RESEARCH ARTICLE OPEN ACCESS

DNA Sequence-Based Disease Classification with Caduceus: A Bi-Directional Equivariant Transformer with GAN Augmentation and SHAP Explainability

Prof. Roopa B, Vaishakh M P, Yashvardhan Chougule, Rokkam Akash, Yogeesh G
Bangalore Institute of Technology, Bangalore, Karnataka, India

----------**********----------

Abstract
Genomic information is increasingly used for diagnosis, preventive medicine and personalized treatment planning. However, converting raw DNA sequences into accurate, interpretable disease predictions remains difficult because genomic data is high-dimensional, sparse and shaped by long-range, non-linear interactions. Disease datasets are also small and imbalanced, particularly for rare diseases, and many deep learning models behave as black boxes that clinicians hesitate to trust. This paper presents a DNA sequence-based disease classification system built on Caduceus, a bi-directional, reverse-complement-equivariant long-range DNA language model. The pretrained Caduceus backbone is fine-tuned with a custom three-layer classification head for both single-disease (binary) and multi-disease (multi-label) prediction. A SeqGAN-style generative module, with an LSTM generator, a CNN discriminator and a rollout mechanism, produces synthetic DNA sequences to reduce class imbalance. Class-weighted binary cross-entropy and threshold tuning further support rare-disease learning. SHAP-based explanations identify the nucleotide positions that influenced each prediction, and a web interface returns predictions, confidence scores, visual explanations and a downloadable report. On the evaluated dataset (sequences of 98–104 bp), Caduceus achieved an F1-Micro of 0.6349 on 15-label classification and 0.5905 on 100-label classification, outperforming DNABERT (0.5408) and a CNN–BiLSTM baseline (0.1168), and reached an F1-Micro of 0.9094 on binary classification.

Keywords: DNA Sequence Classification, Disease Prediction, Caduceus, Transformer, Generative Adversarial Network, Data Augmentation, SHAP, Explainable AI, Multi-Label Classification.

----------**********----------

I. Introduction
Advances in genomics have greatly improved our understanding of the human genome and of the genetic variants linked to complex diseases. With the falling cost of whole-genome and exome sequencing, DNA-based diagnostics are becoming common in clinical settings. The harder problem is turning raw genomic data into predictions that are accurate, interpretable and scalable. Traditional machine learning models struggle with genomic data because of its high dimensionality, sparsity and the long-range, non-linear interactions between genetic elements [2].

These difficulties are most visible in polygenic disorders, where many genes contribute to disease development. The scarcity of balanced genomic datasets, especially for rare diseases, complicates the development of reliable models further. In addition, many existing tools concentrate on a single task, such as SNP prediction or mutation-impact analysis, and do not offer multi-disease classification from long DNA sequences [13]. Their black-box nature also limits clinical trust, because a clinician cannot check a prediction against biological knowledge when the model offers no explanation.

This work proposes a web-based AI platform for disease classification from DNA sequences. It combines a bi-directional equivariant transformer, inspired by DNABERT [2] and Caduceus [6], with a GAN-based augmentation module that generates synthetic sequences to counter data imbalance [3]–[5]. SHAP values
[11] are integrated so that users can see which sequence regions drove each prediction.

The main contributions of this work are:

(1) a fine-tuned Caduceus classifier with a custom classification head that supports both

 ©IJSRED: All Rights are Reserved

binary and multi-label disease prediction; (2) a SeqGAN-style augmentation module for rare-disease classes; (3) class-weighted training with decision-threshold tuning for imbalanced labels; (4) nucleotide-level SHAP explanations with several visualization types; (5) a web application that returns predictions, confidence scores and a downloadable report; and (6) a comparison against DNABERT and a CNN–BiLSTM baseline on the same data and metrics.

The remainder of the paper is organized as follows. Section II gives background, Section III reviews related work, Section IV states the problem, Section V describes the system and methodology, Section VI presents the experimental setup, results and the web application, Section VII discusses applications and limitations, and Section VIII concludes.

II. Research Background

Early computational approaches to disease prediction from sequence data relied on classical models such as Support Vector Machines and Random Forests. These depend on manual feature selection, which limits their ability to generalize across genomic contexts. Convolutional and recurrent networks reduced this burden by learning features automatically, but they are limited in the sequence length they can model and therefore miss long-range genomic interactions.

Transformer-based DNA language models treat DNA as a language. DNABERT applies BERT-style pre-training to k-mer tokens of the human genome and is then fine-tuned for downstream tasks such as promoter and regulatory-element prediction [2]. Attention cost grows quickly with sequence length, which limits context. Caduceus addresses this with a structured state-space (Mamba) backbone, bi-directional processing (BiMamba) and reverse-complement equivariance (MambaDNA), which reflects the fact that the two DNA strands carry equivalent information [6].

Data scarcity and class imbalance are handled in other medical domains with GAN-based augmentation, which has improved classification of liver lesions, arrhythmias and visual-inspection defects [3]–[5]. Sequence-level GANs face a particular difficulty: a discriminator can only score a complete sequence, so the generator receives a sparse reward. Reinforcement-learning style training with Monte-Carlo rollouts is one way to estimate rewards for partial sequences.

Explainability is addressed by attribution methods such as SHAP [11] and DeepLIFT [12], which assign importance scores to input features. In genomics, these scores can be mapped back to nucleotide positions so that experts can compare highlighted regions with known biology. This project combines long-range DNA modelling, generative augmentation and explainability in one pipeline.

III. Literature Review

Several studies have investigated deep learning for genomics, generative augmentation and interpretable prediction.

Sai et al. (2024) surveyed generative AI in healthcare, covering emerging models, applications, case studies and limitations. The study highlights synthetic data generation among other uses, and notes that dependence on large, high-quality, well-annotated medical datasets and concerns about privacy and bias remain open challenges [1].

Ji et al. (2021) proposed DNABERT, a BERT-based model pre-trained on the human genome using k-mer tokenization and fine-tuned for promoter, transcription-factor binding site and splice-site prediction. Pre-training is computationally expensive and the biological meaning of some attention-highlighted regions remains unclear [2].

Frid-Adar et al. (2022), Düzyel and Kuntalp (2023) and Rožanec et al. (2022) showed that GAN-based synthetic data improves classification on small or unbalanced datasets in liver-lesion classification, arrhythmia classification and automated visual inspection respectively [3]–[5].

Schiff et al. (2024) introduced Caduceus, a DNA language model built on the Mamba block. It uses BiMamba for bi-directional modelling and MambaDNA for reverse-complement equivariance, and outperforms CNN, HyenaDNA and transformer baselines on genomics and Nucleotide Transformer benchmarks with far fewer parameters. The variants are more complex to implement and may not win on every task [6].

Mathur (2024) proposed a CNN-based DNA classifier using hot-vector encoding that preserves positional nucleotide information. It reached up to 97.3% accuracy across respiratory-virus, cancer and HPV datasets, but relied on empirically chosen hyperparameters and curated data [7].

Jia et al. (2024) presented ModulePred, which predicts disease–gene associations by combining functional modules, PPI-network augmentation and

 ©IJSRED: All Rights are Reserved

graph neural networks. Its performance depends on the completeness and quality of the biological network data [8].

Saadat and Fellay (2024) combined a DNA language model with an interpretable graph neural network to identify genes and pathways involved in rare diseases [9]. Sharma et al. (2024) reviewed machine learning approaches to DNA sequence classification and analysis [10].

Lundberg and Lee (2017) introduced SHAP, a unified framework for explaining model predictions [11], and Shrikumar et al. (2017) proposed DeepLIFT, which propagates activation differences to attribute importance [12]. Poplin et al. (2018) demonstrated a deep-neural-network variant caller for SNPs and small indels [13], and Zhang et al. (2023) applied graph neural networks to disease–gene prediction [14].

Research Gap. Existing work either classifies sequences without explanation, explains predictions without handling data imbalance, or targets a single narrow task such as SNPs or gene association. Few systems combine long-range DNA modelling, multi-disease classification, generative augmentation of rare classes and nucleotide-level explainability in one deployable tool.

IV. Problem Statement

The objective is to design and develop a robust DNA sequence-based disease classification system using Caduceus, a bi-directional equivariant model that captures complex genomic patterns, augmented with GAN-generated synthetic data to address class imbalance and improve model performance.

The input is a DNA sequence submitted through a web interface as plain text or a FASTA file. The output is the associated disease classification with a confidence score, a SHAP-based visual explanation highlighting influential DNA regions, and a downloadable PDF report for clinical or research use.

Existing approaches fall short in four ways. Classical models depend on manual feature selection. CNNs, RNNs and standard transformers are constrained by sequence length. Many tools handle SNPs well but not complex mutations, and they offer little explanation. Finally, small and imbalanced datasets limit training, and few pipelines use generative augmentation to compensate.

V. System Architecture and Methodology

The system combines genomic preprocessing, transformer-based disease prediction and explainable AI. It supports two modes: binary prediction (positive or negative) and multi-label prediction (several diseases at once). Figure 1 shows the overall pipeline: real sequences are formatted as CSV, imbalance is addressed with GAN augmentation, sequences are tokenized, the Caduceus model is fine-tuned with a new classification head, and predictions are explained with SHAP and compared against the baselines.

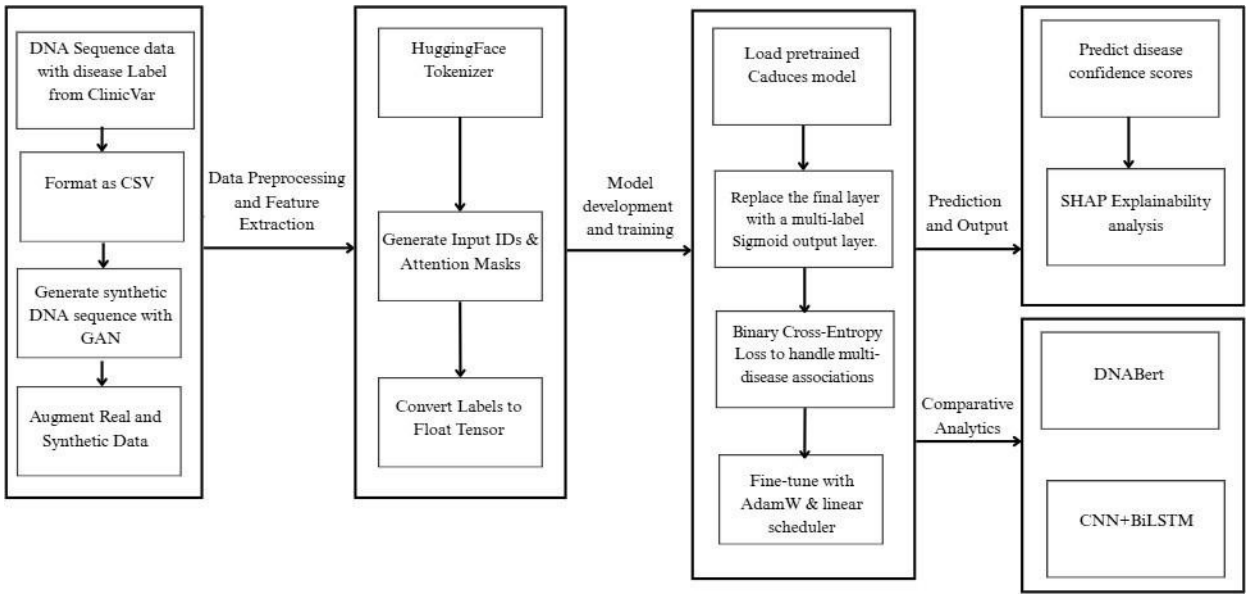


Fig. 1: System architecture of the proposed DNA disease classification system

 ©IJSRED: All Rights are Reserved

1. GAN-Based DNA Sequence Generation

This module generates synthetic DNA sequences to augment the real dataset and reduce class imbalance. The generator is an LSTM with embedding layers, pretrained with Maximum Likelihood Estimation to avoid early instability and then fine-tuned with policy-gradient reinforcement learning. The discriminator is a CNN with convolution layers, max-pooling and a sigmoid output that separates real from synthetic sequences. A rollout mechanism samples multiple sequence completions and averages the discriminator's scores to provide intermediate rewards, easing the sparse-reward problem of sequence-level GANs. Label smoothing, dropout and a replay buffer stabilize training. The generated sequences supplement rare disease categories.

The generation algorithm proceeds in the following stages. The generator is pretrained with the MLE loss $L_{MLE} = \square\square \log p_{\square}(x_t \mid x_{t\square 1})$, and the discriminator is pretrained with binary cross-entropy. During adversarial training, each incomplete sequence is completed N times by Monte-Carlo rollout, and the

(i)

reward is the average discriminator score, $r_t = (1/N) \square D(x$). The generator is updated by policy gradient

using this reward, and an exponential moving average stabilizes the updates. Biological checks are then

applied: nucleotide probabilities must sum to one, and a sequence is accepted only if its GC-content, (G+C)/(A+T+G+C), lies within an acceptable range. Valid synthetic sequences are appended to the real data to form the augmented dataset.

2. Data Preprocessing and Dataset Handling

Sequences are tokenized with the HuggingFace Caduceus tokenizer, which produces input IDs and attention masks. Sequences are padded or truncated to a maximum of 512 tokens. If the tokenizer cannot process a sequence, a fallback vocabulary of the nucleotides A, C, G, T (and N) is used, so every sequence can be encoded. Labels are encoded as a single binary value for single-label prediction or as multi-hot vectors over the top disease categories (up to 100) for multi-label prediction. Each batch consists of the token tensor X, the attention mask M (1 for real tokens, 0 for padding) and the label vector Y. Real and synthetic sequences are stored in CSV format as the standardized input to training.

3. Disease Classifier

The pretrained Caduceus model serves as the backbone for feature extraction. Hidden states are mean-pooled over tokens, with the attention mask excluding padding, to give a fixed-length sequence embedding. The embedding passes through a custom three-layer classification head made of linear projections, Layer Normalization, ReLU activations and dropout, with Xavier initialization. The head outputs one logit per disease class, and a sigmoid converts logits to probabilities. For binary prediction the head has a single output neuron. Transfer learning from the pretrained backbone lets the classifier reach useful accuracy with limited task-specific data.

4. Training and Evaluation

Training uses BCEWithLogitsLoss, with positive-class weights (pos_weight) computed from the label distribution so that rare diseases contribute more to the loss. Optimization uses AdamW with linear warmup and decay, and gradient clipping prevents exploding gradients on long sequences. At evaluation, sigmoid outputs are converted to binary predictions by tuning decision thresholds in the range 0.1–0.7, which improves micro-F1 and the precision–recall trade-off. Performance is reported with Precision, Recall, F1-Micro and F1-Macro.

5. SHAP-Based Interpretation

SHAP computes an attribution value for each token to show which nucleotide positions most influenced a prediction. The sequence is tokenized as in training, the model is frozen in evaluation mode, and SHAP values are computed against a reference baseline. Values are summed across embedding dimensions, normalized and mapped back to nucleotide positions. Because SHAP is expensive on long sequences and multi-label outputs, the module combines sampling-based and gradient-based variants, applies smoothing across sliding windows of nucleotides, and uses nucleotide sub-sampling, mini-batching and GPU acceleration. Results are shown as force, waterfall, decision, bar, beeswarm, heatmap and genomic bar plots.

6. Baseline Models

The Caduceus classifier is compared with two baselines. DNABERT uses overlapping k-mer

 ©IJSRED: All Rights are Reserved

tokenization with a maximum length of 512 and a fine-tuned classification layer, trained with mixed precision (FP16) [2]. The second baseline is a hybrid CNN–BiLSTM operating on precomputed DNA embeddings, in which convolution layers extract local patterns and BiLSTM layers capture sequence dependencies. Both use the same multi-hot labels and metrics as the main classifier, so the comparison is consistent.

7. Web Application

The user interface is built with React and provides fields for a patient identifier, the classification mode (single-label or multi-label) and the DNA sequence, together with a sequence-length indicator for the supported range of 98–104 nucleotides. The backend runs a FastAPI server inside Google Colab so that it can use a GPU, and an ngrok tunnel exposes it to the frontend. On submission the backend validates the sequence, tokenizes it, loads the binary or multi-label head of the fine-tuned model, returns probabilities, and applies SHAP to highlight important positions. The system is implemented in Python with PyTorch, HuggingFace Transformers, SHAP and scikit-learn.

VI. Experiments and Results

A. Experimental Setup

All experiments were run in PyTorch on Google Colab Pro+ with an NVIDIA T4 GPU. The dataset was standardized to a length range of 98–104 bp, tokenized with the Caduceus tokenizer and augmented with SeqGAN to balance minority disease classes. Three label settings were evaluated: single-label (binary), 15-label and 100-label classification. The best Caduceus configuration used a 256-dimensional embedding, 16 transformer layers and 8 attention heads; smaller models underfit, while larger ones added noise and generalized worse.

Precision is TP / (TP + FP) and Recall is TP / (TP + FN). F1-Micro pools true positives, false positives and false negatives across all labels, so it emphasizes frequent diseases. F1-Macro averages the per-disease F1 scores, so it is sensitive to rare diseases.

B. Model Comparison

Table 1: Comparison of all models

Model	Diseases	F1-Micro	F1-Macro	Precision	Recall
Caduceus	100	0.5905	0.5867	0.5537	0.6519
Caduceus	15	0.6349	0.6112	0.5843	0.7013
DNABERT	15	0.5408	0.4855	0.5500	0.5320
CNN–BiLSTM	15	0.1168	0.1168	0.0880	0.1733
Caduceus (Binary)	1	0.9094	0.8220	0.8078	0.8367

In the 15-disease multi-label setting, Caduceus achieved an F1-Micro of 0.6349 and an F1-Macro of 0.6112. DNABERT reached 0.5408 and 0.4855, a moderate result that reflects good short-motif modelling but limited long-range context, with lower recall and lower macro-level scores. The CNN–BiLSTM baseline performed worst (F1-Micro 0.1168), as local convolutions and recurrence did not integrate distant nucleotide dependencies. In the harder 100-disease setting, with highly imbalanced and sparse labels, Caduceus still held an F1-Micro of 0.5905. The binary task reached an F1-Micro of 0.9094, which is expected given its lower complexity and minimal imbalance. Class weighting improved recall on rare diseases.

C. Training Behaviour

Figures 2 to 4 show precision, recall and F1 (micro and macro) over training epochs. Caduceus on 100 labels improves steadily over the first epochs and then fluctuates around a plateau. On 15 labels, all four metrics rise smoothly and remain close together. DNABERT improves gradually but shows more variation at later epochs, and the CNN–BiLSTM curves stay low throughout. For the binary task, the metrics rise quickly in the first few epochs and then stabilize.

 ©IJSRED: All Rights are Reserved

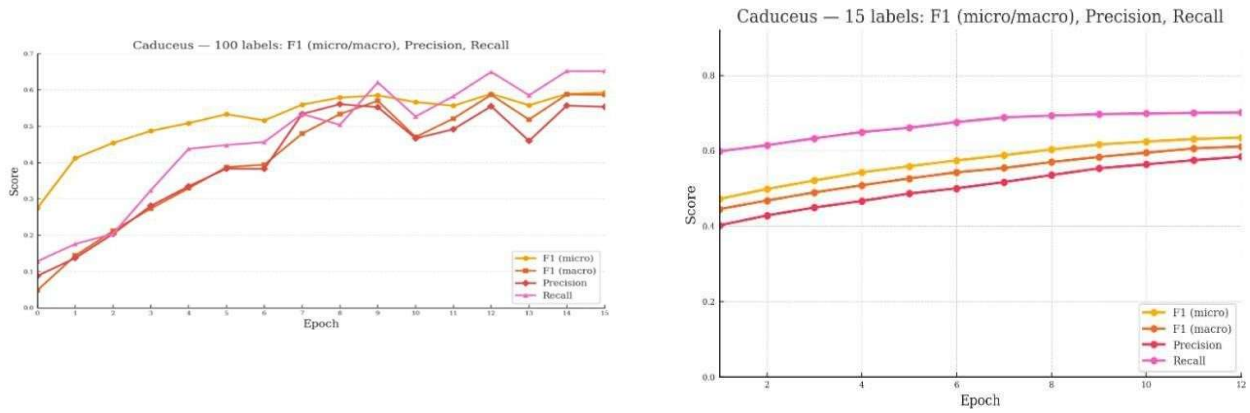


Fig. 2: Caduceus training metrics over epochs: 100-label (left) and 15-label (right)

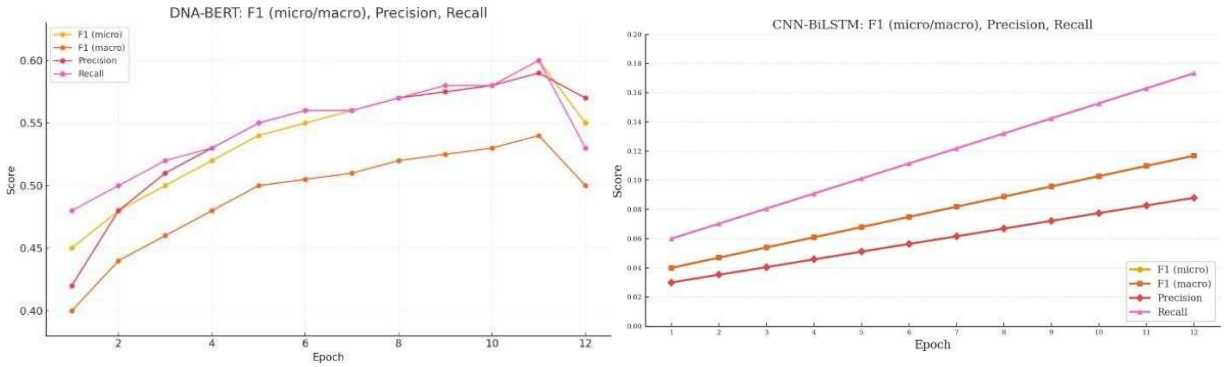


Fig. 3: Baseline training metrics over epochs: DNABERT (left) and CNN–BiLSTM (right), 15-label

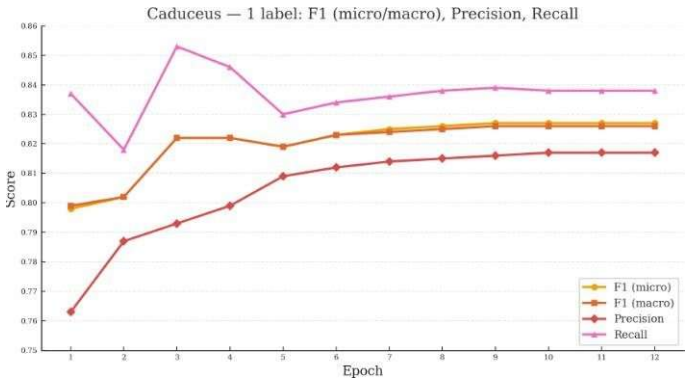


Fig. 4: Caduceus single-label (binary) training metrics over epochs

D. Explainability Results

The SHAP module produced nucleotide-level importance maps for each predicted disease. In a single-label example, a single position (Pos80, T) was identified as the most influential nucleotide, and the waterfall, bar and decision plots agreed on this ranking. In the waterfall plot, nucleotides that increase the predicted probability are drawn in red and those that decrease it in blue, ending at the final model output. The heatmap and genomic bar plot map SHAP values along the whole sequence, so a user can see where risk-increasing and risk-reducing positions cluster.

 ©IJSRED: All Rights are Reserved

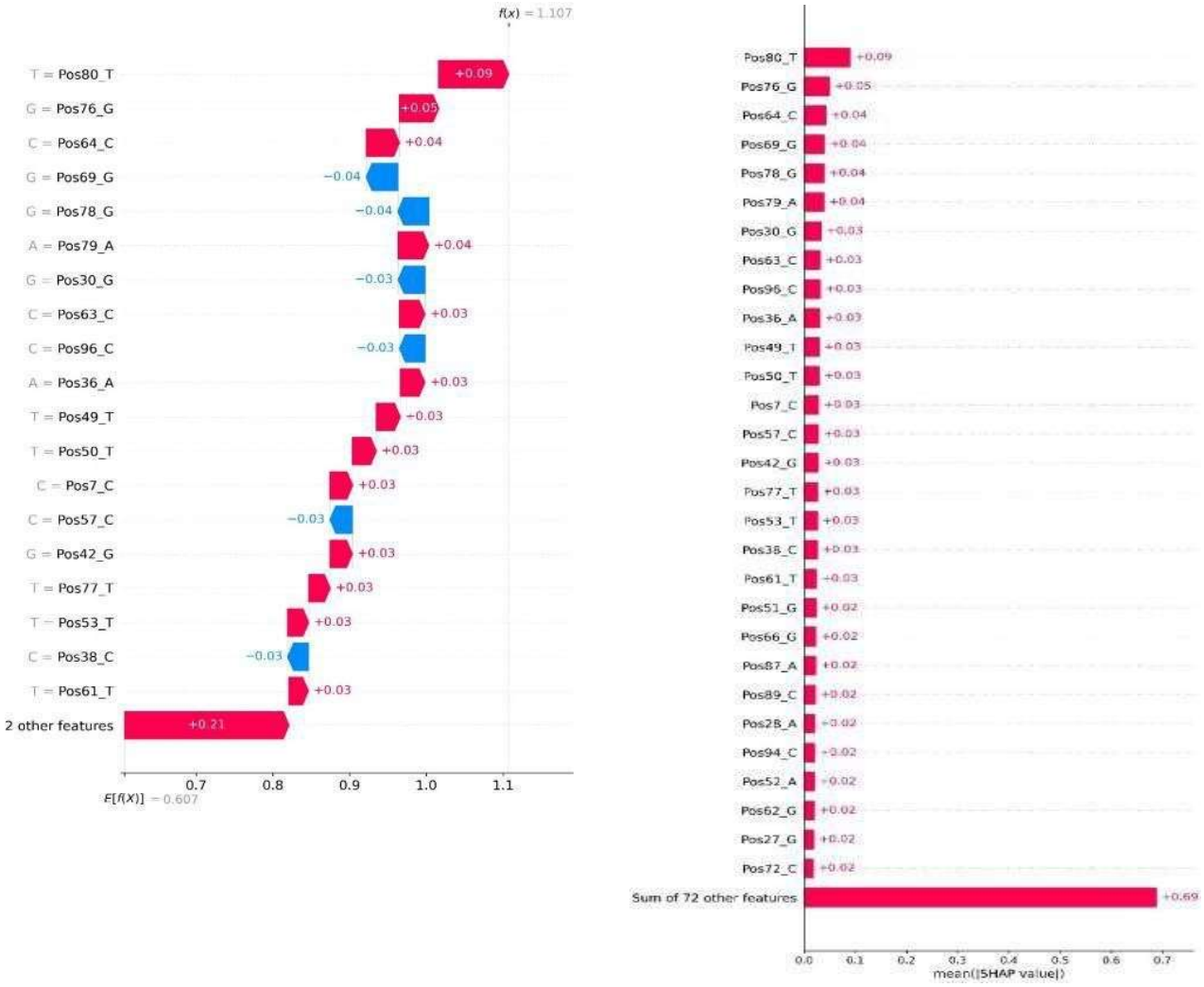


Fig. 5: SHAP waterfall plot (left) and bar plot (right) for a single-label prediction

 ©IJSRED: All Rights are Reserved

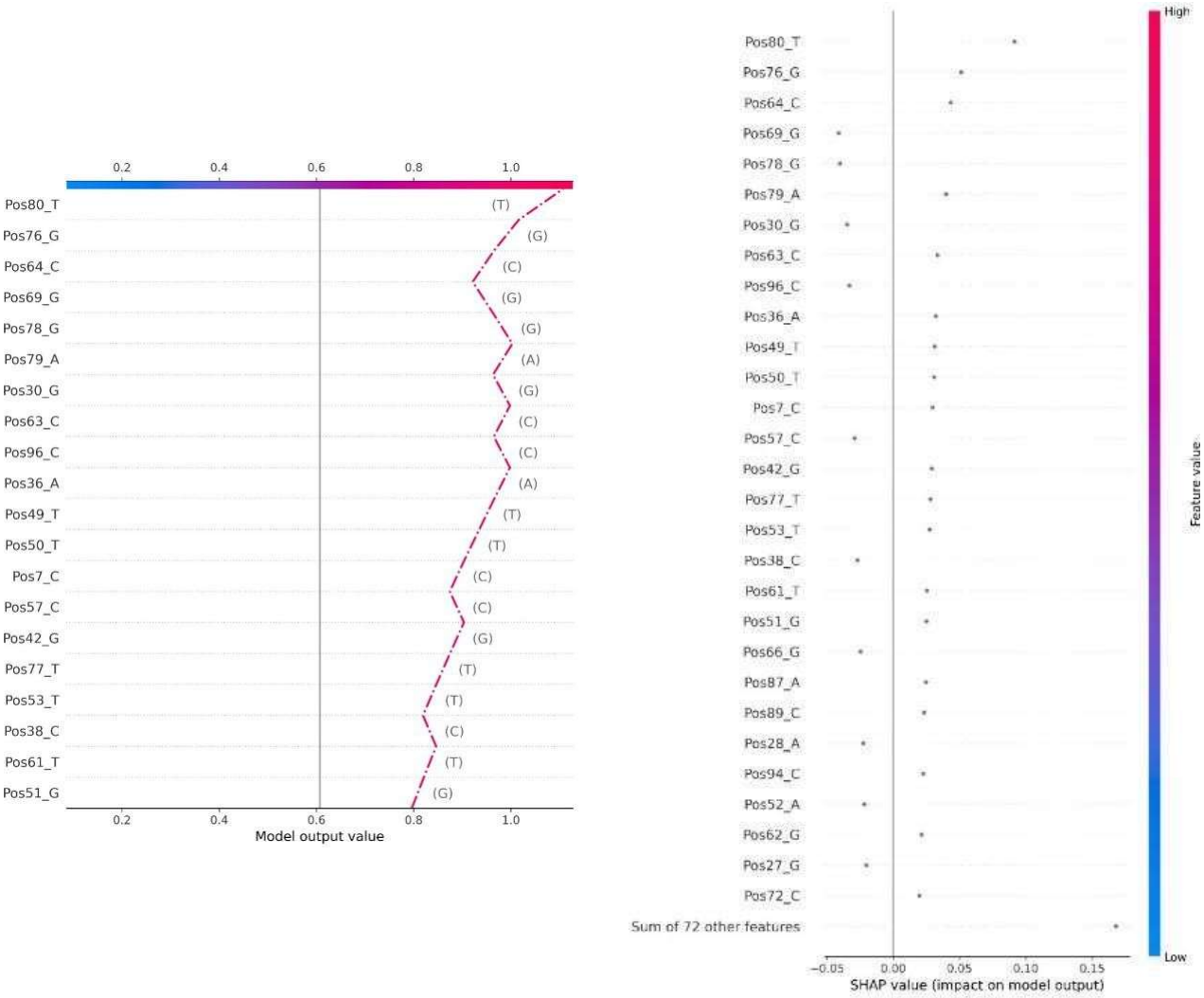


Fig. 6: SHAP decision plot (left) and beeswarm plot (right)

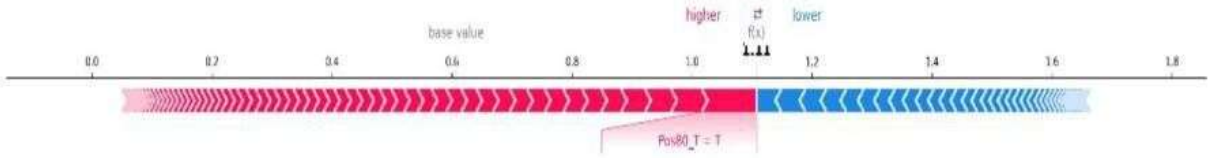


Fig. 7: SHAP force plot; red marks risk-increasing and blue marks risk-reducing nucleotides

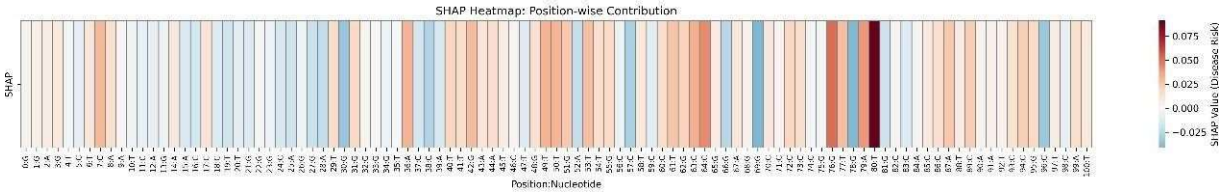


Fig. 8: SHAP heatmap of position-wise importance along the sequence

 ©IJSRED: All Rights are Reserved

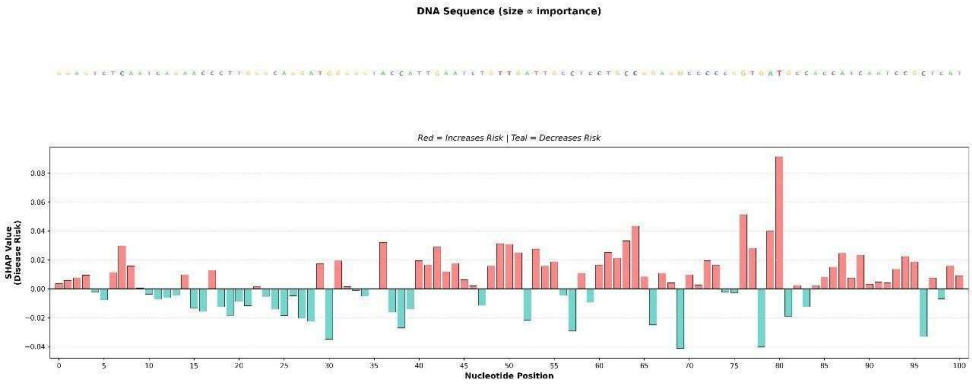


Fig. 9: Genomic bar visualization of SHAP values across the sequence

E. Web Application

The home page lets the user enter an optional patient ID, choose single-label or multi-label mode and paste a DNA sequence (Fig. 10). The single-label result page reports a Positive or Negative outcome with a confidence score, and the user can then run SHAP analysis to see a summary of the predicted probability, sequence composition and the positions that raised or lowered risk (Fig. 11). In multi-label mode, the system ranks diseases by probability and lists those above the decision threshold as detected conditions (Fig. 12).

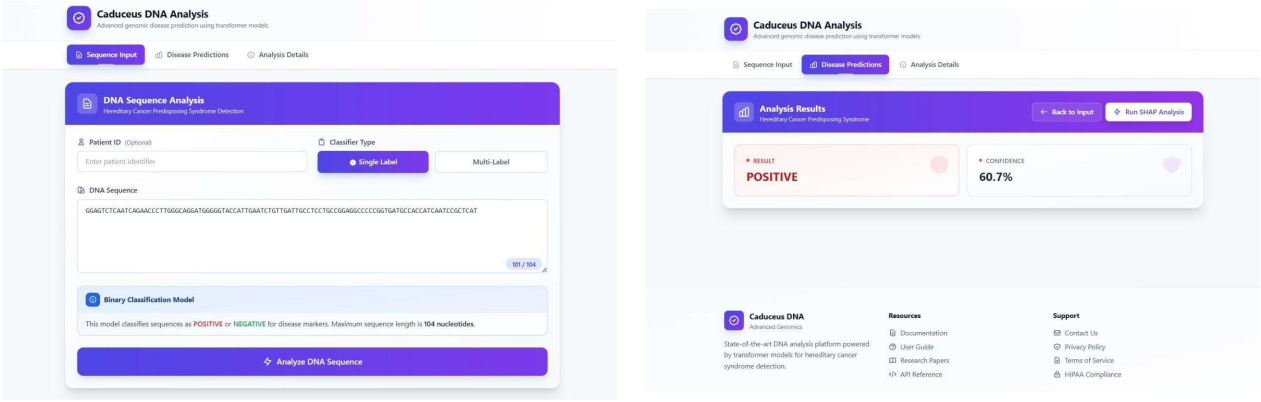


Fig. 10: Home page (left) and single-label prediction page (right)

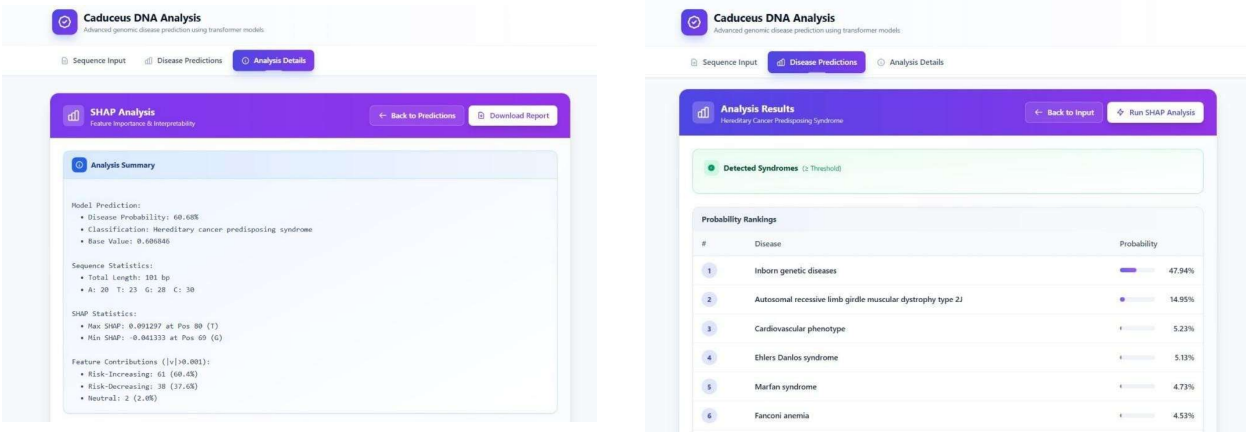


Fig. 11: SHAP analysis summary (left) and multi-label prediction ranking (right)

 ©IJSRED: All Rights are Reserved

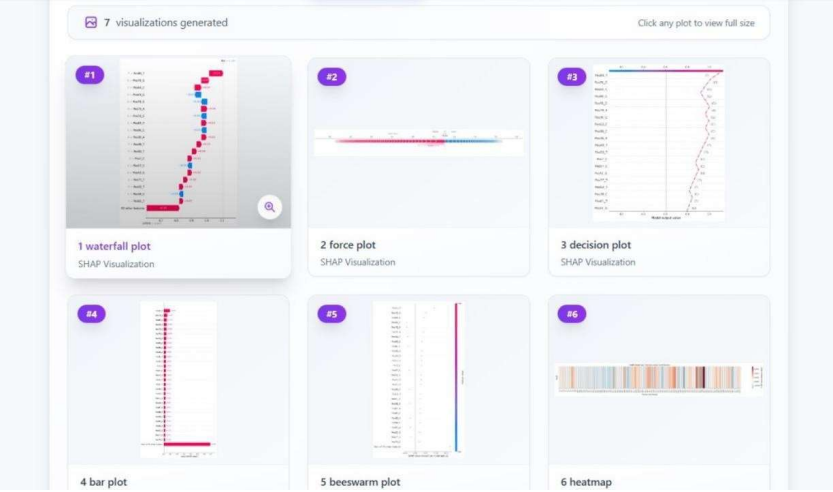


Fig. 12: Visualizations page with the generated SHAP plots

F. Testing

The system was tested at unit and integration level. Unit tests covered the GAN module, preprocessing and tokenization, and the Caduceus classifier. Valid inputs produced valid sequences, consistent tensors and correct probabilities, and invalid characters, missing model weights and over-length sequences were handled with errors, warnings or truncation as expected. Three cases failed and point to known weaknesses: the GAN showed instability in a mode-collapse scenario and needed retraining, the model crashed on a mismatched embedding size, and SHAP exceeded memory limits on some sequences longer than 512 nucleotides. Integration testing confirmed that SHAP correctly processed model outputs for 200 multi-class DNA samples and skipped computation with a warning when the model output file was missing or corrupted.

VII. Discussion, Applications and Limitations

Caduceus gave the best results in every comparison, which supports the view that long-range and bi-directional modelling matters for sequence-level disease classification. Fewer parameters than conventional transformers, reverse-complement equivariance and transfer learning from a pretrained backbone together make it practical to fine-tune on a modest GPU. Combining GAN augmentation with class weighting and threshold tuning addresses imbalance at both the data and the loss level.

The platform could support several uses: clinical decision support for preliminary screening of inherited or polygenic conditions; genomic research on under-represented rare diseases using augmented data; personalized medicine, where individual genetic profiles inform risk assessment; genetic counselling; and teaching, where SHAP views show students how a model reads a sequence.

The work has limitations. Sequences are limited to 98–104 bp rather than whole genomes. Multi-label scores are moderate (F1-Micro of 0.59–0.63), and the results come from one dataset, so they should be read as the performance of a research prototype and not as clinical validation. SHAP is costly on long sequences, and the GAN can be unstable. Synthetic sequences also do not replace real rare-disease data. Any clinical use would require validation on larger, independent datasets and review by domain experts.

VIII. Conclusion

This paper presents a DNA sequence-based disease classification system that combines a bi-directional equivariant Caduceus backbone, GAN-based augmentation and SHAP explainability behind a web interface. The system supports binary and multi-label classification, and the Caduceus classifier outperformed the DNABERT and CNN–BiLSTM baselines in all reported multi-label settings. SHAP

 ©IJSRED: All Rights are Reserved

visualizations make predictions transparent to clinicians and researchers, and the generated report supports clinical and research use.

Future work includes larger genomic databases, whole-genome and whole-exome inputs, protein-level and epigenetic features, more advanced architectures, cloud deployment with GPU-accelerated inference for hospitals and diagnostic centres, and a continual-learning pipeline that retrains the model on newly published genomic data.

References

[1] S. Sai, A. Gaur, R. Sai, V. Chamola, M. Guizani, and J. J. P. C. Rodrigues, "Generative AI for Transformative Healthcare: A Comprehensive Study of Emerging Models, Applications, Case Studies, and Limitations," IEEE Access, vol. 12, pp. 31078–31092, 2024, doi: 10.1109/ACCESS.2024.3367715.

[2] Y. Ji, Z. Zhou, H. Liu, and R. V. Davuluri, "DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome," Bioinformatics, vol. 37, no. 15, pp. 2112–2120, 2021.

[3] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Synthetic Data Augmentation using GAN for Improved Liver Lesion Classification," arXiv preprint arXiv:1801.02385, 2022.

[4] O. Düzyel and M. Kuntalp, "Data Augmentation with GAN increases the Performance of Arrhythmia Classification for an Unbalanced Dataset," arXiv preprint arXiv:2302.13855, 2023.

[5] J. M. Rožanec, P. Zajec, S. Theodoropoulos, E. Koehorst, B. Fortuna, and D. Mladeni■, "Synthetic Data Augmentation Using GAN For Improved Automated Visual Inspection," arXiv preprint arXiv:2212.09317, 2022.

[6] Y. Schiff, C.-H. Kao, A. Gokaslan, T. Dao, A. Gu, and V. Kuleshov, "Caduceus: Bi-Directional Equivariant Long-Range DNA Sequence Modeling," in Proc. 41st Int. Conf. Mach. Learn. (ICML), Vienna, Austria, vol. 235, 2024.

[7] G. Mathur, "Disease prediction model: An efficient machine learning-based DNA classifier," Rajiv Gandhi Technical University, Aug. 9, 2024. [Online]. Available: https://doi.org/10.21203/rs.3.rs-4248864/v1.

[8] X. Jia, W. Luo, J. Li, J. Xing, H. Sun, S. Wu, and X. Su, "A Deep Learning Framework for Predicting Disease-Gene Associations with Functional Modules and Graph Augmentation," BMC Bioinformatics, vol. 25, no. 214, pp. 1–15, Jun. 2024.

[9] A. Saadat and J. Fellay, "DNA Language Model and Interpretable Graph Neural Network Identify Genes and Pathways Involved in Rare Diseases," arXiv preprint arXiv:2410.15367, Oct. 2024.

[10] A. Sharma, P. Gupta, and R. Kumar, "DNA Sequence Classification and Analysis Using Machine Learning," Int. Res. J. Eng. Technol. (IRJET), vol. 11, pp. 102–110, Nov. 2024.

[11] S. M. Lundberg and S. Lee, "A Unified Approach to Interpreting Model Predictions," in Proc. 31st Conf. Neural Inf. Process. Syst. (NeurIPS), 2017.

[12] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning Important Features Through Propagating Activation Differences," in Proc. 34th Int. Conf. Mach. Learn. (ICML), 2017.

[13] R. Poplin et al., "A Universal SNP and Small-Indel Variant Caller Using Deep Neural Networks," Nature Biotechnology, vol. 36, pp. 983–987, 2018.

[14] W. Zhang et al., "Graph Neural Networks for Disease–Gene Prediction," IEEE J. Biomed. Health Inform., vol. 27, no. 4, pp. 1649–1659, 2023.

 ©IJSRED: All Rights are Reserved