Identification of Bacteriophage Genome Pieces with Representation Learning

Zeheng Bai [1], Yao-zhong Zhang [1,*], Satoru Miyano [1,2], Rui Yamaguchi [1,3,4], Kosuke Fujimoto [5,6], Satoshi Uematsu [5,6], and Seiya Imoto [1,6,*]

[1]Division of Health Medical Intelligence, Human Genome Center, The Institute of Medical Science, The University of Tokyo, Minato-ku, Tokyo, Japan

[2]M&D Data Science Center, Tokyo Medical and Dental University, Tokyo, Japan

[3]Division of Cancer Systems Biology, Aichi Cancer Center Research Institute, Nagoya, Japan

[4]Division of Cancer Informatics, Nagoya University Graduate School of Medicine, Nagoya, Japan

[5]Division of Metagenome Medicine, Human Genome Center, The Institute of Medical Science, The University of Tokyo, Minato-ku, Tokyo, Japan

[6]Collaborative Research Institute for Innovative Microbiology, The University of Tokyo, Bunkyo-ku, Tokyo, Japan

*To whom correspondence should be addressed.

Abstract

Motivation: Bacteriophages/Phages are viruses that infect and replicate within bacteria and archaea, and antibiotic resistance is one of the biggest threats to global health. Phages were used to therapeutically provide another potential solution for solving antibiotic resistance. To develop phage therapies, the identification o f p hages f rom m etagenome s equences i s t he fi rst st ep. Cu rrently, th ere ar e tw o main methods for identifying phages: database-based methods and alignment-free methods. Database-based methods typically use a large number of sequences as references but the prediction speed is slow; alignment-free methods are faster, and the amount of training data is limited for the deep learning models.
Results: We propose INHERIT which uses a deep representation learning model with pre-training to integrate both database-based and alignment-free methods, combining the strengths of both. Pre-training is used as an alternative way of acquiring knowledge representations from existing databases, while the BERT-style deep learning framework retains the advantage of alignment-free methods. We compare INHERIT with four existing methods on a third-party benchmark dataset. Our experiments show that INHERIT achieves a better performance with the F1-score of 0.9932. In addition, we find that pre-training two species separately helps the non-alignment deep learning model make more accurate predictions for both species.
Availability: The codes of INHERIT are now available in: https://github.com/Celestial-Bai/INHERIT.
Contact: yaozhong@ims.u-tokyo.ac.jp and imoto@hgc.jp
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The human gut is rich in bacteria and bacteriophages (phages for short) (Fuhrman, 1999; Edwards and Rohwer, 2005; Rohwer and Thurber, 2009; Rodriguez-Valera *et al.*, 2009; Reyes *et al.*, 2012) and a proportion of gastrointestinal diseases are due to specific bacteria (known as pathobionts)(Kamada *et al.*, 2012). One of the most common treatments currently available is the usage of antibiotics. However, this kind of treatment has several weaknesses. For instance, for the disease*CDI*, using antibiotics may harm the beneficial bacteria in the human gut and disrupt the ecological balance of the human intestinal microbes. Meanwhile, it may also cause its pathobiont C. difficile to gradually develop antibiotic resistance, resulting in CDI being prone to recurrence and failing to solve the fundamental problem (Lessa *et al.*, 2015; Davies and Davies, 2010; Munita and Arias, 2016; Andersson and Hughes, 2010; Laxminarayan *et al.*, 2013). Thus, phage therapy, which uses a phage to infect its corresponding host bacterium (Mirzaei and Maurice, 2017; Kutter *et al.*, 2010; Pires *et al.*, 2020), is thought to be one of the best ways to treat this disease currently. This approach avoids damaging the bacteria in the gut that are beneficial to people, compared to antibiotics.

To develop phage therapies, it is fundamental to identify phages from the metagenome nucleotide sequences first. Many methods have been proposed and we have categorized them into two groups: Database-based methods, such as VIBRANT (Kieft *et al.*, 2020) and VirSorter2 (Guo *et al.*, 2021); and Alignment-free methods, such as Seeker (Auslander

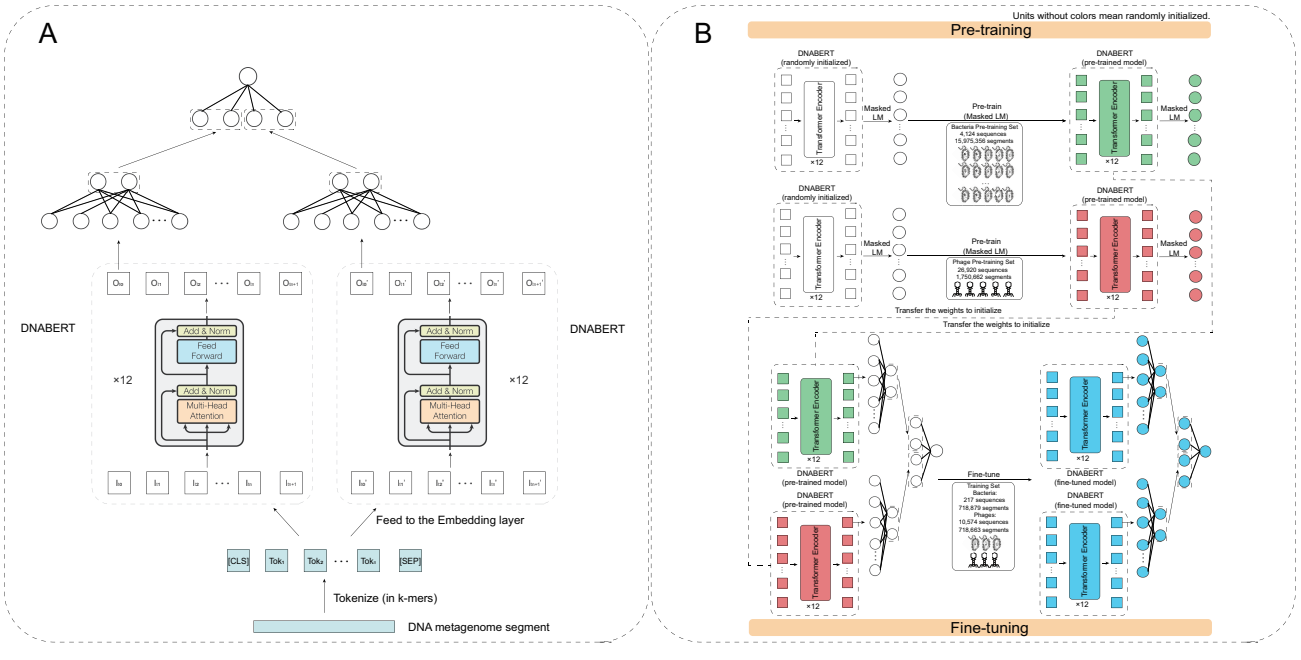

Fig. 1: Figure 1A shows the overall model architecture and explains how the label class is predicted given trained model parameters. For a given 500 bp nucleotide fragment, it is first encoded into tokens as the k-mer inputs. Then, they will be converted to the embeddings for two fine-tuned DNABERTs through the embedding layer independently. These embeddings are then fed into the fine-tuned Transformer encoders. The representations of the [CLS] of the two DNABERTs are extracted and then classified to generate four outputs. Eventually, INHERIT predicts the class from the four outputs of DNABERTs with a dense layer. This is how the label classes are predicted during the fine-tuning process. Figure 1B shows how the parameter weights are assigned through the pre-training and fine-tuning process. In pre-training, two randomly initialized DNABERTs pre-train independently with two datasets of different sizes and species. After that, the pre-trained weights in the lower non-linear-probing layers (green and red-colored in figure 1B) are first used for the model initialization. Then, in the fine-tuning step, all model parameters including those of the two pre-trained DNABERTs are fine-tuned together with a balanced training set.

et al., 2020) and DeepVirFinder (Ren *et al.*, 2020). Both types have their advantages and disadvantages, but they are complementary to some extent. For Database-based approaches, they are commonly based on multiple sequence alignment (Edgar and Batzoglou, 2006) with a large number of Profile Hidden Markov Models (Eddy, 1998), which can achieve good prediction performance. However, such approaches generally take a relatively long time for predictions. Alignment-free methods usually have good prediction speedups because most of the current deep learning frameworks can be accelerated with CUDA. However, limited by the training process, we need to balance the amount of phage and bacteria data (Japkowicz and Stephen, 2002). Moreover, subject to the deep learning framework, we generally need to split the sequence into several small segments to input into the deep learning framework. As a result, the number of bacterial sequences used in the training process is very limited, which affects the amount of information obtained. (See the introduction of them in Supplementary Contents 1.1).

In proposing the MSA Transformer, Rao, Roshan, et al. Rao *et al.* (2021) demonstrate that pre-trained Transformer-based models can have comparable performance to HMM Profiles and are even better in some cases. That indicates the core of the Database-based approaches, HMM Profiles, can be realized for a similar purpose by representation learning. Thus we can use the pre-train-fine-tune paradigm to combine the above two methods.

Here we propose INHERIT: **I**dentificatio**N** of bacteriop**H**ag**E**s using deep **R**epresentat**I**on model with pre-**T**raining. It also means our model "inherits" the characteristics from both Database-based approaches and Alignment-free methods. The code of INHERIT is now available at: https://github.com/Celestial-Bai/INHERIT. We show that using the representation learning framework can make improvements for deep learning models, and INHERIT also achieves the best performance in our tests.

The main contributions of our paper can be summarized as follows:

1. We propose an integrated model called INHERIT, which is based on the DNA sequence language model: DNABERT, and uses two pre-trained models as references. It reaches the best performance compared with four existing state-of-the-art approaches: VIBRANT, VirSorter2, Seeker, and DeepVirFinder. INHERIT outperforms them with the highest F1-score of 0.9932 in our test.
2. We use an independent pre-training strategy to deal with the data imbalance issue of bacteria and phages, and we find this strategy can help the deep learning framework make more accurate predictions for both species.

2 Methods

INHERIT is a model with DNABERT as the backbone (See the introduction in Supplementary Contents 1.1) and uses two pre-trained models as references (see the pipeline in Figure 1). We fine-tune them simultaneously to identify the metagenome sequences. The following will introduce the structure of INHERIT and the dataset we used.

2.1 Model Architecture and Pipeline

Here we show the architecture of INHERIT in Figure 1A and how parameter weights are assigned and transferred in the pre-training and fine-tuning process.

The sequences are split into several 500 bp-long segments as the input of INHERIT. When this sequence is not divisible by 500, we will use the

head of this sequence to complement the end of the sequence until it is divisible, which keeps the same with Seeker. For each segment, it will be encoded to tokens as k-mer inputs. Based on the past experience, (Ji *et al.*, 2021), we use 6-mer as input to DNABERT. Those tokens will be generated to embeddings for two fine-tuned DNABERTs through each embedding layer separately. Then each embedding is fed to each transformer encoder. Same with BERT, the representations of the "[CLS]" token is extracted and generate two outputs for each fine-tuned DNABERT through a dense layer. INHERIT eventually predicts the label class based on the four outputs with a dense layer. The prediction of the whole sequence is the average of the predictions of all the segments, which we call the "score" of the sequence. The default threshold of INHERIT is set to 0.5, which means if the average score is above 0.5, the sequence will be regarded as a phage, otherwise, it will be regarded as a bacterium. The threshold can be adjusted based on the specific situation: When we want INHERIT to predict the phages with higher confidence, for instance, we can adjust the threshold slightly higher.

Here we use the pre-train-fine-tune paradigm to build INHERIT. To deal with the problem of information bias may be caused by data imbalance, we pre-train bacteria and phages independently. Since the number of bacteria we have known is much larger than the number of phages and the length of bacteria is also longer (Chanishvili *et al.*, 2001), there are many more segments belonging to bacteria than to phages if the pre-training set is large enough. Since we want the pre-training set to carry a larger amount of data, then the segments belonging to the bacteria are bound to be much more than those belonging to the phage. If we combine bacteria and phages in one pre-trained model, then the model will learn much more information about bacteria than phages. Therefore, we prepared two pre-trained models for INHERIT. Same with Ji et al.Ji *et al.* (2021), we used Masked LM as the pre-training task.

Fine-tuning is very similar to the traditional training strategy. The difference between them is that in the traditional way, we randomly initialize the deep learning model before we start to train the model. However, we will transfer the weights of most of the pre-trained model as the initialization before we start to fine-tune the deep learning framework. Here we transfer the weights of non-linear-probing layers of the two pre-trained models to initialize INHERIT. The linear layers of INHERIT are still randomly initialized because we cannot transfer the weights from the pre-trained models. All model parameters are fine-tuned together with a balanced training set.

2.2 Datasets

Pre-training Sets

Since the bacteria pre-training set and the phage pre-training set would train two separate pre-trained models, and we wanted them to carry as much biological information as possible, we did not balance the size of the bacteria pre-training set and the phage pre-training set. For the bacteria pre-training set, we used ncbi-genome-download (https://github.com/kblin/ncbi-genome-download) to obtain the complete bacteria genome sequence from the NCBI FTP. The command we used was: ncbi-genome-download –formats fasta –assembly-levels complete bacteria. All of those bacteria sequences are high quality and we call them "bacteria assemblies". We sampled 4,124 bacteria sequences from them randomly because of the limitation of the physical memory. However, these 4,124 sequences can generate 15,975,346 segments and the dataset size is large enough.

For the phage pre-training set, we cannot obtain the phage sequence data in the same way. Since phage sequences cannot be found and downloaded directly in the NCBI FTP like the bacteria sequences, we directly searched for the keyword "phage" on NCBI and downloaded all sequences longer than 500 bp, and checked all of them manually. We also referred to the phage sequences used by Seeker and VIBRANT, and finally generated a pre-training set containing 26,920 phage sequences. To prevent overfitting, it did not include the phage sequences contained in the test set and validation set. These phage sequences can generate 1,750,662 segments, and the size is still large for a phage dataset.

Training Set and Validation Set

To make INHERIT learn more features to have better generalization ability. Thus, for bacteria during fine-tuning, we randomly selected 260 bacteria sequences that were not in the pre-training set and the test set but in bacteria assemblies. 217 bacteria sequences were used as the training set, generating 718,879 segments, and the remaining 43 were used as the validation set, generating 188,149 segments. However, for phages, we did not have as many sequences to choose, so we selected 10,574 phage sequences from the pre-training set that possessed a quality comparable to the bacteria assemblies, generating 718,663 segments. We also choose 2,643 sequences not in the pre-training set as the validation set, generating 186,121 segments.

We have posted the accessions of all the sequences used in each dataset and the sources we obtained in Supplementary File 1. For hyperparameters and platforms, please refer to Supplementary Contents 1.2.

3 Experiments

3.1 Benchmarking INHERIT with VIBRANT, VirSorter2, Seeker, DeepVirFinder

In this section, we compared INHERIT with 4 state-of-the-art methods: VIBRANT, VirSorter2, Seeker, and DeepVirFinder, which are the representatives of Database-based methods and Alignment-free methods respectively. We used a third-party benchmark dataset to make experiments and analyses.

3.1.1 Dataset

The test sets we used were one of the third-party benchmark tests previously proposed by Ho et al. Ho *et al.* (2021) for virus identification methods, called the RefSeq test set. Since our method just identifies phages and not other viruses, we only use data related to phages. The RefSeq test set contains 710 bacteria sequences and 1,028 phage sequences. However, since there were 19 bacteria sequences removed from RefSeq, we used the rest of them, including 691 bacteria sequences and 1,028 phage sequences to examine the performance of the methods on phage identification. It should be added that, in that article (Ho *et al.*, 2021), the authors split the sequences in this test set into 1kb to 15kb segments on average, and predicted the results and calculated metrics on the level of segments to make a benchmark test. However, since we consider that in applications, we want INHERIT to determine whether a sequence is a phage or not, all the predictions in our experiments are on the level of sequences.

3.1.2 Experimental Details

Baselines: INHERIT is the first model that integrates the features of Database-based approaches and Alignment-free approaches. Thus, we choose two representatives which have achieved state-of-the-art from each of the two methods, VIBRANT (Kieft *et al.*, 2020) and VirSorter2 (Guo *et al.*, 2021); and Seeker (Auslander *et al.*, 2020) and DeepVirFinder (Ren *et al.*, 2020), to compare it with INHERIT. To ensure the fairness of the comparison, we use the default commands of each method for predictions as much as possible, and we set corresponding rules for some methods to ensure that the prediction formats of each method are as consistent as possible (See Supplementary Contents 1.3 for details).

Model	Precision	Recall	Accuracy	F1-score	AUROC	AUPRC
VIBRANT	0.9541	0.9903	0.9656	0.9718	0.9595	0.9751
VirSorter2	0.9685	0.9893	0.9728	0.9787	0.9932	0.9971
Seeker	0.9264	0.8453	0.8674	0.8840	0.9382	0.9605
DeepVirFinder	0.8992	0.8502	0.8534	0.8740	0.8971	0.9435
INHERIT	**0.9884**	**0.9981**	**0.9919**	**0.9932**	**0.9996**	**0.9997**

Table 1. The benchmark results of VIBRANT, VirSorter2, Seeker, DeepVirFinder and INHERIT. For all methods we used the default commands for predictions. This can make our benchmarking as fair as possible.

Sequence	VIBRANT	VirSorter2	Seeker	DeepVirFinder	DNABERT	INHERIT
Bacteria	565.7887	1256.7568	**1.6068**	344.4573	53.7031	67.3556
Phage	19.1440	29.1797	**0.1139**	16.2840	2.7851	3.0127

Table 2. The average time (second) required for VIBRANT, VirSorter2, Seeker, DeepVirFinder, DNABERT and INHERIT to predict phage sequences and bacteria sequences in the test set. The time of each model implies their average running time (second) on predicting each bacterium and phage. The average length of bacteria samples on the test set is 3950500 bp, while the average length of phage samples on the test set is 75800 bp.

Evaluation Metrics: The evaluation metrics we chose are:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}},$$
$$\text{Recall} = \text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}},$$
$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}},$$
$$\text{F1} - \text{score} = \frac{2 \cdot \text{TP}}{2 \cdot \text{TP} + \text{FP} + \text{FN}},$$

and AUROC and AUPRC. In this paper, TP is the number of phage sequences successfully identified as phages, while FP is the number of bacteria sequences incorrectly identified as phages. TN is the number of bacteria sequences successfully identified as bacteria and FN is the number of phage sequences incorrectly identified as bacteria. AUROC and AUPRC are calculated based on the score of each sequence and the real value (phages are recorded as 1 and bacteria as 0).

Prediction Results: The predictions of VIBRANT, VirSorter2, Seeker, DeepVirFinder, and INHERIT for all the sequences in the test set can be seen in the Supplementary File 2.

3.1.3 Result and Analysis

INHERIT Performs Better than Existing Methods: From the results (see Table 1), compared to VIBRANT, VirSorter2, Seeker, and DeepVirFinder, INHERIT performs better than other existing methods. Moreover, the overall performance of INHERIT is an order of magnitude more precise. Especially, the high F1-score of INHERIT proves that INHERIT performs well enough when we use the default parameters and is competent for most application scenarios. From the p-values of the DeLong Test (DeLong *et al.*, 1988) of every two methods, the ROC curves for each of the two approaches are statistically significantly different. Therefore, the high AUROC and AUPRC indicate that INHERIT can better distinguish between phages and bacteria. That reflects we do not have to worry about large fluctuations in recall and precision when we want to adjust the threshold of INHERIT. For example, when we want to identify more phages, we can reduce the threshold slightly, so that INHERIT can complete its task without adding too many misidentified bacteria.

The performance of INHERIT is Less Affected by the Length of Nucleotide Sequences: We find the performance of INHERIT is less sensitive to length of genome sequences compared with other methods. We divide the phage and bacteria into five length intervals each based on the quartiles of lengths in the test set. The length intervals are according to the following rules: "minimum - first quartile", "first quartile - median", "median - third quartile", "third quartile - maximum of non-outliers", "the maximum of non-outliers - outliers". For phage sequences, there are "less than 42,000 bp", "42,000 bp - 51,000 bp", "51,500 bp - 91,600 bp" , and "greater than 91,600 bp". For bacteria sequences, there are "less than 2,788,000 bp", "2,788,000 bp - 4,107,500 bp", "4,107,500 bp - 4,938,000 bp ", "greater than 4,938,000 bp". We calculate the true positive rates of VIBRANT, VirSorter2, Seeker, DeepVirFinder, and INHERIT in the four intervals of phage and the true negative rates in the four intervals of bacteria respectively. This allows exploring whether these five methods perform consistently and robustly at different sequence lengths for both phages and bacteria. We can find that INHERIT is the only method that can maintain the true positive rate of 0.99 and the true negative rate of 0.97 in different intervals (see Figure 2). It does not always perform the best in every interval, but it is more robust than other models. For example, the true negative rate of VIBRANT in the "2788000 bp-4107500 bp" interval is 0.994, which is better than INHERIT (0.977), but its performance in the "4938000 bp- 8000000 bp" interval is only 0.767.

INHERIT Has Appropriate Speed of Prediction: INHERIT is still a model based on a deep learning framework, so it does not consume as long a time to predict as Database-based approaches. We calculate the average time required for VIBRANT, VirSorter2, Seeker, DeepVirFinder, and INHERIT to predict phage sequences and bacteria sequences in the test set. The results (see Table 2) show that the predictions of VIBRANT and VirSorter2 take much more time than Seeker and INHERIT. From our experiment, VIBRANT and VirSorter2 even cannot predict the whole bacteria test set at once. This indicates even though Database-based methods have better performance compared to Alignment-free methods, they consume a longer time to predict and are more sensitive to dataset size. However, INHERIT has high performance, and it also predicts the second fastest among VIBRANT, VirSorter, Seeker, and DeepVirFinder. Although INHERIT takes a longer time to predict compared to Seeker, it cost much less time than DeepVirFinder. DeepVirFinder selects different models for prediction based on the length of the target sequence. In our experiments, we offer the same environment to INHERIT, Seeker, and DeepVirFinder, but DeepVirFinder is still much slower than the other two methods, even if it is based on convolutional neural network.We conjecture that it may be because it consumes a lot of time in the process of DNA sequence encoding and model selection. INHERIT uses one model to make predictions for each sequence regardless of the length, and these show that INHERIT can give accurate predictions in an appropriate time budget.

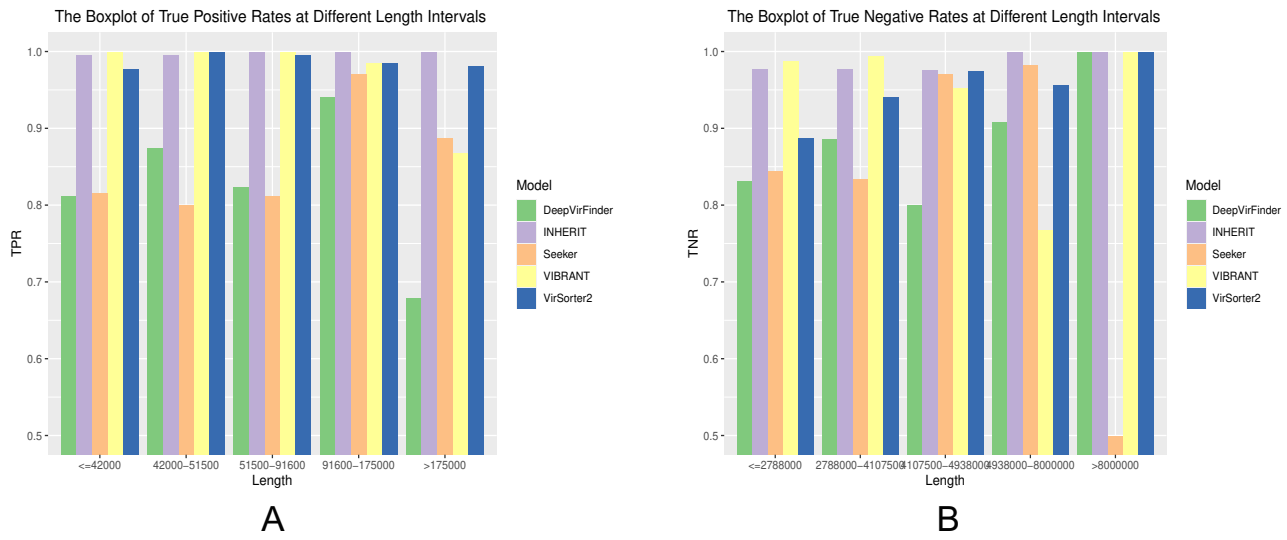

Fig. 2: Figure 2A shows the true positive rates for each method at different length intervals. Figure 2B shows the true negative rate of each method at different length intervals.

Model	Precision	Recall	Accuracy	F1-score	AUROC	AUPRC
Sequence Level						
DNABERT	0.9992	0.9224	0.9229	0.9593	0.9751	0.9996
INHERIT (w/o pre-train)	**0.9996**	0.9255	0.9263	0.9611	0.9839	0.9997
INHERIT	0.9992	**0.9894**	**0.9888**	**0.9943**	**0.9971**	**1.0000**
Segment Level						
DNABERT	0.8767	0.8809	0.8792	0.8788	0.9508	0.9507
INHERIT (w/o pre-train)	0.9044	0.8951	0.9008	0.8997	0.9530	0.9422
INHERIT	**0.9436**	**0.9650**	**0.9539**	**0.9542**	**0.9875**	**0.9882**

Table 3. The comparison among DNABERT (without pre-training, i.e., training from scratch), INHERIT (without pre-training), and INHERIT. We use the validation set to evaluate the performance of the three models. Here we evaluate all the three models both on the level of sequence and segment because the validation set is balanced on the level of the segment while severely imbalanced on the level of sequence.

3.2 Ablation Study

We used the validation set to make an additional experiment for considering ablation. In this section, we discuss how the pre-training strategy and deep learning structure of INHERIT affect the performance.

There are two main differences between INHERIT and the backbone: DNABERT. First, We used a strategy to pre-train two species independently. For the purpose of using pre-training to provide references and to solve the problem of imbalanced bacterial and phage datasets, we pre-train bacteria and phages separately. In addition to this, we utilize two pre-trained models and have them fine-tuned simultaneously. This model structure has twice the number of parameters as DNABERT. To explore whether pre-training would help deep learning frameworks improve performance, we make an ablation study on the validation set. In training INHERIT, we first train two pre-trained models on two separate datasets of different sizes and species. Then, the weights of these two pre-trained models are used as initialization while fine-tuning simultaneously on the training set. Therefore, we chose to compare an INHERIT that does not use any pre-trained models and directly initializes two DNABERTs randomly and fine-tunes them on the training set at the same time. We denote it as INHERIT (w/o pre-train). Further, since we use two DNABERTs, INHERIT is about twice as large as DNABERT in terms of the number of parameters. Therefore, we also train a DNABERT to reflect the effect of the number of parameters on the model performance. All models have used the same hyperparameters as INHERIT during training.

We test all models' performance on the validation set. Here we evaluate all the three models both on the level of sequence and segment because the validation set is balanced on the level of the segment while severely imbalanced on the level of sequence (43 bacteria and 2,643 phages). The differences in performance reflected by different metrics may be biased when we evaluate them on the level of sequence. Therefore, we attach the confusion matrix of the three models on the level of both sequence and segment in the Supplementary File 3.Based on the results (see Table 3), INHERIT improves on the good performance of DNABERT in all metrics and performs better than INHERIT (w/o pre-train) in most of the metrics on the level of sequence and performs the best on the level of segment. Meanwhile, although we train on the level of the segment, the better performance on the segment, the more accurate the prediction of the whole genome sequence. The increase in the number of parameters helps INHERIT, allowing it to outperform DNABERT. However, pre-training can give INHERIT a further boost in very high accuracy without increasing the number of parameters. We also make DeLong Test for these three models with each other and these three curves are statistically significant with each other. Compared to INHERIT (w/o pre-train), INHERIT makes a significant improvement in F1-score from 0.9611 to 0.9943 on the level of sequence and from 0.8788 to 0.9542 on the level of segment. This reflects the independent pre-training strategy can help the deep learning framework to distinguish better and make more accurate predictions on both species. Compared with DNABERT, both two differences help INHERIT make

better performance. We also analyze the superiority of the structure of DNABERT (see Supplementary Contents 1.5).

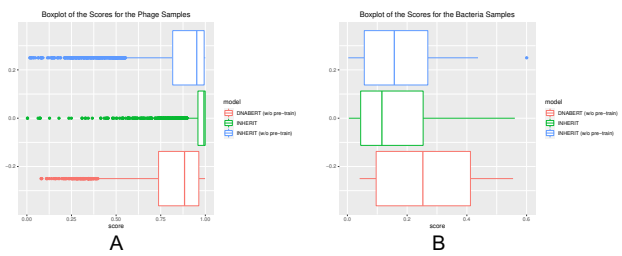

Fig. 3: Figure 3A shows the boxplot of the scores of DNABERT, INHERIT (w/o pre-train), and INHERIT for the phage samples in the validation set. Figure 3B is the boxplot of the scores of DNABERT, INHERIT (w/o pre-train), and INHERIT for the bacteria samples in the validation set.

Pre-training independently will allow the deep learning framework to give predictions that are closer to the true values. We plot boxplots of the scores given by DNABERT, INHERIT (w/o pre-train), and INHERIT for phage samples and bacteria samples in the validation set on sequence level (see Figure 3). For the vast majority of samples, the score generated by INHERIT is closer to the true value (1 for phages and 0 for bacteria). For example, for phage MH576962, DNABERT gives a score of 0.4707, and INHERIT (w/o pre-train) gives a score of 0.4968. Neither model classify it correctly. However, the score from INHERIT is 0.8182. Compared to the other two models, INHERIT not only classifies it correctly but also gets a big improvement in the score, which is more towards 1. This means that independent pre-training provides great help for deep learning frameworks to give more correct predictions on both species. This point even stands for the samples where none of the three predict correctly or all of the three predict correctly. For example, for phage HQ906662, the DNABERT, INHERIT (w/o pre-train), and INHERIT prediction scores were 0.4209, 0.2908, and 0.4547, respectively; for bacterium NZ_CP018197, the DNABERT, INHERIT (w/o pre-train), and INHERIT prediction scores were 0.2237, 0.1210, and 0.0621, respectively.

We also tested the prediction speed of DNABERT. From the results (see Table 2), the average prediction speed of DNABERT is 53.7031 seconds for bacteria sequences and 2.7851 seconds for phage sequences in the test set. It is slightly faster than INHERIT. This is because INHERIT has a more complex structure than DNABERT. However, INHERIT can still give predictions faster than other methods.

4 Conclusions

In this work, we propose INHERIT, an integrated method that combines both Database-based and Alignment-free approaches under a unified deep representation learning framework. It uses two pre-trained models as references and keeps the features of Alignment-free methods by the deep learning structure. On a third-party benchmark dataset, we compared the proposed method with VIBRANT, VirSorter2, Seeker, and DeepVirFinder, representing Database-based methods and Alignment-free methods. We demonstrate that INHERIT can achieve better performance than the four existing methods in all metrics. In particular, INHERIT improves the F1-score from 0.9787 to 0.9932. Meanwhile, we also prove that using an independent pre-training strategy can make deep learning models make better predictions on both species.

References

Alipanahi, B., Delong, A., Weirauch, M. T., and Frey, B. J. (2015). Predicting the sequence specificities of dna-and rna-binding proteins by deep learning. *Nature biotechnology*, **33**(8), 831–838.

Andersson, D. I. and Hughes, D. (2010). Antibiotic resistance and its cost: is it possible to reverse resistance? *Nature Reviews Microbiology*, **8**(4), 260–271.

Auslander, N., Gussow, A. B., Benler, S., Wolf, Y. I., and Koonin, E. V. (2020). Seeker: Alignment-free identification of bacteriophage genomes by deep learning. *Nucleic acids research*, **48**(21), e121–e121.

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, **35**(8), 1798–1828.

Chanishvili, N., Chanishvili, T., Tediashvili, M., and Barrow, P. A. (2001). Phages and their application against drug-resistant bacteria. *Journal of Chemical Technology & Biotechnology*, **76**(7), 689–699.

Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., and Sutskever, I. (2020). Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR.

Choromanski, K., Lin, H., Chen, H., and Parker-Holder, J. (2021). Graph kernel attention transformers. *arXiv preprint arXiv:2107.07999*.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., and Salakhutdinov, R. (2019). Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Davies, J. and Davies, D. (2010). Origins and evolution of antibiotic resistance. *Microbiology and molecular biology reviews*, **74**(3), 417–433.

DeLong, E. R., DeLong, D. M., and Clarke-Pearson, D. L. (1988). Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, pages 837–845.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Eddy, S. R. (1998). Profile hidden markov models. *Bioinformatics (Oxford, England)*, **14**(9), 755–763.

Edgar, R. C. and Batzoglou, S. (2006). Multiple sequence alignment. *Current opinion in structural biology*, **16**(3), 368–373.

Edwards, R. A. and Rohwer, F. (2005). Viral metagenomics. *Nature Reviews Microbiology*, **3**(6), 504–510.

Fuhrman, J. A. (1999). Marine viruses and their biogeochemical and ecological effects. *Nature*, **399**(6736), 541–548.

Guo, J., Bolduc, B., Zayed, A. A., Varsani, A., Dominguez-Huerta, G., Delmont, T. O., Pratama, A. A., Gazitúa, M. C., Vik, D., Sullivan, M. B., *et al.* (2021). Virsorter2: a multi-classifier, expert-guided approach to detect diverse dna and rna viruses. *Microbiome*, **9**(1), 1–13.

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., and Smith, N. A. (2020). Don't stop pretraining: adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Ho, S. F. S., Millard, A. D., and van Schaik, W. (2021). Comprehensive benchmarking of tools to identify phages in metagenomic shotgun sequencing data. *bioRxiv*.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, **9**(8), 1735–1780.

Hyatt, D., Chen, G.-L., LoCascio, P. F., Land, M. L., Larimer, F. W., and Hauser, L. J. (2010). Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC bioinformatics*, **11**(1), 1–11.

Japkowicz, N. (2000). The class imbalance problem: Significance and strategies.

Japkowicz, N. and Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent data analysis*, **6**(5), 429–449.

Ji, Y., Zhou, Z., Liu, H., and Davuluri, R. V. (2021). Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*.

Kamada, N., Kim, Y.-G., Sham, H. P., Vallance, B. A., Puente, J. L., Martens, E. C., and Núñez, G. (2012). Regulated virulence controls the ability of a pathogen to compete with the gut microbiota. *Science*, **336**(6086), 1325–1329.

Kieft, K., Zhou, Z., and Anantharaman, K. (2020). Vibrant: automated recovery, annotation and curation of microbial viruses, and evaluation of viral community function from genomic sequences. *Microbiome*, **8**(1), 1–23.

Kutter, E., De Vos, D., Gvasalia, G., Alavidze, Z., Gogokhia, L., Kuhl, S., and Abedon, S. T. (2010). Phage therapy in clinical practice: treatment of human infections. *Current pharmaceutical biotechnology*, **11**(1), 69–86.

Laxminarayan, R., Duse, A., Wattal, C., Zaidi, A. K., Wertheim, H. F., Sumpradit, N., Vlieghe, E., Hara, G. L., Gould, I. M., Goossens, H., *et al.* (2013). Antibiotic resistance—the need for global solutions. *The Lancet infectious diseases*, **13**(12), 1057–1098.

Lecun, Y. and Bottou, L. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, **86**(11), 2278–2324.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. (2020). Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, **36**(4), 1234–1240.

Lessa, F. C., Mu, Y., Bamberg, W. M., Beldavs, Z. G., Dumyati, G. K., Dunn, J. R., Farley, M. M., Holzbauer, S. M., Meek, J. I., and Phipps, E. C. (2015). Burden of clostridium difficile infection in the united states. *New England Journal of Medicine*, **372**(24), 2369–2370.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2019). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Liu, F., Miao, Y., Liu, Y., and Hou, T. (2020). Rnn-virseeker: a deep learning method for identification of short viral sequences from metagenomes. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

Mirzaei, M. K. and Maurice, C. F. (2017). Ménage à trois in the human gut: interactions between host, bacteria and phages. *Nature Reviews Microbiology*.

Mostavi, M., Chiu, Y.-C., Huang, Y., and Chen, Y. (2020). Convolutional neural network models for cancer type prediction based on gene expression. *BMC medical genomics*, **13**(5), 1–13.

Munita, J. M. and Arias, C. A. (2016). Mechanisms of antibiotic resistance. *Microbiology spectrum*, **4**(2), 4–2.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Pires, D. P., Costa, A. R., Pinto, G., Meneses, L., and Azeredo, J. (2020). Current challenges and future opportunities of phage therapy. *FEMS microbiology reviews*, **44**(6), 684–700.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. (2018). Improving language understanding by generative pre-training.

Rao, R., Liu, J., Verkuil, R., Meier, J., Canny, J. F., Abbeel, P., Sercu, T., and Rives, A. (2021). Msa transformer. *bioRxiv*.

Ren, J., Song, K., Deng, C., Ahlgren, N. A., Fuhrman, J. A., Li, Y., Xie, X., Poplin, R., and Sun, F. (2020). Identifying viruses from metagenomic data using deep learning. *Quantitative Biology*, **8**(1).

Reyes, A., Semenkovich, N. P., Whiteson, K., Rohwer, F., and Gordon, J. I. (2012). Going viral: next-generation sequencing applied to phage populations in the human gut. *Nature Reviews Microbiology*, **10**(9), 607–617.

Rodriguez-Valera, F., Martin-Cuadrado, A.-B., Rodriguez-Brito, B., Pasic, L., Thingstad, T. F., Rohwer, F., and Mira, A. (2009). Explaining microbial population genomics through phage predation. *Nature Precedings*, pages 1–1.

Rohwer, F. and Thurber, R. V. (2009). Viruses manipulate the marine environment. *Nature*, **459**(7244), 207–212.

Storey, J. D. (2003). The positive false discovery rate: a bayesian interpretation and the q-value. *The Annals of Statistics*, **31**(6), 2013–2035.

Tay, Y., Bahri, D., Zheng, C., Brunk, C., Metzler, D., and Tomkins, A. (2020). Reverse engineering configurations of neural text generation models. *arXiv preprint arXiv:2004.06201*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, **32**.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., and Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of AAAI*.