Identifying genomic islands with deep neural networks

Rida Assaf[1*], Fangfang Xia[3,4], Rick Stevens[2,3]

1 Department of Computer Science, University of Chicago, Chicago, IL USA
2 The University of Chicago Consortium for Advanced Science and Engineering, University of Chicago, Chicago, IL, USA
3 Computing Environment and Life Sciences Division, Argonne National Laboratory, Lemont, IL, USA
4 Data Science and Learning Division, Argonne National Laboratory, Lemont, IL, USA

* rida@uchicago.edu

Abstract

Horizontal gene transfer is the main source of adaptability for bacteria, through which genes are obtained from different sources including bacteria, archaea, viruses, and eukaryotes. This process promotes the rapid spread of genetic information across lineages, typically in the form of clusters of genes referred to as genomic islands (GIs). Different types of GIs exist, often classified by the content of their cargo genes or their means of integration and mobility. Various computational methods have been devised to detect different types of GIs, but no single method currently is capable of detecting all GIs. We propose a method, which we call ShutterIsland, that uses a deep learning model (Inception V3, widely used in computer vision) to detect genomic islands. The intrinsic value of deep learning methods lies in their ability to generalize. Via a technique called transfer learning, the model is pre-trained on a large generic dataset and then re-trained on images that we generate to represent genomic fragments. We demonstrate that this image-based approach generalizes better than the existing tools. Using transfer learning and an image based approach, we were able to detect the most out of the correct GI predictions made by other tools, in addition to making novel GI predictions. The fact that the deep neural network was retrained on only a limited number of GI datasets and then successfully generalized indicates that this approach could be applied to other problems in the field where data is still lacking or hard to curate.

Introduction

Interest in genomic islands resurfaced in the 1990s, when some Escherichia coli strains were found to have exclusive virulence genes that were not found in other strains [1, 2]. These genes were thought to have been acquired by these strains horizontally and were referred to as pathogenicity islands (PAIs). Further investigations showed that other types of islands carrying other types of genes exist, giving rise to more names such as "secretion islands," "resistance islands," and "metabolic islands," since the genes carried by these islands could promote not only virulence but also symbiosis or catabolic pathways [3–5]. Aside from functionality, different names are also assigned to islands on the basis of their mobility. Some GIs are mobile and can thus move themselves to new hosts, such as conjugative transposons, integrative and conjugative elements (ICEs), and prophages, whereas other GIs lose their mobility [6, 7]. Prophages are viruses that infect bacteria and then remain inside the cell and replicate with the genome [8]. They

are also referred to as bacteriophages in some literature, constituting the majority of viruses, and outnumbering bacteria by a factor of 10 to 1 [9,10]. A genomic island (GI) then is a cluster of genes that is typically between 10 kb and 200 kb in length and has been transferred horizontally [11].

Horizontal gene transfer (HGT) may contribute to anywhere between 1.6% and 32.6% of the genomes [12–20]. This percentage implies that a major factor in the variability across bacterial species and clades can be attributed to GIs [21]. Thus, they impose an additional challenge to our ability to reconstruct the evolutionary tree of life. The identification of GIs is also important for the advancement of medicine, by helping develop new vaccines and antibiotics [22,23] or cancer therapies [24]. For example, knowing that PAIs can carry many pathogenicity genes and virulence genes [25–27], researchers found that potential vaccine candidates resided within PAIs [28].

While early methods focused on manual inspection of disrupted genomic regions that may resemble GI attachment sites [29] or show unusual nucleotide content [30,31], the most recent computational methods fall into two broad categories: methods that involve sequence composition and methods that involve comparative genomics [1]. They both focus on one or more of the features that make GIs distinct, such as compositional bias, mobility elements, and transfer RNA (tRNA) hotspots [5,7,11,25,26,32–34]. We discuss some of these features in more detail; they are listed by decreasing order of importance in [1,35].

- One of the most important features of GIs is that they are sporadically distributed; that is, they are found only in certain isolates from a given strain or species.
- Since GIs are transferred horizontally across lineages and since different bacterial lineages have different sequence compositions, measures such as GC content or, more generally, oligonucleotides of various lengths (usually 2–9 nucleotides) are being used [36–42]. Codon usage is a well-known metric, which is the special case of oligonucleotides of length 3.
- Since the probability of having outlying measurements decreases as the size of the region increases, tools usually use cut-off values for the minimum size of a region (or gene cluster) to be identified as a GI.
- Another type of evidence comes not from the attachment sites but from what is in between, since some genes (e.g., integrases, transposases, phage genes) are known to be associated with GIs [25].
- In addition to the size of the cluster, evidence from mycobacterial phages [43] suggests that the size of the genes themselves is shorter in GIs than in the rest of the bacterial genome. Different theories suggest that this may confer mobility or packaging or replication advantages [8].
- Some GIs integrate specifically into genomic sites such as tRNA genes, introducing flanking direct repeats. Thus, the presence of such sites and repeats may be used as evidence for the presence of GIs [45–47].

Other research suggests that the directionality of the transcriptional strand and the protein length are key features in GI prediction [8]. The available tools focus on one or more of the mentioned features.

Islander works by first identifying tRNA and transfer-messenger RNA genes and their fragments as endpoints to candidate regions, then disqualifying candidates through a set of filters such as sequence length and the absence of an integrase gene [3]. *IslandPick* identifies GIs by comparing the query genome with a set of related genomes selected by an evolutionary distance function [44]. It uses Blast and Mauve for the

genome alignment. The outcome heavily depends on the choice of reference genomes selected. *Phaster* uses BLAST against a phage-specific sequence database (the NCBI phage database and the database developed by Srividhya et al. [48]), followed by DBSCAN [49] to cluster the hits into prophage regions. *IslandPath-DIMOB* considers a genomic fragment to be an island if it contains at least one mobility gene, in addition to 8 or more consecutive open reading frames with di-nucleotide bias [50]. *SIGI-HMM* uses the Viterbi algorithm to analyze each gene's most probable codon usage states, comparing it against codon tables representing microbial donors or highly expressed genes, and classifying it as native or non-native accordingly [51]. *PAI-IDA* uses the sequence composition features, namely, GC content, codon usage, and dinucleotide frequency, to detect GIs [52]. *Alien Hunter* (or IVOM) uses k-mers of variable length to perform its analysis, assigning more weight to longer k-mers [38]. *Phispy* uses random forests to classify windows based on features that include transcription strand directionality, customized AT and GC skew, protein length, and abundance of phage words [8]. *Phage Finder* classifies 10 kb windows with more than 3 bacteriophage-related proteins as GIs [53]. *IslandViewer* is an ensemble method that combines the results of three other tools—SIGI-HMM, IslandPath-DIMOB, and IslandPick—into one web resource [54].

No single tool is able to detect all GIs in all bacterial genomes [31]. Methods that narrow their search to GIs that integrate under certain conditions, such as into tRNAs, miss out on the other GIs. Similarly, not all GI regions exhibit atypical nucleotide content [22, 55]. Evolutionary events such as gene loss and genomic rearrangement [5] present more challenges. Also, highly expressed genes (e.g., genes in ribosomal protein operons), or those having an island host and donor that belong to the same or closely related species, or the fact that amelioration would pressure even genes from distantly related genomes to adapt to the host over time would lead to the host and the island exhibiting similar nucleotide composition [56] and subsequently to false negatives [23]. Tools that use windows face difficulty in adjusting their size: small sizes lead to large statistical fluctuation, whereas larger sizes result in low resolution [57].

For comparative genomics methods, the outcomes depend strongly on the choice of genomes used in the alignment process. Very distant genomes may lead to false positives, and very close genomes may lead to false negatives. In general, the number of reported GIs may differ across tools, because one large GI is often reported as a few smaller ones or vice versa, also making it harder to detect end points and boundaries accurately, even with the use of hidden Markov models by tools such as AlienHunter and SIGI-HMM.

Furthermore, no reliable GI dataset exists that can validate the predictions of all these computational methods [38]. Although several databases exist, they usually cover only specific types of GIs [Islander, PAIDB, ICEberg], which would flag as false positives any extra predictions made by those tools. Moreover, as Nelson et al. state, "The reliability of the databases has not been verified by any convincing biological evidence" [6].

Results

Validation Following Previously Accepted Methods

In this section, we present a comparison of the tools' performance following definitions accepted by the scientific community and accepted as part of an earlier study [8].

We first present the total number of GI predictions made by each tool over the entire testing dataset(consists of 34 genomes and is described in more detail under the Material and Methods section) in Table 1.

We can see from Table 1 that AlienHunter calls the most GIs, with almost double

Table 1. Number of islands and their total base pair value predicted by each tool over the testing genomes dataset.

Tool	Number of Islands	Number of Base Pairs
ShutterIsland	649	10,700,492
Alien Hunter	1919	19,561,593
IslandViewer	701	10,571,974
IslandPath-Dimob	339	6,871,312
Phaster	109	4,334,225
Phispy	96	3,979,173
Phage Finder	85	3,656,950
IslandPick	362	3,020,733
SIGI-HMM	359	2,543,145
Islander	50	2,019,610

the amount called by ShutterIsland and IslandViewer if measured by base pair count, and even more if measured by island count. The number of predictions made by ShutterIsland is close to that made by IslandViewer, which is an ensemble method combining four other tools' predictions.

Table 2 presents a breakdown of the predictions into True Positives, False Positives, and True Negatives.

Table 2. Number of True Positives, False Positives, and False Negatives, as defined in the Phispy study, for predictions made by each tool over the entire testing dataset, comprised of 34 genomes.

Tool	True Positives(count)	False Positives(count)	False Negatives(count)
ShutterIsland	461	188	38
Alien Hunter	946	973	84
IslandViewer	491	210	66
IslandPath-Dimob	330	9	78
Phaster	109	0	40
Phispy	96	0	44
Phage Finder	85	0	44
IslandPick	200	162	120
SIGI-HMM	249	110	126
Islander	50	0	107

Tables 2 and 3 were constructed with the following definitions:

- A True Positive is a region predicted as a GI and:
  - Contains a phage-related gene, or
  - With at least 50% of its genes having unknown functions
- A False Positive is a region predicted as a GI but does not satisfy the above conditions
- A False Negative is a region with six consecutive phage-related genes that is not predicted as a GI

Phage-related genes are determined by keywords present in their annotations. The set of relevant keywords can be found in the supplementary material and is discussed in more detail in the Discussion section. Table 3 shows the True Positive Rate and the number of False Positives calculated using the aforementioned definitions, calculated on predictions made over the entire testing dataset.

Table 3. True Positive Rate(Sensitivity) and the number of False Positives, as defined in the Phispy study, for predictions made by each tool over the entire testing dataset, comprised of 34 genomes.

Tool	True Positive Rate(%)	False Positives(%)
ShutterIsland	92.4	29
Alien Hunter	91.8	50.7
IslandViewer	88.2	30
IslandPath-Dimob	80.9	2.7
Phaster	73.2	0
Phispy	68.6	0
Phage Finder	65.9	0
IslandPick	62.5	44.8
SIGI-HMM	66.4	30.6
Islander	31.8	0

ShutterIsland has the highest True Positive Rate, and the lowest number of False Negatives. While other tools report 0 False positives, they rank lower on the True Positive Rate metric, where the reason for both is that these tools make fewer predictions relative to the others.

Validation Using Novel Metrics

In this section, we adopt more general metrics to reflect a more objective cross-tool comparison.

Since every tool predicts a subset of all GIs, we capture the coverage of each tool across other tools' predictions in Table 4. We omit the tools we were not able to run, and use the default parameters for all the listed tools.

Table 4. Cross-tool comparison of GI results: % predicted GIs also predicted by other tools, over the testing genomes dataset.

Predictor \ Target	ShutterIsland	IslandViewer	Phispy	PhageFinder	Islander	Phaster	Alien Hunter	IslandPick	Dimob	SIGI
ShutterIsland	N/A	45.7%	97.8%	99.1%	67.4%	92.9%	27%	20.3%	54%	28.8%
IslandViewer	42.8%	N/A	89.3%	89.2%	N/A	82.1%	39.4%	N/A	N/A	N/A
Phispy	29.1%	23.7%	N/A	98.3%	52.8%	79.3%	9%	10.8%	29.1%	11.5%
PhageFinder	28.1%	23.6%	92.8%	N/A	50.4%	79.8%	9%	10.1%	29/3%	12%
Islander	9.2%	21.2%	23.7%	25.7%	N/A	22.2%	8.3%	15.5%	22.9%	17%
Phaster	26.4%	22.5%	82.4%	86%	44.5%	N/A	10.4%	11.3%	27.5%	12.7%
AlienHunter	56.8%	78.9%	87.2%	86.5%	98%	87.2%	N/A	67.1%	82.8%	92.6%
IslandPick	10.6%	43.3%	25.4%	28.7%	51.7%	29%	13.8%	N/A	28.4%	31.2%
Dimob	34.9%	70.5%	86.1%	85.2%	87.8%	76.9%	25.7%	29.5%	N/A	50.3%
SIGI	17.2%	47.7%	34.4%	31.6%	63.2%	27.8%	22.6%	30.9%	44.8%	N/A

Table 4 shows that AlienHunter has the biggest coverage in general when it comes to predicting other tools' results. This coverage is expected given that its predictions' base pair coverage is almost 10 times as much as the tools with least coverage. ShutterIsland comes next and predicts the most out of three tools' predictions. What is clear is the model's ability to generalize, considering that it was trained only on the intersection of the predictions made by Phispy and IslandViewer but nevertheless also got the most predictions for other tools such as PhageFinder and Phaster. Moreover, we note that specific tools such as Islander detect only a subset of the results, whereas the rest of the tools score somewhere in between.

Table 5 presents a breakdown of the percentage of islands with known GI features

that are discussed in detail in the Discussion section. Tools that use these features to perform their classifications were omitted. From Table 5, we can see that on average ShutterIsland is the tool with most predictions showing GI features being missed by other tools. It is also the tool that calls the most predictions showing GI features and misses the fewest such predictions made by other tools. So even though AlienHunter makes more predictions in general, more predictions made by ShutterIsland exhibit known GI features.

Table 5. Quality of tools predictions: % predictions made with GI features — % predictions missed with GI features, over the testing genomes dataset.

Predictor \ Target	ShutterIsland	IslandViewer	AlienHunter	IslandPick	SIGI	**Average**
ShutterIsland	N/A	91% — 64%	87% — 47%	89% — 31%	87% — 36%	**89% — 45%**
IslandViewer	94% — 67%	N/A	89% — 45%	80% — n/a	87% — n/a	**88% — 56%**
AlienHunter	74% — 70%	66% — 60%	N/A	73% — 21%	71% — 42%	**71% — 48%**
IslandPick	69% — 76%	34% — 86%	49% — 53%	N/A	54% — 44%	**52% — 65%**
SIGI	67% — 75%	45% — 77%	48% — 51%	50% — 35%	N/A	**53% — 60%**
Dimob	n/a — 66%	n/a — 28%	n/a — 43%	n/a — 25%	n/a — 23%	**n/a — 37%**
Phispy	n/a — 68%	n/a — 70%	n/a — 50%	n/a — 33%	n/a — 39%	**n/a — 52%**
PhageFinder	n/a — 68%	n/a — 70%	n/a — 50%	n/a — 34%	n/a — 39%	**n/a — 52%**
Islander	n/a — 75%	n/a — 71%	n/a — 51%	n/a — 33%	n/a — 39%	**n/a — 54%**
Phaster	n/a — 75%	n/a — 71%	n/a — 51%	n/a — 33%	n/a — 39%	**n/a — 54%**

Table 6 highlights the unique predictions made by each tool, and the percentage of those predictions that include GI features.

Table 6. Unique predictions made by each tool over the testing genomes dataset.

Tool	Unique (count)	Unique (bp)	with GI features (%)
Alien Hunter	1155	9,583,497	40%
Shutter Island	280	3,647,377	65%
Phaster	2	30,814	0%
Phispy	1	26,890	100%

Both ShutterIsland and AlienHunter have a lot of unique predictions, as is clear in Table 6. AlienHunter's unique predictions alone are almost more than every other tool's total predictions. They average 8 kbp in length. ShutterIsland's unique predictions are also more than most of the other tools' predictions, with an average length of 14 kbp. Applying the same length cutoff threshold (8 kbp) on AlienHunter's unique predictions reduces them to 301 islands with a total of 3,880,000 bp, which is on par with ShutterIsland's unique predictions. Unlike AlienHunter, however, most of ShutterIsland's unique predictions show GI features.

Materials and methods

PATRIC (the Pathosystems Resource Integration Center) is a bacterial bioinformatics resource center that we are part of (https://www.patricbrc.org) [58]. It provides researchers with the tools necessary to analyze their private data and to compare it with public data. PATRIC recently surpassed the 200,000 publicly sequenced genomes mark, ensuring that enough genomes are available for effective comparative genomics studies. PATRIC provides a compare region viewer service, where a query genome is aligned against a set of other related genomes anchored at a specific focus gene. The service finds other genes that are of the same family as the query gene and then aligns

their flanking regions accordingly. Such graphical representations are appealing because they help users visualize the genomic areas of interest. In the resulting plots, genomic islands should appear as gaps in alignment as opposed to conserved regions. We replicated the service by implementing an offline version forked from the production user interface, which is necessary for computational efficiency and for consistency in the face of any future interface changes.

Fig 1 shows sample visualizations of different genomic fragments belonging to the two classes. Each row represents a region in a genome, with the query genome being the top row. Each arrow represents a single gene, capturing its size and strand directionality. Colors represent functionality. The red arrow represents the query gene, at which the alignment with the rest of the genomes is anchored. The remaining genes share the same color if they belong to the same family; they are colored black if they are not found in the query genome's focus region. Some colors are reserved for key genes: green for mobility genes, yellow for tRNA genes, and blue for phage related genes.

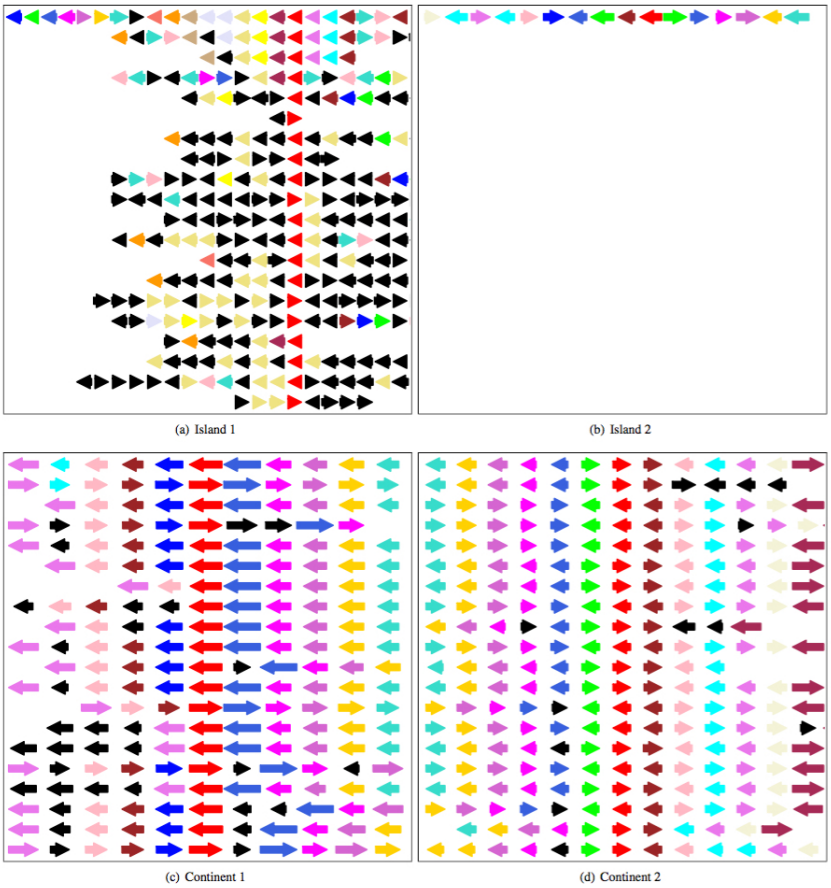


Fig 1. Examples of images generated from the compare region viewer. Each directed arrow represents a gene color coded to match its functionality. The first row is the genome neighborhood of the focus gene (red), and the subsequent rows represent anchored regions from similar genomes sorted by their phylogenetic distances to the query genome.

Figures 1a and 1b are examples of a query genome with a non-conserved neighborhood. The focus peg lacks alignments in general or is being aligned with genes from other genomes with different neighborhoods, containing genes with different

functionalities from those in the query genome (functionality is color coded). In contrast, Figures 1c and 1d show more conserved regions, which are what we expect to see in the absence of GIs(labelled as continents in the image).

This kind of representation also makes it easier to leverage the powerful machine learning (ML) technologies that have become the state of art in solving computer vision problems. Algorithms based on deep neural networks have proven to be superior in competitions such as the ImageNet Large Scale Visual Recognition Challenge (ILSVRS) [59]. Deep learning is the process of training neural networks with many hidden layers. The depth of these networks allows them to learn more complex patterns and higher-order relationships, at the cost of being more computationally expensive and requiring more data to work effectively. Improvements in such algorithms have been translated to improvements in a variety of domains reliant on computer vision tasks [60].

Our initial inspiration for representing genome features as images came from observing how human annotators work. These experts often examine the "compare region" view for a long time before they decide on the gene identity. A critical piece of information they rely on is how the focus gene compares with its homologs in related genomes. This information is cumbersome to represent in tabular data because (1) explicit all-to-all comparison is computationally expensive; (2) the comparisons need to be done at both individual gene and cluster levels including coordinates, length, and neighborhood similarities; and (3) human experts integrate all these different levels of information with an intuition for fuzzy comparison, something that is hard to replicate in tabular learning without additional parameterization or augmentation. Representing genomic features as images mitigates all three issues.

Images offer a natural way to compare genes (horizontally) and clusters across genomes (vertically) with 2D convolution. The fact that the compare region view sorts genomes by evolutionary distance allows the neural network to exploit locality and place more emphasis on close genomes via incremental pooling. An additional benefit of working with images is to be able to leverage the state-of-the-art deep learning models, many of which were first developed in vision tasks and perfected over years of iterations. Google researchers have used spectrograms (instead of text) in direct speech translation [61] and DNA sequence pile-up graphs (instead of alignment data) in genetic variant calling [62]. In both cases, the image-based models outperformed their respective previous state-of-the-art method based on traditional domain features. Further, the low-level visual feature patterns learned in pretrained image models have been demonstrated to transfer to distant learning tasks on non-image data in several preliminary studies ranging from environmental sound classification to cancer gene expression typing [63]. Much like feature-engineering methods, casting tabular data to images encodes information in a way more amenable to learning without explicitly adding information. It can also be easily integrated with other data modalities in the latent vector representation to prevent information loss. We hypothesize that this emerging trend of representing data with images will continue until model tuning and large-scale pretraining in scientific domains start to catch up with those in computer vision.

The images generated by PATRIC capture many of the most important GI features mentioned earlier—the sporadic distribution of islands, the protein length, functionality, and strand directionality—using color-coded arrows of various sizes. So, while PATRIC provides a lot of genomic data, the challenge comes down to building a meaningful training dataset. The databases available are still limited in size and specific in content, which in turn limits the ability even for advanced and deep models to learn and generalize well. Training deep models over a limited dataset puts the model at the risk of overfitting. One way around this problem is using a technique referred to as transfer learning [64]. In transfer learning, a model does not have to be trained from scratch.

Instead, the idea is to retrain a model that has been previously trained on a related task. The newly retrained model should then be able to transfer its existing knowledge and apply it to the new task. This approach affords the ability to reuse models that have been trained on huge amounts of data, while adding the necessary adjustments to make them available to work with more limited datasets, adding a further advantage to our approach of representing the data visually.

In our approach (Shutter Island), we use Google's Inception V3 architecture that has been previously trained on ImageNet. The Inception V3 architecture is a 48-layer-deep convolutional neural network [60]. Training such a deep network on a limited dataset such as the one available for GIs is unlikely to produce good results. ImageNet is a database that contains more than a million images belonging to more than a thousands categories [59]. The ImageNet project runs the ILSVRC dataset annually. The Inception V3 model reaches a 3.5% top-5 error rate on the 2012 ILSVRC dataset, where the winning model that year had a 15.3% error rate. Thus, a model that was previously trained on ImageNet is already good at feature extraction and visual recognition. To make the model compatible with the new task, the top layer of the network is retrained on our GI dataset, while the rest of the network is left intact, a strategy that is more powerful than starting with a deep network with random weights.

For our training data, we used the set of reference+representative genomes found on PATRIC. For each genome, our program produced an image for every non-overlapping 10 kbp window. A balanced dataset was then curated from the total set of images created. Since this is a supervised learning approach and our goal is to generalize over the tools' predictions and beyond, we used Phispy and IslandViewer's predictions to label the images that belong to candidate islands. IslandViewer captures the predictions of different methods that follow different approaches, and Phispy captures different GI features. Although the primary goal is to predict the union of the predictions of other tools and to generalize, we labeled a genomic fragment as a GI only if it belonged to the intersection of the predictions made by these tools. Doing so increases confidence that a certain candidate island is actually so. Overall, the intersection of these tools' predictions spanned only about half of the genome dataset. Our model reached a training accuracy of 92%, with a validation accuracy of 88%.

When treating novel genomes, the same compare genome viewer service was used, aligning the query genome with the set of reference+representative genomes. The only difference is that an image is created for every gene in the genome, providing better resolution than with non-overlapping windows. Next, each window is classified as either part of a GI or not. That label belongs to the focus gene in that window. Eventually, every gene in the genome has a label, and these are clustered into GIs with a minimum cut-off value of 8 kbp.

To test our model, since no reliable benchmark is available, we used the set of genomes mentioned in Phispy [8]. The set consists of 41 bacterial genomes that include 190 GIs. This set served as a good common ground for the comparison with the other tools. Some of those tools have not been updated for a while, but all the tools had predictions made over the genomes in this set. The authors of Phispy reported that the GIs have been manually verified [8]. We discarded the genomes that caused errors with any of the tools used in the comparison and any genomes that were part of the training set, and ended up with a total of 34 genomes out of the original 41(supplementary material).

In addition to using powerful technologies and extensive data, this approach may add an extra advantage over whole-genome alignment methods because performing the alignment over each gene may provide a higher local resolution and aid in resisting evolutionary effects such as recombination and others that may have happened after the integration and that usually affect GI detection efforts.

Discussion

The lack of experimentally verified ground-truth data-sets spanning the different types of GIs makes point-to-point comparison across the tools extremely challenging. Moreover, different tools follow different custom-defined metrics to judge their results, typically by using a threshold representing the minimum values of features (e.g., number of phage words) present in a region to be considered a GI, Which adds to the complications of validating GI predictions and comparing tools' performances. There is no consensus in the research community about what is accepted as a GI, and no clear benchmark that can be used for validation. Admittedly, judging the results is not a trivial task; otherwise the problem of detecting GIs would have been largely solved by the validation method itself. As a first step, we make use of one of the already established metrics that were previously used to validate GI predictions, by using the same definitions for True Positives, False Positives, and False Negatives as described in the Phispy study [8], and we use them to report the True Positive Rate(sensitivity), and the percentage of False Positives. Note that The True Negative Rate(and consequently, the F1 score) could not be generated due to the lack of a definition of a True Negative, as that would not be trivial. Also note that we introduce one change to the definitions stated in the Phispy study: where they define a true positive as a region with at least six phage related genes, we realize that the available datasets are much larger than what was accessible at the time of their publication, and thus the number six that was claimed to have been determined empirically may not be relevant anymore. We argue that if a region is suspected to be a GI, the mere presence of a phage related gene in that region adds to the confidence in its prediction as part of a GI. Moreover, we find that the database used by the study to determine phage functionality may be outdated, and we thus resort to certain keywords in the gene annotations generated by our system, PATRIC [34] to determine functionality. This is on par with Phaster's [21] validation method, whereby the presence of certain keywords(e.g, caspid) is used to verify predictions made by the tool. To identify these keywords, we scoured the literature and identified certain gene annotations that are related to GIs. Such annotations of gene identity are either directly curated by humans or reflect human assessment through exemplar-based computational propagation. We constructed a standard vocabulary of the GI-related keywords(supplementary material) that were also in agreement with the more extensive list of keywords used by Phaster [21] for the same purpose. With that said, we define a True Positive as a predicted region with at least one phage related gene, or where at least 50% of the genes are are unknown. A predicted region that does not satisfy these requirements is then considered a False Positive. We preserved the definition of a False Negative to be any region that contains at least six consecutive phage-related genes but is not predicted as a GI. It is worth emphasizing that while the presence of phage genes in a predicted region may add to the confidence of it being a GI, the absence of such genes from a predicted region does not necessarily make it a non-GI region. However, due to the limited fundamental understanding of GIs and the lack of a better alternative, we are adopting the absence of a phage related gene as an indicator of a False Positive. Following this metric that was established by another tool, our ShutterIsland scored the highest True Positive Rate and had the least False Negatives. In an attempt to present a comprehensive and more objective comparison, we also compare the tools' predictions based on a combination of three factors reflecting on the overlap and uniqueness of tool predictions and how well they capture gene annotation.

- First, we consider the percentage of each tool's results that every tool predicts. This gives us a measure of how much a single tool can generalize and how much of the union of all tools' predictions it can get.

- Second, since some tools make many more predictions than do others, we used the

GI-vocabulary mentioned earlier to get a better idea about the quality of the predictions made and missed by the other tools.

- Third, we investigate each tool's unique predictions and their quality also based on the presence of GI-keywords in their content.

Another challenge in assessing GI prediction is getting precise endpoints for predicted islands. Since different tools report a different number of islands owing to the nature of the features they use, where one island could be reported as many or vice versa, we considered a tool to predict another's islands if any of its predictions overlap with the other tool's predictions. We counted the percentage of base pair coverage of that other tool as represented by its predicted endpoints. This allowed us to compare overlapping islands predicted by different tools even if their coordinates did not match.

Fig 2 shows the receiver operating characteristic (ROC) curve of our classifier.

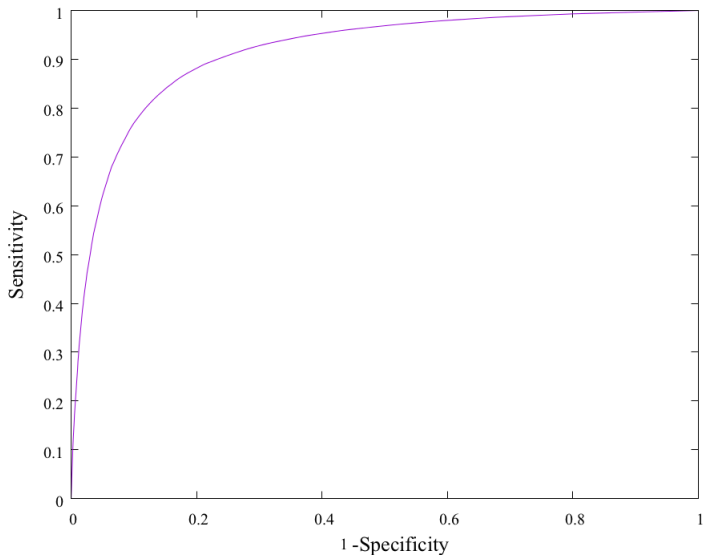


Fig 2. ROC curve for our genomic island binary classifier. The ROC curve plots the true positive rate as a function of the false positive rate. The greater the area under the curve is (the closer it is to the unrealistic ideal top left corner point), the better.

To construct the ROC curve, we used the following more general definitions. We considered the four surrounding genes on each side of every gene, and labeled each such region as:

- A True Positive, if predicted as a GI and:
  - Includes a phage-related gene, or
  - Overlaps with a prediction made by another tool
- A False Positive, if predicted as a GI but does not satisfy the above conditions
- A True Negative, if not predicted as a GI and does include a phage-related gene
- A False Negative , if not predicted as a GI but includes a phage-related gene

To qualitatively assess our model's ability to generalize, we investigate the unique predictions made by ShutterIsland, and present snapshots of the cargo genes typically found in these unique predictions in Fig 3, which shows that a significant number of the included genes carry GI related annotations.

Example 1:

-Mobile element protein
-Phage tail fiber assembly protein
-Phage tail fiber protein
-Tail fiber assembly protein
-Phage tail fiber protein
-Putative phage tail protein
-Putative phage tail protein
-Phage minor tail protein
-Putative phage tail protein
-putative membrane protein
-hypothetical protein

Example 2:

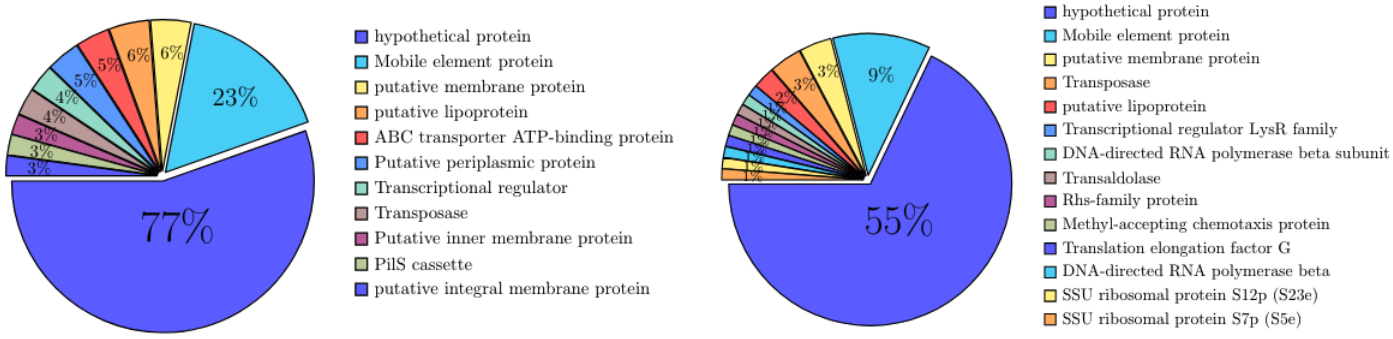
-site-specific recombinase
-Protein translocase subunit SecF
-Protein translocase subunit SecD
-hypothetical protein
-Preprotein translocase subunit YajC (TC 3.A.5.1.1)
-tRNA-guanine transglycosylase (EC 2.4.2.29)
-S-adenosylmethionine:tRNA ribosyltransferase-isomerase
-Transcriptional regulator, AsnC family
-L-lysine 6-aminotransferase
-hypothetical protein
-FIG01209928: hypothetical protein
-FIG01212260: hypothetical protein
-FIG01213367: hypothetical protein
-Mobile element protein
-Mobile element protein
-putative ISXo8 transposase
-Mobile element protein
-Mobile element protein

Example 3:

-IncF plasmid conjugative transfer pilus assembly protein TraB
-IncF plasmid conjugative transfer pilus assembly protein TraK
-IncF plasmid conjugative transfer pilus assembly protein TraE
-IncF plasmid conjugative transfer pilus assembly protein TraL
-IncF plasmid conjugative transfer pilin protein TraA
-hypothetical protein
-Mobile element protein
-IncF plasmid conjugative transfer protein TraD
-hypothetical protein
-IncF plasmid conjugative transfer DNA-nicking and unwinding protein TraI
-IncF plasmid conjugative transfer pilin acetylase TraX
-hypothetical protein
-FinO, putative fertility inhibition protein
-27kDa outer membrane protein
-endonuclease
-Replication regulatory protein repA2 (Protein copB)
-DNA replication protein
-hypothetical protein
-Putative transposase
-Mobile element protein

Fig 3. Examples of regions uniquely predicted by ShutterIsland

To have a more general examination of the unique predictions, we present the most common gene annotations found in these predictions made by ShutterIsland and AlienHunter in Fig 4. We only include these tools since they were the only tools with a significant number of unique predictions to perform the analysis on. We notice that the most frequent genes that are common to these predicted regions are either of unknown functionality or resemble a GI-related functionality, which adds to our confidence in these predictions.



(a) Breakdown of ShutterIsland's unique predictions.

(b) Breakdown of AlienHunter's unique predictions.

Fig 4. Most common gene annotations found in the unique predictions made by ShutterIsland and AlienHunter, with the percentage of unique predictions they reside in.

Authors' contributions

R.A. carried out the implementation and wrote the manuscript. R.S. and F.X. were involved in planning and supervised the work. All authors aided in interpreting the results. All authors provided critical feedback and commented on the manuscript.

Acknowledgments

We thank Dr. James J. Davis for constructive criticism of the manuscript, and Dr. Gail W. Pieper for editing the manuscript.

Funding

PATRIC has been funded in whole or in part with Federal funds from the National Institute of Allergy and Infectious Diseases, National Institutes of Health, Department of Health and Human Services [HHSN272201400027C]. Funding for open access charge: Federal funds from the National Institute of Allergy and Infectious Diseases, National Institutes of Health, Department of Health and Human Services [HHSN272201400027C]. The funding body had no direct role in the design of the study nor the collection, analysis, and interpretation of data and in writing the manuscript.

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

The datasets generated and/or analyzed during the current study are available in the GitHub repository, https://github.com/ridassaf/ShutterIsland.

References

1. Langille, M., Hsiao, W., Brinkman, F. (2010) Detecting genomic islands using bioinformatics approaches. *Nature Reviews Microbiology*, **8(5)**, 373–382.

2. Hacker, J., et al. (1990) Deletions of chromosomal regions coding for fimbriae and hemolysins occur in vitro and in vivo in various extraintestinal Escherichia coli isolates. *Microb. Pathog*, **8**, 213—225.

3. Hudson, C., Lau, B., Williams, K. (2014) Islander: a database of precisely mapped genomic islands in tRNA and tmRNA genes. *Nucleic Acids Research*, **43(D1)**, D48–D53.

4. Barondess, J.J., Beckwith, J. (1990) A bacterial virulence determinant encoded by lysogenic coliphage lambda. *Nature*, **346**, 871—874.

5. Dobrindt, U., Hochhut, B., Hentschel, U., Hacker, J. (2004) Genomic islands in pathogenic and environmental microorganisms. *Nat. Rev. Microbiol.*, **2**, 414—424.

6. Lu, B., Leong, H. (2016) Computational methods for predicting genomic islands in microbial genomes. *Computational And Structural Biotechnology Journal*, **14**, 200–206.

7. Juhas, M., van der Meer, J.R., Gaillard, M., Hood, D.W., et al. (2009) Genomic islands: tools of bacterial horizontal gene transfer and evolution. *FEMS Microbiol. Rev.*, **33**, 376—3793.

8. Akhter, S., Aziz, R., Edwards, R. (2012) PhiSpy: a novel algorithm for finding prophages in bacterial genomes that combines similarity- and composition-based strategies. *Nucleic Acids Research*, **40(16)**, e126–e126.

9. Fogg, P., Colloms, S., Rosser, S., Stark, M., Smith, M. (2014) New applications for phage integrases. *Journal of Molecular Biology*, **426(15)**, 2703–2716.

10. Hambly, E., Suttle, C.A.. (2005) The viriosphere, diversity, and genetic exchange within phage communities. *Curr. Opin. Microbiol.*, **8**, 444-–50.

11. Hacker, J., Kaper, J.B. (2000) Pathogenicity islands and the evolution of microbes. *Annu. Rev. Microbiol.*, **54**, 641—679.

12. Liu, H., Zhu, J. (2010) Analysis of the 3' ends of tRNA as the cause of insertion sites of foreign DNA in Prochlorococcus. *Journal Of Zhejiang University SCIENCE B*, **11(9)**, 708–718.

13. Nelson, K.E., et al. (1999) Evidence for lateral gene transfer between archaea and bacteria from genome sequence of Thermotoga maritime. *Nature*, **399(6734)**, 323-329.

14. Garcia-Vallve, V.S., Romeu, A., Palau, J. (2000) Horizontal gene transfer in bacterial and archaeal complete genomes. *Genome Res.*, **10(11)**, 1719–1725.

15. Koonin, E.V., Makarova, K.S., Aravind, L. (2001) Horizontal gene transfer in prokaryotes: quantification and classification. *Annu. Rev. Microbiol.*, **55(1)**, 709-742.

16. Nakamura, Y., Itoh, T., Matsuda, H., Gojobori, T., (2004) Biased biological functions of horizontally transferred genes in prokaryotic genomes. *Nat. Genet.*, **36(7)**, 760–766.

17. Choi, I.G., Kim, S.H., (2007) Global extent of horizontal gene transfer. *PNAS*, **104(11)**, 4489–4494.

18. Casjens, S. (2003) Prophages and bacterial genomics: what have we learned so far? *Mol. Microbiol.*, **49**, 277—300.

19. Casjens, S., Palmer, N., van Vugt, R., Huang, W.M., Stevenson,B., Rosa, P., Lathigra, R., Sutton, G., Peterson, J., Dodson, R.J. et al. (2000) A bacterial genome in Fux: the twelve linear and nine circular extrachromosomal DNAs in an infectious isolate of the Lyme disease spirochaete Borrelia burgdorferi. *Mol. Microbiol.*, **35**, 490-–516.

20. Canchaya, C., Proux, C., Fournous, G., Bruttin, A., Brussow, H. (2003) Prophage genomics. *Microbiol. Mol. Biol. Rev.*, **67**, 238-–276.

21. Arndt, D., Grant, J., Marcu, A., Sajed, T., Pon, A., Liang, Y., & Wishart, D. (2016) PHASTER: a better, faster version of the PHAST phage search tool. *Nucleic Acids Research*, **44(W1)**, W16–W21.

22. Zhou, Y., Liang, Y., Lynch, K., Dennis, J., & Wishart, D. (2011) PHAST: a fast phage search tool. *Nucleic Acids Research*, **39**, W347–W352.

23. Coates, A.R., Hu, Y. (2007) Novel approaches to developing new antibiotics for bacterial infections. *Br. J. Pharmacol.*, **152**, 1147—1154.

24. Bar, H., Yacoby, I., Benhar,I. (2008) Killing cancer cells by targeted drug-carrying phage nanomedicines. *BMC Biotechnol.*, **8**, 37.

25. Hacker, J., Blum-Oehler, G., Muhldorfer, I., Tschape, H. (1997) Pathogenicity islands of virulent bacteria: structure, function and impact on microbial evolution. *Mol. Microbiol.*, **23**, 1089—1097.

26. Schmidt H, Hensel M. (2004) Pathogenicity Islands in bacterial pathogenesis. *Clin. Mcrobiolog. Rev.*, **17**, 14—56.

27. Ho Sui, S.J,, Fedynak, A., Hsiao, W.W.L., Langille, M.G.I., Brinkman, F.S.L. (2009) The association of virulence factors with genomic islands. *PLoS One*, **4**, e8094.

28. Moriel, D.G., Bertoldi, I., Spagnuolo, A., Marchi, S., Rosini, R., et al. (2010) Identification of protective and broadly conserved vaccine antigens from the genome of extraintestinal pathogenic Escherichia coli. *Proc. Natl. Acad. Sci. U S A*, **107**, 9072-–9077.

29. Fouts, D.E. (2004) Bacteriophage bioinformatics. In Fraser, C.M., Read, T.D., Nelson, K.E. (eds), *Microbial Genomes.* Humana Press Inc., Totowa, NJ, pp. 71—91.

30. Nicolas, P., Bize, L., Muri, F., Hoebeke, M., Rodolphe,F ., Ehrlich, S.D., Prum, B., Bessières,P. (2002) Mining Bacillus subtilis chromosome heterogeneities using hidden Markov models. *Nucleic Acids Research*, **30**, 1418—1426.

31. Srividhya, K., Alaguraj, V., Poornima, G., Kumar, D., Singh, G.P., Raghavenderan, L., Katta, M., Mehta, P., Krishnaswamy, S. (2007) Identification of prophages in bacterial genomes by dinucleotide relative abundance difference. *PLoS One*, **2**, e1193.

32. Gogarten, J.P., Townsend, J.P.. (2005) Horizontal gene transfer, genome innovation and evolution. *Nat. Rev. Microbiol.*, **3**, 679-–687.

33. Soucy, S.M., Huang, J., Gogarten, J.P. (2015) Horizontal gene transfer: building the web of life. *Nat. Rev. Genet.*, **16**, 472–482.

34. Hacker, J., Bender, L., Ott, M., Wingender, J., Lund, B., et al. (1990) Deletions of chromosomal regions coding for fimbriae and hemolysins occur in vitro and in vivo in various extra intestinal Escherichia coli isolates. *Microb. Pathog.*, **8**, 213-–225.

35. Vernikos, G. S., Parkhill, J. (2008) Resolving the structural features of genomic islands: a machine learning approach. *Genome Res.*, **18**, 331—342.

36. Greub, G., Collyn, F., Guy, L., Roten, C.A. (2004) A genomic island present along the bacterial chromosome of the Parachlamydiaceae UWE25, an obligate amoebal endosymbiont, encodes a potentially functional F-like conjugative DNA transfer system. *BMC Microbiology*, **4(1)**, 48

37. Lawrence, J.G., Ochman, H. (1998) Molecular archaeology of the Escherichia coli genome. *Proceedings of the National Academy of Sciences*, **95(16)**, 9413—9417.

38. Vernikos, G. S., Parkhill, J. (2006) Interpolated variable order motifs for identification of horizontally acquired DNA: revisiting the Salmonella pathogenicity islands. *Bioinformatics*, **22**, 2196—2203.

39. Karlin, S., Mrazek, J. & Campbell, A. M. (1998) Codon usages in different gene classes of the Escherichia coli genome. *Mol. Microbiol.*, **29**, 1341—1355.

40. Karlin, S. (2001) Detecting anomalous gene clusters and pathogenicity islands in diverse bacterial genomes. *Trends Microbiol.*, **9**, 335—343.

41. Sandberg, R., et al. (2001) Capturing whole-genome characteristics in short sequences using a naive Bayesian classifier. *Genome Res.*, **11**, 1404—1409.

42. Tsirigos, A., Rigoutsos, I. (2005) A new computational method for the detection of horizontal gene transfer events. *Nucleic Acids Research*, **33**, 922—933.

43. Hatfull, G.F., Jacobs-Sera, D., Lawrence, J.G., Pope, W.H., Russell, D.A., Ko, C.C., Weber, R.J., Patel, M.C., Germane, K.L., Edgar, R.H., et al. (2010) Comparative genomic analysis of 60 mycobacteriophage genomes: genome clustering, gene acquisition, and gene size. *J. Mol. Biol.*, **397**, 119—143.

44. Langille, M.G., Hsiao, W.W., Brinkman, FS. (2008) Evaluation .of genomic island predictors using a comparative genomics approach. *BMC Bioinformatics*, **9**, 329.

45. Williams, K.P. (2002) Integration sites for genetic elements in prokaryotic tRNA and tmRNA genes: sublocation preference of integrase subfamilies. *Nucleic Acids Res.*, **30**, 866-–875.

46. Reiter, W.D., Palm, P., Yeats, S. (1989) Transfer RNA genes frequently serve as integration sites for prokaryotic genetic elements. *Nucleic Acids Research*, **17**, 1907-–1914.

47. Bellanger, X., Payot, S., Leblond-Bourget, N., Guedon, G. (2014) Conjugative and mobilizable genomic islands in bacteria: evolution and diversity. *FEMS Microbiol. Rev.*, **38**, 720—760.

48. Srividhya, K.V., Rao, G.V., Raghavenderan, L., Mehta, P., Prilusky,J., Manicka,S., Sussman, J.L., Krishnaswamy, S. (2006) Database and comparative identification of prophages. In: Huang,D-S, Li,K and Irwin,GW (eds). *Intelligent Control and Automation, Lecture Notes in Control and Information Sciences.* Springer, Berlin, Vol. 344, pp. 863—868.

49. Ester, M., Kriegel, H., Sander, J., Xu, X. (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: *KDD-1996 Proceedings* AAAI Press, Menlo Park, pp. 226—231.

50. Hsiao, W., Wan, I., Jones, S.J., et al. (2003) IslandPath: aiding detection of genomic islands in prokaryotes. *Bioinformatics*, **19(3)**, b418—420.

51. Waack, S., Keller, O. Asper, R., et al. (2006) Score-based prediction of genomic islands in prokaryotic genomes using hidden Markov models. *BMC Bioinformatics*, **7**, 142.

52. Tu, Q., Ding, D. (2003) Detecting pathogenicity islands and anomalous gene clusters by iterative discriminant analysis. *FEMS Microbiol. Lett.*, **221**, 269—275.

53. Fouts, D. (2006) Phage_Finder: automated identification and classification of prophage regions in complete bacterial genome sequences. *Nucleic Acids Res.*, **34**, 5839-–5851.

54. Langille, M.G., Brinkman, F. IslandViewer: an integrated interface for computational identification and visualization of genomic islands. *Bioinformatics*, **25**, 664-–665.

55. Nelson, K.E., Weinel, C., Paulsen, I.T., Dodson, R.J., Hilbert, H., Martins dos Santos, V.A., Fouts, D.E., Gill, S.R., Pop, M., Holmes, M. et al. (2002) Complete genome sequence and comparative analysis of the metabolically versatile Pseudomonas putida KT2440. *Environ. Microbiol.*, **4**, 799—808.

56. Lawrence, J. G., Ochman, H. (1997) Amelioration of bacterial genomes: rates of change and exchange. *J. Mol. Evol.*, **44**, 383—397.

57. Zhang, R., Zhang, C.T. (2004) A systematic method to identify genomic islands and its applications in analyzing the genomes of Corynebacterium glutamicum and Vibrio vulnificus CMCP6 chromosome I. *Bioinformatics*, **20(5)**, 612—622.

58. Wattam, A.R, ZDavis, J.J., Assaf, R., Boisvert, S., Bun, T., Conrad, N., Dietrich, E.M., Disz, T., Gabbard, J.L., Gerdes, S., Henry, C.S., Kenyon, R.W., Machi, D., Mao, C., Nordberg, E.K., Olsen, G.J., Murphy-Olson, D.E., Olson, R., Overbeek, R., Parrello, B., Pusch, G.D., Shukla, M., Vonstein, V., Warren, A., Xia, F., Yoo, H., Stevens, R.L. (2017) Improvements to PATRIC, the all-bacterial Bioinformatics Database and Analysis Resource Center. *Nucleic Acids Research* **45(D1)**, D535–D542.

59. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.. (2015) ImageNet large scale visual recognition challenge. *IJCV*.

60. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z. (2016) Rethinking the Inception Architecture for Computer Vision. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2818–2826.

61. Jia, Y., Weiss, R.J., Biadsy, F., Macherey, W., Johnson, M., Chen, Z. Wu, Y., (2019). Direct speech-to-speech translation with a sequence-to-sequence model. arXiv preprint arXiv:1904.06037.

62. Poplin, R., Chang, P., Alexander, D., Schwartz, S., Colthurst, T., & Ku, A. et al. (2018) A universal SNP and small-indel variant caller using deep neural networks. *Nature Biotechnology*

63. Howard, J. (2019) Lesson 2: Deep Learning 2019 - Data cleaning and production; SGD from scratch. Retrieved from https://www.youtube.com/watch?v=ccMHJeQU4Qw

64. How to Retrain an Image Classifier for New Categories - TensorFlow Hub — TensorFlow. (2018) Retrieved from https://www.tensorflow.org/hub/tutorials/image_retraining