Virus2Vec: Viral Sequence Classification Using Machine Learning

Sarwan Ali, Babatunde Bello, Prakash Chourasia, Ria Thazhe Punathil, Pin-Yu Chen, Imdad Ullah Khan and Dr. Murray Patterson

Georgia State University
August 1, 2023

Table of Contents

Introduction

Sequence data analysis :

- Studies of Alterations in the protein sequence to classify and predict amino acid changes in SARS-CoV-2 are crucial in
  - Understanding the immune invasion and host-to-host transmission properties of SARS-CoV-2 and its variants
  - Identifying transmission patterns of each variant may help policymakers to prevent the rapid spread
  - May help in vaccine design and efficacy
- Unravel the mysteries of genetic info & its functional implications
- Phylogenetic tree construction-based methods - a Traditional way to trace evolution.
- Later Machine Learning and Deep Learning played major role.

Motivation

- In-depth studies of alterations in the protein sequence to classify and predict amino acid changes in SARS-CoV-2 are crucial in
  - Understanding the immune invasion and host-to-host transmission properties of SARS-CoV-2 and its variants
  - Knowledge of mutations and variants will help identify transmission patterns - facilitate public health measures
  - This will also help in vaccine design and efficacy
- Understanding biological sequence classification can unravel the mysteries of genetic information and its functional implications.
- Improve performance and reduce computational cost.
- Insights into the evolutionary relationships between organisms, helping us understand the origins and diversity of life on Earth.
- Advancements in personalized medicine, identifying genetic variants associated with diseases and predict patient responses to treatments.

Real World Application

- Genomic surveillance: Tracking the spread of pathogens in terms of genomic content
- Real time identification of new and rapidly emerging coronavirus variants
- Track the spread of known coronavirus variants in new municipalities, regions, countries and continents

Challenges

- Mutations happen disproportionately in different regions of genome
- Since new variants (for coronavirus) are emerging, not much information is available about these variants
- Generating fixed-length feature vectors from variable-length sequences
- High dimensionality of generated embeddings (e.g. OHE)
- Challenges:
  - The computation time
  - The memory usage (storing an n x n matrix)
  - The usage of kernel matrices limited to kernel-based ML methods (difficult to generalize on non-kernel classifiers)

Kernel Method

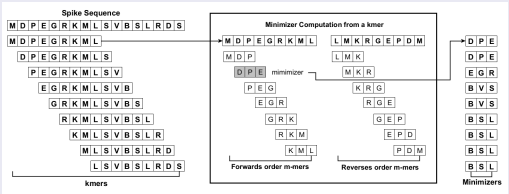
- A method that allows us to apply linear classifiers to non-linear problems by mapping non-linear data into a higher-dimensional space
- Kernel-based methods (e.g., SVM) are proven useful for several machine learning (ML) tasks such as sequence classification
- There are three challenges involved with kernel methods in general:
  - Kernel computation (requires exponential complexity to compute dot product)
  - scalability (storing n x n matrix in memory is not possible when n, the number of data points, is too large)
  - The usage of kernel matrices limited to kernel-based ML methods (difficult to generalize on non-kernel classifiers)
- The computational complexity problem can be solved using approximate methods
- The scalability issue remains for the typical kernel methods in general
- For non-kernel classifiers, we can use kernel PCA (could result in loss of information or computationally expensive)

Methodology

- Virus2Vec is a compact alignment-free embedding approach
- Eliminates the need for the sequence alignment
- Uses a fraction of the information as compared to a more traditional k-mers-based approach.
- Optimizes and reduces efforts in counting k-mers, which can be an expensive — and redundant — task.
- The process involves :
  - Compute minimizer using sliding window on k-mer
  - The lexicographically smallest is selected as the minimizer for that k-mer
  - For each minimizer, we compute a weight using the "Position Weight Matrix" (PWM) method.
  - We use the score of each m-mer (computed using the PWM-based approach) to the corresponding bin to get the final feature vector representation.

Feature Vector Representation

- To convert the sequences into fixed-length numerical representations, we use a recently proposed method called Spike2Vec [1].
- Spike2Vec generates a fixed-length numerical representation using the concept of k-mers (also called n-gram) for a sequence.

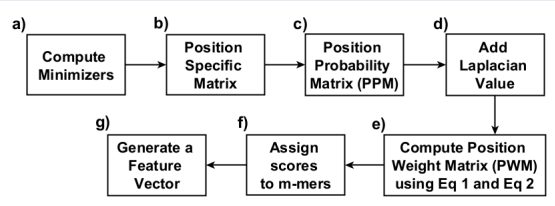


- Uses sliding window to generate k-mers of length k (window size).
- For a set of k-mers in a sequence, the feature vector of length $|\Sigma|^k$ (Σ is the set of alphabets "amino acids" or nucleotides), is generated using their count.

Feature Vector Representation

- To compute the minimizer, a sliding window is again used but this time on k-mer in both directions (forward and reverse).
- Lexicographically smallest is selected as the minimizer for that k-mer.
- Minimizers ignore many amino acids in each k-mer, only preserving a fraction of the m-mers, for which binning of these m-mers becomes much more efficient.
- For each minimizer, we compute a weight using the "Position Weight Matrix" (PWM) method (Explained Later).

Virus2Vec Workflow



- After computing the Minimizers, a Position Frequency Matrix (PFM) is generated which contains the frequency count for each character at each position.
- We have 20 unique amino acids in the spike protein sequence dataset, our PFMs have 20 rows and $m = 3$ columns
- Whereas for rabies data we have 4 unique nucleotide; our PFMs have 4 rows, and $m = 3$ columns.

Virus2Vec Workflow

- Normalize the PFM matrix to create a Position Probability Matrix (PPM) containing the probability of each amino acid at each position
- A position weight matrix (PWM)is then computed from the adjusted probability matrix, by computing the log-likelihood of each amino acid character c, i.e., $c \in A, C, \ldots, Y$ for spike sequences or $c \in A, C, G, T$ for rabies virus sequences.
- PWM is used to compute the absolute scores for each individual minimizer generated from the sequence. It is the sum of the score of bases for the index.
- After getting the score for each m-mer, we generate a vector of length $|\Sigma|^m$. Using the score of each m-mer (computed using the PWM-based approach) to the corresponding bin to get the final feature vector representation.

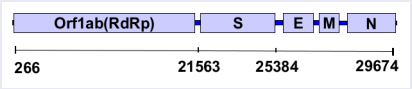
Dataset

- Spike Sequence from the SARS-CoV-2 virus
- Rabies sequences data

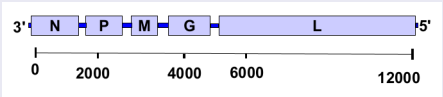
Name	Type	Source	Sequence Count	Classes	Sequence Length			
					Min	Max	Avg	Mode
Coronavirus Host Data	Spike protein sequences for COVID-19 hosts	GISAID, ViPR	5558	22	9	1584	1272.4	1273
Rabies Virus Data	Nucleotide genome sequences for rabies virus hosts	RABV-GLUE	20051	12	90	11930	1948.4	1353

Table: Data Statistics.

Dataset - Spike Sequences Structure



- The SARS-CoV-2 genome, of roughly 30K bps in length
- The structural protein further consists of the spike (or S) protein along with Envelope (E), Membrane (M) and Nucleocapsid (N) proteins.

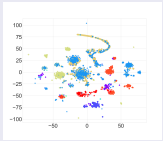


- The rabies genome is 12kb in length and encodes five proteins Nucleoprotein (N), Phosphoprotein (P), Matrix Protein (M), Glycoprotein (G), and Polymerase (L).

Properties of Different Embedding Methods

Embedding	Alignment Free	Low Dim Vectors	Vector (Spike/Rabies)	Space Efficient	Runtime Efficient	Details
One-Hot Encoding	✗	✗	69960 / 5600	✗	✗	length of OHE for a spike sequence 3498
Spike2Vec	✓	✓	8000 / 125	✓	✗	$\Sigma = 20$ and $k = 3$ (for Spike data)
Approx. Kernel	✓	✓	500 / 500	✗	✗	Dimensionality depends on Num of sequences
PWM2Vec	✗	✓	3490 / 125	✓	✓	Length of Spike Seq after alignment 3498 and $k = 9$
LSTM	✓	_	_	✗	✗	End-to-End DL architectures
GRU	✓	_	_	✗	✗	
CNN	✓	_	_	✗	✗	
ProteinBert	✓	_	_	✗	✗	Pretrained Protein language model using Transformer
MFV	✓	✓	8000 / 125	✓	✓	$k = 9$ and $m = 3$
Virus2Vec (ours)	✓	✓	8000 / 125	✓	✓	Proposed method $m = 3$

tSne plots - Host Data

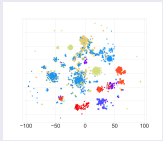


(a) Spike2Vec

(b) Appr. Kernel

(c) MFV

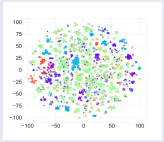
(d) PWM2Vec



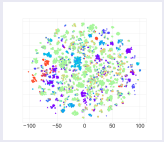
(e) Virus2Vec

- t-SNE plots for **Coronavirus Host** dataset.
- In all of them Environment & Human displays unambiguous grouping.
- Virus2Vec is able to preserve the structure of data in the same way as with the other existing embedding methods.

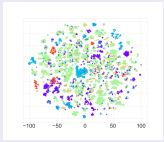
tSne plots - Rabies Data



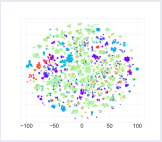
(a) Spike2Vec



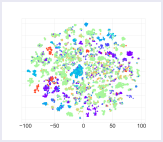
(b) Appr. Kernel



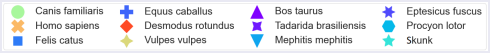
(c) MFV



(d) PWM2Vec



(e) Virus2Vec



- t-SNE plots for **Rabiesn Virus** dataset.
- Virus2Vec does not disturb the structure and even provides better clusters as compared to baseline embeddings.

Results

Method	Classifier	Host Spike Sequences							Rabies Virus						
		Acc. ↑	Prec. ↑	Recall ↑	F1 (Weig.) ↑	F1 (Macro) ↑	ROC AUC ↑	Train Time (sec.) ↓	Acc. ↑	Prec. ↑	Recall ↑	F1 (Weig.) ↑	F1 (Macro) ↑	ROC AUC ↑	Train Time (sec.) ↓
Spike2Vec	SVM	0.84	0.84	0.84	0.83	0.77	0.87	45.36	0.72	0.70	0.72	0.69	0.58	0.76	22.76
	NB	0.69	0.77	0.69	0.67	0.58	0.79	6.02	0.06	0.29	0.06	0.03	0.03	0.52	0.40
	MLP	0.81	0.83	0.81	0.81	0.63	0.83	46.14	0.58	0.45	0.58	0.48	0.23	0.60	1.46
	KNN	0.80	0.81	0.80	0.79	0.59	0.79	1.97	0.75	0.73	0.75	0.74	0.62	0.79	1.07
	RF	0.84	0.85	0.84	0.84	0.73	0.85	10.21	0.78	0.76	0.78	0.76	0.67	0.81	0.88
	LR	0.84	0.85	0.84	0.84	0.76	0.87	31.00	0.71	0.67	0.71	0.67	0.55	0.75	1.14
	DT	0.82	0.83	0.82	0.82	0.71	0.85	2.54	0.68	0.68	0.68	0.68	0.57	0.77	0.21
Approx. Kernel	SVM	0.79	0.80	0.79	0.77	0.57	0.78	18.18	0.73	0.72	0.73	0.71	0.59	0.76	244.82
	NB	0.60	0.66	0.60	0.57	0.51	0.73	**0.07**	0.14	0.51	0.14	0.13	0.20	0.60	0.33
	MLP	0.79	0.78	0.79	0.78	0.59	0.75	7.69	0.77	0.77	0.77	0.76	0.63	0.79	119.56
	KNN	0.86	0.85	0.86	0.86	0.60	0.76	0.21	0.83	0.82	0.83	0.82	0.69	0.83	5.57
	RF	0.82	0.82	0.82	0.81	0.67	0.78	1.80	0.83	0.83	0.83	0.82	0.71	0.83	22.17
	LR	0.76	0.77	0.76	0.74	0.64	0.76	2.36	0.66	0.64	0.66	0.64	0.55	0.73	80.32
	DT	0.78	0.78	0.78	0.77	0.55	0.75	0.24	0.76	0.76	0.76	0.76	0.65	0.80	4.44
Neural Network	LSTM	0.32	0.10	0.32	0.15	0.02	0.50	21634.34	0.49	0.38	0.49	0.36	0.15	0.49	35026.49
	CNN	0.44	0.10	0.11	0.08	0.07	0.53	17856.40	0.73	0.74	0.73	0.72	0.64	0.80	8164.93
	GRU	0.32	0.13	0.32	0.16	0.03	0.50	126585.0	0.59	0.54	0.59	0.51	0.28	0.60	16180.78
Spaced k-mer	SVM	0.81	0.82	0.81	0.81	0.89	0.92	3.12	0.80	0.80	0.80	0.79	0.67	0.81	140.67
	NB	0.66	0.69	0.66	0.66	0.61	0.78	0.03	0.28	0.56	0.28	0.27	0.35	0.69	0.11
	MLP	0.82	0.82	0.82	0.82	0.77	0.87	41.66	0.79	0.78	0.79	0.79	0.66	0.81	84.70
	KNN	0.79	0.80	0.79	0.80	0.77	0.87	0.40	0.83	0.82	0.83	0.82	0.71	0.84	2.54
	RF	0.84	0.85	0.84	0.84	**0.91**	**0.94**	2.84	0.84	**0.84**	0.84	0.83	0.72	0.84	24.28
	LR	0.82	0.83	0.82	0.82	0.89	0.93	2.31	0.79	0.78	0.79	0.78	0.66	0.81	14.08
	DT	0.80	0.80	0.80	0.80	0.85	0.93	0.64	0.76	0.76	0.76	0.76	0.65	0.81	6.36
Protein Bert	-	0.79	0.80	0.79	0.78	0.71	0.84	15742.95	0.79	0.78	0.79	0.76	0.64	0.80	35742.84
MFV	SVM	0.83	0.83	0.83	0.82	0.73	0.85	35.71	0.66	0.61	0.66	0.61	0.48	0.71	241.11
	NB	0.63	0.75	0.63	0.63	0.49	0.72	5.80	0.06	0.34	0.06	0.05	0.08	0.54	0.41
	MLP	0.82	0.82	0.82	0.82	0.66	0.81	53.82	0.61	0.54	0.61	0.56	0.33	0.65	2.17
	KNN	0.79	0.80	0.79	0.78	0.63	0.81	1.60	0.74	0.72	0.74	0.72	0.61	0.79	1.12
	RF	0.84	0.85	0.84	0.84	0.74	0.85	10.79	0.78	0.77	0.78	0.76	0.66	0.80	0.81
	LR	0.83	0.84	0.83	0.83	0.74	0.85	9.24	0.59	0.55	0.59	0.54	0.36	0.64	0.70
	DT	0.83	0.83	0.83	0.82	0.74	0.85	1.15	0.69	0.68	0.69	0.69	0.58	0.77	0.19
PSWM2Vec	SVM	0.81	0.82	0.81	0.80	0.80	0.90	3.46	0.48	0.28	0.48	0.33	0.08	0.48	1.10
	NB	0.58	0.66	0.58	0.57	0.53	0.78	0.25	0.27	0.32	0.27	0.26	0.16	0.27	0.18
	MLP	0.82	0.82	0.82	0.81	0.72	0.87	8.44	0.57	0.50	0.57	0.50	0.33	0.57	2.33
	KNN	0.81	0.80	0.81	0.80	0.70	0.86	1.22	0.64	0.62	0.64	0.62	0.50	0.64	0.49
	RF	0.85	0.85	0.85	0.84	0.83	0.91	1.26	0.66	0.65	0.66	0.65	0.53	0.66	0.79
	LR	0.79	0.80	0.79	0.77	0.70	0.84	1.45	0.48	0.31	0.48	0.34	0.10	0.48	1.41
	DT	0.80	0.81	0.80	0.80	0.73	0.88	**0.23**	0.58	0.59	0.58	0.58	0.47	0.58	0.17
Virus2Vec	SVM	0.85	0.86	0.85	0.85	0.87	0.932	151.5	0.66	0.62	0.66	0.62	0.50	0.72	15931.90
	NB	0.67	0.78	0.67	0.65	0.65	0.83	5.67	0.07	0.34	0.07	0.05	0.10	0.55	**0.17**
	MLP	0.85	0.85	0.85	0.84	0.79	0.90	47.30	0.71	0.69	0.71	0.68	0.56	0.75	11.76
	KNN	0.84	0.85	0.84	0.83	0.76	0.88	78.79	0.71	0.73	0.74	0.71	0.59	0.78	8.54
	RF	0.86	0.86	0.86	0.85	0.84	0.91	13.36	**0.84**	0.83	**0.84**	**0.83**	**0.74**	**0.85**	3.13
	LR	**0.87**	**0.87**	**0.87**	**0.87**	0.88	0.93	8.29	0.59	0.54	0.59	0.53	0.34	0.63	13.94
	DT	0.81	0.82	0.81	0.81	0.76	0.88	2.49	0.77	0.77	0.77	0.77	0.68	0.82	0.55

Results

- Virus2Vec outperforms the SOTA methods
- The runtime to generate the embeddings makes it a huge factor in considering Virus2Vec over other embeddings.
- Virus2Vec outperforms not only the feature engineering-based baselines but also the neural network-based classifiers.
- The findings are reinforced by its visualization counterpart as well, as we saw in t-SNE plots also for Virus2Vec, it does not disrupt the general structure of the data because t-SNE is able to retain the structure of the data.

Results

Method	Coronavirus data Runtime ↓	Rabies virus data Runtime ↓
OHE	196.31 Sec.	**44.17** Sec.
Spike2Vec	1179.66 Sec.	259.86 Sec.
PWM2Vec	1506.63 Sec.	412.254 Sec.
Approx. Kernel	379.47 Sec.	179.47 Sec.
Virus2Vec	**90.65** Sec.	105.78 Sec.

Table: Runtime for generating feature vectors using different embedding methods for Coronavirus-Host data and Rabies Virus-Host dataset.

- Virus2Vec takes the least time to generate embeddings
- It takes 4 times less as compared to the Approximate Kernel method and 15 times less than PWM2Vec, which are comparable when accuracy is considered.

Conclusion and Future Work

- We propose an efficient sequence embedding approach Virus2Vec
- Uses an alignment-free method based on minimizers and PWM to classify genomic sequences.
- Virus2Vec not only performs better but is also an alignment-free approach.
- We show Virus2Vec comparable predictive performance and better runtimes.

Future Work

- Try on larger data to evaluate the scalability of Virus2Vec.
- Such an approach could also work even on *unassembled* (short read) data (not just unaligned), in a similar way that it works for metagenomics.

Thank You

Questions!!


S. Ali and M. Patterson, "Spike2vec: An efficient and scalable embedding approach for covid-19 spike sequences," in *IEEE International Conference on Big Data*, 2021, pp. 1533–1540.