VRIJE UNIVERSITEIT BRUSSEL

UNIVERSITÉ LIBRE DE BRUXELLES

Review of 'Comparing sequences without using alignments: application to HIV/SIV subtyping'

Supervised by Prof. Mathieu Defrance (*Machine Learning Group* - *ULB*)
& Prof. Wim Vranken (*Bioengineering Sciences Department* - *VUB*)

Master's in Computer Science

Tomás GUIJA VALIENTE
Benjamin OBERTHÜR

Abstract

This article presents a review of an innovative method for alignment-free sequence comparison in the field of bioinformatics. The method, introduced in a relatively early published article, enables the identification of evolutionary relationships between sequences and highlights divisions into subtypes. Despite the absence of a description for the cluster merging method in the original article, the remaining parts of the algorithm were successfully re-implemented, demonstrating that the experiments were not environment-dependent and consistently produced reliable results. Furthermore, comparisons with more recent methods were discussed, acknowledging the possibility of newer approaches offering improved efficiency, runtime, and accuracy.

The review emphasizes the significance of the alignment-free sequence comparison approach, particularly in addressing challenges posed by sequence alterations and the limitations of classic alignment methods. By leveraging the concept of N-words (or k-mers in today's scientific literature), dissimilarity matrices and clustering trees were constructed, providing insights into the evolutionary relationships among the sequences.

The article concludes by highlighting the potential of the reviewed method as a valuable tool in phylogenetic analysis. It suggests avenues for further research, including exploring alternative clustering techniques and investigating the impact of various distance measures for enhanced accuracy. Overall, this review contributes to the understanding of alignment-free sequence comparison methods and their applications in studying evolutionary relationships among biological sequences.

Contents

I. Introduction and summary of the article

This article describes the problem in sequence comparison, which arises when attempting to compare a large number of sequences together. Classic sequence alignment methods often struggle to effectively handle certain types of sequence alterations, such as insertions or deletions. Intuitive approaches to address this problem without alignment involve examining the frequency of nucleotides or amino acids. However, this method lacks meaningfulness, as sequences with similar frequencies can still exhibit significant differences. A more sophisticated and effective approach is dealing with what [Did+07] calls N-words. With those, we can compute dissimilarities between sequences [KL94] which can help us show evolutionary relationships between sequences.

I.1 Local decoding method of order N

The first step of this analysis presented by [Did+07] is the local decoding method of order N, or N-local decoding.

Definition I..1 (N-words)

> In a sequence, an N-word is a **contiguous sub-sequence of size** N within the given sequence. The set of N-words for a sequence consists of all its sub-sequences of size N.

Example.

> The set of the 3-words of the sequence $AGTACGT$ is AGT, GTA, TAC, ACG, CGT.

Let $S = S_1S_2\dots S_i\dots S_{|S|}$ be a sequence, where i denotes a site (or index) within S. For a given $N \in \mathbb{N}^*$, we consider the set of N-words of S covering the site i.

Definition I..2 (Direct relation)

> Two sites are said **directly related** if they have the same position in two (or more) occurences of the same N-word.

Example.

> (We take the example on Figure 6a in [Did+07]). Let seq1 = CATTG TCCGC **T**GGAC CACAC and seq2 = CACT**T** GGACA CATAC CATGC. We consider the site 11 in seq1 and the site 5 in seq2 (bolded in their definitions), and look at the 5-words covering this site (contained in the sites colored in red).

C	C	G	C	**T**	G	G	A	C
C	C	G	C	T				
	C	G	C	T	G			
		G	C	T	G	G		
			C	T	G	G	A	
				T	**G**	**G**	**A**	**C**

C	A	C	T	**T**	G	G	A	C
C	A	C	T	T				
	A	C	T	T	G			
		C	T	T	G	G		
			T	T	G	G	A	
				T	**G**	**G**	**A**	**C**

> The 5-word TGGAC appears in both these sequences, and the sites 11 in seq1 and 5 in seq2 are both in first position of the 5-word, so these two sites are **directly related**

Definition I..3 (Transitivity and transitive closure)

> Let $\mathcal{R}$ and $\mathcal{R}'$ be binary relations. $\mathcal{R}$ is said to be **transitive** if it respects the following property [Cau21]:
>
> $$a\mathcal{R}b \wedge b\mathcal{R}c \implies a\mathcal{R}c$$
>
> $\mathcal{R}'$ is the **transitive closure** of $\mathcal{R}$ if [Sch77]:
>
> $$\forall a,\, b;\; a\mathcal{R}b \implies a\mathcal{R}'b$$
>
> and
>
> $$\forall a,\, b,\, c;\; a\mathcal{R}b \wedge b\mathcal{R}c \implies a\mathcal{R}'c$$

Definition I..4

> We define the (simple) relation between two sites as the transitive closure of the direct relation. Therefore, we say that two sites are related if there is a (finite) chain of direct relations linking those sites.

These related sites can be categorized into distinct equivalence classes, as the relation we described satisfies the properties of an equivalence relation, namely, reflexivity, symmetry, and transitivity. [Did+07] refers to these equivalence classes as N-classes.

Definition I..5 (N-classes)

An N-class can be defined as follows,with a being the identifier of the N-class:

$$C(a) = \{x \in [\![1,\, |S|]\!]^1;\ x \text{ is related with } a\}$$

Consequently, by assigning a unique identifier to each N-class, we can rename every site in the entire sequence by appending the nucleotide (or amino acid) with the identifier of the class to which it belongs. However, there may exist sites that are not related to any other site in any other sequence, resulting in singleton N-classes. To avoid excessive and unnecessary identifiers, which could lead to unreadable representations, we represent these sites solely by their nucleotide or amino acid.

I.2 Dissimilarity matrix and clustering tree

Now that we have our new sequences divided into disjoint N-classes of sites, we aim to compare each sequence. To achieve that, we need to select a measure of (dis-)similarity. [Did+07] chose the one defined in [Did+06].

Let's introduce the following notation. $|s|_x$ is the number of occurences of the identifier x in the sequence s. For each pair of sequence and identifier $(s,\, x)$, we compute the value of $|s|_x$.

Example.

We consider the following rewriten sequences:

seq1	$C\,A\,T_0\,T_1\,G_0\,T_2\,C_0\,C_1\,G\,C_2\,T_3\,G_1\,G_2\,A_0\,C_3\,C_4\,A\,C\,A\,C\,C\,T_0\,T_1\,G_0\,T_2\,C_0\,C_1\,C\,T\,A$
seq2	$C_5\,A_1\,C_6\,T_4\,T_3\,G_1\,G_2\,A_0\,C_3\,A\,C\,A\,T\,A\,C\,C\,A\,T\,G\,C$
seq3	$C_5\,A_1\,C_6\,T_4\,T_3\,C\,T\,T\,T\,C\,C_2\,T_3\,G_1\,G_2\,A_0\,C_3\,C_4\,T\,C\,C$

Table 1: Rewriten sequence (picked from [Did+07])

	seq1	seq2	seq3
A_0	1	1	1
A_1	0	1	1
T_0	2	0	0
T_1	2	0	0
T_2	2	0	0
T_3	1	1	2
T_4	0	1	1
C_0	2	0	0
C_1	2	0	0
C_2	1	0	1
C_3	1	1	1
C_4	1	0	1
C_5	0	1	1
C_6	0	1	1
G_0	2	0	0
G_1	1	1	1
G_2	1	1	1

Table 2: Count of each identified site in each sequence

Then to compute the similarity between each sequences, we apply the following formula:

$$\text{sim}(\text{seq},\, \text{seq}') = \frac{\sum_x \min(|\text{seq}|_x,\, |\text{seq}'|_x)}{\min(|\text{seq}|,\, |\text{seq}'|)}$$

Example.

[1] $[\![\cdot,\, \cdot]\!] : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$: Integer interval, $[\![a,\, b]\!] = \{a,\, a+1,\, \ldots,\, b\}$

for $i,\, j \in \{\text{seq1, seq2, seq3}\}$:

$$\mathrm{Sim} = (\mathrm{sim}(i,\, j))_{i<j} = \begin{pmatrix} - & \frac{5}{20} & \frac{7}{20} \\ & - & \frac{9}{20} \\ & & - \end{pmatrix} = \begin{pmatrix} - & 0.25 & 0.35 \\ & - & 0.45 \\ & & - \end{pmatrix}$$

The dissimilarity (or distance) can easily be obtained by taking the complement to 1 of the similarity.

Now that we have our mesure of distance between sequences, we can perform an agglomerative hierarchical clustering [And73] that can give us an idea of a potential mutative evolution of HIV and SIV, that can be interpreted as subtypes of these viruses. However, in order to perform this clustering, we need a method to compute the distance between clusters containing multiple sequences. The article does not present one, so we had to design our own, described in Subsection II.2

I.3 Bootstrapping

Finally, in order to have more confidence in the results, the authors use the statistical technique of 'bootstrapping' [Efr79]. In the case of sequence comparison, the method proposed in [Fel80] was considered and adapted to local decoded sequences. The motivation behind using this method is that the results obtained may possess a certain degree of uncertainty due to data limitations or noise.

For each sequence in the original data, a random replicate of the same size is created by randomly selecting sites from the original data with replacement and concatenating them. Once these artificial sets have been generated, the same analysis performed on the original set is carried out on the new set. By comparing the results obtained from the original set with those from the artificial sets, the stability of the method can be assessed, taking into account the inherent uncertainty introduced by the data.

II. Material & Method

II.1 Data Sets

We used two different sequence sets to run our experiments: one given from Gilles Didier, co-author of [Did+07], and another that we retrieved ourselves, both coming from the data base maintened by the 'Los Alamos National Laboratory'[2]. The first set contains 66 sequences writen prior to 2007 [3], and the second one has been created from the query to have 50 sequences sampled in 2016 in Germany (so that we look at mutations in a restrained place and time).

II.2 Cluster merging method

For our cluster merging method, we drew inspiration from centroid linkage methods [DH+73]. When considering a cluster $\mathcal{C}$ containing more than one sequence, we create an artificial sequence with a length equal to that of the shortest sequence within the cluster. This artificial sequence is defined as follows:

$$\widetilde{\mathrm{seq}}_i = \arg\max_{s_i;\, s\in\mathcal{C}} \frac{s_i}{|\mathcal{C}|}, \quad \forall i \in [\![1,\, n]\!],\, n = \min_{s\in\mathcal{C}} |s|$$

In simpler terms, for each site of the artificial sequence, we select the N-class that appears most frequently at the corresponding site across the different sequences in the cluster. If multiple N-classes have the same frequency of occurrence, we choose the first element with the highest frequency. To compute the distance between two clusters, we utilize the same formula as shown in Subsection I.2.

III. Experiments and Results

III.1 Original set of sequence

III.1.1 Comparison of the clustering trees

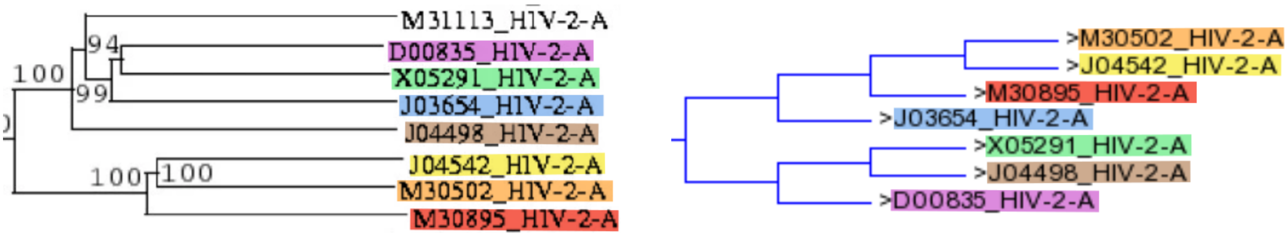
We have followed the procedure outlined in the summary (Section I.) and used the same length of N-words as in the article, i.e. $N = 15$. The resulting dissimilarity matrix for the set of sequences can be found at the following path: `data/Dissimilarity Matrices/dissmatrix__66_sequences.txt` within the repository.

[2] `https://www.hiv.lanl.gov`

[3] The sequences presented in figure 1 of [Did+07], with 4 missing sequences

Since the matrix is too large to be displayed here, please refer to the file for its contents. Additionally, the corresponding cluster tree for this dissimilarity matrix is presented in Figure 4 in the Appendix. It follows the cluster tree from the original article, which can be found in Figure 3.

When looking at both the figures, we can see that they are very alike. Let's take the following sequences in example, that both come from the smaller 'block' of each tree:

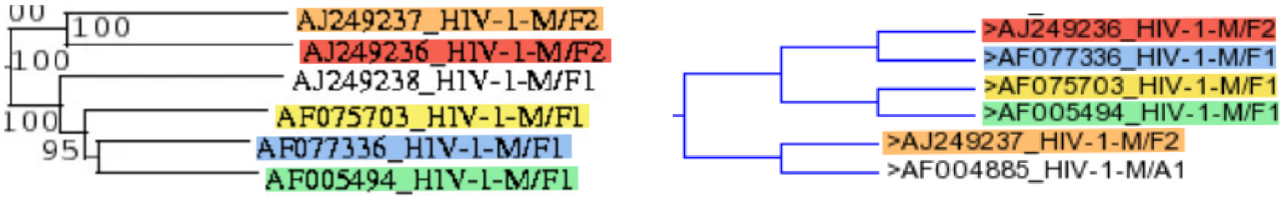


(a) Part of the original tree

(b) Part of the generated tree

On these parts of the trees, we can see that there is a large resemblance between the two. The few differences between the two are permutations between the subtrees (J03654 that should be in the other subtree and D00835 that should be deeper in the subtree), and the sequence M31113 that do not appear in the generated tree, as it is alone as child of the root node. This change can be one of the possibly significant change between the two trees

We can look at a second example:



(a) Another part of the original tree

(b) Another part of the generated tree

There are more difference between these two parts, but still pretty small. AF005494 should take AJ249236's place, and the latter become a sibling of AJ249237, with the additional (non-colored) sequences that should come or leave the subtree

III.1.2 Bootstrapping

We conducted 10 different bootstraps to assess the reliability of our method. The resulting trees can be found at the following path: `data/Tree visualization/Bootstraps_66_sequences`. All the trees exhibit the same overall structure, with three distinct blocks of sequences that remain separate from each other. Additionally, the sequence M31113 consistently maintains the same position in all the bootstrap trees. However, some permutations between sequences within the different blocks can still be observed. These permutations tend to occur in the deeper half of the tree for the "large block".

While the bootstrap trees are not identical to each other or to the original generated tree in terms of their structure, our method still accurately distinguishes between the subtypes of HIV and SIV. All the sequences of subtypes HIV-1-M are grouped together in the larger subtree, and the sequences of subtype HIV-2-A in another subtree. This indicates the robustness of our approach, despite variations in the specific arrangements of the trees and permutations within the sequence blocks.

III.2 'German' set of sequences

In Figure 5, we can remark that the various subtypes of HIV are well-clustered within the clustering tree. Notably, the four sequences belonging to subtype 01AE are grouped together, as are the sequences of subtype B. However, the sequences of subtype C display a slight scattering pattern in the tree. A majority of the subtype C sequences are clustered near the top of the tree, with one sequence positioned slightly higher, and the remaining sequences forming another subtree within the tree.

Based on the observations made for the previous set of sequences, it is worth noting that the bootstraps for the current set - located at `Trees/Bootstraps_50_sequences` in the repository - exhibit a consistent structure.

Furthermore, only minimal permutations among the leaf nodes are observed across the different bootstraps. Considering the previous paragraph, it can be concluded that our algorithm demonstrates a high level of reliability for this particular dataset as well.

IV. Conclusion

In this review, we have observed an innovative method (considering the relatively early publishing year of the article) for alignment-free sequence comparison. This method enables the identification of evolutionary relationships between sequences and highlights divisions into subtypes.

Although the cluster merging method was not described in the article, we were able to re-implement the remaining parts of the described algorithm. This allowed us to demonstrate that the experiments were not environment-dependent, and we obtained consistent results. Additionally, we confirmed the effectiveness and reliability of our cluster merging method.

As mentioned earlier, this article was published in 2007, and since then, other methods may have been developed with better efficiency, runtime, and correctness. This is likely the case with the SeqDistK method [Liu+21]. This method uses the same algorithmic pattern as the article under review but employs seven different k-mer (or N-word) distances. It also utilizes the Unweighted Pair Group Method with Arithmetic Mean algorithm [SM75] to construct the tree.

References

[Cau21] Augustin-Louis Cauchy. *Cours d'analyse de l'École royale polytechnique. Analyse algébrique.* .1ère partie. Imprimerie royale, 1821.

[Sch77] Ernst Schröder. *Der Operationskreis des Logikkalkiils.* Ed. by Teubner-Verlag. Leipzig, 1877.

[And73] Michael R. Anderberg. "CHAPTER 6 - HIERARCHICAL CLUSTERING METHODS". In: *Cluster Analysis for Applications.* Ed. by Michael R. Anderberg. Probability and Mathematical Statistics: A Series of Monographs and Textbooks. Academic Press, 1973, pp. 131–155. DOI: https://doi.org/10.1016/B978-0-12-057650-0.50012-0. URL: https://www.sciencedirect.com/science/article/pii/B9780120576500500120.

[DH+73] Richard O Duda, Peter E Hart, et al. *Pattern classification and scene analysis.* Vol. 3. Wiley New York, 1973.

[SM75] Robert R Sokal and Charles D Michener. "A statistical method for evaluating systematic relationships". In: *Multivariate statistical methods, among-groups covariation* (1975), p. 269.

[Efr79] B. Efron. "Bootstrap Methods: Another Look at the Jackknife". en. In: *Ann. Statist.* 7.1 (1979), pp. 1–26. URL: http://dml.mathdoc.fr/item/1176344552.

[Fel80] J Felsenstein. "Confidence intervals on phylogenies: an approach using bootstrap". In: *Evolution* 39 (1980), pp. 1792–1797.

[KL94] Samuel Karlin and Istvan Ladunga. "Comparisons of eukaryotic genomic sequences." In: *Proceedings of the National Academy of Sciences* 91.26 (1994), pp. 12832–12836.

[Did+06] Gilles Didier et al. "Local Decoding of Sequences and Alignment-Free Comparison". In: *Journal of Computational Biology* 13.8 (2006). PMID: 17061922, pp. 1465–1476. DOI: 10.1089/cmb.2006.13.1465. eprint: https://doi.org/10.1089/cmb.2006.13.1465. URL: https://doi.org/10.1089/cmb.2006.13.1465.

[Did+07] Gilles Didier et al. "Comparing sequences without using alignments: application to HIV/SIV subtyping". In: *BMC Bioinformatics* 8.1 (Jan. 2007), p. 1. ISSN: 1471-2105. DOI: 10.1186/1471-2105-8-1. URL: https://doi.org/10.1186/1471-2105-8-1.

[Liu+21] Xuemei Liu et al. "SeqDistK: a Novel Tool for Alignment-free Phylogenetic Analysis". In: *bioRxiv* (2021). DOI: 10.1101/2021.08.16.456500. eprint: https://www.biorxiv.org/content/early/2021/08/17/2021.08.16.456500.full.pdf. URL: https://www.biorxiv.org/content/early/2021/08/17/2021.08.16.456500.

Appendix

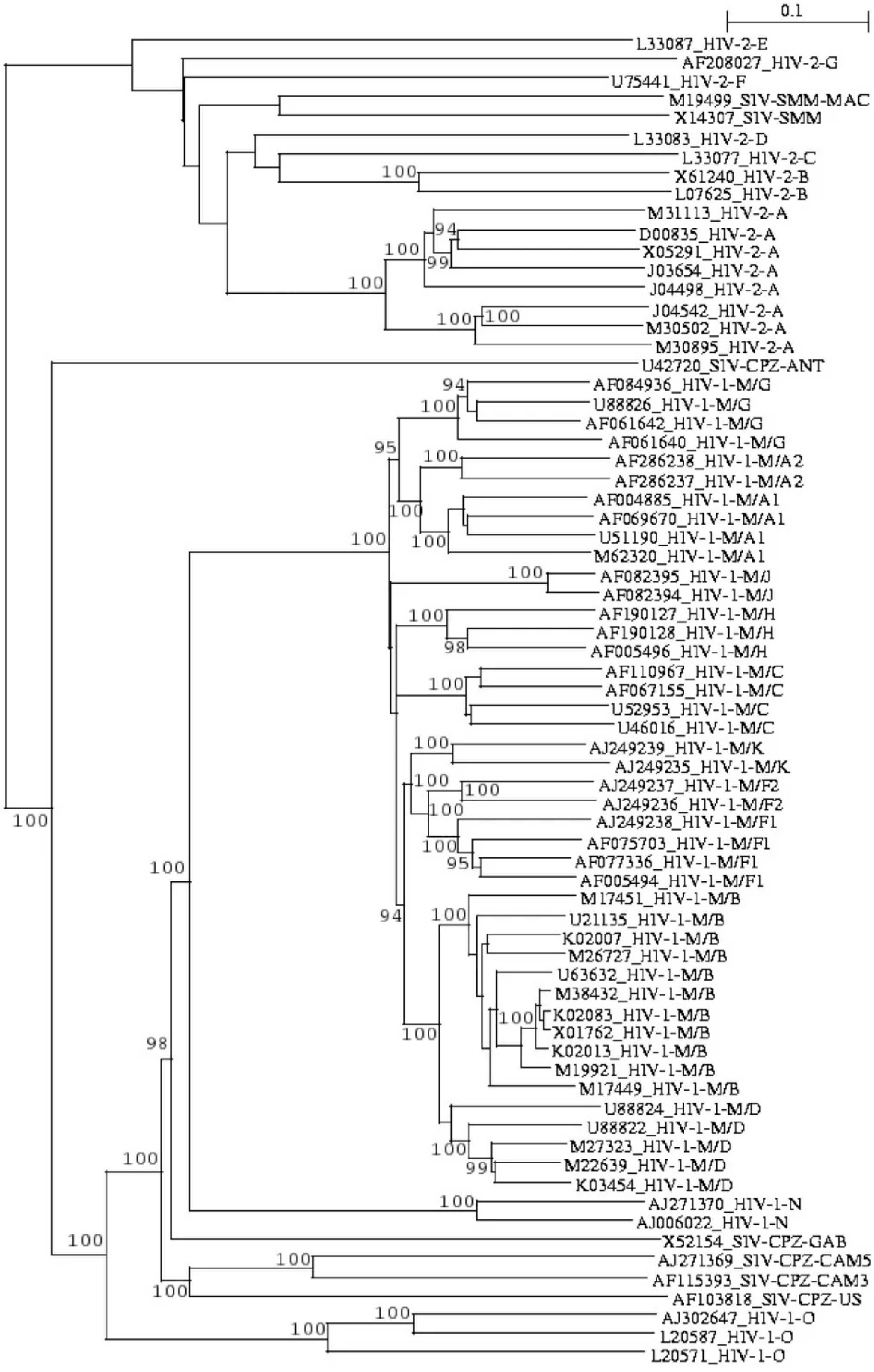


Figure 3: Original tree obtained from the dissimilarity matrix of the sequences of the original article, with $N = 15$ - accesible in the repository at `data/Tree visualization/original_66_tree.png`

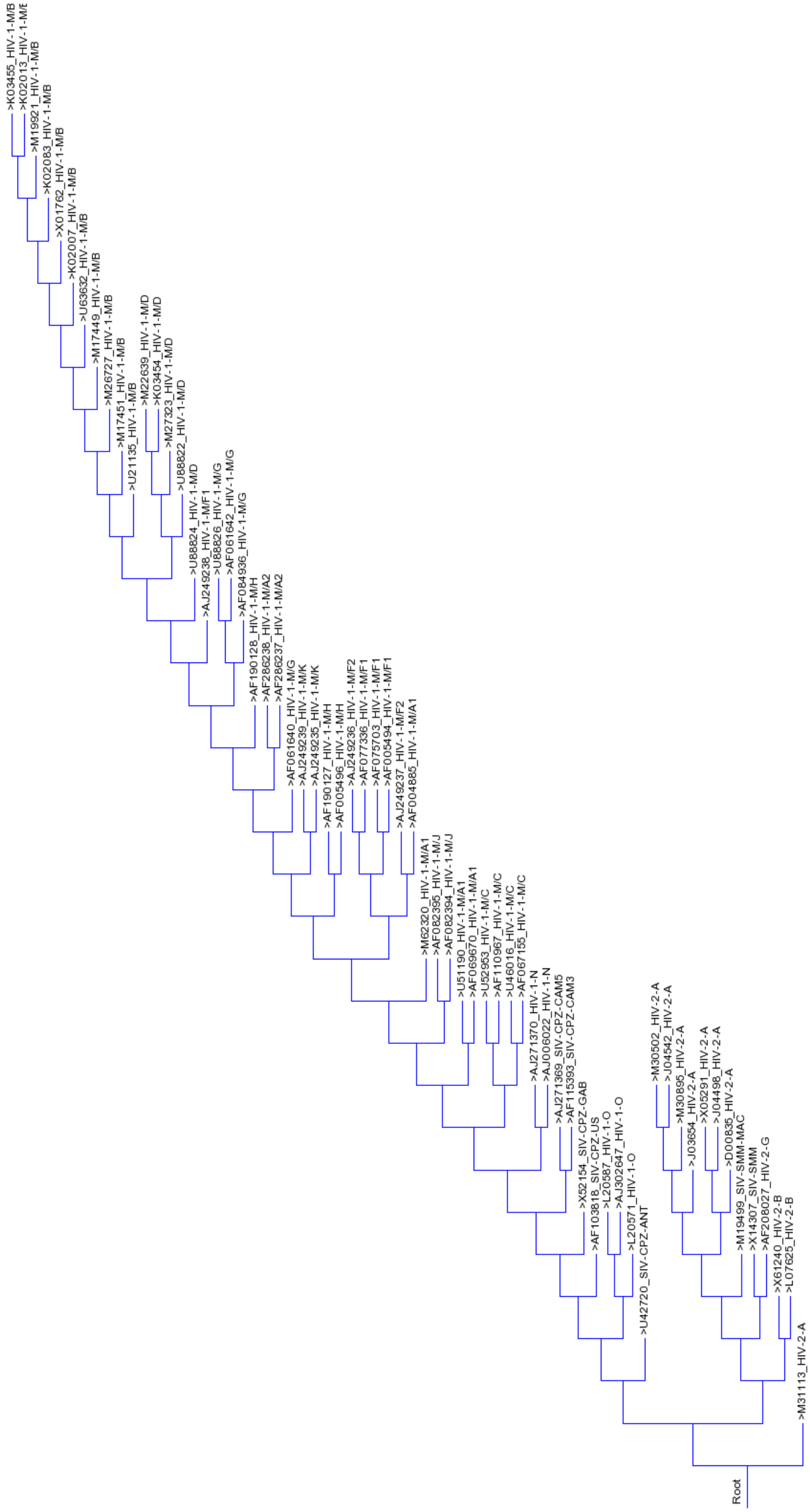


Figure 4: Tree obtained from the dissimilarity matrix of the sequences of the original article, with $N = 15$ - accesible in the repository at `data/Tree visualization/66_tree.png`

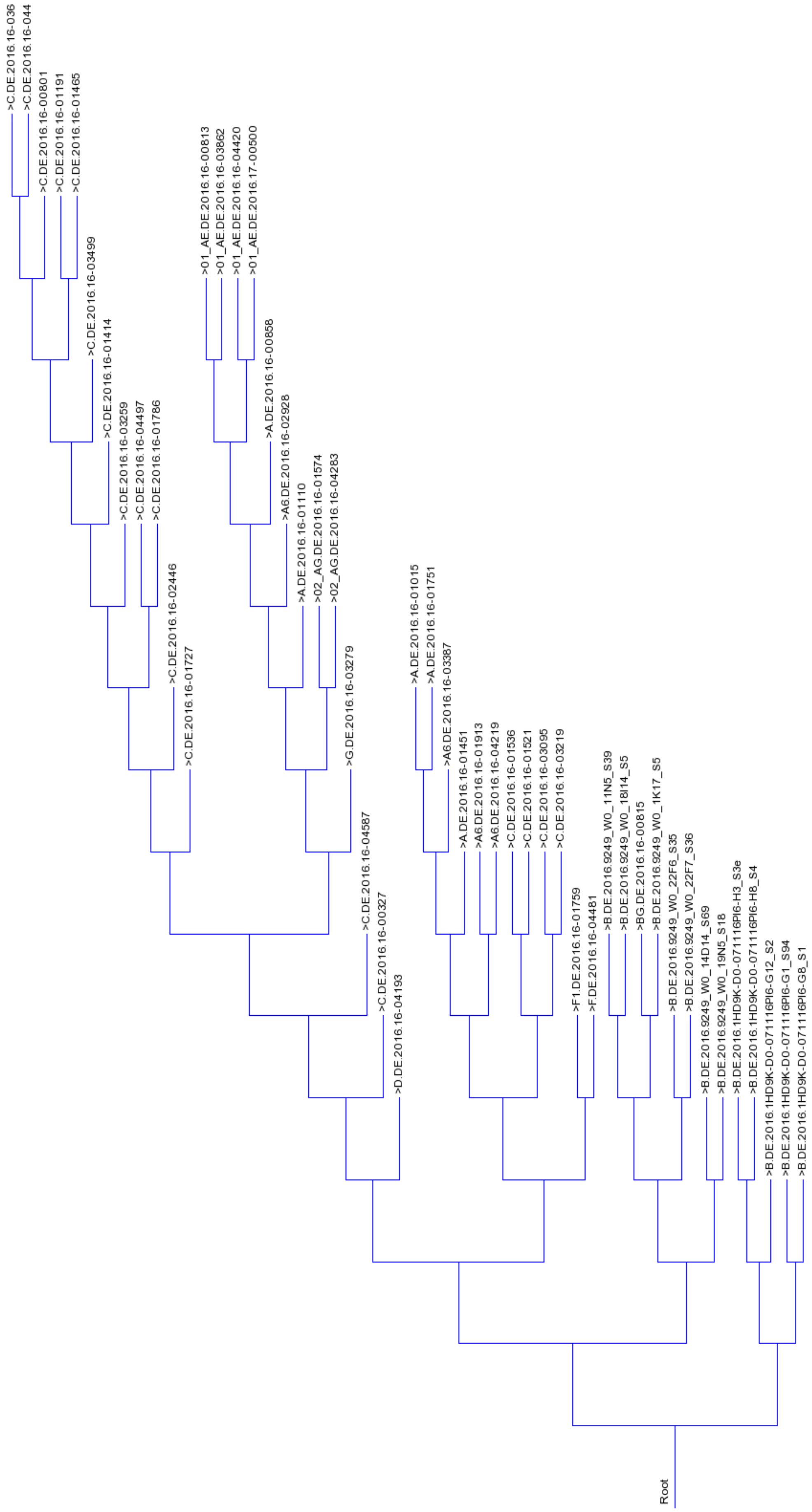


Figure 5: Tree obtained from the dissimilarity matrix of the 50 sequences from Germany, with $N = 15$ - accesible in the repository at `data/Tree visualization/50_tree.png`