

Enhancing Representation Learning for Biological Data Through Tailored Feature Extractors



Chunting Liu

Graduate School of Informatics

Kyoto University

Abstract

Computational methods have achieved significant success in addressing complex biological challenges. Leveraging advanced algorithms and high-dimensional data analysis, these methods have opened new avenues for understanding complex biological systems and transformed research approaches across genetics, molecular biology, and systems biology. A key challenge in bioinformatics and computational biology is designing effective strategies to capture and characterize implicit interactions within biological data. This requires developing sophisticated approaches to uncover underlying patterns and dependencies, enabling a deeper understanding of biological complexity and enhancing the predictive capabilities of computational models.

My research primarily focuses on integrating deep learning with biological prior knowledge to develop specialized representation models for two key types of biological data. For genetic sequence data, we propose a representation learning model to accurately identify DNA N4-methylcytosine (4mC) sites. For structured protein data, we develop a deep learning algorithm to predict the effect of mutations on protein binding affinity changes (i.e., $\Delta\Delta G$) with high precision. These methods enhance data-driven approaches and address existing technical bottlenecks in bioinformatics and computational biology.

We introduce a new approach for learning effective multi-scale representations aimed at accurately identifying 4mC sites. 4mC is an essential kind of epigenetic modification that regulates a wide range of biological processes. However, experimental methods for detecting 4mC sites are time-consuming and labor-intensive. As an alternative, data-driven computational methods that are capable of automatically identifying 4mC with data analysis techniques become a reasonable option. A major challenge is how to develop effective methods to fully exploit the complex interactions within the DNA sequences to improve the predictive capability.

Considering the characteristics of genetic sequences, we propose a multi-scale representation learning model named MSNet-4mC, which captures and models the complex correlations within the flanking bases. Specifically, MSNet-4mC is a

lightweight neural network designed with computational modules that integrate multiple convolutional kernels at different scales to capture both short-range and long-range cross-element relationships within DNA sequences. Furthermore, to address the imbalanced sample distribution across different species, we introduce class weights into the cross-entropy loss function to balance the training process. Extensive benchmarking experiments show that MSNet-4mC significantly improves predictive performance and outperforms other state-of-the-art methods.

We present a new approach involving the selective fusion of multi-way protein feature network to predict binding affinity changes upon mutations, i.e. $\Delta\Delta G$. Accurately predicting the effect of mutations on protein-protein interactions (PPIs) is essential for understanding protein structure and function, as well as providing insights into disease-causing mechanisms. Many recent popular approaches based on the three-dimensional structure of proteins have been proposed to predict binding affinity changes caused by mutations. However, how to effectively use the structural information to comprehensively exploit complex interactions within proteins and integrate multi-source features remains a significant challenge.

Focusing on the characteristics of structured protein data, we consider differences between mutant complex features and wild-type complex representations, as well as the effective and selective fusion of multi-source features. To this end, we propose a multi-level selective feature representation model, SFM-Net. Specifically, SFM-Net is constructed with GNN-based multi-way feature extractors and a new context-aware selective fusion module that jointly leverages the sequence, structural, and evolutionary information. This design enables SFM-Net to effectively and selectively use features from different sources to facilitate binding affinity change prediction. Benchmarking experiments and targeted ablation studies illustrate the effectiveness and robustness of our method for improving the binding affinity change prediction.

The methods proposed in this thesis effectively address the challenges of learning effective representations in the prediction of DNA N4-methylcytosine (4mC) sites and changes in protein binding affinity caused by mutations.

Acknowledgements

I would like to express my sincere gratitude to all those who have supported and guided me throughout my research, studies, and daily life, as this journey would not have been possible without their invaluable contributions.

First and foremost, I extend my deepest thanks to my supervisor, Professor Tatsuya Akutsu, for his constant support, insightful guidance, and unwavering patience. Akutsu-sensei gave me the invaluable opportunity to pursue my Ph.D. at Kyoto University, providing ample time and space to think and explore. His consistent encouragement helped me build the confidence needed to overcome all challenges, along with comprehensive support in various aspects of my studies. Akutsu-sensei's passion for academia and spirit of exploration will continue to inspire and benefit me throughout my life.

I would like to express my sincere gratitude to Professor Jiangning Song at Monash University and Professor Hiroyuki Ogata. Their expertise and valuable guidance have significantly enriched this research. Moreover, their mentorship has played a pivotal part in helping me grow and achieve my goals.

I am grateful to have the rest of my doctoral committees: Professor Yamamoto and Professor Kashima, for their thoughtful feedback and suggestions, which significantly improved the quality of this work. Their expertise and perspectives provided me with a broader understanding of the field.

I would like to express my great thanks to Associate Professor Takeyuki Tamura, Assistant Professor Tomoya Mori, Assistant Professor Motomu Matsui, and Chieko Funato san in our lab. They are always kind and patient. Their kind help supported me a lot in my studies and life in Kyoto. Chieko Funato san's enthusiasm and cheerful smile bring positive energy and brightness to the lab.

I would like to express my particular thanks to my lab colleagues, Liangjie Sun, Ruiming Li, Pengyu Liu, Feiqi Wang, Shogo Nakashima, Lixuan Mu, Yier Ma, Mamoonah Ghafoor, Ziwei Yang, Chenyao Wu, Takuma Iwaki, Jinxiang Kuang, Yinghan Zai, Russell Young Neches, and Tianfang Wei. Thank you for creating

a comfortable and enjoyable research environment. The encouragement and shared experiences have greatly helped me during my time in Kyoto.

I would like to express my particular thanks to my friends at Kyoto University, Yahui Fu, Hao Shi, and Haiyue Song. Thank you for filling my PhD journey with joyful memories and giving me strength when I needed it most. I would like to express my gratitude to my friends from afar, who have given me tremendous emotional support and brought joy to my life.

I would like to express my gratitude to the Student Affairs Division, the office of the Graduate School of Informatics for their patience and kind assistance. I am grateful to the Japan Science and Technology Agency for offering me the fellowship that provided invaluable support throughout my years in Kyoto.

I would like to express my heartfelt thanks to my family, especially my husband Sudong Cai, whose care has brought warmth and strength to my daily life. Throughout this journey, he has been both a reliable companion and an intellectual partner, always discussing ideas, exploring challenges, and growing together with me. His unwavering support has given me the confidence to pursue the path I truly aspire to follow. I would also like to sincerely thank my parents for their support and understanding.

Thank you to everyone who has contributed to this journey in countless ways.

Contents

List of Figures	ix
List of Tables	xiii
1 Introduction	1
1.1 Background	1
1.2 Challenges	2
1.3 Contributions	3
1.4 Thesis Outline	4
2 Preliminaries	7
2.1 Biological Basis	7
2.1.1 DNA N4-Methylcytosine (4mC)	7
2.1.2 Protein-Protein Binding Affinity Changes	8
2.2 Methodological Foundations	10
2.2.1 Convolutional Neural Networks	11
2.2.2 Graph Neural Networks	12
2.2.3 Self-Attention Mechanism	14
3 MSNet-4mC: Learning Effective Multi-Scale Representations for Identifying DNA N4-Methylcytosine Sites	17
3.1 Background	17
3.2 Materials and Methods	20
3.2.1 Datasets	20
3.2.2 MSNet-4mC Architecture	21
Input feature matrix	22
Convolutional neural network	22
Class weights for loss	24
Parameters	26
3.2.3 Performance Evaluation Metrics	28

3.3	Results	29
3.3.1	Performance Evaluation on the Lin_2017 Dataset	29
	Ablation study	29
	Performance comparison with the existing methods on the Lin_2017 dataset	31
3.3.2	Performance Evaluation on the Li_2020 Dataset	34
3.4	Discussions	37
4	SFM-Net: Selective Fusion of Multi-Way Protein Feature Network for Predicting Binding Affinity Changes upon Mutations	41
4.1	Background	41
4.2	Materials	45
4.2.1	Datasets	45
4.2.2	Protein Representation	45
	Node features	46
	Adjacency matrix	46
4.3	Methods	47
4.3.1	SFM-Net Architecture	47
4.3.2	Graph Neural Network-Based Feature Encoder	48
4.3.3	Summation and Difference Components	49
4.3.4	CSF: Context-Aware Selective Fusion Module	50
	Pooling operations	50
	Self-attention-inspired dual-scope feature fusion	51
4.3.5	Evaluation Metrics	53
4.4	Results	53
4.4.1	Performance Evaluation on Cross-Validation Test	53
	Mutation-level cross-validation	53
	Protein-level cross-validation	60
	Threshold sensitivity analysis	62
4.4.2	Ablation Analysis	63
	Ablation study on the proposed modules	64
	Effectiveness of the features	65
4.4.3	Case Study	67
4.5	Discussions	67

5	Conclusion and Future Directions	71
5.1	Conclusion	71
5.2	Future Directions	73
	References	75

List of Figures

1.1	Schematic illustration of the research objective	3
2.1	DNA methylation catalyzed by DNA methyltransferases (DNMTs).	8
2.2	Illustration of the protein-protein binding affinity change.	10
2.3	Convolutional neural network. (a) A typical CNN architecture. (b) The 1D convolution operation with the kernel and padding applied to the input sequence. Here, L denotes the sequence length, C denotes the number of channels (i.e., feature dimensions), and K is the kernel size. A kernel of shape $K \times C$ slides along the input to compute local feature representations.	11
2.4	Message passing in a graph neural network. Given node features V_n and edge features E_n at layer n , the network uses learnable functions f_V and f_E to update the node representations and, if applicable, the edge representations, yielding updated features V_{n+1} and E_{n+1} at layer $n + 1$	13
2.5	Self-attention mechanism. Given an input feature map $\mathbf{X} \in \mathbb{R}^{L \times C}$, the Query ($\mathbf{Q}$), Key ($\mathbf{K}$), and Value ($\mathbf{V}$) matrices are typically obtained via linear projections of $\mathbf{X}$. Here, L denotes the sequence length, C is the number of channels, d is the feature dimension of each attention head, and $\mathbf{A}$ represents the attended output feature map.	15
3.1	An overview of the architecture of MSNet-4mC. The model encodes multi-scale features via a series of dilation convolutions, where convolutions with receptive fields of 3, 5, 9 correspond to dilations of 1, 2, 4, respectively. The class weight is applied with its species sign to the CE-loss, where the species sign serves as an index to enable the correct weight assignment.	23
3.2	Species-specific performance of MSNet-4mC on the Lin_2017 dataset.	31

3.3	t-SNE visualization of the <i>C.elegans</i> dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.	34
3.4	t-SNE visualization of the <i>D.melanogaster</i> dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.	34
3.5	t-SNE visualization of the <i>A.thaliana</i> dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.	35
3.6	t-SNE visualization of the <i>E.coli</i> dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.	35
3.7	t-SNE visualization of the <i>G.subterraneus</i> dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.	35
3.8	t-SNE visualization of the <i>G.pickeringii</i> dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.	36
3.9	Species-specific performance of MSNet-4mC on the Li_2020 dataset.	37
3.10	Performance comparison between MSNet-4mC and other state-of-the-art methods on the Li_2020 dataset.	38
4.1	Overall framework of SFM-Net. (A) The architecture of SFM-Net, comprising (1) input of the feature encoding and the graph construction; (2) four parallel GNN-based blocks; (3) summation and difference components; (4) key ingredients that compose the CSF model, i.e., pooling operations and self-attention-inspired double-scope selective fusion module; and (5) multi-layer perceptron. The key ingredients of CSF encompass: (B) Three different kinds of pooling operations, i.e., center pool, mean pool, and relevant-site pool; (C) The intra-sample fusion processing, $\{\bar{x}_i i = 1, 2, \dots, 9\}$ refers to the nine pooled feature vectors, while the background feature $\bar{x}_b$ is calculated by $\frac{1}{6} \sum_{i=1}^6 \bar{x}_i$	48
4.2	SFM-Net predicted $\Delta\Delta G$ against experimental $\Delta\Delta G$ on three datasets. Color represents point density, while the dashed line indicates the regression line.	55

4.3	SFM-Net predicted $\Delta\Delta G$ against experimental $\Delta\Delta G$ across three distinct ranges on three datasets. Blue contours represent values where Experimental $\Delta\Delta G \leq -5$, orange contours represent values in the range $-5 < \text{Experimental } \Delta\Delta G < 5$, and green contours represent values where Experimental $\Delta\Delta G \geq 5$	56
4.4	Distribution of experimental $\Delta\Delta G$ on three datasets.	56
4.5	Distribution of SFM-Net predicted $\Delta\Delta G$ on three datasets.	57
4.6	Distribution of experimental $\Delta\Delta G$ values across the training and test sets for each fold in the 10-fold cross-validation experiment on S4169.	58
4.7	Distribution of experimental $\Delta\Delta G$ values across the training and test sets for each fold in the 10-fold cross-validation experiment on S1131.	59
4.8	Distribution of experimental $\Delta\Delta G$ values across the training and test sets for each fold in the 10-fold cross-validation experiment on S645.	60
4.9	Comparison of experimental and predicted $\Delta\Delta G$ distributions for interface and non-interface mutations on three datasets.	61
4.10	Performance comparison of SFM-Net and other competing methods in terms of Pearson’s correlation coefficient under the WT protein-protein complex-based cross-validations.	61
4.11	Performance comparison of SFM-Net and other involved methods on M1101 dataset in terms of PCC, RMSE, MAE under the WT protein-protein complex-based cross-validations.	63
4.12	Performance comparison of SFM-Net on the S1131 dataset in terms of Pearson’s correlation coefficient across different thresholds for defining neighboring residues.	63
4.13	Performance comparison of different variant models of SFM-Net across three benchmark datasets. Each box plot illustrates the distribution of 10-fold results for each model (SFM-Net, SDC, CSF, and w/o SDC&CSF) within a dataset. The three subfigures represent different evaluation metrics: (a) Pearson’s correlation coefficient (PCC), (b) root mean squared error (RMSE), and (c) mean absolute error (MAE).	64

4.14	Ablation study on feature importance across three benchmark datasets. The three subfigures correspond to different evaluation metrics: (a) Pearson’s correlation coefficient (PCC), (b) root mean squared error (RMSE), and (c) mean absolute error (MAE).	67
4.15	Distribution of predicted $\Delta\Delta G$ against experimental $\Delta\Delta G$. (a) Scatter plot showing the correlation between the predicted $\Delta\Delta G$ values and the experimental $\Delta\Delta G$ values for SARS-CoV-2 RBD mutations. Each point represents a single mutation, colored according to its mutation site. (b) Box plot comparing predicted and experimental $\Delta\Delta G$ values across different mutation sites. The x-axis represents mutation sites, and the colored circles below the axis correspond to the mutation categories as shown in (a). The blue boxes represent experimental values, while the orange boxes represent predicted values. This plot highlights the distribution of $\Delta\Delta G$ values for each mutation site.	69

List of Tables

3.1	Key characteristics of the existing methods for 4mC site prediction .	19
3.2	Statistical summary of the Lin_2017 dataset	20
3.3	Statistical summary of the Li_2020 dataset	21
3.4	Hyperparameters for the CNN structure	26
3.5	Hyperparameters for the training process	28
3.6	Ablation study of the class weight module on the Lin_2017 dataset .	30
3.7	Performance results on per-class mean and variance of the metrics .	30
3.8	Ablation study of the convolutions with multi-scale receptive fields on the Lin_2017 dataset	30
3.9	Training the species-specific models from scratch: Performance com- parison of MSNet-4mC and DeepTorrent on the Lin_2017 dataset .	32
3.10	After Fine-tuning: Performance of MSNet-4mC for species-specific 4mC site prediction on the independent test dataset of Lin_2017 dataset	33
3.11	After Fine-tuning: Performance comparison of MSNet-4mC and the existing methods on the independent test dataset of Lin_2017 dataset	33
3.12	Training the species-specific models from scratch: Performance of MSNet-4mC on the Li_2020 dataset	36
3.13	After Fine-tuning: Performance comparison of MSNet-4mC and the existing methods on the independent test dataset of Li_2020 dataset	39
4.1	Statistical summary of the datasets used in this study	45
4.2	Performance comparison of SFM-Net and the existing methods on three benchmark datasets in terms of Pearson's correlation coeffi- cient	55
4.3	Performance comparison of our model on interface and non-interface mutations in the S4169 and S645 datasets	57

4.4	The RMSE and MAE results for SFM-Net and other involved methods under the WT protein-protein complex-based cross-validations .	62
4.5	Performance comparison of SFM-Net on the S4169 dataset in terms of Pearson’s correlation coefficient across different thresholds for defining neighboring residues	63
4.6	Performance comparison of different variant models of SFM-Net on three benchmark datasets in terms of Pearson’s correlation coefficient	64
4.7	Ablation study on the proposed modules of SFM-Net with Pearson’s correlation coefficient and other evaluation metrics on three datasets	66
4.8	The ablation study on feature importance with Pearson’s correlation coefficient and other evaluation metrics on three datasets	68
4.9	Performance comparison on SARS-CoV-2 RBD in terms of Pearson’s correlation coefficient	69

Chapter 1

Introduction

1.1 Background

Biology advances our understanding of the fundamental processes and mechanisms of life, providing critical insights into health, disease, and environmental impacts. Traditionally, experimental and statistical methods have served as the primary tools in biological research. While these approaches are valuable, they face significant limitations, including high costs, time constraints, and the challenges of scaling experiments. Additionally, they tend to be labor-intensive and may not fully capture the complexity of biological systems.

In recent years, rapid developments in machine learning and deep learning techniques have transformed various fields, offering powerful tools to address complex problems with improved accuracy and efficiency. Computational methods have emerged as powerful solutions to numerous biological challenges. The exponential growth of biological data, such as genomic sequences, protein structures, and clinical records, has created abundant information that computational models can leverage. The integration of machine learning techniques with extensive biological data resources offers the opportunity to solve biological problems with enhanced precision, speed, and scalability. The computational approaches also reveal patterns and relationships that may be difficult to detect through traditional methods,

driving innovation and discovery in fields such as genomics, drug development, and disease modeling.

In bioinformatics, research efforts often cover multiple molecular layers, among which genomic and proteomic analyses represent two fundamental and widely studied domains. Genomic-level studies investigate DNA-based information such as sequence variations and epigenetic modifications, which are critical for gene regulation. Proteomic analyses, on the other hand, focus on the structure and function of proteins, including how mutations influence molecular interactions. A central issue of deep learning in computational biology lies not simply in leveraging complex models, but in making them adaptive to the specific characteristics of biological data.

1.2 Challenges

Although numerous computational methods have proven effective, a key challenge remains: how to fully extract effective and meaningful information from biological data to further improve model performance. The distinct properties of biological data, such as intricate relationships, structural information, and multi-source features, present both challenges and opportunities for model optimization. Developing approaches tailored to these specific characteristics enables models to capture more relevant biological insights, thereby improving accuracy, robustness, and generalizability.

Across both genomic-level and protein-level analyses, the primary aim of this thesis is to develop specialized models tailored to the properties of biological data. These models are designed to enable effective information fusion and feature selection, providing robust solutions to significant challenges in bioinformatics. Figure 1.1 illustrates the main objective of this thesis: leveraging neural networks to model key challenges in bioinformatics. For genomic data analysis, we focus on a representative task: predicting DNA N4-methylcytosine (4mC) modification sites. For proteomic data analysis, we concentrate on a biologically and computationally challenging task: predicting changes in protein binding affinity caused by mutations ($\Delta\Delta G$).

The data for the DNA 4mC site prediction task mainly consists of DNA sequence information. A variety of computational approaches have already been proposed for predicting DNA 4mC sites, and many of them have demonstrated strong predictive performance. These methods generally employ sophisticated encoding

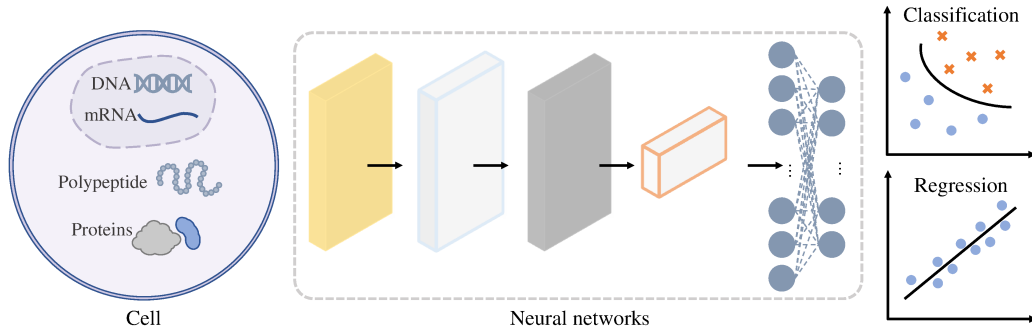


Figure 1.1: Schematic illustration of the research objective

schemes or stack various computational strategies to capture relevant information within the sequences. However, to further improve predictive efficiency and accuracy, the greatest challenge lies in designing lightweight models that can effectively extract inherent and latent relationships within DNA sequences.

The data for the task of PPI $\Delta\Delta G$ prediction consists of protein data, including both sequence and structural information. Additionally, evolutionary information derived from sequences provides crucial insights for protein studies by revealing conservation and variation trends throughout evolutionary processes. Many methods for predicting protein $\Delta\Delta G$ rely on traditional machine learning approaches as decoders to select features from diverse sources. These decoders typically rely on linear or tree-based decision rules. While these methods can be effective, they often struggle to capture deeper and more complex patterns within the data. The key challenge in improving predictive performance lies in the efficient fusion and selection of multi-source features.

1.3 Contributions

This thesis introduces two original studies in bioinformatics based on deep learning methods. The design policy of the proposed methods is driven by the characteristics of biological data. Specifically, the network architectures are tailored to facilitate effective feature extraction and efficient information interaction across different levels or sources of data. The resulting unified and informative representation serves as the foundation for final classification or regression tasks. The main contributions of each study are summarized below.

In the first study, we develop MSNet-4mC, a lightweight neural network based

on convolutional operations to learn effective multi-scale representations for identifying DNA 4mC sites. Inspired by the challenge of uncertain contextual dependency ranges among different sites in biological sequences, we design a built-in scale-aware learning structure with multi-scale receptive fields in convolutional operations to capture both long- and short-range dependencies within DNA sequences. To address the issue of sample imbalance, we incorporate a class-weight factor into the cross-entropy loss function to balance the training process. We evaluate MSNet-4mC through extensive experiments and ablation studies on benchmark datasets. The results consistently show that our method outperforms others, even without complex encoding representations as input, demonstrating its effectiveness in extracting crucial features for 4mC site identification.

In the second study, we propose SFM-Net, a powerful deep learning model constructed with GNN-based multi-way feature extractors and a new context-aware selective fusion module for predicting binding affinity changes upon mutations. We introduce a simple yet effective graph construction and employ GNN blocks to encode refined neural features from multi-source input features. To harmonize and integrate features from different sources, we design a novel self-attention-inspired, context-driven selective feature fusion module that captures both intra-sample and inter-sample contextual dependencies. We also devise the summation and difference components to learn the background of common properties and high-frequency cues. SFM-Net is evaluated on benchmark datasets, and the results demonstrate that it outperforms other methods in comprehensive evaluations by effectively extracting key features relevant to PPI $\Delta\Delta G$ prediction.

1.4 Thesis Outline

In Chapter 2, we provide a brief introduction to the biological background and methodological foundations of this thesis. This includes an overview of the biological concepts relevant to DNA N4-methylcytosine (4mC) and PPI binding affinity changes, as well as the machine learning techniques that underpin our proposed methods.

In Chapter 3, we present the research on identifying DNA N4-methylcytosine (4mC) sites and introduce MSNet-4mC, a method that leverages convolutional operations with multi-scale receptive fields. Comprehensive benchmarking experiments and ablation studies demonstrate the effectiveness and robustness of MSNet-4mC.

In Chapter 4, we present the research on predicting binding affinity changes caused by mutations and introduce SFM-Net, a powerful deep learning model. SFM-Net employs GNN-based multi-way feature extractors and incorporates a novel context-aware selective fusion module, effectively integrating sequence, structural, and evolutionary information to enhance predictive performance. Experimental results demonstrate its robustness and effectiveness, highlighting its ability to selectively utilize features from different sources.

In Chapter 5, we conclude the above studies and discuss potential directions for future research.

Chapter 2

Preliminaries

In this Chapter, we provide a brief overview of the preliminary knowledge and concepts relevant to this thesis, organized into two sections: (a) the biological basis and (b) methodological foundations.

2.1 Biological Basis

In this section, we introduce the biological background relevant to the problems addressed in this thesis, including DNA N4-methylcytosine (4mC) and protein-protein binding affinity changes.

2.1.1 DNA N4-Methylcytosine (4mC)

Methylation is a key mechanism in epigenetics, influencing gene expression, regulation, and cell differentiation without altering the DNA sequence itself [1]. Beyond regulating gene activity, methylation also plays a crucial role in maintaining genome stability, guiding cell differentiation, and supporting proper development [2, 3]. DNA methylation involves adding a methyl group ($-\text{CH}_3$) to specific sites within the DNA molecule to regulate DNA segment activity. It mainly includes three types: 5-methylcytosine (5mC), N6-methyladenine (6mA), and N4-methylcytosine (4mC) [4]. 4mC is formed by methylation of the amino group at the N4 position

of cytosine, catalyzed by specific DNA methyltransferases known as N4-cytosine methyltransferases. Figure 2.1 illustrates the DNA methylation process catalyzed by DNA methyltransferases (DNMTs). Similar to 5mC and 6mA, 4mC is involved in the restriction-modification (RM) systems, which help to protect the host DNA against degradation by restriction enzymes [5, 6]. 4mC methylation is most commonly observed in bacteria and archaea [7]. Despite being discovered decades ago, 4mC methylation has been less extensively studied than 5mC and 6mA. Investigating DNA N4-methylcytosine may provide new insights into gene regulation, genome stability, epigenetic mechanisms, and disease pathways, shedding light on previously unexplored dimensions of molecular biology.

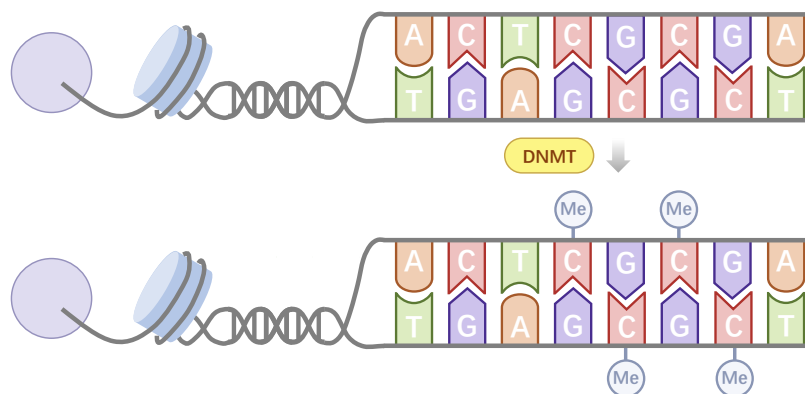


Figure 2.1: DNA methylation catalyzed by DNA methyltransferases (DNMTs).

2.1.2 Protein-Protein Binding Affinity Changes

Proteins are biological macromolecules consisting of amino acids linked by peptide bonds into linear polypeptide chains [8]. These chains fold into secondary structures such as α -helices and β -sheets, which are stabilized by hydrogen bonds between backbone atoms. Further folding produces a unique three-dimensional conformation known as the tertiary structure [9]. The structure of a protein determines its specific function and stability. Proteins play indispensable roles in nearly all cellular processes, and virtually all life activities in organisms depend on their structural and functional diversity [10]. In living organisms, proteins rarely function in isolation. Instead, they interact through protein-protein interactions (PPIs) to carry out complex biological processes such as signal transduction, enzymatic catalysis, and gene expression regulation [11, 12, 13, 14].

Protein-protein interactions (PPIs) refer to the process by which two or more protein molecules interact through non-covalent associations at their binding interfaces. Many cellular functions rely on precise interactions between specific proteins, which are crucial for maintaining normal physiological activities in organisms [15]. Most PPIs occur at the binding interfaces of proteins, where surface binding sites are stabilized by a combination of various physicochemical forces, such as hydrogen bonds, hydrophobic interactions, van der Waals forces, and ionic bonds [16, 17].

In PPIs, binding affinity refers to the strength of the interaction between two protein molecules [18]. Higher binding affinity indicates a stronger interaction, corresponding to a more stable protein complex. Conversely, lower affinity reflects weaker binding and a greater tendency for dissociation. Binding affinity is typically measured and reported by thermodynamic parameters, such as the equilibrium dissociation constant (K_D) or binding free energy (ΔG) [19].

$$K_D = 1/K_A = k_{\text{off}}/k_{\text{on}}, \quad (2.1)$$

$$\Delta G = RT \ln(K_D), \quad (2.2)$$

where k_{off} is the dissociation rate, which quantifies how rapidly a bound complex dissociates into its individual components; k_{on} is the association rate, which indicates the rate at which two molecules associate to form a complex; R is the ideal gas constant; and T is the absolute temperature at which the experiment is conducted. The dissociation constant (K_D) indicates the binding equilibrium. Smaller K_D values indicate stronger binding affinity of the partners and more stable binding. Binding free energy (ΔG) reflects the spontaneity of the interaction, with more negative values indicating higher binding affinity [20].

The binding affinity of PPIs reflects the interaction dynamics of proteins within the cell and influences signaling pathways and reaction rates, thereby affecting cellular physiological responses [21]. Investigating binding affinity is essential for understanding PPIs and intracellular signal transduction. Even minor changes in binding affinity can have profound impacts on biological functions.

In a protein sequence, amino acid mutations can alter the local or global structure of the protein, potentially affecting its stability and, consequently, its function [22]. Particularly, mutations occurring at the protein binding interface can directly impact the interaction between the protein and other molecules (such as

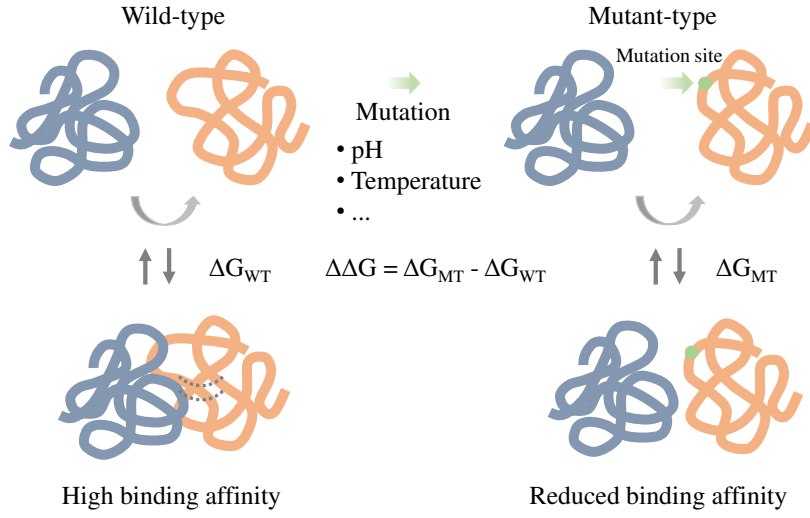


Figure 2.2: Illustration of the protein-protein binding affinity change.

another protein), thereby altering the binding affinity. Figure 2.2 shows the effect of a mutation on protein-protein binding affinity. The change in binding affinity caused by a mutation is typically quantified as the difference in binding free energy ($\Delta\Delta G$), calculated by subtracting the ΔG of the wild-type (WT) from that of the mutant-type (MT) [19]:

$$\Delta\Delta G = \Delta G_{MT} - \Delta G_{WT}. \quad (2.3)$$

Specifically, some mutations may weaken the original interactions, reducing binding affinity and destabilizing the protein-protein interaction, whereas others may strengthen interactions, leading to increased affinity and enhanced complex stability. Changes in binding affinity caused by mutations can significantly impact cellular function. Therefore, predicting these changes is crucial for understanding the role of mutations in disease, providing valuable insights into disease mechanisms and therapeutic drug design.

2.2 Methodological Foundations

In this section, we introduce the theoretical foundations involved in the proposed methods, including convolutional neural networks, graph neural networks, and the self-attention mechanism.

2.2.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are among the most foundational architectures in deep learning, widely used for processing various types of data such as images, speech, and video. More recently, their applications have extended significantly into the field of bioinformatics. The concept of CNNs is inspired by the biological visual system. Studies have shown that neurons in the animal visual cortex exhibit selective responses to specific regions of visual input [23]. These neurons typically possess localized receptive fields, enabling them to focus on particular areas of an image rather than the entire visual scene. This suggests that visual processing in animals is both highly localized and inherently hierarchical. Building on this critical insight, researchers began to investigate the mechanisms of localized perception in the visual domain, ultimately paving the way for the development of convolutional neural networks.

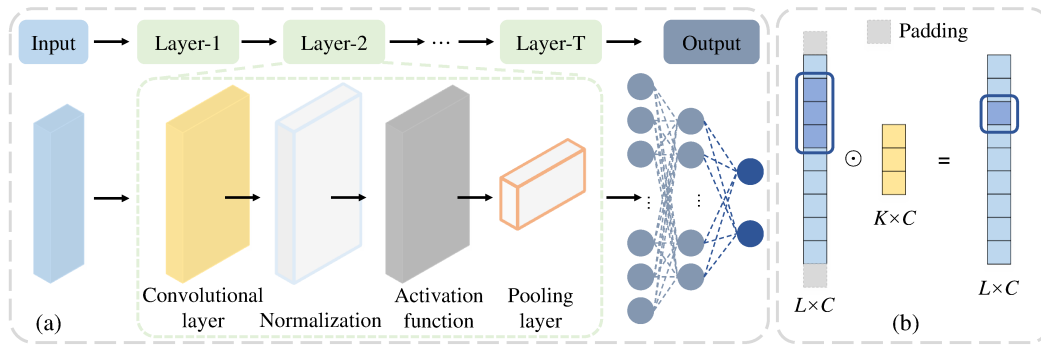


Figure 2.3: Convolutional neural network. (a) A typical CNN architecture. (b) The 1D convolution operation with the kernel and padding applied to the input sequence. Here, L denotes the sequence length, C denotes the number of channels (i.e., feature dimensions), and K is the kernel size. A kernel of shape $K \times C$ slides along the input to compute local feature representations.

A typical convolutional neural network includes a series of sequential building blocks (layers), each of which consists of basic neural operations: e.g., convolutional layers/linear layers, normalization, activation functions, and pooling layers, as shown in Figure 2.3. At the core of this architecture lies the convolutional layer, which forms the foundation of CNNs. It employs a set of filters (also known as convolutional kernels) to perform convolution operations on the input, effectively capturing local features from the data. Each neuron in the convolutional layer is connected to a specific, localized region of neurons in the previous layer [24]. The size of this region, referred to as the receptive field, is determined by the dimensions of the convolutional kernel. In other words, the receptive field describes the specific

region of the input data that affects the output of a particular neuron in the network.

CNNs demonstrate exceptional performance in fields such as image processing and speech recognition [25], largely attributed to the following key advantages: convolutional operations ensure the model effectively extracts localized features from the input data; downsampling or pooling operations reduce the size of feature maps while preserving key features, thereby optimizing computational efficiency and enhancing the model's robustness to data transformations; as the network depth increases, the model becomes capable of extracting more complex high-level (global) features from low-level (local) ones; when training a CNN, algorithms such as back-propagation and optimizers (e.g., SGD, Adam) are typically employed to adjust the network's parameters in order to minimize the loss and enhance model performance.

ResNet (Residual Network) is a landmark convolutional neural network architecture in deep learning, proposed by Kaiming He et al. in their 2015 paper *Deep Residual Learning for Image Recognition* [26]. The key innovation of ResNet lies in the use of residual connections, which effectively address challenges in training deep neural networks, such as vanishing gradients and network degradation, thereby enabling the development of much deeper networks with improved performance. Specifically, ResNet introduces a shortcut connection in each residual block that directly adds the input signal to the convolution output. This structure helps retain information from shallow layers in deep networks and mitigates the vanishing gradient problem. ResNet defines networks of varying depths based on the number of layers, such as ResNet-18, ResNet-34, ResNet-50, ResNet-101, and ResNet-152, to suit datasets of different complexities. These architectures are widely employed in computer vision tasks. For the DNA 4mC site prediction task, a lightweight neural network architecture inspired by ResNet and employing 1D convolutions has been developed and optimized, taking into account the dimensionality and complexity of DNA sequence data.

2.2.2 Graph Neural Networks

While CNNs have achieved remarkable success in various tasks, they often struggle to process topological information in graph-structured data. Graph Neural Networks (GNNs) are specialized deep learning models designed to address this challenge by effectively handling graph-structured data [27]. In real-world scenarios, many types of data, such as social networks, transportation networks, and biological structures, can be naturally represented as graphs. Traditional neural networks

are designed to process regular grid-like data (e.g., vectors or matrices), making it challenging to handle non-Euclidean structures directly. GNNs leverage the inherent structural properties of graphs to learn representations directly from the data, thereby eliminating the need for complex handcrafted feature design for topological information as required by traditional methods [28].

In GNNs, graph-structured data is typically represented using nodes and edges. Nodes correspond to entities, such as users in social networks or amino acids in protein structures. Edges represent relationships between entities, such as friendships in social networks or peptide bonds between amino acids in biological molecules.

Before training GNNs, graph-structured data is typically transformed into matrix or vector representations as input for the neural network. In a graph with N nodes, each node is represented by a D -dimensional feature vector describing its attributes, forming an $N \times D$ dimensional feature matrix. Additionally, an adjacency matrix is commonly used to represent the connections between nodes. For a graph with N nodes, the adjacency matrix A is $N \times N$ dimensional, where $A(i, j)$ indicates whether there is an edge between node i and node j . By taking both the feature matrix and the adjacency matrix as input, GNNs can effectively capture both node features and the structural information of the graph.

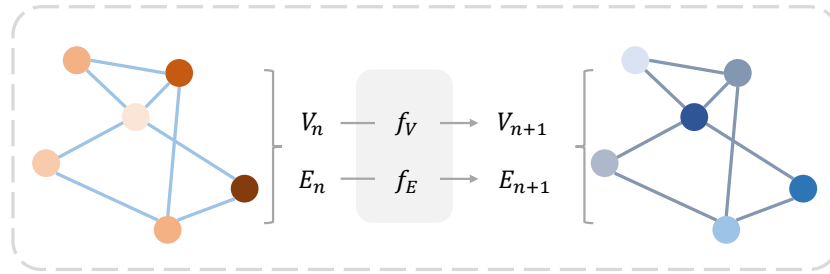


Figure 2.4: Message passing in a graph neural network. Given node features V_n and edge features E_n at layer n , the network uses learnable functions f_V and f_E to update the node representations and, if applicable, the edge representations, yielding updated features V_{n+1} and E_{n+1} at layer $n + 1$.

The core idea of GNNs is the message passing mechanism, as shown in Figure 2.4. In this mechanism, nodes update their features by aggregating information from their neighbors and combining the aggregated information with their own features to generate a new representation [29, 30]. Through multiple iterations, this process allows information to propagate to more distant nodes, enabling the extraction of increasingly informative feature representations.

Common GNN models include Graph Convolutional Network (GCN) [31], GraphSAGE [32], and Graph Attention Network (GAT) [33]. Another representative

model is the Graph Isomorphism Network (GIN), proposed by Keyulu Xu et al. in their 2018 paper *How Powerful are Graph Neural Networks?* [34]. The core idea of GIN is to introduce a learnable or fixed parameter to adjust the node's own features during the aggregation process. The adjusted features are then combined with the aggregated features from neighboring nodes, effectively balancing the contributions of the node's own features and those of its neighbors. This balance enables GIN to achieve strong discriminative power, allowing it to distinguish graphs with different structures or features. The aggregation and update mechanism in GIN significantly improves its effectiveness in addressing both node-level and graph-level learning problems.

In the task of predicting binding affinity changes caused by mutations, a graph is constructed for each mutation, where nodes and edges represent amino acids and structural features, respectively. The goal is to predict binding affinity changes based on these graphs. Given the need to capture subtle structural and feature differences between graphs, we adopt GIN as the base feature encoder due to its strong discriminative power and flexibility in aggregating node and neighborhood features.

2.2.3 Self-Attention Mechanism

The attention mechanism is a foundational concept in deep learning, inspired by the studies of human visual and cognitive systems [35]. These systems exhibit the ability to selectively focus on specific parts of global information while disregarding less relevant details. As a resource allocation strategy, the attention mechanism enables models to dynamically concentrate on the most relevant parts of the input during prediction, effectively suppressing irrelevant information. This mechanism is particularly well-suited for tasks involving long-range dependencies, complex hierarchical structures, or sequences with variable contextual relevance.

In its early application to sequence-to-sequence models for machine translation, the attention mechanism calculates attention scores by measuring the relevance (e.g., using dot products or other similarity metrics methods) between a query and all keys [36]. These scores represent the relative importance of each element (e.g., words in a sentence) with respect to the given query. After normalization, the scores are used to weight the corresponding values, producing a context vector, which serves as input for subsequent layers. This enables the model to focus on task-relevant information and better capture dependencies and context, ultimately enhancing overall performance.

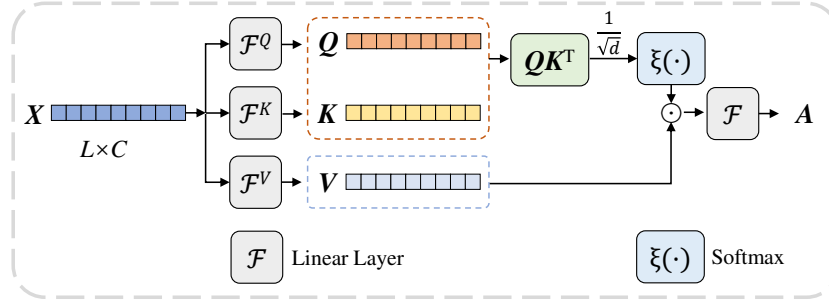


Figure 2.5: Self-attention mechanism. Given an input feature map $X \in \mathbb{R}^{L \times C}$, the Query (Q), Key (K), and Value (V) matrices are typically obtained via linear projections of X . Here, L denotes the sequence length, C is the number of channels, d is the feature dimension of each attention head, and A represents the attended output feature map.

The self-attention mechanism is a specialized form of attention that enables the model to compute dependencies between all elements in a sequence, regardless of their relative positions [37]. Unlike general attention mechanisms, which may rely on external key-value pairs, self-attention derives queries, keys, and values from the same input. Figure 2.5 shows the self-attention mechanism used to compute contextualized representations from the input feature map. Each element is updated by aggregating information from all other elements based on learned attention weights that reflect interactions. This mechanism allows the model to capture complex relationships between elements, regardless of their distance within the sequence, enabling it to effectively model both local and global dependencies.

Chapter 3

MSNet-4mC: Learning Effective Multi-Scale Representations for Identifying DNA N4-Methylcytosine Sites

3.1 Background

DNA methylation is an important type of epigenetic modification and involved in diverse biological processes, such as regulation of gene expression, regulation of chromatin organization, transposon silencing, or genomic imprinting across all domains of life [3, 38, 39, 40]. 5-Methylcytosine (5mC), N6-methyladenine (6mA), and N4-methylcytosine (4mC) are the three prevalent kinds of DNA methylations in genomes [41, 42]. Bisulfite sequencing technique is a commonly used technique for detecting DNA methylation sites across the whole genome. It can detect 5mC modifications but cannot be applied to identify 6mA and 4mC modifications [43]. The single-molecule real-time (SMRT) sequencing method is of great practical value as an effective approach for detecting all of these three modifications [44]. However, it is expensive, time-consuming, and labor-intensive, especially for large-scale

detection experiments.

Machine learning and deep learning have witnessed remarkable achievements in the fields of computer vision, natural language processing, and textual reasoning in recent years, even surpassing human performance in a variety of scenarios [45, 46, 47]. Currently, they have been applied to extract the information from the biological sequences to explore and understand the underlying correlation. With respect to DNA 4mC site prediction, several computational methods have been developed. A good performance has been achieved by applying traditional machine learning-based methods and more recent deep learning-based algorithms. Amongst these methods, iDNA4mC [48] is the first support vector machine based-predictor proposed by Chen *et al.* to predict the 4mC sites. The dataset was first built by Chen *et al.* and further screened against the MethSMRT database [49], which is the common dataset in the later studies. Several conventional machine learning-based methods have been proposed since then, including 4mCPred [50], 4mCPred-SVM [51], Meta-4mCpred [52], and 4mCPred-IFL [53]. More recently, deep learning-based methods have been applied to address this task. To the best of our knowledge, 4mCCNN [54] is the first deep learning-based approach for 4mC site prediction, which utilizes the one-hot encoding as the input and two convolutional operations. In another recent work, DeepTorrent [55] integrates four key features and stacks multi-layer convolutional neural networks, attention layer, and bidirectional long short-term memory, which leads to state-of-the-art performance. Key characteristics of the existing methods for 4mC site prediction are listed in the Table 3.1 with respect to the computational algorithms and features employed. Overall, the vast majority of the competing solutions rely on multiple encoding representations to effectively exploit the sequence information and physicochemical properties or complex cascading strategies to improve the predictive capability.

Although the DNA 4mC site prediction task has been studied for several years, there exist great challenges for excavating the interactions within the DNA sequences and integrating the information implied in the sequences of other species, which is the focus of this work. Herein, we propose MSNet-4mC, a deep learning-based computational architecture consisting of an opening convolutional layer, two similar blocks, and two fully connected layers. Motivated by the endeavors of the research on the complex context dependencies within the biological sequences, we

Table 3.1: Key characteristics of the existing methods for 4mC site prediction

Category	Methods	Year	Algorithm	Features
ML-based	iDNA4mC	2017	SVM	NCP, NF
	4mCPred	2019	SVM	PSTNP, EIIPs
	4mCPred-SVM	2019	SVM	K-mer, BPF, DBE, LPDF
	Meta-4mCpred	2019	SVM, RF, GB, ERT	K-mer, BPF, DBE, LPDF, RFHCP, DPCP, TPCP
	4mCPred-IFL	2019	SVM	BKF, DBPF, PCPs, PseDNC, KNN, EIIPs, MMI, RFHCP
DL-based	4mCNN	2019	CNN	One-hot
	DeepTorrent	2021	CNN, BLSTM	One-hot, CKSNAP, NCP, EIIPs, ENAC, ANF, PSTNP

* *Algorithm*: SVM: Support Vector Machine; RF: Random Forest; GB: Gradient Boosting; ERT: Extremely Randomized Tree; CNN: Convolution Neural Network; BLSTM: Bidirectional Long Short-Term Memory.

* *Features*: NCP: Nucleotide chemical property; NF: Nucleotide frequency; PSTNP: Position-specific trinucleotide propensity; EIIPs: Electron-ion interaction pseudopotentials of trinucleotides; K-mer: K-mer nucleotide frequency; BPF: Binary profile; DBE: Dinucleotide binary encoding; LPDF: Local position-specific dinucleotide frequency; RFHCP: Ring-function-hydrogen-chemical properties; DPCP: Dinucleotide physicochemical properties; TPCP: Trinucleotide physicochemical properties; BKF: Binary and k-mer frequency; DBPF: Dinucleotide binary profile and frequency; PCPs: Physical-chemical properties; PseDNC: Pseudo dinucleotide composition; KNN: K-nearest neighbor; MMI: Multivariate mutual information; CKSNAP: Composition of k-spaced nucleic acid pairs; ENAC: Enhanced nucleic acid composition; ANF: Accumulated nucleotide frequency.

construct the structure of multi-scale receptive fields based on convolutional operations to perceive both the long- and short-range relationships to improve the predictive capability. Moreover, our method not only extracts useful information from one-hot encoding features without the use of complex encoding schemes as the input but also takes into consideration of the unbalanced samples between different species and adjusts the loss with class weights. We conduct a series of experiments to benchmark the performance of our method over two standard datasets. Experimental results show that the proposed method can efficiently identify the 4mC sites and outperform the state-of-the-art approaches.

3.2 Materials and Methods

3.2.1 Datasets

In this study, we adopted two datasets to assess the capability of MSNet-4mC for identifying DNA 4mC sites. The first dataset was constructed by Chen *et al.* [48], which was originally derived from the MethSMRT database. It consisted of six species, *i.e.*, *C.elegans*, *D.melanogaster*, *A.thaliana*, *E.coli*, *G.subterraneus*, and *G.pickeringii*. The positive samples were obtained by applying the Modification QV (modQV) score larger than 30 and the cutoff threshold of CD-HIT at 0.8. The default threshold of modQV score is 30 for calling a position as modified [48]. The CD-HIT software was used to remove the redundant sequences with high similarity [56]. All the sequences of the positive samples had the length of 41 bp with the 4mC sites located in the center. The corresponding negative samples had the same length with the cytosine in the center but were not detected by the SMRT sequencing technology. Therefore, there existed a large number of samples that could be negative samples. To generate a balanced dataset, Chen *et al.* randomly picked out the same number of negative samples as the positive samples for each species. In this work, the samples were randomly divided into the training dataset and the test dataset according to the ratio of 14:1 for each species. For each training or test dataset, it has the same numbers of positive and negative samples. A statistical summary of the dataset is provided in Table 3.2. For the convenience of description and comparison, we used the same name Lin_2017 for the first dataset as Liu *et al.* [55] did.

Table 3.2: Statistical summary of the Lin_2017 dataset

Species	No. 4mC	No. non-4mC	Training dataset		Test dataset	
			No. 4mC	No. non-4mC	No. 4mC	No. non-4mC
<i>C.elegans</i>	1554	1554	1450	1450	104	104
<i>D.melanogaster</i>	1769	1769	1651	1651	118	118
<i>A.thaliana</i>	1978	1978	1846	1846	132	132
<i>E.coli</i>	388	388	362	362	26	26
<i>G.subterraneus</i>	905	905	845	845	60	60
<i>G.pickeringii</i>	569	569	531	531	38	38

* “No.” represents “Number of.”

The second dataset was first used by Liu *et al.* [55], namely Li_2020. Similar to the Lin_2017 dataset, it was also derived from the MethSMRT database and consisted of six species. In this dataset, the positive samples were selected with the

Table 3.3: Statistical summary of the Li_2020 dataset

Species	No. 4mC	No. non-4mC	Training dataset		Test dataset	
			No. 4mC	No. non-4mC	No. 4mC	No. non-4mC
<i>C.elegans</i>	58396	58396	55729	55729	2667	2667
<i>D.melanogaster</i>	57654	57504	53970	53970	3684	3534
<i>A.thaliana</i>	75027	75027	63720	63720	11307	11307
<i>E.coli</i>	2067	2067	1941	1941	126	126
<i>G.subterraneus</i>	15197	15197	9934	9934	5263	5263
<i>G.pickeringii</i>	5724	5724	4514	4514	1210	1210

* “No.” represents “Number of.”

modQV score larger than 30 and the cutoff threshold of CD-HIT at 0.7. Different from the first dataset randomly split as the training and test dataset, the test dataset of the second dataset was filtered with the modQV value larger than 50. A statistical summary of the Li_2020 dataset is provided in Table 3.3. Each species has the same numbers of positive and negative samples, except *D.melanogaster*. The number difference between the positive and the negative samples of *D.melanogaster* is not large. Accordingly, for a fair and objective comparison, the training and test datasets in this work were still kept the same as those of DeepTorrent. Note that there is no overlap between the training and test sets.

3.2.2 MSNet-4mC Architecture

Figure 3.1 illustrates an overview of the architecture of MSNet-4mC. MSNet-4mC employs a novel convolutional neural network framework to predict the 4mC sites given the feature matrix. It consists of an opening convolutional layer, two similar blocks, and two fully connected layers. The two blocks are composed of convolutional layers with the shortcut identity connections inspired by ResNet [26] but with a different structure of multi-scale receptive fields to perceive both the long- and short-range relationships within the DNA sequences. Besides, MSNet-4mC applies the class weights in the cross-entropy loss to adjust the training on different species with unbalanced numbers of samples. This is achieved by the following steps: A species sign is attached to each sequence of the training datasets to indicate the species class when encoding the sequence to obtain the input feature. The class weights for different species are calculated based on the frequencies of the corresponding species. When the sequence is fed into the network, the class weight is applied with its species sign (shown in Figure 3.1) to the CE-loss, where the species

sign is used as an index to select the class weight. Note that the sign is not used in the test phase.

Input feature matrix

Different from previous methods that integrate various feature encoding schemes to represent the sequence as the input to train the model, MSNet-4mC only utilizes the simplest and most common encoding scheme, *i.e.* one-hot encoding. From a theoretical perspective, effective neural networks can extract more powerful and generic features and perceive the inner mathematical relations from the sequence without the need for manually encoding the sequence repeatedly. However, generally speaking, neural networks can only learn from numerical data rather than categorical data. As a consequence, one-hot encoding is used to encode the DNA sequences in this work. In this encoding scheme, each nucleotide is encoded by a four-bit binary vector. Specifically, ‘A’, ‘C’, ‘G’, and ‘T’ are encoded by (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), and (0, 0, 0, 1), respectively. Each DNA sequence with n nucleotides is represented by a $4 \times n$ dimensional binary vector. Additionally, there is a sign indicating the species class for each DNA sequence to introduce the class weights.

Convolutional neural network

Recent years have seen the outstanding success of convolutional neural networks, with various methods proposed and applied in real-world applications. Herein, we design MSNet-4mC by leveraging CNN techniques. The input matrices are fed into the first convolutional layer with 16 output channels, and 3 kernel sizes followed by batch normalization and GELU activation. The output of this layer is then fed to the subsequent block with a shortcut identity connection similar to the ResNet structure. These two blocks consist of a convolution layer with one kernel followed by batch normalization, GELU activation, and dropout layer, three parallel convolution layers with the same output channels and kernel sizes but different receptive fields followed by batch normalization, GELU activation and dropout layer for the merged result of the three anterior outputs, and a convolution layer with one kernel followed by batch normalization. In addition, there is a gating for the output from the convolution layer before merging with the residual. Gating is applied to rectify the channels of the fused feature [57]. More specifically, we employed an additional convolutional layer to aggregate features with multi-scale information.

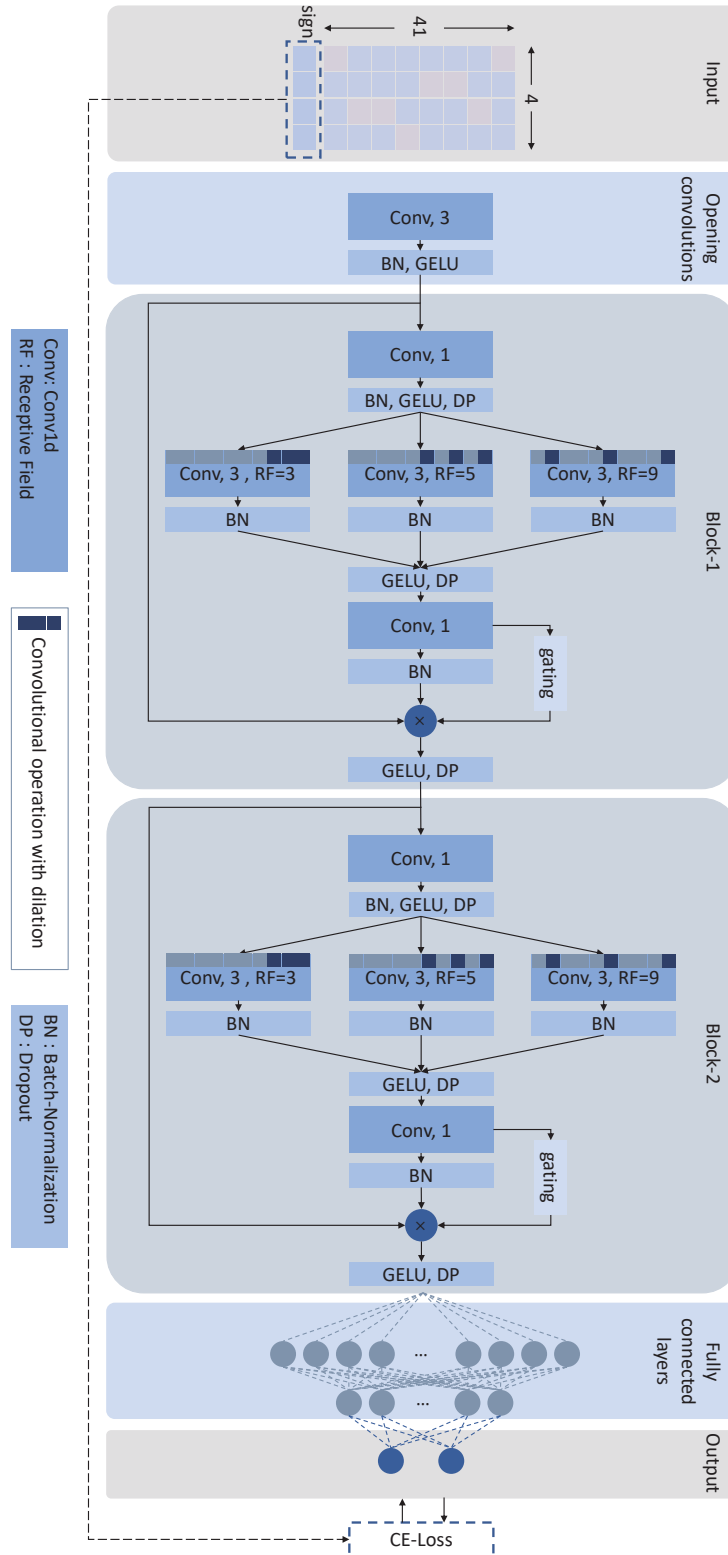


Figure 3.1: An overview of the architecture of MSNet-4mC. The model encodes multi-scale features via a series of dilation convolutions, where convolutions with receptive fields of 3, 5, 9 correspond to dilations of 1, 2, 4, respectively. The class weight is applied with its species sign to the CE-loss, where the species sign serves as an index to enable the correct weight assignment.

However, plugging additional convolutional layers is possible to intensify the difference of mean feature intensities across channels. According, we leverage gating after the multi-scale feature aggregation to alleviate this problem through rectifying features with channel-wise weights computed by a lightweight inter-channel relationship modeling. Between these shortcut blocks, we use GELU activation and dropout layer. The output of the second block is fed to two fully connected layers. These two layers contain 192×41 units and 192 units, respectively. The final output layer is equipped with the weighted cross-entropy loss to balance the number of candidates in different species and the softmax classifier to predict the 4mC sites.

The opening convolutional layer is conducive to handle the issue of vanishing/-exploding gradients which obstruct convergence from the start [58, 59, 60]. Each convolutional layer followed by a batch normalization also serves for the same purpose. The residual is designed as described in ResNet to address the degradation problem, which is realized by the shortcut connections without introducing extra parameters and computational complexity. The three parallel convolutional layers are one of the key components of our framework. Different sites in the biological sequences can have contextual dependencies with each other, where the ranges of these dependencies are difficult to confirm [61]. Several previous studies illustrate that there are also strong correlations within the flanking bases in DNA sequences [62, 63, 64, 65]. This characteristic informs the modeling design, with multi-scale learning offering a simple yet effective realization for this property. Inspired by these insights, we propose to excavate the long- and short-range dependencies of the sequences. Specifically, we realize it through employing convolutional operations with a series of dilations to obtain convolutions with different receptive fields, which can effectively explore the inner relationships. All input sequences are 41 bp in length, with a cytosine nucleotide positioned at the center. This places 20 nucleotides on each side of the central cytosine. To effectively capture contextual patterns at different scales, we employ three parallel convolutional layers with receptive fields of 3, 5, and 9.

Class weights for loss

The number of samples for each species is small. An important step of our pipeline is to integrate the training dataset of six different species into one training dataset and train a base model to avoid overfitting and fully exploit the cross-species relationships. The major problem is that a species with far more samples can have a far

stronger influence on the model optimization than a species with far fewer samples and as such, biased training for the base model can be triggered. Assigning weights to training samples is a common strategy to mitigate the bias caused by imbalanced data [66]. Inspired by this, a possible rectification is to assign larger weights to the species with fewer samples to alleviate the imbalanced training. As a result, we calculated class weights based on the frequencies of corresponding species and applied these weights to the cross-entropy loss (CE-loss) during the training process. Generally, there are different variations of formula that can be used to calculate the class weights. The essential property is to meet the requirement that a smaller size dataset has a greater weight. One optimization goal of the weighted loss is to pay more attention to the data from the small size species. Nevertheless, inappropriate weights may cause over-bias for the small size dataset, or cannot have an obvious effect.

The overall CE-loss can be described as

$$L = \frac{\sum_{n=1}^N l_n}{N}, \quad (3.1)$$

where N spans the minibatch dimension, l_n is the CE-loss for the n -th sequence. Constructed on the vanilla CE-loss, the weighted CE-loss for the n -th sequence can be defined as

$$l_n = - \sum_{i=1}^2 \tilde{\omega}_c p_i \ln q_i, \quad (3.2)$$

where p_i is the ground truth, *i.e.*, represented as a binary indicator (0 or 1), q_i is the probability which is calculated by the softmax function, $\tilde{\omega}_c$ is the normalized weight of the c -th class, respectively. Note that the species sign is used for selecting $\tilde{\omega}_c$. $\tilde{\omega}_c$ is calculated as

$$\tilde{\omega}_c = \frac{C}{\sum_{c'} \omega_{c'}} \omega_c, \quad (3.3)$$

where ω_c is the unnormalized weight of the c -th class. Directly applying the reciprocal of the frequency as the weight for the species to adjust the loss may introduce a concomitant problem of excessive rectification, since our model uses a Softmax-based classifier to perform probability mapping. Considering the sizes of the two datasets used in this work and the difference of the numbers of samples among species, especially for the second dataset where there is a huge imbalance, we adopt

Table 3.4: Hyperparameters for the CNN structure

Name	Module	Hyperparameters
Opening Conv.	Conv-Layer	In channels = 4, Out channels = 16, Kernel size = 3, Dilation = 1
Block 1	Conv-Layer	In channels = 16, Out channels = 16, Kernel size = 1, Dilation = 1
	Conv-Layer	In channels = 16, Out channels = 16, Kernel size = 3, Dilation = 1
	Conv-Layer	In channels = 16, Out channels = 16, Kernel size = 3, Dilation = 2
	Conv-Layer	In channels = 16, Out channels = 16, Kernel size = 3, Dilation = 4
	Conv-Layer	In channels = 48, Out channels = 96, Kernel size = 1, Dilation = 1
Block 2	Conv-Layer	In channels = 96, Out channels = 32, Kernel size = 1, Dilation = 1
	Conv-Layer	In channels = 32, Out channels = 32, Kernel size = 3, Dilation = 1
	Conv-Layer	In channels = 32, Out channels = 32, Kernel size = 3, Dilation = 2
	Conv-Layer	In channels = 32, Out channels = 32, Kernel size = 3, Dilation = 4
	Conv-Layer	In channels = 96, Out channels = 192, Kernel size = 1, Dilation = 1
FC-Layers	FC-Layer	In features = 192×41, Out features = 192
	FC-Layer	In features = 192, Out features = 2

* “Conv.” represents “Convolution;” “Conv-Layer” represents “Convolutional layer;” “FC-Layer” represents “Fully connected layer.”

a relatively smooth formula. To this end, the class weights are calculated by the logarithm of the reciprocal of the frequencies to ensure relatively gentle rectifications. The formula used for the class weights is as follows:

$$\omega_c = \ln \frac{\sum_{c'} f_{c'}}{f_c} = \ln \frac{1}{f_c}, c \in (1, 2, \dots, C), \quad (3.4)$$

where f_c is the frequency of the c -th species, that is, the number of the c -th species’s training dataset divided by the sum of the numbers of the six training datasets, C indicates the number of the classes which is 6 in this work. The sign mentioned in Figure 3.1 is used to indicate the class for the sequence.

Parameters

In this section, we present the hyperparameters for the network and training. Table 3.4 shows the fixed hyperparameters for the CNN. The number of blocks was initially determined based on the data scale of the task. Due to the limited size of the benchmark datasets, we prioritized a compact architecture with fewer blocks. In our experiments, we tested various block configurations. Using only one block resulted in suboptimal performance, while increasing the number of blocks beyond two required a large dropout rate to mitigate overfitting tendencies, without observable improvements in accuracy. Following the principle of Occam’s Razor, we set

the number of blocks to two by default, as it offers a balanced trade-off between model complexity and performance.

Besides, for the first dataset, we trained the network using a batch size of 256 with the SGD optimizer and the cosine learning rate schedule. The number of warm-up epochs was 20, and the sum of training epochs was 300. The initial learning rate was 0.05 for the base model, and 0.01 for the fine-tuning process. In general, for training deep learning models, small datasets can lead to the problem of overfitting. Applying dropout layers is a common method to avoid overfitting. Since the number of samples for each species is different, the dropout rate should be assigned accordingly. That is, a small-size dataset (of a species) requires to be trained with relatively higher dropout rates. With dropouts, some of the feature units are randomly and temporarily deactivated according to the probabilities (*i.e.* the assigned rates of the dropout layers) and will not be optimized during the iterations. There are several dropout layers in the network. More specifically, three dropout layers in block 1 and the first two dropout layers in block 2 were assigned the same dropout rates, respectively. The last one was the last dropout layer in block 2. Three dropout rates were 0.2, 0.5, and 0.8 in turn when training the base model. When it came to performing fine-tuning on the training dataset of each specific species, if the number of the samples for the training set was larger than 2900, three dropout rates were set as 0.2, 0.5, and 0.8 in turn. Otherwise, the three dropout rates were set as 0.2, 0.8, and 0.8 in turn. When training the species-specific models from scratch, the dropout rates were assigned the same values as those used for fine-tuning.

For the second dataset, we trained the base model with a batch size of 512 and an initial learning rate of 0.025, respectively. The initial learning rate was 0.01 for the fine-tuning phase. The optimizer, learning strategy, and training epochs were the same as those for the first dataset. Because the number of samples is substantially increased compared to the first dataset, the dropout rates have been adjusted to a certain extent. The dropout layers in block 1 were no longer used. While training the base model, the first two dropout layers in block 2 were not used. Only the last dropout layer in block 2 was valid with the value of 0.5. When conducting the fine-tuning step, the first two dropout layers and the last dropout layer in block 2 were assigned the dropout rates of 0.25 and 0.5, respectively. Table 3.5 summarizes the training protocols with the corresponding hyperparameters. MSNet-4mC was trained on an Intel(R) Core(TM) i7-11800H @ 2.30 GHz and NVIDIA

Table 3.5: Hyperparameters for the training process

Dataset	Hyperparameters	Values		
		Scratch	Base model	Fine-tuning
Lin_2017	Batch size	256	256	256
	Warm-up epochs	20	20	20
	Total epochs	300	300	300
	Initial learning rate	0.01	0.05	0.01
	Dropout layer	0.2, 0.5 (or 0.8), 0.8	0.2, 0.5, 0.8	0.2, 0.5 (or 0.8), 0.8
Li_2020	Batch size	256	512	256
	Warm-up epochs	20	20	20
	Total epochs	300	300	300
	Initial learning rate	0.01	0.025	0.01
	Dropout layer	0, 0.25, 0.5	0, 0, 0.5	0, 0.25, 0.5

GeForce RTX 3060 GPU. Its deep learning framework was implemented using PyTorch 1.11.0 for training/testing.

3.2.3 Performance Evaluation Metrics

To evaluate the predictive capability of MSNet-4mC, we utilized six common performance evaluation metrics, including sensitivity (SN), specificity (SP), precision, accuracy (ACC), Matthew's correlation coefficient (MCC), and F1-score. The mathematical expressions of these evaluation metrics are provided as follows:

$$SN = \frac{TP}{TP + FN}, \quad (3.5)$$

$$SP = \frac{TN}{TN + FP}, \quad (3.6)$$

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (3.7)$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}, \quad (3.8)$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \quad (3.9)$$

$$F1_{score} = \frac{2 \times TP}{2 \times TP + FP + FN}, \quad (3.10)$$

where TP is True Positive, TN is True Negative, FP is False Positive, and FN is False Negative, respectively. In addition, the receiver-operating characteristic

(ROC) curve is also plotted to intuitively evaluate the overall performance. Another important performance measure is the area-under-the-curve (AUC), which is utilized to quantitatively assess the overall performance of the model.

3.3 Results

In this section, we discuss the performance evaluation results of MSNet-4mC in detail on both the Lin_2017 dataset [48] and Li_2020 dataset [55].

3.3.1 Performance Evaluation on the Lin_2017 Dataset

Ablation study

To investigate the contributions of the class weight module in the proposed method, an ablation study was performed on the Lin_2017 dataset [48]. The experiments were implemented on the merged training dataset by combining six species with or without the class weight module. We conducted each kind of experiment 10 times. The average results on all the performance metrics are presented in the Table 3.6. The results show that the component of class weights resulted in a modest decline of accuracy around of 0.1% on three large size species but a distinct boost on the other three small size species, which is consistent with our expectation. The minor change in the loss resulted in a more than 3% gain on *G.subterraneus* and *G.pickeringii*. Table 3.7 shows the results of the per-class mean and variance on the metrics. Compared with the variance results in Table 3.7, the models with the class weight module achieved a more balanced performance on different species. These results indicate that such component is helpful for the effectiveness of the pre-training model to extract the information derived from the small size species. The cross-species features utilized by the pre-training model can play an important role in the subsequent fine-tuning step on specific-species models.

In addition, to examine the effectiveness of the convolutional operations with multi-scale receptive fields, an ablation study was further conducted on the Lin_2017 dataset. In particular, we removed the two convolutions with receptive fields of 5 and 9 in two blocks for performance comparison. The experiments were implemented on the merged training dataset of six species with or without parallel convolutional layers. Accordingly, we also conducted experiments 10 times for each model. The average performance results in terms of all the performance metrics are

Table 3.6: Ablation study of the class weight module on the Lin_2017 dataset

Class weight	Species	MCC	ACC	F1-score	SN	SP	Precision	AUC
Without	All species	0.5871	0.7930	0.7900	0.7797	0.8063	0.8020	0.8585
	<i>C. elegans</i>	0.5868	0.7923	0.7974	0.8192	0.7654	0.7783	0.8606
	<i>D. melanogaster</i>	0.6176	0.8072	0.8138	0.8441	0.7703	0.7873	0.8858
	<i>A. thaliana</i>	0.5984	0.7966	0.7835	0.7371	0.8561	0.8379	0.8511
	<i>E. coli</i>	0.7900	0.8942	0.8936	0.8885	0.9000	0.9005	0.9318
	<i>G. subterraneus</i>	0.5435	0.7683	0.7508	0.7000	0.8367	0.8131	0.8411
	<i>G. pickeringii</i>	0.4184	0.7079	0.6960	0.6711	0.7447	0.7259	0.7793
With	All species	0.5985	0.7990	0.7972	0.7908	0.8071	0.8044	0.8628
	<i>C. elegans</i>	0.5839	0.7913	0.7954	0.8115	0.7712	0.7808	0.8594
	<i>D. melanogaster</i>	0.6149	0.8055	0.8146	0.8551	0.7559	0.7787	0.8820
	<i>A. thaliana</i>	0.5938	0.7955	0.7861	0.7530	0.8379	0.8236	0.8487
	<i>E. coli</i>	0.7928	0.8962	0.8962	0.8962	0.8962	0.8968	0.9259
	<i>G. subterraneus</i>	0.6073	0.8008	0.7883	0.7417	0.8600	0.8436	0.8741
	<i>G. pickeringii</i>	0.4902	0.7421	0.7215	0.6711	0.8132	0.7831	0.8154

Table 3.7: Performance results on per-class mean and variance of the metrics

Module	Metrics	MCC	ACC	F1_score	SN	SP	Precision	AUC
No class weight	Mean	0.5925	0.7944	0.7892	0.7767	0.8122	0.8072	0.8583
	Variance	0.0121	0.0031	0.0036	0.0062	0.0031	0.0029	0.0021
Class weight	Mean	0.6138	0.8052	0.8004	0.7881	0.8224	0.8178	0.8676
	Variance	0.0081	0.0021	0.0027	0.0056	0.0024	0.0018	0.0011

Table 3.8: Ablation study of the convolutions with multi-scale receptive fields on the Lin_2017 dataset

Para. Convs.	Species	MCC	ACC	F1-score	SN	SP	Precision	AUC
Without	All species	0.5566	0.7781	0.7787	0.7810	0.7753	0.7770	0.8394
	<i>C. elegans</i>	0.5277	0.7635	0.7671	0.7798	0.7471	0.7553	0.8322
	<i>D. melanogaster</i>	0.5961	0.7962	0.8052	0.8424	0.7500	0.7723	0.8697
	<i>A. thaliana</i>	0.5806	0.7898	0.7848	0.7674	0.8121	0.8037	0.8376
	<i>E. coli</i>	0.7393	0.8692	0.8685	0.8654	0.8731	0.8726	0.9419
	<i>G. subterraneus</i>	0.5210	0.7575	0.7382	0.6833	0.8317	0.8030	0.8143
	<i>G. pickeringii</i>	0.3861	0.6921	0.7052	0.7368	0.6474	0.6765	0.7518
With	All species	0.5985	0.7990	0.7972	0.7908	0.8071	0.8044	0.8628
	<i>C. elegans</i>	0.5839	0.7913	0.7954	0.8115	0.7712	0.7808	0.8594
	<i>D. melanogaster</i>	0.6149	0.8055	0.8146	0.8551	0.7559	0.7787	0.8820
	<i>A. thaliana</i>	0.5938	0.7955	0.7861	0.7530	0.8379	0.8236	0.8487
	<i>E. coli</i>	0.7928	0.8962	0.8962	0.8962	0.8962	0.8968	0.9259
	<i>G. subterraneus</i>	0.6073	0.8008	0.7883	0.7417	0.8600	0.8436	0.8741
	<i>G. pickeringii</i>	0.4902	0.7421	0.7215	0.6711	0.8132	0.7831	0.8154

* “Para. Convs.” represents “Parallel Convolutional Layers.”

provided in Table 3.8. As can be seen, the models trained with the convolutional operations with multi-scale receptive fields outperformed the models trained only with a single-scale receptive field in the majority of the cases. This clearly demonstrates the effectiveness of the proposed modules.

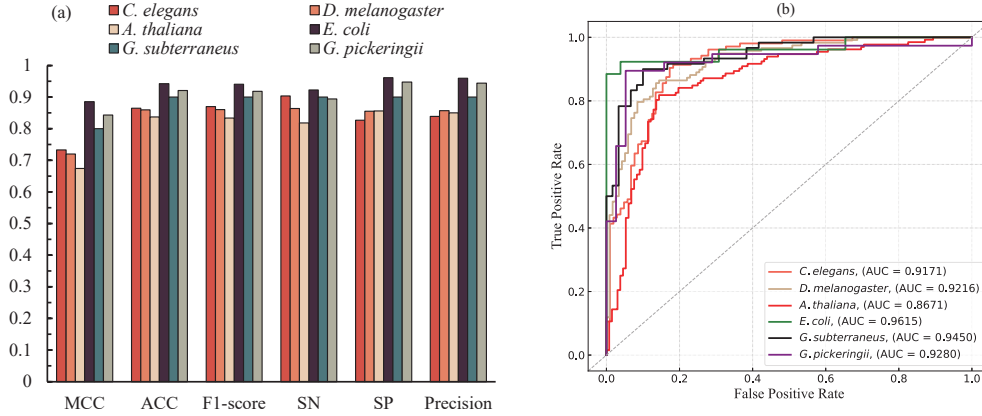


Figure 3.2: Species-specific performance of MSNet-4mC on the Lin_2017 dataset.

Performance comparison with the existing methods on the Lin_2017 dataset

In order to comprehensively evaluate MSNet-4mC, we implemented two sets of experiments: The first one is to train the model from scratch on each species training dataset and test its performance on the corresponding test dataset. The second one consists of two steps, which involve firstly training a base model with the merged training dataset and then fine-tuning the hyperparameters to retrain the species-specific model on each species training dataset. Pre-trained models can serve as the initialization parameters for subsequent training tasks, which help to avoid overfitting. Fine-tuning the learnable parameters to retrain the models can improve classification accuracy. On the other hand, training a base model at first with the merged dataset of six species can be useful for extracting potential cross-species features. Thereby, we conducted the experiments with the two-step training strategy.

The performance results of MSNet-4mC and DeepTorrent [55] trained from scratch for six species in terms of all evaluation metrics are presented in Table 3.9. The best result in the table is highlighted in bold. Our models achieved higher MCC, ACC, SN, and AUC values across all species. It achieved the best performance in terms of all six metrics on five species. Particularly, the results show that our method gained around 6% improvement in accuracy and around 8% improvement in AUC. Our method clearly outperformed DeepTorrent on the Lin_2017 dataset by a larger margin when training the species-specific models from scratch, which demonstrates that our proposed framework could effectively extract and leverage the implicit information from biological sequences.

Table 3.9: Training the species-specific models from scratch: Performance comparison of MSNet-4mC and DeepTorrent on the Lin_2017 dataset

Species	Scratch	MCC	ACC	SN	SP	Precision	AUC
<i>C. elegans</i>	MSNet-4mC	0.6732	0.8365	0.8462	0.8269	0.8302	0.9053
	DeepTorrent	0.5603	0.7788	0.7308	0.8269	0.8085	0.8318
<i>D. melanogaster</i>	MSNet-4mC	0.6695	0.8347	0.8390	0.8305	0.8319	0.9040
	DeepTorrent	0.6017	0.8008	0.8051	0.7966	0.7983	0.8467
<i>A. thaliana</i>	MSNet-4mC	0.6364	0.8182	0.8182	0.8182	0.8182	0.8526
	DeepTorrent	0.4550	0.7273	0.7045	0.7500	0.7381	0.7825
<i>E. coli</i>	MSNet-4mC	0.7357	0.8654	0.9231	0.8077	0.8276	0.9571
	DeepTorrent	0.5034	0.7500	0.8077	0.6923	0.7241	0.7959
<i>G. subterraneus</i>	MSNet-4mC	0.7000	0.8417	0.9500	0.7333	0.7808	0.9408
	DeepTorrent	0.6337	0.8167	0.8000	0.8333	0.8276	0.8558
<i>G. pickeringii</i>	MSNet-4mC	0.7895	0.8947	0.8947	0.8947	0.8947	0.9176
	DeepTorrent	0.6396	0.8158	0.7368	0.8947	0.8750	0.8747

Furthermore, we performed the two-step experiments. The performance of MSNet-4mC is shown in Figure 3.2 (a), with the ROC curves of MSNet-4mC on six species plotted in Figure 3.2 (b). The results show that MSNet-4mC achieved the AUC value of higher than 0.86 and the ACC value of higher than 0.84 across all the six different species, with an average AUC value of 0.92 and ACC value of 0.89, respectively. Table 3.10 provides the performance of MSNet-4mC in terms of all metrics. The detailed performance results of MSNet-4mC and the other three state-of-the-art methods for the species-specific models are provided in Table 3.11. The results of other methods are quoted from the Supplementary materials of DeepTorrent [55]. As can be seen, our method achieved the best performance in terms of all performance metrics (*i.e.* SN, SP, ACC, and MCC) for three species (*C.elegans*, *A.thaliana* and *G.pickeringii*). In addition, our method achieved the best performance in terms of accuracy and MCC on all six species. The results show the identification capability of our method is better than other methods. The results of the species-specific models both trained from scratch and trained after fine-tuning show that MSNet-4mC outperformed DeepTorrent, thereby suggesting the effectiveness of MSNet-4mC.

To better understand how MSNet-4mC learns effective representations and contributes to the improved performance, we further calculated the spatial distributions of the 4mC and non-4mC samples for each species. Specifically, we utilized a visualization tool t-SNE [67] to intuitively visualize and compare the input features and those extracted from MSNet-4mC. Figure 3.6 shows the distributions of the

Table 3.10: After Fine-tuning: Performance of MSNet-4mC for species-specific 4mC site prediction on the independent test dataset of Lin_2017 dataset

Species	MCC	ACC	F1-score	SN	SP	Precision	AUC
<i>C. elegans</i>	0.7329	0.8654	0.8704	0.9038	0.8269	0.8393	0.9171
<i>D. melanogaster</i>	0.7204	0.8602	0.8608	0.8644	0.8559	0.8571	0.9216
<i>A. thaliana</i>	0.6747	0.8371	0.8340	0.8182	0.8561	0.8504	0.8671
<i>E. coli</i>	0.8853	0.9423	0.9412	0.9231	0.9615	0.9600	0.9615
<i>G. subterraneus</i>	0.8000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9450
<i>G. pickeringii</i>	0.8433	0.9211	0.9189	0.8947	0.9474	0.9444	0.9280

Table 3.11: After Fine-tuning: Performance comparison of MSNet-4mC and the existing methods on the independent test dataset of Lin_2017 dataset

Method	Species	SN (%)	SP (%)	ACC (%)	MCC (%)
MSNet-4mC	<i>C. elegans</i>	90.38	82.69	86.54	73.29
	<i>D. melanogaster</i>	86.44	85.59	86.02	72.04
	<i>A. thaliana</i>	81.82	85.61	83.71	67.47
	<i>E. coli</i>	92.31	96.15	94.23	88.53
	<i>G. subterraneus</i>	90.00	90.00	90.00	80.00
	<i>G. pickeringii</i>	89.47	94.74	92.11	84.33
DeepTorrent	<i>C. elegans</i>	90.38	80.77	85.58	71.49
	<i>D. melanogaster</i>	88.98	82.20	85.59	71.35
	<i>A. thaliana</i>	81.82	78.03	79.92	59.89
	<i>E. coli</i>	92.31	84.62	88.46	77.15
	<i>G. subterraneus</i>	83.33	91.67	87.50	75.26
	<i>G. pickeringii</i>	86.84	94.74	90.79	81.83
4mCPred	<i>C. elegans</i>	85.58	78.85	82.21	65.00
	<i>D. melanogaster</i>	83.90	81.36	82.63	65.00
	<i>A. thaliana</i>	76.52	76.52	76.52	53.00
	<i>E. coli</i>	84.62	80.77	82.69	65.00
	<i>G. subterraneus</i>	91.67	75.00	83.33	68.00
	<i>G. pickeringii</i>	86.84	68.42	77.63	56.00
iDNA4mC	<i>C. elegans</i>	80.77	73.08	76.92	54.00
	<i>D. melanogaster</i>	74.58	77.97	76.27	53.00
	<i>A. thaliana</i>	80.30	77.27	78.79	58.00
	<i>E. coli</i>	96.15	69.23	82.69	68.00
	<i>G. subterraneus</i>	85.00	76.67	80.83	62.00
	<i>G. pickeringii</i>	81.58	78.95	80.26	61.00

4mC and non-4mC samples of the *E.coli* dataset in the two-dimensional space. Figure 3.6(a), (b), and (c) are the t-SNE plots of the input features, the output features of block 2, and the output features of second fully connected layer, respectively. It can be seen from Figure 3.6(a) that the original distribution of the positive and negative samples is entirely mixed together whose boundary is difficult to distinguish. In Figure 3.6(b), the distribution of the features learned after block 2 shows two relatively clear clusters despite a few overlapping samples. Furthermore, as shown in Figure 3.6(c), the distribution of the features extracted from the second fully connected layer between the positive and negative samples in the feature space is

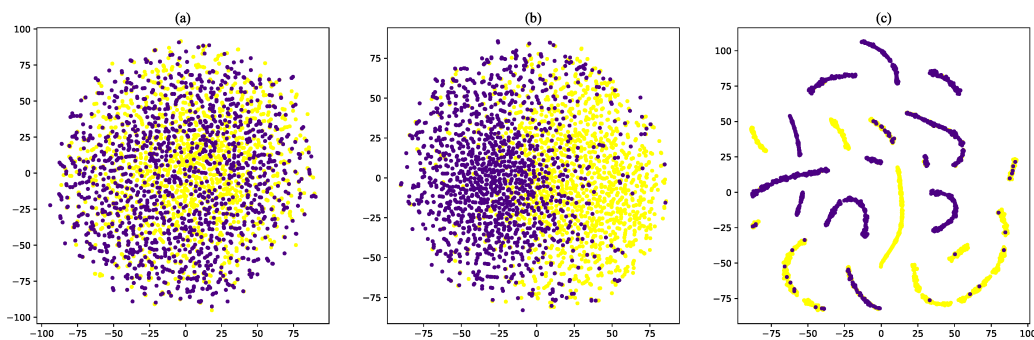


Figure 3.3: t-SNE visualization of the *C.elegans* dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.

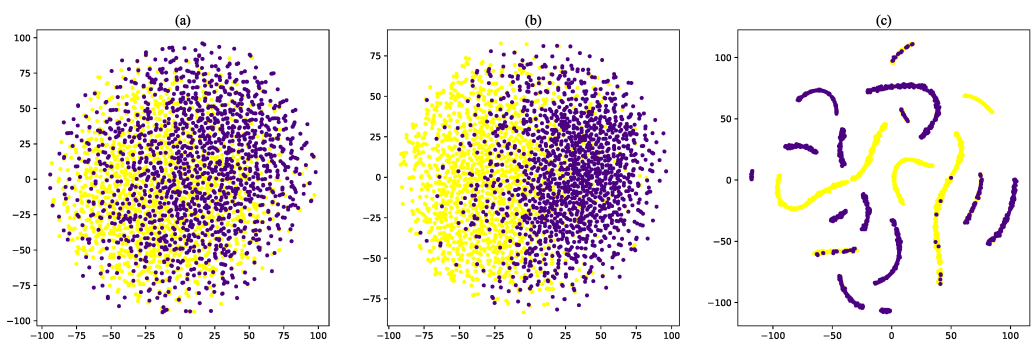


Figure 3.4: t-SNE visualization of the *D.melanogaster* dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.

further clear and separated from each other. In addition, the t-SNE plots for the other five species are provided in Figures 3.3, 3.4, 3.5, 3.7, and 3.8, respectively. Altogether, these results suggest that MSNet-4mC is indeed capable of learning effective representations for 4mC site identification.

3.3.2 Performance Evaluation on the Li_2020 Dataset

To demonstrate the generalization ability and effectiveness of MSNet-4mC, we further conducted experiments on the Li_2020 dataset [55]. Similar to the evaluation on the first dataset, we implemented two sets of experiments. We conducted the experiments by training the models from scratch using the second dataset. The performance results of MSNet-4mC trained from scratch for the six species in terms of all evaluation metrics are provided in Table 3.12. Furthermore, we trained a base model on the training dataset by combining the training datasets of the six species at first. Then we retrained the model on each species-specific training dataset to obtain species-specific models after fine-tuning the unified training framework.

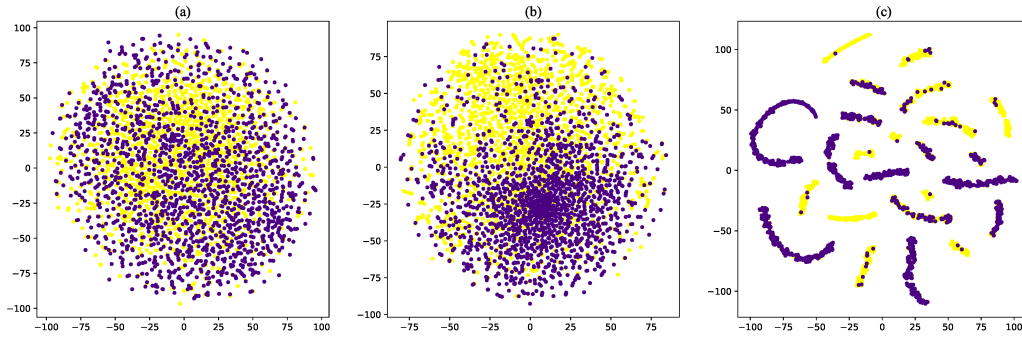


Figure 3.5: t-SNE visualization of the *A.thaliana* dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.

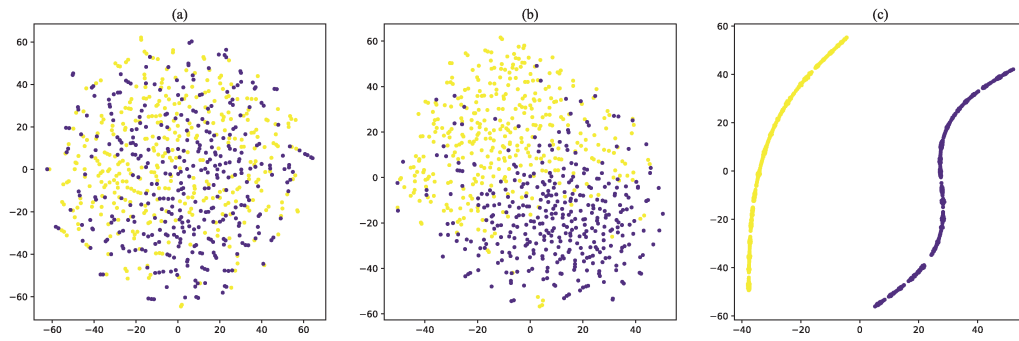


Figure 3.6: t-SNE visualization of the *E.coli* dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.

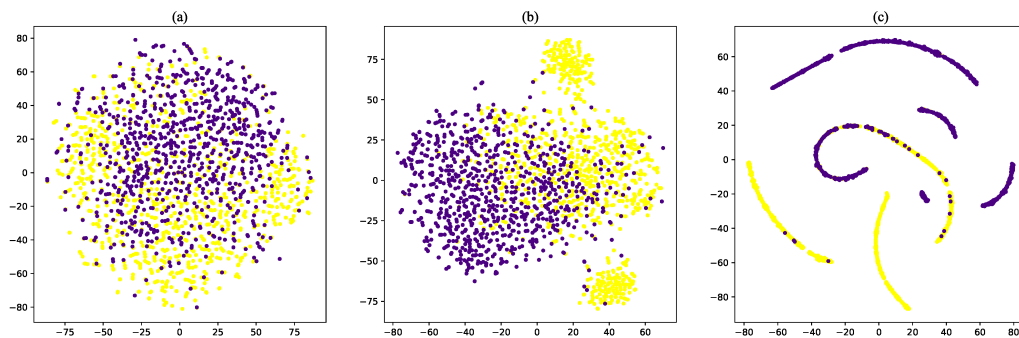


Figure 3.7: t-SNE visualization of the *G.subterraneus* dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.

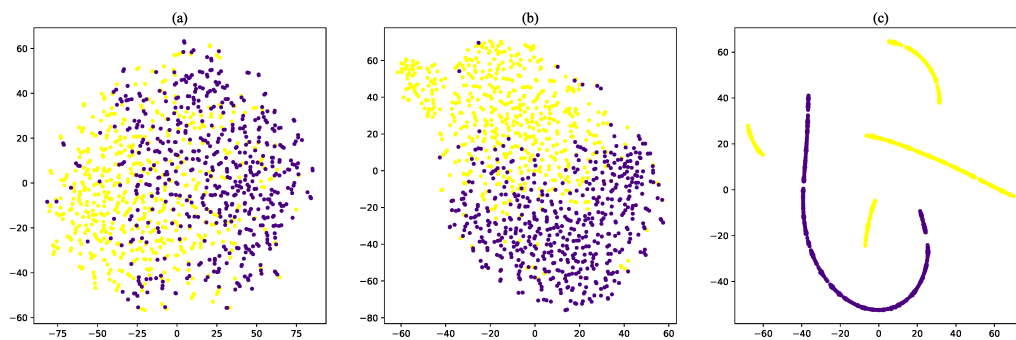


Figure 3.8: t-SNE visualization of the *G.pickeringii* dataset in a two-dimensional feature space: (a) the input features, (b) the features after block 2, and (c) the features of the second fully connected layer.

Table 3.12: Training the species-specific models from scratch: Performance of MSNet-4mC on the Li_2020 dataset

Species	Scratch	AUC	MCC	ACC	F1-score	SN	SP	Precision
<i>C. elegans</i>	Scratch_C	0.9743	0.8446	0.9216	0.9238	0.9501	0.8931	0.8989
<i>D. melanogaster</i>	Scratch_D	0.9792	0.8749	0.9367	0.9398	0.9688	0.9032	0.9126
<i>A. thaliana</i>	Scratch_A	0.9109	0.6818	0.8409	0.8398	0.8345	0.8472	0.8452
<i>E. coli</i>	Scratch_E	0.9834	0.8991	0.9484	0.9502	0.9841	0.9127	0.9185
<i>G. subterraneus</i>	Scratch_GS	0.8706	0.5782	0.7854	0.8013	0.8655	0.7053	0.7460
<i>G. pickeringii</i>	Scratch_GP	0.9399	0.7169	0.8562	0.8638	0.9124	0.8000	0.8202

As a result, the base model achieved a prediction accuracy of 81.09% on the merged test dataset. The performance of the models after fine-tuning is shown in Figure 3.9. Figure 3.9 (a) shows the evaluation metrics, and the ROC curves of MSNet-4mC on six species are plotted in Figure 3.9 (b). Our method achieved an average AUC value of 0.95 and an average ACC value of 0.90, respectively. The AUC for predicting 4mC sites in *C.elegans*, *D.melanogaster*, *A.thaliana*, *E.coli*, *G.subterraneus* and *G.pickeringii* were 0.976, 0.982, 0.916, 0.996, 0.898 and 0.954, respectively.

Table 3.13 presents the performance comparison of MSNet-4mC and other state-of-the-art methods in terms of all evaluation metrics. Figure 3.10 shows the performance comparison results for the species-specific models. The results of other methods are quoted directly from the Supplementary materials of DeepTorrent [55]. As can be seen, our method outperformed the other methods across all species in terms of six performance metrics (including AUC, MCC, ACC, F1-score, SP, and Precision). It achieved the best performance in terms of all seven metrics with five species, with the only exception of *A.thaliana*. Specifically, our method gained around 1.3% performance improvement of accuracy over SOTA. With regard to *G.*

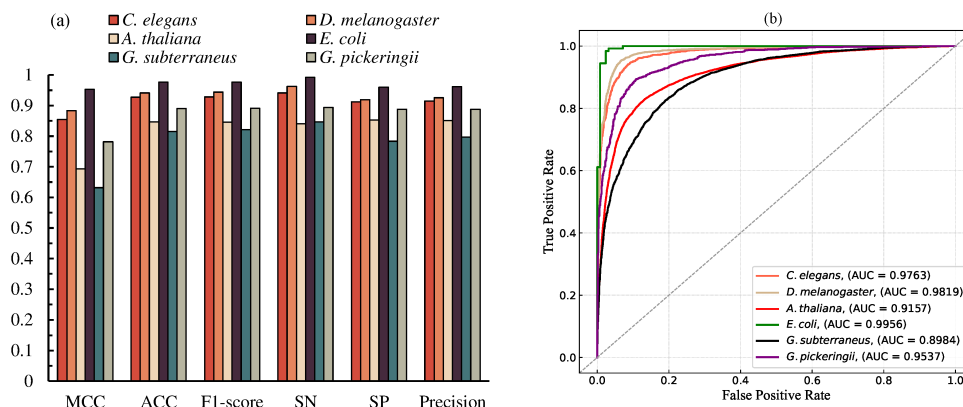


Figure 3.9: Species-specific performance of MSNet-4mC on the Li_2020 dataset.

subterraneus, it resulted in 4.2% improvement on accuracy and 4.4% improvement on AUC. Additionally, some performances in the specific dataset were saturated, making the gain less pronounced. Overall, the performance of MSNet-4mC is superior to others.

3.4 Discussions

In this study, we proposed MSNet-4mC, a novel deep learning-based approach for predicting DNA 4mC sites. MSNet-4mC is developed based on the CNN, consisting of an opening convolutional layer, two similar blocks based on convolutional layers, and two fully connected layers. The main innovation of our work is the construction of a built-in scale-aware learning structure based on convolutional operations with multi-scale receptive fields to perceive the long- and short-range dependencies implied in the biological sequences and incorporation of the species frequency to rectify the training on different species with unbalanced numbers of samples. We performed the experiments on two benchmark datasets. Extensive experiments consistently show that our method outperforms others despite not using complex encoding representations as the input, thereby highlighting the effectiveness of MSNet-4mC to extract the important and relevant features for identification of 4mC sites. In addition, we have also provided an optimization perspective for exploiting the long- and short-range relationships to better address the biological sequence classification problems. The core structure of the network is designed based on the inherent properties of genomic sequence data rather than the specifics of any particular dataset. The use of multi-scale feature extraction reflects general

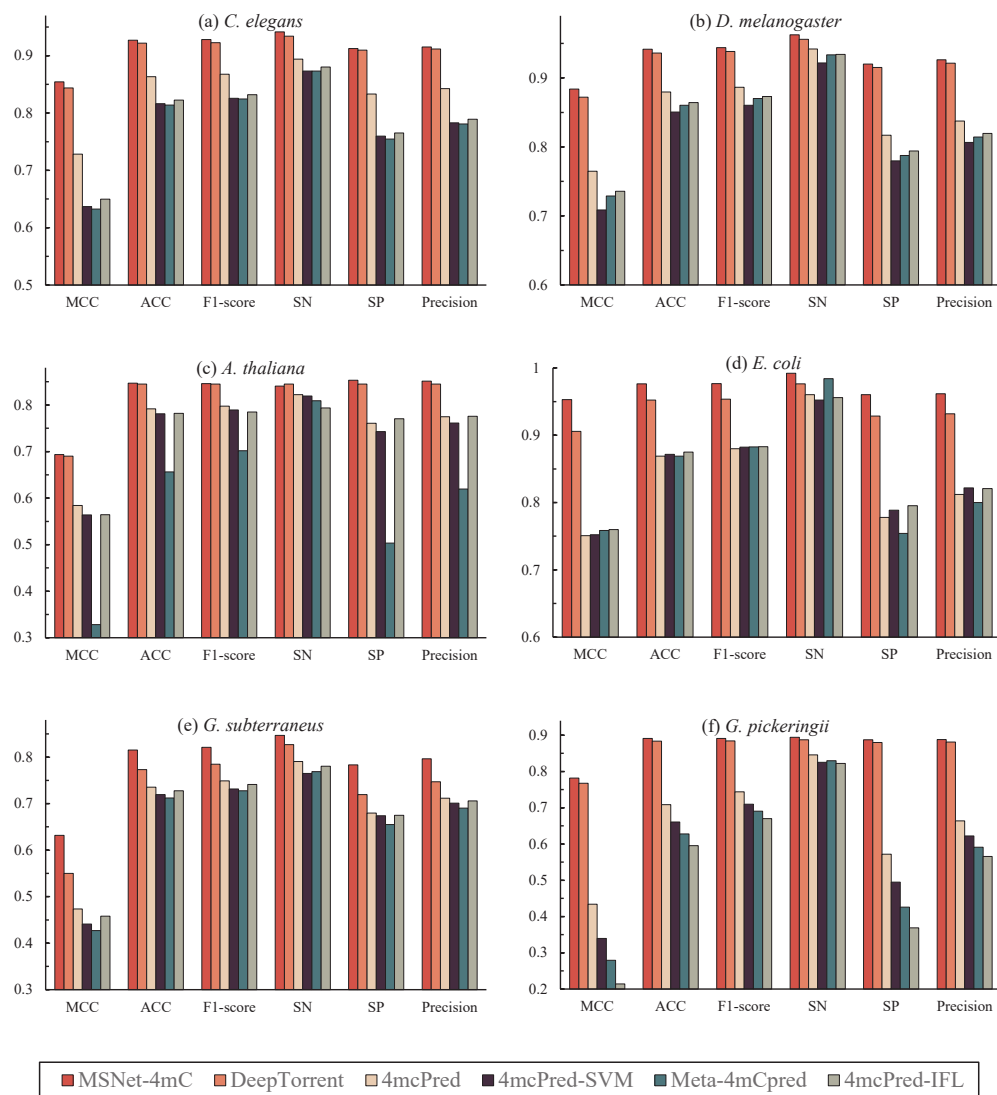


Figure 3.10: Performance comparison between MSNet-4mC and other state-of-the-art methods on the Li₂₀₂₀ dataset.

Table 3.13: After Fine-tuning: Performance comparison of MSNet-4mC and the existing methods on the independent test dataset of Li_2020 dataset

Method	Species	AUC	MCC	ACC	F1-score	SN	SP	Precision
MSNet-4mC	<i>C. elegans</i>	0.9763	0.8545	0.9271	0.9281	0.9415	0.9126	0.9151
	<i>D. melanogaster</i>	0.9819	0.8839	0.9417	0.9440	0.9625	0.9199	0.9261
	<i>A. thaliana</i>	0.9157	0.6937	0.8468	0.8459	0.8405	0.8531	0.8512
	<i>E. coli</i>	0.9956	0.9529	0.9762	0.9766	0.9921	0.9603	0.9615
	<i>G. subterraneus</i>	0.8984	0.6317	0.8152	0.8209	0.8470	0.7834	0.7964
	<i>G. pickeringii</i>	0.9537	0.7818	0.8909	0.8913	0.8940	0.8876	0.8883
DeepTorrent	<i>C. elegans</i>	0.9739	0.8439	0.9218	0.9228	0.9340	0.9096	0.9118
	<i>D. melanogaster</i>	0.9784	0.8722	0.9359	0.9383	0.9558	0.9151	0.9215
	<i>A. thaliana</i>	0.9135	0.6899	0.8450	0.8450	0.8451	0.8449	0.8449
	<i>E. coli</i>	0.9837	0.9058	0.9524	0.9535	0.9762	0.9286	0.9318
	<i>G. subterraneus</i>	0.8543	0.5498	0.7733	0.7849	0.8269	0.7197	0.7469
	<i>G. pickeringii</i>	0.9488	0.7678	0.8839	0.8843	0.8876	0.8802	0.8811
4mCPred	<i>C. elegans</i>	0.9374	0.7284	0.8635	0.8675	0.8939	0.8331	0.8427
	<i>D. melanogaster</i>	0.9526	0.7649	0.8795	0.8866	0.9419	0.8170	0.8374
	<i>A. thaliana</i>	0.8678	0.5841	0.7915	0.7977	0.8221	0.7609	0.7747
	<i>E. coli</i>	0.9525	0.7507	0.8690	0.8800	0.9603	0.7778	0.8121
	<i>G. subterraneus</i>	0.8167	0.4735	0.7353	0.7492	0.7906	0.6800	0.7119
	<i>G. pickeringii</i>	0.8134	0.4339	0.7087	0.7437	0.8455	0.5719	0.6639
4mCPred-SVM	<i>C. elegans</i>	0.9100	0.6370	0.8163	0.8256	0.8733	0.7597	0.7829
	<i>D. melanogaster</i>	0.9290	0.7085	0.8506	0.8603	0.9218	0.7797	0.8065
	<i>A. thaliana</i>	0.8584	0.5637	0.7810	0.7891	0.8192	0.7428	0.7611
	<i>E. coli</i>	0.9475	0.7522	0.8715	0.8824	0.9524	0.7886	0.8219
	<i>G. subterraneus</i>	0.7930	0.4410	0.7196	0.7318	0.7652	0.6740	0.7011
	<i>G. pickeringii</i>	0.7512	0.3396	0.6608	0.7095	0.8254	0.4950	0.6222
Meta-4mCpred	<i>C. elegans</i>	0.8765	0.6325	0.8140	0.8244	0.8733	0.7548	0.7808
	<i>D. melanogaster</i>	0.9228	0.7288	0.8605	0.8700	0.9335	0.7875	0.8145
	<i>A. thaliana</i>	0.6616	0.3280	0.6561	0.7018	0.8091	0.5031	0.6196
	<i>E. coli</i>	0.9751	0.7585	0.8690	0.8826	0.9841	0.7540	0.8000
	<i>G. subterraneus</i>	0.7800	0.4270	0.7121	0.7276	0.7690	0.6553	0.6905
	<i>G. pickeringii</i>	0.7344	0.2792	0.6277	0.6903	0.8298	0.4256	0.5909
4mCPred-IFL	<i>C. elegans</i>	0.8695	0.6496	0.8226	0.8321	0.8801	0.7652	0.7890
	<i>D. melanogaster</i>	0.9255	0.7356	0.8642	0.8731	0.9342	0.7942	0.8196
	<i>A. thaliana</i>	0.8461	0.5644	0.7821	0.7847	0.7937	0.7705	0.7759
	<i>E. coli</i>	0.9483	0.7600	0.8748	0.8831	0.9558	0.7953	0.8207
	<i>G. subterraneus</i>	0.7874	0.4579	0.7277	0.7412	0.7804	0.6750	0.7057
	<i>G. pickeringii</i>	0.6575	0.2142	0.5955	0.6703	0.8223	0.3686	0.5657

principles of capturing long- and short-range sequence patterns, making the architecture applicable to a variety of DNA and RNA sequence-based tasks.

Despite the outstanding performance of MSNet-4mC, we speculate that there is still much room to further improve its potential. Comparing the results of the trained models from scratch on each species dataset and the base model on the merged six species datasets, the average performance of the models trained from scratch was better than that of the base model. The decrease in the classification performance of the base model might be due to the noisy/redundant information between different

species whose removal might help to further improve the performance. Accordingly, effective strategies to improve this aspect may help to achieve a substantial improvement in the predictive performance of identifying 4mC sites from genomic sequences. In addition, there might also exist several potential useful strategies that can be further employed to improve our framework. To enhance the perceptive capability of the framework by leveraging the long- and short-range relationships, one could design new strategies to identify more important representations by taking into consideration varying distances from the central 4mC sites rather than equal distances applied in this study. Moreover, the attention mechanism can be incorporated into the framework to adaptively pay attention to more important information and effectively extract features with strong correlations. Finally, the transfer learning strategy can also be leveraged to obtain a pre-trained model on a larger dataset and retrain the species-specific model to improve its performance.

Chapter 4

SFM-Net: Selective Fusion of Multi-Way Protein Feature Network for Predicting Binding Affinity Changes upon Mutations

4.1 Background

Protein-protein interactions (PPIs) play a vital role in almost all biological processes, such as enzyme catalysis, signal transduction, and gene expression regulation [12, 13, 68]. Mutations introduce variation in the genome, driving diversification and evolution. They can be neutral, beneficial, or detrimental, influencing adaptation to changing environments or contributing to genetic disorders and diseases. When mutations occur in PPIs, they can significantly impact cellular functions and initiate disease processes [69]. Such mutations may either disrupt or enhance protein interactions, directly affecting binding affinity, specificity, and stability of protein complexes, thereby influencing various biological pathways. Understanding the underlying mechanisms and the effects of the amino acid mutations on PPIs

not only advances fundamental scientific research regarding the relationship between protein structure and function, revealing biological functions and regulatory mechanisms within cells but also helps uncover disease-related protein abnormalities and guides drug design, providing valuable insights into disease mechanisms and facilitating drug development. Numerous studies, using both experimental and computational approaches, have explored the impact of mutations on PPIs.

The change of binding affinity $\Delta\Delta G$ upon mutations is calculated by the difference of the binding affinity from the wild-type protein to its mutant-type. Experimentally, Surface Plasmon Resonance (SPR) [70], Isothermal Titration Calorimetry (ITC) [71], and Fluorescence Polarization (FP) [72] are commonly used biophysical techniques to measure changes in binding affinity. However, these experimental methods are often expensive, time-consuming, and labor-intensive. Computational methods can be reasonable as alternatives.

Early computational predictors, such as FoldX [73], Rosetta [74], BeAtMuSiC [75], and BindProfX [76], are developed using physical energies or statistical potential-based functions to estimate the $\Delta\Delta G$. With the growth of machine learning and the availability of large-scale mutation experimental resources, some methods now incorporate machine learning-based classifiers to enhance prediction accuracy by integrating protein energy information, evolutionary information, and other relevant physicochemical features. For instance, iSEE [77] is a random forest-based predictor that uses a total of 31 features to predict binding affinity changes upon single-point mutations. mCSM-PPI2 [78] combines specific graph-based structural signatures, evolutionary information, complex network metrics, and energetic terms, utilizing the ExtraTrees algorithm for predictions. SAAMBE-3D [79] and SAAMBE-SEQ [80] employ structural features and sequence-based features, respectively, in conjunction with machine learning methods for prediction.

In recent years, deep learning-based methods have achieved remarkable progress in computer vision, natural language processing, and textual reasoning [81, 82], while also being widely adopted in bioinformatics for analyzing biological and medical data. TopNetTree [83] integrates convolutional neural networks and gradient-boosting trees to deal with the algebraic topology-based topological representation. PerSpect-EL [84] introduces persistent spectral-based PPI representation and featurization to simplify the structural representation of complexes. It utilizes the

ensemble learning method that stacks convolutional neural networks and gradient boosting trees for prediction. Graph neural networks (GNNs) have demonstrated powerful fitting and predictive capabilities in various fields [85]. Leveraging GNNs to explore complex graphs and extract effective structural features, several methods have been developed to predict changes in binding free energy, achieving good performance. DGCddG [86] employs multi-layer graph convolution to derive deep, contextualized representations for residues within the protein complex structure. These representations are then processed by a multi-layer perceptron to fit the mined mutation site channels to the binding affinity. MpbPPI [87] is a multi-task pre-training-based geometric equivariance-preserving framework, which encodes effective geometric representations of protein-protein complexes through the pre-training model and decodes them using gradient boosting tree-based decoder to predict the PPI $\Delta\Delta G$.

Although recent approaches have shown the strengths of deep learning models for improving PPI $\Delta\Delta G$ predictions, e.g., DGCddG and MpbPPI leverage GNNs to encode discriminative features for representing protein complexes, there are several aspects for further exploration to enhance both the predictive ability and the method pipeline. Particularly, heterogeneous features represent different properties of protein complexes, yet current works rely on tree-based decoders to produce the mixture of neural features and handcraft/statistical descriptors, which often model simple feature-level cross-relationships. This leaves sequence information and the inherent contextual relationships between mutations and neighboring residues under-explored. Furthermore, despite that deep learning techniques like self-attention mechanisms have been widely proven effective for incorporating context cues of objects, the approaches designed to harmonize and integrate features from multi-sources remain a technological gap that needs to be addressed.

To resolve this obstacle, in this work, we devise a novel self-attention-inspired fusion model, namely, Context-aware Selective Fusion (CSF), to enable the effective integration of informative multi-source features to improve the capability of the methods for PPI $\Delta\Delta G$ prediction. By leveraging CSF, we propose a new deep learning architecture that adaptively learns to integrate multi-source GNN-encoded features (including deep neural features based on sequence, structural, and evolutionary features) to achieve accurate binding affinity change predictions, referred to as SFM-Net. Specifically, we introduce a simple yet effective graph construction for deep learning methods based on the protein structural information. To better

learn complete geometric regulations for the wild-type (WT) and mutant-type (MT) protein complexes, the changes in properties from WT to MT are first modeled by the difference components which are constructed by mapping the differences between the GNN features of WT and MT using learnable operations. To stabilize the representation learning, we merge the difference components with summation components that model the low-frequency feature cues as the learning background of common properties.

Next, to obtain informative representations for accurate binding affinity change predictions through CSF, we first generate multiple condensed features by applying three different pooling strategies on the multi-source GNN encoded features, where we introduce the new relevant-site pooling. As the key idea for CSF, in contrast to existing self-attention mechanisms for spatial information aggregation that leverage all the inputs symmetrically to generate the query, key, and value maps, we model an oriented query only with the center- and relevant-site-pooled features to induce a selective fusion driven by the feature context. This idea originates from the task-specific assumption that the importance of heterogeneous features can vary significantly when representing a specific residue. Moreover, to explore meaningful sample-level dependencies, the CSF module further incorporates the inter-sample context to the compact features generated by the intra-sample fusion through balanced attention processing.

As mentioned above, we propose a novel self-attention-inspired fusion model that systematically explores and integrates diverse protein features, addressing a critical gap in protein feature modeling. Additionally, we introduce innovative summation and difference components, designed to effectively extract commonalities and differences between the features of wild-type and mutant-type protein complexes, enabling deeper insights into protein behavior. We comprehensively validate our newly proposed method, termed SFM-Net, on three benchmark datasets, by comparing SFM-Net with the State-of-The-Art (SOTA) approaches. Benchmarking results demonstrate the superior effectiveness of our SFM-Net for predicting protein-protein binding affinity changes caused by mutations. We believe this work not only advances the binding affinity changes prediction but also provides a novel perspective on generic protein feature modeling which may have broad downstream applications in the related areas.

Table 4.1: Statistical summary of the datasets used in this study

Dataset	#Num. mutations	#Num. complexes	$\Delta\Delta G_{WT-MT} > 0$	$\Delta\Delta G_{WT-MT} = 0$	$\Delta\Delta G_{WT-MT} < 0$
S4169	4169	319	901	142	3126
S1131	1131	112	842	14	275
S645	645	29	480	45	120

* “#Num.” represents “Number of.”

4.2 Materials

4.2.1 Datasets

In this study, we adopted three publicly available benchmark datasets, widely used in previous works, to assess the capability of SFM-Net for predicting PPI $\Delta\Delta G$. These datasets contain single-point mutations. Dataset S4169 was constructed by Rodrigues et al. [78], which was originally derived from the SKEMPI 2.0 database [19] by collecting the single-point mutations with available experimental crystal structures of the wild-type. Dataset S1131 was first used by Xiong et al. [76], constructed by filtering the non-redundant interface single-point mutations from the SKEMPI database [88]. Dataset S645, derived from the antibody binding (AB-Bind) database [89], was first used by Pires and Ascher [90] to predict antibody-antigen affinity changes. A statistical summary of the datasets is provided in Table 4.1.

4.2.2 Protein Representation

In this study, we represented the protein mutation as an undirected graph $G(V, E)$, where the center node denotes the mutation site, the remaining nodes represent surrounding residues, and edges indicate interactions between the mutation site and other residues. To learn a meaningful representation, we modeled all amino acid residues as nodes, enabling the network to effectively leverage the full information encoded in the sequence. To reduce complexity, we limited edge construction to residues within 10 Å of the mutated site, which is a widely adopted threshold in protein-protein interaction analysis. Accordingly, edges were established between the mutated site and each node located within this distance threshold. It is motivated by the observation that residues close to the mutation site have a more significant influence on its structure and conformation. Additionally, this strategy reduces computational complexity by focusing on the most relevant interactions.

Furthermore, it ensures that the GNN updates emphasize the central node throughout the graph, effectively preserving and highlighting the information most relevant to the mutation. To construct the graph for the wild-type, we used the raw PDB files directly obtained from the Protein Data Bank (PDB) [91], leveraging their experimentally determined three-dimensional coordinates. For the mutant-type, we employed FoldX [73] to compute the structural changes induced by the mutation, generating the corresponding PDB files. These mutant PDB files were then used to construct the graph for the mutated state. This ensures that both wild-type and mutant graphs are built from reliable structural data, accurately reflecting their respective spatial configurations and mutation-induced changes. As a result, the initial graph representation is defined by the node feature matrix and the node adjacency matrix.

Node features

Node features were constructed using protein sequence, structural, and evolutionary information. ProtT5, a pre-trained protein language model trained on over 390 billion amino acids, can be used to generate complex dependencies and patterns of protein sequences [92]. We employed ProtT5 to encode informative protein sequence embeddings for the sequences with the wild-type and mutation, respectively. Additionally, common features, DSSP and PSSM, were applied to the wild-type complexes to introduce secondary structural information and evolutionary information. Overall, the node matrix of the wild-type complex includes three sets of features, DSSP, PSSM, and the protein representation extracted by ProtT5. For the mutated complex, the node matrix is represented solely by the protein embedding extracted by ProtT5.

Adjacency matrix

In this study, we propose a simple yet effective graph construction strategy for each sample by appointing the mutation site as the center node. Edge features represented the spatial relationships between the center node and the neighboring nodes near the mutation site. The PDB files for mutant complexes were generated using FoldX [73] based on raw PDB files. According to the PDB file, we calculated the Euclidean distances between the mutation site and all remaining residues by using the coordinate of the $C\alpha$ atom of each residue to generate the distance map. The adjacency matrix was constructed by comparing the values in the distance map to

a predefined cutoff distance. If the cutoff value exceeded the value, it was set to 1, denoting an edge between the center node and the specific nodes. Otherwise, it was set to 0, indicating no edge.

4.3 Methods

4.3.1 SFM-Net Architecture

The overall architecture of the Selective Fusion of Multi-way protein feature Network (**SFM-Net**), as illustrated in Figure 4.1, is developed to effectively learn the unified representations of the binding affinity change by exploiting and selectively fusing informative cues from different sources of raw protein features.

For each graph centered on the mutation site according to the adjacency matrix, SFM-Net encodes four different ways of protein features through GNN blocks of similar structures in parallel. The changes from the wild-type to mutant-type protein complexes are first modeled by their feature-level differences (i.e., the difference components) learned from the corresponding embedding features, and mixed with the summation components which incorporate the low-frequency properties to stabilize the learning process. Furthermore, PSSM and DSSP, which reveal highly conserved sites in proteins and describe the secondary structure of a protein, are processed by independent GNN blocks and incorporated into the embedding features to provide complementary structural dependencies of nodes. To jointly leverage the multi-way GNN features diverse in representational properties, we introduce the Context-aware Selective Fusion model. CSF then employs three different kinds of pooling operations at the top of each path, each of which aggregates features of nodes of different ranges centered on the mutation site. This constructs nine different compact features for fusion. Next, CSF learns to fuse the multi-way compact features into a joint representation for each sample driven by rich dual-scope context cues. This is realized by introducing a novel two-tier self-attention-based fusion scheme that encodes and shares both intra-sample and inter-sample contextual information. Specifically, first, during intra-sample processing, different features are emphasized or suppressed by dynamically assigning different weights and aggregated at the top of the self-attention layer. This process learns the fused feature for each sample. Then, the unified representations are generated by an inter-sample attention block applied to model cross-sample dependencies. Binding affinity change

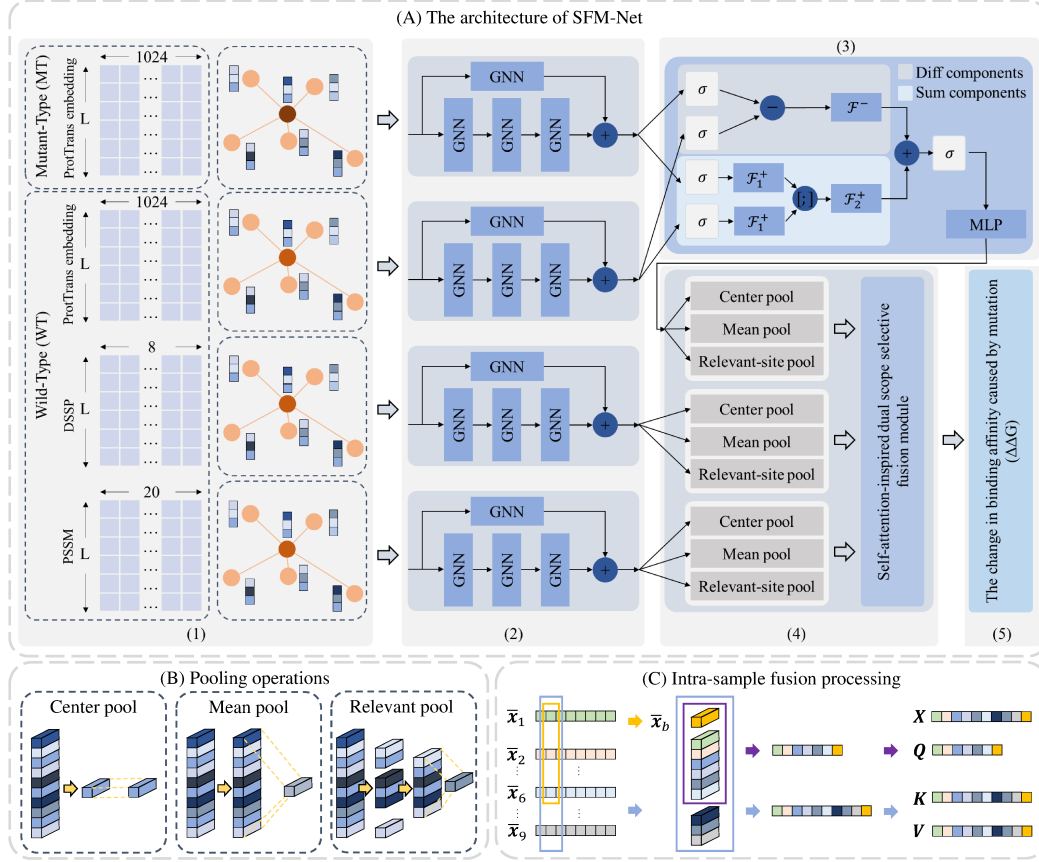


Figure 4.1: Overall framework of SFM-Net. (A) The architecture of SFM-Net, comprising (1) input of the feature encoding and the graph construction; (2) four parallel GNN-based blocks; (3) summation and difference components; (4) key ingredients that compose the CSF model, i.e., pooling operations and self-attention-inspired double-scope selective fusion module; and (5) multi-layer perceptron. The key ingredients of CSF encompass: (B) Three different kinds of pooling operations, i.e., center pool, mean pool, and relevant-site pool; (C) The intra-sample fusion processing, $\{\bar{x}_i | i = 1, 2, \dots, 9\}$ refers to the nine pooled feature vectors, while the background feature $\bar{x}_b$ is calculated by $\frac{1}{6} \sum_{i=1}^6 \bar{x}_i$.

predictions are obtained by processing the unified representations with a multi-layer perceptron.

4.3.2 Graph Neural Network-Based Feature Encoder

Graph Neural Networks (GNNs) have been widely used to handle graph-structured data and model complex relationships and interdependencies between graph entities. In this study, for each mutation sample, we construct two graphs centered on the mutation site: one based on the wild-type (WT) protein complex and the other on the mutant-type (MT) protein complex. After preparing the four groups of node features and two types of graphs, four parallel GNN-based blocks are conducive to aggregating and updating the four different representation information. Specifically,

each block consists of three GNN layers and a residue connection, which enhances the ability to extract deep and effective representations for subsequent analysis and learning and mitigates gradient-related issues simultaneously. The number of GNN layers in each block is set to three, following common practice. Since the input features differ in properties and represent distinct aspects of the data, four parallel encoder blocks are employed to extract and preserve their unique characteristics before integration. The four obtained feature representations are followed by different operations.

4.3.3 Summation and Difference Components

After leveraging the GNN-based feature encoder to encode and update sequence information on the wild-type and mutant graphs, the key issue lies in the effective integration of these two representations to predict the changes in binding affinity. The change in binding affinity caused by AA mutations represents the difference in binding affinity between the wild-type and mutant-type protein complexes. In light of this, we design the summation and difference components of two sets of representations, followed by non-linear mappings through a Multi-Layer Perceptron (MLP), aiming to generate more intricate and enriched representations. Specifically, for the given GNN-encoded features of a WT and a MT, denoted by $\mathbf{X}^w, \mathbf{X}^m \in \mathbb{R}^{L \times C}$, respectively, where L and C denote the numbers of residues and channels of features, respectively, the changes in property from WT to MT can be first modeled by the difference components $\mathbf{X}^- \in \mathbb{R}^{L \times C}$ as follows:

$$\mathbf{X}^- = \mathcal{F}^- (\sigma(\mathbf{X}^m) - \sigma(\mathbf{X}^w)) , \quad (4.1)$$

where σ denotes the popular activation function SiLU [93]; $\mathcal{F}$ denotes a linear layer (marked with a superscript ‘ $-$ ’, representing a specific linear layer operation, distinct from its counterparts). In contrast, the learning on pure difference information (i.e., high-frequency cues) is possibly sensitive to interference (e.g., noise and outliers). To address this issue, we model the low-frequency cues using the summation components (denoted by $\mathbf{X}^+ \in \mathbb{R}^{L \times C}$), as the learning background of common properties of embedding features. This compensates for the learning robustness, which is modeled by

$$\mathbf{X}^+ = \mathcal{F}_2^+ \left(\left[\mathcal{F}_1^+ (\sigma(\mathbf{X}^m)); \mathcal{F}_1^+ (\sigma(\mathbf{X}^w)) \right]_C \right) , \quad (4.2)$$

where $\mathcal{F}^+$ denotes the linear layer operation in the summation components, with subscript ‘1’ and ‘2’ representing the first and second linear layers, respectively; $[\cdot; \cdot]_C$ denotes a concatenation by channels. The difference and summation components are then mixed after layer-normalization and inputted into the MLP module to learn the informative feature relationships about the binding affinity change.

4.3.4 CSF: Context-Aware Selective Fusion Module

Pooling operations

To capture effective intra- and inter-sample protein information while reducing feature dimensionality and mitigating overfitting, CSF refines three range-specified neighborhood-to-center relationships for each GNN learning path by different kinds of pooling operations, i.e., the center pooling, global mean pooling, and relevant-site pooling. Specifically, the center pooling is applied to extract the refined information of multi-way features. Then, the mean pooling calculates the average feature to represent each sample by its global information over different residues. In particular, relevant-site pooling is a new operation we propose to relate the center node with the selected important neighborhoods designated by useful manual prior knowledge. In CSF, we select nodes that are within Euclidean distance of $10\text{\AA}$ from the center node as the relevant nodes. That is, suppose $X \in \mathbb{R}^{L \times C}$ is a given input feature map representing a complex. Then, for relevant-site pooling, we define

$$\bar{x}^r = \text{mean}_{x \in \{x | D(x, x_0) \leq k\}}(x), \quad (4.3)$$

where $\bar{x}^r \in \mathbb{R}^C$ is the relevant pooled feature (a vector); $x_0 \in \mathbb{R}^C$ denotes the GNN feature vector representing the mutation site (i.e., the center node); $D(\cdot, \cdot)$ means Euclidean distance of two vectors and k is the threshold for feature selection. That is, $\{x | D(x, x_0) \leq k\}$ denotes the set of GNN features $x \in \mathbb{R}^C$ associated with the residues that satisfy the threshold criterion.

In CSF, we apply center pooling, global mean pooling, and relevant-site pooling operations to each of the ProtT5, DSSP, and PSSM feature paths, resulting in nine different compact features for fusion, each of which focuses on a specific range of statistical cues of its source feature. These oriented pooling operations extract features of different meanings and serve as the pre-processing for the selective feature fusion driven by intra- and inter-sample contextual relationships.

Self-attention-inspired dual-scope feature fusion

Traditional self-attention mechanisms induce discriminative weighted aggregation of homologous feature vectors on the input feature map (i.e., the set of a group of feature vectors), enabling the model to focus on salient features while integrating long-range context from the entire input sequence [37, 94]. In the calculations of a self-attention, for a given input feature map $\mathbf{X} \in \mathbb{R}^{L \times C}$, the Query ($\mathbf{Q}$), Key ($\mathbf{K}$), and Value ($\mathbf{V}$) maps are typically derived through linear projections of $\mathbf{X}$. It first models the weight matrix by the inner-product-based query-to-key similarity measure, i.e., each input feature vector $\mathbf{x} \in \mathbb{R}^C$ on $\mathbf{X}$ is leveraged in a cyclic symmetric manner for weighted-aggregation. Then it merges different value vectors (i.e., $\mathbf{v} \in \mathbb{R}^C$ on $\mathbf{V}$) into a unified vector $\mathbf{a} \in \mathbb{R}^C$ to re-represent each of the original feature vectors on the feature map, respectively. The original self-attention can be calculated by the following formula:

$$\mathbf{A} = \text{softmax} \left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}} \right) \mathbf{V}, \quad (4.4)$$

where $\mathbf{A} \in \mathbb{R}^{L \times C}$ denotes the attended feature map of $\mathbf{X}$; $d = C/G$ denotes the number of channels for each channel group (i.e., head), where we set $d = 32$ by default.

In CSF, to facilitate the binding affinity change prediction, we devise the novel dual-scope feature fusion scheme inspired by self-attention. It learns to select important information within intra-sample multi-way features that vary from representational properties and merge them into a compact representation for each sample. The cross-sample relationships are also explored to reinforce the representations for predictions.

First, we generalize the original self-attention mechanism for spatial homologous feature aggregation to heterogeneous feature fusion by introducing a biased query-key similarity measure with selective queries. This is built on the assumption that heterogeneous features can be dramatically diverse in importance for representing a certain residue. More specifically, as shown in Figure 4.1C, in the *intra-sample* processing, the center and relevant-site pooled feature vectors from the 3 different source features (i.e., ProtT5, DSSP, and PSSM) total 6 in number, we select these pooled feature vectors $\{\bar{\mathbf{x}}_i \in \mathbb{R}^C | i = 1, 2, \dots, 6\}$ to construct the query map with a

linear projection. This helps select and incorporate useful properties within the homologous features to each integrated representation for the corresponding residue, which leverages a prior knowledge that information extracted from the mutation site and the residues highly relevant to it is likely more critical to the prediction of binding affinity change. For constructing key and value maps, we use all 9 pooled features, where the values will be emphasized and suppressed by the weights modeled by the proposed selective query-to-key similarity measure. Note that to stabilize the weight computation, we also produce an averaged feature vector of the 6 center and relevant-site pooled feature vectors and add it to the source inputs of query, key, and value maps, respectively, i.e., here, $\mathbf{Q} \in \mathbb{R}^{7 \times C}$ and $\mathbf{K}, \mathbf{V} \in \mathbb{R}^{10 \times C}$, respectively. Finally, we apply a reductional post-projection on the attended feature to obtain the compact fused feature $\mathbf{u} \in \mathbb{R}^C$, i.e.,

$$\mathbf{u} = \frac{1}{6} \sum_{i=1}^6 \bar{\mathbf{x}}_i + \mathcal{F}_{\downarrow}(\mathbf{A}), \quad (4.5)$$

where $\mathcal{F}_{\downarrow}$ denotes a linear projection that inputs an attended feature map of size $7 \times C$ and outputs a compact feature vector of size C , $\frac{1}{6} \sum_{i=1}^6 \bar{\mathbf{x}}_i$ denotes the background feature $\bar{\mathbf{x}}_b$ shown in Figure 4.1.

Second, we leverage the self-attention mechanism to model *inter-sample* dependencies. Specifically, after generating the fused features $\mathbf{U} \in \mathbb{R}^{B \times C}$ that integrate intra-sample contextual dependencies, we apply the self-attention to incorporate cross-sample relationship cues within each mini-batch (here, B denotes the number of samples in a mini-batch). More specifically, CSF generates the cross-sample $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{B \times C}$ maps on the fused features $\mathbf{U}$ by three ways of linear projections, respectively, and calculate the self-attention over the $\mathbf{Q}, \mathbf{K}$, and $\mathbf{V}$ maps to produce the dual-attended feature maps. Then, the final representations for the binding affinity change predictions are learned through the MLP.

The attention mechanism in our network is designed to dynamically focus on different protein-level cues depending on the input sample. Specifically, different feature vectors are assigned attention weights that reflect their relative importance in representing the sample. For example, a sample may exhibit a stronger dependence on structural information, while another sample may be more sensitive to the sequential order of residues. Accordingly, the attention weights for the first sample are expected to emphasize features that capture structural differences, whereas the attention for the second sample would highlight features associated with sequence

variation. In this way, the network adaptively attends to the most informative aspects of the input, and the targets of attention are reflected in the learned attention weights during inference.

4.3.5 Evaluation Metrics

Consistent with previous studies, we use the mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and Pearson’s correlation coefficient (PCC) to evaluate the predictive performance of SFM-Net. The mathematical expressions of these evaluation metrics are provided as follows:

$$\text{MSE} = \frac{\sum_{i=1}^{|\mathcal{D}|} (y_i - \hat{y}_i)^2}{|\mathcal{D}|}, \quad (4.6)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{|\mathcal{D}|} (y_i - \hat{y}_i)^2}{|\mathcal{D}|}}, \quad (4.7)$$

$$\text{MAE} = \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} |y_i - \hat{y}_i|, \quad (4.8)$$

$$\text{PCC} = \frac{\sum_{i=1}^{|\mathcal{D}|} (\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{|\mathcal{D}|} (\hat{y}_i - \bar{\hat{y}})^2 \sum_{i=1}^{|\mathcal{D}|} (y_i - \bar{y})^2}}, \quad (4.9)$$

where $\mathcal{D}$ denotes the dataset, y_i and $\hat{y}_i$ represent the experimental and predicted binding affinity values for the i -th sample, respectively.

4.4 Results

In this section, we conducted experiments on three different benchmark datasets S4169, S1131, and S645 to assess the performance of our method, SFM-Net.

4.4.1 Performance Evaluation on Cross-Validation Test

Mutation-level cross-validation

Using the three benchmark datasets, we compared SFM-Net with six relevant popular/SOTA methods, including MpbPPI [87], DGCddG [86], PerSpect-EL [84], GeoPPI [95], TopNetTree [83], and FoldX [73], by conducting 10-fold cross-validation experiments. For a specific dataset, in each of the 10 folds, all mutation samples

were divided into ten subsets, with the former nine used for training and the remaining one for validation. To ensure fair comparisons, we used the same dataset divisions of the 10 folds as MpbPPI. Following the common principle in previous works, Pearson’s correlation coefficient was used as the primary evaluation metric. In particular, as the predicted outputs of neural networks tend to exhibit a broader range without specialized range constraints applied, magnitude-related evaluation metrics (e.g., MSE, RMSE, or MAE) may fail to adequately reflect the practical ability of purely neural network-based methods compared to approaches employing conventional machine learning predictors. Developing specialized range constraints for neural network outputs is beyond the scope of this work. Nevertheless, for comprehensiveness, the results of MSE, RMSE, and MAE are provided as supplementary references.

We report the comparative results in Table 4.2, where the results of the competing methods are directly quoted from MpbPPI. The best performance in the table is highlighted in bold. Our main observations are 4-fold: (1) Our SFM-Net achieves Pearson’s correlation coefficient values of 0.797, 0.881, and 0.624 for S4169, S1131, and S645 datasets, respectively; (2) Compared with the popular and SOTA competing methods, SFM-Net demonstrates significant improvements in prediction accuracy by comprehensively taking into account the results achieved on all the evaluated benchmark datasets. In particular, SFM-Net outperforms the SOTA method, i.e. MpbPPI by circa 1.58% on the evaluation on dataset S1131; (3) Note that the gain yielded by SFM-Net on the S4169 dataset is not marginal, as predictions on the S4169 dataset tend to be saturated. That is, we observed nearly 100% predictions on the training set and this fitting gap between the training and validation sets could not be mitigated by increasing the dropout rate or regularization. (4) On the S645 dataset, SFM-Net ranks the second best but outperformed the SOTA MpbPPI, introducing circa 0.9% improvement of Pearson’s correlation coefficient. It is worth noting that before our SFM-Net was proposed, MpbPPI achieved the best performance on S4169 and S1131 datasets but also the second-best result on the S645 dataset. In this work, our SFM-Net demonstrates SOTA improvements over MpbPPI by comprehensively considering the predictions on different benchmarks. These validate the effectiveness of our new method for predicting binding affinity changes. Additionally, the results of our method on other metrics are provided in Table 4.7.

Table 4.2: Performance comparison of SFM-Net and the existing methods on three benchmark datasets in terms of Pearson’s correlation coefficient

Method	S4169	S1131	S645
SFM-Net	0.797	0.881	0.624
MpbPPI	0.795	0.865	0.615
DGCddG	—	0.848	—
PerSpect-EL	—	0.853	0.590
GeoPPI	0.780	0.850	0.650
TopNetTree	0.790	0.850	0.650
FoldX	0.326	0.423	0.271

* The best two results for each dataset are highlighted in **Bold** and **Gray**, respectively.

* The results of the competing methods are cited from MpbPPI [87].

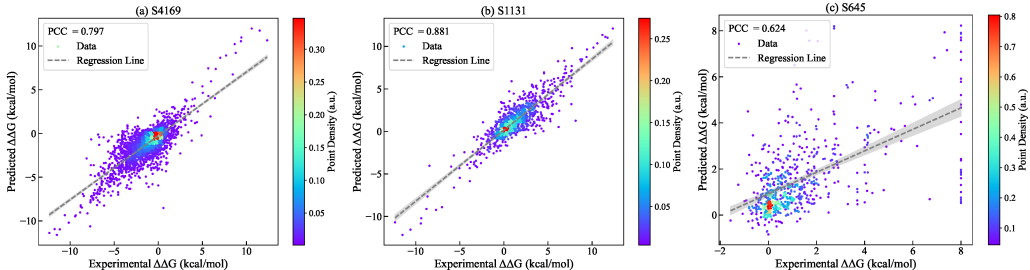


Figure 4.2: SFM-Net predicted $\Delta\Delta G$ against experimental $\Delta\Delta G$ on three datasets. Color represents point density, while the dashed line indicates the regression line.

In Figure 4.2, we also plotted the distribution of predicted $\Delta\Delta G$ against experimental $\Delta\Delta G$ on three datasets. It shows that the predictive model performs reasonably well on S4169 and S1131. As shown in Figure 4.2 (a), the majority of data points are clustered around the diagonal, suggesting good predictive performance for $\Delta\Delta G$ values in the central range ($\Delta\Delta G$ between -8 and 8). However, for the samples with very high experimental $\Delta\Delta G$ values ($\Delta\Delta G \geq 8$ kcal/mol), the points tend to scatter further from the regression line; for the samples with highly negative $\Delta\Delta G$ values ($\Delta\Delta G \leq -8$ kcal/mol), the points also show larger deviations from the regression line, although they are closer to the regression line compared to the $\Delta\Delta G \geq 8$ range. It indicates that prediction errors are larger in the extreme $\Delta\Delta G$ ranges compared to the moderate $\Delta\Delta G$ values. The potential reasons for deviations may come from several aspects. Insufficient data of the extreme $\Delta\Delta G$ values may be underrepresented in the training data, leading to reduced predictive accuracy. High $\Delta\Delta G$ values (indicating high destabilization) often result from significant structural or energetic changes, while extremely negative $\Delta\Delta G$ values reflect strong stabilization that typically arises from intricate binding or molecular

interactions. The extreme values of positive and negative intervals can be particularly challenging to predict from the modeling of protein interactions and structural information without sufficient data. In Figure 4.2 (b), the scatter plot exhibits a similar distribution pattern to Figure 4.2 (a). However, a notable difference is that data points in the $\Delta\Delta G \geq 8$ range are closer to the regression line compared to those in the $\Delta\Delta G \leq -8$ range. This difference may be introduced by the variation in sample distributions between the datasets: S4169 contains more negative samples than positive ones, whereas S1131 has a higher proportion of positive samples. Figure 4.2 (c) for dataset S645 shows a weaker alignment along the diagonal compared to Figures 4.2 (a) and (b). Most data points in the region of $\Delta\Delta G \geq 0$, particularly in the $\Delta\Delta G \geq 4$ range, scatter away from the regression line, showing higher variance compared to Figures (a) and (b). Unlike datasets S4169 and S1131, dataset S645 contains relatively fewer negative $\Delta\Delta G$ values. While predictions in this region are somewhat closer to the regression line, the reduced number of negative samples may limit the model’s ability to generalize well for this dataset.

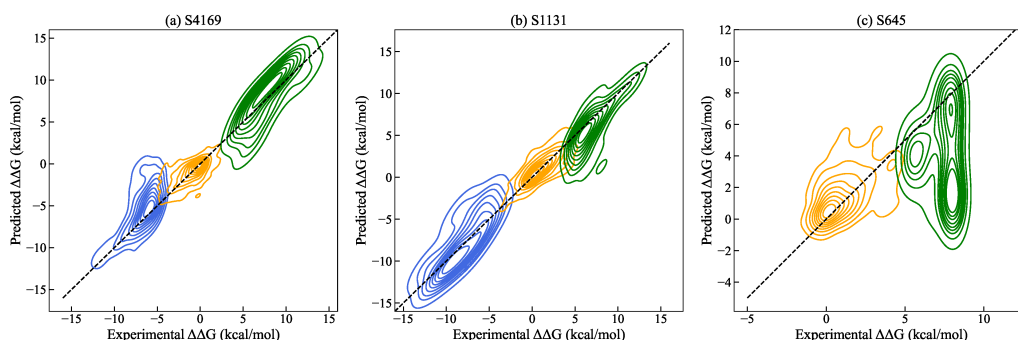


Figure 4.3: SFM-Net predicted $\Delta\Delta G$ against experimental $\Delta\Delta G$ across three distinct ranges on three datasets. Blue contours represent values where Experimental $\Delta\Delta G \leq -5$, orange contours represent values in the range $-5 < \text{Experimental } \Delta\Delta G < 5$, and green contours represent values where Experimental $\Delta\Delta G \geq 5$.

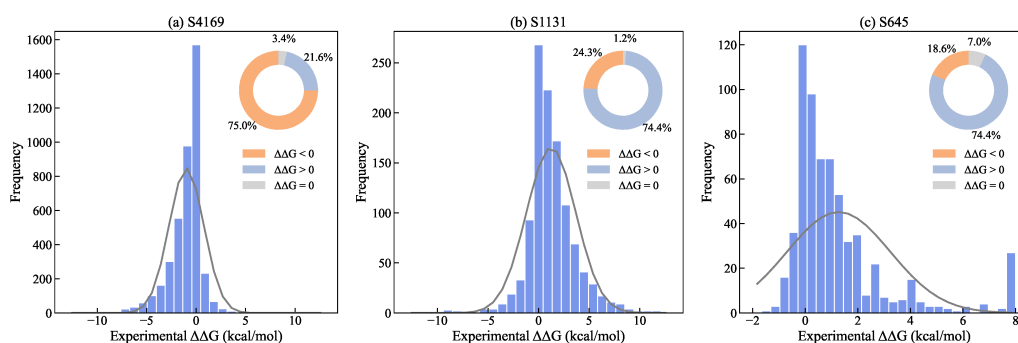
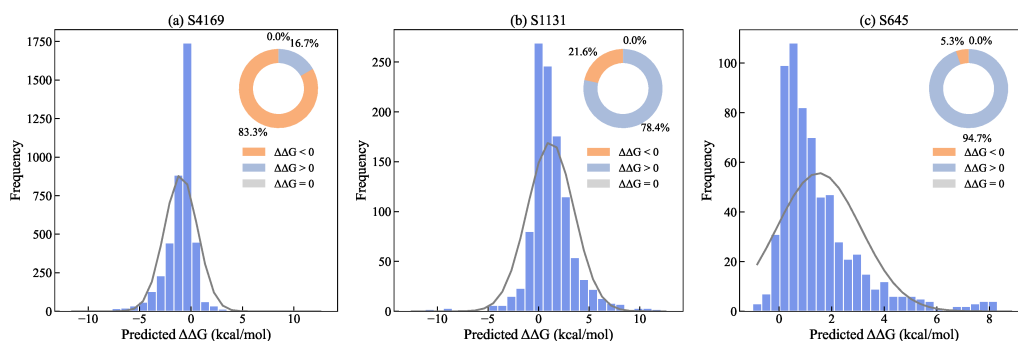


Figure 4.4: Distribution of experimental $\Delta\Delta G$ on three datasets.

Figure 4.5: Distribution of SFM-Net predicted $\Delta\Delta G$ on three datasets.

To further illustrate our model’s performance across the three datasets in both extreme and intermediate $\Delta\Delta G$ ranges, Figure 4.3 presents the relationship between experimental $\Delta\Delta G$ and predicted $\Delta\Delta G$. Data points are categorized into three distinct regions (≤ -5 , -5 to 5 , and ≥ 5) based on experimental $\Delta\Delta G$ values to show the model’s behavior across these ranges. The detailed distribution of experimental and predicted $\Delta\Delta G$ values across the three datasets is provided in Figure 4.4 and Figure 4.5, respectively. From these figures, the similarity in data distribution between the S4169 and S1131 datasets can be clearly observed, as well as the extreme imbalance in the data distribution of the S645 dataset. The distribution of experimental $\Delta\Delta G$ values across the training and test sets for each fold in the 10-fold cross-validation experiment for S4169, S1131, and S645 is shown in Figure 4.6, Figure 4.7, and Figure 4.8, respectively. In the S645 dataset, the histograms reveal a skewed distribution of experimental $\Delta\Delta G$ values, with a higher concentration of samples in the positive $\Delta\Delta G$ range, particularly including a large proportion of extremely high $\Delta\Delta G$ values. Conversely, the negative samples are fewer and mostly clustered near zero. This imbalance in the dataset may account for the poorer performance of our method and other approaches for this dataset, as such distributions are challenging for the model to handle effectively.

Table 4.3: Performance comparison of our model on interface and non-interface mutations in the S4169 and S645 datasets

Dataset	Mutation	PCC	RMSE	MAE	nRMSE
S4169	Interface Mutation	0.7997	1.1736	0.8144	0.6132
	Non-interface Mutation	0.5075	0.6981	0.4547	0.9710
S645	Interface Mutation	0.5424	1.8621	1.2300	0.8835
	Non-interface Mutation	0.3938	0.9279	0.6740	1.3585

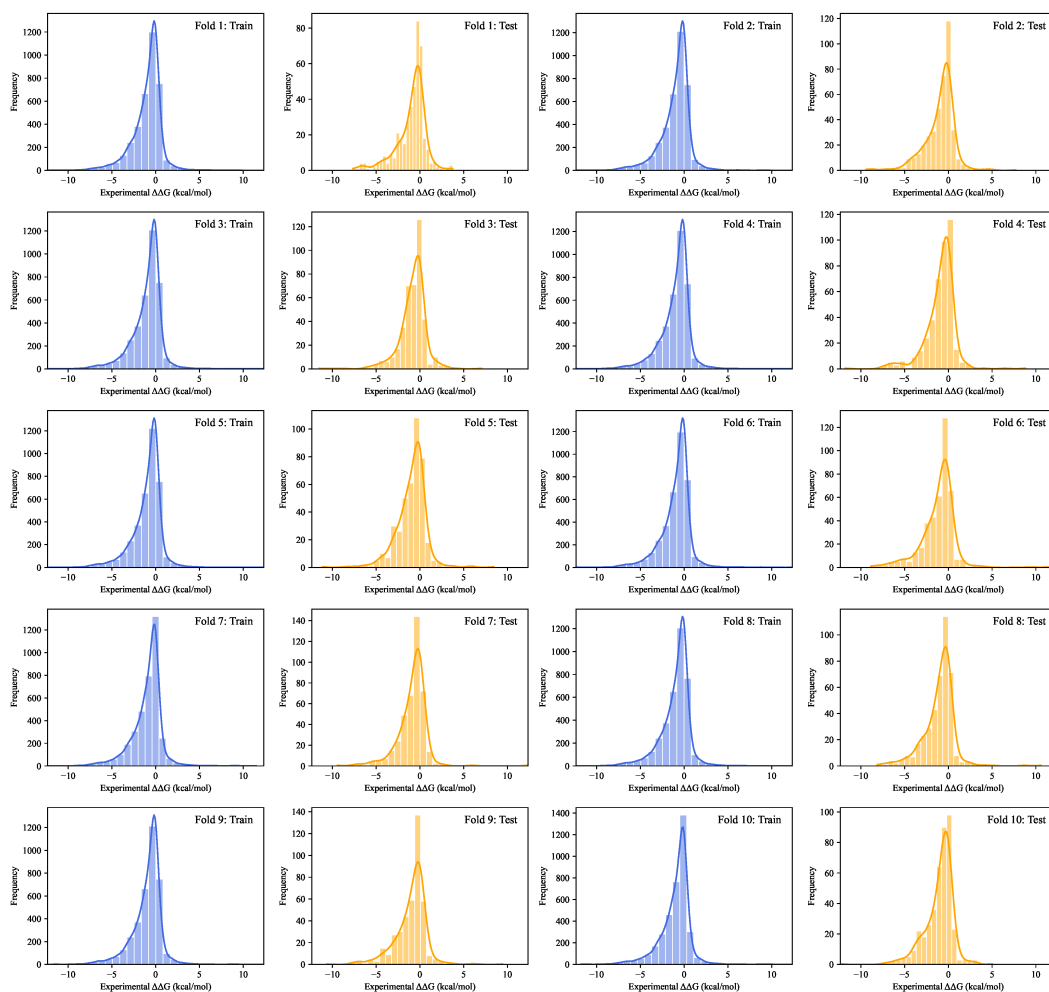


Figure 4.6: Distribution of experimental $\Delta\Delta G$ values across the training and test sets for each fold in the 10-fold cross-validation experiment on S4169.

We further compared the performance on mutations occurring at the PPI interface and non-interface regions. Here, the interface residues are defined as those with any atom within 5 Å of any non-hydrogen atom of the protein partner. Figure 4.9 presents the comparison of experimental and predicted $\Delta\Delta G$ distributions for interface and non-interface mutations across the three datasets. It can be observed that the $\Delta\Delta G$ range for interface mutations is broader in the S4169 and S645 datasets. Furthermore, we compared the performance of our model on interface and non-interface mutations in these two datasets. Detailed metric results are provided in Table 4.3.

Compared to non-interface mutations, the models achieve higher PCC values and perform better on interface mutations. Although the error metrics (RMSE, MAE) appear smaller for non-interface mutations, this is likely due to the narrower

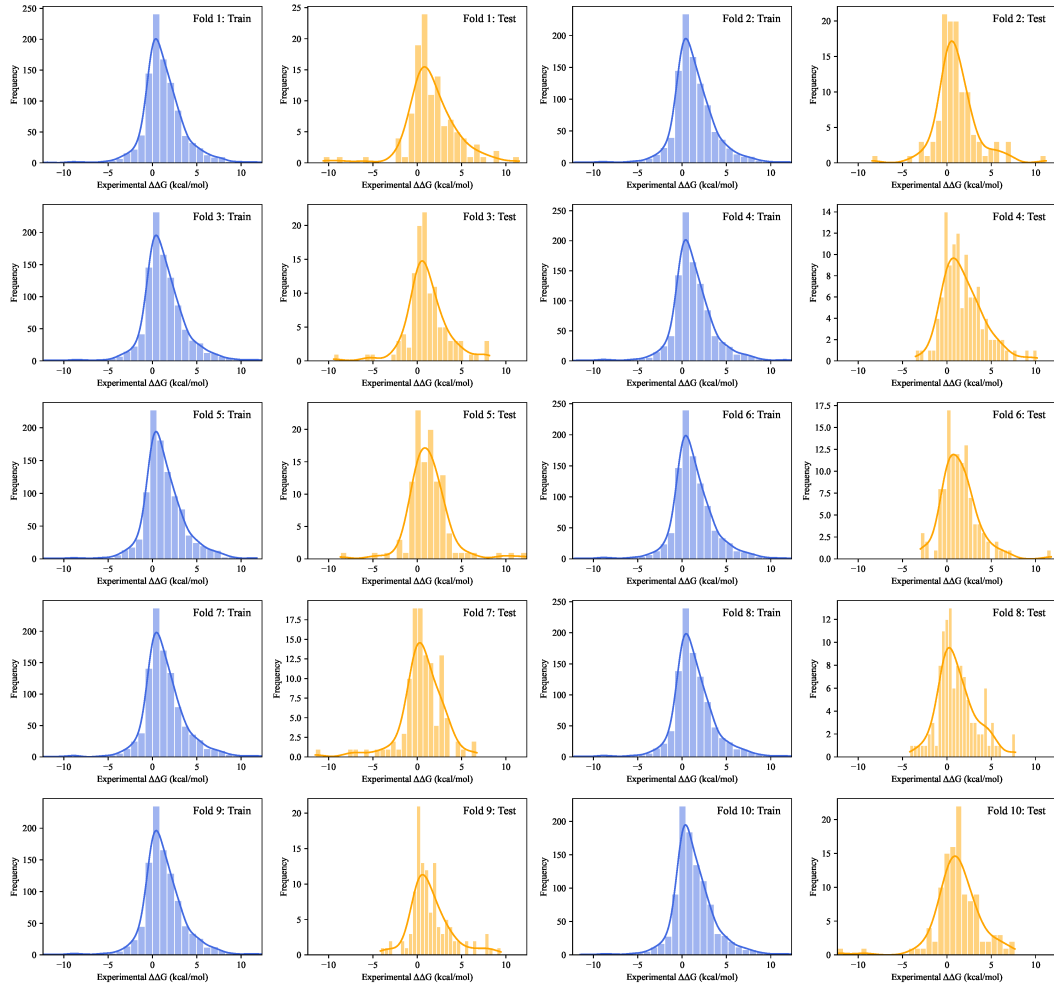


Figure 4.7: Distribution of experimental $\Delta\Delta G$ values across the training and test sets for each fold in the 10-fold cross-validation experiment on S1131.

value range and smaller sample size. To further account for differences in value ranges and sample sizes, we introduce the normalized RMSE (nRMSE) as an additional evaluation metric [96, 97, 98]:

$$\text{nRMSE} = \frac{\text{RMSE}}{\sigma}, \quad (4.10)$$

where σ denotes the standard deviation of all the experimental binding affinity values (i.e., the ground truth values). This normalization allows for a more consistent comparison of relative errors across different mutation categories by adjusting for variations in value dispersion. From the results, we observe that non-interface mutations have higher nRMSE values, despite having lower absolute RMSE and MAE.

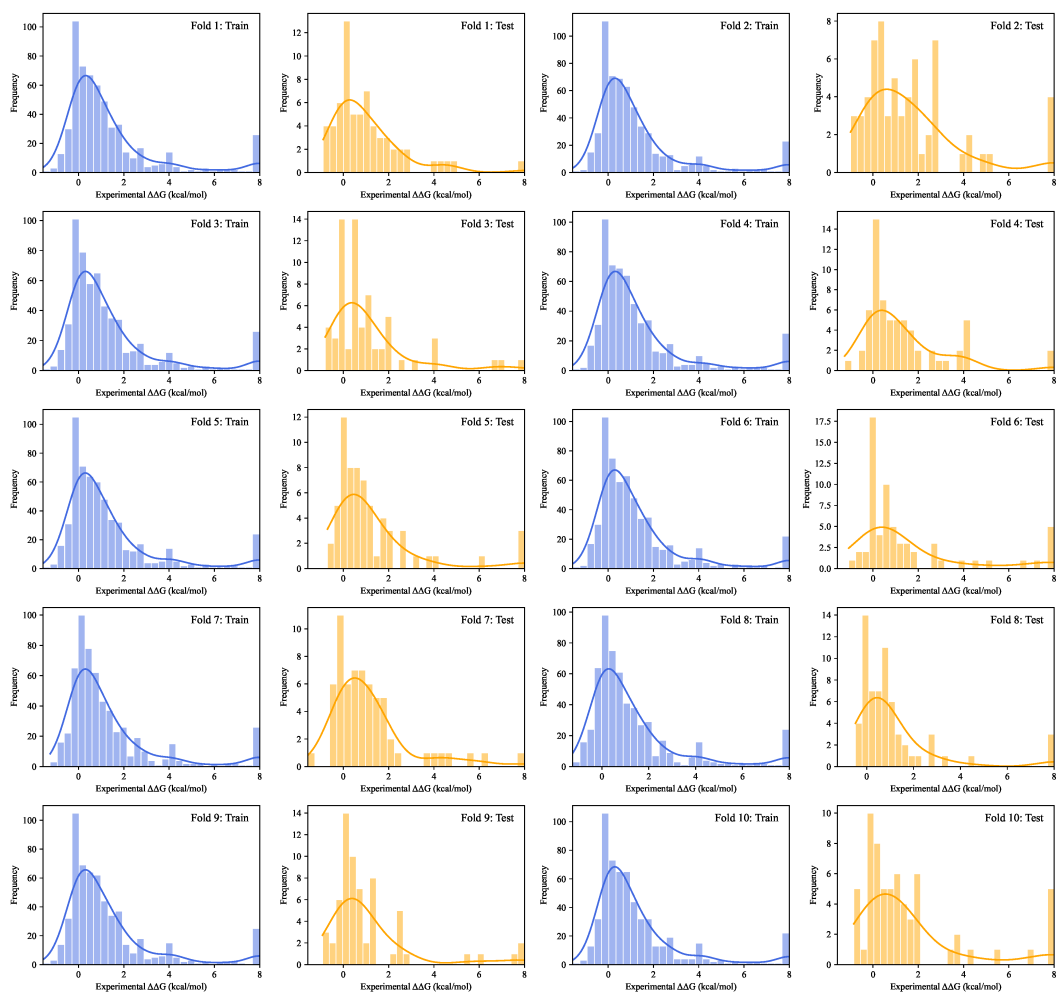


Figure 4.8: Distribution of experimental $\Delta\Delta G$ values across the training and test sets for each fold in the 10-fold cross-validation experiment on S645.

Protein-level cross-validation

To further evaluate the generalization capability of SFM-Net in a more rigorous test scenario, we conducted the 5-fold protein-level cross-validation experiments, ensuring that the same WT protein-protein complex types were kept within the same fold. This setup allows us to assess the performance of our model when the test set consists only of complexes excluded from the training set. To ensure fair comparisons, we used the same complex-level 5-fold dataset divisions as MpbPPI. We report the comparative results of SFM-Net and other methods in terms of Pearson’s correlation coefficient in Figure 4.10, where the results of other methods are directly quoted from MpbPPI. Under this rigorous evaluation setting, our SFM-Net achieves Pearson’s correlation coefficient values of 0.598, 0.727, and 0.428 for S4169, S1131, and S645 datasets, respectively, outperforming MpbPPI and other

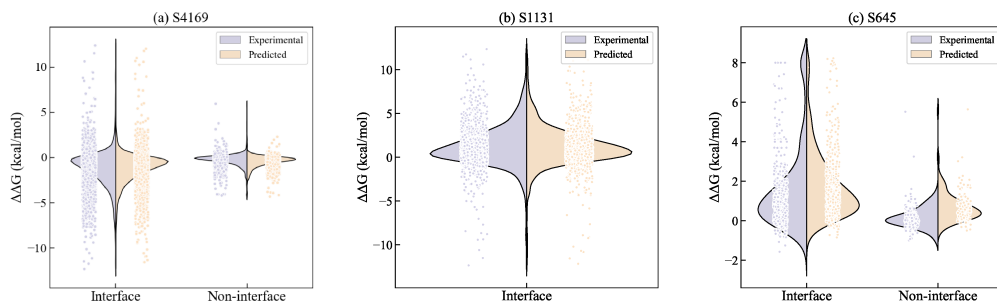


Figure 4.9: Comparison of experimental and predicted $\Delta\Delta G$ distributions for interface and non-interface mutations on three datasets.

methods. Additionally, results in terms of RMSE and MAE are provided in Table 4.4. These results demonstrate that our method is robust and achieves superior performance even under the protein-level cross-validation experiments, further validating the effectiveness of our approach.

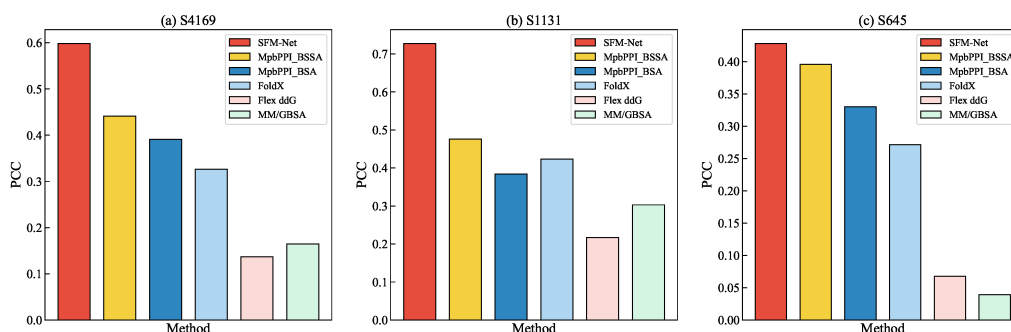


Figure 4.10: Performance comparison of SFM-Net and other competing methods in terms of Pearson's correlation coefficient under the WT protein-protein complex-based cross-validations.

To evaluate the generalizability of our model to multiple mutations, we conducted a cross-validation experiment on the M1101 dataset, a more challenging benchmark dataset containing multiple mutations. The M1101 dataset comprises 1101 samples, including all entries from the antibody binding (AB-Bind) database [89]. In the complex-level cross-validation experiment, same protein complexes were grouped into the same fold to ensure that no identical complex types existed between the training and test sets. This setup enables a rigorous evaluation of the model's capability to generalize to unseen wild-type (WT) protein-protein complexes.

For single-point mutations, the constructed graph was centered on the mutation site. In the case of multiple-point mutations, we extended this approach by centering the graph on each mutation site and constructing the edges between these

Table 4.4: The RMSE and MAE results for SFM-Net and other involved methods under the WT protein-protein complex-based cross-validations

S645	RMSE	MAE
SFM-Net	1.773	1.103
MpbPPI (Backb+Sidec+SASA+AA)	1.765	1.180
MpbPPI (Backb+SASA+AA)	1.792	1.186
FoldX	2.637	1.600
Flex ddG	3.842	2.250
MM/GBSA	8.102	7.174
S1131	RMSE	MAE
SFM-Net	1.737	1.340
MpbPPI (Backb+Sidec+SASA+AA)	2.450	1.772
MpbPPI (Backb+SASA+AA)	2.586	1.808
FoldX	2.661	1.776
Flex ddG	3.505	2.346
MM/GBSA	6.518	5.076
S4169	RMSE	MAE
SFM-Net	1.536	1.058
MpbPPI (Backb+Sidec+SASA+AA)	1.636	1.068
MpbPPI (Backb+SASA+AA)	1.682	1.086
FoldX	2.263	1.461
Flex ddG	4.488	2.972
MM/GBSA	4.962	3.909

center nodes and their neighboring residues within the distance threshold. For fair comparisons, we adopted the same complex-level data split as MpbPPI. The detailed results are provided in Figure 4.11. SFM-Net achieves PCC, RMSE, MAE values of 0.4603, 1.9859, 1.3224, respectively. Our method outperforms MpbPPI and other methods, demonstrating that our model can generalize to multi-mutation scenarios.

Threshold sensitivity analysis

We conducted a threshold sensitivity analysis on the S1131 dataset, varying the distance threshold for defining neighboring residues from 6 Å to 50 Å in increments of 2 Å with an additional evaluation at 50 Å. The results show a gradual improvement in performance as the threshold increased from 6 Å to 20 Å after which the performance saturates. The performance comparison of SFM-Net across different thresholds on the S1131 dataset is presented in Figure 4.12. Additional experiments

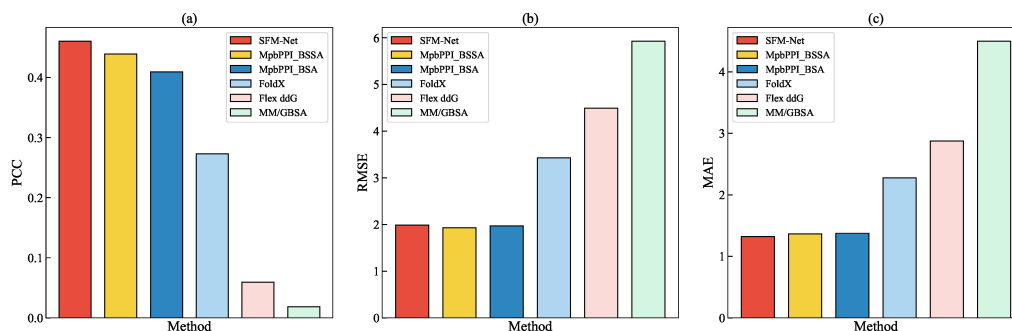


Figure 4.11: Performance comparison of SFM-Net and other involved methods on M1101 dataset in terms of PCC, RMSE, MAE under the WT protein-protein complex-based cross-validations.

on the S4169 dataset (thresholds: 6 Å, 8 Å, 10 Å, 20 Å, and 50 Å) exhibit a similar performance trend, further confirming the robustness of our approach. Detailed experimental results are provided in Table 4.5. Although our model achieves better performance with larger thresholds, we adopt the 10 Å threshold to maintain consistency with prior works.

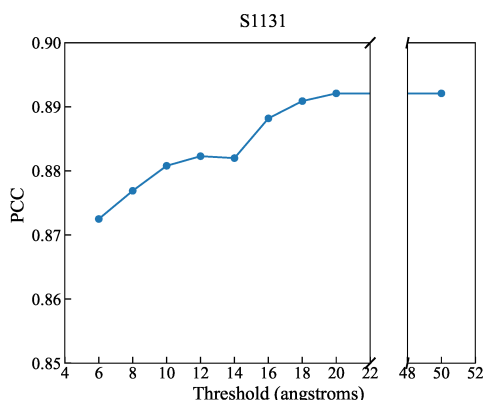


Figure 4.12: Performance comparison of SFM-Net on the S1131 dataset in terms of Pearson's correlation coefficient across different thresholds for defining neighboring residues.

Table 4.5: Performance comparison of SFM-Net on the S4169 dataset in terms of Pearson's correlation coefficient across different thresholds for defining neighboring residues

Threshold (Å)	6	8	10	20	50
PCC	0.7798	0.7876	0.7970	0.8104	0.8091

4.4.2 Ablation Analysis

To evaluate the effectiveness of the summation and difference components and the proposed CSF module for feature fusion, we conduct targeted ablation studies based

on the 10-fold cross-validation test.

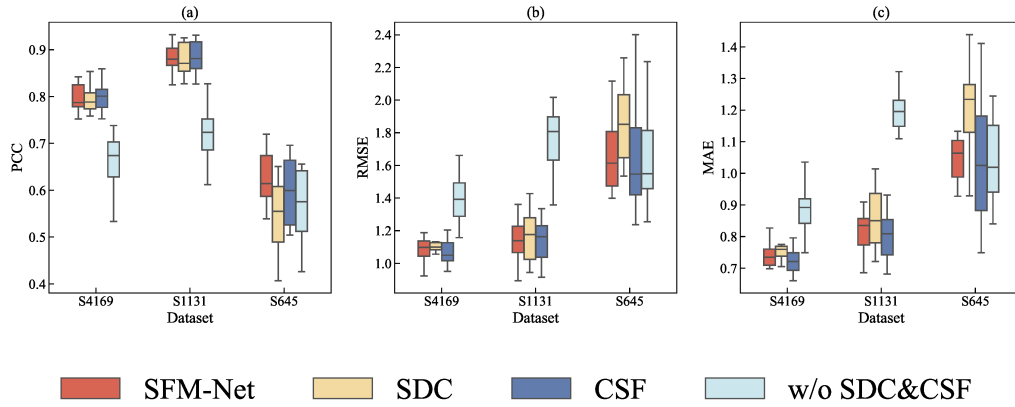


Figure 4.13: Performance comparison of different variant models of SFM-Net across three benchmark datasets. Each box plot illustrates the distribution of 10-fold results for each model (SFM-Net, SDC, CSF, and w/o SDC&CSF) within a dataset. The three subfigures represent different evaluation metrics: (a) Pearson's correlation coefficient (PCC), (b) root mean squared error (RMSE), and (c) mean absolute error (MAE).

Table 4.6: Performance comparison of different variant models of SFM-Net on three benchmark datasets in terms of Pearson's correlation coefficient

Variant	S4169	S1131	S645	Mean
SFM-Net	0.797	0.881	0.624	0.767
SDC	0.794	0.879	0.543	0.739
CSF	0.798	0.883	0.600	0.760
w/o SDC&CSF	0.652	0.719	0.566	0.646

Ablation study on the proposed modules

In this part, we evaluate the contribution introduced by the *summation and difference components* and *context-aware selective fusion module*. Specifically, we conduct or ablate the SFM-Net on all three benchmark datasets and compare the corresponding results. Four control groups are set up: (1) the original SFM-Net; the variant SFM-Nets that (2) retains the summation and difference components while discards the context-aware selective fusion module which is replaced by a multi-layer perceptron to make the final prediction (SDC); (3) removes the summation and difference components and directly feed the pooled features extracted from four parallel GNN-based blocks into the context-aware selective fusion module (CSF); (4) removes both the summation and difference components and context-aware selective fusion module, and replaces them with multi-layer perceptron to make the final prediction (w/o SDC&CSF), i.e., the four compact features obtained from the

four parallel GNN-based blocks are directly generated by the mean pool operations and processed by multi-layer perceptron for final predictions.

In Table 4.6, we report the ablation study results of Pearson’s correlation coefficient on the three benchmark datasets, where our main observations are: (1) on datasets S4169 and S1131, all the models with our proposed module(s) outperform the blank control group by a large margin, which demonstrates the effectiveness of the summation and difference components and the attention-based context-aware selective fusion module; (2) the models with CSF outperform the blank control group on three datasets and show high performance on dataset S4169 and S1131; (3) the models with SDC show comparable performance as original SFM-Net on dataset S4169 and S1131 and decrease on S645, while the original SFM-Net (with SDC and CSF) realizes the best performance and demonstrates significant improvement to other variants on dataset S645; (4) Our method, equipped with the proposed modules tailored for specific functions, demonstrates strong adaptability across datasets with varying statistical attributes. Notably, the full SFM-Net achieves the highest average performance across the three datasets, underscoring the indispensability of each module. The removal of any single module results in a significant decline in overall performance, highlighting the complementary roles these modules play in effectively addressing the diverse characteristics of the datasets, particularly on S645. Figure 4.13 shows the 10-fold result distributions for each model (SFM-Net, SDC, CSF, and w/o SDC&CSF) within each dataset. Detailed metric results are presented in Table 4.7.

It is worth noting that our dual-scope attention module demonstrates particular advantages on the S645 dataset, which exhibits a highly complex data distribution and has a higher proportion of extreme data. These unique characteristics make it especially challenging for data-driven methods to achieve robust performance. Specifically, our attention mechanism refines informative hidden interactions in proteins and builds robustness against distractions from outliers and extreme samples, enabling improved performance in such challenging scenarios.

Effectiveness of the features

We conduct ablation experiments to assess and discuss the effectiveness of three types of features, DSSP, PSSM, and the protein representation extracted by ProtT5. Specifically, we examine the different variants of SFM-Net, including SFM-Net

Table 4.7: Ablation study on the proposed modules of SFM-Net with Pearson’s correlation coefficient and other evaluation metrics on three datasets

Variants	Metrics	S4169	S1131	S645	Mean
SFM-Net	PCC	0.7970	0.8808	0.6241	0.7673
	MSE	1.1763	1.3198	2.8570	1.7844
	RMSE	1.0821	1.1420	1.6727	1.2989
	MAE	0.7331	0.8178	1.0993	0.8834
SDC	PCC	0.7941	0.8788	0.5430	0.7386
	MSE	1.1910	1.4172	3.4914	2.0332
	RMSE	1.0883	1.1788	1.8545	1.3739
	MAE	0.7441	0.8767	1.2041	0.9416
CSF	PCC	0.7979	0.8826	0.5996	0.7600
	MSE	1.1483	1.2882	2.8337	1.7567
	RMSE	1.0690	1.1270	1.6491	1.2817
	MAE	0.7239	0.7996	1.0420	0.8552
w/o SDC&CSF	PCC	0.6521	0.7187	0.5664	0.6457
	MSE	1.9821	3.1430	2.7596	2.6282
	RMSE	1.3996	1.7618	1.6376	1.5997
	MAE	0.8835	1.1876	1.0386	1.0366

(ProtT5), SFM-Net (DSSP), SFM-Net (PSSM), SFM-Net (DSSP+PSSM), SFM-Net (ProtT5+PSSM), SFM-Net (ProtT5+DSSP). These experiments are implemented on three datasets using 10-fold cross-validation.

The comparative results of models with different feature combinations on PCC, RMSE, and MAE across three benchmark datasets are reported in Figure 4.14, where we have two major observations accordingly: (1) SFM-Net (ProtT5+DSSP+PSSM), SFM-Net (ProtT5+PSSM), SFM-Net (ProtT5+DSSP) and SFM-Net (ProtT5) achieve comparable performance on three datasets, while SFM-Net (DSSP+PSSM), SFM-Net (DSSP) and SFM-Net (PSSM) show significantly inferior predictions to the former models, which demonstrates the protein representation extracted by ProtT5 is the most important feature of the three. (2) The original model, SFM-Net (ProtT5+DSSP+PSSM) further improves on SFM-Net (ProtT5+PSSM), SFM-Net (ProtT5+DSSP), and SFM-Net (ProtT5) and achieves the best performance on datasets S4169 and S645, which indicates that DSSP and PSSM also contribute to the prediction of binding affinity changes. Note that the additional results in terms of other metrics are provided in Table 4.8.

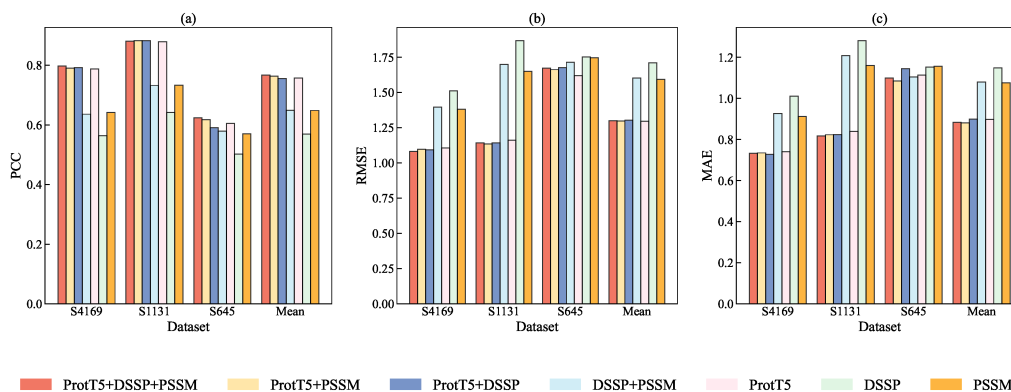


Figure 4.14: Ablation study on feature importance across three benchmark datasets. The three subfigures correspond to different evaluation metrics: (a) Pearson’s correlation coefficient (PCC), (b) root mean squared error (RMSE), and (c) mean absolute error (MAE).

4.4.3 Case Study

The performance of SFM-Net is further evaluated on an additional validation dataset, Wuhan-Hu-1 RBD. This dataset, derived from deep mutational scanning, provides a comprehensive experimental resource focused on the receptor-binding domain (RBD) of the SARS-CoV-2 spike protein (PDB ID: 6M0J) [99]. By systematically applying saturation mutagenesis, the effects of 285 single-point mutations across 15 critical RBD sites on binding affinity to the human angiotensin-converting enzyme 2 (ACE2) receptor were experimentally measured. Our method achieves a Pearson’s correlation coefficient of 0.563, outperforming FoldX [73], RDE-Network [100], DiffAffinity [101], DDAffinity [102], as shown in Table 4.9. We also provide the distribution of predicted $\Delta\Delta G$ against experimental $\Delta\Delta G$ and the comparative analysis of the predicted $\Delta\Delta G$ and experimental $\Delta\Delta G$ for 15 different mutation sites on this dataset in Figure 4.15.

4.5 Discussions

In this study, we have proposed a powerful deep learning method, termed SFM-Net, to predict PPI $\Delta\Delta G$ caused by mutations. We introduced a simple yet effective graph construction for each sample by centering the mutation and connecting the surrounding residues as neighboring nodes and adopting sequence, structural, and evolutionary information as node features. SFM-Net employs the GNN blocks to encode refined neural features from multi-source raw features. The changes in binding affinity are then mapped from the unified representations generated by a

Table 4.8: The ablation study on feature importance with Pearson’s correlation coefficient and other evaluation metrics on three datasets

Variants	Metrics	S4169	S1131	S645	Mean
ProtT5+DSSP+PSSM	PCC	0.7970	0.8808	0.6241	0.7673
	MSE	1.1763	1.3198	2.8570	1.7844
	RMSE	1.0821	1.1420	1.6727	1.2989
	MAE	0.7331	0.8178	1.0993	0.8834
ProtT5	PCC	0.7873	0.8792	0.6047	0.7571
	MSE	1.2320	1.3734	2.7208	1.7754
	RMSE	1.1067	1.1610	1.6200	1.2959
	MAE	0.7406	0.8394	1.1132	0.8977
DSSP	PCC	0.5638	0.6419	0.5024	0.5694
	MSE	2.3104	3.5470	3.1617	3.0064
	RMSE	1.5122	1.8681	1.7529	1.7111
	MAE	1.0105	1.2813	1.1530	1.1483
PSSM	PCC	0.6420	0.7332	0.5699	0.6484
	MSE	1.9193	2.7606	3.1171	2.5990
	RMSE	1.3804	1.6502	1.7465	1.5924
	MAE	0.9127	1.1604	1.1560	1.0764
DSSP+PSSM	PCC	0.6358	0.7325	0.5786	0.6490
	MSE	1.9696	2.9245	3.0092	2.6344
	RMSE	1.3964	1.6984	1.7151	1.6033
	MAE	0.9267	1.2080	1.1043	1.0797
ProtT5+PSSM	PCC	0.7902	0.8826	0.6174	0.7634
	MSE	1.2088	1.3035	2.8333	1.7819
	RMSE	1.0965	1.1347	1.6622	1.2978
	MAE	0.7347	0.8231	1.0852	0.8810
ProtT5+DSSP	PCC	0.7917	0.8826	0.5906	0.7550
	MSE	1.1993	1.3191	2.8710	1.7965
	RMSE	1.0923	1.1425	1.6771	1.3040
	MAE	0.7276	0.8240	1.1445	0.8987

context-driven selective feature fusion model, CSF. This new fusion model is induced by the proposed self-attention-inspired dual-scope fusion strategy that integrates both intra- and inter-sample attention in order. Our method addresses the void left by the absence of an effective, purely deep learning decoder capable of adaptively integrating multi-source protein features within their contextual framework for accurate PPI $\Delta\Delta G$ prediction. We believe this represents a significant contribution to addressing the current task and other related challenges. We conducted experiments on three mutation benchmark datasets, showing that SFM-Net could achieve preferable performance than SOTA methods in comprehensive evaluations and can effectively extract the important and relevant features for PPI $\Delta\Delta G$

Table 4.9: Performance comparison on SARS-CoV-2 RBD in terms of Pearson’s correlation coefficient

Method	FoldX	RDE-Network	DiffAffinity	DDAffinity	SFM-Net
PCC	0.385	0.438	0.466	0.456	0.563

* The best two results for each dataset are highlighted in **Bold** and **Gray**, respectively.

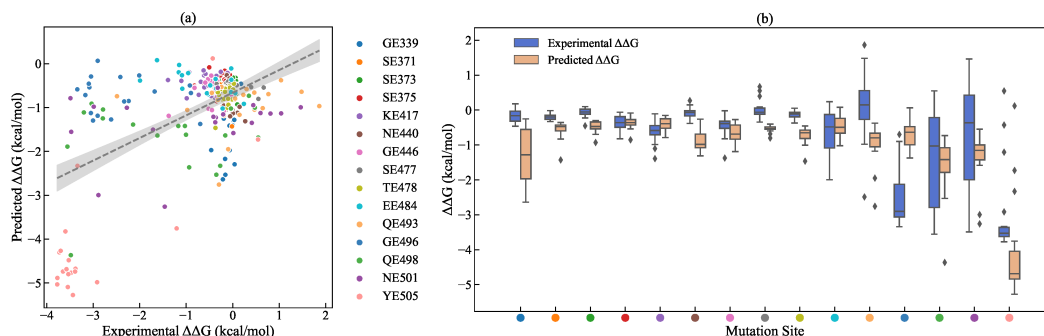


Figure 4.15: Distribution of predicted $\Delta\Delta G$ against experimental $\Delta\Delta G$. (a) Scatter plot showing the correlation between the predicted $\Delta\Delta G$ values and the experimental $\Delta\Delta G$ values for SARS-CoV-2 RBD mutations. Each point represents a single mutation, colored according to its mutation site. (b) Box plot comparing predicted and experimental $\Delta\Delta G$ values across different mutation sites. The x-axis represents mutation sites, and the colored circles below the axis correspond to the mutation categories as shown in (a). The blue boxes represent experimental values, while the orange boxes represent predicted values. This plot highlights the distribution of $\Delta\Delta G$ values for each mutation site.

prediction.

Despite the superior performance of SFM-Net, there is further room for improvement on this task. For instance, mutations on residues located at the interface can have differential impacts on the binding affinity than those in non-interface regions, as mutations at the interface may disrupt or strengthen the interactions, leading to a decrease or increase in the binding affinity, or alter the conformation of the interface, making the fit between the chains less optimal. When constructing the graph, expanding the range of neighboring nodes for the mutation—specifically, establishing edges between mutation and neighboring residues located in other chains—may provide a more comprehensive representation of the mutation’s spatial conformation, thereby contributing to training a better model. In our future work, we plan to (1) explore the development of tailored graph-based deep learning mechanisms that enable the efficient and effective detection, selection, and integration of key information, to further advance the SFM framework; (2)

develop tailored approaches to enhance our framework for processing multiple mutations. This includes deriving advanced graph-learning algorithms with an adaptive attention range and efficient multi-feature adaptive information selection and aggregation mechanisms, which dynamically focus on critical intra- and inter-chain mutation information, extract contextual information, and conduct a comprehensive analysis to accommodate different types of mutations.

Chapter 5

Conclusion and Future Directions

5.1 Conclusion

In this dissertation, we investigated key challenges in deep learning-based bioinformatics, focusing on representation learning that effectively integrates and selects features tailored to the characteristics of biological data. Specifically, our research targeted two distinct data types: sequence-based DNA data and structure-based protein data. We proposed two novel frameworks, namely MSNet-4mC, designed for DNA 4mC site prediction, and SFM-Net, developed for predicting PPI binding affinity changes upon mutations. These models incorporate task-specific feature extractors and advanced fusion mechanisms to capture complex contextual dependencies within the input representations. Our proposed approaches are highly inspired by advances in machine learning and computer vision. However, methods from these domains are not naturally effective when directly applied to bioinformatics tasks. To address this, our approach not only differs in implementation from conventional machine learning and computer vision methods, but also incorporates methodological adaptations tailored to the characteristics of biological data and the specific demands of the tasks. These adaptations enable us to leverage deep learning techniques more effectively for improving data analysis in bioinformatics.

First, as presented in Chapter 3, we addressed the task of predicting DNA N4-methylcytosine (4mC) sites, aiming to develop lightweight models to capture complex contextual dependencies within biological sequence data. To achieve this, we proposed MSNet-4mC, a built-in scale-aware neural network that leverages convolutional operations with multi-scale receptive fields to capture cross-element relationships across both short and long ranges in DNA sequences. The framework adjusts receptive fields to effectively explore internal dependencies, exploiting strong correlations among flanking bases. Furthermore, species-specific class weights, computed based on frequency distributions, are integrated into the cross-entropy loss function to balance training. Comprehensive experimental evaluations and ablation studies on benchmark datasets show that MSNet-4mC outperforms state-of-the-art (SOTA) methods, despite not relying on complex encoding schemes. These results validate the effectiveness and robustness of the model in identifying DNA 4mC sites.

In addition to designing effective feature extractors for sequence data, we further developed customized extractors for protein data with complex three-dimensional structures. Specifically, in Chapter 4, we addressed the task of predicting changes in protein-protein interaction (PPI) binding affinity upon mutations, with a focus on the effective and selective fusion of multi-source features. Specifically, heterogeneous features represent different properties of protein complexes across various aspects and scales. Direct concatenation of these features may limit the effective exploration of the inherent contextual relationships at the feature level. To achieve this, we proposed SFM-Net, a robust deep learning model that integrates GNN-based multi-way feature extractors and a novel context-aware selective fusion (CSF) module to leverage sequence, structural, and evolutionary information in a unified framework. The CSF module first applies three range-specific pooling operations to each of the ProtT5, DSSP, and PSSM feature paths, resulting in nine different compact representations for fusion. This is followed by a novel two-tier self-attention-inspired dual-scope fusion strategy designed to capture effective intra-sample and inter-sample protein information. Additionally, to integrate the representations from the wild-type and mutant complexes, we introduced the summation and difference components to learn low-frequency cues and high-frequency cues, respectively. We validate SFM-Net through comprehensive experimental analysis and targeted ablation studies on benchmark datasets. The results indicate that SFM-Net achieves

superior performance compared to SOTA methods and demonstrate that the proposed modules effectively extract important and relevant features for PPI $\Delta\Delta G$ prediction.

5.2 Future Directions

Building on advancements in DNA 4mC site identification, several promising directions remain open for exploration. Noisy or redundant information across species may exist in the datasets used in this thesis. Removing such information or employing strategies such as using longer DNA segments may further improve model performance. Additionally, the current framework uses fixed receptive fields for long- and short-range relationships. Developing adaptive strategies that account for varying distances from central 4mC sites may help uncover critical information and enhance predictive performance.

Based on the study of PPI $\Delta\Delta G$ prediction, we discuss the unexplored challenges worth further investigation. For instance, during graph construction, adding edges between the mutation site and neighboring residues from other chains may provide a more accurate representation of the structural context surrounding the mutation. In addition, exploring task-specific feature engineering techniques, such as graph-based representations of protein complexes, may enhance the model's ability to capture relevant patterns across different scales. Further advancement may be achieved by designing adaptive graph-learning algorithms that dynamically focus on critical intra- and inter-chain mutation information and effectively integrate multi-source features for robust modeling.

There are some future research directions for both tasks. Developing real-time prediction tools based on these models would make them more accessible to researchers and easier to integrate with high-throughput experimental pipelines, thereby improving research efficiency. Exploring Transformer models and other novel neural architectures or strategies on these tasks may help capture complex dependencies within DNA methylation patterns and protein-protein interaction data, leading to significant performance improvements and uncovering previously hidden biological insights. Progress along these directions not only has the potential to refine current models but also to open new avenues for impactful biological research and applications.

Moreover, the specialized feature extractors are expected to be transferable to other problems with similar data characteristics. Although the two proposed networks are tailored to specific tasks in genomic and proteomic data analysis, their design principles are broadly applicable and can be generalized to other bioinformatics problems. The use of multi-scale convolutional layers in MSNet-4mC enables effective capture of local and contextual sequence patterns. This strategy can be readily extended to other sequence-based tasks involving DNA or RNA, such as transcription factor binding site prediction or RNA modification site identification. The architecture is flexible with respect to input length and encoding schemes, making it adaptable to a variety of sequence-based prediction problems. The graph-based encoders, summation and difference components, and context-aware selective fusion module in SFM-Net are designed to model structural and relational information in proteins. This framework is not limited to $\Delta\Delta G$ prediction and can be generalized to other structure-informed tasks such as protein-protein interaction prediction, binding site identification, or stability change estimation. Minor adjustments to input representations (e.g., node features or edge construction rules) would allow the network to be adapted to a wide range of mutation-related and structure-based tasks. Overall, the modular design and biologically tailored feature extractors of the networks enhance their adaptability to a broad range of biological tasks and demonstrate strong potential for generalization.

References

- [1] L. D. Moore, T. Le, and G. Fan, “DNA methylation and its basic function,” *Neuropsychopharmacology*, vol. 38, no. 1, pp. 23–38, 2013. [Cited on page 7]
- [2] Z. D. Smith and A. Meissner, “DNA methylation: roles in mammalian development,” *Nature Reviews Genetics*, vol. 14, no. 3, pp. 204–220, 2013. [Cited on page 7]
- [3] M. V. Greenberg and D. Bourc’his, “The diverse roles of DNA methylation in mammalian development and disease,” *Nature Reviews Molecular Cell Biology*, vol. 20, no. 10, pp. 590–607, 2019. [Cited on pages 7 and 17]
- [4] R. L. Adams and R. H. Burdon, *Molecular biology of DNA methylation*. Springer Science & Business Media, 2012. [Cited on page 7]
- [5] F. Rodriguez, I. A. Yushenova, D. DiCorpo, and I. R. Arkhipova, “Bacterial N4-methylcytosine as an epigenetic mark in eukaryotic DNA,” *Nature Communications*, vol. 13, no. 1, p. 1072, 2022. [Cited on page 8]
- [6] H. Lv, F.-Y. Dao, D. Zhang, H. Yang, and H. Lin, “Advances in mapping the epigenetic modifications of 5-methylcytosine (5mC), N6-methyladenine (6mA), and N4-methylcytosine (4mC),” *Biotechnology and Bioengineering*, vol. 118, no. 11, pp. 4204–4216, 2021. [Cited on page 8]
- [7] R. A. Gaultney, A. T. Vincent, C. Lorigou, J.-Y. Coppée, O. Sismeiro, H. Varet, R. Legendre, C. A. Cockram, F. J. Veyrier, and M. Picardeau, “4-Methylcytosine DNA modification is critical for global epigenetic regulation and virulence in the human pathogen *Leptospira interrogans*,” *Nucleic Acids Research*, vol. 48, no. 21, pp. 12102–12115, 2020. [Cited on page 8]
- [8] S. Damodaran, “Amino acids, peptides and proteins,” *Fennema’s Food Chemistry*, vol. 4, pp. 425–439, 2008. [Cited on page 8]

- [9] J. P. Schneider and J. W. Kelly, “Templates that Induce. Alpha-helical,. Beta.-sheet, and loop conformations,” *Chemical Reviews*, vol. 95, no. 6, pp. 2169–2187, 1995. [Cited on page 8]
- [10] B. Alberts, D. Bray, K. Hopkin, A. D. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter, *Essential Cell Biology*. Garland Science, 2015. [Cited on page 8]
- [11] T. Berggård, S. Linse, and P. James, “Methods for the detection and analysis of protein–protein interactions,” *Proteomics*, vol. 7, no. 16, pp. 2833–2842, 2007. [Cited on page 8]
- [12] P. Braun and A.-C. Gingras, “History of protein–protein interactions: From egg-white to complex networks,” *Proteomics*, vol. 12, no. 10, pp. 1478–1498, 2012. [Cited on pages 8 and 41]
- [13] L.-G. Milroy, T. N. Grossmann, S. Hennig, L. Brunsveld, and C. Ottmann, “Modulators of protein–protein interactions,” *Chemical Reviews*, vol. 114, no. 9, pp. 4695–4748, 2014. [Cited on pages 8 and 41]
- [14] O. Keskin, N. Tuncbag, and A. Gursoy, “Predicting protein–protein interactions from the molecular to the proteome level,” *Chemical Reviews*, vol. 116, no. 8, pp. 4884–4909, 2016. [Cited on page 8]
- [15] T. Pawson and P. Nash, “Assembly of cell regulatory systems through protein interaction domains,” *Science*, vol. 300, no. 5618, pp. 445–452, 2003. [Cited on page 9]
- [16] W. E. Stites, “Protein–protein interactions: Interface structure, binding thermodynamics, and mutational analysis,” *Chemical Reviews*, vol. 97, no. 5, pp. 1233–1250, 1997. [Cited on page 9]
- [17] L. Hao, J. Sun, M. Pei, G. Zhang, C. Li, C. Li, X. Ma, S. He, and L. Liu, “Impact of non-covalent bound polyphenols on conformational, functional properties and in vitro digestibility of pea protein,” *Food Chemistry*, vol. 383, p. 132623, 2022. [Cited on page 9]
- [18] P. L. Kastiris and A. M. Bonvin, “On the binding affinity of macromolecular interactions: Daring to ask why proteins interact,” *Journal of the Royal Society Interface*, vol. 10, no. 79, p. 20120835, 2013. [Cited on page 9]

- [19] J. Jankauskaitė, B. Jiménez-García, J. Dapkūnas, J. Fernández-Recio, and I. H. Moal, “SKEMPI 2.0: An updated benchmark of changes in protein–protein binding energy, kinetics and thermodynamics upon mutation,” *Bioinformatics*, vol. 35, no. 3, pp. 462–469, 2019. [Cited on pages 9, 10, and 45]
- [20] A. T. S. Bodapati, R. S. Reddy, K. Lavanya, S. R. Madku, and B. K. Sahoo, “A comprehensive biophysical and theoretical study on the binding of dexlansoprazole with human serum albumin,” *Journal of Molecular Liquids*, vol. 380, p. 121777, 2023. [Cited on page 9]
- [21] I. M. Nooren and J. M. Thornton, “Diversity of protein–protein interactions,” *The EMBO Journal*, vol. 22, no. 14, pp. 3486–3492, 2003. [Cited on page 9]
- [22] C. Schaefer and B. Rost, “Predict impact of single amino acid change upon protein structure,” in *BMC Genomics*, vol. 13, pp. 1–10, Springer, 2012. [Cited on page 9]
- [23] G. W. Lindsay, “Convolutional neural networks as a model of the visual system: Past, present, and future,” *Journal of Cognitive Neuroscience*, vol. 33, no. 10, pp. 2017–2031, 2021. [Cited on page 11]
- [24] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, *et al.*, “Recent advances in convolutional neural networks,” *Pattern Recognition*, vol. 77, pp. 354–377, 2018. [Cited on page 11]
- [25] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. [Cited on page 12]
- [26] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. [Cited on pages 12 and 21]
- [27] B. Khemani, S. Patil, K. Kotecha, and S. Tanwar, “A review of graph neural networks: Concepts, architectures, techniques, challenges, datasets, applications, and future directions,” *Journal of Big Data*, vol. 11, no. 1, p. 18, 2024. [Cited on page 12]
- [28] Z. Chen, Y. Wang, F. Ma, H. Yuan, and X. Wang, “GPL-GNN: Graph prompt learning for graph neural network,” *Knowledge-Based Systems*, vol. 286, p. 111391, 2024. [Cited on page 13]

- [29] W. L. Hamilton, *Graph representation learning*. Morgan & Claypool Publishers, 2020. [Cited on page 13]
- [30] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, “Graph neural networks: A review of methods and applications,” *AI Open*, vol. 1, pp. 57–81, 2020. [Cited on page 13]
- [31] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *Proc. International Conference on Learning Representations (ICLR)*, 2017. [Cited on page 13]
- [32] W. Hamilton, Z. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017. [Cited on page 13]
- [33] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, “Graph attention networks,” in *Proc. International Conference on Learning Representations (ICLR)*, 2018. [Cited on page 13]
- [34] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, “How powerful are graph neural networks?,” in *Proc. International Conference on Learning Representations (ICLR)*, 2019. [Cited on page 14]
- [35] L. Itti, C. Koch, and E. Niebur, “A model of saliency-based visual attention for rapid scene analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998. [Cited on page 14]
- [36] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” in *Proc. International Conference on Learning Representations (ICLR)*, 2015. [Cited on page 14]
- [37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017. [Cited on pages 15 and 51]
- [38] J. Arand, D. Spieler, T. Karius, M. R. Branco, D. Meilinger, A. Meissner, T. Jenuwein, G. Xu, H. Leonhardt, V. Wolf, *et al.*, “In vivo control of CpG and non-CpG DNA methylation by DNA methyltransferases,” *PLoS Genetics*, vol. 8, no. 6, p. e1002750, 2012. [Cited on page 17]

- [39] S. Jeudy, S. Rigou, J.-M. Alempic, J.-M. Claverie, C. Abergel, and M. Legendre, “The DNA methylation landscape of giant viruses,” *Nature Communications*, vol. 11, no. 1, pp. 1–12, 2020. [Cited on page 17]
- [40] P. A. Jones, “Functions of DNA methylation: Islands, start sites, gene bodies and beyond,” *Nature Reviews Genetics*, vol. 13, no. 7, pp. 484–492, 2012. [Cited on page 17]
- [41] B. M. Davis, M. C. Chao, and M. K. Waldor, “Entering the era of bacterial epigenomics with single molecule real time DNA sequencing,” *Current Opinion in Microbiology*, vol. 16, no. 2, pp. 192–198, 2013. [Cited on page 17]
- [42] R. J. Roberts, T. Vincze, J. Posfai, and D. Macelis, “REBASE—a database for DNA restriction and modification: Enzymes, genes and genomes,” *Nucleic Acids Research*, vol. 43, no. D1, pp. D298–D299, 2015. [Cited on page 17]
- [43] D. Barros-Silva, C. J. Marques, R. Henrique, and C. Jerónimo, “Profiling DNA methylation based on next-generation sequencing approaches: New insights and clinical applications,” *Genes*, vol. 9, no. 9, p. 429, 2018. [Cited on page 17]
- [44] B. A. Flusberg, D. R. Webster, J. H. Lee, K. J. Travers, E. C. Olivares, T. A. Clark, J. Korlach, and S. W. Turner, “Direct detection of DNA methylation during single-molecule, real-time sequencing,” *Nature Methods*, vol. 7, no. 6, pp. 461–465, 2010. [Cited on page 17]
- [45] S. Cai, Y. Guo, S. Khan, J. Hu, and G. Wen, “Ground-to-aerial image geo-localization with a hard exemplar reweighting triplet loss,” in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 8391–8400, 2019. [Cited on page 18]
- [46] Y. Fu, S. Okada, L. Wang, L. Guo, Y. Song, J. Liu, and J. Dang, “CONSK-GCN: Conversational semantic-and knowledge-oriented graph convolutional network for multimodal emotion recognition,” in *IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, IEEE, 2021. [Cited on page 18]
- [47] Y. Guo, M. Choi, K. Li, F. Boussaid, and M. Bennamoun, “Soft exemplar highlighting for cross-view image-based geo-localization,” *IEEE Transactions on Image Processing*, vol. 31, pp. 2094–2105, 2022. [Cited on page 18]

- [48] W. Chen, H. Yang, P. Feng, H. Ding, and H. Lin, “iDNA4mC: Identifying DNA N4-methylcytosine sites based on nucleotide chemical properties,” *Bioinformatics*, vol. 33, no. 22, pp. 3518–3523, 2017. [Cited on pages 18, 20, and 29]
- [49] P. Ye, Y. Luan, K. Chen, Y. Liu, C. Xiao, and Z. Xie, “MethSMRT: An integrative database for DNA N6-methyladenine and N4-methylcytosine generated by single-molecular real-time sequencing,” *Nucleic Acids Research*, p. gkw950, 2016. [Cited on page 18]
- [50] W. He, C. Jia, and Q. Zou, “4mCPred: Machine learning methods for DNA N4-methylcytosine sites prediction,” *Bioinformatics*, vol. 35, no. 4, pp. 593–601, 2019. [Cited on page 18]
- [51] L. Wei, S. Luan, L. A. E. Nagai, R. Su, and Q. Zou, “Exploring sequence-based features for the improved prediction of DNA N4-methylcytosine sites in multiple species,” *Bioinformatics*, vol. 35, no. 8, pp. 1326–1333, 2019. [Cited on page 18]
- [52] B. Manavalan, S. Basith, T. H. Shin, L. Wei, and G. Lee, “Meta-4mCpred: A sequence-based meta-predictor for accurate DNA 4mC site prediction using effective feature representation,” *Molecular Therapy-Nucleic Acids*, vol. 16, pp. 733–744, 2019. [Cited on page 18]
- [53] L. Wei, R. Su, S. Luan, Z. Liao, B. Manavalan, Q. Zou, and X. Shi, “Iterative feature representations improve N4-methylcytosine site prediction,” *Bioinformatics*, vol. 35, no. 23, pp. 4930–4937, 2019. [Cited on page 18]
- [54] J. Khanal, I. Nazari, H. Tayara, and K. T. Chong, “4mCCNN: Identification of N4-methylcytosine sites in prokaryotes using convolutional neural network,” *IEEE Access*, vol. 7, pp. 145455–145461, 2019. [Cited on page 18]
- [55] Q. Liu, J. Chen, Y. Wang, S. Li, C. Jia, J. Song, and F. Li, “DeepTorrent: A deep learning-based approach for predicting DNA N4-methylcytosine sites,” *Briefings in Bioinformatics*, vol. 22, no. 3, p. bbaa124, 2021. [Cited on pages 18, 20, 29, 31, 32, 34, and 36]
- [56] L. Fu, B. Niu, Z. Zhu, S. Wu, and W. Li, “CD-HIT: Accelerated for clustering the next-generation sequencing data,” *Bioinformatics*, vol. 28, no. 23, pp. 3150–3152, 2012. [Cited on page 20]

- [57] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7132–7141, 2018. [Cited on page 22]
- [58] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feed-forward neural networks,” in *Proc. the thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, JMLR Workshop and Conference Proceedings, 2010. [Cited on page 24]
- [59] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification,” in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 1026–1034, 2015. [Cited on page 24]
- [60] A. M. Saxe, J. L. McClelland, and S. Ganguli, “Exact solutions to the non-linear dynamics of learning in deep linear neural networks,” in *Proc. International Conference on Learning Representations (ICLR)*, 2014. [Cited on page 24]
- [61] K.-C. Wong, Y. Li, and C. Peng, “Identification of coupling DNA motif pairs on long-range chromatin interactions in human K562 cells,” *Bioinformatics*, vol. 32, no. 3, pp. 321–324, 2016. [Cited on page 24]
- [62] M. Arenas, “Trends in substitution models of molecular evolution,” *Frontiers in Genetics*, vol. 6, p. 319, 2015. [Cited on page 24]
- [63] A. P. Bird, “DNA methylation and the frequency of CpG in animal DNA,” *Nucleic Acids Research*, vol. 8, no. 7, pp. 1499–1504, 1980. [Cited on page 24]
- [64] D. Lim and M. Blanchette, “EvoLSTM: Context-dependent models of sequence evolution using a sequence-to-sequence LSTM,” *Bioinformatics*, vol. 36, no. Supplement_1, pp. i353–i361, 2020. [Cited on page 24]
- [65] K. D. Makova and R. C. Hardison, “The effects of chromatin organization on variation in mutation rates in the genome,” *Nature Reviews Genetics*, vol. 16, no. 4, pp. 213–223, 2015. [Cited on page 24]
- [66] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988, 2017. [Cited on page 25]

- [67] L. Van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *Journal of Machine Learning Research*, vol. 9, no. 11, 2008. [Cited on page 32]
- [68] T. Pawson and P. Nash, “Protein–protein interactions define specificity in signal transduction,” *Genes & Development*, vol. 14, no. 9, pp. 1027–1047, 2000. [Cited on page 41]
- [69] F. Cheng, J. Zhao, Y. Wang, W. Lu, Z. Liu, Y. Zhou, W. R. Martin, R. Wang, J. Huang, T. Hao, *et al.*, “Comprehensive characterization of protein–protein interactions perturbed by disease mutations,” *Nature Genetics*, vol. 53, no. 3, pp. 342–353, 2021. [Cited on page 41]
- [70] P. Schuck, “Reliable determination of binding affinity and kinetics using surface plasmon resonance biosensors,” *Current Opinion in Biotechnology*, vol. 8, no. 4, pp. 498–502, 1997. [Cited on page 42]
- [71] A. Velazquez-Campoy and E. Freire, “Isothermal titration calorimetry to determine association constants for high-affinity ligands,” *Nature Protocols*, vol. 1, no. 1, pp. 186–191, 2006. [Cited on page 42]
- [72] A. M. Rossi and C. W. Taylor, “Analysis of protein–ligand interactions by fluorescence polarization,” *Nature Protocols*, vol. 6, no. 3, pp. 365–387, 2011. [Cited on page 42]
- [73] J. Schymkowitz, J. Borg, F. Stricher, R. Nys, F. Rousseau, and L. Serrano, “The FoldX web server: An online force field,” *Nucleic Acids Research*, vol. 33, no. suppl_2, pp. W382–W388, 2005. [Cited on pages 42, 46, 53, and 67]
- [74] T. Kortemme, A. V. Morozov, and D. Baker, “An orientation-dependent hydrogen bonding potential improves prediction of specificity and structure for proteins and protein–protein complexes,” *Journal of Molecular Biology*, vol. 326, no. 4, pp. 1239–1259, 2003. [Cited on page 42]
- [75] Y. Dehouck, J. M. Kwasigroch, M. Rooman, and D. Gilis, “BeAtMuSiC: Prediction of changes in protein–protein binding affinity on mutations,” *Nucleic Acids Research*, vol. 41, no. W1, pp. W333–W339, 2013. [Cited on page 42]
- [76] P. Xiong, C. Zhang, W. Zheng, and Y. Zhang, “BindProfX: Assessing mutation-induced binding affinity change by protein interface profiles with pseudo-counts,” *Journal of Molecular Biology*, vol. 429, no. 3, pp. 426–434, 2017. [Cited on pages 42 and 45]

- [77] C. Geng, A. Vangone, G. E. Folkers, L. C. Xue, and A. M. Bonvin, “iSEE: Interface structure, evolution, and energy-based machine learning predictor of binding affinity changes upon mutations,” *Proteins: Structure, Function, and Bioinformatics*, vol. 87, no. 2, pp. 110–119, 2019. [Cited on page 42]
- [78] C. H. Rodrigues, Y. Myung, D. E. Pires, and D. B. Ascher, “mCSM-PPI2: Predicting the effects of mutations on protein–protein interactions,” *Nucleic Acids Research*, vol. 47, no. W1, pp. W338–W344, 2019. [Cited on pages 42 and 45]
- [79] S. Pahari, G. Li, A. K. Murthy, S. Liang, R. Fragoza, H. Yu, and E. Alexov, “SAAMBE-3D: Predicting effect of mutations on protein–protein interactions,” *International Journal of Molecular Sciences*, vol. 21, no. 7, p. 2563, 2020. [Cited on page 42]
- [80] G. Li, S. Pahari, A. K. Murthy, S. Liang, R. Fragoza, H. Yu, and E. Alexov, “SAAMBE-SEQ: A sequence-based method for predicting mutation effect on protein–protein binding affinity,” *Bioinformatics*, vol. 37, no. 7, pp. 992–999, 2021. [Cited on page 42]
- [81] S. Cai, “IIEU: Rethinking neural feature activation from decision-making,” in *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 5796–5806, 2023. [Cited on page 42]
- [82] S. Cai, “Adashift: Learning discriminative self-gated neural feature activation with an adaptive shift factor,” in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5947–5956, 2024. [Cited on page 42]
- [83] M. Wang, Z. Cang, and G.-W. Wei, “A topology-based network tree for the prediction of protein–protein binding affinity changes following mutation,” *Nature Machine Intelligence*, vol. 2, no. 2, pp. 116–123, 2020. [Cited on pages 42 and 53]
- [84] J. Wee and K. Xia, “Persistent spectral based ensemble learning (PerSpect-EL) for protein–protein binding affinity prediction,” *Briefings in Bioinformatics*, vol. 23, no. 2, p. bbac024, 2022. [Cited on pages 42 and 53]
- [85] L. Wu, P. Cui, J. Pei, L. Zhao, and X. Guo, “Graph neural networks: Foundation, frontiers and applications,” in *Proc. ACM SIGKDD Conference on*

- Knowledge Discovery and Data Mining*, pp. 4840–4841, 2022. [Cited on page 43]
- [86] Y. Jiang, L. Quan, K. Li, Y. Li, Y. Zhou, T. Wu, and Q. Lyu, “DGCddG: Deep graph convolution for predicting protein-protein binding affinity changes upon mutations,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 20, no. 3, pp. 2089–2100, 2023. [Cited on pages 43 and 53]
- [87] Y. Yue, S. Li, L. Wang, H. Liu, H. H. Tong, and S. He, “MpbPPI: A multi-task pre-training-based equivariant approach for the prediction of the effect of amino acid mutations on protein–protein interactions,” *Briefings in Bioinformatics*, vol. 24, no. 5, p. bbad310, 2023. [Cited on pages 43, 53, and 55]
- [88] I. H. Moal and J. Fernández-Recio, “SKEMPI: A structural kinetic and energetic database of mutant protein interactions and its use in empirical models,” *Bioinformatics*, vol. 28, no. 20, pp. 2600–2607, 2012. [Cited on page 45]
- [89] S. Sirin, J. R. Apgar, E. M. Bennett, and A. E. Keating, “AB-bind: Antibody binding mutational database for computational affinity predictions,” *Protein Science*, vol. 25, no. 2, pp. 393–409, 2016. [Cited on pages 45 and 61]
- [90] D. E. Pires and D. B. Ascher, “mCSM-AB: A web server for predicting antibody–antigen affinity changes upon mutation with graph-based signatures,” *Nucleic Acids Research*, vol. 44, no. W1, pp. W469–W473, 2016. [Cited on page 45]
- [91] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, “The protein data bank,” *Nucleic Acids Research*, vol. 28, no. 1, pp. 235–242, 2000. [Cited on page 46]
- [92] A. Elnaggar, M. Heinzinger, C. Dallago, G. Rehawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, M. Steinegger, *et al.*, “ProtTrans: Toward understanding the language of life through self-supervised learning,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 7112–7127, 2021. [Cited on page 46]
- [93] T. Nishiyama, A. Kumagai, K. Kamiya, and K. Takahashi, “SILU: Strategy involving large-scale unlabeled logs for improving malware detector,” in *2020 IEEE Symposium on Computers and Communications (ISCC)*, pp. 1–7, IEEE, 2020. [Cited on page 49]

- [94] C. Subakan, M. Ravanelli, S. Cornell, M. Bronzi, and J. Zhong, “Attention is all you need in speech separation,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 21–25, IEEE, 2021. [Cited on page 51]
- [95] X. Liu, Y. Luo, P. Li, S. Song, and J. Peng, “Deep geometric representations for modeling effects of mutations on protein–protein binding affinity,” *PLoS Computational Biology*, vol. 17, no. 8, p. e1009284, 2021. [Cited on page 53]
- [96] O. G. Sani, Y. Yang, M. B. Lee, H. E. Dawes, E. F. Chang, and M. M. Shanechi, “Mood variations decoded from multi-site intracranial human brain activity,” *Nature Biotechnology*, vol. 36, no. 10, pp. 954–961, 2018. [Cited on page 59]
- [97] L. Jin, B. Liu, F. Zhao, S. Hahn, B. Dong, R. Song, T. C. Elston, Y. Xu, and K. M. Hahn, “Deep learning enables structured illumination microscopy with low light levels and enhanced speed,” *Nature Communications*, vol. 11, no. 1, p. 1934, 2020. [Cited on page 59]
- [98] J. Li, X. S. Yan, D. Chaudhary, V. Avula, S. Mudiganti, H. Husby, S. Shahjouei, A. Afshar, W. F. Stewart, M. Yeasin, *et al.*, “Imputation of missing values for electronic health record laboratory data,” *NPJ Digital Medicine*, vol. 4, no. 1, p. 147, 2021. [Cited on page 59]
- [99] T. N. Starr, A. J. Greaney, W. W. Hannon, A. N. Loes, K. Hauser, J. R. Dillen, E. Ferri, A. G. Farrell, B. Dadonaite, M. McCallum, *et al.*, “Shifting mutational constraints in the SARS-CoV-2 receptor-binding domain during viral evolution,” *Science*, vol. 377, no. 6604, pp. 420–424, 2022. [Cited on page 67]
- [100] S. Luo, Y. Su, Z. Wu, C. Su, J. Peng, and J. Ma, “Rotamer density estimator is an unsupervised learner of the effect of mutations on protein–protein interaction,” in *Proc. International Conference on Learning Representations (ICLR)*, 2023. [Cited on page 67]
- [101] S. Liu, T. Zhu, M. Ren, C. Yu, D. Bu, and H. Zhang, “Predicting mutational effects on protein–protein binding via a side-chain diffusion probabilistic model,” in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2024. [Cited on page 67]

- [102] G. Yu, Q. Zhao, X. Bi, and J. Wang, “DDAffinity: Predicting the changes in binding affinity of multiple point mutations using protein 3D structure,” *Bioinformatics*, vol. 40, no. Supplement_1, pp. i418–i427, 2024. [Cited on page 67]

List of publications

- (Chapter 3): Chunting Liu, Jiangning Song, Hiroyuki Ogata, Tatsuya Akutsu. (2022). “MSNet-4mC: Learning effective multi-scale representations for identifying DNA N4-methylcytosine sites”, *Bioinformatics*, 38(23), 5160-5167.
- (Chapter 4): Chunting Liu, Sudong Cai, Tong Pan, Hiroyuki Ogata, Jiangning Song, Tatsuya Akutsu. (2025). “SFM-Net: Selective fusion of multi-way protein feature network for predicting binding affinity changes upon mutations”, *Journal of Chemical Information and Modeling*, 65(7), 3854-3865.