DeepViFi: Detecting Oncoviral Infections in Cancer Genomes using Transformers

Utkrisht Rajkumar*
Sara Javadzadeh*
urajkuma@eng.ucsd.edu
Department of Computer Science & Engineering, UC San Diego
La Jolla, California, USA

Mihir Bafna
Georgia Institute of Technology
Atlanta, Georgia, USA

Dongxia Wu
Department of Computer Science & Engineering, UC San Diego
La Jolla, California, USA

Rose Yu
Department of Computer Science & Engineering, UC San Diego
La Jolla, California, USA

Jingbo Shang
Department of Computer Science & Engineering, UC San Diego
La Jolla, California, USA

Vineet Bafna
Department of Computer Science & Engineering, UC San Diego
La Jolla, California, USA
vbafna@cs.ucsd.edu

ABSTRACT

We consider the problem of identifying viral reads in human host genome data. We pose the problem as open-set classification as reads can originate from unknown sources such as bacterial and fungal genomes. Sequence-matching methods have low sensitivity in recognizing viral reads when the viral family is highly diverged. Hidden Markov models have higher sensitivity but require domain-specific training and are difficult to repurpose for identifying different viral families. Supervised learning methods can be trained with little domain-specific knowledge but have reduced sensitivity in open-set scenarios. We present DeepViFi, a transformer-based pipeline, to detect viral reads in short-read whole genome sequence data. At 90% precision, DeepViFi achieves 90% recall compared to 15% for other deep learning methods. DeepViFi provides a semi-supervised framework to learn representations of viral families without domain-specific knowledge, and rapidly and accurately identify target sequences in open-set settings.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; **Natural language processing**; • **Applied computing** → **Bioinformatics**.

KEYWORDS

natural language processing, neural networks, open-set classification, viral detection

ACM Reference Format:
Utkrisht Rajkumar, Sara Javadzadeh, Mihir Bafna, Dongxia Wu, Rose Yu, Jingbo Shang, and Vineet Bafna. 2022. DeepViFi: Detecting Oncoviral Infections in Cancer Genomes using Transformers. In *13th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '22), August 7–10, 2022, Northbrook, IL, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3535508.3545551

*Both authors contributed equally to this research.

BCB '22, August 7–10, 2022, Northbrook, IL, USA

ACM ISBN 978-1-4503-9386-7/22/08.
https://doi.org/10.1145/3535508.3545551

INTRODUCTION

Viral infections in (human) hosts are pervasive and occur through a variety of mechanisms. Viral genomes may be encoded using RNA (e.g., Hepatitis C Virus, influenza viruses, Coronavirus) or DNA (e.g. Hepatitis B, Papilloma virus) [23]. Retroviruses like HIV convert their RNA genomes into DNA and then back into RNA for transcription [9]. In all cases, the virus utilizes the host machinery to express viral genes and allow the virus to replicate in the host. Viral infections are directly responsible for many human diseases, and new strains may lead to epidemics or pandemics when introduced into an immunologically naive population. Thus, rapid detection of a viral infection is important.

When the viral family is known, specific sequences can be probed directly by searching databases of known viral sequences. If the viral family shows high divergence between members, detection based on direct sequence match can fail. Here, we address the following question: given training sequences from a diverged oncoviral family, can we learn latent representations that allow us to determine if a query sequence belongs to that viral family without a database search.

Specifically, we take the Papilloma virus (PVs) as an exemplar of a diverged oncoviral family. Human Papillomaviruses (HPV), especially HPV16 and HPV18 are important mediators of cervical and orapharyngeal cancers [4, 6, 16]. HPV mediated oropharyngeal cancers are reaching epidemic proportions, accounting for nearly 5% of all cancers [1]. In 2017, cervical cancer was the second most common cause of cancer related death for women across the world. Other Papilloma viruses (PVs), albeit less well understood, have also been implicated in human diseases including skin warts and rare diseases such as epidermodysplasia verruciformis. Furthermore, there is huge diversity of PVs with hundreds of strains identified [15].

Given its clinical importance, many tools have been developed to identify viral sequences in human cancer sequencing data [8, 14, 17, 28, 29], as well as tools to detect integration into the host

genome which is known to increase pathogenicity [2]. Even these specialized methods have suboptimal sensitivity for highly diverged sequences [7, 17]. To address this, ViFi [17], utilized an ensemble of hidden Markov models to identify viral sequences with high sensitivity. However, ViFi is slow and the high runtime is especially burdensome for analyzing large datasets. More importantly, ViFi requires specialized training, including phylogenetic reconstruction followed by construction of an ensemble of hidden Markov models for each sub-family.

While other classification-based approaches might be utilized, we also consider that human (host) genome samples often contain significant numbers (up to 5%) of uncharacterized microbial sequences [5]. Therefore, a strict classification-based learning using a 'closed set' approach might not work. In this work, we present DeepViFi, a transformer-based pipeline, to identify oncoviral reads in an open-set learning framework. We show that at 90% precision, DeepViFi achieves 90% recall and outperforms other neural network-based tools that use a 'closed-set' approach. Additionally, we demonstrate DeepViFi's efficacy in identifying HPV reads and the viral sub-family of the infecting strain in nine oropharyngeal tumour NGS datasets. Finally, DeepViFi can be retrained for other viral families without the need for the host, or contaminant genomes.

1 RELATED WORKS

Recently, deep learning tools have made tremendous progress in various biological applications such as protein folding [11], variant detection [20], and cell segmentation in images [22]. DeepVirFinder [21] and ViraMiner [25] leverage supervised learning with convolutional neural networks (CNNs) to address the kingdom-membership problem, with the goal of identifying viral sequences in metagenomic samples. They make the 'closed-set' assumption that the training and test sequences have the same label space.

In a different setting, DNABERT [10] uses the transformer architecture [27] to analyze human DNA contigs and produce latent representations of features on the human genome. These representations can be used for various downstream tasks such as predicting promoter regions and identifying transcription factor binding sites. However, DNABERT cannot be readily applied to short reads as it was trained on large contigs and tokenized at 3,4,5,6-mer level. Finally, it was trained only on human DNA. In this paper, we apply a similar framework to address the family-membership question for viruses using short read sequences.

2 METHOD

2.1 Overview

DeepViFi consists of three components: a transformer to produce latent representations of NGS short-reads, a random-forest (RF) model to classify the viral status of the latent representations, and a LightGBM model to identify the sub-family of the viral latent representations (Figure 1).

2.2 Method Details

Input Pre-processing. Given a read r of n base pairs, we tokenize each base-pair as a token, which empirically works better compared to tokens of larger substrings. We encode each token using the following mapping function $t : (A, C, G, T, N) \rightarrow (0, 1, 2, 3, 4)$ to obtain an encoded read vector $\mathbf{t_r} \in \mathbb{R}^{n\times 1}$ for each read r, where row i represents an encoding of the i-th token. We additionally define a vector, $\mathbf{p} = (1, 2, ..., n-1, n) \in \mathbb{R}^{n\times 1}$, to encode the position of each base pair in r.

Transformer with Self-attention Heads. DeepViFi utilizes a transformer to learn embedding matrices $\mathbf{M}_t, \mathbf{M}_p \in \mathbb{R}^{1\times d}$, where the embedding dimension d is a user-defined parameter, to obtain a dense representation combining $\mathbf{t_r}$ and $\mathbf{p}$. The initial encoding, denoted by $\mathbf{X}^{(0)} \in \mathbb{R}^{n\times d}$, is obtained using

$$\mathbf{X}^{(0)} = \mathbf{t_r}\mathbf{M}_t + \mathbf{p}\mathbf{M}_p,$$

The transformer has $\ell = 8$ encoders and $h = 16$ attention heads per encoder, where ℓ and h are hyperparameters. Each encoder i $(1 \leq i \leq \ell)$ transforms the input $\mathbf{X}^{(i-1)}$ (where $\mathbf{X}^{(0)}$ is the initial input encoding) from the previous layer to $X^{(i)}$ using self-attention heads as follows.

We denote $\mathbf{X}$(dropping the super-script) as the input to the encoders. We denote the weights of an attention head as $\mathbf{W_Q}, \mathbf{W_V}, \mathbf{W_K}$, without additional subscripts, for ease of exposition.

Let $\mathbf{V} = \mathbf{X}\mathbf{W_V}$ denote a learned representation of the input where $\mathbf{W_V} \in \mathbb{R}^{d\times\frac{d}{h}}$. The transformer outputs $\mathbf{Z} = \mathbf{S}\mathbf{V}$. Each resulting token $\mathbf{v_k}$ is mapped to $\mathbf{z_k} = \sum_j S_{kj}\mathbf{v_j}$, where $\sum_j S_{kj} = 1$. Intuitively, S_{kj} corresponds to the importance or attention of j-th token for the k-th token. To compute $\mathbf{S}$, we use the following:

(1) $\mathbf{Q} = \mathbf{X}\mathbf{W_Q}$, where $\mathbf{Q} \in \mathbb{R}^{n\times\frac{d}{h}}, \mathbf{W_Q} \in \mathbb{R}^{d\times\frac{d}{h}}$

(2) $\mathbf{K} = \mathbf{X}\mathbf{W_K}$ where $\mathbf{K} \in \mathbb{R}^{n\times\frac{d}{h}}, \mathbf{W_K} \in \mathbb{R}^{d\times\frac{d}{h}}$

(3) $\mathbf{D} = \frac{\mathbf{Q}\mathbf{K}^\mathbf{T}}{\sqrt{\frac{d}{h}}}; \mathbf{D} \in \mathbb{R}^{n\times n}$

(4) $\mathbf{S} = \text{Softmax}(\mathbf{D})$

where the Softmax operator is applied along the row dimension with $S_{ij} = \frac{e^{D_{ij}}}{\sum_l e^{D_{il}}}$, so that $0 \leq S_{ij} \leq 1$ for all i, j, and, $\sum_j S_{ij} = 1$ for all i.

The h outputs are concatenated and transformed using a dense-layer, and supplied to a final feed-forward network to produce the input for the next encoder. The output of the final encoder $(\mathbf{X}^{(\ell)} \in \mathbb{R}^{n\times d})$ represents a transformation of the original input read r. The complete architecture is shown in (Figure 4).

Random Forest Classification of Viral Reads. We use a random forest model to determine if the read was PV positive or negative by classifying its latent representation from the transformer (Figure 1). Specifically, we use an ensemble of 500 individual decision trees. Each individual tree outputs a class (HPV+ or HPV-) prediction of the input. The class with the most votes is the final prediction.

LightGBM for Viral Sub-family Classification. We use a LightGBM model [12] to further segregate the detected viral-reads into sub-families after experimenting with other classification methods, including a RF model (Figure 1). We fine-tuned the hyperparameters and found that a tree depth limit of 5 and the maximum number of leaves of 31 worked best. The model classified the latent representation of reads into one of Alpha, Beta, Gamma, or 'Other.'

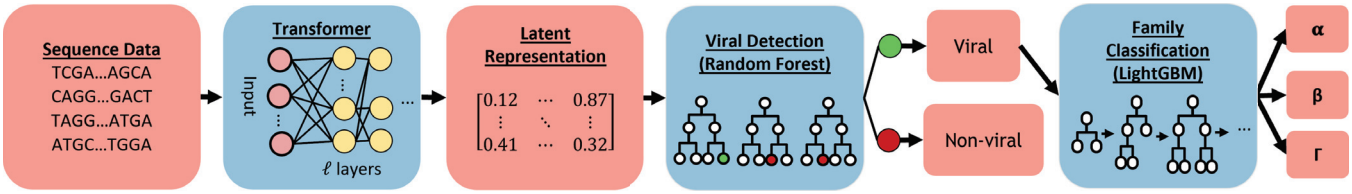

Figure 1: DeepViFi pipeline. The input to DeepViFi is DNA sequencing short reads. The transformer produces latent representations of the reads. These latent representations are fed to a random forest to determine if the read is HPV positive. The latent representations of the HPV positive reads are fed to a LightGBM model to determine their HPV subfamily.

2.3 Training and Inference

We trained the transformer using the masked language modeling paradigm. Random tokens are masked or replaced in the input and the transformer computes the likelihood for each token (A,T,C,G,N) in each position. We appended a fully connected layer with Softmax activation to the final encoder to produce

$$\mathbf{O} = \text{Softmax}(\mathbf{X}^{(\ell)}\mathbf{W}_{\mathbf{O}}),$$

where $O \in \mathbb{R}^{n\times 5}$ represents the likelihoods of each basepair at each position. The ground truth to the model is the unmasked read. We computed the loss by comparing the predicted tokens and the ground truth.

Masking. For masking a viral read of 150bp, we randomly chose 20% (30 positions) of the tokens for a masking procedure. Of these 30 chosen positions, we replaced 80% (24) of the tokens with a [MASK] token, 10% (3) with a random token, and 10% (3) with the original token (i.e. no change). Had we replaced all 20% of the to-be-masked tokens with a [MASK] token, the encoder would have learned to only observe the [MASK] tokens, and assumed that all non-masked tokens were correct. Hence, we replaced some of the to-be-masked tokens with a random or original token, forcing the encoder to keep a distributional contextual representation of every input token.

Hyper-parameter optimization. We optimized for the sparse categorical Cross-Entropy loss function using the Adam optimizer with a dynamic learning rate [27]. We trained on 8 GPUs for 150 epochs with early stopping with a patience of 10 epochs. We experimented with various other masking ratios. We masked 30% of the input sequence. However, this did not significantly change our loss convergence. We also tried masking contiguous regions in the input sequence instead of selecting random positions. In this case however, the loss did not converge despite experimenting with very low learning rates. When 30 base pairs (20%) of the sequence were contiguously masked, the network did not have enough context with the remaining 120 base pairs (80%) to accurately predict the continuous missing sequence. We also experimented with tokenizing 2-mers and 3-mers instead of single base pairs. In both experiments, the loss never converged despite various network configurations and learning rates.

We experimented with different values of the hyper-parameters $\ell \in \{6, 8, 10\}$, $d \in \{128, 256, 384\}$, and $h \in \{8, 16\}$, and empirically settled on $\ell = 8, d = 256, h = 16$.

Inference. For inference, we removed the final fully connected layer from the transformer and used the output of the final encoder ($\mathbf{X}^{(\ell)}$) as the latent representation of the input sequence. Recall that $\mathbf{X}^{(\ell)} \in \mathbb{R}^{n\times d}$, where we chose $n = 150$ and $d = 256$. For inference, we averaged the latent representation along the column dimension to produce a single vector of dimensionality 256. We treated this vector as the final latent representation of the input read.

3 DATASET GENERATION.

In a typical NGS experiment, bacteria and fungi can contaminate the target sequenced human and viral reads[19]. It is not possible currently to model all the contaminants, and instead we used an 'open-set' approach. Specifically, we trained DeepViFi exclusively on viral reads but tested on unseen classes such as contaminant and human reads.

3.1 Training Set.

We completely separated training and testing data by restricting training to reads generated from 337 PV reference genomes identified *prior to* 2018 from PaVE [26]. The training reference PV genomes ranged in length from 6953-8607 bp. We simulated reads of length 150 bp at 0.5×coverage, resulting in 1,145,800 reads.

While only the viral reads were used in the transformer to generate latent representations, we also used a negative data-set for training the random-forest classifier. In keeping with the open-set paradigm, we used 5,000 *randomly generated* reads for the negative set, but tested using real contaminant reads that were not part of training. These reads were combined with 5,000 HPV reads and used for classification using random forests.

We further classified the 337 pre-2018 references into alpha, beta, gamma, and "other" categories and randomly generated 6808, 4324, 5980, and 10856 reads, respectively. The imbalance in reads reflects the uneven number of references in each category.

3.2 Test Set

We exclusively used PV genomes from PaVE deposited on or after 2018 for testing. We simulated reads from each test genomes and generated 4 test sets. Each test set contained reads from 10 viral strains with similar genomic distances from the training genomes. We labeled the test sets as easy, intermediate, hard, and non-human, based on their increasing genomic distance from the training genomes. To maintain an open set paradigm, we added contaminant and human reads to each test set. We randomly chose the contaminant

and human reads from known contaminant and human genomes. The contaminant and human reads are considered non-viral.

We evaluated the LightGBM model on the 318 post-2018 references. We generated 0 Alpha, 150 Beta, 1200 Gamma, and 570 "other" reads. There were an uneven number of references for each category in the post-2018 strains. Notably, there were no alpha strains in this (realistic) test set which also represents a harder scenario as alpha strains are the easiest to identify. To rectify the balance, we also evaluated subfamily classification on HPV-mediated tumor patient data, where the alpha subfamily was over-represented.

3.3 HPV Mediated Primary Oropharyngeal Cancer Samples

We also evaluated on 9 oropharyngeal tumour samples from a recent study by Pang et al.[18], where the HPV status for each sample was previously determined. Each sample was HPV-16 positive and contained on average 800 million reads. After alignment filtering, each sample contained approximately 11.5 million reads on average. The ratio of viral to non-viral reads in each sample was .7% on average.

4 RESULTS

4.1 Method Comparisons

We compared DeepViFi against ViraMiner, DeepVirFinder, ViFi, and an off-the-shelf seq2seq model (Figure 2a). DeepViFi achieved a precision-recall AUC of 0.94, 0.94, 0.91, and 0.16 for detecting HPV reads on the easy, intermediate, hard, and non-human test sets, respectively. We retrained DeepVirFinder and ViraMiner model on a custom training set before evaluation (Methods). Despite retraining, ViraMiner and DeepVirFinder both achieved an AUC value of less than 0.5 on all 4 test sets (Methods).

We also trained and tested an off-the-shelf seq2seq model using eight bidirectional long short term memory (LSTM) encoders [24]. Similar to the transformer, the seq2seq model also generates latent representations which we used to detect viral sequences. We found that although the seq2seq model outperforms DeepVirFinder and ViraMiner on the easy and intermediate test set it still performs worse than DeepViFi. It also underperforms DeepViFi on the hard and non-human test sets (Figure 2a).

On the other hand, ViFi had high precision and recall values, achieving (0.996, 1.0), (0.996, 0.983), (0.996, 0.992), and (0.991, 0.481) on the intermediate, validation, difficult, and non-human test sets. ViFi utilizes HMM ensembles to learn representations that lead to highly accurate classification. While ViFi consistently achieved the highest accuracy, it also required prior construction of a phylogeny of the PV family, followed by a selection of clades to make an ensemble of HMMs. Therefore, DeepViFi reduces some major bottlenecks of ViFi—computational resources, expertise in setup/execution, and difficulty in repurposing to other applications—while maintaining comparable accuracy.

Sub-family Classification Accuracy. We trained sub-family identification by using a LightGBM classifier on the learned representations. The training data had 6,808, 4,324, 5,980 and 10,856 reads, respectively, in the four classes. We used a 70/30 split into training and validation, and the F1-score (harmonic mean of precision and recall) to measure accuracy of sub-family classification. The accuracy on the validation data was high at 0.88, 0.82, 0.83, and 0.9 for the four sub-families.

In contrast, the test data-set (drawn from strains discovered after 2018) had 0, 150, 1200, and 570 reads in the four classes, which did not match the training distribution. Nevertheless, the LightGBM achieved an overall F1-score of 87%, with accuracies of 0.63, 0.87, and 0.93 on Beta, Gamma, and Other classes.

4.2 Qualitative Analysis

At large genomic distances, viral reads are far enough apart that they cannot be distinguished from random reads, based solely on percent identity. HMMs address this by assigning different weights to different genomic locations. To understand what DeepViFi is learning, we plotted the distribution of the starting positions of all HPV reads in the easy testset (Figure 2c; top-panel) and compared them to the distribution of start positions of the viral reads that were *separated* from non-viral reads—specifically, reads that had first PC value greater than 1, second PC value > 0, and third PC value > 2 (Figure 2c; bottom-panel). The sharp distinction between the two plots suggests that the discriminating reads are drawn from specific locations of the HPV genome.

We then tested if the representations learned by DeepViFi could distinguish between PV sub-families Alpha, Beta, Gamma, and 'Other'. A PCA plot of the latent representations labeled by viral subfamily showed 4 visually distinct (although not linearly separable) clusters for each sub-family (Figure 2d).

4.3 Detecting HPV in Oropharyngeal tumor samples

The tumor WGS (whole genome sequencing) experiments contained $\sim$ 800M paired-ends per sample on the average and were available in the form of mappings to the human genome using the Burrows-Wheeler Aligner (BWA) [13]. All tumor samples were positive for HPV-16, which belongs to the Alpha subfamily of HPV. We filtered reads where both ends mapped to human sequence, and ran DeepViFi on the remaining reads using ViFi results as the ground truth. Each read from the paired-end was analyzed separately. For the 9 samples, we achieved an average precision-recall AUC of 0.90723.

DeepViFi also classified over 90% of the reads as belonging to the Alpha subfamily in each of the samples. The results are consistent with HPV-16 infection as HPV-16 belongs to the Alpha family. As low levels of other strains might be present, it was not possible to tell if the small number of misclassifications were due to classification error or the presence of other strains.

We performed PCA on the non-human reads in sample T49 to visualize if the representations of the HPV reads mapped to the same latent space as the representations of viral reads from the simulated test sets (Figure 3b). We demonstrate that representations were well separated from other reads and had a third PC value > 2, consistent with PC representations of the test sets.

DeepViFi took 12 hours to process 2 million reads on a CPU with 16 GB of memory. The time reduced to 50 minutes on a single Titan X GPU with 12 GB of memory. This was a significant speedup over the 48 hours taken by ViFi.

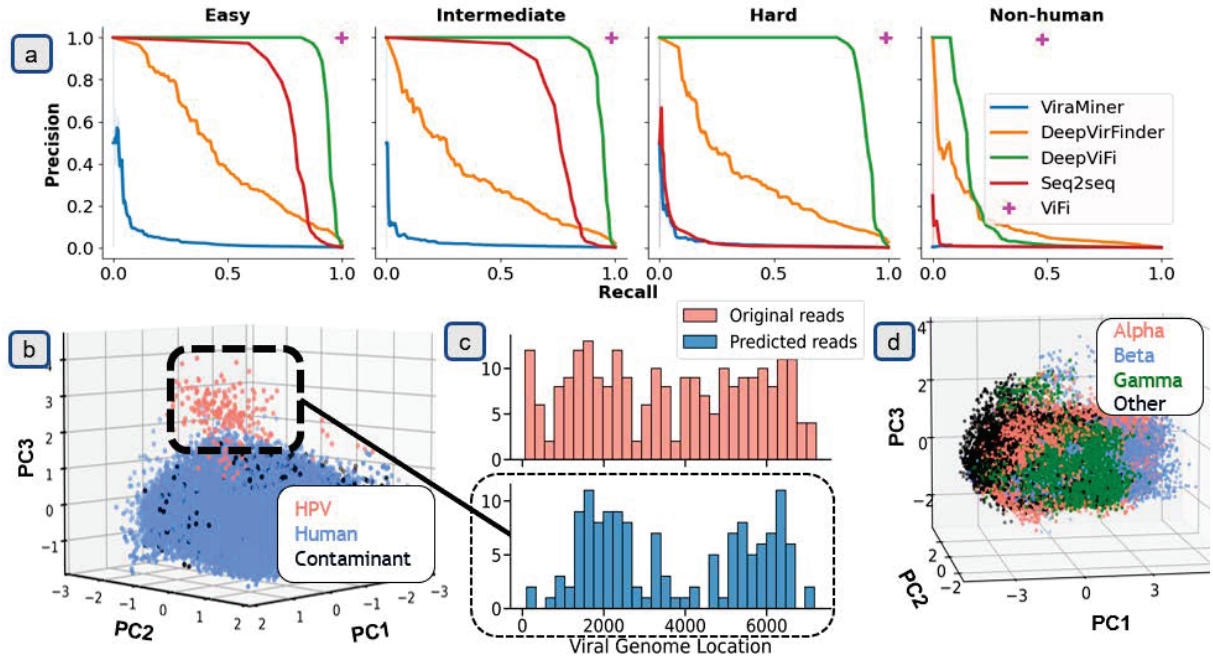

Figure 2: Test Set Performance and Analysis. (a) Precision-recall curves comparing different methods on the four test datasets. (b) PCA plot of latent representations of easy test set. (c) Read start locations of viral reads in easy test set. The top panel shows the start locations of all viral reads in the easy set. The bottom panel shows the start locations of the subset of viral reads that were highly separable in the PCA plot; i.e. the transformer was most confident of these reads as HPV fragments.

4.4 Detecting HBV in tumor samples

As an additional exemplar, we also trained and evaluated DeepViFi to detect HBV reads. We trained the DeepViFi pipeline on 73 known HBV genomes. We then evaluated the pipeline on three HBV-negative and three HBV-positive tumour samples. DeepViFi detected less than 30 reads as viral per million on the HBV negative samples.However, it detected more than a 100 reads as viral per million on the HBV positive samples.

5 DISCUSSION AND CONCLUSION

The identification of genomic sequences from a taxonomic group is an important problem that is not completely addressed. With highly diverged sequences, database search methods may not work. Hidden Markov models improve sensitivity by focusing the scoring on specific, conserved positions. However, they are a challenge to build, as they require extensive feature engineering that have to be tuned for each taxonomic group. Therefore, HMMs are not widely utilized, and sequence based searches continue to be widely used.

Recently, deep learning methods have provided many breakthroughs, especially in vision and natural language processing. Once an architecture is specified, the training does not require domain specific expertise, making them very attractive for multiple tasks. Here, we show that the taxonomic family identification is not successful using a closed-set modeling with neural architectures, because most real life examples provide instances of open-set learning.

In the context of viral family identification, we achieved very significant improvements by employing a transformer to learn latent representations of PV sequences. While our results easily outperformed closed-set learning using CNNs, they were still lower in sensitivity to a carefully trained ensemble of HMMs. This suggests that additional training using better sampling of the PV sequences is needed to improve representations.

Along the same vein, we can also group the methods surveyed in this paper as supervised and semi-supervised methods. The CNN based methods used here represent end-to-end supervised methods while seq2seq and DeepViFi represent semi-supervised methods. The supervised methods prioritize learning a classifier based primarily on the annotations to differentiate inputs. Meanwhile, the semi-supervised methods learn features to *characterize* the input. We show that simply learning a classifier is insufficient for problems such as identifying viral sequences in NGS data. Our results also match earlier observations on supervised and semi-supervised methods [3].

The representations of the viral sequences were so well separated from the other sequences that a simple, random-forest classifier was sufficient to identify viral reads. However, sub-family classification is a harder problem, and we had to use more sophisticated gradient boosting methods to achieve good results.

In summary, DeepViFi provides a framework for rapidly learning of representations from families, and a fast test for quickly and accurately identifying the target sequences in a larger data-set. The methods presented here are easily adaptable to a multitude of

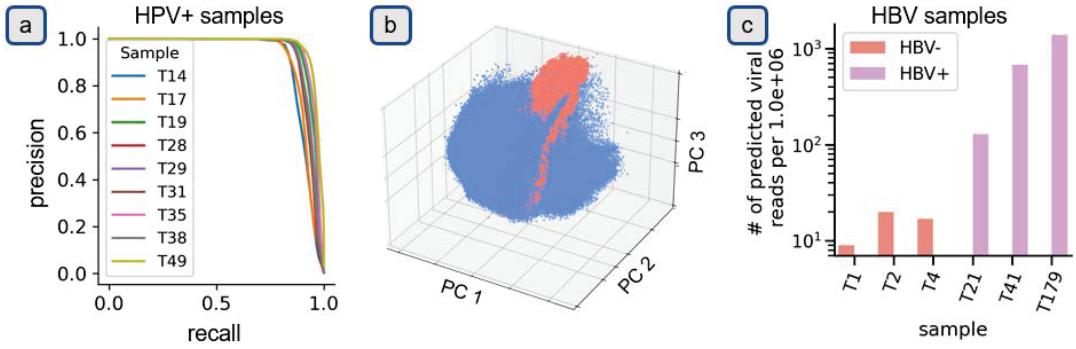

Figure 3: Tumour Sample Analysis. (a) Precision-recall curves for detecting HPV reads in tumour samples. (b) PCA of latent representations of T49 reads. (c)Number of predicted as HBV-positive per million reads in tumour samples.

viral families and likely to help with many tasks, including identification of novel, pathogenic viruses and removal of contaminant reads from whole genome sequencing runs. DeepViFi is available at https://github.com/UCRajkumar/DeepViFi.

DECLARATION OF INTERESTS

V.B. is a co-founder, consultant, SAB member and has equity interest in Boundless Bio, inc. and Abterra, Inc. The terms of this arrangement have been reviewed and approved by the University of California, San Diego in accordance with its conflict of interest policies.

REFERENCES

[1] Tara A Berman and John T Schiller. 2017. Human papillomavirus in cervical cancer and oropharyngeal cancer: one cause, two diseases. *Cancer* 123, 12 (2017), 2219–2229.

[2] D. L. Cameron, N. Jacobs, P. Roepman, P. Priestley, E. Cuppen, and A. T. Papenfuss. 2021. VIRUSBreakend: Viral Integration Recognition Using Single Breakends. *Bioinformatics* (May 2021).

[3] Ayan Chatterjee, Omair Shafi Ahmed, Robin Walters, Zohair Shafi, Deisy Gysi, Rose Yu, Tina Eliassi-Rad, Albert-László Barabási, and Giulia Menichetti. 2021. AI-Bind: Improving Binding Predictions for Novel Protein Targets and Ligands. arXiv:2112.13168

[4] A. K. Chaturvedi, E. A. Engels, R. M. Pfeiffer, B. Y. Hernandez, W. Xiao, E. Kim, B. Jiang, M. T. Goodman, M. Sibug-Saber, W. Cozen, L. Liu, C. F. Lynch, N. Wentzensen, R. C. Jordan, S. Altekruse, W. F. Anderson, P. S. Rosenberg, and M. L. Gillison. 2011. Human papillomavirus and rising oropharyngeal cancer incidence in the United States. *J Clin Oncol* 29, 32 (Nov 2011), 4294–4301.

[5] A. Gihawi, G. Rallapalli, R. Hurst, C. S. Cooper, R. M. Leggett, and D. S. Brewer. 2019. SEPATH: benchmarking the search for pathogens in human tissue whole genome sequence data leads to template pipelines. *Genome Biol* 20, 1 (10 2019), 208.

[6] I. J. Groves and N. Coleman. 2018. J PatholHuman papillomavirus genome integration in squamous carcinogenesis: what have next-generation sequencing studies taught us? *J Pathol* 245, 1 (05 2018), 9–18.

[7] Yusuke Hirose, Mayuko Yamaguchi-Naka, Mamiko Onuki, Yuri Tenjimbayashi, Nobutaka Tasaka, Toyomi Satoh, Kohsei Tanaka, Takashi Iwata, Akihiko Sekizawa, Koji Matsumoto, et al. 2020. High Levels of Within-Host Variations of Human Papillomavirus 16 E1/E2 Genes in Invasive Cervical Cancer. *Frontiers in microbiology* 11 (2020).

[8] D. W. Ho, K. M. Sze, and I. O. Ng. 2015. Virus-Clip: a fast and memory-efficient viral integration site detection tool at single-base resolution with annotation capability. *Oncotarget* 6 (2015), 20959–20963.

[9] Wei-Shau Hu and Stephen H Hughes. 2012. HIV-1 reverse transcription. *Cold Spring Harbor perspectives in medicine* 2, 10 (2012), a006882.

[10] Y. Ji, Z. Zhou, H. Liu, and R. V. Davuluri. 2021. DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. *Bioinformatics* (Feb 2021).

[11] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis. 2021. Highly accurate protein structure prediction with AlphaFold. *Nature* 596, 7873 (Aug 2021), 583–589.

[12] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc.

[13] H. Li and R. Durbin. 2009. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics* 25, 14 (Jul 2009), 1754–1760.

[14] J. W. Li, R. Wan, C. S. Yu, N. N. Co, N. Wong, and T. F. Chan. 2013. ViralFusionSeq: accurately discover viral integration events and reconstruct fusion transcripts at single-base resolution. *Bioinformatics* 29 (Mar 2013), 649–651.

[15] A. A. McBride. 2021. Human papillomaviruses: diversity, infection and host interactions. *Nat Rev Microbiol* (Sep 2021).

[16] I. M. Morgan, L. J. DiNardo, and B. Windle. 2017. Integration of Human Papillomavirus Genomes in Head and Neck Cancer: Is It Time to Consider a Paradigm Shift? *Viruses* 9, 8 (08 2017).

[17] N. D. Nguyen, V. Deshpande, J. Luebeck, P. S. Mischel, and V. Bafna. 2018. ViFi: accurate detection of viral integration and mRNA fusion reveals indiscriminate and unregulated transcription in proximal genomic regions in cervical cancer. *Nucleic Acids Res.* 46, 7 (Apr 2018), 3309–3325.

[18] J. Pang, N. Nguyen, J. Luebeck, L. Ball, A. Finegersh, S. Ren, T. Nakagawa, M. Flagg, S. Sadat, P. S. Mischel, G. Xu, K. Fisch, T. Guo, G. Cahill, B. Panuganti, V. Bafna, and J. Califano. 2021. Extrachromosomal DNA in HPV-Mediated Oropharyngeal Cancer Drives Diverse Oncogene Transcription. *Clin Cancer Res* 27, 24 (Dec 2021), 6772–6786.

[19] Sung-Joon Park, Satoru Onizuka, Masahide Seki, Yutaka Suzuki, Takanori Iwata, and Kenta Nakai. 2019. A systematic sequencing-based approach for microbial contaminant detection and functional inference. *BMC biology* 17, 1 (2019), 1–15.

[20] R. Poplin, P. C. Chang, D. Alexander, S. Schwartz, T. Colthurst, A. Ku, D. Newburger, J. Dijamco, N. Nguyen, P. T. Afshar, S. S. Gross, L. Dorfman, C. Y. McLean, and M. A. DePristo. 2018. A universal SNP and small-indel variant caller using deep neural networks. *Nat Biotechnol* 36, 10 (11 2018), 983–987.

[21] J. Ren, K. Song, C. Deng, N. A. Ahlgren, J. A. Fuhrman, Y. Li, X. Xie, R. Poplin, and F. Sun. 2020. Identifying viruses from metagenomic data using deep learning. *Quant Biol* 8, 1 (Mar 2020), 64–77.

[22] O. Ronneberger, P.Fischer, and T. Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention*. 234–241.

[23] John T Schiller and Douglas R Lowy. 2014. Virus infection and human cancer: an overview. *Viruses and human cancer* (2014), 1–10.

[24] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger (Eds.), Vol. 27. Curran Associates, Inc.

[25] A. Tampuu, Z. Bzhalava, J. Dillner, and R. Vicente. 2019. ViraMiner: Deep learning on raw DNA sequences for identifying viral genomes in human samples. *PLoS One* 14, 9 (2019), e0222271.

[26] Koenraad Van Doorslaer, Qina Tan, Sandhya Xirasagar, Sandya Bandaru, Vivek Gopalan, Yasmin Mohamoud, Yentram Huyen, and Alison A McBride. 2012. The Papillomavirus Episteme: a central resource for papillomavirus sequence data and analysis. *Nucleic acids research* 41, D1 (2012), D571–D578.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*. 6000–6010.

[28] Q. Wang, P. Jia, and Z. Zhao. 2013. VirusFinder: software for efficient and accurate detection of viruses and their integration sites in host genomes through next generation sequencing data. *PLoS One* 8 (2013), e64465.

[29] Q. Wang, P. Jia, and Z. Zhao. 2015. VERSE: a novel approach to detect virus integration in host genomes through reference genome customization. *Genome Med* 7 (2015), 2.

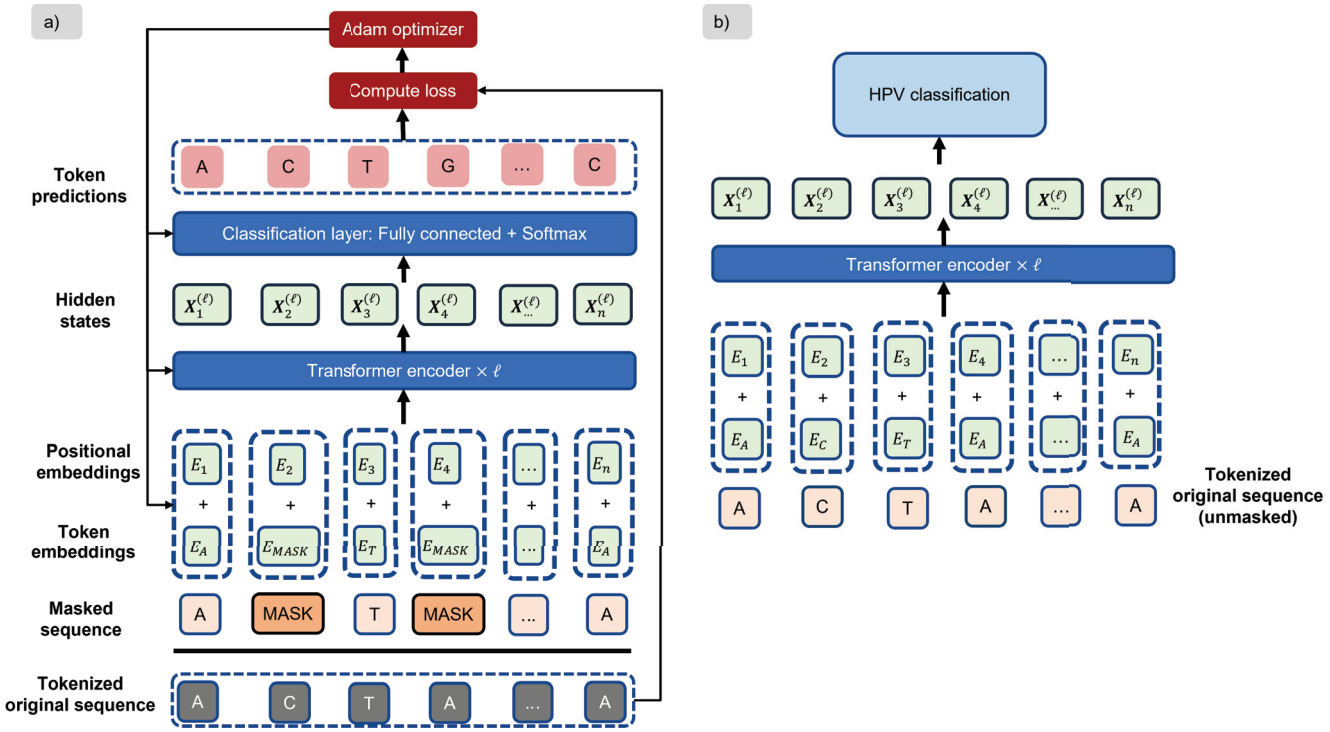

Figure 4: Detailed architecture of DeepViFi transformer. Left panel presents the training pipeline for DeepViFi's tranformer. Right panel presents the inference pipeline of the transformer. Related to figure 1.