# I. Introduction

Drug feature extraction is a key step in DTI prediction, which aims to extract representative features in drug molecules to help the model more accurately distinguish differences between drugs. Currently, the more commonly used methods include Morgan Fingerprints and Mol2Vec vectors, which is derived from the Word2Vec technique in natural language processing and maps molecular fragments into a continuous vector space through unsupervised learning. This method can fully express the chemical information in the molecular structure and generate continuous vectors. In 2019, Honda et al had compared various molecular representations such as ECFP, SMILES Transformer, and Mol2Vec and Mol2Vec showed stronger generalization ability on small sample datasets[1]. In 2022, Hua et al. willMol2Vec method for vector representation of drug molecular structures and combined it with graph structural features extracted by message-passing neural network, which improved the performance of the model in small sample tasks[2].

Target feature extraction is also an indispensable part of DTI prediction. In recent years Large Language Models (LLMs) have gradually shown advantages in processing target sequences, especially in tasks such as drug discovery and protein function prediction [3-9]. By pre-training Large Models can learn structural and functional information from sequences, which helps to improve the prediction of downstream tasks[10, 11]. 2022 Zhang et al. proposed a single sequence protein structure prediction method based on Protein Language Models around the task of protein function prediction[12].The method compares the prediction performance of the pre-trained ProtBERT model and the ProtBERT\_random model with random weight initialization, and the experimental results show that the pre-trained ProtBERT model is more stable in terms of classification performance, which highlights the efficient feature extraction capability of the ProtBERT model in biological sequence analysis.2023, Zhang et al. use the ProtBERT model to achieve efficient screening of anti-hypertensive peptides from soybean isolate proteins, and the AUC (area under the operator characteristic curve of the subjects) value in the experiment reached 0.9785[13]. It demonstrated that the model has high discriminative ability, and it is practical in the screening and prediction of protein sequences. In the same year, Djeddi et al. proposed the DTIOG prediction method, which combines Knowledge Graph Embedding (KGE) with ProtBERT-based protein sequence pre-training model, fusing structural knowledge with sequence context information[14]. The method was evaluated on several types of target datasets (e.g., enzymes, ion channels, and G-protein-coupled receptors), and all of them showed good adaptability and prediction ability, demonstrating its potential in the field of drug discovery.

# II. Materials and Methods

## B. Drug Sequence Feature Extraction

Mol2Vec innovatively introduces chemical linguistics concepts into molecular representation: first, the RDKit tool is used to parse SMILES into Morgan molecular fingerprint fragments, where each fingerprint fragment represents the structural characteristics of a specific atom within a particular neighborhood, analogous to a ‘word’; subsequently, the Word2Vec model is applied for distributed training in the chemical semantic space, ultimately generating the corresponding vector representation. The trained model can be used to convert new SMILES sequences into a set of fragment vectors, which are then integrated into a complete molecular feature representation through averaging or weighted averaging. Using the Mol2Vec method for feature encoding of drug sequences preserves the structural semantic information of the molecule and adapts to the requirements of subsequent model training. The process of feature encoding for drug molecules is shown in APPENDIX Figure 1.

|  |
| --- |
| APPENDIX Fig. 1. The process of feature encoding of drug molecules by the Mol2Vec model |

## C. Target sequence feature extraction

The ProtBERT model encodes each amino acid (e.g. M, A, P, etc.) as a ‘word’ when processing protein sequences, and adds two special tokens [CLS] and [SEP] before and after, so that the processed data is two tokens longer than the original data. Each token (amino acid character) is mapped into a 1024-dimensional vector after Tokenization encoding. In practice, the CLS token ([CLS] token) in the ProtBERT model is usually adopted as the global feature representation of the whole protein sequence, because [CLS] is a special token, which captures the semantics of the whole sequence and will not be specific to a particular amino acid. The pseudo-code flow shown in APPENDIX TABLEⅠdetails the standard operating paradigm for implementing protein feature extraction using the ProtBERT model.

APPENDIX TABLEⅠ

Pseudo-code for feature extraction of target sequences using ProtBERT

|  |
| --- |
| ***Algorithm 1*: F*eature extraction of target sequences using ProtBERT*** |
| ***input***: *sequence*: target sequence，A string composed of capital letters of amino acids；*moldel\_name*: ProtBERT |
| ***output***:*features*: The extracted 1024-dimensional feature vector |
| ***1.*** Load ProtBERT and the corresponding tokenizer  ***2.*** Initialize the empty list features\_list and use it to store the CLS vector of each sequence  ***3.*** **FOR EACH** seq in sequences **DO** |
| ***4.*** Connect the amino acid letters in the sequence with Spaces (conforming to the ProtBERT input format) |
| ***5.*** Encode the processed sequence using the tokenizer to obtain inputs |
| ***6.*** Feed the input data into the ProtBERT model to obtain the output tensor as outputs |
| ***7.*** Extract the vector representation of the [CLS] position from the output: cls\_embedding = outputs.last\_hidden\_state[:, 0, :] |
| ***8.*** Add cls\_embedding to the features\_list |
| ***9.*** Convert the features\_list to a tensor or array as features |
| ***10.* RETURN** features |

## D. SHAP dimensionality reduction

SHAP, as an innovative feature importance assessment tool, derives its theoretical basis from the Shapley value assignment principle in game theory. The method provides transparent interpretation capabilities for machine learning models by quantifying the marginal contribution of each feature to the model prediction results. In ensemble learning scenarios, SHAP demonstrates significant advantages: not only can it globally rank feature importance, but also generate heat maps of feature contributions with causal relationships, which greatly improves the scientific and interpretable nature of feature selection. Compared with the traditional dimensionality reduction techniques relying on statistical correlation or model weights, SHAP ensures the fairness of feature evaluation through rigorous mathematical derivation, and its visual interpretation capability makes the feature interactions of complex models clearly discernible, which provides data scientists with a more reliable basis for decision-making.

In this experiment, we use the Mol2Vec algorithm to encode the features of the drug molecule sequences to generate a 300-dimensional vector representation; meanwhile, we apply the ProtBERT pre-training model to extract the deep features of the target protein sequences to obtain a 1024-dimensional high-dimensional vector. In order to effectively deal with the high-dimensional feature space, this study innovatively introduces the SHAP interpretable analysis framework to implement the dimensionality reduction of the target feature vector, and the specific implementation process is as follows:

Firstly, the XGBoost classification model is constructed as a benchmark framework based on the complete feature space. Following that, the contribution of each feature dimension in the training set to the model prediction is quantitatively assessed by the SHAP value analysis tool, and the average absolute SHAP value is adopted as the core metric of feature importance. By systematically filtering the top 300 key features with the highest contribution, the feature dimensionality is significantly reduced while the discriminative information of the original data is retained to the maximum extent. Eventually, the prediction model is retrained on the optimized feature subset, and the impact of the dimensionality reduction process on the model performance is verified through comparative experiments. This feature selection method incorporating interpretable artificial intelligence techniques provides a new research idea for multimodal data integration in the biomedical field.

When applying the SHAP method for feature importance analysis during model training, we can obtain the SHAP values for each feature, which accurately quantify how much each feature contributes to the prediction results. By visualizing these SHAP values, it is possible to see which features have the greatest impact on the model predictions, and SHAP's feature summary plots not only rank all the features in order of importance, but also show the distribution of the specific impact of each feature on the model output.

APPENDIX Figure 2(1) shows the results of the visual analysis of the importance of SHAP features (Beeswarm diagram), with the horizontal coordinates indicating the distribution range of feature SHAP values, whose numerical magnitude directly reflects the strength of the feature's influence on the prediction results. The feature value distribution information coded using gradient colors (blue for the low value range and red for the high value range) forms a multidimensional data mapping relationship with the SHAP values. Taking feature\_102 as an example, it has the highest average absolute SHAP value among all features, indicating that it is one of the most significant features influencing the model prediction results. From the figure, it can be seen that when feature\_102 takes a lower value (blue), its SHAP value is mostly positive, which means that it tends to predict the samples to be positive; on the contrary, when the feature takes a higher value (red), the SHAP value is mostly negative, which pushes the model to predict the samples to be negative. This suggests that the lower the value of feature\_102, the more it contributes to the model's prediction of positive categories. APPENDIX Figure 2(2) A histogram of feature importance ranking was constructed by calculating the average absolute SHAP value of each feature using quantitative analysis. The chart is arranged in descending order, with the horizontal coordinate being the mean absolute SHAP value of the feature and the vertical coordinate being the feature name. The chart is sorted according to the importance of the features from high to low, which can quantitatively demonstrate the overall level of the contribution of each feature to the predictive output of the model.

|  |  |
| --- | --- |
| APPENDIX Fig. 2. Summary of the Importance of SHAP Features |  |

## E. Undersampling algorithm

Fuzzy Undersampling (Fuzzy Undersampling) is a kind of data sampling method for class imbalance problems, and its core idea is to use the affiliation function in fuzzy set theory to calculate the ‘fuzziness’ of the samples in the positive and negative classes[15]. The higher the value of fuzziness, the more fuzzy the boundary between the negative sample and the positive category, the more atypical the characteristics, identify and delete those redundant or fuzzy negative samples with large fuzziness, and retain the negative samples with smaller fuzziness to achieve undersampling.

The specific steps for fuzzy undersampling used in this study are as follows:

The first step is to train a preliminary classification model on the training set, in this experiment a K-Nearest Neighbors (KNN) model is used, which is used to estimate each sample's affiliation to the category to which it belongs (i.e., the ‘degree’ to which the sample belongs to a particular category)[16].

In the second step, the trained KNN model is used to predict the probability of a negative sample being classified as a positive sample. Based on the predicted probability, the fuzziness (fuzziness) metric of each negative sample is defined, where the closer the fuzziness is to 1 indicates that the sample is more difficult to distinguish (i.e., closer to the decision boundary), and the closer the fuzziness is to 0 indicates that the sample is easier to distinguish.

In the third step, the negative samples are ranked according to their fuzziness, and the samples are divided into the clearest (less fuzziness) and the fuzziest (more fuzziness) parts. According to the preset ratio (5:5 ratio of clear samples to fuzzy samples), a certain number of clearest samples and fuzzy samples are selected from the sorting result respectively to form a new set of negative samples.

The formula for calculating fuzziness is shown in Eq. (1):

Fuzziness=1-2×|p-0.5| (1)

where p denotes the probability that a negative sample is predicted to be positive.

This formula captures how close the negative sample is to the decision boundary (p = 0.5).If p is close to 0 or 1, the sample is easy to classify and has a low degree of ambiguity; if p is close to 0.5, the sample is located near the decision boundary and cannot be accurately classified, resulting in high ambiguity.

# III. Model Construction and Selection

## A. XGBoost

In this experiment, XGBoost predicts the probability of drug-target interactions by constructing a decision tree step by step. As shown in Eq. (2), first the model sets the initial prediction value as the global prior based on the proportion of positive and negative samples, where *p* is the proportion of positive samples in the dataset; then for each sample () in each iteration, as shown in Eq. (3-4), a new decision tree is trained based on the error of the current prediction (first-order gradient) and its rate of change (second-order gradient).The algorithm searches for optimal feature splits by split gain (Gain), while using regularisation to prevent overfitting. The output of each tree is scaled by the learning rate η and accrued to the model, and the final strong classifier obtained from multiple rounds of iterations is capable of mapping drug-target features into interaction probabilities and converting them into predicted values between 0 and 1 by means of a Sigmoid function. The whole training process ensures the prediction accuracy while controlling the complexity of the model.

) (2)

(3)

(4)

## C. Model structure

As shown in APPENDIX Figure 3, ProtFPreDTI innovatively constructs a dual-channel parallel processing architecture, where drug molecule sequences and protein target sequences are characterized and learnt through independent feature encoding modules respectively. In the model training phase, the feature-engineered data are fed into two parallel models, XGBoost and Random Forest, for independent training. In the prediction stage, based on the performance evaluation results of the validation set, the system dynamically optimizes the model fusion weights using a grid search algorithm (typical ratios are 80% weight contributed by XGBoost and 20% weight by Random Forest), and intelligently fuses the prediction probabilities of the two base models through a weighted integration strategy. This integrated learning method not only effectively retains the advantageous features of each base model, but also significantly improves the prediction performance through feature complementation, and the final output prediction results have higher credibility and generalization ability, providing a powerful computational biology tool for drug-target interaction research.

|  |
| --- |
| APPENDIX Fig. 3. The integrated structure of ProtFPreDTI |

The search process for the optimal hyperparameters of XGBoost and Random Forest models is achieved by Grid Search combined with Cross-Validation. Firstly, the hyperparameter grids to be searched are defined for XGBoost and Random Forest, such as n\_estimators in the range of [100,200], max\_depth in the range of [4,6,8,10,14], etc., and then GridSearchCV is used to carry out a 3-fold cross-validation for each set of parameter combinations, and the AUC, a robustness index, is used as the criterion for judging the model's merits. After rigorous parameter space traversal and validation set testing, the algorithm automatically filters out the hyper-parameter combinations that enable the model to achieve the optimal balance between generalization ability and prediction accuracy. This parameter optimization method is both efficient and reliable, and is particularly suitable for the task of model tuning in a moderately complex parameter space. The final hyperparameters of XGBoost and Random Forest are shown in APPENDIX TABLE II:

APPENDIX TABLE II

The hyperparameters of the model

|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **n\_estimators** | **learning\_rate** | **max\_depth** |
| XGBoost | 200 | 0.1 | 8 |
| RandomForest | 200 | **-** | 14 |

# IV. Results and Discussion

## A. Model Evaluation

In this study, a binary classification labelling method was used: if there is an interaction between the drug and the target, it is labelled as a positive sample (Label = 1), and if there is no interaction, it is labelled as a negative sample (Label = 0).To comprehensively assess the model prediction performance, five key metrics were selected: accuracy (ACC), area under the subject operating characteristic curve (AUC), area under the precision-recall curve (AUPR), sensitivity (SN) and specificity (SP)[17-21].Their definitions and formulas are shown in Eq. (5-9), respectively:

(5)

Accuracy is the ratio of the number of samples correctly classified by the model to the total number of samples. Where TP denotes the number of positive samples correctly predicted by the model as drug-target interactions, TN denotes the number of negative samples correctly predicted as non-interactions, FP denotes the number of negative samples misclassified as positive, and FN denotes the number of positive samples misclassified as negative.

(6)

The Receiver Operating Characteristic Curve (ROC Curve) is plotted by using sensitivity as the horizontal axis and specificity as the vertical axis. The area below the curve is the AUC value, and the larger the AUC value, the better the performance of the model.

(7)

AUPR is the area under the Precision-Recall Curve. P is the proportion of samples predicted to be positive classes that are truly positive and R is the proportion of all true positive class samples that are correctly predicted to be positive. A higher value of AUPR indicates that the model predicts a minority of classes better.

(8)

Sensitivity indicates the proportion of true positive examples correctly identified by the model to all actual positive examples.

(9)

Specificity indicates the proportion of true negative cases correctly identified by the model out of all actual negative cases.

## B. Effectiveness of ProtFPreDTI

In order to avoid the ‘curse of dimensionality’ and keep all the key features as much as possible, we experimented and compared the feature selection dimensions in the range of 250-350, and as shown in APPENDIX TABLE III, the prediction results of the model with SHAP feature selection are relatively higher compared with the results without feature selection. Under these feature selection comparisons, when the dimension of feature selection is 300 dimensions, all the indicators are relatively high, of which sensitivity and specificity are especially prominent, indicating that the pre-trained model after feature selection of this dimension has a stronger ability to discriminate positive and negative samples, and the model prediction ability is excellent.

APPENDIX TABLE III

Comparison of different outcomes of feature selection

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Feature dimension** | **ACC** | **AUC** | **AUPR** | **Sn** | **Sp** |
| **250** | 0.8473 | 0.9242 | 0.8329 | 0.8291 | 0.8751 |
| **260** | 0.8559 | 0.9282 | 0.8322 | 0.8264 | 0.8743 |
| **270** | 0.8537 | 0.9269 | 0.8353 | 0.8318 | 0.8722 |
| **280** | 0.8583 | 0.9265 | 0.8324 | 0.8199 | 0.8701 |
| **290** | 0.8576 | 0.9277 | 0.8302 | 0.8232 | 0.8727 |
| **300** | 0.8595 | 0.9273 | 0.8318 | 0.8261 | 0.8726 |
| **310** | 0.8601 | 0.9265 | 0.8312 | 0.8271 | 0.8715 |
| **320** | 0.8552 | 0.9301 | 0.8334 | 0.8259 | 0.8722 |
| **330** | 0.8534 | 0.9233 | 0.8285 | 0.8227 | 0.8704 |
| **340** | 0.8447 | 0.9258 | 0.8299 | 0.8237 | 0.8706 |
| **350** | 0.8579 | 0.9245 | 0.8273 | 0.8296 | 0.8717 |
| **1024** | 0.8395 | 0.9103 | 0.8233 | 0.8205 | 0.8654 |

# References

1. Honda, S., S. Shi, and H.R. Ueda, Smiles transformer: Pre-trained molecular fingerprint for low data drug discovery. arXiv preprint arXiv:1911.04738, 2019.

2. Hua, Y., et al., Protein drug interaction prediction based on attention feature fusion. J Comput Res Develop, 2022. 59(9): p. 2051–65.

3. Tian, Z., et al., Drug-target interaction prediction with collaborative contrastive learning and adaptive self-paced sampling strategy. Bmc Biology, 2024. 22(1).

4. Liu, J., et al., Drug-target Interaction Prediction By Combining Transformer and Graph Neural Networks. Current Bioinformatics, 2024. 19(4): p. 316–326.

5. Qiao, J., et al., A self-conformation-aware pre-training framework for molecular property prediction with substructure interpretability. Nature Communications, 2025. 16(1): p. 1–16.

6. Pang, C., et al., Deep generative models in de novo drug molecule generation. Journal of Chemical Information and Modeling.

7. Li, H., Y. Pang, and B. Liu, BioSeq-BLM: a platform for analyzing DNA, RNA, and protein sequences based on biological language models. Nucleic Acids Research, 2021. 49(22): p. e129.

8. Liu, B., X. Gao, and H. Zhang, BioSeq-Analysis2.0: an updated platform for analyzing DNA, RNA and protein sequences at sequence level and residue level based on machine learning approaches. Nucleic Acids Research, 2019. 47(20): p. e127.

9. Song, N., et al., Pmf-cpi: assessing drug selectivity with a pretrained multi-functional model for compound-protein interactions. Journal of Cheminformatics, 2023. 15(1).

10. Yao, Y., et al., A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. High-Confidence Computing, 2024. 4(2): p. 100211.

11. Liu, M., et al., Geometric deep learning for drug discovery. Expert Systems with Applications, 2023: p. 122498.

12. Zhang, Y., et al., HLAB: learning the BiLSTM features from the ProtBert-encoded proteins for the class I HLA-peptide binding prediction. Briefings in Bioinformatics, 2022. 23(5).

13. Zhang, Y., et al., Deep learning drives efficient discovery of novel antihypertensive peptides from soybean protein isolate. Food Chemistry, 2023. 404: p. 134690.

14. Djeddi, W.E., et al., Advancing drug–target interaction prediction: a comprehensive graph-based approach integrating knowledge graph embedding and ProtBert pretraining. BMC bioinformatics, 2023. 24(1): p. 488.

15. Jia, C., Y. Zuo, and Q. Zou, O-GlcNAcPRED-II: an integrated classification algorithm for identifying O-GlcNAcylation sites based on fuzzy undersampling and a K-means PCA oversampling technique. Bioinformatics, 2018. 34(12): p. 2029–2036.

16. Zhang, X., Z. He, and Y. Yang, *A fuzzy rough set-based undersampling approach for imbalanced data.* International Journal of Machine Learning and Cybernetics, 2024. **15**(7): p. 2799-2810.

17. Dao, F., et al., *RepliChrom: Interpretable machine learning predicts cancer-associated enhancer-promoter interactions using DNA replication timing.* iMeta, 2025. **n/a**(n/a): p. e70052.

18. Abdelkader, G.A. and J.D. Kim, *Advances in Protein-Ligand Binding Affinity Prediction via Deep Learning: A Comprehensive Study of Datasets, Data Preprocessing Techniques, and Model Architectures.* Current Drug Targets, 2024. **25**(15): p. 1041-1065.

19. Zhu, H., H. Hao, and L. Yu, *Identification of microbe–disease signed associations via multi-scale variational graph autoencoder based on signed message propagation.* BMC biology, 2024. **22**(1): p. 172.

20. Huang, Z., et al., *Accurate RNA velocity estimation based on multibatch network reveals complex lineage in batch scRNA-seq data.* BMC Biology, 2024. **22**(1): p. 290.

21. Guo, X., et al., *Highly Accurate Estimation of Cell Type Abundance in Bulk Tissues Based on Single‐Cell Reference and Domain Adaptive Matching.* Advanced Science, 2024. **11**(7): p. 2306329.