Contents lists available at ScienceDirect

International Journal of Biological Macromolecules

journal homepage: www.elsevier.com/locate/ijbiomac

PEL-PVP: Application of plant vacuolar protein discriminator based on PEFT ESM-2 and bilayer LSTM in an unbalanced dataset

Cuilin Xiao, Zheyu Zhou, Jiayi She, Jinfen Yin, Feifei Cui, Zilong Zhang [*]

School of Computer Science and Technology, Hainan University, Haikou 570228, China

ARTICLE INFO

Keywords:
Plant vacuolar proteins
ESM-2
Unbalanced dataset

ABSTRACT

Plant vacuoles, play a crucial role in maintaining cellular stability, adapting to environmental changes, and responding to external pressures. The accurate identification of vacuolar proteins (PVPs) is crucial for understanding the biosynthetic mechanisms of intracellular vacuoles and the adaptive mechanisms of plants. In order to more accurately identify vacuole proteins, this study developed a new predictive model PEL-PVP based on ESM-2. Through this study, the feasibility and effectiveness of using advanced pre-training models and fine-tuning techniques for bioinformatics tasks were demonstrated, providing new methods and ideas for plant vacuolar protein research. In addition, previous datasets for vacuolar proteins were balanced, but imbalance is more closely related to the actual situation. Therefore, this study constructed an imbalanced dataset UB-PVP from the UniProt database , helping the model better adapt to the complexity and uncertainty in real environments, thereby improving the model's generalization ability and practicality. The experimental results show that compared with existing recognition techniques, achieving significant improvements in multiple indicators, with 6.08 %, 13.51 %, 11.9 %, and 5 % improvements in ACC, SP, MCC, and AUC, respectively. The accuracy reaches 94.59 %, significantly higher than the previous best model GraphIdn. This provides an efficient and precise tool for the study of plant vacuole proteins.

1. Introduction

Plant vacuoles are multifunctional organelles crucial for maintaining cell turgor [1,2], regulating pH balance, participating in cell signaling, and responding to stress conditions. The storage of water, ions, nutrients, pigments, and secondary metabolites within vacuoles is vital for the development and growth of plants [3]. The three-dimensional structure of vacuolar proteins significantly influences their function, such as the vacuolar sorting receptors VSR4 and VSR6 [4,5], which promote protein localization to vacuoles by recognizing the C-terminal HDEL motif. The accurate localization of H + -pyrophosphatase-phosphatase (AVP1) is essential for regulating vacuolar ion concentration and pH, affecting plant growth regulation and stress response [6,7]. These vacuolar proteins also play roles in plant immune responses, such as activating disease resistance proteins upon pathogen signal detection, triggering cell defense mechanisms. This highlights the central role of vacuoles in regulating the intracellular environment of plant cells and responding to environmental changes, making the in-depth study of their protein structures and functions crucial for understanding plant adaptation mechanisms.

In order to fully understand the mechanism of vacuolar biogenesis, it is necessary to conduct in-depth research on the physiological and biochemical functions of plant vacuole proteins [8]. Traditional biological experimental methods not only consume time and cost, but also limit the rapid research progress in this field. Simultaneously, with the availability of the complete genome or proteome of any plant, subcellular localization becomes a viable option, depending on the subcellular location of the protein. However, experimental methods for subcellular localization are highly tedious and time-consuming. In the field of computational biology, techniques such as DeepLoc and WoLF PSORT [9] are commonly used for the localization of proteins to multiple organelles. These computational methods and tools can predict the distribution of proteins across various subcellular locations. Despite the existence of technologies like the OrganelX Web Server [10], which uses specialized classifiers for organelle classification, the accuracy of identifying specific organelles still needs improvement. Therefore, developing efficient computing strategies to identify plant vacuole proteins has become an urgent need.

[*] Corresponding author.
E-mail address: zhangzilong@hainanu.edu.cn (Z. Zhang).

https://doi.org/10.1016/j.ijbiomac.2024.134317
Received 27 April 2024; Received in revised form 10 July 2024; Accepted 28 July 2024
Available online 31 July 2024

In recent years, a wide variety of computational models for identifying organelle proteins have emerged [11–19]. For example, Lv et al. proposed a classifier for localizing proteins in the Golgi apparatus called GP-DRLF [20]; in 2021, Anteghini and colleagues developed the In-Pero model for identifying peroxisomal proteins [21]. However, research related to the computational prediction of plant vacuolar proteins (PVPs) is relatively scarce. In 2020, Yadav et al. introduced VacPred [22], a machine learning model, which utilizes the Support Vector Machine (SVM) algorithm combined with dipeptide composition (DPC) and k-interval position-specific scoring matrix (K-PSSM) feature extraction methods for predicting plant vacuolar proteins. In 2022, Jiao et al. [23] recognized the greater informational content of deep representation learning features compared to classical sequence features. They integrated deep learning embedded features with classical sequence features for feature representation, and input these features into a two-step feature selection process using Sequential Forward Selection (SFS) and DRLF. In 2023, Marco Anteghini (citation) and others combined deep learning-based pre-trained protein embeddings with machine learning classification methods. However, these studies primarily focus on protein sequences, overlooking the three-dimensional characteristics of proteins. Additionally, due to the limited data available in the datasets, it is not possible to use more advanced deep learning techniques to better extract the features of plant vacuoles and achieve more accurate identification. In 2023, Sui et al. pioneered GraphIdn [24], a cutting-edge model for identifying plant vacuolar proteins. This innovative approach integrates the AlphaFold2 algorithm with graph convolutional neural networks to extract intricate protein structural features. Subsequently, these features are sequentially fed into a multi-head attention module and a fully connected layer, culminating in the precise identification of plant vacuolar proteins.

With the development of large models in recent years and their application across various fields, there has been significant progress in the application of deep learning technologies in protein science [25–28], exemplified by the Evolutionary Scale Modeling (ESM) series of protein language models [29–35]. Since Facebook AI Research introduced the ESM-1 model, this series has effectively captured the evolutionary information of protein sequences through self-supervised learning methods, enhancing the understanding of the relationship between protein structure and function. Subsequent versions, such as ESM-1b and ESM-2 [36–38], have further improved performance in protein property prediction by expanding the model size and employing more advanced training techniques.

Given the limitations of existing tools in the identification of plant vacuolar proteins, this study employed the ESM-2 large model for fine-tuning and made corresponding improvements to develop a new plant vacuolar protein identification model PEL-PVP Parameter-Efficient Fine-Tuning (PEFT) on ESM-2, Implemented multidimensional feature extraction and feature fusion of plant vacuole protein sequences and spatial structures, and the large parameter set of ESM-2 maintained the generalization capability for protein recognition and ensured the accurate identification of plant vacuole proteins. Finally, with bilayer LSTM to identify PVP. To more closely align with the practical needs of the biological field, we meticulously selected and compiled data from the UniProt database to construct a specialized, imbalanced dataset for plant vacuolar proteins UB-PVP. The use of the UB-PVP dataset simulated real-world scenarios, while also expanding the quantity of the training dataset. This increase in data volume allowed for the simultaneous recognition of plant vacuole proteins during training, maintaining excellent generalization capability. Validation results on this dataset indicate that our proposed model exhibits outstanding performance. We anticipate that the successful implementation of this model will offer robust backing for research endeavors in the biological sciences, with broad applicability across various bioinformatics studies.

2. Materials and methods

2.1. Datasets

The selection of an appropriate and accurate dataset is crucial for the precision of a model, particularly in the field of scientific research, including bioinformatics. Imbalanced datasets, which are closer to real-world scenarios, reflect the complexity of nature and experimental conditions. Thus, we constructed an imbalanced dataset based on the publicly available UniProtKB/SwissProt database (as of March 11, 2024). UniProt (Universal Protein Resource) (UniProt) is a comprehensive protein resource database aimed at providing the scientific community with extensive information on protein sequences and functions. It serves as a vital resource for global biomedical and biotechnological research, offering a wealth of tools and services for protein study and analysis.

We used the following query to search the database: (taxonomy: viridiplantae, location: SL-0272, length:>50, and reviewed: Yes) and (taxonomy: viridiplantae, NOT location: SL-0272, length:>50 and reviewed: Yes). This search resulted in 38,202 plant vacuolar proteins (PVPs) and 609 non-plant vacuolar proteins (non-PVPs). To facilitate comparison with other models, we adopted the same independent balanced test set used by Yadav et al., consisting of 74 plant vacuolar proteins and 74 non-plant vacuolar proteins (non-PVPs), ensuring the independence of the test set by excluding these data from our queried dataset. This left us with 38,128 plant vacuolar proteins and 532 non-plant vacuolar proteins.

We use the CD-HIT [39] program to remove redundancy from the sequence and excluded entries containing non-standard amino acid sequences. Ultimately, we obtained 9916 non-plant vacuolar proteins and 408 plant vacuolar proteins, respectively, forming our imbalanced training set. Finally, we have created a plant vacuolar protein dataset composed of an imbalanced training set and a balanced independent test set.

2.2. Focal loss

Given the significant imbalance between PVP and on-PVP in the dataset, using traditional cross-entropy loss might lead the model to overemphasize the dominant class while neglecting the minority class. This imbalance could result in subpar performance on the minority class.

To address this issue, we have implemented the Focal Loss function [40,41]. This function introduces a focusing parameter that attenuates the loss for easier-to-classify examples, thus shifting the model's focus towards hard-to-classify samples. Specifically, Focal Loss reduces the loss associated with easier-to-classify examples, thereby minimizing their impact on the model's learning process. This mechanism enhances the model's focus on challenging samples, significantly improving its performance on minority classes.

$$\mathrm{FL}(\boldsymbol{p}_t) = f(\boldsymbol{x}) = \begin{cases} -\alpha_t(1-\boldsymbol{p}_t)^{\gamma} log(\boldsymbol{p}_t), & y = 1 \\ -(1-\alpha_t)\boldsymbol{p}_t{}^{\gamma} log(1-\boldsymbol{p}_t), & y = 0 \end{cases} \quad (1)$$

In our model, $\boldsymbol{p}_t$ represents the predicted probability assigned by the model for the actual category y, with α_t serving as a weight parameter designed to alleviate the imbalance among classes. The modulating factor γ, which we have set to 2, increases the emphasis on challenging samples, effectively reducing the loss contribution from those that are easily classified. This adjustment ensures a focused effort on the harder-to-classify instances, crucial for enhancing model performance in scenarios characterized by class imbalances.

2.3. Transformer layer

The Flowchart of PEL-PVP is shown in Fig. 1. The main architecture of PEL-PVP is a 30 layer cyclic Transformer, where the self-attention

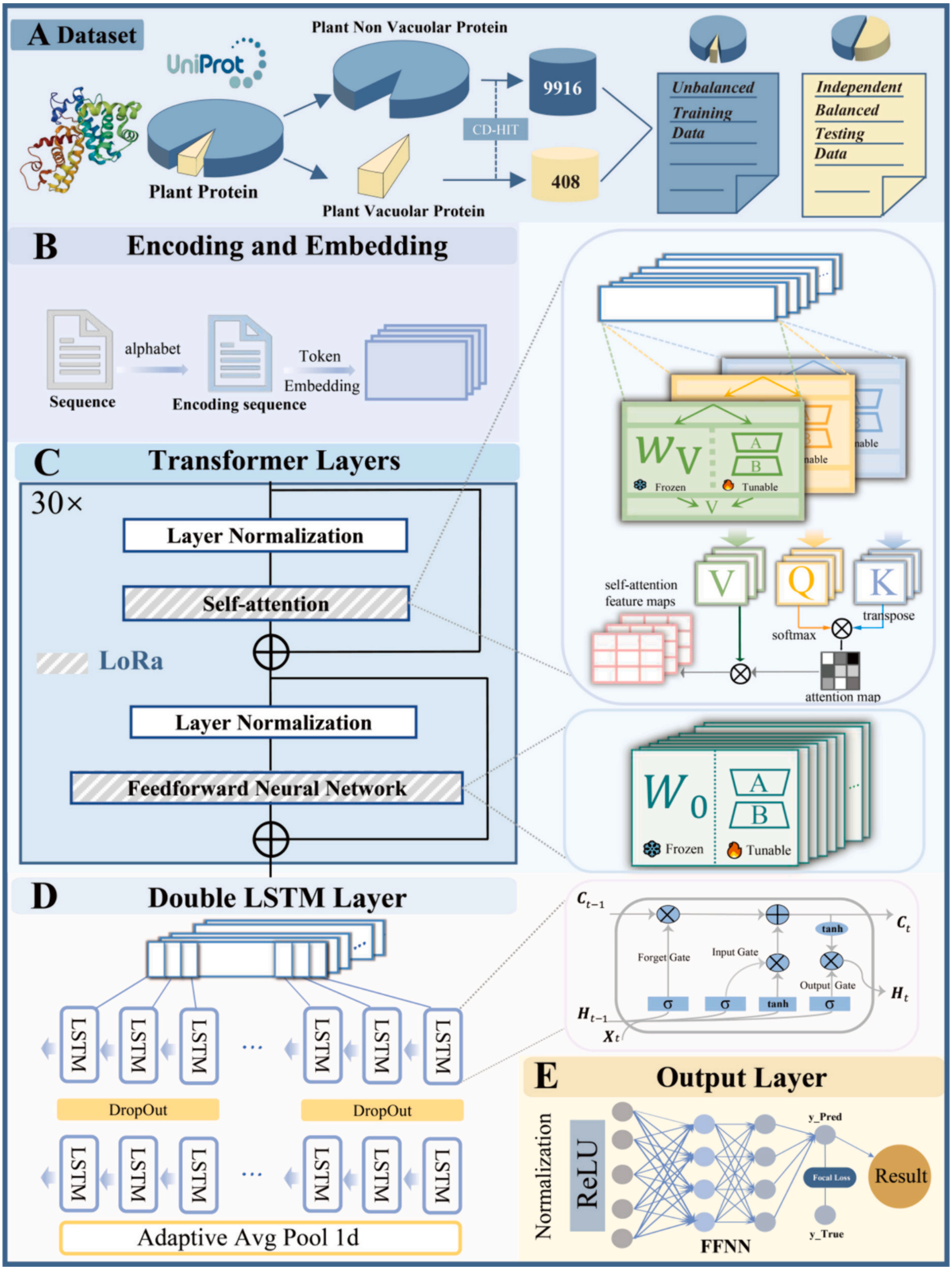

Fig. 1. The Flowchart of PEL-PVP. A. Dataset. The data were obtained from UniProt query, after CD-HIT the unbalanced training dataset of 408:9916 was obtained, and the independent test dataset is the same as the one prepared by Yadav et al. B. Encoding and Embedding. The input sequence is encoded with alphabet and then embedded using the embedding method in the torch library. C. Transformer Layers. (1–5 Layers) After layer normalization, the input sequence enters the self-attention module, and the output enters the feedforward neural network after layer normalization again. (5–30 Layers) The data flow is the same, but the parameters of q, k, v of the self-attention module and feedforward neural network (FFNN) of the original ESM2 are frozen, and the bottleneck structure of the LoRa low-rank adapter is trained for fine-tuning to learn the sequence characteristics of plant vacuolar proteins. D. Double LSTM Layer. Bidirectional LSTM is used to extract more complex latent features to enhance the expressive ability of the model, and a dropout layer is sandwiched in the middle to prevent overfitting. E. Output Layer. The extracted features are normalized and activated by the ReLU, and finally enter the FFNN for prediction, then use the Focal loss function to iterate the model parameters or obtain the final prediction results.

mechanism can directly calculate pairwise contacts between residues in the sequence, capturing dependencies and interactions between amino acid residues at different positions. These interactions are determined by the structure of the protein and reflect the protein structure in the sequence pattern.

PEL-PVP receives protein sequences as input and encodes it into Token embeddings that the Transformer can process. These embeddings are then passed through multiple stacked Transformer layers. Sure, here's a rephrased version: Each layer of the Transformer model comprises a self-attention mechanism along with a position-wise feedforward neural network (FFNN) enveloped by residual connections. The mechanism for self-attention achieves this by transforming the input features into vectors of three different dimensions—Q(query), K(key), V (value)—using three distinct weight matrices. It performs dot products to compute scaled attention scores, which are then converted into a probability distribution. These scores are used to weight the sum of the values, resulting in the output of the self-attention. The Feedforward Neural Network (FFNN) is comprised of two simple linear structures, followed with a ReLU activation function to generate a series of hidden states. The formula is as follows:

$$\mathbf{Q} = \mathbf{W}_q\mathbf{X} \tag{2}$$

$$\mathbf{K} = \mathbf{W}_k\mathbf{X} \tag{3}$$

$$\mathbf{V} = \mathbf{W}_v\mathbf{X} \tag{4}$$

$$\text{Self} - \text{Atention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{QK}^T}{\sqrt{\mathbf{d}}}\right)\mathbf{V} \tag{5}$$

Within this framework, $\mathbf{W}_q$, $\mathbf{W}_k$, and $\mathbf{W}_v$ represent three distinct linear transformations that project the input XX into vectors $\mathbf{W}_q$, $\mathbf{W}_k$, and $\mathbf{W}_v$, respectively. Here, dkdk denotes the dimensionality of the query vectors $\mathbf{Q}$ and key vectors $\mathbf{K}$. Initially, the dot product of the query vectors $\mathbf{Q}$ and key vectors $\mathbf{K}$ is scaled by $\sqrt{\mathbf{d}}$, followed by the conversion into attention weights through the softmax function. Subsequently, these attention weights are multiplied with the value vectors vv to generate the output. This process enables the model to assign different levels of importance to various positions within the computation, thus dynamically capturing pertinent information across the input sequence.

2.4. Using LoRa to parameter-efficient fine tune ESM-2 parameters

To leverage the powerful performance of ESM-2 in protein sequence related tasks, we loaded the massive pre-training parameters of the Transformer module in ESM-2 into PEL-PVP. Although we opted for the ESM2-150 M with fewer parameters as our pre-trained backbone, fine-tuning still requires substantial computational resources. Traditional fine-tuning methods such as Fine Tuning and Prompt-Tuning typically necessitate adjusting a large number of parameters or providing ample annotated data. To minimize the number of parameters and computational complexity during the fine-tuning process, enhance the performance of the pre-trained model on new tasks, and reduce the training costs of the pre-trained model, we employed Parameter-Efficient Fine-Tuning (PEFT) techniques, including Adapter, Prefix Tuning, and LoRA [42–44]. Prior to the advent of LoRA, the two most prevalent local fine-tuning methods were Adapter Tuning and Prefix Tuning. The introduction of Adapters increased the overall depth of the model, complicating the training process and reducing inference speed. Prefix Tuning appears to be straightforward, it presents two significant disadvantages: difficulty in training and the model's performance not strictly improving with the increase in prefix parameter volume, as mentioned in the original publication. Considering these issues, we opted for the state-of-the-art Low-Rank Adaptation (LoRA) technique.

LoRA is built on the idea that its trainable parameters can be effectively learned even when they are projected into a lower-dimensional subspace. This is because these parameters exhibit a low “intrinsic rank”. This approach reduces the number of fine-tuning parameters from $\boldsymbol{d} \times \boldsymbol{d}$ to $2 \times \boldsymbol{r} \times \boldsymbol{d}$, without altering the dimensionality of the output data. We achieved fine-tuning of ESM-2 by low rank adaptation of components in the Transformer architecture, with minimal computational cost [45]. Specifically, we apply this reparameterization to queries (Q), keys (K), values (V), and feedforward neural networks (FFNN) in Transformer [46]. Initially, we randomly initialize a Gaussian matrix A. Then, another low-rank decomposition matrix B is initialized to zero, ensuring that BA = 0 at the outset to preserve the primitive knowledge embedded in ESM-2. We freeze the pretrained weight matrix $\boldsymbol{W}_0$, keeping it unchanged. This means that during the training process, only the low-rank decomposition matrix $\Delta\boldsymbol{W}$ is trained and updated, while the original pretrained weights remain fixed. Throughout the training phase, we update A and B through gradient updates to train the low-rank decomposition matrix $\Delta\boldsymbol{W}$. This allows the model to dynamically adjust and learn new feature representations based on the training data without altering the pretrained weight matrix $\boldsymbol{W_0}$.

$$\boldsymbol{W} = \boldsymbol{W}_0 + \frac{\alpha}{r}\Delta\boldsymbol{W} \tag{6}$$

$$\Delta\boldsymbol{W} = \boldsymbol{BA} \tag{7}$$

Among them $\boldsymbol{W}$ are the model parameters after adding Lora, $\boldsymbol{W}_0$ are the primitive parameters of the pre-trained model, $\boldsymbol{A}$ and $\boldsymbol{B}$ are two low rank matrices. This method effectively reduces the number of updated model parameters during training while maintaining the model's expressive power and performance. By integrating with the original weight matrix, LoRa enables lightweight fine-tuning of the model in resource-constrained environments, ensuring the model's efficiency and adaptability.

2.5. Bilayer LSTM

In bioinformatics, LSTM (Long Short-Term Memory) [47] networks find extensive application across genomics, protein structure prediction, and various other domains. Their efficacy in these fields stems from their ability to capture long-term dependencies in sequence data effectively and mitigate issues such as vanishing or exploding gradients. Moreover, they excel in tasks like genomics, protein structure prediction, sequence alignment, classification, and clustering, making them well-suited for biological sequence analysis. In this article, in order to extract deeper and longer sequence features from plant vacuole proteins and achieve better expression, we used a double-layer LSTM module, and to prevent overfitting, a dropout layer was sandwiched between the double-layer LSTM.

The LSTM architecture consists of input gates, output gates, and forget gates. These components serve distinct functions: the input gate manages the integration of new input information into the cell state, the forget gate regulates the retention of old information within the cell state, and the output gate governs the output of information from the cell state. The calculation formula is as follows:

$$\boldsymbol{f}_t = \sigma\left(\boldsymbol{W}_f \bullet [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_f\right) \tag{8}$$

Among them, σ is the sigmoid function, which maps the input real value to an output in the range of 0 to 1; $\boldsymbol{W}_f$ represents the weighting matrix associated with the forget gate; $\boldsymbol{h}_{t-1}$ denotes the hidden state from the preceding time step $\boldsymbol{t} - 1$; $\boldsymbol{x}_t$ signifies the input at the temporal juncture $\boldsymbol{t}$; $\boldsymbol{b}_f$ denotes a bias component.

The input gate assimilates new data and adjudicates the extent of information to be incorporated into the cell's internal state. The calculation formula is as follows:

$$\boldsymbol{i}_t = \sigma(\boldsymbol{W}_i \bullet [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_i) \tag{9}$$

$$\widetilde{\boldsymbol{C}}_t = tanh(\boldsymbol{W}_c \bullet [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_c) \tag{10}$$

Among them, $\boldsymbol{W}_i$ and $\boldsymbol{W}_c$is the corresponding weight matrix, $\boldsymbol{b}_i$ and $\boldsymbol{b}_c$

is a bias term. Add the internal state $\widetilde{\boldsymbol{C}}_t$ of the cell from the previous time step $\boldsymbol{C}_{t-1}$ to obtain the current state, and the calculation formula is as follows (where ∘ represents Hadamard product - element by element multiplication):

$$\boldsymbol{C}_t = \boldsymbol{f}_t \circ \boldsymbol{C}_{t-1} + \boldsymbol{i}_t \widetilde{\boldsymbol{C}}_t \tag{11}$$

The output gate orchestrates the conveyance of information from the cell's internal state to the subsequent temporal interval. The calculation equation is as follows:

$$\boldsymbol{o}_t = \sigma(\boldsymbol{W}_o \bullet [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_o) \tag{12}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \circ tanh(\boldsymbol{C}_t)$$

Among them, $\boldsymbol{W}_o$is the corresponding weight matrix and $\boldsymbol{b}_o$ denotes a bias component. By leveraging these three gating mechanisms, LSTM effectively tackles the issue of gradient vanishing while adeptly handling sequential data.

2.6. Performance evaluation

In evaluating our model's performance, we employed a suite of rigorous metrics to provide a comprehensive and objective assessment. These metrics included Sensitivity (Sn), Specificity (Sp), Accuracy (ACC), Matthews Correlation Coefficient (MCC), Area Under the Receiver Operating Characteristic Curve (AUC), Area Under the Precision-Recall Curve(AUPRC), Precision, Recall, and F1 Score.

Sensitivity (Sn) gauges the model's proficiency in accurately classifying positive samples, i.e., the proportion of true positives correctly identified. Specificity (Sp) measures the model's effectiveness in identifying negative samples, i.e., the proportion of true negatives correctly predicted. Accuracy (ACC) quantifies the overall prediction accuracy across all samples, representing the percentage of all samples that are correctly classified. However, in imbalanced datasets, the metric's reliability may wane due to the disproportionate ratio of positive to negative samples, necessitating the use of additional metrics for a more rounded evaluation.

Matthews Correlation Coefficient (MCC) offers a holistic measure of model performance, encapsulating all four fundamental indicators: TP, TN, FP, and FN. Area Under the ROC Curve (AUC) serves as a quantitative index of the model's predictive accuracy. This metric reflects the model's discriminative ability at various threshold levels, taking into account both the True Positive Rate (TPR) and False Positive Rate (FPR). Precision and Recall provide insights into the model's performance on positive and negative samples respectively. Precision assesses the proportion of positive identifications that are correct, while Recall measures the percentage of actual positives that the model successfully predicts. These metrics are particularly valuable in imbalanced datasets, illuminating the model's performance dynamics across different classes. Area Under the Precision-Recall Curve (AUPRC) reflects the model's predictive performance across various recall levels and is calculated based on the precision-recall curve at different classification thresholds. Generally ranging from 0 to 1, higher AUPRC values denote better performance and are especially useful for evaluating models in datasets where positive samples are scarce. The F1 Score, a harmonic mean of Precision and Recall, synthesizes both metrics to provide a single measure that balances the precision-recall trade-off. This score is especially indicative of a model's efficacy in imbalanced datasets, often providing a more representative gauge of performance than accuracy alone. These metrics collectively facilitate a nuanced understanding of the model's capabilities, ensuring robust evaluation in diverse dataset conditions.

During the experiment, to comprehensively evaluate the model's performance on unbalanced datasets, we focused particularly on the Matthews Correlation Coefficient, Precision, Recall, Area Under the Precision-Recall Curve (AUPRC), and F1 Score [48]. On the other hand, for the balanced independent test set, we primarily used Accuracy (ACC) to compare our model against other existing models. By utilizing these stringent evaluation metrics, we can more accurately assess the model's performance in predicting vacuolar proteins, providing more reliable outcomes for future research and applications. The formula is as follows:

$$\mathrm{SN} = \frac{TP}{TP + FN} \tag{13}$$

$$SP = \frac{TN}{TN + FP} \tag{14}$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

$$\mathrm{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TN + FN) \times (TP + FP) \times (TN + FP) \times (TP + FN)}} \tag{16}$$

$$Recall = \frac{TP}{TP + FN} \tag{17}$$

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{18}$$

$$F_1 = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{19}$$

3. Experiment and discussions

3.1. Perform ablation experiments with the baseline model to demonstrate the effectiveness of the model

In the process of dataset selection, we fully considered the practical research scenarios and decided to construct an imbalanced dataset for model training. We trained the model using both balanced and imbalanced datasets, and the results are as follows:

From Table 1 and Fig. 2, it is evident that due to the limitations in data quantity, the model trained on the balanced dataset did not perform ideally. In contrast, the model trained on the imbalanced dataset showed outstanding performance across all metrics, significantly outperforming the model trained on the balanced dataset in every aspect. Specifically, the model trained on the imbalanced dataset excelled in accuracy (Acc), specificity, Matthews correlation coefficient (MCC), F1 score, AUC value, and AUPRC value.

We believe this performance improvement is primarily due to the increased data volume provided by the imbalanced dataset. This expansion of the original dataset compensates for the suboptimal training results seen with smaller data quantities and enhances the model's generalization ability. Therefore, we chose to use the imbalanced dataset for training to achieve better model performance and generalization capability.

3.2. Perform ablation experiments with the baseline model to demonstrate the effectiveness of the model

PEL-PVP, is composed of several fundamental modules, including four main architectural elements: the Transformer architecture, the LoRa low-rank adapter, a dual-layer LSTM, and a Feedforward Neural Network (FFNN). The FFNN module, which is crucial for the final output, is essential and therefore does not undergo ablation. The first three components are key to the structure of PEL-PVP, and we conduct ablation studies on these to explore their importance. We assess their impact using metrics such as Sensitivity (Sc), Sensitivity (Sn), Accuracy (Acc), Matthews Correlation Coefficient (Mcc), Area Under the Precision-Recall Curve (AUPRC), Precision, Recall, and F1-Score. By removing these core architectures from PEL PVP, we created various baseline models. Transformer (PEFT esm2): Transformer that performs

Table 1
Comparison of our model trained by imbalanced dataset and balanced dataset.

Dataset	Acc (%)	Specificity(%)	Sensitivity(%)	MCC(%)	F1(%)	AUC(%)	AUPRC(%)
Banlanced	82.43	72.97	91.89	66	83.95	89.66	89.16
Imbanlanced	**94.59**	**98.65**	**90.54**	**89.5**	**94.37**	**98.34**	**98.65**

Bold values are the models that achieve the best performance.

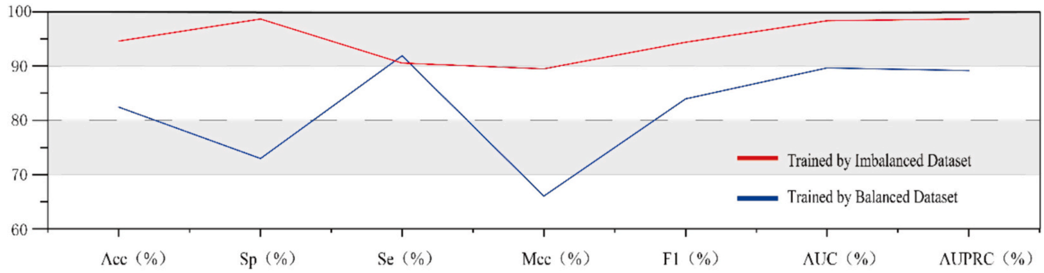

Fig. 2. Comparison of our model trained by imbalanced dataset and balanced dataset.

PEFT on ESM-2 parameters; Transformer+2LSTM: Transformer that does not include ESM-2 parameters combined with double-layer LSTM; Transformer (esm2): Transformer that does not perform PEFT on ESM-2 parameters; 2LSTM: double-layer LSTM. Evaluate their predictive capabilities on an unbalanced dataset, as illustrated in Fig. 3.

From Fig. 3, it's evident that while Accuracy largely remains at a high level across the unbalanced dataset, the elimination of any architectural component results in a decline in the model's predictive capabilities, specifically reflected in decreases in Precision, Recall, and F1-Score. This underscores the crucial role of each architectural element in the overall model structure. It is worth noting that using Transformer that does not perform PEFT on ESM-2 parameters and double-layer LSTM performs poorly, indicating that they do not perform well on imbalanced datasets. When these two components are combined, the F1 score of the model is

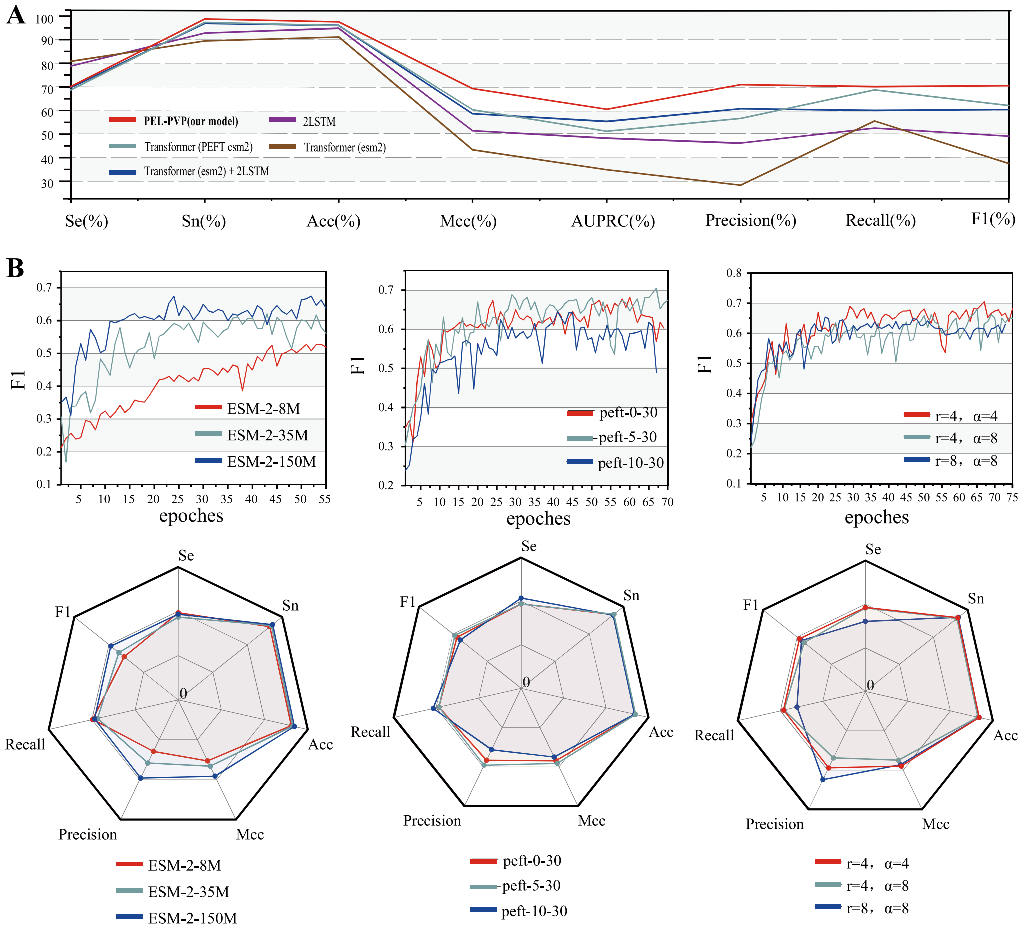

Fig. 3. A. Baseline model ablation experiment Perform ablation on the baseline model, including unmodified ESM-2, double-layer LSTM, combine unmodified ESM −2 with double-layer LSTM, fine-tuning ESM −2 and our model PEL-PVP. B. Selection of model related architecture and parameters Explored different pre trained ESM-2 models, LoRa low rank adaptation applied to different regions, and the impact of rank size in LoRa on the model.

60.38 %. This combination model benefits from the spatial information extraction ability of Transformer and the long-term dependency processing of double-layer LSTM, which improves its overall recognition performance. In addition, Transformer that performs PEFT on ESM −2 parameters can further improve performance, with an F1 score of 62.1 %, indicating that adaptive fine-tuning specifically targeting plant vacuole proteins can significantly enhance ESM-2's ability to recognize these proteins.

Integrating these components, the final PEL-PVP model achieves an F1-Score of 70.11 %. This performance is attributed to the model not only leveraging the extensive pre-trained parameters of ESM-2 but also effectively combining the spatial information processing of the Transformer with the long-range dependency management of the dual-layer LSTM, and applying adaptive fine-tuning tailored for plant vacuolar proteins. This strategic integration and enhancement make PEL-PVP a robust tool in recognizing plant vacuolar proteins with high efficacy.

3.3. Impact of different ESM-2 pre-training parameter sizes

In the adjustment process of PEL-PVP, we sequentially used three pre trained ESM-2 parameter with different parameter sizes - ESM-2-8 M, ESM-2-35 M, and ESM-2-150 M. We recorded the changes in their F1 scores during the training process. As shown in the Fig. 3, the results show that all models have a significant initial improvement in F1 scores, indicating that even after several training iterations, the knowledge gained during the pre-training process can quickly benefit. However, as the training progresses, the performance differences between different models become apparent. ESM-2-8M showed the slowest improvement and achieved the lowest F1 final score. Therefore, we will focus on ESM-2-35M and ESM-2-t150M. The ESM-2-35M model showed a moderate performance improvement rate, while the ESM-2-150M model consistently maintained the highest F1 score during most training periods.

It's also worth mentioning that while other larger pretrained models like ESM2-650M, ESM2-3B, and ESM2-15B are available and might offer improved performance, computational resource limitations necessitated their exclusion from our study. Consequently, we chose to use the parameters of ESM-2-150M and make minor adjustments. This decision was based on balancing computational feasibility and the potential for achieving superior model performance, making ESM-2-t30 the optimal choice within the constraints of our available resources.

3.4. LoRa applied to different regions for fine-tuning

In our initial experiment, we applied PEFT to the ESM-2 parameters of each layer of Transformer. However, this approach risked diminishing the original ESM-2 model's expressive capabilities. To address this concern, we conducted an ablation analysis on the layers to which the LoRa were applied, intended to evaluate the impact of different LoRa application strategies on model performance during the fine-tuning process. Specifically, we tested three strategies: Applying LoRa to all 30 layers, layers 5 to 30, and layers 10 to 30., using the F1 score as the performance evaluation metric.

The results, as shown in the Fig. 3. Applying LoRa to all 30 layers does not yield the best results, as the strategy of applying LoRa to layers 5 to 30 achieved the highest F1 score for most of the training period. The performance of applying LoRa to the strategy of layers 10 to 30 is the least effective. This discovery emphasizes that completely fine-tuning all layers of the model does not yield the best results. However, fine-tuning too few layers cannot effectively learn the characteristics of plant vacuole proteins. Freezing a certain number of layers appropriately is crucial for maintaining pre trained knowledge and effectively adapting to new tasks. We believe that due to the strong expression ability of the original ESM-2 model, retaining some of its parameters during fine-tuning can help better extract protein sequence features.

Considering these results, PEL-PVP model adopts the strategy applying LoRa to layers 5 to 30 to achieve the best possible performance. This approach balances the need to retain valuable pre-trained knowledge and the necessity to adapt to the specific requirements of predicting plant vacuolar proteins.

3.5. Exploring the effect of rank size on modeling in LoRa

In the LoRa architecture, two crucial hyperparameters, r and α, play pivotal roles. Here, α acts as a scaling factor, akin to a learning rate in traditional settings, affecting how adjustments are made to the learning process. The parameter r represents the rank in low-rank adaptation, influencing the complexity and detail of the information captured from the data. With a smaller r, the model focuses on extracting the most significant and informative dimensions, leading to a refined but potentially incomplete view of the data. Conversely, a larger r allows for a low-rank approximation that more closely mirrors the full data complexity, providing a comprehensive overview but possibly introducing more noise and redundant information.

To identify the optimal settings for these parameters, we conducted ablation experiments with different values of r. As shown in the Fig. 3, the training data indicates that the model's F1 score on the validation set is significantly influenced by the values set for r and α. Our observations revealed that setting both r and α to 4 yields the highest F1 score throughout the training cycle. This outcome suggests that the configuration with $r = 4$ and $\alpha = 4$ extracts the richest content in terms of information and delivers the best performance for this specific task. This optimal parameter setting helps the model balance between detail and noise, ensuring effective learning and generalization.

3.6. PEL-PVP can extract spatial information

In protein structure research, contact maps and distance maps are two common graphical representation methods used to depict the structural features and inter-atomic relationships within proteins. Both play a significant role in studying the structure, function, and interactions of proteins. Contact maps primarily showcase the interactions between residues within a protein. By representing protein residues as nodes and the contacts between different residues as edges, contact maps visually display the compactness of the protein structure, distribution of functional areas, and the modes of interaction between residues. Distance maps, on the other hand, focus more on showing the spatial distances between atoms or residues within the protein. Representing these distances with lines or curves, distance maps provide a more intuitive view of protein structure, reflecting the spatial relationships between different parts. These graphical methods transform protein structural information into visual formats, making it easier to understand the spatial contacts among residues, thereby providing crucial insights for studying protein structure, function, and interactions.

To better understand how PEL-PVP captures the spatial structural information of proteins, we conducted a visualization experiment. We randomly selected two protein sequences and searched for their structures in the Protein Data Bank (PDB) database [49], downloading their structural files. Using the tool available at https://nanohub.org/resources/contactmaps [50], developed by Benjamin Rafferty and colleagues, we visualized their contact and distance maps. Subsequently, we visualized the contact maps using features extracted by PEL-PVP and compared them. This visualization not only facilitates a direct comparison between the original and the model-extracted features but also illustrates how effectively PEL-PVP captures and interprets the intricate spatial arrangements within protein structures, offering insights into its predictive capabilities and accuracy.

From the Fig. 4, the protein contact maps generated by our model and those produced by the nanohub tool demonstrates a significant similarity in overall morphology. This similarity indicates that our model is capable of accurately capturing the structural features of proteins, effectively extracting spatial information. Such capability is

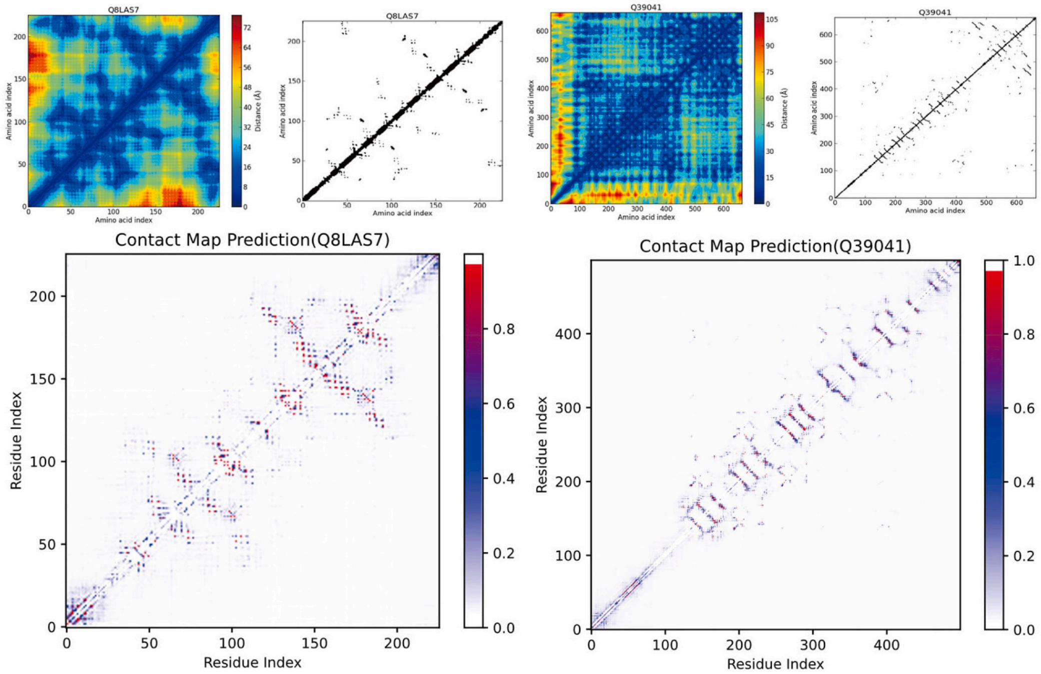

Fig. 4. Visualization of contact maps. Randomly select two plant vacuole protein sequences for visualization analysis. The upper part shows the distance map and contact map visualized using the protein precursor structure file (.pdb), while the lower part shows the visualization of the contact map based on the features extracted by PEL-PVP.

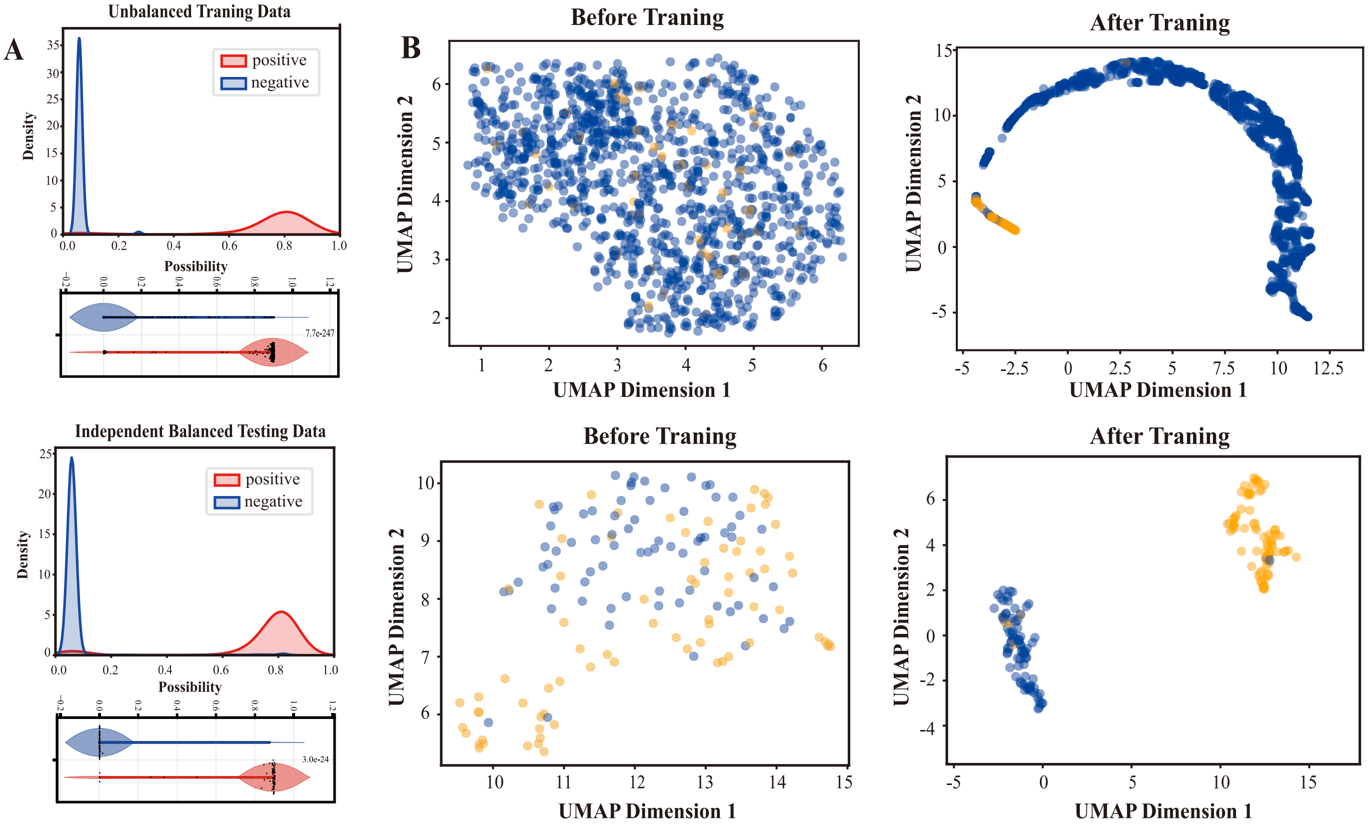

Fig. 5. A. Model prediction probability distribution. Prediction probability distribution and violin plot of the model on balanced and imbalanced datasets Whether in balanced or imbalanced datasets, the probability density of positive categories is almost entirely concentrated in the higher probability range, while the probability density of negative categories is concentrated at very low probability values and is very concentrated. B. UMAP feature extraction map Reduce the dimensionality and visualize the feature space before and after model training, with the upper half being the imbalanced training set and the lower half being the independent balanced testing set.

crucial for understanding the intricate details of protein structures, which are essential for insights into biological functions and interactions. This alignment not only confirms the reliability of our model but also showcases its potential in aiding advanced studies in protein dynamics and interactions.

3.7. Feature extraction maps characterize model effectiveness

UMAP (Uniform Manifold Approximation and Projection) [51] technology is utilized to evaluate a model's learning and generalization capabilities in dealing with uneven data distributions. UMAP is an efficient dimensionality reduction technique that maps data points from a high-dimensional feature space to a lower-dimensional space, enabling visualization of the data distribution. To assess the model's learning and generalization abilities under conditions of uneven data distribution, we employed UMAP to visualize the feature space before and after model training. The visualization results as shown in Fig. 5, reveal that prior to training, the data features are intermixed in the high-dimensional space with unclear boundaries between categories. After training, clear boundaries between categories emerge, both on the unbalanced training set and the balanced test set, indicating that the model has successfully learned key structures in the feature space to differentiate between categories. Additionally, the separation ability displayed by the model on the balanced test set, especially its performance on unseen data, highlights its excellent generalizability. This aspect is crucial for deep learning models because it signifies the model's capability to handle the commonly encountered imbalanced data distributions in the real world.

3.8. Model probability distribution proves model confidence

In addition, in order to gain a deeper understanding of the predictive behavior of the model, this paper presents the probability density distribution of positive and negative samples on the imbalanced training set and the independently balanced test set, as shown in Fig. 5. The figure illustrates the probability distributions of each class. It can be seen that the probability density of positive classes is almost entirely concentrated in the high probability range (0.6–1.0), while the probability density of negative classes is concentrated in very low probability values (0–0.1), and the peak density is much higher than that of positive classes. This denser peak indicates an enhanced ability of the model to recognize negative samples, largely due to the highly imbalanced nature of the training set. However, using the focus loss function helps to maintain the model's ability to recognize positive samples, as demonstrated by the concentration of positive class probability density in a higher probability range.

This observation indicates that the model is very effective in learning to distinguish between two categories during the training process. After training on imbalanced datasets, the model also demonstrated good generalization ability on independent balanced test sets. Similar to the training set, the probability density of negative classes is highly concentrated at very low probability values with very high peak densities. This indicates that the trained model maintains high confidence in class recognition on unknown datasets.

Moreover, the extreme probability distribution on the training set indicates that it can effectively learn distinguishing features from unbalanced data. These results demonstrate that the model has a strong learning ability when dealing with unbalanced data and can maintain high accuracy on a balanced test set, which is crucial for practical applications. Especially in the real world, where unbalanced data distributions are common, this capability highlights the model's strong adaptability and potential for application.

3.9. Comparison of our model (PEL-PVP) with previous models

In our comparisons with other models on unbalanced datasets, we faced limitations as VacPred has not been made open source and the GitHub repository for GraphIdn is no longer maintained. Consequently, on the unbalanced datasets, we primarily compared the performance of our proposed model, PEL-PVP, with iPVP-DRLF. In the context of unbalanced datasets, using accuracy as the sole metric may not provide a comprehensive understanding due to the influence of the ratio between positive and negative samples. Therefore, to thoroughly assess model performance throughout the experimental process on unbalanced datasets, we focused on precision, recall, AUC-ROC, AUPRC, and F1-scores. The results are presented in Table 2 as follows:

It is clear that PEL-PVP significantly outperforms iPVP-DRLF on unbalanced datasets across all metrics. This superior performance is largely attributed to the adoption of the Focal Loss function, tailored to address the challenges of unbalanced datasets. By introducing a focusing parameter that suppresses the loss for non-plant vacuolar protein samples, the model is steered to pay more attention to plant vacuolar protein samples. This mechanism enables PEL-PVP to concentrate more effectively on learning from plant vacuolar protein samples, thus enhancing its performance on minority classes and overall outcomes on unbalanced datasets.

PEL-PVP not only excels in unbalanced dataset scenarios but also performs exceptionally well on balanced datasets. In this study, we used the same independent balanced test set as previous studies to compare our proposed model, PEL-PVP, against four prior models: VacPred-DPC, VacPred-PSSM, iPVP-DRLF, and GraphIdn. The models were evaluated on this independent test set using a range of metrics, including Accuracy (Acc), Sensitivity (Sn), Specificity (Sp), Matthews Correlation Coefficient (MCC), and the Area Under the ROC Curve (AUC).

Our model, PEL-PVP, has demonstrated outstanding performance across multiple metrics, showing significant improvements compared to the current state-of-the-art models. Specifically, it achieved increases of 6.08 % in Accuracy, 13.51 % in Specificity, 11.9 % in Matthews Correlation Coefficient (MCC), and 5 % in Area Under the ROC Curve (AUC), with an accuracy rate reaching 94.59 %, markedly higher than the previous best model, GraphIdn, which had an accuracy of 88.51 % (Table 3). The MCC and AUC values, at 0.895 and 0.983 respectively, further highlight the robustness of PEL-PVP, emphasizing its high consistency between actual outcomes and predictions, as well as its capability to effectively differentiate between positive and negative classes (Fig. 6).

Compared with the most advanced models currently available, our model PEL-PVP has realized substantial improvements across all performance indicators, showcasing its overall superior performance. These achievements not only validate the theoretical innovation of PEL-PVP but also prove its powerful predictive ability and high analytical precision in practical applications. Especially notable are its performances in accuracy, specificity, and composite metrics such as the Matthews Correlation Coefficient and the Area Under the ROC Curve. These significant improvements have not only advanced the field of study but also provide robust support for future applications, highlighting PEL-PVP's exceptional robustness and predictive accuracy when dealing with complex datasets.

4. Web server

In order to advance research in the field of plant vacuole proteins, we have developed a network server application. Free access is available through the following link: http://www.bioai-lab.com/PEL-PVP. It receives protein sequence files, each sequence is evaluated to determine whether it is a plant vacuole protein, and is quickly displayed in a concise format. So as to assist researchers in identification. For comprehensive guidance on application usage, detailed information can be obtained in the help section of the website.

5. Conclusion

Plant vacuoles, as crucial multifunctional organelles within plant

Table 2
Comparison of our model (PEL-PVP) with iPVP-DRLF on the unbalanced dataset.

Model	Acc (%)	Precision (%)	recall (%)	F1 (%)	MCC	AUC	AUPRC
iPVP-DRLF	78.46	11.75	68.38	20.06	21.88	0.824	0.158
PEL-PVP	**97.53**	**70.93**	**70.11**	**70.52**	**69.23**	**0.845**	**0.605**

Bold values are the models that achieve the best performance.

Table 3
Comparison of our model (PEL-PVP) with previous models on the independent test set.

Model	Acc (%)	Sn (%)	Sp (%)	MCC	AUC
VacPred-DPC	80.41	82.43	78.38	0.610	0.840
VacPred-PSSM	86.49	90.54	82.43	0.730	0.930
iPVP-DRLF	87.16	89.19	85.14	0.744	0.916
GraphIdn	88.51	**94.59**	82.43	0.776	0.933
PEL-PVP	**94.59**	90.54	**98.65**	**0.895**	**0.983**

Bold values are the models that achieve the best performance.

cells, play a vital role in cellular processes. Traditional biological experimental methods are not only time-consuming but also expensive, and there has been limited exploration in the field of bioinformatics. Additionally, previous datasets on vacuolar proteins were balanced, whereas unbalanced datasets are closer to real-world conditions, reflecting the complexity of natural environments and experimental setups. Using unbalanced datasets can help models adapt better to the complexity and uncertainty of real-world environments. To more accurately identify vacuolar proteins, in this study, we constructed a new unbalanced dataset named UB-PVP and developed a novel plant vacuolar protein prediction model, PEL-PVP. This model leverages the Transformer architecture and self-attention mechanisms to calculate pairwise interactions between residues in sequences, capturing the interdependencies and interactions between amino acids at different positions to extract spatial information. Additionally, a dual-layer LSTM with unique memory units is utilized to handle long-range dependencies, capturing more complex latent features suitable for longer protein sequences. Building on the vast pretrained parameters of the ESM-2 model, it is adaptively fine-tuned for vacuolar proteins using LoRa low-rank adaptation technology, effectively reducing the number of parameters and computational complexity during fine-tuning. PEL-PVP outperforms existing PVP predictors on both balanced and unbalanced datasets. To facilitate further research, we have developed a user-friendly online web server for PEL-PVP, available for public use as a supplement to PVP recognition biological experiments. This demonstrates the feasibility and effectiveness of utilizing advanced pretrained models and fine-tuning techniques for bioinformatics tasks, offering new methods and insights for the study of plant vacuolar proteins.

Funding

The work is supported by the National Natural Science Foundation of China (Nos. 62101100, 62262015).

CRediT authorship contribution statement

Cuilin Xiao: Writing – original draft, Visualization, Validation, Software, Methodology, Conceptualization. **Zheyu Zhou:** Writing – original draft, Visualization, Validation. **Jiayi She:** Writing – original draft, Visualization. **Jinfen Yin:** Writing – original draft, Visualization. **Feifei Cui:** Writing – review & editing, Supervision, Project

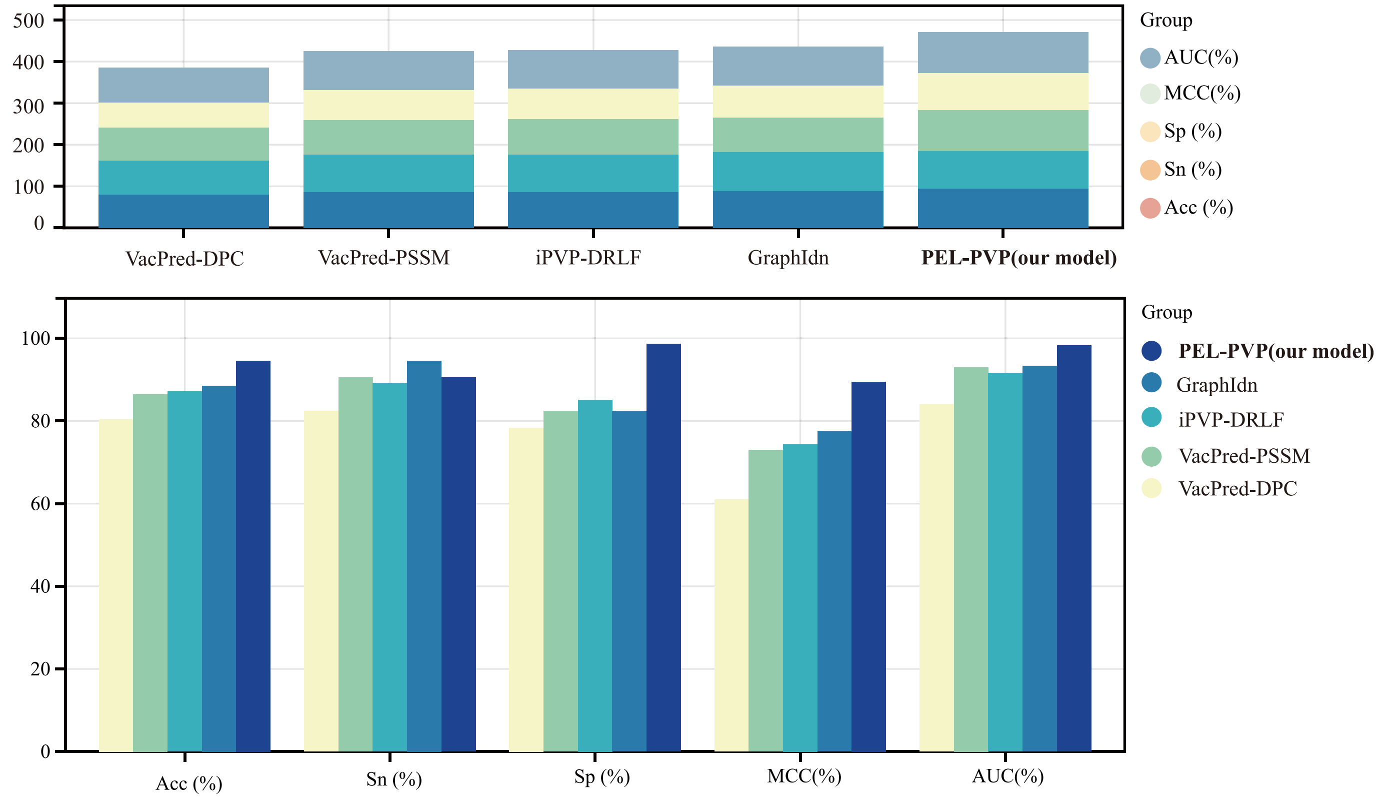

Fig. 6. Comparison of our model (PEL-PVP) with previous models on the independent test set

administration, Funding acquisition. **Zilong Zhang:** Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

We declare that we have no financial and personal relationships with other people animations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service that could be construed as influencing the position presented in, or the review of, the manuscript entitled.

Data availability

The source code for PEL-PVP, available for exploration and collaborative enhancement, can be freely accessed on https://github.com/Arthur200208/PEL-PVP.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.ijbiomac.2024.134317.

References

[1] C. Kolb, et al., FYVE1 is essential for vacuole biogenesis and intracellular trafficking in Arabidopsis, Plant Physiol. 167 (4) (2015) 1361–1373.

[2] C. Zhang, G.R. Hicks, N.V. Raikhel, Molecular composition of plant vacuoles: important but less understood regulations and roles of tonoplast lipids, Plants 4 (2) (2015) 320–333.

[3] S.-P. Shi, et al., Identify submitochondria and subchloroplast locations with pseudo amino acid composition: approach from the strategy of discrete wavelet transform feature extraction, Biochim. Biophys. Acta (BBA)-Molec. Cell Res. 1813 (3) (2011) 424–430.

[4] J. Zouhar, A. Muñoz, E. Rojo, Functional specialization within the vacuolar sorting receptor family: VSR1, VSR3 and VSR4 sort vacuolar storage cargo in seeds and vegetative tissues, Plant J. 64 (4) (2010) 577–588.

[5] Z.-Y. Wang, C. Gehring, J. Zhu, F.-M. Li, J.-K. Zhu, L. Xiong, The Arabidopsis vacuolar sorting receptor1 is required for osmotic stress-induced abscisic acid biosynthesis, Plant Physiol. 167 (1) (2015) 137–152.

[6] S. Segami, M. Asaoka, S. Kinoshita, M. Fukuda, Y. Nakanishi, M. Maeshima, Biochemical, structural and physiological characteristics of vacuolar H+-pyrophosphatase, Plant Cell Physiol. 59 (7) (2018) 1300–1308.

[7] J.-Y. Tsai, et al., Roles of the hydrophobic gate and exit channel in Vigna radiata pyrophosphatase ion translocation, J. Mol. Biol. 431 (8) (2019) 1619–1632.

[8] Y. Cui, Q. Zhao, S. Hu, L. Jiang, Vacuole biogenesis in plants: how many vacuoles, how many models? Trends Plant Sci. 25 (6) (2020) 538–548.

[9] J.J. Almagro Armenteros, C.K. Sønderby, S.K. Sønderby, H. Nielsen, O. Winther, DeepLoc: prediction of protein subcellular localization using deep learning, Bioinformatics 33 (21) (2017) 3387–3395.

[10] P. Horton, et al., WoLF PSORT: protein localization predictor, Nucleic Acids Res. 35 (suppl_2) (2007) W585–W587.

[11] J. Ahmad, M. Hayat, MFSC: multi-voting based feature selection for classification of Golgi proteins by adopting the general form of Chou's PseAAC components, J. Theor. Biol. 463 (2019) 99–109.

[12] H. Zhou, C. Chen, M. Wang, Q. Ma, B. Yu, Predicting golgi-resident protein types using conditional covariance minimization with XGBoost based on multiple features fusion, IEEE Access 7 (2019) 144154–144164.

[13] P. Du, Y. Li, Prediction of protein submitochondria locations by hybridizing pseudo-amino acid composition with various physicochemical features of segmented sequence, BMC Bioinform. 7 (2006) 1–8.

[14] H. Lin, W. Chen, L.-F. Yuan, Z.-Q. Li, H. Ding, Using over-represented tetrapeptides to predict protein submitochondria locations, Acta Biotheor. 61 (2013) 259–268.

[15] W. Qiu, et al., Predicting protein submitochondrial locations by incorporating the pseudo-position specific scoring matrix into the general Chou's pseudo-amino acid composition, J. Theor. Biol. 450 (2018) 86–103.

[16] C. Savojardo, N. Bruciaferri, G. Tartari, P.L. Martelli, R. Casadio, DeepMito: accurate prediction of protein sub-mitochondrial localization using convolutional neural networks, Bioinformatics 36 (1) (2020) 56–64.

[17] W. Zhao, G.-P. Li, J. Wang, Y.-K. Zhou, Y. Gao, P.-F. Du, Predicting protein sub-Golgi locations by combining functional domain enrichment scores with pseudo-amino acid compositions, J. Theor. Biol. 473 (2019) 38–43.

[18] H. Ding, et al., Prediction of Golgi-resident protein types by using feature selection technique, Chemom. Intell. Lab. Syst. 124 (2013) 9–13.

[19] S. Jiao, X. Ye, C. Ao, T. Sakurai, Q. Zou, L. Xu, Adaptive learning embedding features to improve the predictive performance of SARS-CoV-2 phosphorylation sites, Bioinformatics 39 (11) (2023) btad627.

[20] Z. Lv, S. Jin, H. Ding, Q. Zou, A random forest sub-Golgi protein classifier optimized via dipeptide and amino acid composition features, Front. Bioeng. Biotechnol. 7 (2019) 215.

[21] M. Anteghini, V. Martins dos Santos, E. Saccenti, In-pero: exploiting deep learning embeddings of protein sequences to predict the localisation of peroxisomal proteins, Int. J. Mol. Sci. 22 (12) (2021) 6409.

[22] A.K. Yadav, D. Singla, VacPred: sequence-based prediction of plant vacuole proteins using machine-learning techniques, J. Biosci. 45 (2020) 1–9.

[23] S. Jiao, Q. Zou, Identification of plant vacuole proteins by exploiting deep representation learning features, Comput. Struct. Biotechnol. J. 20 (2022) 2921–2927.

[24] J. Sui, J. Chen, Y. Chen, N. Iwamori, J. Sun, Identification of plant vacuole proteins by using graph neural network and contact maps, BMC Bioinform. 24 (1) (2023) 357.

[25] Z. Zhou, et al., PSAC-6mA: 6mA site identifier using self-attention capsule network based on sequence-positioning, Comput. Biol. Med. 171 (2024) 108129, https://doi.org/10.1016/j.compbiomed.2024.108129.

[26] X. Fu, et al., AGF-PPIS: a protein–protein interaction site predictor based on an attention mechanism and graph convolutional networks, Methods 222 (2024) 142–151, https://doi.org/10.1016/j.ymeth.2024.01.006.

[27] F. Cui, et al., DeepMC-iNABP: deep learning for multiclass identification and classification of nucleic acid-binding proteins, Comput. Struct. Biotechnol. J. 20 (2022) 2020–2028, https://doi.org/10.1016/j.csbj.2022.04.029.

[28] C. Ao, S. Jiao, Y. Wang, L. Yu, Q. Zou, Biological sequence classification: a review on data and general methods, Res. Rev. (2022) 0011, https://doi.org/10.34133/research.0011.

[29] M. Ertelt, J. Meiler, C.T. Schoeder, Combining Rosetta sequence design with protein language model predictions using Evolutionary Scale Modeling (ESM) as restraint, ACS Synth. Biol. 13 (4) (2024) 1085–1092.

[30] L. Xu, Deep learning for protein-protein contact prediction using Evolutionary Scale Modeling (ESM) feature, in: International Artificial Intelligence Conference, Springer, 2023, pp. 98–111.

[31] Z. Lin, et al., Evolutionary-scale prediction of atomic-level protein structure with a language model, Science 379 (6637) (2023) 1123–1130.

[32] Z.H. Kilimci, M. Yalcin, ACP-ESM: a novel framework for classification of anticancer peptides using protein-oriented transformer approach, in: arXiv preprint arXiv: 2401.02124, 2024.

[33] W. Han, et al., Predicting the antigenic evolution of SARS-COV-2 with deep learning, Nat. Commun. 14 (1) (2023) 3478.

[34] S. Pokharel, P. Pratyush, H.D. Ismail, J. Ma, D.B. Kc, Integrating embeddings from multiple protein language models to improve protein O-GlcNAc site prediction, Int. J. Mol. Sci. 24 (21) (2023) 16000.

[35] D. Joshi, S. Pradhan, R. Sajeed, R. Sriniva, S. Rana, An augmented transformer model trained on family specific variant data leads to improved prediction of variants of uncertain significance, in: ed: Research Square, 2023.

[36] D.J. Beal, ESM 2.0: state of the art and future potential of experience sampling methods in organizational research, Annu. Rev. Organ. Psych. Organ. Behav. 2 (1) (2015) 383–407.

[37] S. Sarrazin, et al., Endocan or endothelial cell specific molecule-1 (ESM-1): a potential novel endothelial cell marker and a new target for cancer therapy, Biochim. Biophys. Acta (BBA)-Rev. Cancer 1765 (1) (2006) 25–37.

[38] W. Yeung, Z. Zhou, L. Mathew, N. Gravel, R. Taujale, A. Venkat, W. Lanzilotta, S. Li, N. Kannan, An explainable unsupervised framework for alignment-free protein classification using sequence embeddings, bioRxiv (2022), 2022.02.08.478871.

[39] L. Fu, B. Niu, Z. Zhu, S. Wu, W. Li, CD-HIT: accelerated for clustering the next-generation sequencing data, Bioinformatics 28 (23) (2012) 3150–3152.

[40] L. Dou, Z. Zhang, L. Xu, Q. Zou, iKcr_CNN: a novel computational tool for imbalance classification of human nonhistone crotonylation sites based on convolutional neural networks with focal loss, Comput. Struct. Biotechnol. J. 20 (2022) 3268–3279.

[41] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2980–2988.

[42] R. He, et al., On the effectiveness of adapter-based tuning for pretrained language model adaptation, in: arXiv Preprint arXiv: 2106.03164, 2021.

[43] X.L. Li, P. Liang, Prefix-tuning: optimizing continuous prompts for generation, in: arXiv Preprint arXiv:2101.00190, 2021.

[44] E.J. Hu, et al., Lora: low-rank adaptation of large language models, in: arXiv Preprint arXiv:2106.09685, 2021.

[45] A. Vaswani *et al.*, "Attention Is All You Need," p. arXiv: 1706.03762 10.48550/arXiv.1706.03762.

[46] G. Bebis, M. Georgiopoulos, Feed-forward neural networks, IEEE Potent. 13 (4) (1994) 27–31.

[47] A. Graves, Long short-term memory, in: Supervised Sequence Labelling With Recurrent Neural Networks, 2012, pp. 37–45.

[48] Y. Wang, Y. Zhai, Y. Ding, Q. Zou, SBSM-Pro: support bio-sequence machine for proteins, in: arXiv Preprint arXiv:2308.10275, 2023.

[49] S.K. Burley, H.M. Berman, G.J. Kleywegt, J.L. Markley, H. Nakamura, S. Velankar, Protein Data Bank (PDB): the single global macromolecular structure archive, Protein Crystallogr. Methods Prot. (2017) 627–641.

[50] I.A. Emerson, A. Amala, Protein contact maps: a binary depiction of protein 3D structures, Phys. A Stat. Mech. Appl. 465 (2017) 782–791.

[51] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction," p. arXiv: 1802.03426doi: 10.48550S1802.03426.