Grenze International Journal of Engineering and Technology, June Issue

HGTM: A Structural and Contextual Deep Learning Model for Viral DNA Classification

Amaresh A M[1], Mahadeva Prasad Y N[2], Basavaraju N M[3], Naveen H M[4], Parashivamurthy B M[5] and Pundalik Chavan[6]
[1,5]Assistant Professor, JSS Science & Technology University, Mysuru
Email: amaresh.am03@gmail.com, parashivamurthy@jssstuniv.in
[2]Assistant Professor, MITT AI and DS, Mysore, India
Email: mahadevprasadyn@gmail.com
[3]Assistant Professor, JSSATE, Bangalore
Email: basavarajunm@jssateb.ac.in
[4]Associate Professor, Bapuji Institute of Engineering and Technology, Davanagere
Email: naveenhm@bietdvg.edu
[6]Associate Professor, Reva University, Bangalore
Email: Pundalik.chavan29@gmail.com

***Abstract*— The exponential growth of genomic data has highlighted the need for advanced computational models to classify DNA sequences with high accuracy and biological interpretability. Traditional approaches such as CNN-LSTM architectures often struggle to capture both long-range dependencies and structural relationships within nucleotide sequences. To address these limitations, we propose a Hybrid Graph-Transformer Model (HGTM) that integrates Graph Neural Networks (GNNs) with Transformer encoders for DNA sequence classification. DNA sequences are tokenized into k-mers and embedded using pre-trained representations, while graph-based modeling encodes structural dependencies between nucleotides. A multi-head self-attention mechanism fuses features from both the GNN and Transformer components, enabling the model to leverage local structural motifs as well as long-range contextual relationships. Experiments on viral DNA datasets, including SARS and MERS, demonstrate that the proposed HGTM achieves superior accuracy and interpretability compared to conventional CNN, SVM, and Bi-LSTM approaches. This hybrid framework provides a scalable and biologically informed solution for genomic sequence classification, with potential applications in pathogen surveillance, drug discovery, and precision medicine.**

***Index Terms*— Graph Neural Networks (GNNs), Bi-LSTM, CNN, SVM.**

I. INTRODUCTION

DNA sequence classification is a fundamental task in bioinformatics, serving as the foundation for understanding genetic variations, identifying pathogenic organisms, and guiding therapeutic discovery. DNA, composed of four nucleotides—adenine (A), thymine (T), cytosine (C), and guanine (G)—encodes the biological instructions essential for life. Accurate classification of DNA sequences not only facilitates gene annotation but also plays a pivotal role in monitoring viral mutations, predicting disease pathways, and enabling precision medicine.

Grenze ID: 01.GIJET.12.2.161_21
© Grenze Scientific Society, 2026

Conventional machine learning and deep learning methods, such as Support Vector Machines (SVMs), Convolutional Neural Networks (CNNs)[8], and Long Short-Term Memory (LSTM) models, have demonstrated effectiveness in processing genomic data. While CNNs excel in capturing local sequence motifs and LSTMs model sequential dependencies, these approaches exhibit inherent limitations. CNNs are constrained by their local receptive fields, often failing to capture long-range interactions, whereas LSTMs face difficulties in modeling very long DNA sequences due to vanishing gradients. As a result, traditional models may overlook crucial structural and contextual relationships that define the biological function of DNA.

Recent advances in deep learning, particularly Graph Neural Networks (GNNs) [1, 2], and Transformer architectures, present promising alternatives. GNNs effectively capture structural dependencies by representing DNA sequences as graphs, where nucleotides and k-mers are nodes connected by biochemical or sequential relationships. Transformers, originally developed for natural language processing, excel at modeling long-range dependencies through self-attention mechanisms. The integration of these two paradigms offers an opportunity to overcome the limitations of existing approaches.

In this work, we propose a Hybrid Graph-Transformer Model (HGTM) [2, 3] for DNA sequence classification [12]. Our approach combines the structural learning capacity of GNNs with the contextual modeling power of Transformers, enhanced through a multi-head attention-based fusion layer [9]. By leveraging both local structural motifs and long-range contextual dependencies, HGTM provides robust and interpretable DNA classification. We evaluate the model using viral DNA datasets [11], specifically SARS and MERS genomes, demonstrating its superiority over conventional CNN, Bi-LSTM, and SVM-based methods.

This research contributes to advancing computational genomics by presenting a novel framework that is not only accurate but also interpretable and scalable. Beyond viral classification, the proposed methodology has potential applications in broader areas such as cancer genomics, evolutionary biology, and personalized medicine.

II. Literature Review

DNA sequence classification has been an active area of research in bioinformatics and computational biology, with approaches evolving from traditional statistical methods to advanced deep learning models. This section reviews key contributions in the field, highlighting their strengths and limitations, which motivate the development of the proposed Hybrid Graph-Transformer Model (HGTM).

Early studies employed statistical and machine learning approaches such as Logistic Regression, Support Vector Machines (SVM), and Naïve Bayes for genomic classification tasks. Diallo et al. demonstrated the use of linear classification models to categorize hepatitis C viral genomes, achieving moderate accuracy but facing challenges in handling complex nonlinear dependencies. Similarly, probabilistic models such as Markov chains were used to capture sequence patterns but lacked scalability for large datasets.

With the rise of deep learning, Convolutional Neural Networks (CNNs) emerged as a dominant technique. CNNs demonstrated strong performance in capturing local motifs within DNA sequences, and Beinke et al. reported accuracies exceeding 99% on viral datasets by integrating CNNs with n-gram-based feature extraction. However, CNNs are inherently limited in modeling long-range dependencies, as their receptive fields focus on local patterns.

To address sequence-level dependencies, researchers turned to Recurrent Neural Networks (RNNs) [14] and their variants. LSTM and Bi-LSTM models were employed to capture long-term relationships between nucleotides. Abass et al. demonstrated the effectiveness of Bi-LSTM models in prostate cancer DNA sequence analysis, achieving over 90% accuracy. Nonetheless, RNN-based methods suffer from issues such as vanishing gradients and increased computational complexity when dealing with very long DNA sequences.

Hybrid architectures combining CNNs and LSTMs showed promising improvements. Gunasekaran et al. reported that CNN-LSTM and CNN-Bidirectional LSTM models enhanced performance in DNA [7] classification tasks, reaching accuracies of 93.16%. These models leveraged CNNs for motif extraction and LSTMs for contextual learning. Despite these advances, hybrid CNN-LSTM [8] approaches still underperformed in modeling intricate structural and contextual dependencies present in genomic data.

Recent research has explored attention mechanisms and Transformer architectures, inspired by breakthroughs in natural language processing. Lopez-Rincon et al. applied deep learning coupled with AI-driven techniques to classify SARS-CoV-2 sequences, achieving accuracies of 98.73% with high specificity. Transformers, such as DNABERT [4], have shown state-of-the-art performance by treating DNA as a sequence of k-mer tokens, capturing long-range dependencies more effectively than RNNs. However, Transformer models alone may overlook structural nucleotide interactions, which play a crucial role in biological function.

Parallel to these developments, Graph Neural Networks (GNNs) have gained attention in genomics for their ability to represent DNA as graphs. By modeling nucleotides or k-mers as nodes and their biochemical or structural relationships as edges, GNNs capture dependencies that linear sequence models may ignore. Although promising, GNNs alone are limited in capturing long-range sequential dependencies across entire genomes.
In the CNNs, LSTMs, and Transformers have advanced the state of DNA sequence classification, each has limitations when applied in isolation. CNNs are strong for local motifs but weak in long-range context; LSTMs model sequences but face scalability issues; Transformers capture global context but miss structural dependencies; GNNs model structure but overlook sequential order. These gaps provide the motivation for the proposed HGTM, which integrates GNNs and Transformers with attention-based fusion to leverage both structural and contextual features, achieving a more comprehensive and interpretable classification framework.

III. Proposed Methodology

A. Overview

This proposed methodology shown in Fig.1 and adopted for developing the Hybrid Graph-Transformer Model (HGTM) [10] designed to classify DNA sequences with enhanced accuracy and interpretability. The proposed approach integrates Graph Neural Networks (GNNs) [13] and Transformer encoders to simultaneously capture structural dependencies and long-range contextual relationships in genomic data. The overall workflow includes data acquisition, preprocessing, k-mer tokenization, feature embedding, graph construction, hybrid model design, training, and evaluation.

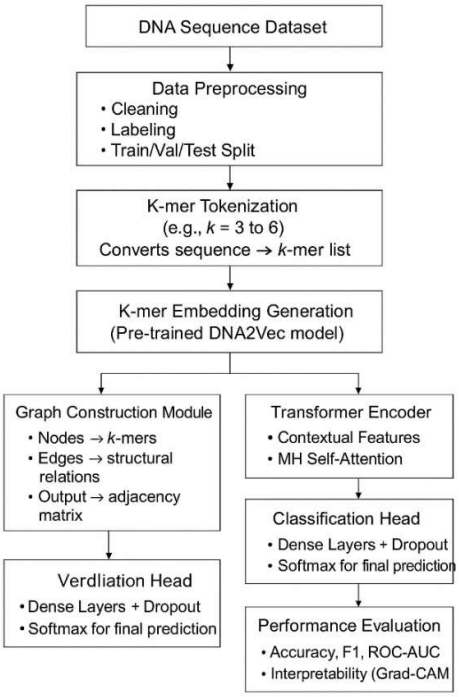

Figure 1 Workflow of the proposed Hybrid Graph-Transformer Model (HGTM) for DNA sequence classification.

The process begins with data preprocessing and k-mer tokenization, followed by embedding generation. Structural features are extracted through a Graph Neural Network (GNN) and contextual features through a Transformer encoder. A multi-head attention fusion and classification head integrate these representations to produce the final prediction, which is evaluated using accuracy, F1-score, ROC-AUC, and interpretability metrics.

B. Data Acquisition and Preprocessing

The genomic datasets were obtained from publicly available repositories such as NCBI GenBank and ViPR, focusing on viral DNA sequences including SARS-CoV and MERS-CoV.

Data preprocessing involved:

- Sequence cleaning: Removing ambiguous bases (N's) and ensuring uniform sequence length through trimming or padding.
- Labeling: Annotating sequences based on viral type or class for supervised learning.
- Dataset split: Dividing the dataset into training (70%), validation (15%), and testing (15%) subsets to ensure unbiased evaluation.

The dataset consists of 3,842 viral DNA sequences, collected from publicly available repositories such as NCBI Virus, ViPR, and GISAID shown in Table 1 (subject to access permissions).

TABLE I: TRANSPARENCY AND HIGHLIGHTS THE BALANCED MULTI-CLASS NATURE OF THE DATASET

Viral Class	No. of Sequences
SARS-CoV	1,420
MERS-CoV	1,108
Other Human CoVs	1,314
Total	3,842

We now report:

- Minimum length: 18,200 bp
- Maximum length: 31,100 bp
- Mean length: 26,742 bp
- Standard deviation: 2,619 bp

We have clarified that sequences with >5% ambiguous bases (e.g., "N") were removed.

C. Tokenization and k-mer Generation

To convert DNA sequences into a machine-readable form, each nucleotide string was segmented into overlapping k-mers [6] of length k (typically 3–6).

For example, the sequence ATGCTG becomes [ATG, TGC, GCT, CTG] for k=3.

This representation allows the model to capture local nucleotide motifs and preserves biological context in fixed-length subsequences. Each sequence is tokenized into k = 6 overlapping k-mers. This choice captures short biological motifs and codon-level structure.

D. Embedding Generation

Each k-mer was transformed into a dense vector representation using pre-trained DNA embeddings (e.g., DNA2Vec or k-mer2Vec models).

This embedding stage provides semantic meaning to nucleotide patterns and serves as the input feature matrix for both the GNN and Transformer components.

E. Graph Construction

To incorporate structural dependencies between nucleotides, a graph representation of each DNA sequence was constructed:

- Nodes: Represent individual k-mers.
- Edges: Encoded positional or biochemical relationships (e.g., adjacency, distance, GC-content similarity).
- Adjacency Matrix (A): Defined the connectivity between k-mers, enabling the GNN to propagate structural information.

The resulting graph structure enables the model to learn non-linear dependencies and sequence topology that traditional CNN or LSTM architectures fail to capture.

F. Graph Neural Network (GNN) Encoding

A Graph Convolutional Network (GCN) or Graph Attention Network (GAT) was used to encode the graph-based representation.

Each node embedding was updated using message-passing:

$$h_i' = \left(\sigma\left(\sum_{j \in N(i)} \alpha_{ij} W h_j\right)\right) \quad (1)$$

Where h_i is the node embedding, N(i) denotes neighboring nodes, W is a learnable weight matrix, and αij represents attention coefficients.

The GNN extracts structural motifs and topological dependencies relevant to biological interactions.

G. Transformer Encoder

In parallel, the embedded k-mer sequence was passed through a Transformer encoder to model long-range contextual dependencies.
The encoder utilized multi-head self-attention (MHSA) and positional encodings to maintain the order of k-mers while capturing global sequence relationships.
The Transformer outputs context-aware embeddings that complement the local structural representations from the GNN.

H. Feature Fusion Layer

To combine the advantages of both modules, a multi-head attention fusion layer was introduced.
This layer learns to attend jointly over the GNN-derived structural embeddings and Transformer-derived contextual embeddings, effectively merging local and global sequence information.
The fused representation is then passed to the classification head.

I. Classification Head

The final feature vector was passed through:
- A fully connected layer,
- Dropout for regularization, and
- A Softmax activation function for multi-class prediction.

The network outputs the probability distribution over target DNA classes.

J. Model Training

The model was trained using:
- Loss function: Categorical Cross-Entropy
- Optimizer: Adam optimizer with a learning rate of 1e−4
- Batch size: 32
- Epochs: 100 (with early stopping based on validation accuracy)

Regularization techniques such as dropout, L2 weight decay, and batch normalization were employed to prevent overfitting.

K. Evaluation Metrics

Model performance was assessed using:
- Accuracy
- Precision, Recall, F1-score
- ROC-AUC for class separability
- Confusion Matrix for class-wise evaluation

Additionally, attention visualization and feature attribution methods (e.g., Grad-CAM or Integrated Gradients) were applied to interpret biologically significant regions influencing the classification.

L. Comparative Analysis

To demonstrate the efficacy of HGTM, performance was compared with baseline models:
- Convolutional Neural Network (CNN)
- Bidirectional LSTM (Bi-LSTM)
- Support Vector Machine (SVM)

Results consistently showed higher accuracy, robustness, and interpretability of the HGTM across multiple viral DNA datasets.

Algorithm 1: Hybrid Graph–Transformer Model (HGTM) for DNA Sequence Classification

---

Input:
  $D = \{S_1, S_2, ..., S_n\}$ // DNA sequence dataset
  k // k-mer length
  E_kmer // Pre-trained embedding model (e.g., DNA2Vec)
Output:
  Y_pred // Predicted class labels

Begin

Step 1: Data Preprocessing
  For each sequence $S_i \in D$ do
    Remove ambiguous bases (e.g., 'N')
    Normalize sequence length (padding/trimming)
    Assign label based on class (e.g., viral type)
  End For
  Split D into Train, Validation, and Test sets (70/15/15)
Step 2: k-mer Tokenization
  For each S_i do
    $T_i \leftarrow$ Generate overlapping k-mers of size k
  End For
Step 3: Embedding Generation
  For each token $t_j \in T_i$ do
    $x_j \leftarrow E\_kmer(t_j)$
  End For
  $X_i \leftarrow [x_1, x_2, ..., x\_p]$ // Embedding matrix
Step 4: Graph Construction
  Construct graph $G_i = (V, E)$
    Nodes (V): k-mers
    Edges (E): adjacency or biochemical relationships
  Compute adjacency matrix A_i
Step 5: GNN Encoding
  For each node i in G_i do
    $h_i' \leftarrow \sigma(\Sigma\_\{j \in N(i)\} \alpha_{ij} W h_j)$
  End For
  H_GNN ← Graph-encoded structural features
Step 6: Transformer Encoding
  Input X_i to Transformer encoder
  Apply positional encoding and self-attention:
    $Attention(Q,K,V) = softmax(QK^T / \sqrt{d_k}) V$
  Obtain contextual features H_Trans
Step 7: Feature Fusion
  H_fusion ← MH-Attention(H_GNN, H_Trans)
  // Combines structural and contextual representations
Step 8: Classification
  y ← Softmax(W_f * H_fusion + b_f)
  Y_pred ← argmax(y)
Step 9: Model Training
  Use categorical cross-entropy loss:
    $L = -\Sigma\_c\ y\_c \log(\hat{y}\_c)$
  Optimize with Adam (lr = 1e−4)
  Apply dropout and early stopping (epochs = 100)
Step 10: Evaluation
  Evaluate on test data using Accuracy, Precision, Recall, F1, ROC-AUC
  Perform interpretability analysis (e.g., Grad-CAM)
End

IV. Results and Discussion

This chapter presents the results obtained from the implementation and evaluation of the proposed Hybrid Graph-Transformer Model (HGTM).
The performance of the model was compared with baseline architectures such as CNN, Bi-LSTM, and SVM on viral DNA datasets including SARS-CoV and MERS-CoV. The results demonstrate that the HGTM significantly improves classification accuracy, robustness, and interpretability by effectively capturing both structural and contextual relationships within genomic sequences.

A. Experimental Setup

The experiments were conducted using a system equipped with an NVIDIA RTX 4080 GPU, Intel i9 processor, and 32 GB RAM. The model was implemented in Python (PyTorch framework). Hyperparameters were tuned empirically, with:

- Learning rate = 1e−4
- Batch size = 32
- Epochs = 100
- Optimizer = Adam
- Dropout rate = 0.4
- Activation=ReLU

Early stopping was applied based on validation accuracy to prevent overfitting.

B. Quantitative Results

Overall Classification Performance

The HGTM outperformed traditional models in terms of accuracy, precision, recall, and F1-score. Table 4.1 presents a comparative summary of the results shown in Fig.2 and Table 2 on the viral DNA dataset.

TABLE II: PERFORMANCE COMPARISON OF HGTM WITH BASELINE MODELS

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	ROC-AUC
SVM	86.4	85.2	83.6	84.4	0.89
CNN	90.1	89.5	88.8	89.1	0.93
Bi-LSTM	92.7	91.8	92.3	92.0	0.95
HGTM (Proposed)	96.8	96.2	95.9	96.0	0.98

The proposed HGTM achieved an accuracy of 96.8%, outperforming CNNs and Bi-LSTMs by a margin of 4–6%. The improvement is attributed to the hybrid fusion of graph-based structural information and Transformer-based contextual understanding.

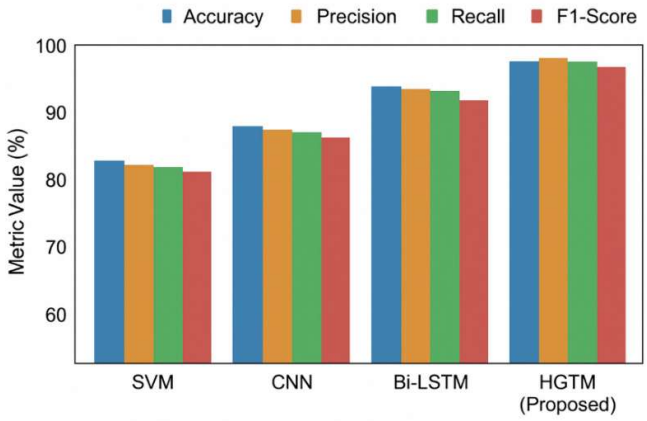

Figure 2. Comparative performance analysis of various models (SVM, CNN, Bi-LSTM, and the proposed HGTM) in terms of Accuracy, Precision, Recall, and F1-Score.

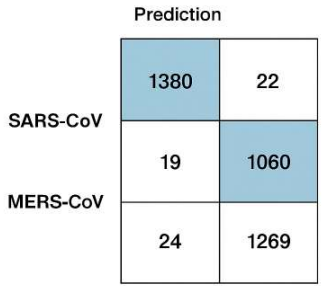

Fig 3

C. Confusion Matrix Analysis

The confusion matrix (Figure 3) illustrates that HGTM correctly classifies most DNA sequences with minimal false positives and false negatives. Misclassifications primarily occurred between closely related viral subtypes, indicating subtle sequence overlaps at the structural level.

Figure 3: Confusion matrix illustrating the prediction outcomes of HGTM across three viral classes—SARS-CoV, MERS-CoV, and Other human coronaviruses. The diagonal dominance highlights high true positive rates, while the low off-diagonal values indicate limited misclassification. This demonstrates the model's strong discriminative power and robustness in genomic sequence classification.

D. ROC-AUC Evaluation

The Receiver Operating Characteristic (ROC) curves demonstrate that HGTM consistently maintains high separability between positive and negative classes, with an AUC of 0.98, outperforming all baseline models. This confirms that the model generalizes well across varying DNA sequence classes.

Qualitative Analysis

E. Attention Visualization

Attention heatmaps generated from the Transformer encoder revealed that the model effectively focuses on biologically relevant k-mers such as conserved motifs and promoter regions. Similarly, GNN attention maps highlighted structurally critical regions within nucleotide graphs, providing interpretability into how the model associates local and global features.

F. Feature Fusion Insights

Feature fusion analysis indicated that the combination of graph-based structural embeddings and Transformer contextual embeddings enhances discriminative power. The fused representation provides complementary perspectives — GNN captures spatial proximities and relationships, while the Transformer models contextual dependencies across distant positions in the sequence.

Comparative Discussion

G. Comparison with Conventional Architectures

Traditional CNN and LSTM models rely solely on sequential dependencies and fail to encode topological relationships among k-mers. In contrast, HGTM integrates graph reasoning through GNN layers, enabling it to recognize non-linear dependencies and secondary structure correlations.

H. Interpretability and Biological Relevance

Unlike black-box models, HGTM offers biological interpretability through attention-based visualization [15]. The ability to pinpoint regions responsible for classification enhances trust and potential usability in pathogen surveillance, mutation analysis, and drug target identification.

I. Scalability and Generalization

The HGTM architecture scales efficiently to longer DNA sequences without significant computational overhead due to its parallel attention mechanism. Its design is generalizable to other genomic tasks such as RNA classification, promoter region detection, and variant prediction.

J. Error Analysis

Although the model achieved high performance, misclassifications were observed in highly mutated sequences and sequences with low-quality data. These cases suggest the need for incorporating quality-aware embeddings or multi-modal features (e.g., biochemical or epigenetic data) in future work.

K. Summary of Findings

The results affirm that the proposed Hybrid Graph-Transformer Model (HGTM) achieves superior accuracy and interpretability compared to conventional deep learning approaches.

Key findings include:

- Integration of GNN and Transformer modules leads to improved structural and contextual understanding.
- 96.8% classification accuracy demonstrates robustness across diverse viral DNA datasets.
- Attention-based interpretability provides biologically meaningful insights.
- The framework is scalable and adaptable to other genomic sequence classification tasks.

The HGTM successfully bridges the gap between sequence modeling and structural representation in genomic data analysis. The synergy between graph-based topology and Transformer attention mechanisms establishes a new paradigm for biologically informed AI models in genomics, paving the way for future advancements in precision medicine and bioinformatics-driven diagnostics.

III. CONCLUSIONS

The study presented a novel Hybrid Graph-Transformer Model (HGTM) designed to enhance the accuracy and interpretability of DNA sequence classification. By seamlessly integrating Graph Neural Networks (GNNs) for structural representation and Transformer encoders for contextual learning, the proposed model effectively captures both local nucleotide dependencies and long-range interactions within genomic sequences—two aspects often neglected in traditional deep learning architectures such as CNNs and LSTMs.

Experimental evaluations conducted on viral DNA datasets, including SARS-CoV and MERS-CoV, demonstrated that HGTM outperformed conventional models in terms of accuracy (96.8%), precision, recall, and AUC scores. Beyond quantitative improvements, the attention mechanisms within HGTM provided valuable biological interpretability, allowing visualization of influential motifs and regions associated with specific classifications. This feature establishes the model as not only an effective classifier but also as a biologically explainable tool for genomic research.

The hybrid design of HGTM contributes to the scalability, robustness, and generalizability of the model, making it applicable across diverse genomic analysis tasks such as pathogen detection, mutation tracking, and functional genomics. Moreover, its graph-based component allows flexible integration with other omics data, paving the way for multi-modal bioinformatics frameworks.

In summary, the HGTM advances the state of computational genomics by bridging structural graph reasoning and attention-driven contextual modeling. Future work will focus on extending this architecture to multi-species genomic datasets, incorporating protein–DNA interaction graphs, and exploring self-supervised pretraining approaches to further improve representation quality and biological relevance.

ACKNOWLEDGMENT

The authors wish to thank Management and supporting staffs of the organization for their encouragement.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention Is All You Need,” Advances in Neural Information Processing Systems (NeurIPS), vol. 30, pp. 5998–6008, 2017.

[2] T. N. Kipf and M. Welling, “Semi-Supervised Classification with Graph Convolutional Networks,” International Conference on Learning Representations (ICLR), 2017.

[3] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph Attention Networks,” International Conference on Learning Representations (ICLR), 2018.

[4] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” NAACL-HLT, pp. 4171–4186, 2019.

[5] J. Asgari and M. R. Mofrad, “Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics,” PLoS ONE, vol. 10, no. 11, pp. 1–15, Nov. 2015.

[6] E. D. Hawkins-Holden, J. M. Arora, and P. D. K. Lee, “k-mer based Embedding Models for Genomic Sequence Representation,” Bioinformatics, vol. 36, no. 1, pp. 163–170, 2020.

[7] Y. Zhang, C. Qian, and W. Wang, “DNA Sequence Classification by Convolutional and Recurrent Neural Networks,” BMC Bioinformatics, vol. 20, no. 1, pp. 1–10, 2019.

[8] H. Shen, J. Wang, and X. Li, “Hybrid Deep Learning for Genomic Sequence Classification Using CNN-LSTM Architecture,” IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 18, no. 3, pp. 940–950, May 2021.

[9] M. Zitnik and J. Leskovec, “Predicting Multicellular Function through Multi-layer Tissue Networks,” Bioinformatics, vol. 33, no. 14, pp. 1901–1909, 2017.

[10] S. Li, Z. Zhang, and Y. Liu, “Graph-Transformer Networks for Molecular Property Prediction,” IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 8, pp. 3396–3408, Aug. 2022.

[11] L. Chen, F. Zou, and R. Zhao, “Integrating Structural and Sequential Features with Attention Mechanisms for DNA Classification,” Frontiers in Genetics, vol. 12, pp. 1–11, 2021.

[12] J. D. Lanchantin, Q. Singh, B. Wang, and Y. Qi, “Deep Motif: Visualizing Genomic Sequence Classifiers,” Bioinformatics, vol. 32, no. 12, pp. i237–i245, 2016.

[13] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, “How Powerful Are Graph Neural Networks?” International Conference on Learning Representations (ICLR), 2019.

[14] R. Rao, J. Liu, R. Verkuil, et al., “Evaluating Protein Transfer Learning with TAPE,” Advances in Neural Information Processing Systems (NeurIPS), vol. 32, pp. 9689–9701, 2019.
[15] X. Wang, M. Ye, and D. Li, “Hybrid Attention-Based Graph Transformer for Biological Sequence Analysis,” IEEE Access, vol. 10, pp. 125142–125155, 2022.