ACS OMEGA

http://pubs.acs.org/journal/acsodf

Article

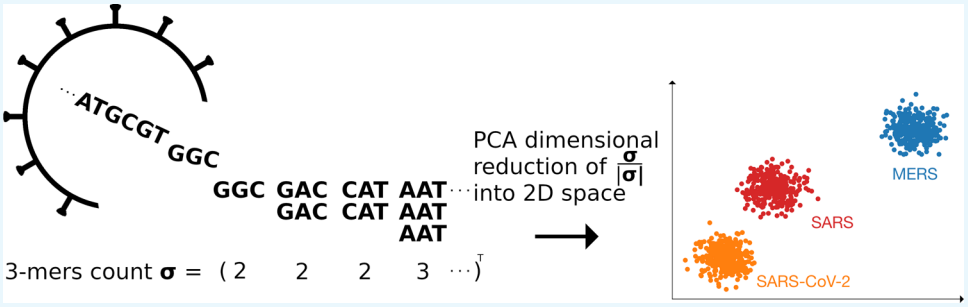
Simple Classification of RNA Sequences of Respiratory-Related Coronaviruses

Louis Oberer, Angel Diaz Carral, and Maria Fyta*

Cite This: *ACS Omega* 2021, 6, 20158−20165

Read Online

ACCESS | Metrics & More | Article Recommendations



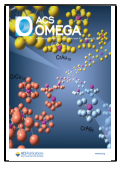
ABSTRACT: A very simple, fast, and efficient approach to analyze and identify respiratory-related virus sequences based on machine learning is proposed. Such schemes are very important in identifying viruses, especially in view of spreading pandemics. The method is based on genetic code rules and the open reading frame (ORF). Data from the respiratory-related coronaviruses are collected and features are extracted based on reoccurring nucleobase 3-tuples in the RNA. Our methodology is simply based on counting nucleobase triplets, normalizing the count to the length of the sequence, and applying principal component analysis (PCA) techniques. The triplet counting can be further used for classification purposes. DNA sequences from the herpes virus family can be considered as the first step towards a complete and accurate classification including more complex factors, such as mutations. The proposed classification scheme is simply based on "counting" biological information. It can serve as the first fast detection method, widely accessible and portable to a variety of distinct architectures for fast and on-the-fly detection. We provide an approach that can be further optimized and combined with supervised techniques to allow for more accurate detection and read out of the exact virus type or sequence. We discuss the relevance of this scheme in identifying differences in similar viruses and their impact on biochemical analysis.

INTRODUCTION

The recently discovered coronavirus SARS-CoV-2 is spreading over the globe with increasing attempts to isolate and stop the spreading.[1−4] Since the identification of this virus, a very large number of genome sequences have been collected.[5] The majority of research studies working with these sequences focus on the development of a drug or vaccine.[6,7] However, the correct identification and categorization of the virus are very important in view of reducing the spread of the disease.[8] Algorithmic approaches for the identification of the SARS-CoV-2 virus have been published showing promising results in correctly identifying SARS-CoV-2 viruses in virus genome data sets.[9] Among the analysis algorithms, the uniform manifold approximation and projection (UMAP) is one of the most frequently used algorithms in bioinformatics and clustering visualization.[10] It has recently also been proven very efficient in clustering SARS-CoV-2 genome isolates,[11] though these methods are computationally complex[12] and are not suited for cheaper and smaller computer architectures such as microcontroller chips. To make the identification of the virus widely accessible and easier, as well as allow fast first identification reducing the complexity of existing algorithms, straightforward and efficient approaches are needed. This could be achieved through a simple theory-based approach that takes the advantage of the biological information hidden in viruses.

The number of substrings of length k or k-mers in a sequence is similar for viruses belonging to the same family.[13] Several techniques are focused on RNA genome substring detection using higher k-mer sizes. Some techniques also rely on natural vectors establishing a very large and detailed space, in which each biological molecule is uniquely represented by a

Received: March 26, 2021
Accepted: July 6, 2021
Published: July 28, 2021



https://doi.org/10.1021/acsomega.1c01625
ACS Omega 2021, 6, 20158−20165

vector.[14] Nevertheless, our method is capable of giving more interpretability of the variation between the frequencies of RNA codons, a problem known as codon bias.[15] Accordingly, using the genetic code rules (3-mers) to build biology-based features is a natural choice. The viral proteins within the human body are encoded based on such virus sequences whose nucleobase 3-mers or triplets, named codons, are translated into amino acids via protein synthesis. The latter are in turn concatenated into a protein. The part of the sequence where the protein information is stored is called an open reading frame (ORF).[16,17] ORFs can be related to overlapping and "hidden" genes in viruses, such as SARS-Cov-2.[18] The ORF is identified by a start codon followed by the protein sequence and stopped by a stop codon. Differences in these ORF regions within a virus family link to the differentiation among the virus types of this family.

SARS-CoV-2 and SARS-type viruses in general show a very large ORF, called ORF1ab,[19,20] which is about 13000 nucleobases in length. ORF1ab includes the structural proteins, which are used to replicate the virus.[21] Based on the genetic code rules and ORFs, we propose a natural very efficient approach to identify latent spaces that encode the whole sequence from SARS viruses into biological features. The Middle East respiratory syndrome-related coronavirus (MERS-CoV), the severe acute respiratory syndrome coronavirus (SARS-CoV), the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), and other viruses of the same family are analyzed to allow for their identification. To this end, we collect the data from the coronavirus family, extract the features based on ORFs and the codon counts, and depict low-dimensional latent spaces to identify well-separated clusters. In order to both reveal the efficiency and further validate and strengthen our proposed approach, we increase the complexity and diversity of the viral RNA data we analyze. To this end, we only begin with SARS-CoV-2, SARS-CoV, and MERS-CoV, then include more members of the coronavirus family, and in the end also include members from other viruses, such as the herpes DNA virus family.

METHODS

Data Collection. A large amount of data has been collected for the RNA sequences of the respiratory-related coronavirus family. The data refer to various viruses and were obtained from the NCBI database[22] and the Covid Predictor Project (CPP),[23] as summarized in Table 1. To ensure that the sequence data did not only contain a protein sequence but also the whole genome, the flag "complete sequences" has been used in the GUI API of the databases. The virus data were stored in the FASTA format, which allows us to store a large amount of genomic data in one file separated by a header line. To classify the data, we refer to the three viruses SARS-CoV, SARS-CoV-2, and MERS-CoV as "SARS/MERS viruses", and for the complete list of the respiratory-related coronaviruses in the table, as the "coronavirus family". For simplicity in the following, we will use the notation "SARS" for SARS-CoV and "MERS" for MERS-CoV. Representative data from the herpes virus family are also listed and will be used in the end for additional validation.

Data Preprocessing. A large number of virus sequences had to be processed for clustering and identification. Reading and processing the data were possible using the BIO python library.[25] The FASTA format was loaded using the SeqIO function. The files used for reading out the data contained more than one sequence. Accordingly, a bulk reading function was used to import the sequences into a list of dictionaries containing the name and sequence of a virus. The list of virus DNA was then stored in a sequence object, which was used in the process of information extraction. The sequence was first scanned for an open reading frame,[26] i.e., the part of the sequence where the protein information is stored. A sliding window traverses the sequence with strides of three (for triplets) to identify the start and stop codons, as depicted in Figure 1. In this figure, the open reading frame (ORF) identification process within a sequence of nucleobases is sketched. Three different frames of reference are translated and emphasized by different colors (green, red, and blue). The first (green) identification frame shows the start codon "ATG" and the stop codon "TAA" at the end of the reading out, which indicates a complete ORF with start and stop codons in contrast to the blue and red frames where they are not present. The labels in the three colors refer to the amino acids that are made up of respective codons. Regarding the variability in the triplet sequences, there are different frames of reference, as a shift of one or two nucleobases in the sequence can lead to different starting points of ORFs.[27] The pair sequence that contains the negative image of the original one is also subject to the protein synthesis and increases the number of possible frames to six. Typically, six different frames are scanned for ORFs.[27] However, three different frames were analyzed for the

Table 1. Types of Viruses, Approximate Length of a Virus Genome Sequence, Date on which the Data were Accessed, Number of Complete Virus Genome Sequences, and Database for all RNA and DNA Data Used in the Analysis

virus type	approx sequence length	date accessed	no. of sequences	database
SARS-CoV	29 751	16 June 2020	340	CPP[24]
SARS-CoV-2	29 903	29 September 2020	22 654	NCBI[22]
SARS-CoV-2 (PCA set)	29 903	21 August 2020	11 118	NCBI
MERS-CoV	30 111	07 July 2020	530	NCBI
bovine coronavirus	31 028	23 September 2020	309	NCBI
camel alpha coronavirus	27 395	23 September 2020	70	NCBI
duck coronavirus	27 754	23 September 2020	425	NCBI
alpha herpes virus	178 101	05 October 2020	195	NCBI
beta herpes virus	236 100	05 October 2020	325	NCBI
gamma herpes virus	172 669	05 October 2020	657	NCBI

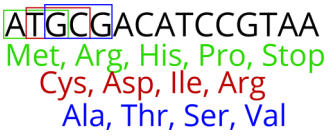


Figure 1. Sketch depicting the open reading frame (ORF) identification process within a sequence of nucleobases (see text for more explanation). The labels in green, red, and blue denote the amino acids ("Met" is methionine, "Cys" is cysteine, etc.) that are made up of respective codons.

https://doi.org/10.1021/acsomega.1c01625
ACS Omega 2021, 6, 20158−20165

positive-strand RNA viruses we study. Such types of viruses can be directly translated via protein synthesis.

Feature Extraction. To extract the feature vectors for viruses of the respiratory-related coronavirus family, the open reading frame ORF1ab scheme was used for relatively long sequences. A large number of other virus types and families show shorter ORFs. Accordingly, the length can also play the role of an identifier for the SARS virus family. To this end, the ORFs used here in the feature extraction have to be at least 11 000 nucleobases in length. We have extracted the relevant features from the data by sliding through a given sequence with strides of three to "cut out" nucleobase triplets, as depicted in Figure 2. For this analysis, the start codon ATG and the stop

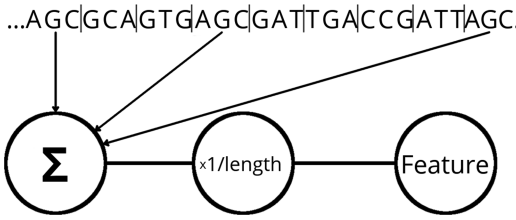


Figure 2. Sketch of the feature extraction scheme. Nucleobase triplets (the codons), shown on the top, are counted through the counter "∑" and normalized over the total length of the sequence to lead to each feature.

codons TAG, TGA, and TAA were used. In this work, these triplet counts are used to define and correspond to the features we further used for our clustering purposes in the following.

The nucleobase triplets in each of the ORF sequences are counted and divided by the length of the sequence to form a feature vector. For the data clustering and analysis purposes, as well as extracting the information of data, the sklearn library was used.[28] Principal component analysis (PCA) was applied to minimize and reduce the dimensionality. The resulting feature vector includes the new features consisting of linear combinations of the counted triplet features. A feature vector was used as an input to the PCA and then projected to a two-dimensional feature space including the variance-minimized features. The latter again corresponds to the triplet counts. The virus data in Table 1 were used to build the PCA matrix using all 64 different triplet combinations. Accordingly, the codon degeneracy of the genetic code, where several codons correspond to the same amino acid, was taken into account. For the SARS-CoV-2 case, we have used the set marked as "PCA set" in the table. Furthermore, the resulting PCA matrix is used to transform incoming feature vectors of different virus types to vectors in the PCA feature space, benefiting from the data similarity. Accordingly, the feature space is the two-dimensional space formed by two feature vectors from the PCA.

Implementation and Optimization. To make our analysis tools easily accessible and portable to different computer architectures, we have chosen to use python. Since, in this case, the operations need more time compared to a code written in C, we have performed the compilations using the cython[29] library to speed up all operations. Interestingly, the main sequence manipulation was carried out with just a sliding window method listed below. This very short piece of code strongly underlines the simplicity of the implementation and can return the feature vector containing the normalized triplet counts. In this short piece of code, collections, split_DNA, and Tuplecomb denote a standard python library, which includes a tool for the counting of list elements, the list containing the DNA sequence cut into nucleobase sequences of length N, and the list with the final normalized count of the respective nucleobase combination of length *N*. ORFs contain a list of dictionaries with the ORF sequences. The resulting feature vector is used as an input to the sklearn PCA function for reducing the dimensionality to a preselected value. The short code used for the main sequence manipulation is given in the following.

```
from collections import Counter
for listing in ORFs:
    for item in listing:
        Split_DNA=[item["ORF"][i:i+N] \
        for i in range(0,len(item["ORF"]),N)]
        Tuplecomb = dict(Counter(Split_DNA))
        for item1 in Tuplecomb:
            Tuplecomb[item1]=Tuplecomb[item1]/len(item["ORF"])
        featurecount.append(Tuplecomb)
```

The feature vectors allow for the further clustering of the features to identify clusters in the viruses and quantify their

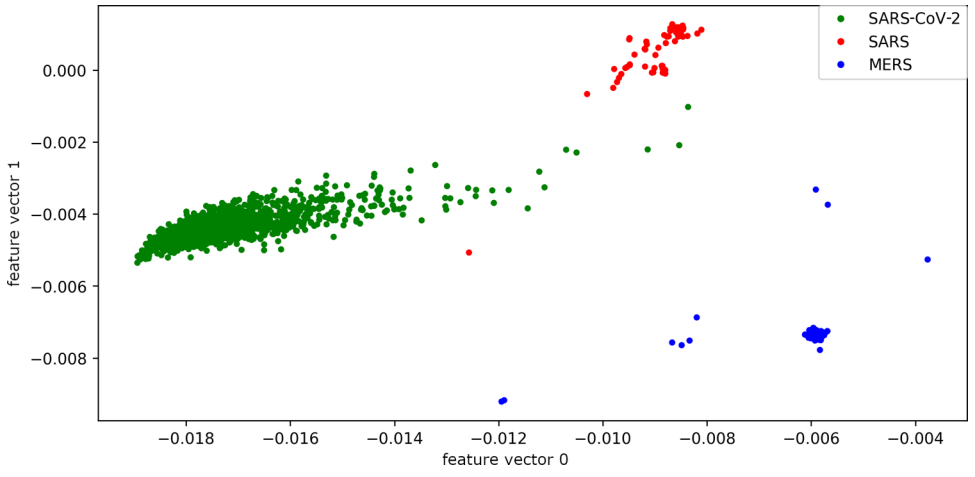


Figure 3. Feature space formed by two feature vectors ("0" and "1") from PCA for the SARS/MERS virus family. The green, red, and blue symbols correspond to the SARS-CoV-2, SARS, and MERS viruses, respectively.

https://doi.org/10.1021/acsomega.1c01625
ACS Omega 2021, 6, 20158−20165

Table 2. Clustering Scores Obtained with DBSCAN (Top) and k-Means (Bottom) for the Set of the SARS/MERS Virus Family Feature Vectors[a]

clusters	SH	CH	DB	S_Dbw	SD	eps value
DBSCAN						
2	0.947	17522.153	1.064	0.810	15.773	0.1400
3	0.977	**212370.191**	0.872	0.619	**11.259**	0.1200
4	0.966	171857.284	**0.602**	0.513	18.466	0.0200
6	0.918	107184.453	0.891	**0.477**	40.449	0.0100
k-means						
2	0.964	69637.592	0.530	2.328	3.683	
3	**0.980**	329053.238	**0.054**	0.303	**2.161**	
4	0.927	567515.868	0.295	0.340	12.550	
5	0.901	666952.415	0.411	0.312	25.373	
6	0.885	**681855.942**	0.478	0.292	36.660	
7	0.853	664477.133	0.559	0.295	69.113	
8	0.851	665182.423	0.602	0.307	60.904	
9	0.853	665133.915	0.545	**0.237**	64.804	

[a]The bold numbers in the first column ("clusters") indicate the expected number of resulting clusters. The bold numbers in the other columns emphasize the best scoring result. The eps value in the last column (top results) denotes the value used with DBSCAN.

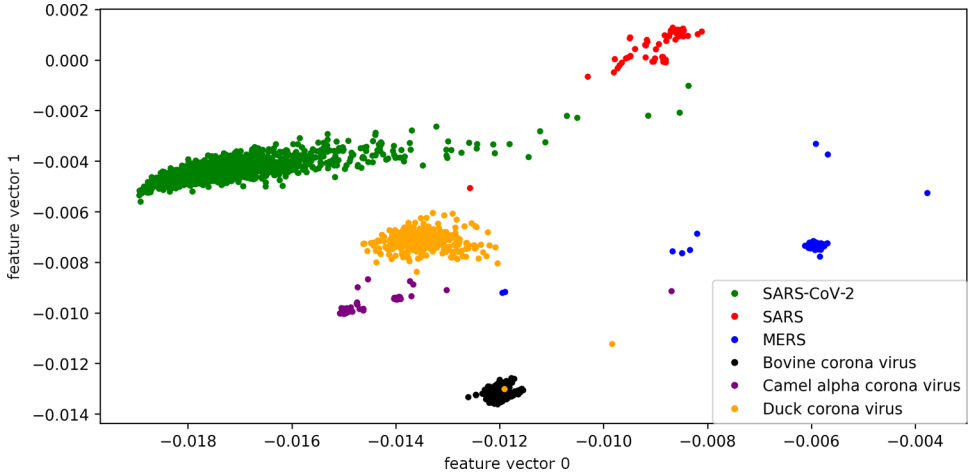


Figure 4. Feature space formed by two feature vectors ("0" and "1") from PCA for the coronavirus family. The colors correspond to the different viruses as denoted by the legend.

separation in the feature space. To this end, the clustering schemes DBSCAN[30] and k-means[31] were used. To identify the optimal number of clusters and check the accuracy of this number, we have used different clustering scores. These include the Silhouette score[32] (SH), the Calinski−Harabasz score[33] (CH), the Davies−Bouldin index[34] (DB), the S_Dbw score,[35] and the SD_score.[36] From this list, the S_dbw typically leads to better cluster identification,[37] which is not confirmed through our analysis. Accordingly, for cases for which small differences or large deviations in the S_dbw scores among the viruses were found, we have decided to use one of the other clustering scores. Note that for the case of SARS-CoV-2, we have used both relevant sets in Table 1 for clustering, though for building the PCA matrix only, the older set was used. Accordingly, we have enriched the data set used for this virus with updated releases possibly also including mutated sequences as will be discussed below. Note that all calculations involved in this work were tested on a Raspberry Pi 4 with a time consumption of ∼30 min for the whole set of coronaviruses, which is very much acceptable for the size of the set used.

RESULTS AND DISCUSSION

The main results are presented next, starting with the feature space analysis for the viruses of the coronavirus family investigated here. Following the methodology outlined above, the respective features have been detected and extracted for ORFs with more than 11 000 nucleobases. The resulting feature spaces show a good separation between different virus families. This is first manifested for the three important members of this family, SARS-CoV-2, SARS, and MERS in Figure 3. The results are represented in the feature space resulting from PCA through two of its respected vectors. All clusters are not only well separated within the feature space but also show a very dense center with a few outliers. Especially, the center of the SARS-CoV-2 cluster is further apart from the other two, denoting a clear separation. The outliers are virus genomes that have been collected after August 21st, 2020 probably already including mutations of the SARS-CoV-2 virus. To exactly measure the separation of the clusters, we have calculated the optimal number of clusters using the clustering algorithms mentioned previously. To achieve this, the set of feature vectors was normalized within the range of [0,1] for all features. This ensures that a direction with larger

https://doi.org/10.1021/acsomega.1c01625
ACS Omega 2021, 6, 20158−20165

Table 3. Clustering Scores Obtained with DBSCAN (Top) and k-Means (Bottom) for the Set of the Coronavirus Family Feature Vectors[a]

cluster	SH	CH	DB	S_Dbw	SD	eps value
DBSCAN						
2	0.899	15961.958	**0.535**	0.750	**5.136**	0.1900
3	0.919	15077.836	1.353	0.559	7.591	0.1500
4	0.948	46796.690	0.811	0.407	13.382	0.0900
5	**0.954**	**162956.426**	0.776	0.401	14.003	0.0800
6	0.951	145818.938	0.781	0.374	14.351	0.0400
8	0.951	119730.571	0.731	**0.261**	33.030	0.0200
12	0.914	62971.981	1.087	0.289	105.356	0.0100
k-means						
2	0.927	50233.283	0.626	1.932	**4.121**	
3	0.940	62878.163	0.727	1.331	5.377	
4	0.954	109480.204	0.395	0.547	4.341	
5	**0.955**	231772.508	**0.205**	0.276	5.931	
6	0.926	386880.318	0.289	0.263	15.574	
7	0.904	453096.583	0.370	0.246	29.796	
8	0.904	543059.277	0.372	**0.221**	30.940	
9	0.893	**601798.301**	0.408	0.230	41.851	

[a]The bold number in the first column ("clusters") indicates the expected number of resulting clusters. The bold numbers in the other columns emphasize the best scoring result. The eps value in the last column (top results) denotes the value at which the DBSCAN clustering was performed.

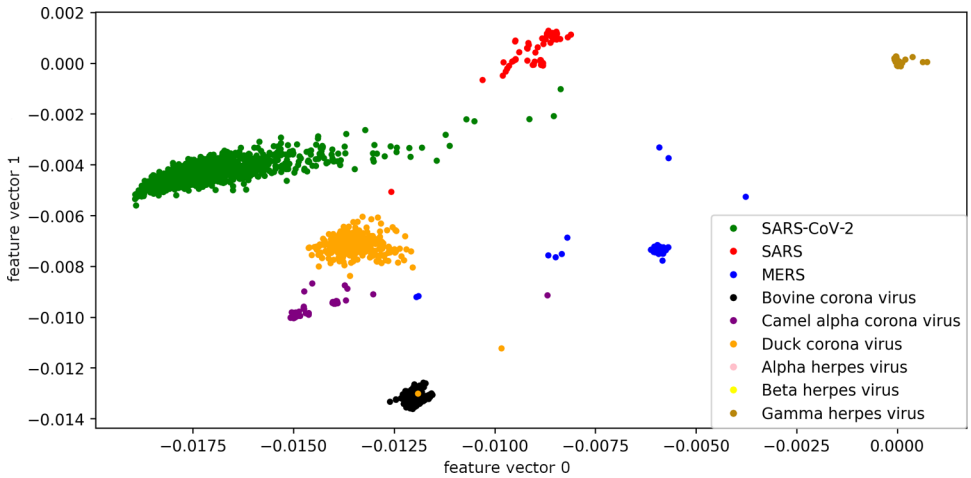


Figure 5. Feature space formed by two feature vectors ("0" and "1") from PCA for the corona and herpes virus families. The colors correspond to the different viruses as denoted by the legend.

absolute values does not dominate during clustering. Interestingly, clustering the viruses of the SARS/MERS family results in a number of predicted clusters, equal to the number of viruses within the clustered data set. The resulting clustering scores obtained using the DBSCAN and k-means clustering algorithms are summarized in Table 2.

With this very promising result of very clear and distinct clustering of the three viruses, SARS, SARS-CoV-2, and MERS, we move on with increasing the complexity and richness of the data sets. In the following, we perform feature extraction and clustering for all viruses of the corona family listed in Table 1. The clusters in the feature space for the viruses of the corona family are depicted in Figure 4. In most of the cases, well-defined clusters are observed having a dense center with outliers further away. In the case of the camel alpha coronavirus, this is not exactly the case, as the respective features are more spread out. There is no exact explanation for this, as many factors, such as mutations over time or mutations due to different geographic positions do play a role. Note that for this virus genome, the outlier distribution is also based on a very small data set in comparison to the larger three data sets of SARS, MERS, and SARS-CoV-2. Overall, due to the shorter sequences of the viruses compared to the SARS ones, the respective clusters are smaller. The clustering scores and data for these clusters and the clustering scores are summarized in Table 3.

We further increase the diversity in the virus data sets by including data from the herpes virus family together with the coronavirus family data used previously. These data refer to the last three entries in Table 1 and are used for validating the efficiency of our approach in distinguishing among viruses and their families. The clustering results in the feature space are depicted in Figure 5. The well-defined clusters of the herpes family are clearly separated from the rest in the PCA feature space. Note that only the gamma herpes virus can be seen in this feature space, as it is the only one of its family with an ORF above 11 000 nucleobases on the positive sense frames of its genome. The results of this clustering analysis show that

https://doi.org/10.1021/acsomega.1c01625
ACS Omega 2021, 6, 20158−20165

more data are required to identify a clear separation of the different coronaviruses in the PCA feature space. However, even with the data at hand, well-defined clusters are observed for most of the cases. Regarding the overall score analysis, the S_Dbw score was revealed to be inefficient in identifying the right amount of clusters for the virus families considered here, while other clustering scores varied around the expected amount of clusters S_Dbw indicated the largest cluster number in most cases. DBSCAN did perform better in this feature space distribution than k-means. This is indicated by the resulting clustering scores that appear to be closer to the expected resulting number of clusters for DBSCAN.

Overall, Figure 5 provides a very intuitive visual separation of the different virus types. No significant mixing of the different virus types in the feature space could be observed. The SARS-CoV-2 virus is spreading from the dense cluster center into a more sparse distribution. A possible reason for this are the different mutations in ORF1ab since the first discovery of the virus (note in Table 1 the access date of the respective data). The MERS virus shows a broad spread in this feature space. This could also be attributed also to mutations and possible other variations of the MERS virus. The bovine and duck coronaviruses showed excellent clustering. On the other hand, the sparsity of the data for the camel alpha coronavirus did not allow us to determine the cluster shape, while it is located close to the duck coronavirus cluster. In the end, the fact that our results do not reveal any overlap in the feature space between the shorter herpes ORFs and the coronavirus family opens a line for research toward a more complete virus classifier. However, to draw a solid conclusion, an extensive scan of other virus families should be performed. This was not the task of the current work, which focused mainly on providing a proof-of-principle study on the concept of the efficient identification of virus clusters.

SUMMARY

In this work, we have analyzed the data from different viruses. We have used nucleobase triplets contained in virus RNA sequences as features for PCA. These features were in turn used to identify cluster formations in the resulting PCA feature space. A very distinct separation among viruses of different families was observed, with most of the clusters having a clear dense central region, though the sparsity of some of the virus data sets did not allow for clear clustering in some cases. The SARS-CoV-2 viruses showed the best clustering (note that these were the richest data sets). The spreading of some features away from the dense cluster in the PCA feature space links to mutations observed in the virus. Other viruses, such as the bovine and duck coronaviruses, revealed very clear clusters without having outliers indicating that no mutations are included. SARS, MERS, and the camel alpha coronaviruses are the ones revealing clusters with more spread-out clusters and outliers. Interestingly, MERS showed a distribution that could be interpreted as splitting up into different subtypes of MERS viruses. The inclusion of herpes viruses, to verify how distinct the virus families are, denoted that there was no overlap between the feature space regions. There were no other candidates found on the databases that could fulfill the requirements to further test these observations.

Despite the fact that we do not focus on finding the best set of biological features for a high-accurate general virus classifier, we were able to discriminate among respiratory-related viruses through a fast and efficient scheme. We could find a clustered feature space solely based on the ORFs and the genetic code and provide open source implementation on portable devices (e.g., Raspberry Pi 4b), which are easily accessible also beyond the scientific community. Our proposed scheme provides a pathway on how to use simple biological information for the first screening of virus types. Another important aspect is mutations on the diagnostic targets, which are continuously being identified and keep increasing in type and number as a pandemic keeps spreading out.[38] The information on the mutations enters directly the ORFs, leading to different clusters. We have attributed many outliers in the clustering of some of the viruses to early mutations in the sequences. Based on this, as mutations increase, these are expected to form separate clusters closer to the initial virus cluster. These should be identified through our analysis as clusters of viruses with certain mutations. Nonetheless, our proposed analysis pipeline and feature extraction scheme using simply the occurrence of nucleobase triplets in ORF1ab was revealed to be highly efficient in detecting and distinguishing among virus types.

Our work has clearly underlined that using the inherently hidden biological information in the ORFs is both essential and a necessary condition in analyzing biological data. In the end, we have proposed a biology-driven analysis scheme that is highly efficient in identifying and distinguishing among viruses. At the same time, this approach provides a technique portable to a variety of distinct architectures, making it widely accessible for fast and on-the-fly detection. To best classify the virus, other descriptors like the virus morphology, the area of occurrence, the symptoms the virus causes, etc. have to be considered. This is not included in the method we propose. Therefore it is a necessary but not a sufficient condition for classification. Here, we claim and propose a very simple scheme based mainly on "counting" biological information, a method that is very efficiently portable to a variety of distinct architectures for fast and on-the-fly detection pointing to new avenues in virus detection. Our proposed classification scheme is simple and efficient and should be considered as a part of a full diagnostic tool. It can be further refined by also including features beyond the ORFs and can be combined with supervised techniques to allow for more accurate detection and read out of the exact virus type or sequence. In this way, the efficiency towards the aim of competing with other more complex detection schemes can be easily enhanced. Based on our analysis, we expect that our approach will remain efficient with respect to other schemes also in the case of very large numbers of available sequences. In such situations, some technical modifications might be necessary, e.g., an online update of the PCA matrix,[39] splitting the data set into smaller ones of well-known mutations, etc. The classification scheme presented here is certainly prone to further refinement based on mutation information and is the first step towards a complete, detailed, and more accurate algorithmic pipeline starting with classification and moving into the direction of an exact virus identification.

AUTHOR INFORMATION

Corresponding Author

Maria Fyta − *Institute for Computational Physics, Universität Stuttgart, 70569 Stuttgart, Germany;* orcid.org/0000-0002-5425-7907; Email: mfyta@icp.uni-stuttgart.de

https://doi.org/10.1021/acsomega.1c01625
ACS Omega 2021, 6, 20158−20165

Authors

Louis Oberer − *Institute for Computational Physics, Universität Stuttgart, 70569 Stuttgart, Germany*

Angel Diaz Carral − *Institute for Computational Physics, Universität Stuttgart, 70569 Stuttgart, Germany*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsomega.1c01625

Notes

The authors declare no competing financial interest.

The software used for the analysis in this work can be downloaded from https://github.com/LouisOb/VirusPredictor.

■ ACKNOWLEDGMENTS

This work was supported by the EXC 2075 SimTech Cluster of the University of Stuttgart funded by the German Funding Agency (DFG).

■ REFERENCES

(1) Hellewell, J.; Abbott, S.; Gimma, A.; Bosse, N. I.; Jarvis, C. I.; Russell, T. W.; Munday, J. D.; Kucharski, A. J.; Edmunds, W. J.; Sun, F.; et al. Feasibility of controlling COVID-19 outbreaks by isolation of cases and contacts. *Lancet Global Health* **2020**, *8*, e488−e496.

(2) Bedford, J.; Enria, D.; Giesecke, J.; Heymann, D. L.; Ihekweazu, C.; Kobinger, G.; Lane, H. C.; Memish, Z.; don Oh, M.; Sall, A. A.; et al. COVID-19: towards controlling of a pandemic. *Lancet* **2020**, *395*, 1015−1018.

(3) Hopman, J.; Allegranzi, B.; Mehtar, S. Managing COVID-19 in Low- and Middle-Income Countries. *JAMA* **2020**, *323*, 1549−1550.

(4) Yatoo, M. I.; Hamid, Z.; Parray, O. R.; Wani, A. H.; Haq, A. U.; Saxena, A.; Patel, S. K.; Pathak, M.; Tiwari, R.; Malik, Y. S.; et al. COVID-19 - Recent advancements in identifying novel vaccine candidates and current status of upcoming SARS-CoV-2 vaccines. *Hum. Vaccines Immunother.* **2020**, *16*, 2891−2904.

(5) Brister, J. R.; Ako-adjei, D.; Bao, Y.; Blinkova, O. NCBI Viral Genomes Resource. *Nucleic Acids Res.* **2015**, *43*, D571−D577.

(6) Dong, L.; Hu, S.; Gao, J. Discovering drugs to treat coronavirus disease 2019 (COVID-19). *Drug Discoveries Ther.* **2020**, *14*, 58−60.

(7) Rome, B. N.; Avorn, J. Drug Evaluation during the Covid-19 Pandemic. *N. Engl. J. Med.* **2020**, *382*, 2282−2284.

(8) Udugama, B.; Kadhiresan, P.; Kozlowski, H. N.; Malekjahani, A.; Osborne, M.; Li, V. Y. C.; Chen, H.; Mubareka, S.; Gubbay, J. B.; Chan, W. C. W. Diagnosing COVID-19: The Disease and Tools for Detection. *ACS Nano* **2020**, *14*, 3822−3835.

(9) Lopez-Rincon, A.; Tonda, A.; Mendoza-Maldonado, L.; Mulders, D. G. J. C.; Molenkamp, R.; Perez-Romero, C. A.; Claassen, E.; Garssen, J.; Kraneveld, A. D. Classification and specific primer design for accurate detection of SARS-CoV-2 using deep learning. *Sci. Rep.* **2021**, *11*, No. 947.

(10) McInnes, L.; Healy, J.; Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, 2018. arXiv:1802.03426v3, https://arxiv.org/abs/1802.03426v3.

(11) Hozumi, Y.; Wang, R.; Yin, C.; Wei, G.-W. UMAP-assisted K-means clustering of large-scale SARS-CoV-2 mutation datasets. *Comput. Biol. Med.* **2021**, *131*, No. 104264.

(12) D'Angelo, G.; Palmieri, F. Discovering genomic patterns in SARS-CoV-2 variants. *Int. J. Intell. Syst.* **2020**, *35*, 1680−1698.

(13) Yao, T.; Zheng, J. Visualizations of Multiple Probability Measures for SARS-CoV-2 Genomes, 2020. DOI: 10.21203/rs.3.rs-74631/v1+.

(14) Wang, Y.; Tian, K.; Yau, S. S.-T. Protein sequence classification using natural vector and convex Hull method. *J. Comput. Biol.* **2019**, *26*, 315−321.

(15) Hershberg, R.; Petrov, D. Selection on Codon Bias. *Annu. Rev. Genet.* **2008**, *42*, 287−299.

(16) Brown, T. A. *Genomes*, 2nd ed.; Wiley-Liss: Oxford, 2002.

(17) Sieber, P.; Platzer, M.; Schuster, S. The Definition of Open Reading Frame Revisited. *Trends Genet.* **2018**, *34*, 167−170.

(18) Nelson, C. W.; Ardern, Z.; Goldberg, T. L.; Meng, C.; Kuo, C.-H.; Ludwig, C.; Kolokotronis, S.-O.; Wei, X. Dynamically evolving novel overlapping gene as a factor in the SARS-CoV-2 pandemic. *eLife* **2020**, *9*, No. e59633.

(19) van der Meer, Y.; van Tol, H.; Locker, J. K.; Snijder, E. J. ORF1a-encoded replicase subunits are involved in the membrane association of the arterivirus replication complex. *J. Virol.* **1998**, *72*, 6689−6698.

(20) Méndez, E.; Salas-Ocampo, M. E.; Munguía, M. E.; Arias, C. F. Protein products of the open reading frames encoding nonstructural proteins of human astrovirus serotype 8. *J. Virol.* **2003**, *77*, 11378−11384.

(21) Graham, R. L.; Sparks, J. S.; Eckerle, L. D.; Sims, A. C.; Denison, M. R. SARS coronavirus replicase proteins in pathogenesis. *Virus Res.* **2008**, *133*, 88−100.

(22) NCBI, National Center for Biotechnology Information. https://www.ncbi.nlm.nih.gov/genome/viruses/ (accessed October 13, 2020).

(23) Sarkar, J. P.; Saha, I.; Seal, A.; Maity, D. COVID-Predictor: RNA Sequence based Prediction of Coronavirus, 2020. DOI: 10.21203/rs.3.rs-23913/v1.

(24) CovidPredictor, COVID-Predictor: Machine Learning to Predict Novel Coronavirus from Other Pathogenic Viruses. http://www.nitttrkol.ac.in/indrajit/projects/COVID-Predictor/ (accessed June 16, 2020).

(25) Cock, P. J. A.; Antao, T.; Chang, J. T.; Chapman, B. A.; Cox, C. J.; Dalke, A.; Friedberg, I.; Hamelryck, T.; Kauff, F.; Wilczynski, B.; et al. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **2009**, *25*, 1422−1423.

(26) Claverie, J.-M. Computational Methods for the Identification of Genes in Vertebrate Genomic Sequences. *Hum. Mol. Genet.* **1997**, *6*, 1735−1744.

(27) Rombel, I. T.; Sykes, K. F.; Rayner, S.; Johnston, S. A. ORF-FINDER: a vector for high-throughput gene identification. *Gene* **2002**, *282*, 33−41.

(28) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825−2830.

(29) Behnel, S.; Bradshaw, R.; Citro, C.; Dalcin, L.; Seljebotn, D. S.; Smith, K. Cython: The best of both worlds. *Comput. Sci. Eng.* **2011**, *13*, 31−39.

(30) Hahsler, M.; Piekenbrock, M.; Doran, D. dbscan: Fast Density-Based Clustering with R. *J. Stat. Software* **2019**, *91*, 1−30.

(31) Lloyd, S. P. Least squares quantization in PCM. *IEEE Trans. Inf. Theory* **1982**, *28*, 129−136.

(32) Rousseeuw, P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **1987**, *20*, 53−65.

(33) Cengizler, C.; Kerem Un, M. Evaluation of Calinski-Harabasz Criterion as Fitness Measure for Genetic Algorithm Based Segmentation of Cervical Cell Nuclei. *J. Adv. Math. Comput. Sci.* **2017**, *22*, 1−13.

(34) Davies, D. L.; Bouldin, D. W. A Cluster Separation Measure. *IEEE Trans. Pattern Anal. Mach. Intell.* **1979**, *PAMI-1*, 224−227.

(35) Halkidi, M.; Vazirgiannis, M. In *Clustering Validity Assessment: Finding the Optimal Partitioning of a Data Set*, IEEE International Conference on Data Mining, ICDM, 2001; pp 187−194.

(36) Halkidi, M.; Vazirgiannis, M.; Batistakis, Y. Quality Scheme Assessment in the Clustering Process. In *Lecture Notes in Computer Science*; Springer: Berlin, 2000; Vol. *1910*, pp 265−276.

(37) Liu, Y.; Li, Z.; Xiong, H.; Gao, X.; Wu, J. In *Understanding of Internal Clustering Validation Measures*, IEEE International Conference on Data Mining, 2011.

(38) Wang, R.; Hozumi, Y.; Yin, C.; Wei, G.-W. Mutations on COVID-19 diagnostic targets. *Genomics* **2020**, *112*, 5204−5213.

(39) Cardot, H.; Degras, D. Online principal component analysis in high dimension: Which algorithm to choose. *Int. Stat. Rev.* **2018**, *86*, 29−50.

https://doi.org/10.1021/acsomega.1c01625
ACS Omega 2021, 6, 20158−20165