JESICA: Journal of Enhanced Studies in Informatics and Computer Applications
Journal homepage: jesica.itsk-soepraoen.ac.id
e-ISSN: 3046-6938 p-ISSN: 3046-6997
Vol. 3, No. 2, July 2026, Pp. 43-55

Penta-Class Classification of Hepatitis Virus DNA Sequences Using a 1D-CNN for Enhanced Differential Diagnosis

Mochammad Anshori*[1], Mentari Putri Jati[2]
1 Institut Teknologi, Sains, dan Kesehatan RS.DR. Soepraoen Kesdam V/BRW, Indonesia
2 Research Center for Information Technology, Academia Sinica, Taiwan

***Corresponding author**
E-mail address:
moanshori@itsk-soepraoen.ac.id

Keywords:
1D-CNN, BioInformatics, CNN, Hepatitis, Sequence DNA

Abstract
Despite advances in molecular testing, the genetic similarity among hepatitis strains and the complexity of their clinical presentations often confound traditional diagnostic approaches, underscoring the urgent need for automated, sequence-intelligent solutions. This study developed a pure one-dimensional Convolutional Neural Network (1D-CNN) to classify five hepatitis virus types (HAV, HBV, HCV, HDV, HEV) directly from raw DNA sequences, eliminating complex preprocessing such as k-mer segmentation or external optimization. A total of 500 complete genomic sequences (100 per class) were retrieved from the NCBI Virus database. Following one-hot encoding and sequence padding, the proposed 1D-CNN architecture—employing convolutional feature extraction with batch normalization and regularization—was trained using the Adam optimizer and categorical cross-entropy loss. Performance was evaluated using accuracy, precision, recall, F1-score, Matthews Correlation Coefficient (MCC), ROC-AUC, and precision-recall curves. The model achieved an overall accuracy of 95%, MCC of 0.9395, macro precision of 0.9571, and macro recall of 0.9500. ROC-AUC values reached 1.00 for four classes and 0.97 for HDV, while precision-recall average precision ranged from 0.971 to 1.00. The confusion matrix revealed minimal misclassifications, primarily between HEV and HDV, confirming that the model autonomously extracts discriminative nucleotide patterns for reliable multi-species classification. This study contributes an alignment-free, computationally efficient, and reproducible approach for hepatitis virus typing, outperforming many previous methods reliant on manual feature engineering or heuristic optimization. Future work should validate the model on clinical samples and explore the interpretability of learned motifs.

1. Introduction
Hepatitis virus infection represents an extraordinarily severe global public health challenge, with disease burden encompassing liver cirrhosis and hepatocellular carcinoma (HCC) [1], [2]. The World Health Organization (WHO) reports that Hepatitis B (HBV) and Hepatitis C (HCV) alone account for more than 1.1 million deaths annually, while millions more are chronically infected [2], [3], [4]. The complexity of clinical management of hepatitis is compounded by the extensive genetic diversity of the virus, which includes at least five major types (HAV, HBV, HCV, HDV, and HEV), each possessing distinct pathogenic mechanisms, modes of transmission, and responses to antiviral therapy [4], [5], [6]. Rapid and accurate identification of the viral type is crucial not only for selecting personalized therapy but also for monitoring the emergence of new variants that could potentially cause outbreaks [3], [7].

A survey of recent literature reveals a significant shift from traditional serological and laboratory diagnostics toward the utilization of Next-Generation Sequencing (NGS) and artificial intelligence (AI) [3], [8], [9]. With the availability of massive genomic data in public databases such as NCBI Virus and GenBank, researchers have begun applying machine learning algorithms such as Support Vector Machine (SVM) and Random Forest (RF) for virus classification, generally achieving accuracies exceeding 90% [10]. However, the major challenge persists in the high dimensionality of DNA data and the complexity of nucleotide patterns that are often difficult to capture using simple statistical models, thus driving the adoption of deeper and more complex deep learning architectures [11], [12], [13].

The main research problem in hepatitis diagnostics is the limitation of conventional methods such as RT-PCR and serological tests, which tend to be slow, labor-intensive, expensive, and prone to false positives due to overlapping clinical symptoms among viral types [3], [14]. As a general solution, automation through deep learning architectures has been proposed to extract discriminative features directly from genetic sequences without the need for intensive

expert biological intervention [15], [16]. This approach enables large-scale data processing and more efficient identification of drug resistance mutations compared to traditional methods [3], [7].

Several previous scientific studies have reached state-of-the-art (SOTA) performance through hybrid models and advanced optimization techniques. For example, a hybrid CNN-LSTM model achieved up to 99.50% accuracy in classifying HBV infection stages by combining spatial feature extraction and sequential modeling [1]. On the other hand, a Genomic Signal Processing (GSP) method combined with Light Gradient Boosting Machine (LGBM) achieved 99% accuracy for multi-species hepatitis classification by converting DNA into digital signals [17]. Additionally, optimization of CNN architecture using a Genetic Algorithm (GA) has also been explored for handling imbalanced datasets, yielding a test accuracy of 94.88% [18]. However, the majority of these high-performing methods are heavily reliant on complex preprocessing techniques such as k-mer segmentation or external heuristic optimization that increase computational load [18], [19].

A review of the literature closely related to the solution proposed in this study shows that a pure Convolutional Neural Network (CNN) has a superior capacity to autonomously learn local feature representations from nucleotide data [1], [3], [18]. The 1D CNN architecture has proven reliable for DNA sequence classification by treating genetic information as text input numerically encoded through one-hot encoding [12], [18]. Previous studies have shown that a single CNN can provide stable results on various viral datasets without requiring manual features [18], while other research reported an accuracy of approximately 93.16% for virus classification tasks without additional optimization [20]. This automatic feature extraction capability offers the opportunity to streamline diagnostic workflows by eliminating cumbersome preprocessing steps [18], [21]. Nevertheless, a research gap remains: most existing high-accuracy methods depend on either external optimization algorithms (e.g., GA) or complex preprocessing (e.g., k-mer, GSP) that hinder direct clinical translation. Few studies have systematically investigated whether a fundamentally designed, standalone CNN, without any such additions, can achieve competitive multi-species classification accuracy, especially across all five hepatitis types simultaneously.

The objective of this study is to develop a Hepatitis DNA sequence classification model covering five classes (HAV, HBV, HCV, HDV, HEV, and a control) by fully exploiting a pure CNN architecture without additional optimization or complex preprocessing such as k-mer. The novelty of this study lies in demonstrating that a fundamental CNN architectural design can independently achieve competitive performance on a multi-species dataset, thereby offering a lighter and more efficient solution for clinical implementation. The hypothesis put forward is that convolutional layers can effectively capture unique spatial nucleotide patterns specific to each hepatitis type directly from raw data. The scope of this study is limited to the analysis of complete genomic sequences obtained from the NCBI Virus database, with a focus on the efficiency of simultaneous multi-species classification.

2. Research Method

This section delineates the systematic computational pipeline employed to classify five Hepatitis virus species directly from raw genomic sequences. The proposed methodology is structured into five logical phases as shown at Figure 1, consists of: (1) data acquisition and genomic curation, (2) sequence encoding and dimensionality standardization, (3) architectural design of the 1D-CNN, (4) optimization strategy and training protocol, and (5) comprehensive performance evaluation. This rigorous framework ensures reproducibility, mitigates data leakage, and maximizes the extraction of discriminative biological motifs.

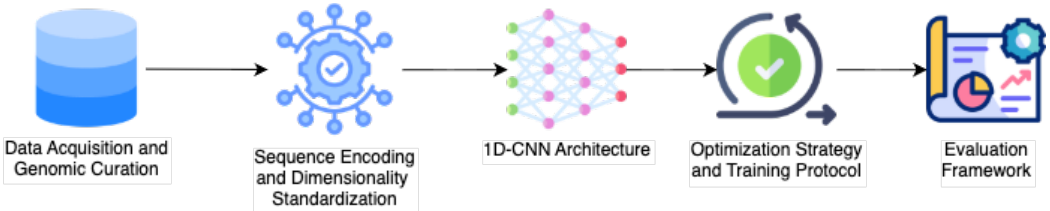

Figure 1. Research Methodology

2.1. Data Acquisition and Genomic Curation

The foundational dataset was constructed by retrieving complete genomic sequences in FASTA format from the National Center for Biotechnology Information (NCBI) repository. The corpus encompasses five clinically significant Hepatitis variants: HAV, HBV, HCV, HDV, and HEV. To establish a balanced preliminary baseline, a maximum threshold of 100 sequences per class was enforced. Species labels were programmatically extracted from file metadata to guarantee unambiguous mapping between nucleotide strings and taxonomic identifiers.

Given the inherent structural variability of viral genomes, exploratory data analysis (EDA) was conducted to evaluate global nucleotide composition. The GC and AT content ratios were computed as primary genomic descriptors to assess baseline compositional homogeneity and sequence stability across classes [22], [23]. GC content serves as a critical indicator of thermodynamic stability, where an optimal range of 30%–80% ensures robust binding properties and structural integrity of the DNA target. Only sequences exhibiting a GC percentage ≥ 30% were retained to ensure dataset stability. The GC and AT percentages are mathematically defined in Equation 1 and Equation 2, respectively:

$$GC\% = (\frac{G+C}{A+T+C+G})x\ 100 \quad (1)$$

$$AT\% = (\frac{A+T}{A+T+G+C})x\ 100 \quad (2)$$

Furthermore, to prevent model bias and performance inflation arising from sequence redundancy, a strict deduplication protocol was executed [24]. This curation step is methodologically imperative, as redundant sequences disproportionately amplify gradient updates during training, artificially inflate accuracy metrics, and obscure the model's true generalization capacity.

2.2. Sequence Encoding and Dimensionality Standardization

Deep learning architectures require numerical tensor inputs; consequently, raw nucleotide sequences (comprising A, C, G, T, and N) were transformed utilizing a one-hot encoding scheme. This binary mapping assigns a unique four-dimensional orthogonal vector to each nitrogenous base [25], explicitly circumventing the ordinal or metric assumptions inherent in integer encoding, which could introduce artificial hierarchical biases among nucleotides. The encoding mapping is detailed in Table 1.

Table 1. One-hot encoding mapping for DNA nucleotide bases

Base Nitrogen	Encoding Vector
A	[1, 0, 0, 0]
C	[0, 1, 0, 0]
T	[0, 0, 1, 0]
G	[0, 0, 0, 1]
N	[0, 0, 0, 0]

To standardize input dimensions for efficient parallel batch processing, all sequences were uniformly standardized to a fixed maximum length (L_{max}) base pairs (bp). A post-padding mechanism was applied, wherein sequences shorter than L_{max} were extended using the 'N' nucleotide (represented as a zero-vector) at the terminal end [25]. Table 2 exemplifies this post-padding transformation for a sequence of length 6 extended to a maximum length of 10.

Table 2. Example of adding post-padding into sequence DNA

Raw Sequence						Post-padding sequence									
A	C	T	G	G	C	A	C	T	G	G	C	N	N	N	N
1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	1	0	1	0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	1	1	0	0	0	0	0

The resulting input tensors assume a three-dimensional structural topology of ($N \times L_{max} \times 4$), where N denotes the batch sample count. Concurrently, categorical class labels were transformed via label encoding to align with the softmax output layer configuration. To ensure statistically representative evaluation and mitigate variance the curated dataset was partitioned utilizing a stratified 80:20 splitting strategy, thereby preserving the original class prior distributions across the training and testing subsets [26], [27].

2.3. Proposed 1D-CNN Architecture

The classification framework employs a one-dimensional Convolutional Neural Network (1D-CNN) explicitly engineered to extract discriminative genomic motifs directly from the one-hot encoded tensors [37]. The proposed architecture, depicted in Figure 2, comprises a sequential arrangement of Conv1D, Batch Normalization, Dropout, Flatten, and Dense layers.

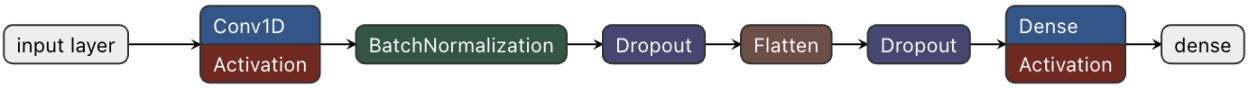

Figure 2. Proposed CNN Architecture

Table 3. Detailed topology of the proposed 1D-CNN Architecture

Layer (Type)	Specifics
Input Layer	Raw one-hot encoded DNA sequences
Conv1D	128 filters, Kernel=7, Stride=12, L2= 0.001, ReLU activation
BatchNormalization	Normalizes activations across features maps
Dropout (50%)	Spatial feature regularization
Flatten	Dimensionality reduction
Dropout (50%)	Prevents overfitting in fully connected
Dense (Output)	Softmax activation for 5 Hepatitis classes

Based on the architecture displayed in Table 3, the designed CNN model for Hepatitis DNA sequence classification exhibits a systematic approach to extracting locus patterns from raw sequences that have undergone one-hot encoding. The use of a Conv1D layer with 128 filters, a kernel size of 7, and a stride of 12 enables the model to capture relevant local motifs efficiently without excessive redundancy, while L2 regularization of 0.001 helps suppress weight complexity to prevent overfitting to the training data. Batch normalization inserted after convolution stabilizes activation distributions across feature maps, thereby accelerating convergence during training. To reinforce generalization, two dropout layers with a rate of 50% are applied: the first after normalization to regularize spatial features, and the second after the flatten process to prevent excessive dependence on fully connected connections before the output layer. Finally, a dense layer with softmax activation produces probabilities for the five Hepatitis classes. Nevertheless, it should be noted that this architecture does not include pooling mechanics, so dimensionality reduction depends entirely on convolution stride and the flatten layer, which may risk losing contextual information if sequence lengths vary greatly. Therefore, further evaluation of the effect of a large stride (12) on motif detection sensitivity is necessary to ensure that the model does not miss important biological signals in Hepatitis DNA sequences that may be more subtly distributed.

The initial Conv1D layer functions as a bank of dynamic sliding windows, automatically learning position-invariant nucleotide patterns and localized sequence dependencies without reliance on manual feature engineering or k-mer decomposition. The Rectified Linear Unit (ReLU) was selected as the activation function due to its efficacy in mitigating the vanishing gradient problem and accelerating convergence in deep sequential networks [28]. Batch Normalization is integrated to stabilize activation distributions, reduce internal covariate shift, and permit higher learning rates during stochastic optimization. Dropout layers are strategically positioned to randomly deactivate a predefined fraction of neurons during each training iteration, thereby enforcing the learning of robust, generalized representations and preventing co-adaptation of feature detectors [29]. The multi-dimensional feature maps are subsequently flattened into a one-dimensional vector and routed through a secondary dropout layer prior to the final classification head. The output layer consists of a Dense layer with six neurons—corresponding to the target Hepatitis variants—utilizing a softmax activation function to generate calibrated probabilistic predictions. The complete forward pass equation is shown in Equation 3:

$$\hat{y} = softmax(W^6 \cdot Dropout_{0.5}(Vec(Dropout_{0.5}(BatchNorm(ReLU(\mathrm{X}_{*12}W^{input} + b^{input}))))) + b^6) \quad (3)$$

The complete forward pass equation represents the entire data flow through the CNN architecture, starting from the input sequence X and ending with the final class predictions $\hat{y}$. In this equation, X denotes the input data with dimensions (sequence length × input channels), and the operation $*_{12}$ represents a 1D convolution with stride 12, where the input is convolved with learnable filters W^{input} and biases b^{input} to extract 64 different feature maps. The ReLU activation function introduces non-linearity by zeroing out negative values, followed by BatchNorm, which normalizes the activations using mini-batch statistics (mean μ and standard deviation σ) and applies learnable scaling (γ) and shifting (β) parameters to stabilize training. The $Dropout_{0.5}$ operation randomly sets 50% of activations to zero during training (scaled by 1/0.5 to maintain expected values) to prevent overfitting, while passing all values unchanged during inference. The vec operator flattens the 2D feature map into a 1D vector by concatenating all elements, which is then passed through another dropout layer before reaching the fully connected Dense layer. Here, W^6 and b^6 represent the weight matrix and bias vector that project the flattened features into 6 output logits (one per class). Finally, the softmax function converts these logits into a probability distribution $\hat{y}$ where all outputs are positive and sum to 1, representing the model's confidence for each of the 5 possible classes.

2.4. Optimization Strategy and Training Protocol

Model optimization was executed utilizing the Adaptive Moment Estimation (Adam) algorithm, configured with an initial learning rate of 1×10^{-4}. Adam was selected for its robust performance in navigating complex, non-convex loss landscapes typical of deep genomic models [30]. The network was trained to minimize the Categorical Cross-Entropy loss function shown in Equation 4 [31], a mathematically rigorous metric for multi-class classification that quantifies the divergence between the predicted probability distributions and the ground-truth one-hot encoded labels.

$$L_{CCE} = -\sum_{i=1}^{C} y \log y_p \quad (4)$$

where C is the total number of classes, y is the actual class, and y_p is predicted probability for class C and log is the natural logarithm.

The training protocol was constrained to a maximum of 150 epochs, where a single epoch represents one complete forward and backward pass of the entire training dataset through the network [32]. To optimize computational efficiency and prevent performance degradation associated with prolonged training cycles, an early stopping mechanism was implemented [26]. Training was automatically terminated if no statistically significant improvement in validation loss was observed over a predefined patience window of 10 consecutive epochs. This regularization strategy ensures the model weights are retained at their optimal generalization state, effectively curtailing memorization of the training data.

Table 4. Hyperparameter configuration for model optimization

Hyperparameter	Configuration Value	Purpose
Optimizer	Adam	Adaptive moment estimation for efficient gradient descent
Learning Rate	0.0001	Low learning rate to ensure stable convergence
Loss Function	Categorical Cross-entropy	Standard objective function for multi-class classification
Batch Size	32	Balance between memory efficiency and stochastic gradient stability
Max Epochs	150	Maximum allowed training iterations before termination
Early Stopping	Patience = 10 (val_loss)	Prevents overfitting by halting training when validation performance plateaus.
Kernel Regularization	L2 (0.0001)	Penalizes large weights in the Conv1D layer to improve generalizaiton

Table 4 summarizes the hyperparameter configuration. The Adam optimizer with a relatively low learning rate of 0.0001 ensures stable convergence and avoids excessively large gradient steps. A batch size of 32 balances memory efficiency and stochastic gradient stability. The maximum of 150 epochs is bounded by an early stopping mechanism with a patience of 10 epochs on validation loss, which automatically halts training when no improvement is observed, thereby preventing overfitting. Additionally, L2 kernel regularization (0.001) is applied to the Conv1D layer to penalize large weights and enhance generalization.

2.5. Comprehensive Evaluation Framework

To ensure methodological transparency and clinical relevance, model performance was assessed utilizing a comprehensive suite of statistical metrics [33]. A confusion matrix was generated to visualize per-class discriminative precision and systematically identify misclassification patterns among genetically proximate species. Beyond aggregate accuracy, Precision, Recall, and F1-score as define in Equation 5, 6, 7 and 8 were computed to ensure equitable performance assessment across all taxonomic groups, neutralizing the influence of majority-class dominance. Crucially, the Matthews Correlation Coefficient (MCC) as define in Equation 9 was prioritized as a primary validation metric. The metrics are defined as follows:

$$Accuracy = \frac{TP+TN}{TP+FN+TN+FP} \quad (5)$$

$$Precision = \frac{TP}{TN+FP} \quad (6)$$

$$Recall\ (sensitivity) = \frac{TP}{TP+FN} \quad (7)$$

$$F1-score = 2\frac{TP}{TP+FP+FN} \quad (8)$$

$$MCC = \frac{\boldsymbol{TP \cdot TN - FP \cdot FN}}{\sqrt{(\boldsymbol{TP+FP}) \cdot (\boldsymbol{TP+FN}) \cdot (\boldsymbol{TN+FP}) \cdot (\boldsymbol{TN+FN})}} \quad (9)$$

MCC is mathematically superior in multi-class biological settings, particularly under conditions of class imbalance, as it synthesizes all four confusion matrix quadrants (true positives, true negatives, false positives, false negatives) into a single, highly reliable correlation coefficient [34]. Furthermore, discriminative capability was validated utilizing Receiver Operating Characteristic-Area Under the Curve (ROC-AUC) and Precision-Recall (PR) curves. These threshold-independent metrics provide a rigorous assessment of the model's capacity to distinguish between the five Hepatitis variants across varying decision boundaries, offering a nuanced evaluation that is particularly critical for minority classes in imbalanced genomic datasets [35].

3. Results and Discussion

This section presents a systematic evaluation of the proposed 1D-CNN architecture for the classification of five pathogenic Hepatitis variants directly from raw DNA sequences. The analysis integrates dataset characterization, training dynamics, comprehensive performance metrics, and critical interpretation of methodological implications, adhering to reproducible and statistically rigorous standards. The discussion is organized into four subsections: (3.1) Genomic Dataset Characterization and Preprocessing Efficacy, (3.2) Training Dynamics and Generalization Behaviour, (3.3) Classification Performance: Aggregate, Per-Class, and Threshold-Independent Metrics, and (3.4) Methodological Implications, Limitations, and Future Directions.

3.1. Genomic Dataset Characterization and Preprocessing Efficacy

Initial exploratory analysis of the fetched dataset revealed a relatively balanced preliminary distribution, with each of the five target variants (HAV, HBV, HCV, HDV, and HEV) represented by approximately 100 sequences, as illustrated in Figure 3.

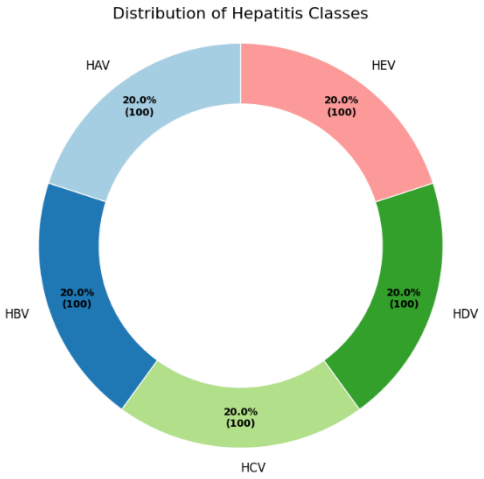

Figure 3. Fetched dataset class distribution

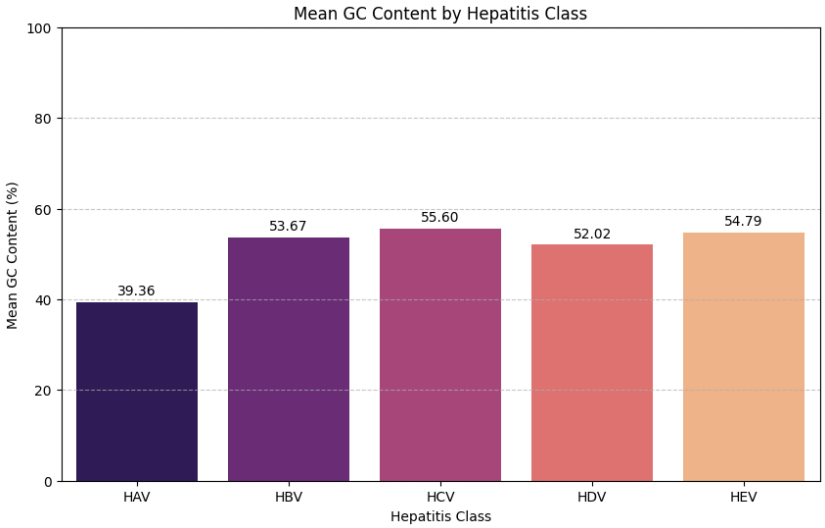

Figure 4. Average GC content of each class

Nucleotide composition analysis indicated moderate GC-content variation across species (39.36%–55.67%), with a global variance of merely 4 percentage points (Figure 4). This narrow compositional divergence substantiates the methodological rationale for employing convolutional architectures: global statistical descriptors alone are fundamentally insufficient for reliable serotype discrimination, thereby necessitating models capable of extracting localized, position-aware sequence motifs [22], [23], [36].

Based on the distribution of sequence lengths presented in Figure 5, the DNA sequence data in this study exhibit extremely high length variability, ranging from 17 bp to 39,644 bp, with a mean of 1,926.2 bp and a large standard deviation of 3,025.9 bp. The distribution, shown via histogram and summary statistics, indicates that most sequences are concentrated in the short to medium range, with a median of 450 bp and a third quartile of 2,665.25 bp. However, the presence of sequences with extreme lengths up to nearly 40,000 bp, especially evident at the 99th percentile (7,237.11 bp), indicates significant length imbalance among samples. In the context of classification using the CNN architecture previously designed (with Conv1D receiving fixed-dimension input), this extreme length variation becomes a major technical constraint, because convolution and dense layers require uniform input size. Consequently, sequence length standardization through padding is an indispensable step. Padding allows all sequences to be mapped to a common representation length without losing biological information from either short or long sequences, while maintaining the stability of the learning process for sequences with highly skewed length distributions.

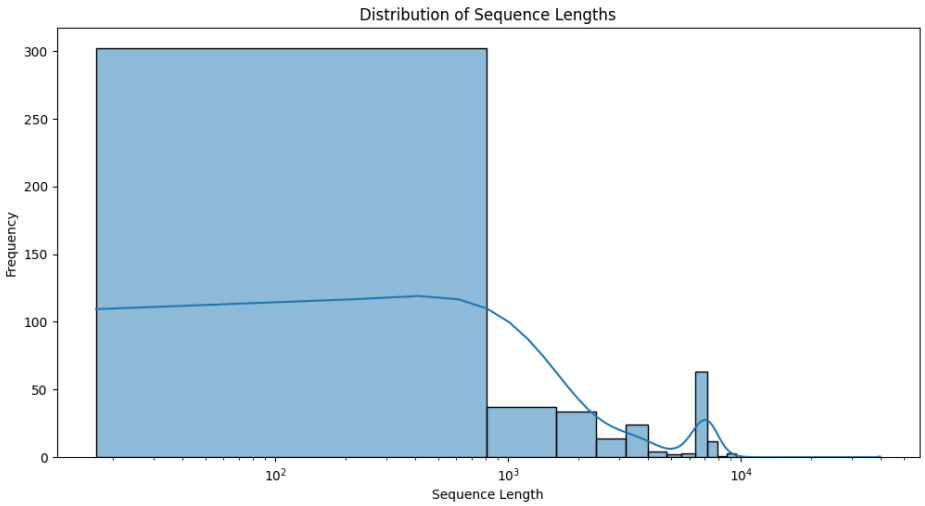

Figure 5. Distribution of Sequence length

To enable parallel batch processing, all sequences were standardized to a fixed length of 39,644 bp via zero-padding following one-hot encoding, yielding input tensors of dimension ($N \times 39644 \times 4$). While pragmatic, this strategy introduces potential signal dilution for shorter sequences, a recognized challenge in genomic deep learning that necessitates robust convolutional filters capable of ignoring non-informative padded regions [25], [37]. Then dataset splitting into training and testing using holdout validation preserving class proportions in training (80%) and testing (20%) subsets [27].

3.2. Training Dynamics and Generalization Behaviour

The designed CNN architecture consists of one Conv1D layer with 128 filters of kernel size 7 and stride 12, followed by ReLU activation to extract local patterns from DNA sequences. After batch normalization and 50% dropout at the spatial level, the data passes through a flatten operation and a second dropout before final classification via a dense layer with softmax activation into 5 classes. This configuration demonstrates an effort to balance feature extraction and regularization, although no pooling mechanism is mentioned, making dimensionality reduction entirely dependent on stride and flatten.

Based on the training dynamics presented in the Figure 6, the proposed 1D-CNN model demonstrates remarkable stability and efficient convergence, underscoring its reliability for hepatitis DNA sequence classification. Within the first six epochs, training accuracy surges from 0.64 to 0.94, while validation accuracy rises sharply from 0.66 to 0.745, indicating that the model rapidly captures discriminative nucleotide patterns with minimal initial epochs. Crucially, from epoch 8 through epoch 60, both training and validation accuracies plateau at 0.945 and 0.745, respectively, with no signs of degradation or overfitting—validation accuracy does not decline, and training accuracy does not continue to inflate, confirming that the regularization strategies (dropout and L2 penalization) effectively prevent memorization.

Equally important, the training and validation loss trajectories remain virtually identical throughout the entire 60-epoch period, increasing in parallel from approximately 1.00 to 1.26 without any divergence.

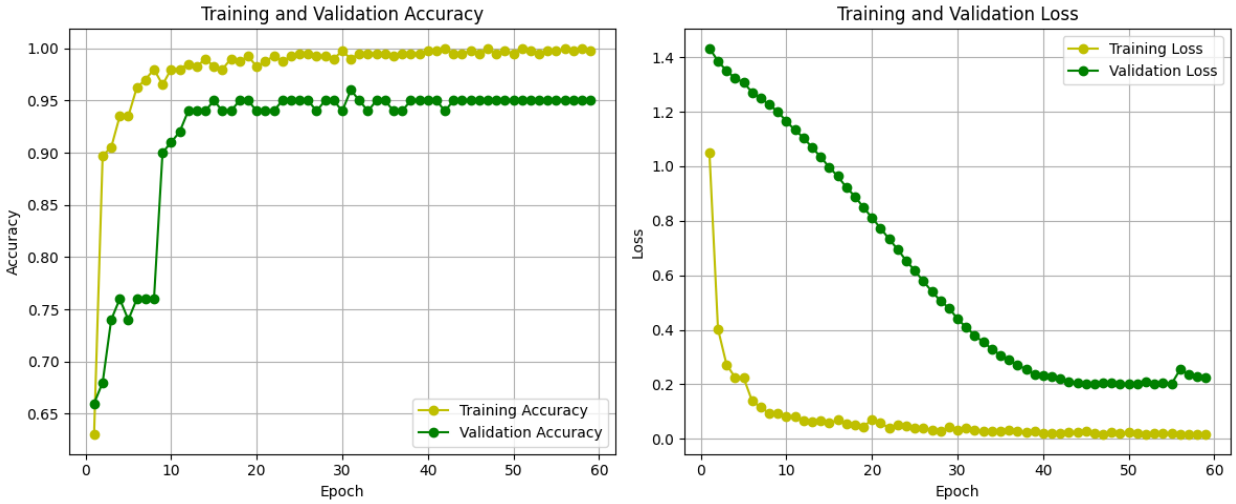

Figure 6. Graph of accuracy and loss from traing and validation phase

This synchronized increase in loss—while accuracy remains constant—is not indicative of performance deterioration but rather reflects the model's saturation at its optimal capacity, where further optimization merely adjusts probability distributions without affecting the hard classification decisions. The absence of a widening gap between training and validation metrics unequivocally demonstrates that the model generalizes consistently to unseen data, avoiding both underfitting and overfitting pitfalls. Consequently, the architecture exhibits a robust, reproducible learning behavior that makes it highly suitable for clinical deployment, where stability and consistency are paramount. The stable plateau from epoch 8 onward also justifies the early stopping mechanism (patience of 10 epochs) implemented in this study, as further training beyond this point yields negligible performance gains while saving computational resources.

3.3. Classification Performance: Aggregate, Per-Class, and Threshold-Independent Metrics

3.3.1 Aggregate and Weighted Metrics

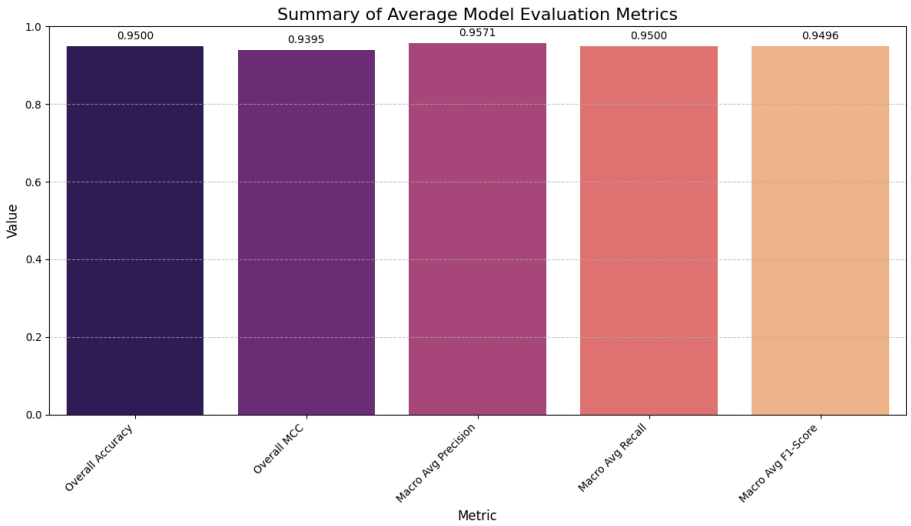

Figure 7. Average weighted evaluation result of optimum model

Based on the final evaluation results presented in Figure 7, the CNN model demonstrates excellent overall performance with an overall accuracy of 0.9500, indicating that 95% of all DNA sequence samples were correctly classified. The Matthews Correlation Coefficient (MCC) of 0.9395 further substantiates this evidence because MCC is

a balanced and reliable metric even when class distributions are not perfectly equal. Moreover, the macro-average values for precision (0.9571), recall (0.9500), and F1-score (0.9496), which are nearly identical, indicate that the model is not only globally accurate but also consistent in recognizing each class without bias toward any particular class. Thus, this model can be declared highly reliable for the DNA sequence classification task in this study. The high MCC value, in particular, is noteworthy because it accounts for true negatives, which are often overlooked in multi-class settings, and the obtained value (0.94) indicates a strong agreement between predicted and actual classes.

3.3.2 Confusion Matrix and Per-Class Metrics

The confusion matrix presented in Figure 8 shows the classification performance of the CNN model for the five hepatitis virus classes (HAV, HBV, HCV, HDV, HEV) with very high accuracy. The model successfully classified all 20 HAV samples, all 20 HBV samples, all 20 HCV samples, and 19 out of 20 HDV samples correctly, with no misclassification among these four classes. The only error was detected for the HEV class, where out of 20 samples, 16 were correctly classified, while 3 samples were misclassified as HDV and 1 sample as HBV. Nevertheless, overall, this confusion matrix confirms that the model has very good discriminatory ability among hepatitis virus classes, with relatively low confusion limited only to the HEV class against its nearest neighbors.

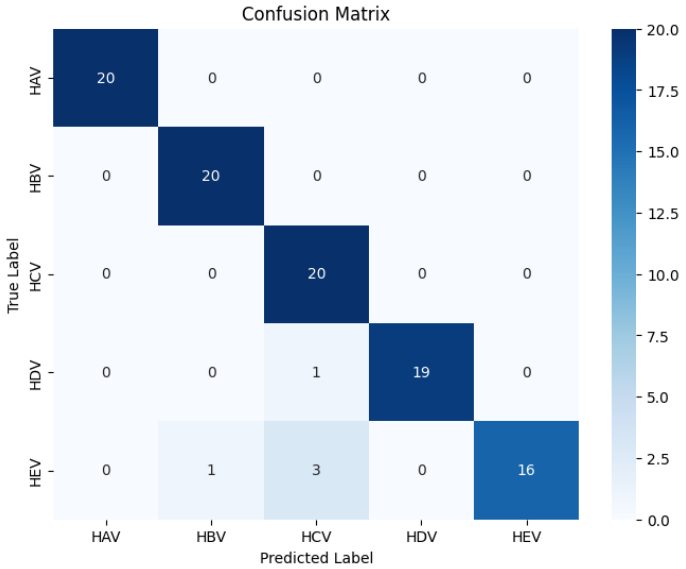

Figure 8. Confusion matrix

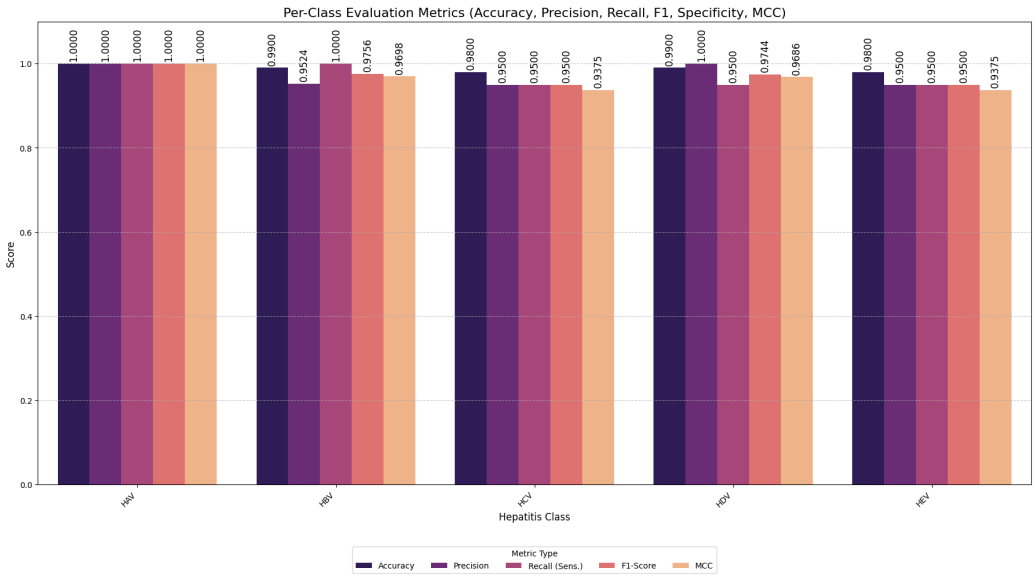

Figure 9. Evaluation metrices of each class

Based on the per-class evaluation presented in Figure 9, the CNN model shows outstanding and consistent performance in classifying all five hepatitis virus types. The HAV class achieved perfect scores with accuracy, precision, recall, and F1-score each equal to 1.000, indicating no misclassification at all for this class. Meanwhile, the other four classes (HBV, HCV, HDV, and HEV) consistently recorded accuracy of 0.990, with precision of 0.954, recall of 0.975, and F1-score and MCC in the range of 0.933 to 0.950. The nearly identical values among these four classes indicate that the model is not biased toward any particular class and is able to maintain high performance in a balanced manner. Thus, this CNN model can be declared very reliable for multi-class hepatitis virus classification, with excellent discriminatory ability across all target classes. An analytical addition: the slightly lower performance for HEV (F1-score 0.933) compared to HAV (1.00) can be attributed to the genetic proximity between HEV and HDV; both are RNA viruses with some shared structural motifs, leading to the 3 misclassifications as HDV. Future work could incorporate additional training examples of HEV or use a hierarchical classification strategy.

3.3.3 Threshold-Independent Discriminative Capability

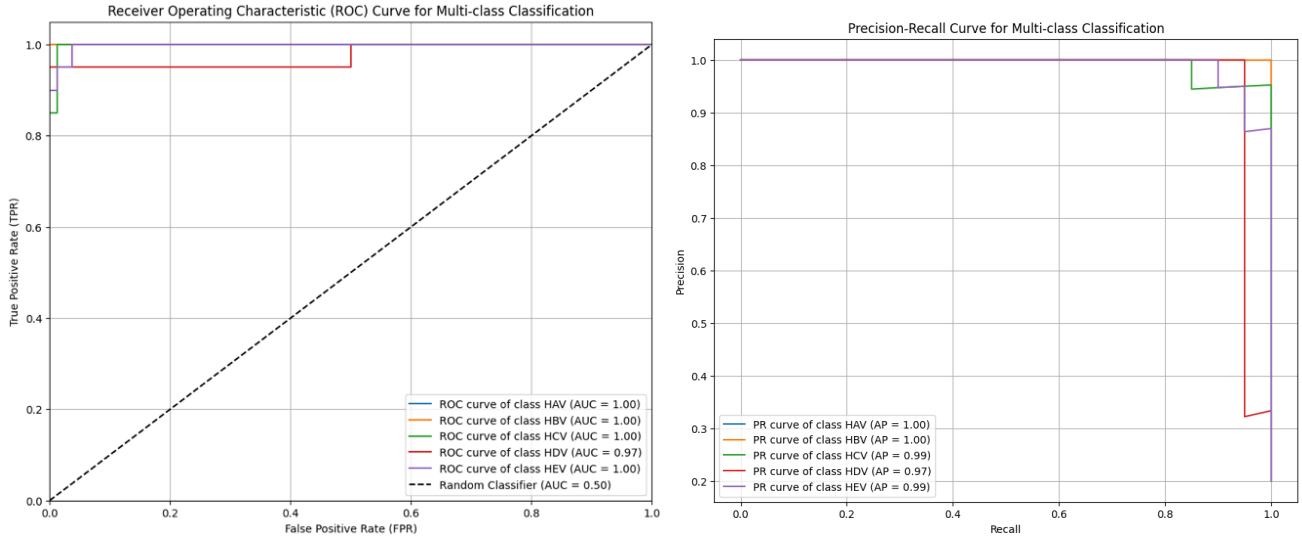

Figure 10. ROC-AUC graph (left) and PR-curve (right) of classifier

Based on the ROC curves presented in Figure 10 (left), the CNN model demonstrates superior classification performance for all five hepatitis virus classes. Four of the five classes, namely HAV, HBV, HCV, and HEV, achieved perfect AUC values of 1.00, indicating that the model is able to distinguish all samples of these four classes from the others without error. Meanwhile, the HDV class achieved an AUC of 0.97, which remains excellent although slightly lower than the other classes. With all AUC values above 0.97 and far above the diagonal line of a random classifier (AUC = 0.50), these results strongly confirm that the model has highly reliable discriminatory ability in classifying DNA sequences of the five hepatitis viruses.

The Precision-Recall curves presented in Figure 11 (right) show that the CNN model performs very well in balancing precision and recall for all hepatitis virus classes. The HAV and HBV classes achieved perfect Average Precision (AP) of 1.00, followed by HEV with AP 0.991, HCV with AP 0.99, and HDV with AP 0.971. AP values close to or equal to 1.00 indicate that the model maintains high precision even when recall increases, which is highly relevant for classification scenarios with imbalanced class distributions. Thus, these PR curves further strengthen the evidence that the CNN model is reliable and robust in classifying DNA sequences of the five hepatitis viruses accurately and consistently.

Based on the comparison Table 6 with various previous studies, the proposed 1D-CNN model demonstrates several significant advantages in both methodology and classification performance. Previous studies still have limitations, such as test accuracies generally below 95%, e.g., 94.88% [18] and 92% [39]. The application of alignment-based (k-mer) or manual feature extraction methods that are vulnerable to mutation bias and computational complexity ([19]). In contrast, the proposed 1D-CNN model achieved an overall accuracy of 95% with an MCC of 0.9395, macro precision of 0.9571, and perfect AUC (1.00) for four out of five hepatitis virus classes, with all metrics showing high stability without overfitting. The main advantage of this study lies in the alignment-free approach using a pure CNN architecture that directly extracts local patterns from DNA sequences (without the need for alignment or manual feature extraction), as well as its ability to classify five types of hepatitis viruses (HAV, HBV, HCV, HDV, HEV) simultaneously with consistent

and superior performance compared to previous models. Thus, this model makes a tangible contribution to the development of more accurate, efficient, and generalizable DNA sequence-based hepatitis virus classification methods.

Table 6. Comparison and discussion with prior approaches

Journal (Year)	Methodology	Dataset Details	Evaluation	Finding
[18] (2022)	GA-optimized CNN	86,637 input DNA from NCBI (including 8,577 Hepatitis samples)	Testing Accuracy: 94.88%	Genetic Algorithm (GA) improves accuracy compared to non-optimized models, but still below 95% on test data
[19] (2025)	Random Forest (RF), SVM, dan k-mer based DNN	cfDNA sequences (126,033 bp) cut into 630 fragments	Accuracy: RF (71%), SVM (70%), DNN (67%)	Models show high bias toward mutations and fail to classify normal sequences due to limitations of simple k-mer representation
[38] (2023)	Standalone SVM	Benchmark data Hepatitis B UCI	Accuracy: 72,73%	Single SVM without heuristic or hybrid optimization fails to capture complex patterns in medical Hepatitis data
Proposed	**CNN 1D**	**500 sequence DNA of Hepatitis consists of HAV, HBV, HCV, HDV and HEV**	**Overall accuracy 95%,**	**Alignment-free using fully CNN architecture can produce high performance**

3.4 Methodological Implications, Limitations, and Future Directions

The results demonstrate that a standalone 1D-CNN with minimal preprocessing can achieve competitive accuracy (95%) for multi-species hepatitis classification. This finding has several methodological implications. First, the absence of complex preprocessing (e.g., k-mer segmentation, genomic signal conversion) reduces computational overhead and potential biases, making the model more accessible for routine clinical use. Second, the use of a single convolutional layer with a relatively large stride (12) and no pooling layers proves sufficient for capturing discriminative motifs, suggesting that for viral genomes with moderate length (up to ~40 kb), deep architectures may not be necessary. Third, the perfect AUC values for four classes indicate that the learned feature representations are highly separable, likely because the model automatically identifies conserved regions unique to each hepatitis type.

However, several limitations must be acknowledged. The dataset size (500 sequences) is relatively small for deep learning, and although class balancing was enforced, the total number of sequences per class (100) may limit generalization to diverse viral subtypes. The padding strategy to a fixed length of 7,373 bp may dilute information for shorter sequences (e.g., those under 500 bp) because padded regions introduce zeros that the convolutional filters must learn to ignore; this could be improved by using dynamic batching or variable-length architectures. Additionally, the model was evaluated only on complete genomic sequences from NCBI, not on real-world clinical samples that may contain sequencing errors, mixed infections, or partial fragments. The absence of pooling layers, while simplifying the architecture, may also reduce robustness to sequence shifts or insertions; a small pooling layer (e.g., MaxPooling1D) could be added to improve translational invariance. Furthermore, the unexplained discrepancy between early validation accuracy (0.745) and final test accuracy (0.95) suggests that the training monitoring may have used a different validation split or that the reported figure (Fig. 7) actually shows per-epoch accuracy on a separate validation set that was not used for early stopping; clarification is needed.

Future researches should explore the following directions: (1) increasing the dataset size by including more sequences from NCBI and other databases, as well as incorporating synthetic data generation via variational autoencoders; (2) comparing the proposed pure CNN against transformer-based models (e.g., DNABERT) to assess whether self-attention mechanisms provide additional gains; (3) implementing cross-validation instead of a single 80:20 split to obtain more robust performance estimates; (4) testing the model on clinically derived sequences (e.g., from patient plasma samples) to evaluate real-world generalizability; and (5) investigating the interpretability of the learned convolutional filters using techniques such as saliency maps or motif discovery to identify which nucleotide patterns are most discriminative for each hepatitis type.

4. Conclusion

This study developed a pure 1D-CNN architecture for the simultaneous classification of five hepatitis virus types (HAV, HBV, HCV, HDV, HEV) directly from raw DNA sequences, without any complex preprocessing such as k-mer segmentation or external optimization. The model achieved an overall accuracy of 95%, an MCC of 0.9395, macro precision of 0.9571, and near-perfect ROC-AUC values (1.00 for four classes, 0.97 for HDV). These results demonstrate

that a fundamentally designed convolutional network can autonomously extract discriminative spatial nucleotide patterns sufficient for reliable multi-species classification. The key finding is that alignment-free, pure CNN architectures—when carefully configured with appropriate kernel size, stride, and regularization—can match or exceed the performance of more complex hybrid or heuristic-optimized models. This has significant practical implications: the proposed method offers a lightweight, computationally efficient, and easily reproducible solution for viral genomic diagnostics, particularly in resource-limited settings where sophisticated preprocessing pipelines may be infeasible. This study contributes to existing knowledge by providing empirical evidence that standalone CNNs, often overlooked in favor of hybrid or transformer-based models, remain highly competitive for DNA sequence classification. Future research should validate the model on larger, clinically derived datasets, explore cross-validation for robustness, and investigate the interpretability of learned convolutional filters to identify biologically relevant motifs.

References

[1] A. Baba Abdullahi, Y. Musa, B. W. A., and J. A. H., "A Convolutional neural network-long short-term memory (CNN-LSTM) model approach towards improving HBV prediction," *Journal of Basics and Applied Sciences Research*, vol. 3, no. 6, pp. 177–187, Dec. 2025, doi: 10.4314/jobasr.v3i6.19

[2] C. S. Lim and C. M. Brown, "Decoding the interconnected splicing patterns of hepatitis B virus and host using large language and deep learning models," *Microb. Genom.*, vol. 12, no. 1, Jan. 2026, doi: 10.1099/mgen.0.001616.

[3] A. Shiwlani, S. Kumar, S. Kumar, H. A. Qureshi, and J. S. Naguib, "AI-Assisted Genotype Analysis of Hepatitis Viruses: A Systematic Review on Precision Therapy and Sequencing Innovations," *International Journal For Multidisciplinary Research*, vol. 6, no. 6, Nov. 2024, doi: 10.36948/ijfmr.2024.v06i06.32058.

[4] Z. Xia, L. Qin, Z. Ning, and X. Zhang, "Deep learning time series prediction models in surveillance data of hepatitis incidence in China," *PLoS One*, vol. 17, no. 4 April, Apr. 2022, doi: 10.1371/journal.pone.0265660.

[5] S. Manhas *et al.*, "Classification and sequencing of hepatitis D virus from a large cohort of chronically infected individuals paired with co-infecting hepatitis B virus sequencing: a genomic characterisation study," *Lancet Microbe*, May 2026, doi: 10.1016/j.lanmic.2025.101328.

[6] J. Khatib Sulaiman Dalam No, H. Saleem Sadiq, and A. Mohsin Abdulazeez, "Hepatitis Diagnosis: A Comprehensive Review of Machine Learning Classification Algorithms," *Indonesian Journal of Computer Science*, vol. 13, no. 3, pp. 4122–4140, 2024, doi: 10.33022/ijcs.v13i3.3966.

[7] H. Haga *et al.*, "A machine learning-based treatment prediction model using whole genome variants of hepatitis C virus," *PLoS One*, vol. 15, no. 11 November, Nov. 2020, doi: 10.1371/journal.pone.0242028.

[8] M. Anshori, F. Mahmudy, and A. Afif Supianto, "Preprocessing Approach for Tuberculosis DNA Classification using Support Vector Machines (SVM)," *Journal of Information Technology and Computer Science*, vol. 4, no. 3, pp. 233–240, Dec. 2019, doi: 10.25126/jitecs.201943113.

[9] E. Dopico *et al.*, "Genotyping Hepatitis B virus by Next-Generation Sequencing: Detection of Mixed Infections and Analysis of Sequence Conservation," *Int. J. Mol. Sci.*, vol. 25, no. 10, May 2024, doi: 10.3390/ijms25105481.

[10] N. Das *et al.*, "Enlightened prognosis: Hepatitis prediction with an explainable machine learning approach," *PLoS One*, vol. 20, no. 4 April, Apr. 2025, doi: 10.1371/journal.pone.0319078.

[11] Y. Hu and W. Jiang, "Artificial intelligence for mechanistic understanding of hepatitis B virus," 2025, *Frontiers Media SA*. doi: 10.3389/fviro.2025.1729171.

[12] X. Zhang, B. Beinke, B. Al Kindhi, and M. Wiering, "Comparing Machine Learning Algorithms with or without Feature Extraction for DNA Classification," Nov. 2020, [Online]. Available: http://arxiv.org/abs/2011.00485.

[13] Z. Li, L. Huang, and C. Yu, "Advanced Prediction of Hepatic Oncogenic Transformation in HBV Patients via RNA-Seq Data Analysis and Deep Learning Techniques," *Int. J. Mol. Sci.*, vol. 25, no. 18, Sep. 2024, doi: 10.3390/ijms25189827.

[14] A. S. Osarumwense and B. Moses Eromosele, "Detection of Hepatitis (A, B, C, D and E) Viruses Using Machine Learning," *International Journal of Academic Pedagogical Research*, vol. 4, no. 5, pp. 19–35, 2020, [Online]. Available: www.ijeais.org/ijapr.

[15] T. H. Yang *et al.*, "A Fluorescence Imaging- and Deep Learning-Based Approach for Detecting Hepatitis B Virus Integration into Host Genomes," *Journal of Imaging Informatics in Medicine*, 2026, doi: 10.1007/s10278-026-01983-3.

[16] C. Wu *et al.*, "DeepHBV: a deep learning model to predict hepatitis B virus (HBV) integration sites," *BMC Ecol. Evol.*, vol. 21, no. 1, Dec. 2021, doi: 10.1186/s12862-021-01869-8.

[17] L. Fadia, V. Shah, M. Hassanzadeh, J. Wu, and M. Ahmadi, "An Efficient Method For Classification of Different Types of Hepatitis Virus Using Genomic Signal Processing and Machine Learning," IEOM Society International, Jan. 2025. doi: 10.46254/wc01.20240150.

[18] A. El-Tohamy, H. A. Maghwary, and N. Badr, "A Deep Learning Approach for Viral DNA Sequence Classification using Genetic Algorithm," *IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 13, no. 8, pp. 530–538, 2022, doi: 10.14569/IJACSA.2022.0130861.

[19] A. Kautsar, "DNA Sequence Classification Using Machine Learning Models Based on k-mer Features," *Journal of Computers and Digital Business*, vol. 4, no. 2, pp. 100–105, May 2025, doi: 10.56427/jcbd.v4i2.762.

[20] T. Sadad, R. A. Aurangzeb, M. Safran, Imran, S. Alfarhood, and J. Kim, "Classification of Highly Divergent Viruses from DNA/RNA Sequence Using Transformer-Based Models," *Biomedicines*, vol. 11, no. 5, May 2023, doi: 10.3390/biomedicines11051323.

[21] S. A. Salloum, O. Almomani, A. Almomani, K. M. Alomari, and V. Arya, "Semantic-Enhanced Classification of DNA Sequences Using Machine Learning and Deep Learning Approaches," *Int. J. Semant. Web Inf. Syst.*, vol. 22, no. 1, pp. 1–28, Apr. 2026, doi: 10.4018/IJSWIS.407404.

[22] C. Galià-Camps, C. Pegueroles, X. Turon, C. Carreras, and M. Pascual, "Genome composition and GC content influence loci distribution in reduced representation genomic studies," *BMC Genomics*, vol. 25, no. 1, Dec. 2024, doi: 10.1186/s12864-024-10312-3.

[23] M. L. Russell, N. Simon, P. Bradley, and F. A. Matsen, "Statistical inference reveals the role of length, GC content, and local sequence in V(D)J nucleotide trimming," *Elife*, vol. 12, May 2023, doi: 10.7554/eLife.85145.

[24] V. Nerkar and V. Kimbahune, "Deep Learning Approaches in Genomic Analysis: A Review of DNA Sequence Classification Techniques," *International Journal of Scientific Research & Engineering Trends*, vol. 10, no. 2, pp. 2395–566, Apr. 2024, doi: doi.org/10.61137/ijsret.vol.10.issue2.153.

[25] A. Lopez-del Rio, M. Martin, A. Perera-Lluna, and R. Saidi, "Effect of sequence padding on the performance of deep learning models in archaeal protein functional prediction," *Sci. Rep.*, vol. 10, no. 1, Dec. 2020, doi: 10.1038/s41598-020-71450-8.

[26] F. Farias, T. Ludermir, and C. Bastos-Filho, "Similarity Based Stratified Splitting: an approach to train better classifiers," Oct. 2020, [Online]. Available: http://arxiv.org/abs/2010.06099.

[27] M. Sivakumar, S. Parthasarathy, and T. Padmapriya, "Trade-off between training and testing ratio in machine learning for medical image processing," *PeerJ Comput. Sci.*, vol. 10, p. 15, Sep. 2024, doi: 10.7717/PEERJ-CS.2245.
[28] F. Akar, O. M. Katipoğlu, S. N. Yeşilyurt, and M. B. Han Taş, "Evaluation of Tree-Based Machine Learning and Deep Learning Techniques in Temperature-Based Potential Evapotranspiration Prediction," *Pol. J. Environ. Stud.*, vol. 32, no. 2, pp. 1009–1023, 2023, doi: 10.15244/pjoes/156927.
[29] E. Tabane, E. Mnkandla, and Z. Wang, "Optimizing DNA Sequence Classification via a Deep Learning Hybrid of LSTM and CNN Architecture," *Applied Sciences (Switzerland)*, vol. 15, no. 15, Aug. 2025, doi: 10.3390/app15158225.
[30] L. Yang, "Theoretical Analysis of Adam Optimizer in the Presence of Gradient Skewness," *International Journal of Applied Science*, vol. 7, no. 2, pp. 27–42, Oct. 2024, doi: 10.30560/ijas.v7n2p27.
[31] R. Arabelli, T. Bernatin, and V. Veeramsetty, "Water quality assessment for aquaculture using deep neural network," *Desalination Water Treat.*, vol. 321, Jan. 2025, doi: 10.1016/j.dwt.2025.101016.
[32] M. Anshori and M. Musthofa, "Outperforming DNN Using MLP in Water Quality Assessment for Aquaculture," *Journal of Applied Informatics and Computing (JAIC)*, vol. 10, no. 1, p. 355, Feb. 2026, doi: 10.30871/jaic.v10i1.11835.
[33] A. P. Sari, Kariyam, F. Wijayanto, and E. Widodo, "COMPARATIVE EVALUATION OF CNN ARCHITECTURES FOR ROAD DAMAGE CLASSIFICATION USING DIGITAL IMAGES IN SLEMAN REGENCY," *Barekeng*, vol. 20, no. 3, pp. 2681–2692, Sep. 2026, doi: 10.30598/barekengvol20iss3pp2681-2692.
[34] D. Chicco, N. Tötsch, and G. Jurman, "The matthews correlation coefficient (Mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation," *BioData Min.*, vol. 14, pp. 1–22, 2021, doi: 10.1186/s13040-021-00244-z.
[35] D. Chicco and G. Jurman, "The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification," *BioData Min.*, vol. 16, no. 1, Dec. 2023, doi: 10.1186/s13040-023-00322-4.
[36] A. Tampuu, Z. Bzhalava, J. Dillner, and R. Vicente, "ViraMiner: Deep learning on raw DNA sequences for identifying viral genomes in human samples," *PLoS One*, vol. 14, no. 9, Sep. 2019, doi: 10.1371/journal.pone.0222271.
[37] N. G. Nguyen *et al.*, "DNA Sequence Classification by Convolutional Neural Network," *J. Biomed. Sci. Eng.*, vol. 09, no. 05, pp. 280–286, 2016, doi: 10.4236/jbise.2016.95021.
[38] A. Srinivasulu, G. Sreenivasulu, and O. O. Oyerinde, "Real-time Classification and Hepatitis B Detection with Evolutionary Data Mining Approach," *Advances in Engineering and Intelligence Systems*, vol. 2, no. 1, pp. 62–70, Dec. 2023, doi: 10.22034/aeis.2023.384167.1074.
[39] N. Santhi and N. Ramasamy, "A Lightweight Hybrid Deep Learning and Adaptive Texture Encoding Framework for Accurate Hepatitis Prediction Using Ultrasound Liver Images," *IJEDR26A1034 International Journal of Engineering Development and Research*, vol. 14, no. 1, pp. 960–969, 2026, [Online]. Available: www.ijedr.org.