Journal of Artificial Intelligence and Data Mining (JAIDM), Vol. 14, No. 2, 2026, 257-273.

Shahrood University of Technology

Journal of Artificial Intelligence and Data Mining (JAIDM)
Journal homepage: http://jad.shahroodut.ac.ir

Research paper

Balancing and Refining Representations for DTI Prediction: A Framework Combining One-SVM-US and a Modified VAE

Ali Ghanbari Sorkhi*, Mohaddeseh Keyhanian and Jamshid Pirgazi

Department of Computer Engineering, University of Science and Technology of Mazandaran, Behshahr, Iran.

Article Info

Article History:
Received 05 October 2025
Revised 19 December 2025
Accepted 05 January 2026

DOI:10.22044/JADM.2026.16927.2825

Keywords:
Drug–target Interactions, Variational Autoencoder, Feature Representations, Data Balancing.

**Corresponding author: ali.ghanbari@mazust.ac.ir (A. Ghanbari).*

Abstract

Accurate prediction of drug–target interactions is essential for advancing drug discovery and repositioning efforts. This study introduces a comprehensive framework that effectively addresses key challenges in DTI prediction, including dataset imbalance and high-dimensional feature representations. The approach integrates multiple protein descriptors—specifically, nine statistical and sequence-based features—and drug molecular fingerprints encoded via Morgan algorithms, with optimal feature combinations selected through validation to capture diverse biological and chemical information. To mitigate dataset imbalance, a one-class SVM-based undersampling method (One-SVM-US) models the distribution of positive interactions to guide the selective reduction of the majority class, thereby effectively balancing positive and negative samples. Furthermore, a supervised, classification-oriented variational autoencoder is employed to compress the high-dimensional features into a lower-dimensional space while preserving class-discriminative information relevant to interaction prediction. The refined features are then classified using machine learning models to predict potential drug–target pairs. Experimental evaluations on benchmark datasets demonstrate the effectiveness of the proposed framework, with results showing perfect AUC-ROC scores of 1.00 on the EN, GPCR, and NR datasets, and a score of 0.9731 on the IC dataset, indicating performance improvements over existing methods. These findings confirm the robustness and potential of the approach as a reliable tool for drug–target interaction prediction.

1. Introduction

In recent years, drug-target interaction (DTI) prediction has emerged as a fundamental challenge within the fields of bioinformatics and drug discovery, serving as a critical step toward elucidating the mechanisms underlying pharmacological efficacy and adverse reactions. This process involves the systematic identification of interactions between chemical compounds—drugs—and biological entities such as proteins or nucleic acids, which are integral to understanding therapeutic effects and facilitating the development of novel therapeutics. An accurate characterization of these interactions can significantly accelerate the identification of promising drug candidates and reduce reliance on costly experimental procedures [1]. DTIs are defined by the binding affinity between a drug and a biological target, often resulting in functional modifications of the target molecule. Proteins, as essential cellular components, perform vital roles including signal transduction, enzymatic catalysis, and regulation of biological pathways. Traditional experimental approaches for assessing DTIs, such as reverse pharmacology and biological assays, have been increasingly replaced by computational methodologies due to their superior efficiency and

reduced resource demands. These computational strategies not only expedite the screening process but also improve predictive accuracy, thereby providing a scalable alternative for large-scale DTI analysis [2].

Despite notable progress in developing DTI prediction models, several persistent challenges hinder their effectiveness. Foremost among these is the issue of data imbalance, characterized by a disproportionate number of negative (non-interacting) samples relative to positive (interacting) instances, which impairs the model's ability to detect rare but biologically significant interactions [3]. Furthermore, the scarcity of high-quality, comprehensive datasets, combined with the inherent complexity and heterogeneity of biological systems, complicates the learning process for machine learning algorithms. Many available datasets suffer from incompleteness and insufficient annotation, constraining model performance and generalizability. Consequently, there is an urgent need for advanced machine learning frameworks capable of resiliently handling data limitations and class imbalance while effectively capturing salient features of DTIs [4].

Recent methodological innovations have aimed to address these challenges through diverse algorithmic strategies. For example, Xie et al. proposed the LRF-DTIs framework, leveraging random forest classifiers combined with Position-Specific Scoring Matrix pseudo-features (PsePSSM) and FP2 molecular fingerprints, supplemented by Synthetic Minority Oversampling Technique (SMOTE) to mitigate imbalance [5]. Similarly, Mahmoud et al. introduced iDTi-CSsmoteB, which employs PseAAC descriptors and Molecular Substructure Fingerprints (MSF), alongside SMOTE and XGBoost classifiers, to enhance predictive robustness [6].

While these approaches have demonstrated improvements, they remain limited in their capacity to fully address the complexity of rare interaction detection and the preservation of critical discriminative information during feature reduction. Their imbalance handling strategies may not sufficiently capture the nuanced patterns of sparse positive samples, and their feature selection or reduction techniques risk losing essential class-specific information in highly heterogeneous datasets.

To overcome these limitations, this study proposes an integrated framework that synergistically combines advanced feature encoding, innovative imbalance mitigation, and supervised dimensionality reduction. Specifically, drugs are encoded using Morgan fingerprints, which effectively capture structural information, while proteins are represented via nine complementary statistical and sequence-based descriptors—including AAC, DPC, GAAC, DDE, PseAAC, PsePSSM, CKSAAGP, GDPC, and GTPC—whose optimal combination is determined through a rigorous internal validation strategy. To address the severe class imbalance inherent in drug–protein pair datasets, a One-SVM-US undersampling technique is introduced, wherein a one-class SVM models the majority class distribution; samples located far from the decision boundary are selectively removed, preserving boundary samples and all minority class instances, thus producing a balanced yet informative dataset. Subsequently, a supervised Variational Autoencoder (VAE) is employed for feature dimensionality reduction; unlike traditional VAEs that focus solely on reconstruction, our model incorporates an auxiliary supervised loss on the latent space, encouraging the separation of different classes and maintaining discriminative information. The resulting reduced features are then classified using multiple algorithms, enabling flexible and accurate prediction of DTIs.

The core innovations of this framework can be summarized as follows: the integration of nine multi-representational protein descriptors with an optimal selection process; the application of One-SVM-US for effective imbalance mitigation; and the deployment of a tailored supervised VAE to generate compact, discriminative feature representations. Collectively, these components establish a unified, robust pipeline that effectively captures complex drug-target relationships and enhances predictive performance relative to existing methodologies.

The structure of the article is organized as follows: In Section 2, related works on the research topic are reviewed, and existing methods for drug–target interaction prediction are analyzed. Section 3 presents the proposed method and explains its different steps in detail. Section 4 provides the analysis and evaluation of the results obtained from applying the proposed method to the experimental datasets. Finally, Section 5 concludes the study.

2- Related Works

Existing methods for drug–target interaction (DTI) prediction can be broadly divided into similarity-based and feature-based approaches, with recent advances leveraging graph-based, deep learning, and hybrid models. Similarity-based methods rely on the assumption that drugs or proteins with similar properties exhibit similar interaction

patterns, typically using chemical similarity for drugs and sequence similarity for proteins [7, 8]. Although these approaches offer interpretability and computational simplicity, they often fail to capture the complex, non-linear relationships inherent in biological interactions, thereby limiting their predictive capacity for novel or unseen drug–target pairs.

Feature-based approaches formulate DTI prediction as a binary classification task, employing a diverse array of machine learning algorithms such as Support Vector Machines (SVM) [9], Random Forests, Rotation Forests, XGBoost, and various deep learning models [10]. These methods typically involve extracting high-level features from drug molecular structures and protein sequences, utilizing techniques including convolutional neural networks (CNNs), multilayer perceptrons (MLPs), autoencoders, and tensor-based embedding strategies [10-13]. Despite their capacity to model intricate feature representations, these approaches are challenged by the high dimensionality of feature spaces, data imbalance, and substantial computational demands, which can impede scalability and generalization.

Graph-based and attention-enhanced models have further propelled DTI prediction by explicitly modeling relationships within heterogeneous biological networks. For instance, DTRE employs heterogeneous graphs coupled with graph neural networks (GNNs) and attention mechanisms to improve predictive accuracy in specific contexts such as endometrial cancer [14]. Similarly, CSCo-DTA integrates molecular-level and network-level features via graph contrastive learning, achieving notable performance improvements; however, these models necessitate high-quality, comprehensive datasets and significant computational resources [15]. Approaches combining Node2vec embeddings with convolutional neural networks (CNNs) and bidirectional attention mechanisms facilitate multimodal feature fusion, demonstrating superior results over traditional methods [11]. The SSLDTI framework employs self-supervised learning paradigms and GNNs to extract features from heterogeneous graphs, yielding enhanced predictive performance, yet it remains computationally intensive and reliant on extensive datasets [16]

The application of pre-trained language models has also gained prominence in DTI prediction. DLM-DTI utilizes ProtBERT to transfer intermediate features from a teacher model to a student model, demonstrating robust performance across datasets such as BIOSNAP, DAVIS, and BindingDB [17]. Similarly, DCGAN-DTA leverages deep convolutional generative adversarial networks (GANs) to extract complex features from protein sequences and drug SMILES representations, achieving superior metrics in CI, MSE, and AUPR, although its effectiveness depends on high-quality training data [12]. The iGRLDTI model incorporates heterogeneous biological interaction networks (HBIN) with node-dependent local smoothing, resulting in improved accuracy metrics such as AUC and AUPR relative to baseline models [18].

Transfer learning methodologies have been employed to mitigate data scarcity issues by pre-training on large-scale source datasets and fine-tuning on smaller, domain-specific datasets, thereby reducing training time and enhancing model performance [19]. Transformer-based models with hierarchical attention mechanisms, exemplified by MHTAN-DTI, effectively capture complex relationships within biological data, outperforming several existing DTI prediction frameworks [20]. Moreover, statistical and experimental database-driven approaches utilize curated datasets to infer potential interactions; however, their applicability is often constrained by incomplete or biased data coverage, limiting their capacity to model the full spectrum of biological complexity [21]

Deep neural network (DNN) models process drug and target features separately before integrating them via convergence functions, resulting in improved predictive accuracy over traditional classifiers [13]. Clustering algorithms provide alternative solutions in low-label scenarios, enabling semi-supervised learning in the absence of extensive annotations [22]. Evolutionary algorithms, including Genetic Algorithms and other metaheuristic optimization techniques, have been applied to identify effective features or parameter configurations for DTI prediction, albeit with increased computational overhead [23]. SVM-based methods continue to serve as reliable tools for feature selection and classification tasks within this domain [9]. Reinforcement learning approaches model the interaction process dynamically, learning policies that simulate drug–target binding behaviors, with promising initial results [24]. Natural language processing (NLP) techniques extract information from scientific literature to predict potential DTIs on a large scale, complementing experimental data-driven methods [25]. Hybrid models integrating deep learning with optimization algorithms have demonstrated further improvements in predictive accuracy [26].

Furthermore, models such as DTI-Voodoo combine molecular and phenotypic drug features with protein–protein interaction networks through graph convolutional networks, establishing state-of-the-art performance metrics on datasets sourced from STRING and STITCH [27]. DeepDTA and DTA-Deep utilize CNN architectures to analyze drug and target features, achieving high scalability and predictive accuracy across extensive datasets. Nonetheless, persistent challenges include data imbalance, the high dimensionality and heterogeneity of feature spaces, and limited applicability to small or domain-specific datasets. To address these issues, the proposed approach integrates One-SVM-US with a modified variational autoencoder (VAE), specifically designed to handle imbalanced datasets by efficiently learning low-dimensional, informative feature representations. This methodology effectively reduces computational complexity, enhances predictive performance on limited datasets, and complements existing state-of-the-art techniques, thereby offering a robust solution for drug–target interaction prediction.

3. Proposed Method

The proposed method is illustrated in Figure 1. As shown, the approach consists of several steps. In the first step, drug and protein data are transformed into feature vectors. For drugs, which are composed of molecular structural units, the Morgan algorithm is employed to generate molecular fingerprints. The output of this step is binary vectors, where each element indicates the presence or absence of specific substructures in the molecular composition of a given drug (assigned a value of 1 if present and 0 otherwise).

For proteins, amino acid sequences are used as the initial input. These sequences, consisting of 20 standard amino acids, may vary in length. In the next step, statistical feature extraction techniques are applied to the protein sequences. In this study, nine distinct feature extraction methods are utilized, and different combinations of these features are then explored to construct the final protein feature vectors. A validation set is used to select the most effective feature combinations.

Subsequently, by selecting the optimal protein feature set, paired feature vectors of proteins and drug fingerprints are constructed. Since drug–protein interactions may exist, all possible protein–drug pairings are generated. One of the major challenges in this context is data imbalance: many of the generated combinations reflect the disproportion between interacting and non-interacting pairs, which reduces classification accuracy for the “interaction” class. To address this, the one-SVM-US method is applied to balance the training dataset.

Another challenge lies in the large number of sample combinations and the high dimensionality of the features. To mitigate this issue, a VAE–based approach is employed for dimensionality reduction and for mapping the features into a new latent space. Finally, both training and testing sets are projected into this latent space, and classification is performed to assign interaction labels to the samples. The detailed description of each stage of the proposed method is provided in the following sections.

3.1. Feature Extraction

Drug fingerprinting is one of the emerging techniques in bioinformatics, designed to identify the unique characteristics of drugs. This method utilizes the chemical and biological profiles of drugs to analyze and predict their interactions with protein targets. A drug fingerprint consists of a set of molecular features that may include chemical structure, atomic composition, molecular bonds, and other physicochemical properties of the compound. These features are represented as numerical vectors or sets of codes and can be employed as inputs for machine learning models.

This approach enables researchers to compare drug similarities and predict their biological behaviors. Drug fingerprinting plays a critical role in drug discovery and in identifying compounds with similar pharmacological activities. Moreover, by applying advanced techniques such as graph neural networks, drug fingerprints can be modeled more accurately, thereby improving the predictive performance of drug–target interaction tasks. As previously mentioned, in this study, nine different feature extraction methods based on amino acid character sequences were employed. Each of these methods is described in the following section.

Amino Acid Composition (AAC):

Amino acid composition [28] is represented as a 20-dimensional vector that calculates the frequency of occurrence of all 20 natural amino acids (i.e., ‘A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y’). The calculation method is presented in Equation (1).

$$f_t = \frac{N(t)}{N}, \qquad t \in \{A, C, D, ..., Y\} \tag{1}$$

where $N(t)$ denotes the number of amino acids of type t, while N represents the length of the protein sequence.

Dipeptide Composition (DPC):

Dipeptide composition [29] provides 400 descriptors for a protein sequence. This

composition is calculated according to Equation (2).

$$D(r,s)=\frac{N_{rs}}{N-1} \tag{2}$$

Where N_{rs} denotes the number of dipeptides formed by amino acids of types r,s and N represents the length of the protein sequence.

Grouped Amino Acid Composition (GAAC):

In GAAC encoding [30], the 20 amino acids are classified into five groups based on their physicochemical properties. The GAAC descriptor represents the frequency of each amino acid group and is calculated according to Equations (3) and (4).

$$f(g)=\frac{N(g)}{N},\quad t\in\{g_1,g_2,g_3,g_4,g_5\} \tag{3}$$

$$N(g_t)=\sum N(t),\quad t\in g \tag{4}$$

Here, $N(g)$ denotes the number of amino acids in group g, $N(t)$ represents the number of amino acids of type t and N indicates the length of the protein sequence.

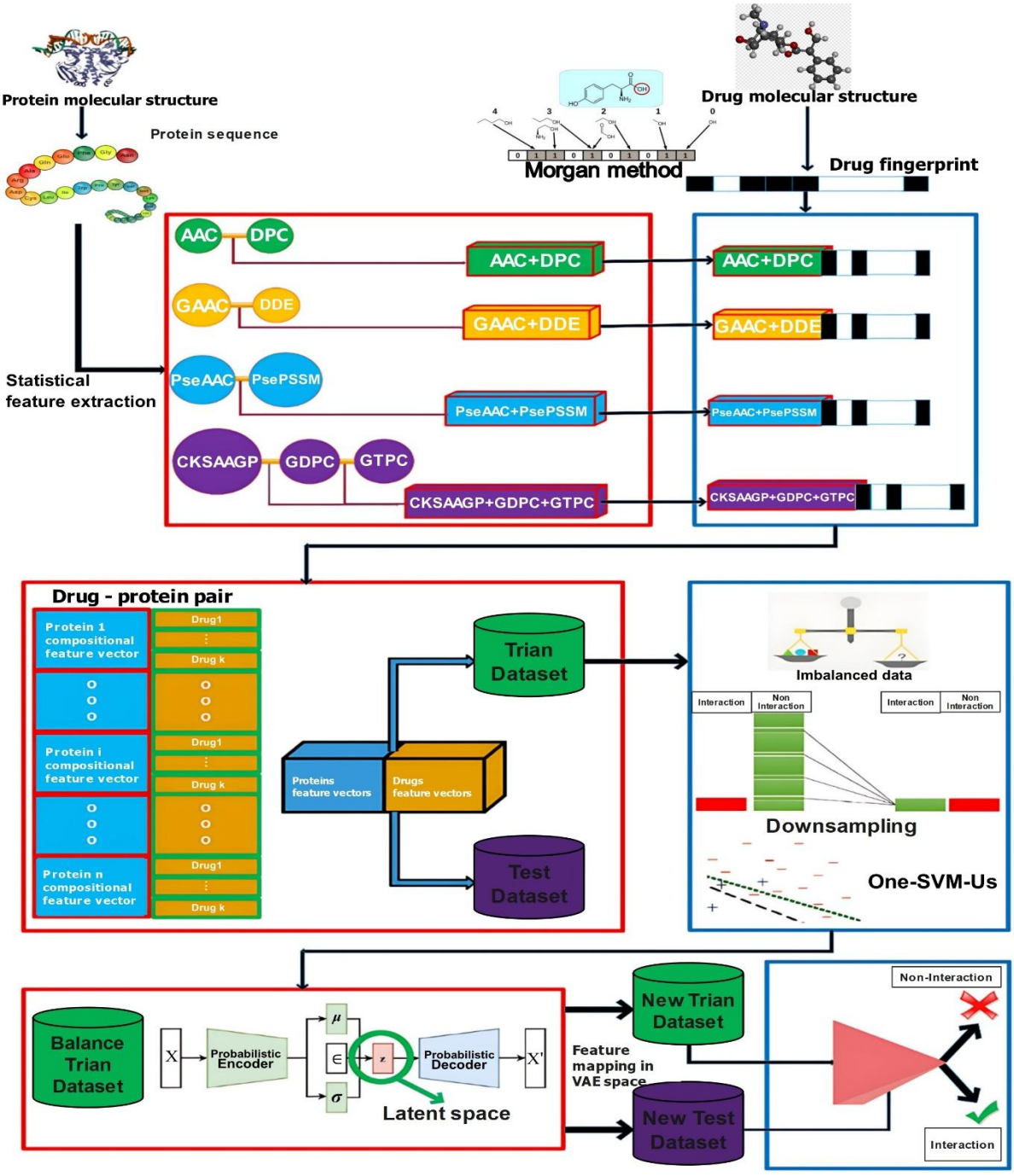

Figure 1. General schematic of the proposed method.

Dipeptide Deviation from Expected Mean (DDE):

Dipeptide Deviation from Expected Mean (DDE) [14] is a feature vector constructed by calculating three parameters: dipeptide composition (Dc), theoretical mean (Tm) and theoretical variance (Tv). These three parameters, along with DDE, are defined as follows. $D(r,s)$, the dipeptide composition measure for the dipeptide 'rs', is expressed in Equation (5).

$$D_c(r,s)=\frac{N_{rs}}{N-1},\qquad r,s\in\{A,C,D,...,Y\} \quad (5)$$

Where N_{rs} denotes the number of dipeptides composed of amino acids r and s and N represents the length of the protein sequence. The theoretical mean $T_m(r,s)$ is given by Equation (6).

$$T_m(r,s)=\frac{C_r}{C_N}\times\frac{C_s}{C_N} \quad (6)$$

where C_r denotes the number of codons encoding the first amino acid, C_s represents the number of codons encoding the second amino acid in the dipeptide rs and C_N is the total number of possible codons. The theoretical variance $T_v(r,s)$ for the dipeptide 'rs' is given by Equation (7).

$$T_v(r,s)=\frac{T_m(r,s)(1-T_m(r,s))}{N-1} \quad (7)$$

Finally, $DDE(r,s)$ is calculated according to Equation (8).

$$DDE(r,s)=\frac{D_c(r,s)-T_m(r,s)}{\sqrt{T_v(r,s)}} \quad (8)$$

Pseudo Amino Acid Composition (PseAAC):

To prevent the complete loss of sequence-order information, Pseudo Amino Acid Composition (PseAAC) was proposed by Chou [31]. The concept of PseAAC has been widely applied in bioinformatics, including proteomics [32], systems biology [33], protein structural class prediction [34], subcellular localization prediction, DNA-binding protein prediction [35], and many other applications.

In comparison to AAC, which consists of 20 components representing the frequency of each of the 20 amino acids in a protein, PseAAC includes a set of more than 20 distinct factors. The first 20 factors correspond to the conventional amino acid composition components, while the additional factors represent correlation factors of varying ranks along the protein sequence. According to the PseAAC concept, each protein sequence is formulated as a PseAAC vector, which is expressed in Equation (9).

$$x=[x_1,x_2,...,x_{19},x_{20},x_{20+1},...,x_{20+\lambda}]^T,(\lambda<L) \quad (9)$$

In this equation, L denotes the length of the protein sequence, and λ is the sequence-related factor, where choosing different values for λ results in PseAAC vectors of varying dimensions. Each component is defined according to Equation (10).

$$J_{i,i+k}=\frac{1}{\Gamma}\sum_{q-1}^{\Gamma}\left[\Phi_q(R_{i+k})-\Phi_q(R_i)\right]^2$$

$$x_u=\begin{cases}\dfrac{f_i}{\sum_{i=1}^{20}f_i+w\sum_{k=1}^{\lambda}\tau_k} & 1\le u\le 20\\ \dfrac{w\tau_{u-20}}{\sum_{i=1}^{20}f_i+w\sum_{k=1}^{\lambda}\tau_k} & 20+1\le u\le 20+\lambda\end{cases} \quad (10)$$

where w is the weighting factor and f_i represents the frequency of the i-th amino acid in the protein sequence. τ_k, the k-th correlation factor, denotes the sequence-order correlation between all adjacent k-th residue pairs and is formulated as shown in Equations (11) and (12).

$$\tau_k=\frac{1}{L-K}\sum_{i=1}^{L-K}J_{i,i+k}\qquad K<L \quad (11)$$

$$J_{i,i+k}=\frac{1}{\Gamma}\left[\Phi_q(R_{i+k})-\Phi_q(R_i)\right]^2 \quad (12)$$

where $\Phi_q(R_i)$ is the q-th amino acid property function for residue R_i and Γ denotes the total number of considered functions. In this study, the protein properties considered include hydrophobicity, hydrophilicity and amino acid side-chain mass; thus, $\Gamma=3$. In this work, λ is set to 1 and w to 0.05. The output feature dimension for each target protein using the PseAAC descriptor is therefore 28.

Position-Specific Scoring Matrix (PsePSSM):

To represent amino acid sequence features in protein sequences, Position-Specific Scoring Matrix (PsePSSM) features, introduced by Shen et al. [36], are employed. PsePSSM encodes evolutionary information and sequence patterns of proteins and has been widely used in bioinformatics research [36-38]. For a target sequence P of length L, the PSSM is used as its descriptor, as originally proposed by Jones et al. [39]. The Position-Specific Scoring Matrix (PSSM), with dimensions L×20, can be defined as shown in Equation (13).

$$P_{PSSM}=\begin{bmatrix} M_{1\to1} & M_{1\to2} & \dots & M_{1\to20} \\ M_{2\to1} & M_{2\to2} & & M_{2\to20} \\ \vdots & \vdots & \vdots & \vdots \\ M_{i\to1} & M_{i\to2} & \dots & M_{i\to20} \\ \vdots & \vdots & \vdots & \vdots \\ M_{L\to1} & M_{L\to2} & & M_{L\to20} \end{bmatrix} \quad (13)$$

Where $M_{i\to j}$ represents the score of the amino acid at the i-th position of the protein sequence being substituted by amino acid type j during the evolutionary process. For simplification, numerical codes 1, 2, …, 20 are used to represent the 20 native amino acids according to the alphabetical order of their single-letter codes. This matrix can be generated using PSI-BLAST searches against the Swiss-Prot database. A positive score indicates that the corresponding amino acid is substituted more frequently than expected, while a negative score indicates the opposite.

In this study, the PSI-BLAST parameters are set as follows: E-value threshold of 0.001, maximum number of iterations for multiple searches set to 3, and all other parameters kept at default values. Each element in the original PSSM is normalized to the range (0,1) and obtained using Equation (14).

$$\overline{M}_{i\to j}=\frac{1}{1+\exp\left(-M_{i\to j}\right)} \quad (14)$$

However, due to the varying lengths of target sequences, constructing a PSSM descriptor as a uniform representation can be advantageous. One possible representation for a protein sample P_i is illustrated in Equations (15) and (16).

$$\overline{P}_{PSSM}=\left[\overline{M}_1,\overline{M}_2,...,\overline{M}_{20}\right] \quad (15)$$

$$\overline{M}_J=\frac{1}{L}\sum_{i=1}^{L}M_{i\to j} \qquad \left(j=1,2,...,20\right) \quad (16)$$

where $\overline{M}_J$ represents the average score of amino acids in protein P that have been substituted by amino acid j during the evolutionary process. If the PSSM of protein P is used directly as in Equation [20], all sequence-order information would be lost. To prevent the complete loss of sequence-order information, the concept of PsePSSM, introduced by Chou [21], is employed to represent protein P, as formulated in Equations (17) and (18).

$$P^{\lambda}_{psePSSM}=\left[\overline{M}_1,...,\overline{M}_{20},G^1_1,...,G^1_{20},G^{\lambda}_1,...,G^{\lambda}_{20}\right]^T \quad (17)$$

$$G^{\lambda}_j=\frac{1}{L-\lambda}\sum_{i=1}^{L-\lambda}\left[\overline{M}_{i\to j}-\overline{M}_{(i+\lambda)\to j}\right]^2 \quad \left(j=1,2,...,20;\quad 0\le\lambda\le L\right) \quad (18)$$

where G^{λ}_j represents the correlation factor of the j-th amino acid and λ denotes the sequential distance along the protein chain. This means that G^1_j is the correlation factor associated with the nearest PSSM score in the protein chain for amino acid type j, G^2_j corresponds to the second nearest PSSM score and so on. Therefore, a protein sequence can be represented using PsePSSM, producing a feature vector of dimension $20+20\times\lambda$. In this study, λ is set to 10. The output feature dimension for each target protein using the PsePSSM descriptor is 220.

k-Spaced Amino Acid Group Pair Composition (CKSAAGP):

The k-spaced amino acid group pair composition (CKSAAGP) [42] defines the frequency of amino acid group pairs separated by k residues (with the default maximum k set to 5). If $k=0$, the amino acid group pairs with zero spacing are represented as shown in Equation (19).

$$\left(\frac{N_{g_1g_1}}{N_{total}},\frac{N_{g_1g_2}}{N_{total}},\frac{N_{g_1g_3}}{N_{total}},...,\frac{N_{g_5g_5}}{N_{total}}\right)_{25} \quad (19)$$

where the value of each descriptor represents the composition of the corresponding residue group pairs in the protein sequence. For a protein of length P and $k=0,1,2,3,4,5$, the values of N_{total} are P_1, P_2, P_3, P_4, P_5 and P_6, respectively.

Grouped Dipeptide Composition (GDPC):

The Grouped Dipeptide Composition (GDPC) [42] is a 25-dimensional vector representing another type of DPC descriptor. It is defined according to Equation (20).

$$f\left(r,s\right)=\frac{N_{rs}}{N-1}, \quad r,s\in\left\{g_1,g_2,g_3,g_4,g_5\right\} \quad (20)$$

where N_{rs} denotes the number of dipeptides composed of amino acids r,s and N represents the length of the protein.

Grouped Tripeptide Composition (GTPC):

The Grouped Tripeptide Composition (GTPC) is another type of TPC descriptor, producing a 125-dimensional vector. It is defined according to Equation (21).

$$f\left(r,s\right)=\frac{N_{rst}}{N-2}, \quad r,s,t\in\left\{g_1,g_2,g_3,g_4,g_5\right\} \quad (21)$$

where N_{rst} denotes the number of tripeptides composed of amino acids r,s and t and also N represents the length of the protein.

3.2. Feature Combination

In this step, the features extracted in the previous section, including both protein and drug features, are combined to construct input samples. For each protein, interactions with different drugs are

examined and aggregated; if an interaction exists, the sample is labeled as 1, otherwise it is labeled as 0. Specifically, for n proteins and k drugs, $n \times k$ samples are generated. Since the number of known DTIs is very limited, the majority of samples represent non-interactions, leading to a severe data imbalance problem. The data balancing method employed to address this issue is described in the following section.

3.3. One-SVM-US Method for Balancing DTI Data

In this application, sequential combinations of proteins and drugs are used. The generated data related to DTIs is extremely sparse. To address the problem of imbalanced data in this study, a new undersampling algorithm called One-SVM-US was developed, which leverages a one-class SVM to handle data imbalance. In the first step, known DTIs are treated as positive samples.

For enzymes, ion channels, GPCRs, nuclear receptors, and the Davis dataset, the numbers of positive samples are 2,926, 1,476, 635, 90, and 2,502, respectively. In the next step, the algorithm considers all possible interactions in the five datasets as negative samples, excluding those known as positive. By applying the One-SVM-US algorithm, a balanced dataset with equal numbers of positive and negative samples is obtained.

The one-class SVM [22] is a global semi-supervised anomaly detector. This algorithm requires a training set consisting of only one class. The One-SVM-US technique, based on the one-class SVM, considers all possible drug-target combinations while excluding positive samples. Instead of using a hyperplane to separate two classes, this algorithm uses a hypersphere to encompass all samples. The RBF kernel was employed for the SVM. The parameter γ was determined using a simple empirical approach, calculated as $\gamma = 1$ number of data points. To compute the anomaly score, the maximum value of the decision function is first obtained as shown in Equation (22).

$$Q = MAX_x decision\_function(x') \quad (22)$$

where x refers to the score vector. The anomaly score is then obtained as defined in Equation (23).

$$outlier\_scores = Q - decision\_function(x) \quad (23)$$

The anomaly scores for the majority class (non-interacting pairs) are sorted in ascending order. The top n samples from this sorted list (i.e., the majority class samples with the lowest anomaly scores, considered most similar to the minority (interacting) class distribution) are selected. Crucially, the parameter n is set to be exactly equal to the number of minority class (interacting) samples in the original training set. It is important to note that this balancing procedure is applied only to the training set to prevent any information leakage and to ensure a fair evaluation on the untouched test set. The final balanced training dataset is then constructed by combining all original minority class samples with these n selected majority class samples.

3.4. Feature Selection and Dimensionality Reduction

Given the high dimensionality of features, employing feature reduction and selection methods becomes essential. Large feature dimensions can lead to model overfitting.

Feature selection is the process of choosing an optimal subset of features (variables) from the available features in the dataset. The goal of this process is to improve predictive model accuracy, reduce model complexity, and enhance generalizability. Feature selection techniques can be broadly categorized into three main types: filter, wrapper, and embedded methods. Filter methods select relevant features based on independent evaluation using statistical measures or machine learning criteria. Wrapper methods involve search algorithms to identify the best subset of features, typically evaluating model accuracy for different feature combinations. Embedded methods select important features simultaneously with the model training process. Proper feature selection can reduce overfitting, enhance model performance, and accelerate training time.

Feature reduction focuses on decreasing the number of input features in the dataset, aiming to simplify the model, decrease computational time, and prevent issues such as overfitting. Feature reduction techniques are mainly divided into feature selection and feature extraction. In feature selection methods, important and relevant features are selectively chosen, whereas in feature extraction, new features are generated based on nonlinear combinations or analysis of existing features.

One of the most well-known techniques in this domain is Principal Component Analysis (PCA), which analyzes the feature correlation matrix to produce a set of new features called principal components that capture the maximum variance in the data. By reducing the dataset's dimensionality, feature reduction helps improve model generalizability and decrease training time, making it particularly useful for large and complex datasets.

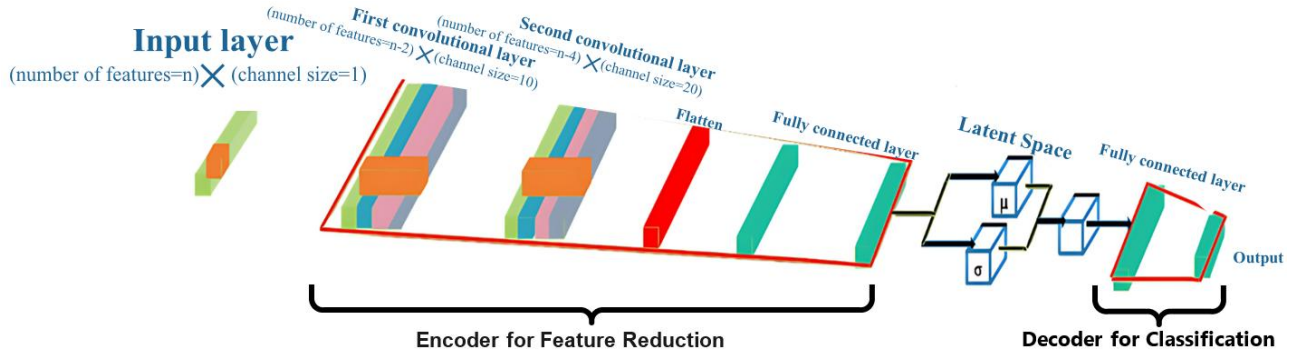

Figure 2. Architecture of the VAE in the Proposed Method.

3.4.1. VAE and Feature Reduction

The VAE is a powerful deep learning technique, particularly useful for feature reduction. A VAE is a generative model capable of mapping input data features into a lower-dimensional latent space. In this process, the VAE consists of two main components: the encoder and the decoder.

The encoder maps inputs to a probabilistic distribution in the latent feature space, which has fewer dimensions than the original input data. This dimensionality reduction effectively compresses and simplifies the data within the feature space. The decoder then reconstructs the original data from these compressed representations. This method efficiently extracts the most important and representative features of the data while filtering out redundant information or noise, ensuring that the latent space accurately reflects the underlying structure of the data.

Using techniques such as the variational loss, the VAE learns meaningful probabilistic distributions and efficient representations of the data. Consequently, it effectively reduces feature dimensionality while simultaneously preserving the quality and accuracy of the reconstructed data.

3.4.2. VAE for Feature Reduction

In this study, since the goal of the VAE is feature reduction based on class labels, modifications were made to the standard VAE architecture. A schematic overview of the VAE architecture used is shown in Figure 2. As illustrated, this architecture consists of two main components: the encoder, which performs feature reduction and constructs the latent space and the decoder. Unlike the conventional VAE, the decoder in this architecture does not aim to reconstruct the input. Instead, it is utilized for label prediction and model training based on the latent space representations.

3.4.3. Proposed VAE for Feature Reduction

The details of the proposed VAE are presented in Table 1. In this approach, after training the dataset, the mean parameters corresponding to the dimensions of the latent layer neurons are used for feature selection. Essentially, for each original feature vector, a new mapping to 100 features is performed for both the training and testing sets.

Table 1. Proposed Architecture for the VAE.

Layer	Layer type	Layer dimensions	Activation function
1	Input Vector	Number of features	-
2	1D Convolution	(Number of features-2)×100	RELU
3	1D Convolution	(Number of features-2)×100	RELU
4	Flatten	-	-
5	Fully Connected (Linear)	(Number of features-4)×20	RELU
6	Fully Connected (Linear)	100	RELU
7	Fully Connected (Linear)	100	RELU
8	Linear	2	SIGMOID

3.5. Classification

Classification is a machine learning technique aimed at assigning input samples to one of several predefined groups or classes. In this method, the model is trained on labeled training data, consisting of samples and their corresponding class labels, so that it can predict the appropriate class for new, unseen data. In this study, various classification methods were employed, and the results indicate that the proposed approach demonstrated strong performance across all classifiers used.

4. Results Analysis

In this section, the results of the proposed method are analyzed. Initially, the dataset, evaluation metrics, and well-known databases in the DTI domain are introduced. Subsequently, the results of the various stages of the proposed method are described in detail. In this context, the impact of different protein feature extraction methods, the effect of the data balancing technique, and the extraction of new features using the VAE are thoroughly analyzed. Finally, the results obtained from the proposed method are compared and analyzed against the most recent and state-of-the-art methods presented in recent years.

It is worth noting that, in this study, the Python programming language was used along with hardware with the following specifications: CPU: Intel(R) Xeon(R) CPU E5-2670 v2 @ 2.50GHz, 2.49 GHz; RAM: 48 GB; GPU: NVIDIA GeForce RTX 2060 Super – 32 GB. For deep learning, the widely used PyTorch framework was employed. All implementations and comparisons with other studies were conducted using the same dataset.

Table 2 shows all the hyperparameters used in all experiments. It should be noted that these values were used consistently across all experiments. For parameters not explicitly specified, the default values of the Scikit-learn library were used.

Table 2. Hyperparameters used in this study.

Model Name	hyperparameters Name	hyperparameters Value
VAE	Loss function	Binary-Cross-Entropy
	Latent Space	100
SVC	Kernel	Linear
	C	0.025
	random_state	42
	Probability	True
KNeighborsClassifier	K	3
DecisionTreeClassifier	max_depth	5
RandomForestClassifier	max_depth	5
	n_estimators	10
MLPClassifier	Alpha	1
	max_iter	1000
AdaBoostClassifier	Algorithm	SAMME

4.1. Evaluation Metrics

In this study, several evaluation metrics were used to assess the performance of the proposed method. These metrics include accuracy (Acc), precision (Pre), recall, F1-score (F1), and the receiver operating characteristic (ROC) curve. In these definitions:

- TP (True Positive) refers to the number of samples correctly identified as having an interaction between a drug–protein pair.
- TN (True Negative) refers to the number of samples correctly identified as having no interaction.
- FN (False Negative) refers to drug–protein pairs that actually interact but are incorrectly predicted as non-interacting by the model.
- FP (False Positive) refers to drug–protein pairs that do not interact but are incorrectly predicted as interacting.

The accuracy metric represents the proportion of correct predictions out of the total number of samples. This metric indicates the percentage of predictions, both positive and negative, that were correct. Here, accuracy is calculated using the counts of TP and TN. Accuracy (Acc) is calculated using the following formula (Equation 24):

$$ACC = \frac{TP + TN}{TP + FP + TN + FN} \tag{24}$$

The Precision metric represents the proportion of correct positive predictions out of all positive predictions. In other words, this metric indicates the percentage of samples predicted as positive by the model that are actually positive. Equation (25) was used to calculate Precision.

$$\Pr ecision = \frac{TP}{TP + FP} \tag{25}$$

The Recall metric indicates how many of the actual drug–target interactions are correctly identified by the model. It represents the proportion of true positive samples that are correctly detected out of all actual positive samples, helping to assess the model's effectiveness in identifying real interactions. Equation (26) was used to calculate Recall.

$$\mathrm{Re}\, call = \frac{TP}{TP + FN} \tag{26}$$

The F1 Score metric acts as a combined measure of Precision and Recall. It evaluates the balance between these two metrics in the model's performance. This measure is particularly useful when the data is imbalanced, as it considers both Precision and Recall simultaneously. Equation (27) was used to calculate the F1 Score.

$$F1\_score = \frac{2 \times \mathrm{Re}\, call \times \Pr ecision}{\mathrm{Re}\, call + \Pr ecision} \tag{27}$$

The Receiver Operating Characteristic (ROC) curve is a graphical representation of a classifier's performance across different threshold settings. It plots the True Positive Rate (Recall) against the False Positive Rate (FPR), showing the trade-off between sensitivity and specificity. The area under the ROC curve (AUC) provides a single measure of overall model performance, with higher values indicating better discrimination between positive and negative classes. Equation (28) was used to calculate the False Positive Rate (FPR).

$$FPR = \frac{FP}{FP + TN} \tag{28}$$

4.2. Datasets

In this study, the gold standard dataset provided by Yamanishi et al. [2] was used. This dataset contains drug–target interaction information collected from various sources, including BRITE, KEGG, BRENDA, Target-Super, and DrugBank. The dataset is divided into four main groups: enzymes (EN), ion channels (IC), G protein-coupled receptors (GPCRs), and nuclear receptors (NRs).

In addition, the Davis dataset was also used. This dataset includes drug–target interactions specifically focused on enzymes and G protein-coupled receptors. The data from the Davis dataset

were used for more detailed analysis and validation of the results obtained from the gold standard dataset. Combining these datasets allows for a more comprehensive evaluation of the performance of drug–target interaction prediction models and improves the accuracy of predictions.
Table 3 compares the number of drugs, proteins and known and unknown interactions for each dataset. As shown in this table, the number of non-interacting samples is much higher than the interacting samples, indicating that the dataset suffers from class imbalance. It is worth noting that in this study, 80% of samples from each class were used for training, 10% for validation and 10% for testing.

Table 3. Information of the datasets used.

Datasets	Number of drugs	Number of proteins	Intractions	Non-Intractions
EN	445	664	2926	292554
IC	210	204	1476	41364
GPCR	223	95	635	20550
NR	54	26	90	1210
Davis	68	442	2502 (Affinity≥7)	27554

4.3. Evaluation of Feature Extraction and Data Balancing Methods

As mentioned in the proposed method, nine protein feature extraction methods and one drug-based Morgan fingerprint method were applied to the entire dataset. The number of features extracted from proteins for each method is as follows: AAC produced 20 features, DPC 400 features, GAAC 5 features, DDE 400 features, PseAAC 28 features, PsePSSM 220 features, CKSAAGP 150 features, GDPC 25 features, and GTPC 125 features. The length of the drug fingerprint vector was constant across all datasets and equal to 256.
At this stage, to evaluate the performance of the extracted features, a baseline SVM classifier was used. The results obtained using this method on the entire dataset with individual features are presented in Table 4. It should be noted that one of the most suitable metrics for evaluating imbalanced data is the F1 score, which has been used here to assess performance. As shown in the results in Table 4, due to the imbalanced nature of the data, the performance of the baseline SVM classifier is not satisfactory.

As mentioned in the proposed method, the One-SVM-US approach was used for data balancing. After balancing, the number of samples in each class was set in a 1:2 ratio; that is, the number of non-interacting class samples is twice that of the interacting class samples. The results obtained from data balancing using the baseline SVM classifier are presented in Table 5. As shown in the table, balancing the data improved performance compared to the previous state in Table 4.
Next, to investigate the impact of feature combination, different features were combined. The results of these feature combinations are presented in Table 6 based on the baseline SVM classifier. According to the obtained results, the AAC+DPC combination achieved the best performance; therefore, this feature combination was used in the proposed method.

4.4. Results of the VAE for Feature Reduction

A key step in the proposed method is the use of a VAE to map input features into a latent space. For this purpose, the proposed architecture was employed. Figure 3 illustrates the comparison of the VAE loss across the five datasets. As shown in the figure, the loss decreases with successive iterations of the algorithm.
The results indicate that the model parameters are well-trained and that the algorithm is moving toward optimal performance. Furthermore, it is evident that the proposed model is independent of the dataset, as the loss decreases consistently across all datasets.

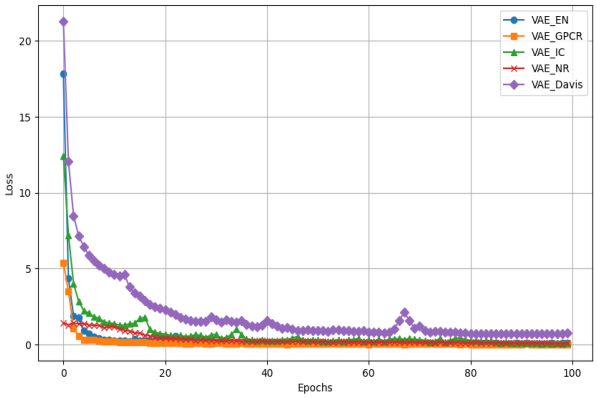

Figure 3. Loss during VAE training phases for the five datasets based on AAC+DPC features.

Table 4. Results of each individual feature on five datasets based on F1 metric without data balancing

Datasets \ Features	AAC	DPC	GACC	DDE	PseAAC	PsePSSM	CKSAAGP	GDPC	GTPC
EN	0.686	0.760	0.566	0.63	0.731	0.64	0.633	0.387	0.568
GPCR	0.594	0.673	0.465	0.447	0.457	0.56	0.489	0.522	0.568
IC	0.537	0.478	0.378	0.51	0.499	0.466	0.551	0.499	0.489
NR	0.445	0.566	0.504	0.455	0.561	0.489	0.591	0.478	0.501
Davis	0.48	0.472	0.466	0.388	0.388	0.415	0.382	0.387	0.411

Table 5. Results of each individual feature on five datasets based on the F1 metric after data balancing.

Datasets \ Features	AAC	DPC	GACC	DDE	PseAAC	PsePSSM	CKSAAGP	GDPC	GTPC
EN	0.842	0.862	0.810	0.801	0.798	0.779	0.821	0.733	0.814
GPCR	0.783	0.855	0.680	0.781	0.688	0.791	0.799	0.810	0.821
IC	0.799	0.765	0.645	0.681	0.771	0.819	0.831	0.873	0.875
NR	0.851	0.845	0.787	0.692	0.791	0.564	0.704	0.710	0.674
Davis	0.789	0.81	0.770	0.71	0.561	0.590	0.661	0.658	0.788

Table 6. Results of feature combinations on the five datasets based on the F1 metric with data balancing.

Datasets \ Features	AAC+DPC	GACC+ DDE	PseAAC+ PsePSSM	CKSAAGP+ GDPC+ GTPC
EN	0.922	0.854	0.891	0.901
GPCR	0.911	0.802	0.812	0.845
IC	0.907	0.722	0.852	0.894
NR	0.896	0.744	0.673	0.754
Davis	0.884	0.681	0.851	0.733

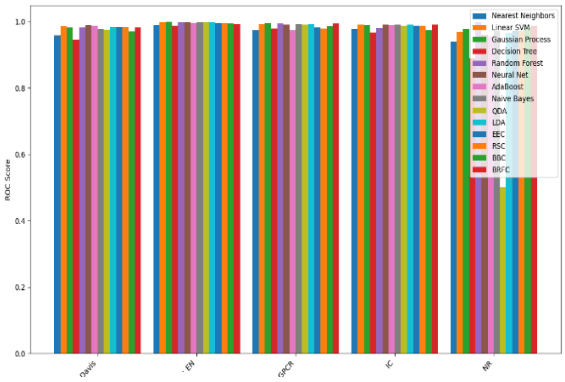

Figure 4. Performance of the Proposed Method Based on AUC-ROC Using Different Classifiers.

4.5. DTI Classification Based on the Proposed Method

In this section, to evaluate the proposed method, various classifiers were employed for comparison. Figure 4 illustrates the performance comparison of the proposed method using different classifiers on the test set based on the AUC-ROC metric.

Furthermore, since the data are imbalanced across the two classes, Figures 5 and 6 show the ROC curves of the proposed method based on four different feature extraction combinations on the EN and NR datasets. As can be seen, the AUC-ROC values of the proposed model are close to 1, indicating strong performance in distinguishing positive and negative samples. Therefore, it can be concluded that the proposed method is effective and reliable in predicting drug–protein interactions. This analysis demonstrates that the model can accurately identify positive samples while minimizing false positives. Another significant advantage is that this method is dataset-independent. Unlike other approaches, which are typically tested on only one or two datasets, the proposed method has been evaluated on multiple datasets with varying sizes and characteristics, showing consistently strong performance across them.

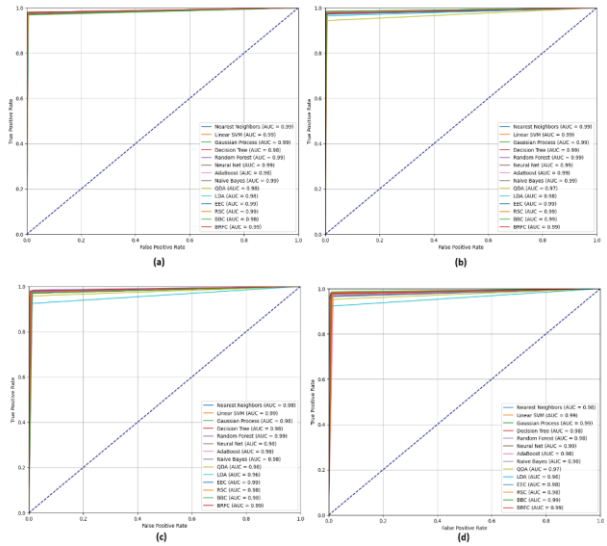

Figure 5. Results of the Proposed Method Based on Four Different Feature Extraction Combinations on the Enzymes (EN) Dataset: (a) GACC + DDE, (b) AAC + DPC, (c) CKSAAGP + GDPC + GTPC, (d) PseAAC + PsePSSM.

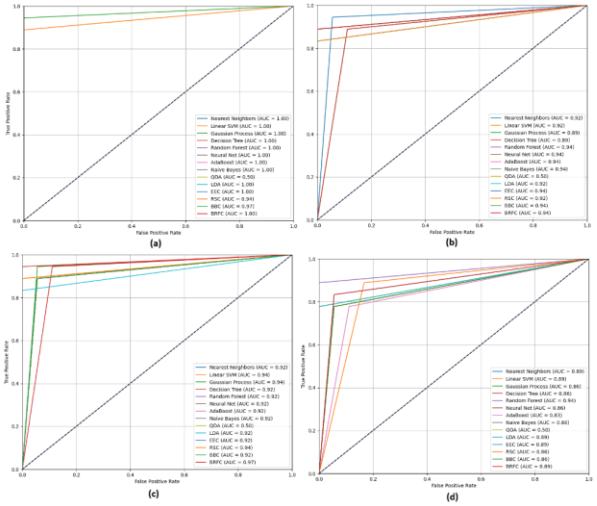

Figure 6. Results of the Proposed Method Based on Four Different Feature Extraction Combinations on the Nuclear Receptors (NR) Dataset: (a) GACC + DDE, (b) AAC + DPC, (c) CKSAAGP + GDPC + GTPC, (d) PseAAC + PsePSSM.

In addition, to provide a better analysis of the proposed method and evaluate its performance from different perspectives, the evaluation results based on three metrics—Recall, F1-Score and Accuracy—are presented for the different datasets EN, GPCR, IC and NR in Figure 7. According to the results, it is evident that the extracted and selected features have improved the performance of various classifiers. Moreover, combining features has further enhanced the performance of the proposed model across different classifiers. The chart analysis across the EN, GPCR, and Davis datasets demonstrates a robust and consistent performance trajectory for Recall, F1-Score, and Accuracy metrics among various classifiers. In the EN dataset, the lines for these metrics remain tightly clustered above 0.97, with Gaussian Process and Random Forest exhibiting peaks near 0.99 in F1-Score and Accuracy, indicating minimal variance and high reliability in classification tasks. Similarly, GPCR shows parallel trends where QDA and BRFC classifiers achieve superior F1-Scores around 0.98, while Recall fluctuates slightly below 0.96 for some models like Random Forest, suggesting a balanced trade-off between precision and sensitivity. The Davis dataset mirrors this stability, with Neural Net leading at 0.956 in Accuracy and 0.954 in F1-Score, underscoring the efficacy of ensemble methods in handling diverse data distributions.

Overall, these patterns highlight the datasets suitability for applications requiring high predictive consistency, as the overlapping lines reflect low sensitivity to classifier choice. In contrast, the IC and NR datasets reveal more pronounced divergences in the multi-line charts, For IC, the Recall line dips notably for LDA at 0.85, creating a wider spread compared to F1-Score and Accuracy, which hover around 0.96–0.97 for top performers like RSC; this variability implies potential overfitting or underrepresentation in certain classes, necessitating feature engineering or hyperparameter tuning. The NR dataset exhibits the most erratic behavior, with drastic drops in Recall for Nearest Neighbors and Linear SVM at 0.833, while F1-Score and Accuracy lines stabilize around 0.94 for models like Random Forest and BRFC, likely due to smaller sample sizes or noisy labels.

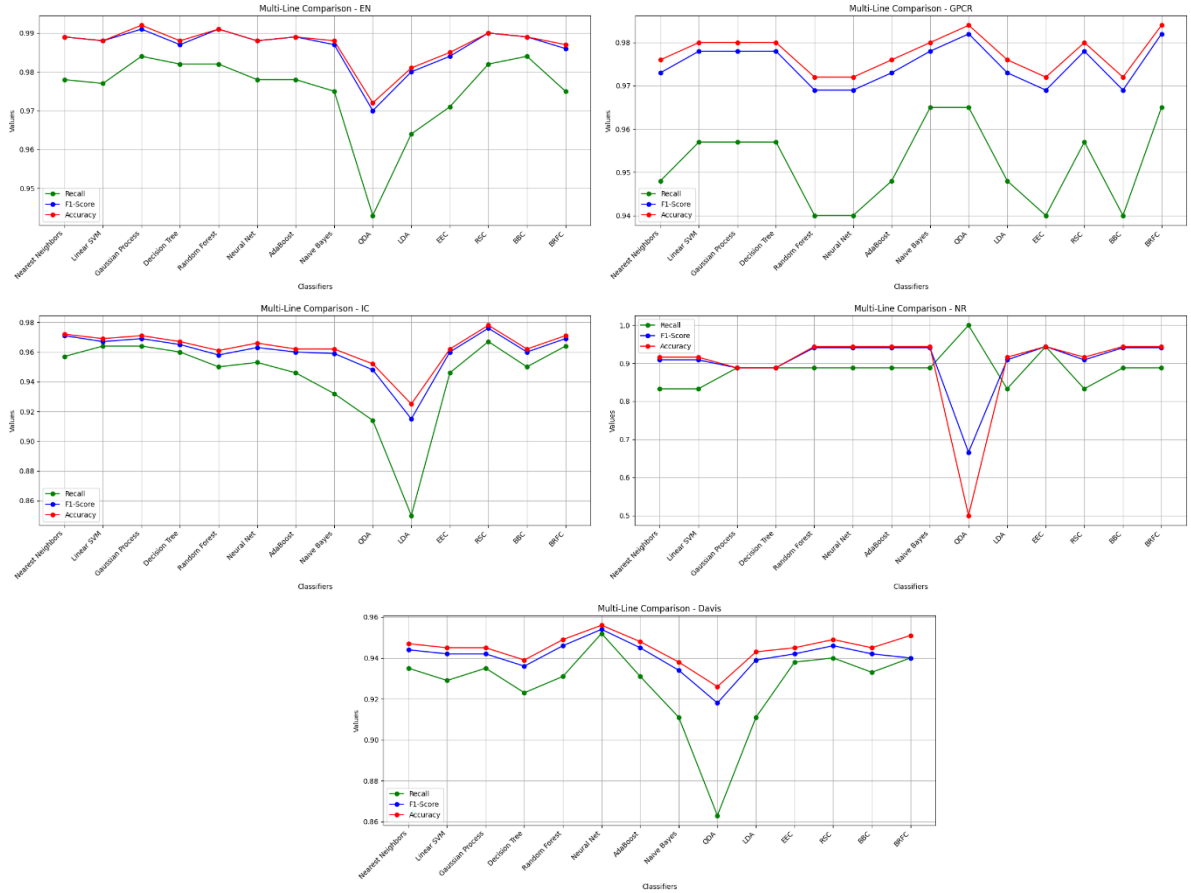

Figure 7. Illustrates the classification performance results of various classifiers after applying VAE for effective feature extraction. The first row (from left to right) corresponds to the EN and GPCR datasets, the second row (from left to right) represents the IC and NR datasets, and the final row displays the Davis dataset.

These insights from the multi-line visualization emphasize the need for dataset-specific classifier optimization, as the diverging trajectories in NR and IC contrast sharply with the convergent patterns in EN, GPCR, and Davis, informing strategies for improving model generalization in scientific computing applications. To demonstrate the effectiveness of the features extracted by the VAE, a Box Plot diagram was employed. Figure 8 presents the performance of all classifiers after utilizing the VAE for feature extraction, evaluated using five performance metrics across different datasets. As observed, all methods achieved satisfactory performance on all datasets. These results indicate that the features extracted by the VAE exhibit strong stability and robustness, consistently leading to reliable performance across all classifiers. The analysis of the Box Plot for the EN, GPCR, and Davis datasets reveals a consistently high and stable performance across most metrics, with medians ranging from 0.98 to 0.99 for Precision, Recall, F1-score, and Accuracy, accompanied by a narrow interquartile range (IQR) of less than 0.01 in many cases. This suggests a high degree of uniformity among classifiers in these datasets, where even the minimum values remain above 0.92, and only a few outliers are observed, particularly in ROC, which maintains an average above 0.98. In contrast, the IC dataset exhibits greater variability, especially in Recall (with an IQR of 0.017 and a minimum of 0.85), likely due to the sensitivity of certain classifiers like LDA to specific data characteristics. Nevertheless, Precision and ROC remain robust, indicating a generally reliable performance with room for improvement in recall consistency. The NR dataset, however, stands out with significant variability as highlighted by its Box Plot, featuring a high IQR of 0.098 in Precision and a minimum value of zero (e.g., for AdaBoost), which points to instability and the presence of extreme outliers. This dataset exhibits the weakest overall performance, though its ROC average of 0.93 suggests some resilience. A comparative overview across datasets indicates that EN and GPCR are better suited for precision-critical applications, while NR requires further optimization of classifiers to reduce variability and achieve a more balanced performance.

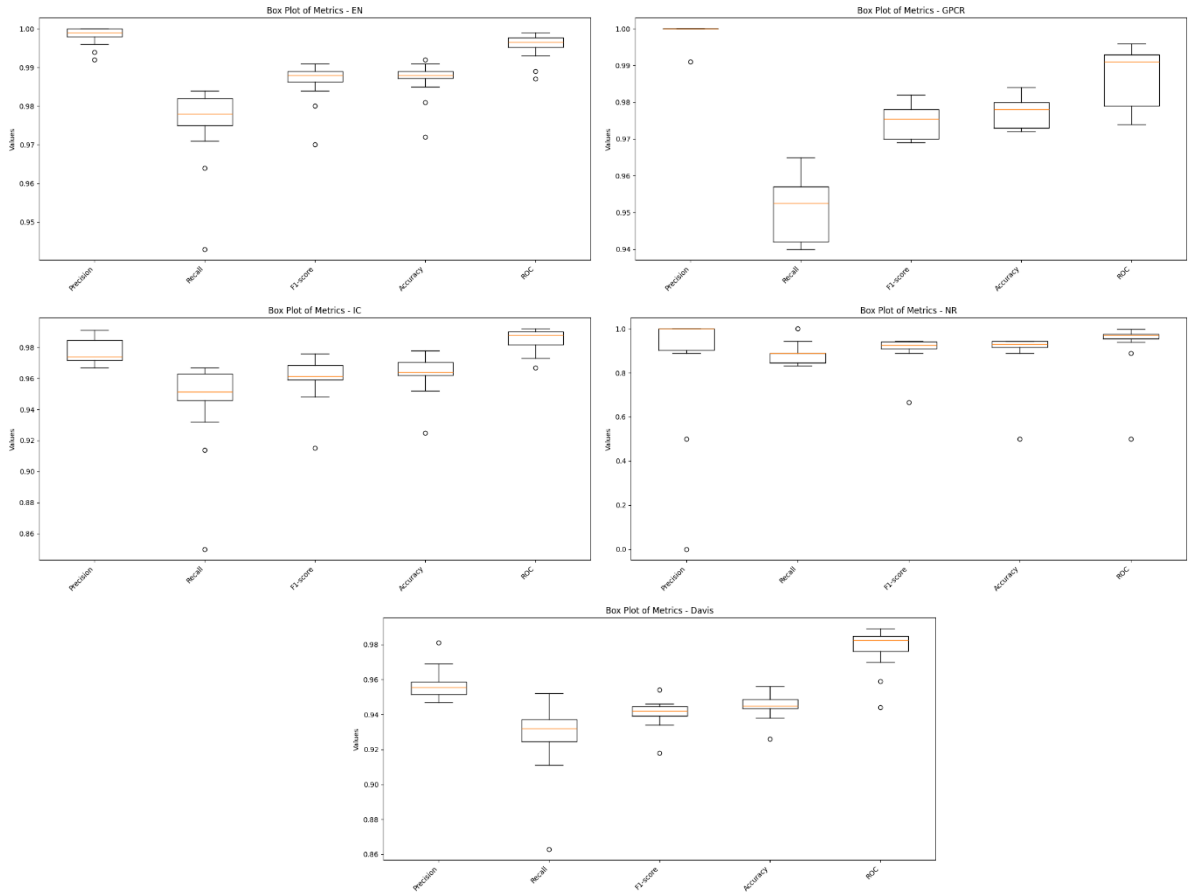

Figure 8. illustrates the output of the proposed methods after feature extraction using the VAE. The first row (from left to right) corresponds to datasets EN and GPCR, the second row (from left to right) represents datasets IC and NR, and the last row shows dataset Davis.

These insights underscore the importance of tailoring classifier selection to the specific characteristics and challenges of each dataset.

4.6. Comparison with Other Methods

In this section, the results obtained from the proposed method are compared with recent approaches in this field. For comparison, the methods of Musavian et al. [43], Pirgazi et al. [44], Mahmud et al. [45], Wang et al. [46], Mahmud et al. [47], Wang et al. [48], Meng et al. [49], and Li et al. [50] were used. To ensure a more precise evaluation, the results are based on 10 runs with averaging. Figure 9 presents the comparison between the proposed method and other approaches. As shown in the bar chart, the proposed method demonstrates very good performance based on the AUC-ROC metric and achieves the best results on four datasets: EN, GPCR, IC, and NR, outperforming the current state-of-the-art methods.

5. Conclusion

In this study, a novel framework for DTI prediction was proposed, combining diverse feature sets with advanced machine learning techniques. Experimental results on multiple benchmark datasets (EN, GPCR, IC, and NR) demonstrated superior performance in Precision, Recall, F1-Score, Accuracy, and AUCROC compared to state-of-the-art approaches. The primary strength of the proposed method lies in its two novel components. The One-SVM-US algorithm effectively addresses severe class imbalance, ensuring that minority class interactions are properly represented and reducing false negatives.

The modified, classification-oriented VAE refines feature representations in a supervised latent space, enhancing the model's ability to capture complex drug–target relationships. Our evaluations indicate that the synergistic combination of these components is the main reason for the observed performance improvement. While selecting informative protein features provides a solid foundation, substantial gains are consistently achieved only when the class imbalance is mitigated and the features are refined through the modified VAE. This demonstrates that both novel components are crucial to the framework's success. Overall, the proposed methodology offers a robust and effective tool for identifying novel drug–target interactions and can serve as a foundation for future studies in computational pharmacology and bioinformatics.

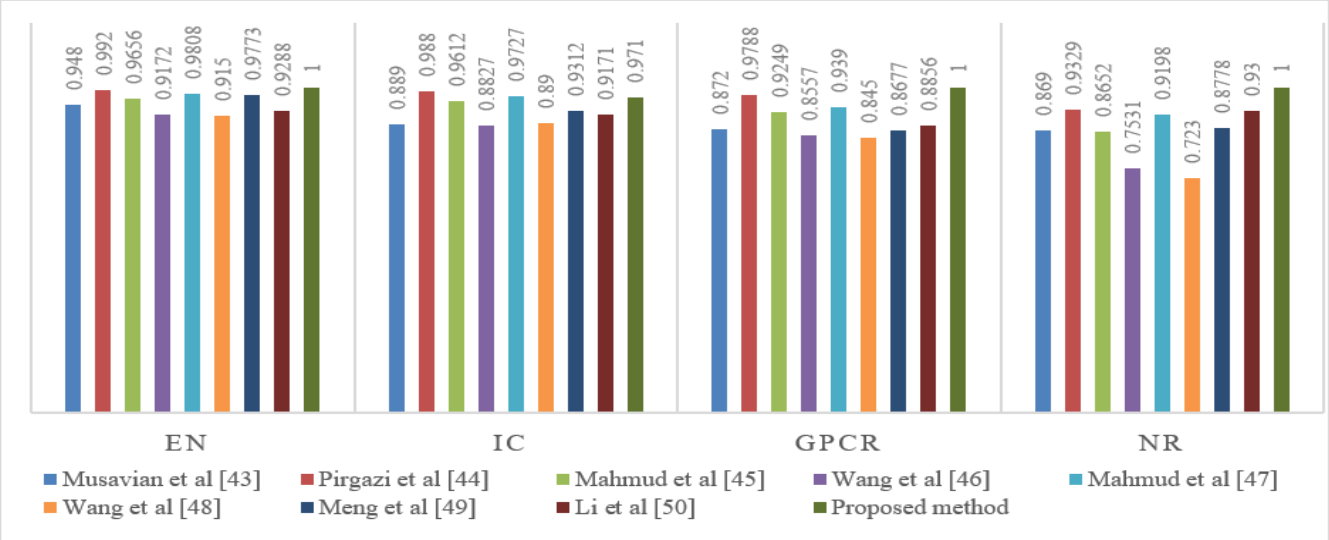

Figure 9. Comparison of the proposed method with recent state-of-the-art methods.

References

[1] S. M. Ivanov, A. A. Lagunin, P. V. Pogodin, D. A. Filimonov, and V. V. Poroikov, "Identification of drug targets related to the induction of ventricular tachyarrhythmia through a systems chemical biology approach," Toxicol. Sci., vol. 145, no. 2, pp. 321–336, 2015.

[2] O. E. Ibitoye and M. E. Soliman, "Machine learning in enhancing protein binding sites predictions – what has changed since then?" Comb. Chem. High Throughput Screen., vol. 28, no. 10, pp. 1640–1653, 2025.

[3] F. Yang, Q. Zhang, X. Ji, Y. Zhang, W. Li, S. Peng, and F. Xue, "Machine learning applications in drug repurposing," Interdiscip. Sci. Comput. Life Sci., vol. 14, no. 1, pp. 15–21, 2022.

[4] A. Ezzat, M. Wu, X.-L. Li, and C.-K. Kwoh, "Drug-target interaction prediction via class imbalance-aware ensemble learning," BMC Bioinformatics, vol. 17, no. Suppl 19, p. 509, 2016.

[5] H. Shi, S. Liu, J. Chen, X. Li, Q. Ma, and B. Yu, "Predicting drug-target interactions using Lasso with random forest based on evolutionary information and chemical structure," Genomics, vol. 111, no. 6, pp. 1839–1852, 2019.

[6] S. H. Mahmud, W. Chen, H. Jahan, Y. Liu, N. I. Sujan, and S. Ahmed, "iDTi-CSsmoteB: Identification of drug–target interaction based on drug chemical structure and protein sequence using XGBoost with over-sampling technique SMOTE," IEEE Access, vol. 7, pp. 48699–48714, 2019.

[7] Q. An and L. Yu, "A heterogeneous network embedding framework for predicting similarity-based drug-target interactions," Brief. Bioinform., vol. 22, no. 6, p. bbab275, 2021.

[8] A. G. Sorkhi, Z. Abbasi, M. I. Mobarakeh, and J. Pirgazi, "Drug–target interaction prediction using unifying of graph regularized nuclear norm with bilinear factorization," BMC Bioinformatics, vol. 22, no. 1, p. 555, 2021.

[9] H. El-Behery, A.-F. Attia, N. El-Fishawy, and H. Torkey, "Efficient machine learning model for predicting drug-target interactions with case study for Covid-19," Comput. Biol. Chem., vol. 93, p. 107536, 2021.

[10] D. Iliadis, B. De Baets, T. Pahikkala, and W. Waegeman, "A comparison of embedding aggregation strategies in drug–target interaction prediction," BMC Bioinformatics, vol. 25, no. 1, p. 59, 2024.

[11] A. Dehghan, K. Abbasi, P. Razzaghi, H. Banadkuki, and S. Gharaghani, "CCL-DTI: Contributing the contrastive loss in drug–target interaction prediction," BMC Bioinformatics, vol. 25, no. 1, p. 48, 2024.

[12] M. Kalemati, M. Zamani Emani, and S. Koohi, "DCGAN-DTA: Predicting drug-target binding affinity with deep convolutional generative adversarial networks," BMC Genomics, vol. 25, no. 1, p. 411, 2024.

[13] M. Yazdani-Jahromi, N. Yousefi, A. Tayebi, E. Kolanthai, C. J. Neal, S. Seal, and O. O. Garibay, "AttentionSiteDTI: An interpretable graph-based model for drug-target interaction prediction using NLP sentence-level relation classification," Brief. Bioinform., vol. 23, no. 4, p. bbac272, 2022.

[14] M. Li, H. Liu, F. Kong, and P. Lv, "DTRE: A model for predicting drug-target interactions of endometrial cancer based on heterogeneous graph," Future Gener. Comput. Syst., vol. 161, pp. 478–486, 2024.

[15] J. Wang, Y. Xiao, X. Shang, and J. Peng, "Predicting drug–target binding affinity with cross-scale graph contrastive learning," Brief. Bioinform., vol. 25, no. 1, p. bbad516, 2024.

[16] Z. Liu, Q. Chen, W. Lan, H. Lu, and S. Zhang, "SSLDTI: A novel method for drug-target interaction prediction based on self-supervised learning," Artif. Intell. Med., vol. 149, p. 102778, 2024.

[17] J. Lee, D. W. Jun, I. Song, and Y. Kim, "DLM-DTI: A dual language model for the prediction of drug-target interaction with hint-based learning," J. Cheminformatics, vol. 16, no. 1, p. 14, 2024.

[18] B.-W. Zhao, X.-R. Su, P.-W. Hu, Y.-A. Huang, Z.-H. You, and L. Hu, "iGRLDTI: An improved graph representation learning method for predicting drug–target interactions over heterogeneous biological information network," Bioinformatics, vol. 39, no. 8, p. btad451, 2023.

[19] A. Dalkıran, A. Atakan, A. S. Rifaioğlu, M. J. Martin, R. Ç. Atalay, A. C. Acar, T. Doğan, and V. Atalay, "Transfer learning for drug–target interaction prediction," Bioinformatics, vol. 39, no. Supplement_1, pp. i103–i110, 2023.

[20] R. Zhang, Z. Wang, X. Wang, Z. Meng, and W. Cui, "Mhtan-dti: Metapath-based hierarchical transformer and attention network for drug–target interaction prediction," Brief. Bioinform., vol. 24, no. 2, p. bbad079, 2023.

[21] M. Li, X. Cai, S. Xu, and H. Ji, "Metapath-aggregated heterogeneous graph neural network for drug–target interaction prediction," Brief. Bioinform., vol. 24, no. 1, p. bbac578, 2023.

[22] Q. Ye, X. Zhang, and X. Lin, "Drug–target interaction prediction via multiple classification strategies," BMC Bioinformatics, vol. 22, no. Suppl 12, p. 461, 2022.

[23] K. Huang, C. Xiao, L. M. Glass, and J. Sun, "MolTrans: Molecular interaction transformer for drug–target interaction prediction," Bioinformatics, vol. 37, no. 6, pp. 830–836, 2021.

[24] G. Liu, M. Singha, L. Pu, P. Neupane, J. Feinstein, H.-C. Wu, J. Ramanujam, and M. Brylinski, "GraphDTI: A robust deep learning predictor of drug-target interactions from multiple heterogeneous data," J. Cheminformatics, vol. 13, no. 1, p. 58, 2021.

[25] Y. Chu, A. C. Kaushik, X. Wang, W. Wang, Y. Zhang, X. Shan, D. R. Salahub, Y. Xiong, and D.-Q. Wei, "DTI-CDF: A cascade deep forest model towards the prediction of drug-target interactions based on hybrid features," Brief. Bioinform., vol. 22, no. 1, pp. 451–462, 2021.

[26] Y. Chu, X. Shan, D. R. Salahub, Y. Xiong, and D.-Q. Wei, "Predicting drug-target interactions using multi-label learning with community detection method (DTI-MLCD)," bioRxiv, 2020. [Online]. Available: https://doi.org/10.1101/2020.05.11.087734.

[27] T. Hinnerichs and R. Hoehndorf, "DTI-Voodoo: Machine learning over interaction networks and

ontology-based background knowledge predicts drug–target interactions," Bioinformatics, vol. 37, no. 24, pp. 4835–4843, 2021.

[28] M. Bhasin and G. P. Raghava, "Classification of nuclear receptors based on amino acid composition and dipeptide composition," J. Biol. Chem., vol. 279, no. 22, pp. 23262–23266, 2004.

[29] V. Saravanan and N. Gautham, "Harnessing computational biology for exact linear B-cell epitope prediction: A novel amino acid composition-based feature descriptor," OMICS, vol. 19, no. 10, pp. 648–658, 2015.

[30] T.-Y. Lee, Z.-Q. Lin, S.-J. Hsieh, N. A. Bretaña, and C.-T. Lu, "Exploiting maximal dependence decomposition to identify conserved motifs from a group of aligned signal sequences," Bioinformatics, vol. 27, no. 13, pp. 1780–1787, 2011.

[31] K.-C. Chou, "Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes," Bioinformatics, vol. 21, no. 1, pp. 10–19, 2005.

[32] T. I. Baig, Y. D. Khan, T. M. Alam, B. Biswal, H. Aljuaid, and D. Q. Gillani, "ILipo-PseAAC: Identification of lipoylation sites using statistical moments and general PseAAC," Comput. Mater. Contin., vol. 71, no. 1, pp. 215–230, 2022.

[33] Y. D. Khan, N. Rasool, W. Hussain, S. A. Khan, and K.-C. Chou, "iPhosT-PseAAC: Identify phosphothreonine sites by incorporating sequence statistical moments into PseAAC," Anal. Biochem., vol. 550, pp. 109–116, 2018.

[34] B. R. Donald, Algorithms in Structural Molecular Biology. Cambridge, MA, USA: MIT Press, 2023.

[35] E. Contreras-Torres, "Predicting structural classes of proteins by incorporating their global and local physicochemical and conformational properties into general Chou's PseAAC," J. Theor. Biol., vol. 454, pp. 139–145, 2018.

[36] H.-B. Shen and K.-C. Chou, "Nuc-PLoc: A new web-server for predicting protein subnuclear localization by fusing PseAA composition and PsePSSM," Protein Eng. Des. Sel., vol. 20, no. 11, pp. 561–567, 2007.

[37] S. Akbar, S. Khan, F. Ali, M. Hayat, M. Qasim, and S. Gul, "iHBP-DeepPSSM: Identifying hormone binding proteins using PsePSSM based evolutionary features and deep learning approach," Chemom. Intell. Lab. Syst., vol. 204, p. 104103, 2020.

[38] B. Yu, S. Li, W.-Y. Qiu, C. Chen, R.-X. Chen, L. Wang, M.-H. Wang, and Y. Zhang, "Accurate prediction of subcellular location of apoptosis proteins combining Chou's PseAAC and PsePSSM based on wavelet denoising," Oncotarget, vol. 8, no. 64, p. 107640, 2017.

[39] D. T. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," J. Mol. Biol., vol. 292, no. 2, pp. 195–202, 1999.

[40] X. Chen, Z. L. Ji, and Y. Z. Chen, "TTD: Therapeutic target database," Nucleic Acids Res., vol. 30, no. 1, pp. 412–415, 2002.

[41] K. C. Chou, "Prediction of protein cellular attributes using pseudo-amino acid composition," Proteins, vol. 43, no. 3, pp. 246–255, 2001.

[42] Z. Chen, P. Zhao, F. Li, A. Leier, T. T. Marquez-Lago, Y. Wang, G. I. Webb, A. I. Smith, R. J. Daly, and K.-C. Chou, "iFeature: A python package and web server for features extraction and selection from protein and peptide sequences," Bioinformatics, vol. 34, no. 14, pp. 2499–2502, 2018.

[43] Z. Mousavian, S. Khakabimamaghani, K. Kavousi, and A. Masoudi-Nejad, "Drug–target interaction prediction from PSSM based evolutionary information," J. Pharmacol. Toxicol. Methods, vol. 78, pp. 42–51, 2016.

[44] H. Khojasteh, J. Pirgazi, and A. Ghanbari Sorkhi, "Improving prediction of drug-target interactions based on fusing multiple features with data balancing and feature selection techniques," PLoS ONE, vol. 18, no. 8, p. e0288173, 2023.

[45] S. H. Mahmud, W. Chen, H. Meng, H. Jahan, Y. Liu, and S. M. Hasan, "Prediction of drug-target interaction based on protein features using undersampling and feature selection techniques with boosting," Anal. Biochem., vol. 589, p. 113507, 2020.

[46] L. Wang, Z.-H. You, X. Chen, X. Yan, G. Liu, and W. Zhang, "Rfdt: A rotation forest-based predictor for predicting drug-target interactions using drug structure and protein sequence information," Curr. Protein Pept. Sci., vol. 19, no. 5, pp. 445–454, 2018.

[47] S. H. Mahmud, W. Chen, Y. Liu, M. A. Awal, K. Ahmed, M. H. Rahman, and M. A. Moni, "PreDTIs: Prediction of drug–target interactions based on multiple feature information using gradient boosting framework with data balancing and feature selection techniques," Brief. Bioinform., vol. 22, no. 5, p. bbab046, 2021.

[48] Y. Wang, L. Wang, L. Wong, B. Zhao, X. Su, Y. Li, and Z. You, "ROFDT: Identification of drug–target interactions from protein sequence and drug molecular structure using rotation forest," Biology, vol. 11, no. 5, p. 741, 2022.

[49] F.-R. Meng, Z.-H. You, X. Chen, Y. Zhou, and J.-Y. An, "Prediction of drug–target interaction networks from the integration of protein sequences and drug chemical structures," Molecules, vol. 22, no. 7, p. 1119, 2017.

[50] Z. Li, P. Han, Z.-H. You, X. Li, Y. Zhang, H. Yu, R. Nie, and X. Chen, "In silico prediction of drug-target interaction networks based on drug chemical structure and protein sequences," Sci. Rep., vol. 7, no. 1, p. 11174, 2017.

متعادل‌سازی و اصلاح نمایش‌ها برای پیش‌بینی DTI: چارچوبی ترکیبی از یک SVM-US و یک VAE اصلاح‌شده

علی قنبری سرخی*، محدثه کیهانیان و جمشید پیرگزی

دانشکده مهندسی برق و کامپیوتر، دانشگاه علم و فناوری مازندران، بهشهر، بهشهر، ایران.

ارسال ۲۰۲۵/۱۰/۰۵؛ بازنگری ۲۰۲۵/۱۲/۱۹؛ پذیرش ۲۰۲۶/۰۱/۰۵

چکیده:

پیش‌بینی دقیق تعاملات دارو-هدف برای پیشبرد تلاش‌های کشف دارو و تغییر جایگاه آن ضروری است. این مطالعه چارچوبی جامع را معرفی می‌کند که به طور مؤثر چالش‌های کلیدی در پیش‌بینی DTI، از جمله عدم تعادل مجموعه داده‌ها و نمایش ویژگی‌ها با ابعاد بالا را برطرف می‌کند. این رویکرد، توصیف‌گرهای پروتئینی متعدد - به طور خاص، نه ویژگی آماری و مبتنی بر توالی - و اثر انگشت‌های مولکولی دارو را که از طریق الگوریتم‌های مورگان کدگذاری شده‌اند، با ترکیب‌های بهینه ویژگی که از طریق اعتبارسنجی انتخاب شده‌اند، ادغام می‌کند تا اطلاعات بیولوژیکی و شیمیایی متنوع را ثبت کند. برای کاهش عدم تعادل مجموعه داده‌ها، یک روش کم‌نمونه‌برداری مبتنی بر SVM تک کلاسه (One-SVM-US)، توزیع تعاملات مثبت را مدل‌سازی می‌کند تا منجر به کاهش نمونه‌ها از کلاس اکثریت شود و در نتیجه به طور مؤثر نمونه‌های مثبت و منفی را متعادل کند. علاوه بر این، یک روش تحت نظارت با عنوان خودرمزگذار متغیر مبتنی بر طبقه‌بندی برای فشرده‌سازی ویژگی‌های با ابعاد بالا به فضایی با ابعاد کمتر به کار گرفته می‌شود و در عین حال اطلاعات تمایز کلاس مربوط به پیش‌بینی تعامل را حفظ می‌کند. سپس ویژگی‌های اصلاح‌شده با استفاده از مدل‌های یادگیری ماشین برای پیش‌بینی جفت‌های بالقوه دارو-هدف طبقه‌بندی می‌شوند. ارزیابی‌های تجربی روی مجموعه داده‌های معیار، اثربخشی چارچوب پیشنهادی را نشان می‌دهد، به طوری که نتایج، امتیاز کامل AUC-ROC برابر با ۱.۰۰ را روی مجموعه داده‌های EN، GPCR و NR و امتیاز ۰.۹۷۳۱ را روی مجموعه داده‌های IC نشان می‌دهد که نشان‌دهنده بهبود عملکرد نسبت به روش‌های موجود است. این یافته‌ها، استحکام و پتانسیل این رویکرد را به عنوان ابزاری قابل اعتماد برای پیش‌بینی تعاملات دارو-هدف تأیید می‌کنند.

کلمات کلیدی: تعاملات دارو-هدف، خودرمزگذار متغیر، بازنمایی ویژگی‌ها، متعادل‌سازی داده‌ها.