**Enhancing Viral DNA Sequence Classification with a Novel 1D Convolutional Neural Network Approach**

**Abstract**

Classifying DNA sequences is crucial for identifying and characterizing viral pathogens. Traditionally, this involves analyzing k-mers, substrings of length k from DNA sequences. Recent advancements in machine learning have enhanced this process, using models trained on large datasets to improve classification accuracy. Techniques such as k-mer frequency encoding, Position-Specific Scoring Matrices (PSSM), and ensemble methods like Random Forest and Gradient Boosting have shown promise.

This study explores the impact of various factors, including k-mer size and feature engineering methods, on classification performance. We introduce a novel approach using a one-dimensional convolutional neural network (1D CNN), which splits the DNA sequence into overlapping subsets and employs a voting scheme for prediction. This method leverages the 1D CNN’s ability to capture local patterns, aiming to boost classification accuracy and robustness. Our results indicate that this new model significantly surpasses traditional methods, offering a promising direction for future research and practical applications in viral pathogen detection.

**Introduction**

Classification, or taxonomy, is the scientific process of categorizing and organizing living organisms into a hierarchical system based on shared characteristics. This framework provides a standardized method for naming and classifying species, which involves grouping organisms into various ranks or categories according to their similarities and differences. The most fundamental unit of classification is the species, defined as a group of individuals capable of interbreeding and producing fertile offspring. The hierarchical system typically starts with broad categories and narrows down to more specific groups, traditionally including eight ranks: Domain, Kingdom, Phylum, Class, Order, Family, Genus, and Species.

Historically, taxonomy relied on morphological characteristics—observable attributes such as size, shape, and color—to classify organisms. However, advancements in molecular biology and DNA sequencing have revolutionized taxonomy by providing a more objective and accurate method for determining relationships among species. DNA carries hereditary information passed down through generations, allowing scientists to categorize organisms based on their genetic makeup rather than solely on physical traits.

A 2020 study by Sosa et al. suggests that while DNA classification offers significant advantages, morphological and ecological factors should also be considered to enhance species recognition and classification. Thus, DNA classification should complement rather than replace traditional methods.

Machine learning (ML) has emerged as a transformative technology in data analysis, enabling systems to learn from data, identify patterns, and make decisions with minimal human intervention. In the realm of DNA classification, ML techniques offer increased efficiency and accuracy by automating processes such as DNA sequence segmentation. ML algorithms, including Support Vector Machines (SVMs) and Artificial Neural Networks (ANNs), have been applied to classify genetic data, enhancing the speed and precision of analysis compared to traditional methods. These technologies are also valuable for identifying genetic variations, determining genomic relationships, and predicting the functions of genetic elements.

By integrating ML with traditional classification methods, researchers can achieve higher accuracy and efficiency in DNA sequence analysis. This synergy opens new possibilities in genetics, providing deeper insights into gene functions, evolutionary relationships, and the molecular basis of diseases.

### Methodology

We employed a hierarchical learning approach to determine the taxonomy of a given virus from its DNA sequence. Our methodology involves multiple stages of classification, each focusing on a different taxonomic level, which allows for a more precise and manageable classification process. The Hierarchical Classification Process can be summarized as follow:

**Order Prediction**: The first stage involves predicting the order of the virus using a dedicated classifier. This classifier is trained to differentiate between various viral orders, providing a foundational classification that guides subsequent steps.

**Family Prediction**: Once the order is predicted, a second classifier is used to determine the family within the predicted order. Although it is possible to apply additional classifiers to predict the genus and species of the virus, our research focuses on the classifier itself. Therefore, we considered only two orders: Norzivirales and Timlovirales.

**Norzivirales Classification**: For the Norzivirales order, a family classifier is employed to predict the specific family of the virus. We focused on three families within this order and excluded the Duinviridae family due to its limited number of species.

**Timlovirales Classification**: Similarly, for the Timlovirales order, a family classifier is used to predict the family of the virus, focusing on two primary families.

This hierarchical approach allows us to accurately classify the virus based on its DNA sequence, providing valuable insights into its characteristics and potential impact. By breaking down the classification process into smaller, more manageable steps, we can effectively navigate the complex taxonomy of viruses and enhance our ability to study and combat them.

#### Data Preparation and Preprocessing

**Data Set**: In the following subsection, we discuss the dataset used in this study. The dataset comprises DNA sequences of viruses from the Norzivirales and Timlovirales orders, along with their respective families.

**Preprocessing**: The DNA sequences undergo preprocessing steps to ensure they are in a suitable format for machine learning models. This includes:

* **K-mer Calculation**: Generating k-mers (substrings of length k) from each DNA sequence to capture essential patterns.
* **Encoding**: Transforming the k-mers into feature vectors that represent the DNA sequences.
* **Splitting into Subsets**: Dividing the encoded sequences into overlapping subsets to capture local sequence patterns and context.

#### Machine Learning Model

**1D CNN Model**: We employ a one-dimensional convolutional neural network (1D CNN) for the classification tasks. The model is trained using the overlapping subsets of encoded DNA sequences, allowing it to learn and capture patterns effectively. The architecture includes convolutional layers, pooling layers, and dense layers to handle the hierarchical structure of the data.

**Prediction and Aggregation**: For the prediction phase, the test DNA sequences are encoded and split into overlapping subsets. The trained 1D CNN model predicts the class for each subset. A voting mechanism is used to aggregate the predictions and determine the final classification, ensuring robustness and improving accuracy.

### Conclusion

By employing a hierarchical learning approach with dedicated classifiers at each taxonomic level, combined with advanced preprocessing techniques and a 1D CNN model, we achieve a precise and efficient method for classifying viral DNA sequences. This methodology not only enhances our understanding of viral taxonomy but also provides a framework for future studies in virology.

1. **Data set**

The dataset used in this research was obtained from the National Center for Biotechnology Information (NCBI). We utilized RefSeq data, which consists of high-quality, curated sequences that have been reviewed and annotated by experts. This ensures both accuracy and reliability. RefSeq provides a single, well-characterized reference sequence for each virus, thereby reducing redundancy and simplifying data interpretation. Additionally, RefSeq entries have stable identifiers, which are helpful for consistent referencing and reporting.

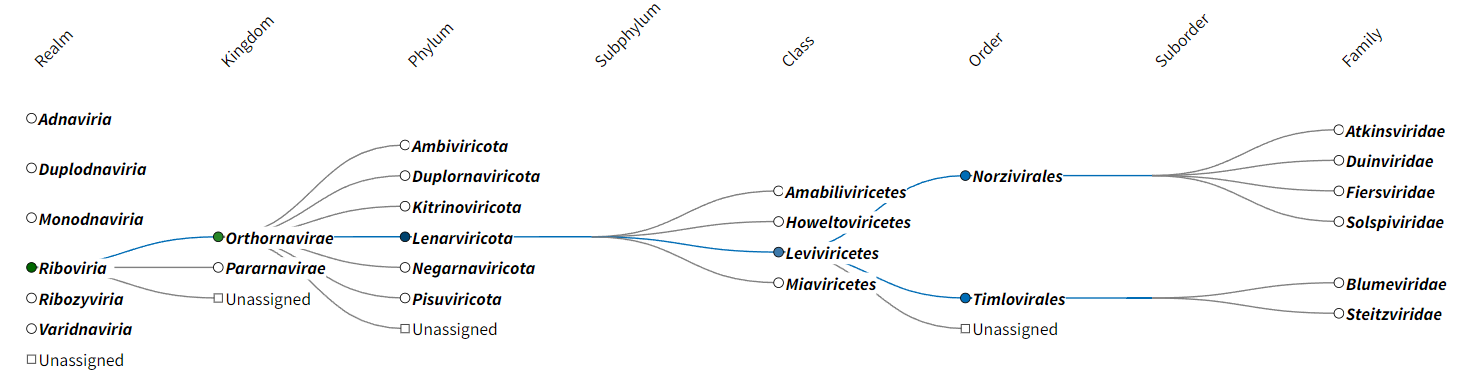
To obtain the RefSeq sequences from the database, we used the most recent Virus Metadata Resource (VMR) spreadsheet available from the International Committee on Taxonomy of Viruses (ICTV) website at <https://ictv.global/vmr/current> and available on the associated github. This file contains metadata for all viruses. Using the RefSeq accession numbers from the VMR spreadsheet, we downloaded the DNA sequences for the taxa considered in this research.

In this study, we employed a classifier to categorize viruses into two orders: Norzivirales and Timlovirales. Furthermore, we used another classifier to differentiate among the families Atkinsviridae, Fiersviridae, and Solspiviridae, all of which belong to the order Norzivirales. This approach allowed us to systematically analyze and classify the viral sequences with a high degree of precision. The following table shows the number of familes, species in each order. The following table shows the number of families and species in each order

|  |  |  |
| --- | --- | --- |
| Order | Family | Species |
| Norzivirales | Atkinsviridae | 262 |
|  | Fiersviridae | 827 |
|  | Solspiviridae | 101 |
|  | Duinviridae | 12 |
| Timlovirales | Blumeviridae | 38 |
|  | Steitzviridae | 1289 |

**Meathodology**

We employed a hierarchical learning approach to determine the taxonomy of a given virus from its DNA sequence. In this research, we begin by predicting the order of the virus using a dedicated classifier. Once the order is predicted, we use a second classifier to predict the family. Although it is possible to apply another classifier to predict the genus and species of the virus, our focus in this research is on the classifier itself, so we considered only the two orders Norzivirales and Timlovirales. Once the model predicts that the virus belongs to Norzivirales, a family classifier is used to predict the family of the virus. We considered only three families for Norzivirales and two families for Timlovirales. We excluded the family Duinviridae because it contains few species. This hierarchical approach allows us to accurately classify the virus based on its DNA sequence, providing valuable information for understanding its characteristics and potential impact. By breaking down the classification process into smaller, more manageable steps, we can effectively navigate the complex taxonomy of viruses and enhance our ability to study and combat them. In the following subsection we discuss the used data set, the preprocessing of the data, the machine learning model.



### Hierarchical Classes

The ICTV scheme organizes classes into a hierarchical taxonomic tree. From the highest to the deepest levels, these are: Realm, Kingdom, Phylum, Subphylum, Class, Order, Family, Subfamily, Genus, and Species. In hierarchical classification, our aim is to predict a set of hierarchically structured classes for each virus. A comprehensive review of hierarchical classification is available in [81][82].

Hierarchical classifiers can be categorized into three types based on how they utilize hierarchical information: the flat classifier approach, the local classifier approach, and the global classifier (big-bang) strategy.

#### Flat Classifier Approach: The flat approach is the most straightforward method for hierarchical classification. It involves using a standard multi-class classifier to make predictions at the lowest level of the hierarchy. Subsequently, a post-processing phase is applied to assign higher-level labels based on these predictions.

#### Local Classifier Approach: The local approach explicitly incorporates the class hierarchy into the categorization process. This technique utilizes a combination of multi-class non-hierarchical classifiers, often referred to as base classifiers, which operate at different levels or nodes of the hierarchy. Each classifier is responsible for distinguishing between immediate child classes of a given parent class, thus leveraging the hierarchical structure during classification. There are three types of local hierarchical classification, depending on what data the classifier is trained on. The inference process is essentially the same across these approaches:

1. **Local Classifiers per Node (LCN):** A classifier is trained for each internal node in the hierarchy to distinguish between its child nodes. When classifying a new instance, the system traverses the hierarchy from the root, making decisions at each node until it reaches a leaf node. For example, in virus taxonomy, a classifier at the "Order" level decides between different orders, and for each order, another classifier distinguishes between families within that order.
2. **Local Classifiers per Parent Node (LCPN)**: Similar to LCN, but instead of having a classifier for each node, there is one classifier for each parent node, distinguishing among its children. This approach reduces the number of classifiers compared to LCN and is used similarly by traversing the hierarchy from the root. For instance, a classifier at the "Order" level decides among orders, and if it chooses "Norzivirales," a classifier specific to "Norzivirales" determines the family. In this paper, we adapted this approach.
3. **Local Classifiers per Level (LCL)**: A single classifier is trained for each level of the hierarchy. For a given instance, the classifier for the first level determines the category, and this process is repeated at each subsequent level using the corresponding classifier. For example, one classifier determines the order, another determines the family based on the predicted order, and so on.

#### Global Classifier Approach (Big-Bang Strategy): The global approach defines a single optimization problem that considers the entire class hierarchy simultaneously. Instead of handling hierarchical information at different levels or nodes independently, this strategy integrates the hierarchical structure into a unified model, aiming to optimize the classification across all levels of the hierarchy in one step.

In this paper, we adopted the Local Classifiers per Parent Node (LCPN) approach. This approach reduces the number of classifiers compared to the Local Classifiers per Node (LCN) method, while still capturing hierarchical information. Unlike the flat classifier approach, LCPN explicitly incorporates the class hierarchy. Moreover, it is more intuitive and simpler than the big-bang classification strategy.

Since the focus of this research is on the LCPN approach, we used one classifier to distinguish the order and two classifiers to predict the family as shown in figure 1.

.

=============

In the following sections, we will provide a comprehensive explanation of the steps involved in applying flat, local, and global classifiers to categorize samples into hierarchical classes.

**1. Local Classifiers per Node (LCN):** A classifier is trained for each internal node in the hierarchy to distinguish between its child nodes. When classifying a new instance, it traverses the hierarchy from the root, making decisions at each node until it reaches a leaf node.In virus taxonomy, a classifier at the "Order" level decides between different orders, and for each order, another classifier distinguishes between families within that order.

**2. Local Classifiers per Parent Node (LCPN):** Similar to LCN, but instead of having a classifier for each node, there is one classifier for each parent node, distinguishing among its children.This approach reduces the number of classifiers compared to LCN and is used similarly by traversing the hierarchy from the root.A classifier at the "Order" level decides among orders, and if it chooses "Norzivirales", a classifier specific to "Norzivirales" determines the family.

**3. Local Classifiers per Level (LCL):** A single classifier is trained for each level of the hierarchy.For a given instance, the classifier for the first level determines the category, and this process is repeated at each subsequent level using the corresponding classifier.One classifier determines the order, another determines the family based on the predicted order, and so on.

**4. Global Hierarchical Classifier (GHC):** A single classifier is trained to directly predict the full path in the hierarchy, considering all levels simultaneously.This approach can be complex due to the need to handle the entire hierarchical structure in a single model but can be effective in capturing dependencies between levels.The classifier directly predicts the order, family, genus, and species of a virus in one step.

**5. Big-Bang Approach:** A single, flat classifier is trained using a special encoding of hierarchical information, where the output space includes all possible paths in the hierarchy. This approach can be computationally intensive but is straightforward in terms of classification as it considers the entire hierarchy at once.The classifier outputs a specific path like "Norzivirales -> Atkinsviridae -> VirusA".

**6. Hybrid Approaches:** Combines aspects of local and global approaches to leverage their strengths and mitigate their weaknesses.May involve using local classifiers for certain levels and a global classifier for others, or incorporating hierarchical regularization into a global classifier.Using local classifiers for the top levels of the hierarchy and a global classifier for the lower levels where fine-grained distinctions are needed.

**Advantages and Challenges:** Hierarchical classifiers can handle large and complex label spaces efficiently.By leveraging the hierarchical structure, these classifiers can achieve better accuracy, especially in domains with nested categories.The hierarchical structure can make the decision-making process more interpretable.

**Challenges**: Designing and training hierarchical classifiers can be complex, especially for deep hierarchies. They often require large amounts of labeled data to train effectively at all levels of the hierarchy. Mistakes at higher levels of the hierarchy can propagate downwards, affecting the final classification.

Hierarchical classifiers offer a structured approach to classification tasks involving nested categories. Choosing the right type of hierarchical classifier depends on the specific problem, the available data, and the desired balance between complexity and performance. Each type has its strengths and can be tailored to fit the needs of different applications, such as biological taxonomy, document classification, and image categorization.

flat classification approach,

**The flat classification approach, which is the simplest one to deal with hierarchical classification problems, consists of completely ignoring the class hierarchy, typically predicting only classes at the leaf nodes. This approach behaves like a traditional classification algorithm during training and testing. However, it provides an indirect solution to the problem of hierarchical classification, because, when a leaf class is assigned to an example, one can consider that all its ancestor classes are also implicitly assigned to that instance (recall that we assume a \IS-A" class hierarchy).**

However, this very simple approach has the serious disadvantage of having to build a classifier to discriminate among a large number of classes (all leaf classes), without exploring information about parent-child class relationships present in the class hierarchy.

One simple, straight-forward approach for taxonomic classification is **flat classification**. This is where you don’t bother yourself with those pesky parent-categories, and just classify each example to its final, leaf-level label. Siamese cat is the same to a French Bulldog as it is to a Sphynx cat (also known as “that weird, no-fur one”).

## ****Pros and Cons of Flat Classification****

* The obvious advantage of this approach is its simplicity. You come, you see, you classify. This is a simple solution, which can be easily implemented with one, out-of-the-box classifier.
* On the cons side, you obviously lose some important information. The natural hierarchy of the data could have highly valuable classification mojo, and ignoring those parent-child class relationships could reduce performance.

4 Local Classi¯er Approaches

**1D Convolution Neural Network**

**in**

**\**

In this research we used sequences of Vertebrate viruses obtained from Virosaurus database in FASTA format. Virosaurus contains full-length genomes (monopartite genomes) or segments (segmented genomes) for all virus families comprising at least one species infecting vertebrates. Virosaurus also provides complete virus sequence dataset for all those viruses, which comprises complete genomes for non-segmented viruses, and complete segments for segmented viruses. The FASTA headers have been annotated with metadata, such as the viral nucleic acid (RNA, DNA, or RNA/DNA), and other information like the virus species, hosts, virus family, taxonomy identifier, and the official acronym name of the species. The header contains 11 different topics annotated by a controlled vocabulary. Data comes from GenBank, ICTV, ViralZone and manual curation. FASTA header:

[81] C. N. Silla and A. A. Freitas. A survey of hierarchical classification across different application domains. Data Mining and Knowledge Discovery,

22(1):31–72, 2011.

[82] A. A. Freitas and A. Carvalho. A tutorial on hierarchical classification with applications in bioinformatics. In D. Taniar, editor, Research and Trends in Data Mining Technologies and Applications, pages 175–208, 2007.