Identification of Bacteriophages Using Deep Representation Model with Pre-training

Zeheng Bai [1], Yao-zhong Zhang [1*], Satoru Miyano [3], Rui Yamaguchi [4], Satoshi Uematsu [2], Seiya Imoto [1,2*]
[1]Division of Health Medical Intelligence, Institute of Medical Science, the University of Tokyo
[2]Human genome center, Institute of Medical Science, the University of Tokyo
[3]M&D Data Science Center, Tokyo Medical and Dental University
[4]Division of Cancer Systems Biology, Aichi Cancer Center Research Institute

Abstract

Bacteriophages/Phages are viruses that infect and replicate within bacteria and archaea. Antibiotic resistance is one of the biggest threats to global health. The therapeutic use of bacteriophages provides another potential solution for solving antibiotic resistance. To develop phage therapies, the identification of phages from metagenome sequences is the fundamental step. Currently, several methods have been developed for identifying phages. These methods can be categorized into two types: database-based methods and alignment-free methods. The database-based approach, such as VIBRANT, utilizes existing databases and compares sequence similarity between candidates and those in the databases. The alignment-free method, such as Seeker and DeepVirFinder, uses deep learning models to directly predict phages based on nucleotide sequences. Both approaches have their advantages and disadvantages.

In this work, we propose using a deep representation learning model with pre-training to integrate the database-based and non-alignment-based methods (we call it INHERIT). The pre-training is used as an alternative way for acquiring knowledge representations from existing databases, while the BERT-style deep learning framework retains the advantage of alignment-free methods. We compared the proposed method with VIBRANT and Seeker on a third-party benchmark dataset. Our experiments show that INHERIT achieves better performance than the database-based approach and the alignment-free method, with the best F1-score of 0.9868. Meanwhile, we demonstrated that using pre-trained models helps to improve the non-alignment deep learning model further.

*To whom correspondence should be addressed: Yao-zhong Zhang <yaozhong@ims.u-tokyo.ac.jp>, and Seiya Imoto <imoto@hgc.jp>

1 Introduction

The human gut is rich in bacteria and bacteriophages (phages for short) and a proportion of gastrointestinal diseases are due to specific bacteria (known as pathobionts)(Kamada et al., 2012), and one of the most common treatments available is the usage of antibiotics at present. However, this kind of treatment has several weaknesses. For instance, for the disease CDI, the use of antibiotics may harm the beneficial bacteria in the human gut and disrupt the ecological balance of the human intestinal microbes. Meanwhile, the use of antibiotics 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). Thus, it is thought to be the best way to treat this disease currently is phage therapy, which uses a phage to infect its corresponding host bacterium(Mirzaei and Maurice, 2017). This approach avoids damaging the bacteria in the gut that are beneficial to people compared to antibiotics. Therefore, it is necessary to investigate the relationship between phages and their host bacteria in the human gut.

In recent years, researchers have been working on this topic. Fujimoto, K. et al. (Fujimoto et al., 2020) analyzed fecal samples from 101 healthy Japanese individuals with CDI and identified novel antibacterial enzymes that could target the pathobiont of the disease. The researchers extracted the metagenome sequences from the samples after which they needed to process phage identification. It is a fundamental step and researchers have proposed many methods recently. We roughly summarize several approaches which work on identifying phages, and they can be roughly classified into two categories: database-based methods, such as VIBRANT (Kieft et al., 2020) and VirSorter2 (Guo et al., 2021); and alignment-free methods, such as Seeker (Auslander et al., 2020) and DeepVirFinder

(Ren et al., 2020). Both types have their advantages and disadvantages, but they are complementary to some extent.

Recently, the pre-train-fine-tune paradigm using the Transformer architecture is very popular in other areas such as natural language processing. Among them, BERT has excelled in many fields and even reached state-of-the-art in many tasks. DNA sequence as an important medium for conveying biological information just like a language, we believe that BERT can also be used in DNA sequence analysis. Ji, Y. et al proposed DNABERT (Ji et al., 2021), an extension of BERT that can use DNA sequences as input. DNABERT can be used for the pre-training process to fully learn the information about phages and bacteria, which is very similar to HMM Profiles in database-based approaches. That indicates that the core of the database-based approach: sequence alignment, can be used for a similar purpose by representation learning approaches. Therefore, we can learn the biological features of bacteria and phages by using the pre-train-fine-tune paradigm with DNABERT, so we can unify the advantages of both methods into a model with fast prediction and high accuracy simultaneously. Thus, 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 codes of INHERIT are now available in: `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 test.

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

1 BERT-style deep learning framework is feasible for identifying phages, even if better than LSTM in our test.

2 Adding pre-trained models can help deep learning models make improvements on identifying phages. We also trained DNABERT without any pre-trained models, and INHERIT performs better on most of the metrics.

3 INHERIT reaches the best performance compared with two state-of-the-art approaches: VIBRANT and Seeker. Because INHERIT is the first integrated model with the representation learning framework, we compare it with two representatives of database-based methods and alignment-free methods. INHERIT performs the best in our test with an F1-score of 0.9868.

Related Work

Database-based methods

This kind of method takes the genome sequence and first predicts its compiled protein using tools such as Prodigal (Hyatt et al., 2010), then compares it with the sequence in the database by Profile Hidden Markov Models to determine whether the sequence is a phage. While these methods can identify phages with high accuracy in general, they also have two drawbacks. First, the computational time required to identify phages by these methods is usually long. If a large number of metagenome sequences need to be identified, or if the sequences need to be identified quickly, database-based methods are not suitable for these situations. At the same time, such methods are largely limited by the sequences within the reference database, so it is difficult for such methods to identify phages with little sequence similarity to those in the reference databases.

Alignment-free methods

This kind of method uses deep learning to extract features directly from DNA sequences to determine whether they are phages or not. One of the popular approaches is to use Long Short Term Memory (LSTM) models for training this problem. When the sequences are converted from bases to values, there are usually two ways: one is to convert bases to four values of 1,2,3,4, and the other is to use the one-hot encoding. A typical example of this approach is Seeker. It uses both of these strategies to encode the sequence and then uses LSTM to classify the sequences.

The DNA sequence is changed from a sequence to a matrix with one-hot encoding, so another common idea is to treat this matrix like an image and use Convolutional Neural Network (CNN) to train. Therefore, researchers proposed the DeepVirFinder which is trained by CNN. It is special because it takes not only the original sequences as input, but also includes the reverse complemented sequences. The other feature is that it chooses the different models to predict the sequence based on its length.

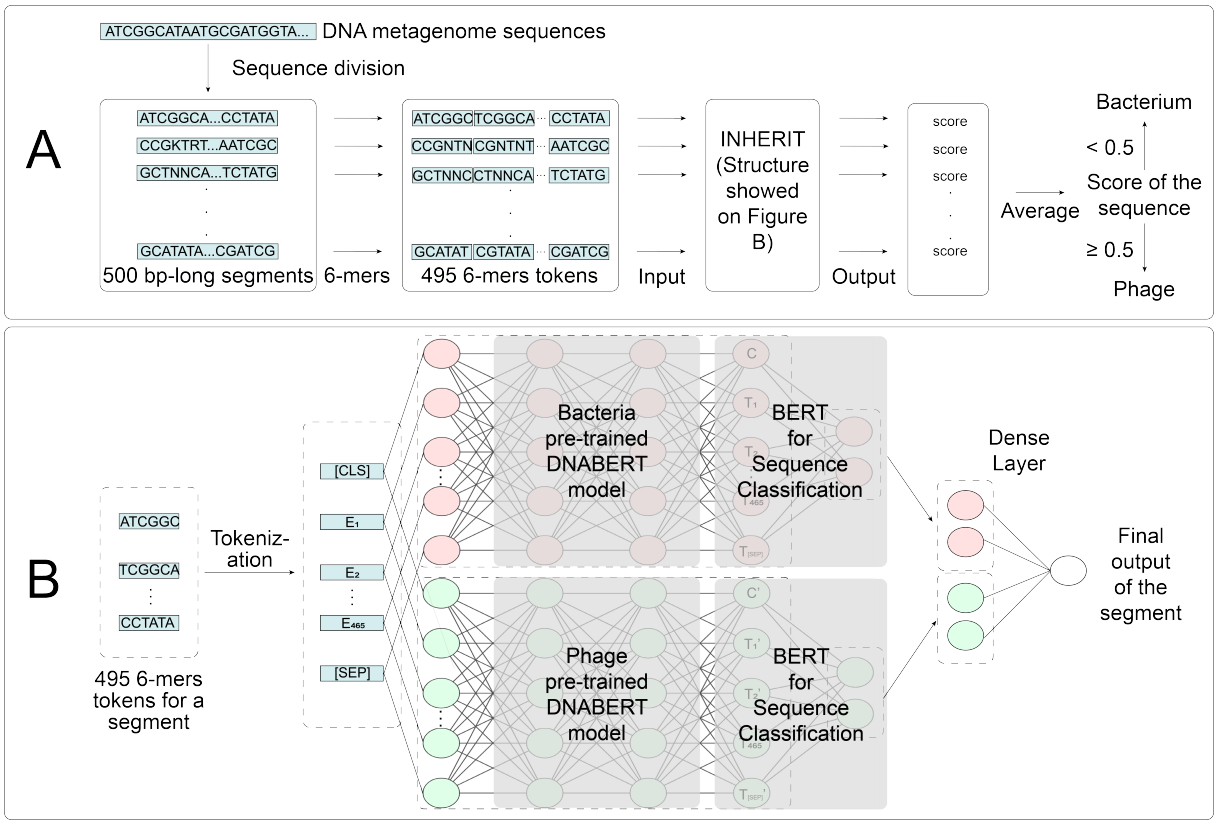

Figure 1: Figure 1 mainly illustrates the pipeline of INHERIT. For such a DNA metagenome sequence as the example, as Figure 1A shows, it is first divided into several 500 bp-long segments. Then each segment is generated into 495 k-mers tokens. Any degenerate bases (like "K", "R", and "N") are replaced with "N" consistently during this process. Those tokens are the inputs of INHERIT, and Firgure 1B shows the structure of INHERIT. For the 495 tokens for one segment as an instance, they are first tokenized into numeric vectors by DNABERT tokenizer, added the "[CLS]" token on the head, and appended "[SEP]" token at the end. All of them are input into the two pre-trained models, bacteria pre-trained DNABERT model and phage pre-trained DNABERT model. Used the BERT for Sequence Classification Function, both of them can output two logits outputs respectively. Those four outputs are run through the dense layer and we can get the output (called "score") for a segment. The score of the sequence is the average of the score of its segments, and we set the threshold at 0.5 as default. If the score of the sequence is above 0.5, it will be identified as a phage, otherwise it will be identified as a bacterium.

However, the alignment-free methods can only extract some biological features from the training set itself during the training process, but this process of extracting features is limited. Because when we train a deep learning model on a classification task, we usually need an equal or similar amount of positive and negative data. However, the number of phages we can obtain from databases is much less than their hosts, bacteria, and the genome sequence lengths of phages usually are much shorter than those of bacteria. Thus, in the past, the number of bacteria selected by these methods tended to be small, which caused information about bacteria to rely too much on these small amounts of bacterial sequence data with a relatively high degree of randomness. Therefore, there is room of improvement for this kind of method.

2 Methods

Here is the pipeline of INHERIT (see Figure 1). INHERIT uses two pre-trained models as references and we fine-tune them simultaneously to identify the metagenome sequences. The following will introduce the features of INHERIT by sections.

Two pre-trained models for INHERIT

INHERIT is a model based on DNABERT, a specific BERT model that modifies the way of tokenizing for DNA sequences. BERT (Devlin et al., 2018) stands for Bidirectional Encoder Representations from Transformers and has been widely used in the field of natural language processing, demonstrating the superiority and power of its structure. The success of BERT has also made the pre-train-fine-tune paradigm popular. Since BERT can be successful with human language, it is straightforward to think that for the language of cells and other biological tissues (i.e., the genome), BERT might be also useful. The feasibility of this assumption is demonstrated by DNABERT. It divides the DNA sequence into several tokens by the k-mers method, so that there will be a finite vocabulary and can be applied to BERT. Simultaneously, we enlarged the vocabulary of the DNABERT. We unified the degenerate bases as "N" and added them to our vocabulary to ensure fuller information of the sequence read-in. For example, for a sequence ATCKNTCG, its sequence using 6-mers segmentation is {ATCKNT, TCKNTC, CKNTCG}. The authors of DNABERT made pre-trained models with the human genome samples and achieved state-of-the-art in solving both the human genome and the mammalian genome sequences, again demonstrating that the structure of BERT can be used to solve genome-related problems. After experiments, we also found that DNABERT is more suitable for identifying phages than LSTM (see in Section 3.1).

We prepared two brand new pre-trained models for INHERIT. Based on the past experience, we divided the sequences into 500 bp-long segments and converted them into the form of 6-mers as input to DNABERT. What is more, since the number of bacteria we have known is much larger than the number of phages and the length of bacteria is also longer, 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 belonging to bacteria than phages. Thus, in order to make the phage information acquired by the pre-trained model can be equally weighted with the bacterial information and can contain as many sequences as possible, we generated two different pre-training sets for bacteria and phages and included two pre-trained models for the fine-tuning process.

The pre-trained model is an important part of INHERIT. One of the major drawbacks of the database-based methods is that when we need to identify a large number of sequences at the same time, it takes a long time to get the predictions due to the large file size. This can make it difficult for us to identify phages. However, this turns out to be the advantage of the alignment-free methods. Since alignment-free methods are usually based on deep learning models, they can usually take advantage of the current GPU computing and can perform the recognition and prediction of sequences faster. In proposing the MSA Transformer, Rao, Roshan, et al. (Rao et al., 2021) demonstrated that pre-trained models can have comparable performance to HMM Profiles and even better in some cases. In this case, the alignment-free methods can be combined with the database-based approaches by using the pre-train-fine-tune paradigm. In addition, the number of bacteria and phage segments should be balanced when we train the deep learning model on alignment-free methods, which results in a lim-

ited number of bacteria training samples. However, after using pre-train-fine-tune paradigm, the pre-training sets can be chosen large enough to reduce the limitations caused by the balance of the dataset in the downstream tasks. Thus, we chose the pre-trained models as the references of INHERIT. After our experiments, we demonstrate that the pre-trained models can bring some improvements to the prediction performance of the deep learning model (see in Section 3.2).

Both pre-trained DNABERT models have the default BERT structure, i.e., 12 hidden layers, 12 attention heads, and 768 embedding size, and since we included the degenerated bases as "N" compared to vanilla DNABERT, the vocabulary of our DNABERT is permutated by five letters (i.e., "A", "T", "C", "G", "N"), with a total of 16530 words. Both of them were trained on A100 GPUs. The unsupervised learning task used in the pre-training process of the models is the Masked Language Model.

Pre-training sets

For the pre-training sets, we wanted them to be as large as possible. 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 download the complete bacteria genome sequence from the NCBI FTP. The command we used was: ncbi-genome-download –formats fasta –assembly-levels complete bacteria, and sampled 4124 bacteria sequences from them randomly because of the limitation of the physical memory. However, these 4124 sequences can generate 15975346 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 26920 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 1750662 segments, and the size is still large for a phage dataset.

Input and output

Although many models have been proposed in recent years to work on this problem, there is no consistent input format and rules of identification. For example, VIBRANT is to first split the target sequence into segments of length 3kb to 15kb to simulate scaffolds for alignment. The final outputs of VIBRANT are also the predictions by each fragment. However, Seeker divides the target sequence into 1000 bp-long segments on average at first, but it offers the final predictions by each sequence. The input and output rules of INHERIT are similar to those of Seeker. Due to the limitation of DNABERT and past experience, first, the sequences should be 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. As discussed above, each segment should be converted to 6-mers format so that each segment is generated to a segment with 495 tokens (hereafter called "6-mers segments"). INHERIT gives each segment a prediction, and the prediction of the whole sequence is the average of the predictions of all the 6-mers 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: If we want INHERIT to predict the phages with higher confidence, for instance, we can adjust the threshold slightly larger.

Fine-tuning (training) process

The two pre-trained models are used during the fine-tuning process. Typically, researchers only use one pre-trained model for fine-tuning. However, if we fine-tune the two pre-trained models separately, then for a segment, two predictions will be generated. We then need to average these two predictions to get the final score for this segment. Nevertheless, it is difficult to make a precise decision on the weights used for averaging. Therefore, we design a network stucture that allows two pre-

Model	Precision	Recall	Accuracy	F_1-score	AUROC	AUPRC
LSTM	0.8430	0.8619	0.8214	0.8523	0.9199	0.9516
DNABERT (w/o pre-train)	0.9854	**0.9835**	0.9814	0.9844	0.9966	0.9978
INHERIT (w pre-train)	**0.9902**	**0.9835**	**0.9843**	**0.9868**	**0.9981**	**0.9987**

Table 1: The benchmark test for LSTM, DNABERT and INHERIT. On the first two rows of this table, we compare the performance of two different network structures: LSTM and DNABERT to show the feasibility of using DNABERT to identify phages and the reason we use the DNABERT-based model. The last two rows show the difference in the performance of DNABERT and INHERIT. One does not use any pre-trained models during training and the other uses two pre-trained models as references respectively.

trained models to simultaneously participate in a fine-tuning process and derive a prediction directly for each segment (see Figure 1B). We use the "BertForSequenceClassification" Function from Huggingface's Transformers (Wolf et al., 2020) to obtain the "logits" output from both models. After concatenating the "logits" outputs from the two models together, we derive the final classification results by a fully connected layer and output the prediction with a value range of 0 to 1 by using the Sigmoid function. The batch size of both training and validation sets is 64, and the learning rate is 10^{-5} for both pre-trained models, without weight decay and warmup. We also use early stopping based on validation accuracy. This strategy has also been used in the field of natural language processing, for example in the paper presented by Tay, Yi, et al. in 2020 (Tay et al., 2020). We also set the patience to 3, i.e., if the best validation accuracy does not rise in all 3 epochs, the training process will stop. The random seed is set to 6.

Training set and validation set

For the fine-tuning part, we hope INHERIT can learn more features to have better generalization ability. Thus, for the bacterial dataset, we randomly selected 260 bacteria sequences that were not in the pre-training set and the test set, of which 217 bacteria sequences were divided as the training set, generating 718879 segments, and the remaining 43 were used as the validation set, generating 188149 segments. However, for phages, we did not have as many sequences to choose, so we selected the phage sequences with higher quality from the pre-training set. We chose 13217 phage sequences that were not included in the test set by referring to the phage sequences selected by Seeker and VIBRANT, of which 10574 phage sequences were used as the training set, generating 718663 segments, and the remaining 2643 sequences were used as the validation set, generating 186121 segments.

Test set

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 1028 phage sequences. However, since there were 19 bacteria sequences removed from RefSeq, we used rest of them, including 691 bacteria sequences and 1028 phage sequences, to examine the performance of INHERIT for phage identification. It should be added that, in that article, the authors split the sequences in this test set into 1kb to 15kb segments on average, and predicted the results and calculated metrics in terms 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 in terms of sequences.

We have posted the accessions of all the sequences used in each dataset and their sources obtained in Supplement File 1.

3 Experiments

We mainly focus our experiments on the following three research questions:

1. For Seeker, they used LSTM to solve this problem. Thus, for LSTM and DNABERT, which is more suitable for our research?
2. Can add pre-trained models help the final performance of the deep learning models?
3. Compared to other methods, can our proposed INHERIT perform better and be easier to use, like VIBRANT and Seeker?

In this section, we first answer the 3 problems described above by the first 3 sections and explain the features of INHERIT in Section D.

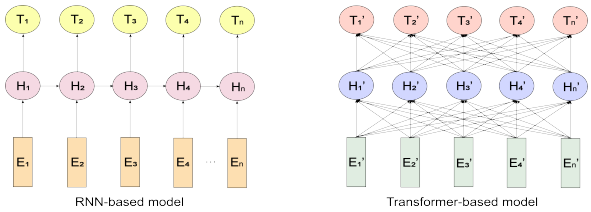

Figure 2: Figure 2 shows the difference between the network structure of RNN-based model and the one of Transformer-based model. Each hidden unit in the RNN-based model will learn from the previous hidden units, while the Transformer-based model will learn from every unit in the former layer all together. This difference in network structure results in different performance, and in most cases, the Transformer-based model performs better.

3.1 Evaluation of different network structures

Although Transformer-based pre-trained models are currently the most popular, pre-trained models based on other neural networks also exist. For example, ELMo(Peters et al., 2018) is a bi-directional LSTM-based pre-trained model. The proposal of Seeker demonstrates the feasibility of LSTM for this problem. However, no Transformer-based model has been applied to this problem before. Therefore, we first compare the performance of LSTM and the DNABERT we used on this problem to verify that it is feasible to solve this problem with DNABERT. For DNABERT, we used the same hyperparameters as INHERIT but did not add the pre-trained models. For the LSTM, we refer to the network structure of Seeker: a one-way LSTM structure with 5 hidden units. However, we used a dense layer and output a prediction with a value range of 0 to 1 by using the Sigmoid function, and the training batch size was 64, which was slightly different from Seeker. DNABERT used a learning rate of 10^{-5}, while LSTM used a learning rate of 10^{-3}. Both models used the random seed 6, the early stopping based on validation accuracy with the patience of 3, and ran on the same number of A100 GPUs. We measured their performance by making predictions on our test set, and the prediction procedure kept the same as INHERIT.

The evaluation metrics we chose are:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

$$\text{F}_1 - \text{score} = \frac{2 \cdot \text{TP}}{2 \cdot \text{TP} + \text{FP} + \text{FN}}$$

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

From the results (see the first two rows of Table 1) we can see that DNABERT performed significantly better than LSTM. The results implied that DNABERT and LSTM were both feasible for analyzing sequences, and DNABERT performed better. This is because the Transformer-based model, DNABERT, is more advantageous in terms of network structure compared to the LSTM we built, i.e., RNN-based model, which we constructed based on Seeker. This advantage is not only reflected in our experiments, but also in other fields, such as natural language processing, where Transformer-based models tend to perform better than RNN-based models. From the Figure 2, the RNN-based model is a unidirectional structure that obtains information from the previous hidden units. The Transformer-based model, however, is a fully connected, bidirectional network structure. Therefore, the RNN-based model can only obtain information from the previous units during training, while the Transformer-based model can obtain information from all units in the previous layer in parallel, which makes each hidden unit of the latter more comprehensive, and therefore can have better performance theoretically. Not only that, DNABERT is a model with a 12-layer Transformer structure, while our Seeker-based LSTM is a single-layer unidirectional network. Therefore, DNABERT is deeper and more complex, which is one of the rea-

sons why it can perform better. Thus, we finally chose DNABERT as the basis for INHERIT.

3.2 Use pre-trained models as the references to integrate two types of methods

The database-based approach has been the traditional way of solving such problems. The alignment-free approach, on the other hand, has gained much attention with the popularity of deep learning. As mentioned earlier, both of them have their advantages and disadvantages, but they have some complementary relationship with each other. The two can be combined by the pre-train-fine-tune paradigm. The pre-trained model learns features on a large dataset, which is used as a reference for downstream tasks to help train the model. In this experiment, we explored whether INHERIT would be improved compared to DNABERT with the help of two pre-trained models.

We used DNABERT from the previous experiment to compare with INHERIT. They set the same hyperparameters, except that INHERIT included two pre-trained models to help with training. Based on the results (see the last two rows of Table 1), the pre-training did help in the performance improvement of the model. Although the difference between the two is not that large in terms of results, this situation can be explained by the following reasons: First, DNABERT has already reached a high level of performance, and it would become much difficult to continue improving its performance. Second, although we used a pre-trained model, we used checkpoints from an earlier stage. If the pre-training process continues further, the pre-training may bring more improvements to INHERIT. Finally, our bacteria pre-trained model could not include more samples of bacteria, and we welcome more researchers to train pre-trained models with more samples, which will likely continue to improve the prediction performance.

However, from the results, INHERIT improved in most of the metrics compared with DNABERT, which proved out that pre-training would help for model prediction. When we fine-tuned the two pre-trained models simultaneously, we could finally obtain INHERIT, a unique integrated model that used the pre-trained model as references.

3.3 Benchmark test among Seeker, VIBRANT and INHERIT

In this section, we compared INHERIT with VIBRANT and Seeker, which are the representa-

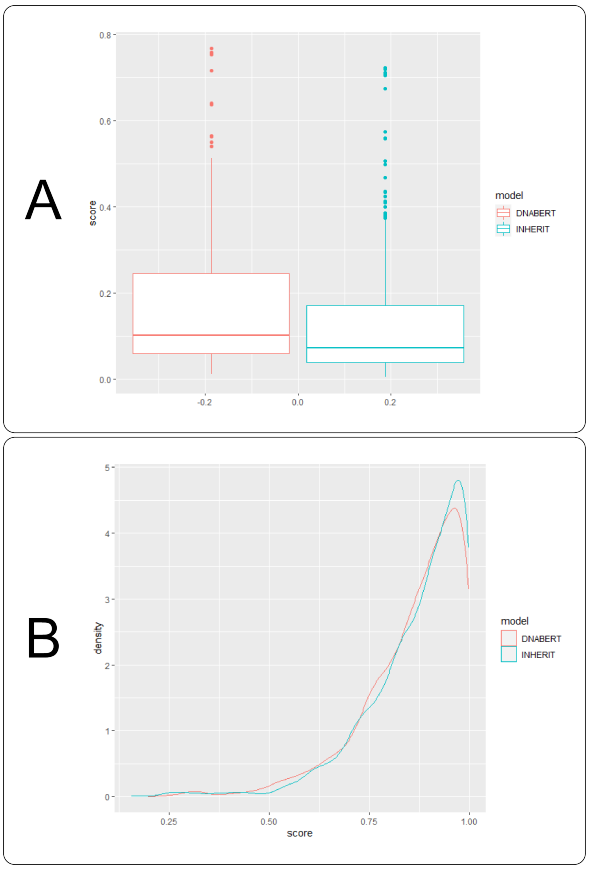

Figure 3: Figure 3A shows the boxplot of the scores of DNABERT and INHERIT for the bacteria samples in the test set. It is used to show the difference in the distribution of the predictions with or without pre-training. Figure 3B is the density plot of the scores of DNABERT and INHERIT for the phage samples in the test set. Since the recall of DNABERT and INHERIT are the same, the box plot cannot show the difference in the distribution of their predictions on phages. Thus, we use the density plot.

Model	Precision	Recall	Accuracy	F_1-score	AUROC	AUPRC
Seeker	0.9264	0.8453	0.8674	0.8840	0.9382	0.9605
VIBRANT	0.9541	**0.9903**	0.9656	0.9719	0.9595	0.9521
INHERIT	**0.9902**	0.9835	**0.9843**	**0.9868**	**0.9981**	**0.9987**

Table 2: The benchmark results of Seeker, VIBRANT and INHERIT. We used the default codes of hyperparameters for each model to run our test set. For VIBRANT, there were 3 sequences not given results by VIBRANT. Additionally, since we gave 1 as the score of predicting phages of VIBRANT while 0 as the score of predicting bacteria of VIBRANT, the AUROC and AUPRC of VIBRANT would be slightly underestimated by using this strategy.

tives of database-based methods and alignment-free methods respectively. Both of them have proven themselves to be one of the best of their respective types of methods. Since the input and output of these three methods for phage identification were inconsistent, we proposed several rules to make them unified.

The input and output rules of Seeker were similar to INHERIT, so we did not make any adjustments to it. Seeker would first divide the target sequence into 1000 bp-long segments and make predictions for each segment. The score of each sequence would be the average of its segments and the threshold was 0.5 by default. However, since VIBRANT is a multi-classifier and its outputs are in segments, identified as "organism", "plasmid" and "virus", we need to propose some strategies to make the results consistent.

First, we should define the prediction of VIBRANT for each sequence. We chose the prediction with the highest frequency in the segments to which the target sequence belonged as the prediction for this sequence. If there were two or more predictions with the highest frequency, we would randomly select one. For example, if the predictions of the segments of the target sequence are: "organism", "plasmid", "virus", we will regard the prediction of VIBRANT for this sequence as a virus.

Then, since we only identified bacteria and phage sequences, the sequence identified as "organism" and "plasmid" can be considered as "non-phage". In VIBRANT, we equated "non-phage" with the category "Bacteria", scoring with 0, and "virus" with "Phage", scoring with 1. However, these strategies would cause the AUROC and AUPRC of VIBRANT to be smaller. Additionally, there were 3 bacteria sequences not given results by VIBRANT and we did not include them in the calculation of the metrics of VIBRANT.

All of the models were used the default codes and hyperparameters to predict the test set. For Seeker and INHERIT, they ran on the same number of A100 GPUs, and for VIBRANT, it ran 3 CPUs, i.e. 108 cores for predictions, because VIBRANT cannot be accelerated with CUDA. The predictions of INHERIT, Seeker and VIBRANT for all the sequences in the test set can be seen in the Supplement File 2.

From the result (see Table 2), compared to Seeker and VIBRANT, INHERIT performed very outstandingly. Moreover, compared with Seeker and VIBRANT, the accuracy of INHERIT was an order of magnitude more precise. VIBRANT had a slightly higher recall than INHERIT, which meant VIBRANT may be more appropriate for identification in cases where it is known that there is a high proportion of phage sequences. However, in such cases, INHERIT is also competent: we only need to slightly lower the threshold of INHERIT to make the recall rise, and more phages can be identified as well. Even if we used the default threshold of 0.5, the recall of INHERIT did not differ much from that of VIBRANT. INHERIT is well ahead of Seeker and VIBRANT in F_1-score, AUROC and AUPRC. the prominence of F_1-score proves that INHERIT performs well enough when we use the default parameters and is competent for most application scenarios, which indicates that researchers do not need to consider other special factors to adjust the hyperparameters anymore. Second, the high AUROC and AUPRC indicate that INHERIT can better differentiate between phages and bacteria, which also reflects that researchers do not have to worry about large fluctuations in recall and precision due to threshold adjustment when they want to adjust the threshold value to make their research work more efficient. For example, when we want to identify more phages, we can reduce the threshold slightly, so that INHERIT can complete its task without adding many misidentified bacteria. INHERIT made the best performance on the rest of

the metrics, which implied that it could be applied in many more situations.

3.4 Analysis and summary

Based on our test results, it is not difficult to find some advantages of INHERIT, which we will explain specifically with some phage samples.

3.4.1 INHERIT learns features from pre-trained models

From the experiments in section 3.2, INHERIT is less likely to misclassify bacteria as phages than DNABERT which does not include any of the pre-trained models. For example, NZ_CP022939.1 has a score of 0.5624 in DNABERT, and since we have a threshold of 0.5, it is incorrectly identified as a phage even though it scored just a little bit over. However, INHERIT scores 0.4674 and is successfully identified as a bacterium. According to our analysis, this phenomenon is common. We drew a box plot of the scores of DNABERT and INHERIT for the bacteria samples in the test set (see Figure 3A). Based on this box plot, we can see that the score of INHERIT for these bacterial samples tends more towards 0 compared to DNABERT. The same phenomenon is observed for samples where both models are misidentified. For instance, NZ_CP028859.1 has a score of 0.6386 in DNABERT, but it drops to 0.5593 in INHERIT. In addition, although the recall of INHERIT does not rise compared to DNABERT, it does not mean that the pre-trained models cannot help the model to train the phages. The boxplot does not visualize this change, but it is obvious from the density plot (see Figure 3B) of the scores for the phage samples in the test set of both models. The high-score samples (score > 0.9) of INHERIT are more than those of DNABERT. These are solid evidence of the helpfulness of pre-trained models as references during the fine-tuning process of INHERIT.

3.4.2 INHERIT retains the strong features of the alignment-free methods

From the paper of Seeker, we can find that alignment-free methods (e.g. Seeker and DeepVirFinder) are not sensitive to the length of the target sequence, but the performance of database-based methods (e.g. VIBRANT and VirSorter) are affected by the length. For the phage samples in the test set, the phages for which VIBRANT formed an error identification in this test are all above 100000 bp in length. From our boxplot of the length of the

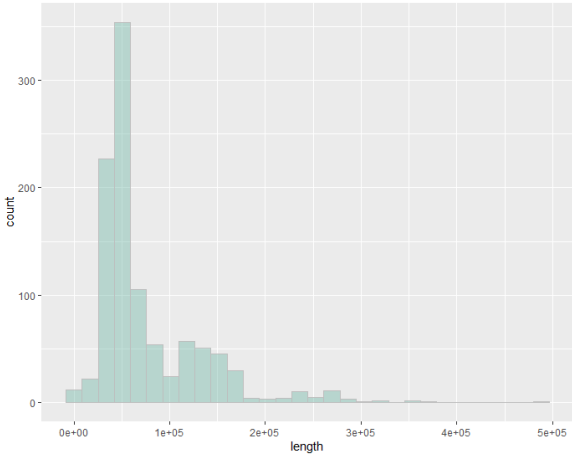

Figure 4: The histogram of the length of the phage samples in the test set. The lengths for most of the phages are lower than 100000 bp. The length of one phage being longer than 100000 bp may be regarded as the longer one in the test set.

phage samples in the test set (see Figure 4), these samples belong to the longer sequences. For example, the longest sequence in the phage samples was NC_042013.1, which reached 490380 bp and even exceeded the sequence length of some bacteria. Because of its excessive length, VIBRANT gave it high v-scores, but the score after normalization became too low, causing VIBRANT to judge it as a bacterium. Although there is a conflict on the glance with the conclusion in the paper of Seeker that VIBRANT performs worse for exceeded short sequences, however, in fact, both findings show the sensitivity of VIBRANT to the phage sequence length: if the phage sequence is too short, then it is difficult for this sequence to be compared to get a high score; if the phage sequence is too long, then it may be misidentified because the score after normalization is too low. However, since both Seeker and INHERIT first split the sequence into very short segments (1000 bp and 500 bp) for prediction, they are not influenced by the length of NC_042013.1 and they both successfully identify it as a phage (0.6665 and 0.8383).

However, at the same time, there is also a limitation of INHERIT: its recall is not optimal. It still does not perform as well as VIBRANT when we know that we need to predict a dataset with a relatively large proportion of phage sequences. Additionally, compared to DNABERT, the recall of INHERIT does not get more improvement. We assume that this may be because one of the pre-trained models we used consists of only phage sequences. Even though it also has millions of

Model name	Type	Input	Output	Commands	Threshold
Seeker	Alignment-free method	1000 bp-long segments	Asssessed over the sequences	seeker_fasta = SeekerFasta("sample.fasta") predictions = seeker_fasta.phage_or_bacteria()	0.5
VIBRANT	Database-based method	3-15kb fragments	Assessed over the fragments	python3 VIBRANT_run.py -i sample.fasta	None
INHERIT	Integrated method	500 bp-long segments	Asssessed over the sequences	python3 IHT_predict.py –sequence sample.fasta –withpretrain True –model INHERIT.pt –out sample_out.txt	0.5

Table 3: Table 3 shows the features of Seeker, VIBRANT and INHERIT. To make our benchmark test fair, all models used the default commands and hyperparameters to predict our datasets to simulate their performance in most application scenarios. All commands can be found in the GitHub pages of the respective models.

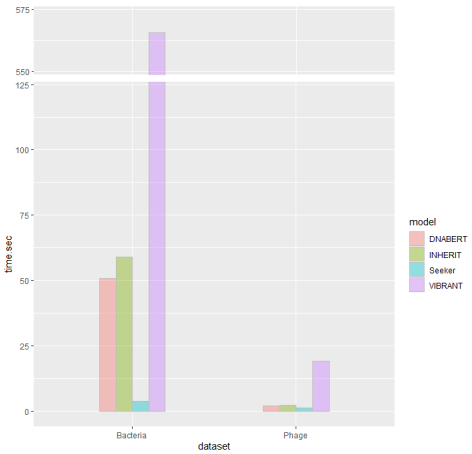

Figure 5: The average time required for Seeker, DNABERT, INHERIT, and VIBRANT to predict phage sequences and bacteria sequences in the test set. The time of each model implies their average running time (seconds) on predicting each bacterium and phage. The average length of bacteria samples on the test set is 3950500 bp, while the average length of phage samples on the test set is 75800 bp. Since the average running time of VIBRANT is too long, we cut off a part of the y-axis.

segments, it is still relatively small compared to the bacteria pre-trained model. However, from the dataset and HMM Profiles prepared by VIBRANT, we speculate that additional virus sequences can be added to this pre-trained model, which may solve the problem of the limited size and improve the prediction performance, which will be one of the areas of our future work.

3.4.3 Appropriate speed of prediction

Although INHERIT adds pre-trained models, this does not make INHERIT take a too long time to predict as the database-based approaches such as VIBRANT. We calculate the average time required for Seeker, DNABERT (i.e., without including any pre-trained models), INHERIT, and VIBRANT to predict phage sequences and bacteria sequences in the test set. However, VIBRANT cannot predict the entire bacteria test set at once, and the minimum time required for VIBRANT to predict one piece of them still reached 8 hours when we split the bacterial dataset into 8 pieces on average. The results (see Figure 5) show that VIBRANT's predictions take much more time than Seeker, DNABERT, and INHERIT. Although INHERIT takes a longer time to predict compared to Seeker, it is not much higher than DNABERT. This indicates that the time required for INHERIT prediction is mainly due to the usage of the Transformer-based model: pre-trained models do not have a large impact on the time required for the prediction of the model. This also shows that INHERIT can predict with high accuracy while being able to end the prediction of a large number of metagenome sequences in a reasonable amount of time

3.4.4 Summary of results

We made several experiments and analyses to solve our research problems. From Section 3.1, we found

DNABERT performed significantly than LSTM. That implies both LSTM and DNABERT show feasibility in identifying phages, while DNABERT is more suitable according to our task. That answers Problem 1.

Moreover, we also built DNABERT without any pre-trained models to compare with INHERIT. The results from Section 3.2 show that INHERIT achieves better performance on most of the metrics. We also made an analysis on Section 3.4.1 . Those show INHERIT takes the performance a step further from already good enough of DNABERT with the help of pre-trained models. Those answer Problem 2.

Most importantly, we tested the performance of INHERIT compared with VIBRANT and Seeker on a third-party benchmark dataset. Details of the usage and their detailed information of all models for this benchmark test are shown in Table 3. Since VIBRANT gives the predictions assessed over the fragments divided by the sequences, we proposed several rules to obtain the predictions over the sequences for VIBRANT. All the models were used the default hyperparameters and commands in our test. From Section 3.3, INHERIT performs the best in our test with an F1-score of 0.9868. According to our analysis in Section 3.4, INHERIT can not only learn the knowledge from the pre-trained models, which resembles database-based approaches but retain the features of alignment-free methods. This kind of integrated approach can just take the appropriate time to predict the sequences, and pre-trained models do not affect too much on the speed. Therefore, INHERIT performs better than VIBRANT and Seeker and it is easy to use. Those answer Problem 3.

Hence, based on answering the research problems, we summarize our contributions as described above.

4 Conclusions

In this work, we proposed 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 and Seeker, the state-of-the-art approaches. We demonstrated that INHERIT could achieve a better performance than the database-based method and alignment-free method alone. INHERIT improved the F1-score from 0.9719 to 0.9868. Meanwhile, we proved that using pre-trained models can help to improve the performance of phage identification further.

References

Noam Auslander, Ayal B Gussow, Sean Benler, Yuri I Wolf, and Eugene V Koonin. 2020. Seeker: Alignment-free identification of bacteriophage genomes by deep learning. *Nucleic acids research*, 48(21):e121–e121.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

K. Fujimoto, Y. Kimura, M. Shimohigoshi, T. Satoh, and S. Uematsu. 2020. Metagenome data on intestinal phage-bacteria associations aids the development of phage therapy against pathobionts. *Cell Host Microbe*, 28(3).

Jiarong Guo, Ben Bolduc, Ahmed A Zayed, Arvind Varsani, Guillermo Dominguez-Huerta, Tom O Delmont, Akbar Adjie Pratama, M Consuelo Gazitúa, Dean Vik, Matthew B Sullivan, et al. 2021. Virsorter2: a multi-classifier, expert-guided approach to detect diverse dna and rna viruses. *Microbiome*, 9(1):1–13.

Siu Fung Stanley Ho, Andrew D Millard, and Willem van Schaik. 2021. Comprehensive benchmarking of tools to identify phages in metagenomic shotgun sequencing data. *bioRxiv*.

Doug Hyatt, Gwo-Liang Chen, Philip F LoCascio, Miriam L Land, Frank W Larimer, and Loren J Hauser. 2010. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC bioinformatics*, 11(1):1–11.

Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. 2021. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*.

Nobuhiko Kamada, Yun-Gi Kim, Ho Pan Sham, Bruce A Vallance, José L Puente, Eric C Martens, and Gabriel Núñez. 2012. Regulated virulence controls the ability of a pathogen to compete with the gut microbiota. *Science*, 336(6086):1325–1329.

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

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

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

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Roshan Rao, Jason Liu, Robert Verkuil, Joshua Meier, John F Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. 2021. Msa transformer. *bioRxiv*.

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. *Quantitative Biology*, 8(1).

Yi Tay, Dara Bahri, Che Zheng, Clifford Brunk, Donald Metzler, and Andrew Tomkins. 2020. Reverse engineering configurations of neural text generation models. *arXiv preprint arXiv:2004.06201*.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.