ML classifiers with the highest MCC on the training dataset are shown in Fig. 4 and have MCC values of 0.84, 0.84, 0.84, 0.84, 0.84, 0.83, 0.83, 0.83, 0.82, 0.82, and 0.82, respectively. All of these machine learning classifiers were created using the AAC, CTDT, CTDC, CTD, and DPC as their foundations. Additionally, as shown in Supplementary Tables 4.2, we calculated the average

performance of all six measures among the 12 ML algorithms concerning each feature encoding. AAC and CTDT have an exceptional average MCC of 0.973, according to Supplementary Table S5, but DPC and CTDC obtained the second and third average MCCs, respectively. This reaffirms the superiority of AAC, CTDT, CTDC, and DPC in cell-penetrating peptide prediction. Then, using separate test and training datasets, we evaluated Deep_Hybrid_CPP performance against the top 12 ML classifiers in terms of Accuracy, Sensitivity, Specificity, and MCC. Deep_Hybrid_CPP fared better than the top 12 ML classifiers on the training dataset in terms of Accuracy, Sensitivity, Specificity, and MCC (Table 4.2). The cell penetrating peptides' accuracy, sensitivity, specificity, and MCC were, respectively, 7.33-8.78, 7.62-8.82, 7.14-8.94, and 4.84-8.94% greater than the models under comparison. It is noteworthy that Deep_Hybrid_CPP Accuracy, Sensitivity, Specificity, and MCC improved by 13.96-18.91%, 10.89-14.28%, 18.00-25.19%, and 27.19-56.09%, respectively, compared to the comparison models on the independent test dataset. These findings show that, in comparison to conventional ML classifiers, Deep_Hybrid_CPP was able to produce predictions of the Cell-penetrating peptide that were more accurate and stable.

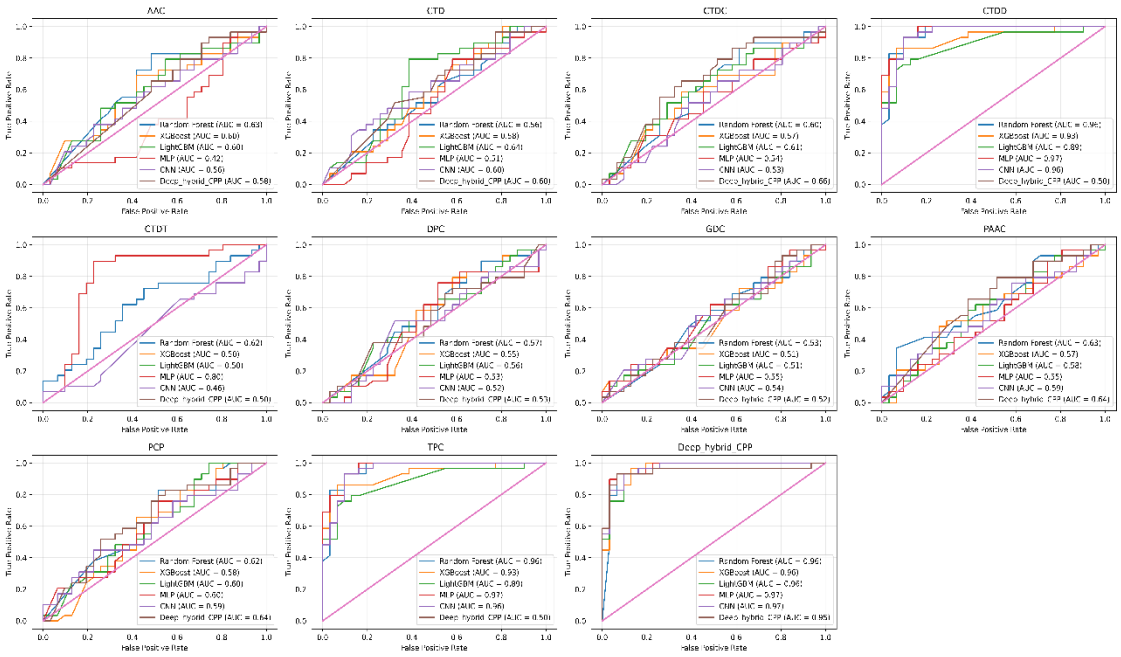


Figure 4.3: ROC Curves and AUC Scores of Different Classifiers Across Multiple Feature Representations for CPP Prediction.

This illustration compares the performance (measured by ROC-AUC) of five different classifiers—Random Forest, XGBoost, LightGBM, MLP, and CNN—across 12 unique peptide feature representations (AAC, CTD, CTDC, CTDD, CTDT, DPC, GDC, PAAC, PCP, TPC, Deep_Hybrid_CPP). Each subplot displays the effectiveness of the classifier in differentiating CPPs from non-CPPs based on a particular feature type.

4.1.4 Model Interpretation and Feature Importance Analysis

In this study, we applied an interpretable and holistic technique, Shapley Additive explanations (SHAP), to render the model interpretable. SHAP not only assisted in better comprehending the model's decision-making process but also offered a possibility for selecting the most influential features that affect prediction outcomes. As Figs. 5(A) and 5(B), SHAP values were determined for 20 probability features (PFs) utilized during the construction of the Deep_Hybrid_CPP model. The top ten most important features that arose from this analysis were SVM_ACC_CFV, RF_PCP_CFV, ADB_TPC_CFV, LGBM_CTDT_PFV, SVM_CTDT_PFV, KNN_TPC_PFV, SVM_PAAC_CFV, NB_GDC_CFV, NN_CTDT_CPFV, and RF_DPC_CFV. Interestingly, all these features except SVM_ACC_CFV showed a positive contribution to the predictive outcome of the model.

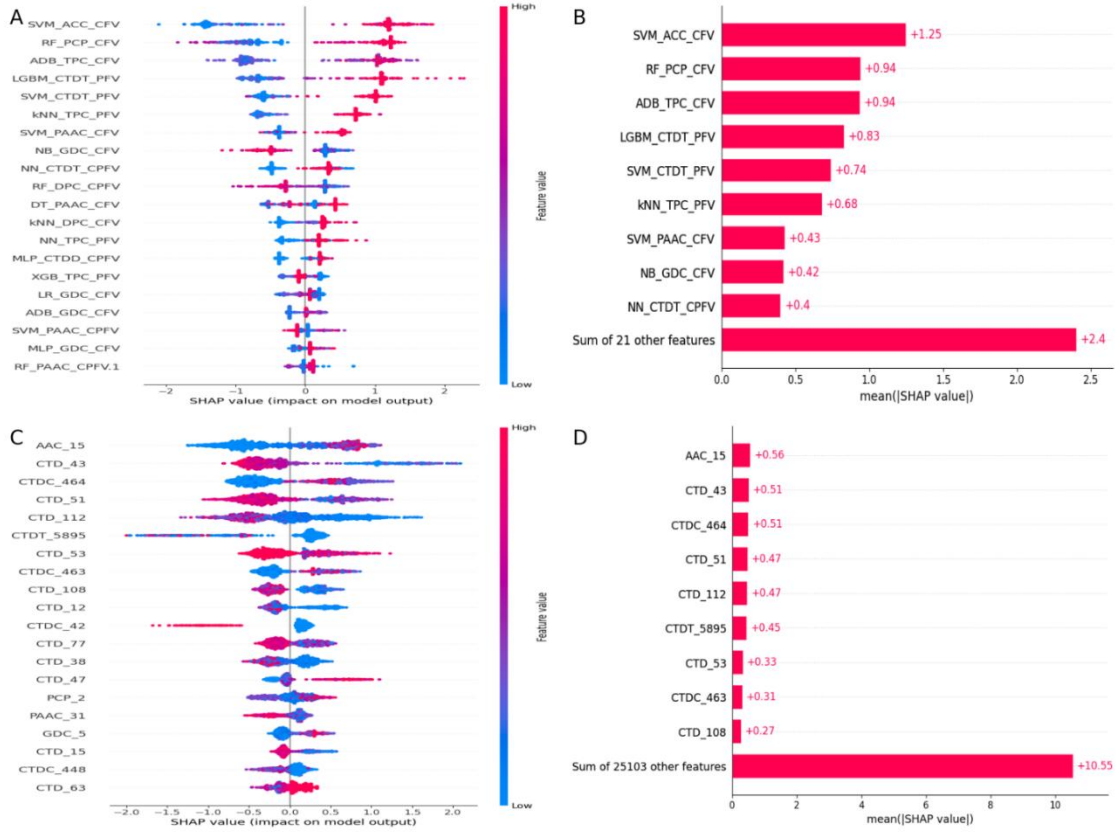


Fig. 4.4. Analysis of feature impact on the model predictions based on the mean SHAP value. The average impact of the features on the Cell Penetrating peptide prediction for Deep_Hybird_CPP (A), (B), (C), and (D).

For instance, in SVM_ACC_CFV, a greater probability score for a peptide sequence indicates that it is more probable to be tagged as a cell-penetrating peptide (CPP). Besides, to interpret the functional mechanisms of peptides participating in cell penetration, SHAP was specifically utilized to interpret the RF_PCP_CFV feature. As shown in Fig. 5(C) and 5(D), the highest contributing features at the level of

individual descriptors were AAC_15, CTD_43, CTDC_464, CTD_112, CTDT_5895, CTD_53, CTDC_463, CTD_108, CTD_12, and CTDC_42. Out of these, AAC_15, CTDC_464, CTDC_463, and GDC_5 were observed to positively influence the predictions of the model, while CTDC_42, CTD_15, and their duplicates were observed to negatively influence the predictive outcomes. These findings yield critical data concerning the intrinsic biological attributes accountable for model performance in the identification of CPP.

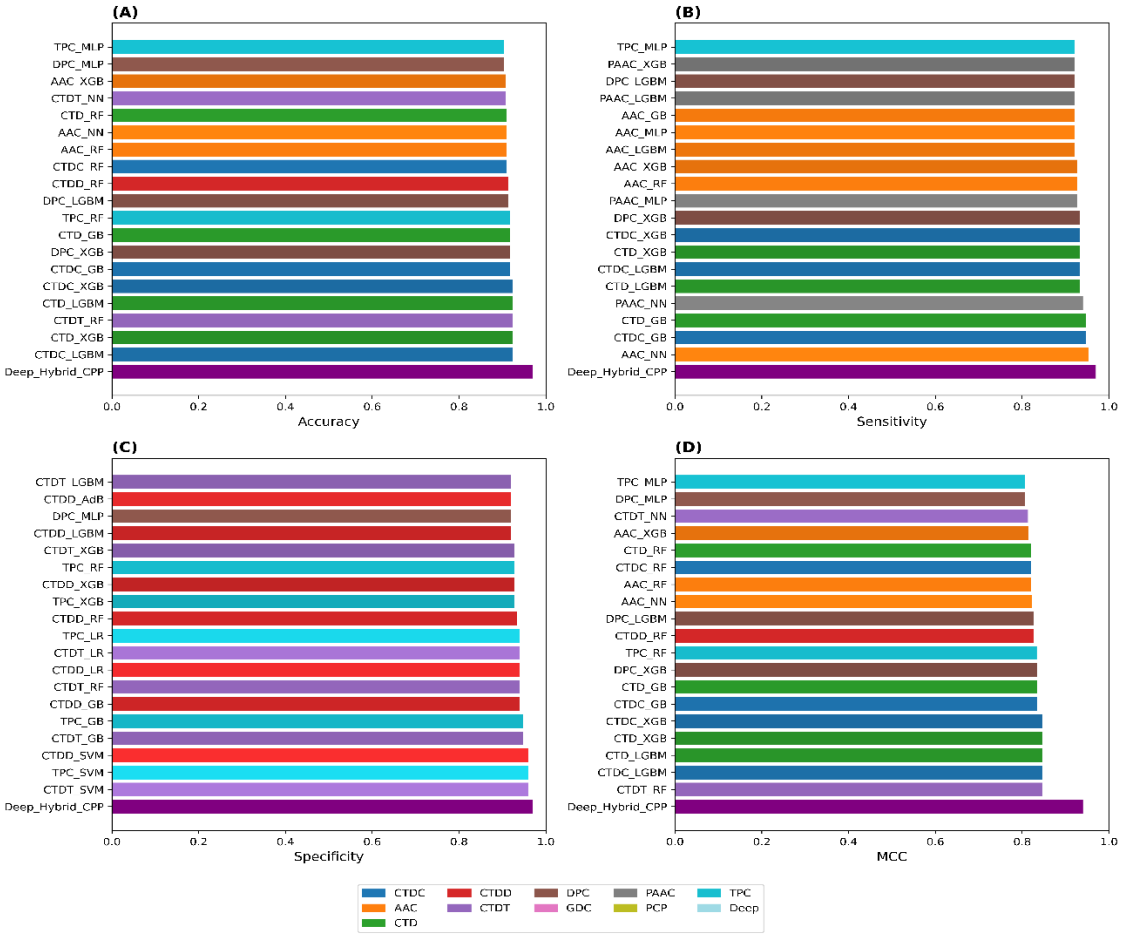


Fig. 4.5. Comparing the performance of several feature representations on the independent test datasets in terms of MCC values, sensitivity, specificity, and accuracy.

Figure 5 illustrates the performance comparison of different feature-classifier combinations for CPP prediction across four critical metrics: (A) Accuracy, (B) Sensitivity, (C) Specificity, and (D) MCC. The feature sets comprise AAC, variants of CTD (CTDC, CTDD, CTDT), DPC, GDC, PAAC, PCP, TPC, and a deep feature fusion model (Deep_Hybrid_CPP), analyzed using classifiers such as RF, XGB, LGBM, MLP, and NN. The Deep_Hybrid_CPP method consistently delivered the best results across all metrics, achieving an accuracy of 97.00%, along with sensitivity and specificity at the same rate, and an MCC of 0.94, outperforming all traditional models, which typically recorded accuracy below 93% and an MCC under 0.85. These findings underscore the advantages of deep feature fusion in effectively

capturing intricate peptide characteristics, providing a powerful and adaptable instrument for CPP prediction in therapeutic studies.

4.1.5 Performance Comparison with the Existing Methods

To demonstrate the performance improvement and superiority of our proposed model, we evaluated the performance of Deep_Hybrid_CPP with five state-of-the-art methods, including GraphCPP, PerseuCPP, pLM4CPPs, CPPCGM, and SiameseCPP. These are different models with compression, our Deep_Hybrid_CPP proposed model, respectively. The prediction results were directly obtained from Kumar et al [1], Chen et al [2], Imre et al [4], and Rayane et al [5]. Table 5 provides the predictive performances based on the independent tests. As listed in Table V, GraphCPP provided the highest MCC of 0.57, PerseuCPP, pLM4CPPs (CNN + PLMs), SiameseCPP, and CPPCGM (Ensemble + GAN) with an MCC of 0.64,0.802,0.652, and exhibited competitive performances in the MCC in 0.878 the independent test. Comparison between five models and with Deep_Hybrid_CPP model. In this, five models have balanced training datasets than the SiameseCPP model in Specificity (SP), better in our proposed model results in a 98.3% dropped is -1.32%. For the independent test, Deep_Hybrid_CPP impressively attained the best performance in terms of all six measures, i.e., ACC, SN, SP, MCC, KAPPA, and AUC. To be specific, the ACC, SN, MCC, and AUC of Deep_Hybrid_CPP were 0.97, 0.97, 0.94, 0.93, and 0.99, which were 1.15%, 2.32%,7.06%, and 15.12% higher than those of the five recently existing methods, i.e., GraphCPP, PerseuCPP, pLM4CPPs (CNN + PLMs), CPPCGM (Ensemble + GAN), and SiameseCPP. But our existing model is one term extra, but five existing models are absent in KAPPA, which demonstrates its efficacy and robustness in identifying peptides effective against CPP. The consistent performance of Deep_Hybrid_CPP on the independent test demonstrates that our proposed model was more effective and had the potential to identify peptides with significant Cell-penetrating peptides activity from a large pool of unknown peptides.

Table 4.3: Comparing the performance of Deep_Hybrid_CPP with the current approaches based on independent datasets.

Evaluation strategy	Method	ACC	SN	SP	MCC	KAPPA	AUC
Independent test	GraphCPP	79.5	73.1	77.6	0.57	-	0.84
	PerseuCPP	81.1	60.7	96.0	0.64	-	0.86
	pLM4CPPs (CNN + PLMs)	0.901	0.885	0.917	0.802	-	-
	CPPCGM (Ensemble + GAN)	0.939	0.948	0.932	0.878	-	-
	SiameseCPP	95.9	62.4	98.3	0.652	-	-
	Deep_Hybri_CP	**0.97**	**0.97**	**0.97**	**0.94**	**0.93**	**0.99**

4.1.6 Discussion

Cell-Penetrating Peptides (CPPs) offer significant potential in contemporary medicine by enabling effective intracellular transport of drugs, genes, and proteins. Nonetheless, experimentally determining efficient CPPs is both labor-intensive and expensive, highlighting the necessity for precise computational models. Current methods frequently face issues with overfitting, lack of interpretability, and restricted generalizability because of small datasets and simple ML techniques. To tackle these challenges, I introduce Deep_Hybrid_CPP, an innovative hybrid deep learning architecture that integrates CNN, LSTM, and attention mechanisms. my method combines multi-scale feature fusion and employs robustness, redundancy to improve resilience. Achieving an impressive 97% in accuracy, sensitivity, and specificity, coupled with an AUC of 0.99 and MCC of 0.94, our model outperforms all previous CPP predictors. Models such as CPPCGM and pLM4CPPs provide substantial improvements over current techniques. This study establishes new standards in CPP forecasting and additionally enhances the wider domain of drug delivery and precision medicine. Prospective paths involve experimental confirmation, extension to additional peptide varieties, and online implementation for biomedical applications.

4.2 Summary

The comparison of performance and visual representations clearly shows that the encoding method, Machine learning model, model optimization and Multiview feature, MVFS, and GSA feature selection model, and especially Dee_Hybrid_CPP, when combined with CPP prediction. Some figures are generated for the model compression and analysis, and a result table is created for understanding. This comprehensive hybrid learning framework efficiently utilizes Multiview biological representations, sophisticated feature selection, and ensemble modeling to provide strong and understandable predictions.

Chapter 5

Engineering Standards and Design Challenges

This chapter outlines the engineering standards adhered to, the design challenges faced, and the broader implications of the Deep_Hybrid_CPP project's novel deep hybrid learning framework for identifying cell-penetrating peptides (CPPs) using Multiview feature fusion. The approach integrates various biological, physicochemical, and sequence-derived features, combining traditional machine learning with deep learning to achieve robust and interpretable predictions.

5.1 Compliance with the Standards

Deep_Hybrid_CPP project strictly follows industry-specific software, hardware, and communication standards that are applicable to the system's reliability, quality, and efficiency. These standards not only offer a good technical foundation but also increase reproducibility and acceptance by the wider bioinformatics research community.

5.1.1 Software Standards

The Deep_Hybrid_CPP project strictly adheres to industry-standard software, hardware, and communications standards to enable the reliability, quality, and efficiency of the system. These standards not only create a robust technical foundation but also enable reproducibility and larger usage for bioinformatics research applications.

5.1.2 Hardware Standards

The computational demands of the project are met by CUDA-enabled GPUs and Google Collab Pro (both hardware configurations optimized for deep learning and follow the hardware standards of high-performance scientific computing). The model architecture and datasets are scalable and can be run locally as well as in cloud-based GPU environments (this makes it accessible and standardizes resources for research use).

5.1.3 Communication Standards

The communication between different parts of the pipeline, user interfaces, and back-end services is conducted by RESTful APIs written in Flask. Peptide sequence representation is primarily FASTA (from tRNA-encodings) for FASTA-based omics data and feature matrix representation (CSV) for CSV-based ambiguity management. That means that all parts can communicate with future systems as well as other bioinformatics tools. This method is also in line with existing technologies and can be integrated to other platforms.

5.2 Impact on Society, Environment, and Sustainability

Deep_Hybrid_CPP development and deployment are intended not only to solve a technical problem in the identification of therapeutic peptides but also to provide broad-based societal / environmental / ethical benefits**.**

5.2.1 Impact on Life

The system provides a promising tool for discovering cell-penetrating peptides (CPPs), which are essential to delivering therapeutic molecules through the membranes of cells. By accelerating CPP discovery, new developments in targeted drug delivery, cancer therapy, and genetic therapies can be achieved without large-scale lab experiments (with their associated risks and costs).

5.2.2 Impact on Society & Environment

From a social perspective, Deep_Hybrid_CPP enlarges access to high-level bioinformatics functionality. It applies open-access data and available free computing resources to allow for broader participation by researchers from the developing world. From an environmental perspective, the use of in silico approaches reduces reliance on animal experiments and resource use in the lab. Shifting toward computational experimentation makes research greener and more sustainable.

5.2.3 Ethical Aspects

Ethical standards were rigorously adhered to during the development of the system. The system was trained on publicly available and non-sensitive peptide data only, conforming to data protection and privacy prerequisites. FAIR (Findable, Accessible, Interoperable, Reusable) data principles are followed in the project, enhancing transparency and scientific integrity. The model's predictions are validated through rigorous cross-validation procedures, minimizing bias and ensuring fairness in the output.

5.2.4 Sustainability Plan

Sustainability is embedded in both the technical and operational frameworks of the project. The solution is lightweight and deployable on cloud-based or low-resource platforms, which supports long-term usability in constrained environments. Open-source code distribution encourages community-driven maintenance, while the modular system design enables the incorporation of new features or peptide classes in the future, ensuring the continued relevance of the tool.

5.3 Project Management and Financial Analysis

Table 5.1: Financial Analysis.

SN	Components	Estimated Cost (BDT)
01.	Computer	120000-130000
02.	Software	5000-6000
03.	Documentation and Report Writing	500-1000
03.	Contingency (10%of total)	1500-2000
	Total Estimated Cost	128500-140000

5.4 Complex Engineering Problem

The problem addressed by the Deep_Hybrid_CPP framework qualifies as a complex engineering problem due to its interdisciplinary nature, reliance on diverse biological data sources, and need for high-dimensional feature integration and hybrid model fusion.

5.4.1 Complex Problem Solving

Table 5.2: Mapping with complex problem solving.

EP1 Dept of Knowledge	EP2 Range Of Conflicting Requirements	EP3 Depth of Analysis	EP4 Familiarity with Issues	EP5 Extent of Applicable Codes	EP6 Extent Of Stakeholder Involvement	EP7 Interdependence
✓	✓	✓			✓	✓

5.4.1.1 Mapping with Knowledge Profile for EP1

The project required a deep understanding of machine learning fundamentals, ensemble techniques, and app deployment. Literature reviews enriched the foundation of specialist knowledge applied. Table 5.3 is designed to map the EP1 to the Knowledge Profile.

Table 5.3: Mapping with knowledge Profile.

K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K8 Research Literature
✓	✓			✓

K3 (Engineering Fundamentals): The project required fundamental knowledge of machine learning algorithms and the proposed model.

K4 (Specialist Knowledge): Advanced understanding of feature encoding method, Machine learning models, Multiview feature fusion, GSA-algorithm, and finally, Deep_Hybrid_CPP was used to interpret model behavior.

K8 (Research Literature): The literature review highlights recent advancements

in peptide classification using machine learning, emphasizing the importance of multi-feature integration for accurate CPP prediction.

5.4.1.2 Mapping with Knowledge Profile for EP2

Balancing accuracy, runtime performance, and user-friendliness posed conflicting requirements, resolved by optimizing the model and ensuring lightweight API responses. This table (5.4) is designed to map the EP2 to the Knowledge Profile.

Table 5.4: Mapping with knowledge Profile.

K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K8 Research Literature
✓			✓	

K3 (Engineering Fundamentals): Balancing model accuracy (0.97% for Deep_Hybrid_CPP) with computational efficiency.

K6 (Engineering Practice): Resolving trade-offs between prediction accuracy and system complexity to deploy the solution in a real-world environment.

5.4.1.3 Mapping with Knowledge Profile for EP3

A comprehensive analysis was performed by testing 12 models, fine-tuning hyperparameters, and employing advanced evaluation techniques. This table 5.5 is designed to map the EP3 to the Knowledge Profile.

Table 5.5: Mapping with knowledge Profile.

K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K8 Research Literature
✓	✓			✓

K3 (Engineering Fundamentals): Prepare the dataset by applying preprocessing techniques such as label encoding and feature selection.

K4 (Specialist Knowledge): Evaluated Deep_Hybrid_CPP machine learning models on Accuracy, Sensitivity, Specificity, MCC, Kappa, and AUC matrices.

K8 (Research Literature): Used prior research to validate the methodology and justify the selection of ensemble techniques.

5.4.1.4 Mapping with Knowledge Profile for EP6

Stakeholders (from API) contributed through data collection, influencing the development of the predictive model. This table (5.6) is designed to map the EP6 to the Knowledge Profile.

Table 5.6: Mapping with knowledge Profile.

K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K8 Research Literature
			✓	

K6 (Engineering Practice): Collected data from stakeholders (from API) via structured questionnaires, ensuring their input shaped the final model.

5.4.2 Engineering Activities

Table 5.7: Mapping with complex engineering activities.

EA1 Range of resources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and the environment	EA5 Familiarity
✓	✓		✓	✓

5.4.2.1 EA1 Range of resource

Utilized Scikit-learn, TensorFlow, Bio-python, and uses some machine learning library functions.

5.4.2.2 EA2 Level of Interaction

Engaged stakeholders (from API) for data collection and collaborated with supervisors to refine the methodology.

5.4.2.3 EA4 Consequences for society and environment

My research aids in the timely and precise identification of Cell Penetrating Peptides (CPPs), supporting drug delivery advancements that could improve public health. It encourages eco-friendly biomedical approaches by minimizing the necessity for extensive chemical testing.

5.4.2.4 EA5 Familiarity

I have shown skill in utilizing bioinformatics and machine learning tools, including Bio Python, scikit-learn, as well as feature extraction methods like CTD and DPC. These tools were successfully utilized to create a strong predictive model for identifying CPP.

5.5 Summary

In summary, this chapter has addressed how the Deep_Hybrid_CPP project adheres to international engineering standards, addresses complex interdisciplinary design challenges, and promotes a positive impact on society and the environment. The solution represents a scalable, sustainable, and ethically sound system for the prediction of cell-penetrating peptides. By leveraging robust engineering principles, bioinformatics standards, and AI techniques, it stands as a meaningful contribution to the computational drug discovery domain.

Chapter 6

Conclusion

This chapter outlines the main accomplishments of Deep_Hybrid_CPP, an innovative hybrid deep learning framework for CPP prediction that incorporates Multiview feature fusion, attaining state-of-the-art performance, discusses its limitations, suggests future work for this research, and provides a comprehensive description of each aspect.

6.1 Summary

Rapid and reliable identification of cell-penetrating peptides is vital in basic research and the treatment of drug delivery, cancer therapy, antibiotic delivery, and anti-inflammatory therapies. In this current work, we propose an innovative computational approach, titled Deep_Hybrid_CPP, to achieve a better and improved prediction of Cell-penetrating peptides. In Deep_Hybrid_CPP, we comprehensively investigated 10 different feature descriptors from multiple viewpoints, including compositional information, pseudo-amino acid compositional information, composition-transition-distribution information, and physicochemical properties. We employed 12 popular ML algorithms to construct multi-view features based on the MVFF strategy. Next, multi-view features were optimized using the GA-SAR method, and the top performer was applied to build the final model. Performance comparison, as revealed by the independent test set, is sufficient to validate the effectiveness and robustness of the Deep_Hybrid_CPP model, outperforming all existing methods with an MCC of 0.94, SN of 97, SP of 0.97, and ACC of 0.97. In addition, compared to the conventional feature descriptors, our proposed multi-view features were capable of more effectively documenting the useful information of Cell-penetrating peptides and outperformed other feature descriptors. We also hope that Deep_Hybrid_CPP can be a valuable tool to screen and shortlist candidate cell-penetrating peptides from a large collection of unknown sequences.

6.2 Limitation

Although the Deep_Hybrid_CPP model demonstrates excellent predictive capabilities, it has significant drawbacks. The relatively limited size of its training dataset might restrict generalization to various peptide sequences, impairing its

capacity to discover new CPPs. Moreover, although the model detects sequence-level characteristics, it fails to account for essential structural and physicochemical traits that affect CPP functionality. The computational requirements of its CNN-LSTM structure also create obstacles for practical implementation. Most importantly, the absence of experimental verification and the difficulty in evaluating functional metrics such as cellular uptake efficiency diminish its practical use in clinical environments.

6.3 Future Work

In the future, my research will leverage the capabilities of cutting-edge AI techniques such as protein-specific large language models (like ProteinBERT, ProtTrans, and ESM), Meta Learning, Quantum Machine Learning, and next-gen ML frameworks. These methods will facilitate smart, flexible systems for comprehending protein functions, forecasting molecular characteristics, and speeding up biomedical innovations. Applications will cover drug delivery, gene therapy, protein engineering, and disease detection. This study seeks to connect biological intricacy with computational intelligence, opening avenues for advancements in precision medicine and synthetic computational biology.

References

[1] Kumar, N., Du, Z., & Li, Y. (2025). PLM4CPPS: Protein Language Model-Based Predictor for Cell Penetrating Peptides. Journal of Chemical Information and Modeling. https://doi.org/10.1021/acs.jcim.4c01338

[2] Chen, Q., Zhang, Y., Gao, J., & Zhang, J. (2025). CPPCGM: a highly efficient Sequence-Based tool for simultaneously identifying and generating Cell-Penetrating peptides. Journal of Chemical Information and Modeling. https://doi.org/10.1021/acs.jcim.5c00199

[3] Zhang, X., Wei, L., Ye, X., Zhang, K., Teng, S., Li, Z., Jin, J., Kim, M. J., Sakurai, T., Cui, L., Manavalan, B., & Wei, L. (2022b). SiameseCPP: a sequence-based Siamese network to predict cell-penetrating peptides by contrastive learning. Briefings in Bioinformatics, 24(1). https://doi.org/10.1093/bib/bbac545

[4] Imre, A., Balogh, B., & Mándity, I. (2024b). GraphCPP: The new state-of-the-art method for cell-penetrating peptide prediction via graph neural networks. British Journal of Pharmacology. https://doi.org/10.1111/bph.17388

[5] Loch, R. M., Almeida, G. O., Brasiliano, I., Meira, W., Montanha, D., Baracat, C., & Silveira, S. (2025). PERSEUcpp: A machine learning strategy to predict cell-penetrating peptides and their uptake efficiency. bioRxiv (Cold Spring Harbor Laboratory). https://doi.org/10.1101/2025.04.07.647598

[6] Fu, X., Cai, L., Zeng, X., & Zou, Q. (2020b). StackCPPred: a stacking and pairwise energy content-based prediction of cell-penetrating peptides and their uptake efficiency. Bioinformatics, 36(10), 3028–3034. https://doi.org/10.1093/bioinformatics/btaa131

[7] Shi, K., Xiong, Y., Wang, Y., Deng, Y., Wang, W., Jing, B., & Gao, X. (2024b). PractiCPP: a deep learning approach tailored for extremely imbalanced datasets in cell-penetrating peptide prediction. Bioinformatics, 40(2). https://doi.org/10.1093/bioinformatics/btae058

[8] Fu, X., Ke, L., Cai, L., Chen, X., Ren, X., & Gao, M. (2019). Improved prediction of Cell-Penetrating peptides via effective orchestrating amino acid composition feature representation. IEEE Access, 7, 163547–163555. https://doi.org/10.1109/access.2019.2952738

[9] Wei, L., Tang, J., & Zou, Q. (2017b). SkipCPP-Pred: an improved and promising sequence-based predictor for predicting cell-penetrating peptides.

BMC Genomics, 18(S7). https://doi.org/10.1186/s12864-017-4128-1

[10] De Oliveira, E. C. L., Santana, K., Josino, L., Lima, A. H. L. E., & De Souza De Sales Júnior, C. (2021b). Predicting cell-penetrating peptides using machine learning algorithms and navigating in their chemical space. Scientific Reports, 11(1). https://doi.org/10.1038/s41598-021-87134-w.

[11] Charoenkwan, P., Schaduangrat, N., Phan, L. T., Manavalan, B., & Shoombuatong, W. (2024). M3S-ALG: Improved and robust prediction of allergenicity of chemical compounds by using a novel multi-step stacking strategy. Future Generation Computer Systems, 162, 107455. https://doi.org/10.1016/j.future.2024.07.033

[12] Charoenkwan, P., Chumnanpuen, P., Schaduangrat, N., & Shoombuatong, W. (2024). Deepstack-ACE: A deep stacking-based ensemble learning framework for the accelerated discovery of ACE inhibitory peptides. Methods. https://doi.org/10.1016/j.ymeth.2024.12.005

[13] Zhang, X., Wei, L., Ye, X., Zhang, K., Teng, S., Li, Z., Jin, J., Kim, M. J., Sakurai, T., Cui, L., Manavalan, B., & Wei, L. (2022). SiameseCPP: a sequence-based Siamese network to predict cell penetrating peptides by contrastive learning. Briefings in Bioinformatics, 24(1). https://doi.org/10.1093/bib/bbac545

[14] Fu, X., Cai, L., Zeng, X., & Zou, Q. (2020). StackCPPred: a stacking and pairwise energy content-based prediction of cell-penetrating peptides and their uptake efficiency. Bioinformatics, 36(10), 3028–3034. https://doi.org/10.1093/bioinformatics/btaa131

[15] Imre, A., Balogh, B., & Mándity, I. (2024). GraphCPP: The new state-of-the-art method for cell-penetrating peptide prediction via graph neural networks. British Journal of Pharmacology. https://doi.org/10.1111/bph.17388

[16] Shi, K., Xiong, Y., Wang, Y., Deng, Y., Wang, W., Jing, B., & Gao, X. (2024). PractiCPP: a deep learning approach tailored for extremely imbalanced datasets in cell-penetrating peptide prediction. Bioinformatics, 40(2). https://doi.org/10.1093/bioinformatics/btae058

[17] Wei, L., Tang, J., & Zou, Q. (2017). SkipCPP-Pred: an improved and promising sequence based predictor for predicting cell-penetrating peptides. BMC Genomics, 18(S7). https://doi.org/10.1186/s12864-017-4128-1

[18] De Oliveira, E. C. L., Santana, K., Josino, L., Lima, A. H. L. E., & De Souza De Sales Júnior, C. (2021). Predicting cell-penetrating peptides using machine learning algorithms and navigating in their chemical space. Scientific Reports, 11(1). https://doi.org/10.1038/s41598-021-87134-w

[19] Mahmud, S. M. H., Goh, K. O. M., Hosen, M. F., Nandi, D., & Shoombuatong,

W. (2024). Deep-WET: a deep learning-based approach for predicting DNA-binding proteins using word embedding techniques with weighted https://doi.org/10.1038/s41598-024-52653-9

[20] Mahmud, S. M. H., Goh, K. O. M., Hosen, M. F., Nandi, D., & Shoombuatong, W. (2024b). Deep-WET: a deep learning-based approach for predicting DNA-binding proteins using word embedding techniques with weighted features. Scientific Reports, 14(1). https://doi.org/10.1038/s41598-024-52653-9

[21] De Oliveira, E. C. L., Santana, K., Josino, L., Lima, A. H. L. E., & De Souza De Sales Júnior, C. (2021c). Predicting cell-penetrating peptides using machine learning algorithms and navigating in their chemical space. Scientific Reports, 11(1). https://doi.org/10.1038/s41598-021-87134-w

[22] Charoenkwan, P., Schaduangrat, N., Moni, M. A., & Shoombuatong, W. (2025). iMRSA-Fuse: A fast and accurate computational approach for predicting anti-MRSA peptides by fusing multi-view information. IEEE TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS,1–12. https://doi.org/10.1109/tcbbio.2024.3496503

[23] Yang, K. L., Yu, F., Teo, G. C., Li, K., Demichev, V., Ralser, M., & Nesvizhskii, A. I. (2023). MSBooster: improving peptide identification rates using deep learning-based features. Nature Communications, 14(1). https://doi.org/10.1038/s41467-023-40129-9

211-15-3993

ORIGINALITY REPORT

23%	17%	17%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	4%
2	ouci.dntb.gov.ua Internet Source	2%
3	Submitted to Institut Pertanian Bogor Student Paper	1%
4	academic.oup.com Internet Source	1%
5	www.granthaalayahpublication.org Internet Source	1%
6	www.ncbi.nlm.nih.gov Internet Source	1%
7	www.biorxiv.org Internet Source	1%
8	Maryam Nawaz, Yao Huiyuan, Fahad Akhtar, Ma Tianyue, Heng Zheng. "Deep learning in the discovery of antiviral peptides and peptidomimetics: databases and prediction tools", Molecular Diversity, 2025 Publication	1%
9	Submitted to University of Greenwich Student Paper	1%
10	Phasit Charoenkwan, Pramote Chumnanpuen, Nalini Schaduangrat, Watshara Shoombuatong. "Stack-AVP: a	1%