A contextualised protein language model reveals the functional syntax of bacterial evolution

Maciej Wiatrak[†1,2], Ramon Viñas Torné[3], Maria Ntemourtsidou[1,2], Adam Dinan[1,2], David C. Abelson[1,2,6], Divya Arora[2], Maria Brbić[3,4,5], Aaron Weimann[1,2†], R. Andres Floto[†1,2,6]

[1]Cambridge Centre for AI in Medicine, University of Cambridge. Cambridge, UK
[2]Victor Phillip Dahdaleh Heart & Lung Research Institute, Department of Medicine, University of Cambridge, Cambridge, UK
[3]School of Computer and Communication Sciences, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland
[4]School of Life Sciences, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland
[5]Swiss Institute of Bioinformatics, Lausanne, Switzerland
[6]Royal Papworth Hospital NHS Foundation Trust, Cambridge, UK

[†]Corespondence to M.W. (mw896@cam.ac.uk), A.W. (aw807@cam.ac.uk) or R.A.F. (arf27@cam.ac.uk)

Bacteria have evolved a vast diversity of functions and behaviours which are currently incompletely understood and poorly predicted from DNA sequence alone. To understand the syntax of bacterial evolution and discover genome-to-phenotype relationships, we curated over 1.3 million genomes spanning bacterial phylogenetic space, representing each as an ordered sequence of proteins which collectively were used to train a transformer-based, contextualised protein language model, *Bacformer*. By pretraining the model to learn genome-wide evolutionary patterns, *Bacformer* captures the compositional and positional relationships of proteins and can accurately: predict protein-protein interactions, operon structure (which we validated experimentally), and protein function; infer phenotypic traits and identify likely causal genes; and design template synthethic genomes with desired properties. Thus, *Bacformer* represents a new foundation model for bacterial genomics that provide biological insights and a framework for prediction, inference, and generative tasks.

Introduction

Bacteria are arguably the most successful organisms on Earth, having evolved over millions of years to inhabit diverse geographically environments [1], biological niches [2, 3] and ecosystems [4]; acquiring a vast metabolic repertoire to exploit diverse substrates to generate energy and create biomass [5], and developing complex behavioural traits such as virulence [6], immune evasion [7], and antibiotic resistance [8].

Bacterial genomes are known to encode these adaptations in a highly structured manner. For example: functionally related genes are usually found contiguously in operons [9]; transcription factors mostly regulate proximal rather than distant genes [10, 11]; and the genomic distance from the origin of replication generally predicts gene identity [12, 13]. Thus, the architecture of genomes may provide a framework to create informative representations of bacteria that could be used to understand the basis of evolution and enable generalisable functional predictions across taxa.

While previous large-scale analyses of bacterial genomes have been focused on specific applications (such as biosynthetic gene cluster discovery [14, 15] or predicting antimicrobial resistance [16, 17, 18]), and have been hampered by biases in public genome collections towards a few pathogenic bacteria [19, 20], we sought to develop a foundation model of bacteria that captures genomic structure and function across the full spectrum of bacterial diversity. To do this, we leveraged and curated existing diverse collections of metagenome-assembled genomes (MAGs) of bacteria [21, 22] derived from environmental sequencing across diverse habitats totalling over 1.3M MAGs and encoding almost 3 billion proteins, to create a scalable and data-driven foundation model of bacterial life.

We represent individual bacterial genomes as ordered sequences of proteins and use these as inputs to train a transformer-based, contextualised protein language model (pLM), called *Bacformer* (combining 'bacteria' and 'transformer'), which learns the functional organisation of bacterial genomes. By pretraining the model to learn genome-wide evolutionary patterns, *Bacformer* captures the compositional and positional relationships of proteins and excels at fundamental functional genomic tasks - including predicting protein-protein interactions operon identification (which we validated experimentally), determining gene essentiality, predicting protein function, and accurately inferring phenotypic traits - as well as generative modelling, designing template synthethic genomes with desired properties.

Results

Training a genome-contextualised protein language model

To develop a foundation model that captures evolutionary processes across bacteria, we first curated a broad taxonomic corpus of over 1.3 million metagenome-assembled genomes (MAGs) from over 25,000 species, spanning almost 1,500 orders and more than 70 biomes (Fig. 1A).

Every individual assembled genome was then modelled as an ordered sequence of proteins, each of

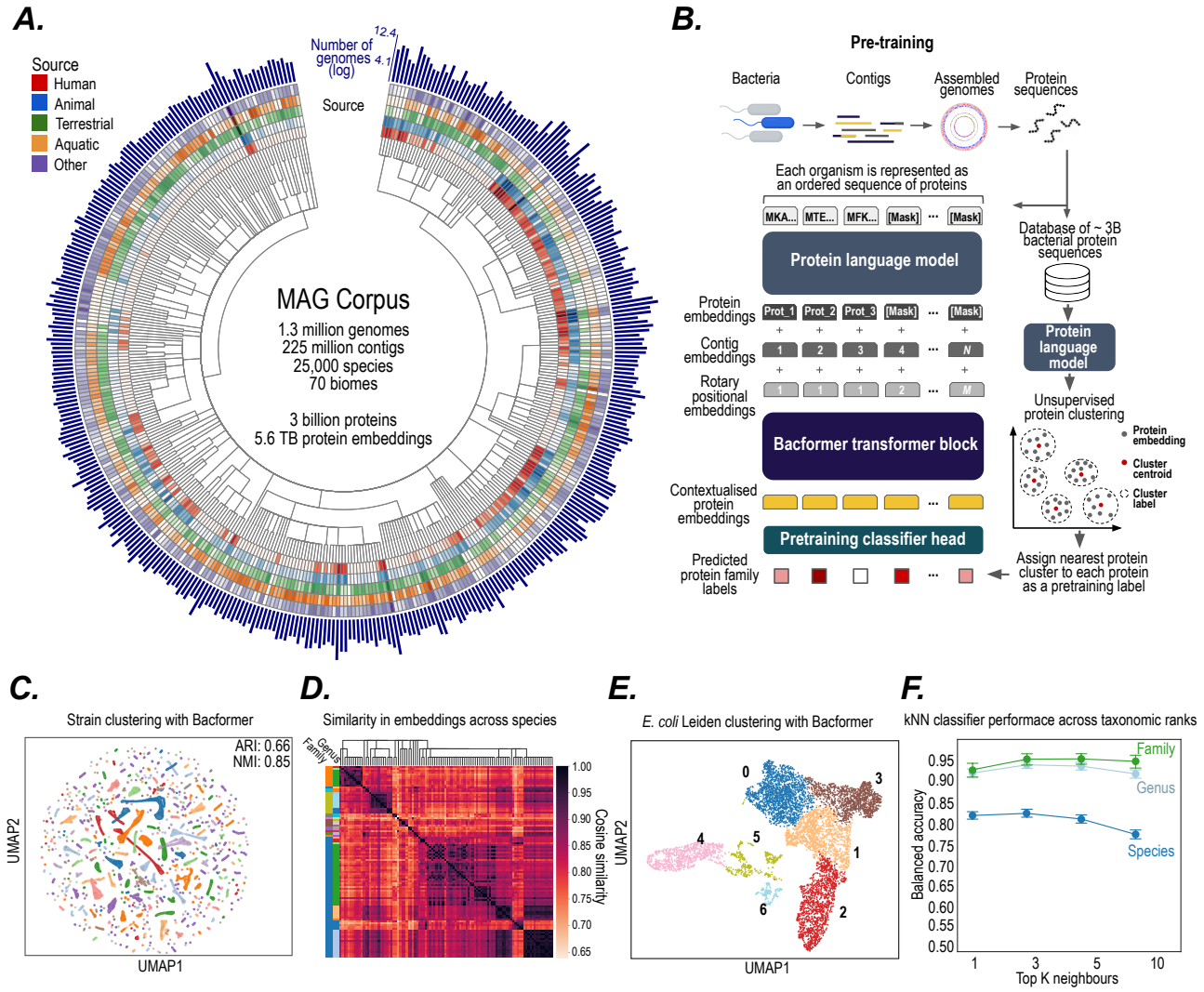

FIGURE 1. **Bacformer, a bacterial foundation model, represents bacterial genomes as a sequence of contextualised protein embeddings. (A)** To pretrain *Bacformer*, we constructed a metagenome-assembled genomes (MAG) corpus, a dataset of more than 1.3 million genomes, 3 billion proteins, and 225 million contigs collected across 25,000 species and 70 distinct environments. For visualisation, the taxonomy tree represents a set of the 200 orders with the highest number of genomes in the MAG corpus, annotated by number of genomes (*log*; dark blue outer circle) and distribution across biospheres. **(B)** *Bacformer* model architecture and pretraining scheme. *Bacformer* takes as input the set of protein sequences present in a bacterial strain, ordered by genome location. Protein sequences are then embedded with a pretrained protein language model (*ESM-2* [23]) and supplemented with rotary positional embeddings [24] as well as learnable contig representations. The resulting protein representations are then used as input to the transformer, which computes contextualised protein embeddings using other proteins present in the genome. During pretraining, the model predicts a protein family of the masked protein based on the genomic context. The discrete vocabulary of protein family labels necessary for pretraining were established by embedding the non-contextual protein sequences with a protein language model and performing unsupervised clustering. **(C)** *Bacformer* accurately clusters MAGs across diverse genera, in an unsupervised manner. UMAP plot showing strain clustering using *Bacformer* embeddings of assemblies from the 10 most frequent genera in the MAG corpus (clustering metrics: Adjusted Rand Index (ARI); Normalised Mutual Information (NMI); and Average Silhouette Width (ASW): all metrics vary from 0 to 1, higher is better; see Fig. S3 for plot legend). **(D)** *Bacformer* representation of MAGs maintains evolutionary relationships. Cosine similarity heatmap for 100 most common species of Bacteroidales showing higher similarity between species in the same family (rows) and genus (columns). **(E)** *Bacformer* reveals subpopulations within a single species. UMAP plot of Escherichia coli MAGs from the *MGnify* dataset [21] using genome embeddings from *Bacformer* annotated with *Leiden* clustering [25] labels (colours). **(F)** *k*NN classifier results across taxonomic ranks (family (green), genus (light blue), and species (dark blue)) demonstrate that *Bacformer* accurately identifies similar bacteria. Error bars represent 95% confidence interval across genomes.

which is represented using existing (*ESM-2* [23]) protein language model (pLM) embeddings which capture evolutionary and functional properties of proteins [26, 23, 27], supplemented with rotary positional embeddings [24] and special tokens marking contigs; an arrangement that preserves the genomic ordering of proteins (Fig. 1B; see *Supplementary Methods*). *Bacformer* then utilises these enhanced embeddings of the entire proteome of an organism as inputs to a 12-layer transformer model [28], treating each protein sequence as a single token and processing the complete genome in a single pass.

Unlike DNA or protein-based language models that rely on pre-defined and constrained token dictionaries [29, 30, 31, 23, 26], Bacfomer accommodates the vast diversity of possible proteins by clustering them within embedding space into protein families, and using these labels as a discrete vocabulary for the autoregressive pretraining task, where the model learns to predict the family of masked proteins based on the context of other proteins in the genome (Fig. 1B). To ensure that our protein family clusters reflect genuine evolutionary relationships, we benchmarked them against established orthology databases (eggNOG [32]) and clustering methods (MMseqs2 [33]), obtaining normalised mutual information (NMI) scores of 74% and 88% respectively.

We observed that the pretraining accuracy and perplexity of *Bacformer* increased progressively during pretraining, improving with model and dataset size (demonstrating robust scaling laws seen with other transformer architectures [34, 35]), and generalised across species, to previously unseen genera, and across a range of genome completeness (Fig. S1; Fig. S2). We further refined *Bacformer* by extending pretraining to include 48,932 complete genomes [36] (which we expect to provide important additional contextual information). Having access to complete genome boosted both zero-shot and finetuned performance, highlighting the value of an uninterrupted genomic context for the model (Fig. S2).

Genome embeddings uncover evolutionary relationships across bacteria

We next explored how the unified representation of bacterial genomes learned by *Bacformer* might provide insights into strain-level diversity and reveal sub-populations with distinct functional or evolutionary characteristics.

Existing clustering methods for bacteria operate only on closely related strains or rely on a limited set of marker genes and fail to consider gene content [37, 38, 39, 40, 41]. In contrast, *Bacformer* represents individual genomes across species in a shared embedding space (by averaging their contextual protein embeddings across the genome), enabling more robust strain-level comparisons based on the organisation of proteins within each isolate, and learning evolutionary relationships in an unsupervised way.

When applied to different sets of species, genera, and families (see *Supplementary Methods*), *Bacformer* accurately clustered strains (based on Adjusted Rand Index (ARI), Adjusted Silhouette Width (ASW), and NMI), considerably outperforming non-contextual *ESM-2* embeddings by between 5

and 13% on ARI (Fig. 1C; Fig. S3). Moreover, even in this fully unsupervised manner, *Bacformer* preserved phylogenetic relationships. For example, for the *Bacteroidales* order, *Bacformer* embeddings produced average cosine similarities among species belonging to the same genus or to the same family of 0.96 ± 0.02 and 0.93 ± 0.05, respectively (Fig. 1D).

To evaluate whether *Bacformer* can resolve fine-scale diversity within a single species and identify meaningful evolutionary differences, we examined the *Escherichia coli* sample collection from the *MGnify* dataset [21]. We extracted *Bacformer* embeddings for each MAG and applied *Leiden* clustering [25], which clearly identified six distinct subpopulations (Fig. 1E) that were not well delineated when we used *ESM-2* embeddings (Fig. S3), and were not driven by artifacts, such as genome sample or study origin (Fig. S4). Using pre-computed graphical pangenome analysis (Panaroo [42]), we identified several core and accessory genes that were significantly cluster-specific (adjusted $p < 0.05$, Fisher's exact test) including: *cbtA* (in Cluster 2) that regulates cell division and elongation [43], *stxA* (in Cluster 5) that produces Shiga toxin responsible for life-threatening complications [44, 45], and *fldP* (in Cluster 3) that shields cells from oxidative stress and aids survival in low-oxygen environments [46, 47].

Bacformer embeddings can also be used to rapidly search for related genomes and infer evolutionary context, ecological niche and bacterial behaviours. We linked the 1.3M *Bacformer* genome embeddings to associated biological and clinical metadata to allow users to input an unannotated genome and retrieve similar genomes and associated phenotypes from our database. We validated the effectiveness of this method by applying a k-nearest neighbour (kNN) classifier to a held-out subset of bacterial genomes, demonstrating highly accurate retrieval across taxonomic ranks (Fig. 1F), with the method achieving almost 95% balanced accuracy across diverse genera.

Thus, *Bacformer*'s learned representations of complete contextualised genomes can reveal important evolutionary signals at both broad phylogenetic and localised strain-specific scales, and can be used to infer bacterial properties of unseen genomes.

Bacformer reveals functional organisation of proteins

Since *Bacformer* was trained to predict proteins from their genomic context, we anticipated that it would learn the evolutionary relationships between proteins and hence their organisation into functional networks.

To explore this potential emergent property, we first examined the model attention weights across thousands of genes in diverse genomes and found that *Bacformer* focuses on neighbouring genes (Fig. S5). While early layers distribute attention evenly, middle layers concentrate attention strongly on closest neighbouring genes, whereas later layers spread attention further along the chromosome (Fig. S5). *Bacformer* strongly updates embeddings of adjacent proteins (assessed by their cosine similarity), highlighting the importance of rotary positional embeddings in capturing gene order (Fig. S5), and indicating that it has learnt evolutionarily-conserved arrangements of contiguous

proteins.

We therefore tested whether *Bacformer* could predict the presence of operons (physically adjacent co-transcribed genes that represent the fundamental regulatory units of bacterial genomics [9]). Using Operon DB [48] as ground truth, *Bacformer* accurately predicted operons across a wide range of gene sizes, markedly out-performing non-contextual methods, such as *ESM-2* (Fig. 2A & B; Fig. S6).

Since current operon annotations are incomplete and potentially inaccurate, we performed long-read RNA sequencing of strains from five bacterial species (*Mycobacterium tuberculosis*, *Mycobacterium abscessus*, *Staphylococcus aureus*, *Pseudomonas aeruginosa*, and *Escherichia coli*), and obtained accurate zero-shot *Bacformer* predictions (AUC 0.76 – 0.92) when evaluated against experimentally validated operons (Fig. S2; Fig. S6).

We next evaluated whether *Bacformer* also implicitly learns genome-wide protein-protein interactions and hence would be able to predict functional protein networks. We computed pairwise cosine similarities in protein embeddings for a diverse test set of 2,088 bacterial genomes across 1,573 species and employed these in a zero-shot classification task, using different types of interaction score (combined, co-occurrence, neighbourhood, co-expression) from the *STRING* database [49] as binary labels. *Bacformer* considerably outperformed non-contextual protein language models (*ESM-2*) across all tasks (Fig. 2D) and was further significantly improved by fine-tuning (p-value < 0.01, t-test; see *Supplementary Methods*, Fig. 2E & F; Fig. S6).

When applied to *P. aeruginosa* (PAO1), the finetuned model uncovered previously known, as well as novel, high-confidence protein-protein interactions (Fig. 2G). We then used *AlphaFold3* [50] to validate the novel interactions by predicting the structure of protein complexes (Fig. 2H). Protein complexes predicted by *Bacformer* had significantly higher *ipTM* scores (a metric that evaluates the accuracy of predicted protein-protein interfaces [51]) than for complexes based on random pairs ($p < 0.003$; Welch's t-test; Supp Fig. S9), indicating direct physical binding.

Bacformer was also able to identify broader regulatory networks using its embedding similarity matrix, even without direct input of transcriptional information. For example, in *P. aeruginosa*, *Bacformer* was able to predict clusters of co-expressed genes more accurately than *ESM-2* (Fig. S10) and identify networks mediating canonical quorum-sensing [52], biofilm formation [53], and motility [54] (Fig. 2I; Fig. S10), showcasing how *Bacformer* embeddings can reveal co-evolved, co-regulated, and interacting protein families.

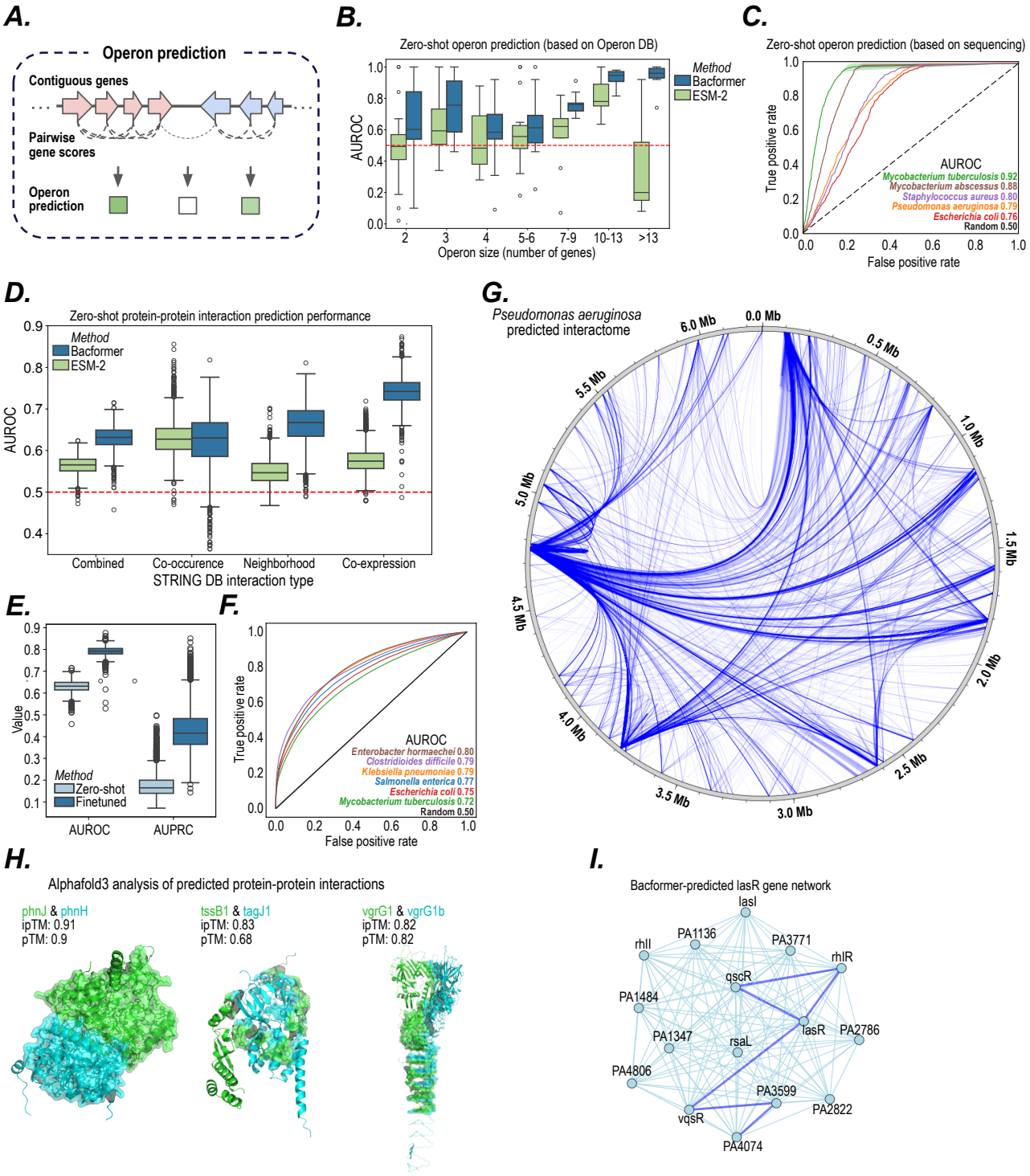
A.
Operon prediction
Contiguous genes
Pairwise gene scores
Operon prediction
B.
Zero-shot operon prediction (based on Operon DB)
AUROC
Method
Bacformer
ESM-2
2
3
4
5-6
7-9
10-13
>13
Operon size (number of genes)
C.
Zero-shot operon prediction (based on sequencing)
True positive rate
False positive rate
AUROC
Mycobacterium tuberculosis 0.92
Mycobacterium abscessus 0.88
Staphylococcus aureus 0.80
Pseudomonas aeruginosa 0.79
Escherichia coli 0.76
Random 0.50
D.
Zero-shot protein-protein interaction prediction performance
AUROC
Method
Bacformer
ESM-2
Combined
Co-occurence
Neighborhood
Co-expression
STRING DB interaction type
E.
Value
Method
Zero-shot
Finetuned
AUROC
AUPRC
F.
True positive rate
False positive rate
AUROC
Enterobacter hormaechei 0.80
Clostridioides difficile 0.79
Klebsiella pneumoniae 0.79
Salmonella enterica 0.77
Escherichia coli 0.75
Mycobacterium tuberculosis 0.72
Random 0.50
G.
Pseudomonas aeruginosa predicted interactome
0.0 Mb
0.5 Mb
1.0 Mb
1.5 Mb
2.0 Mb
2.5 Mb
3.0 Mb
3.5 Mb
4.0 Mb
4.5 Mb
5.0 Mb
5.5 Mb
6.0 Mb
H.
Alphafold3 analysis of predicted protein-protein interactions
phnJ & phnH
ipTM: 0.91
pTM: 0.9
tssB1 & tagJ1
ipTM: 0.83
pTM: 0.68
vgrG1 & vgrG1b
ipTM: 0.82
pTM: 0.82
I.
Bacformer-predicted lasR gene network
lasI
PA1136
PA3771
rhlI
rhlR
qscR
PA1484
lasR
PA2786
PA1347
rsaL
PA4806
PA3599
PA2822
vqsR
PA4074

FIGURE 2. **Bacformer identifies protein-protein interactions across bacterial genomes. (A)** *Bacformer* can identify operons by computing the pairwise interaction scores of contiguous genes. **(B)** Zero-shot operon prediction performance (evaluated by area under the receiver operator characteristic (ROC) curve; AUROC) for *Bacformer* (blue) and *ESM-2* (green, [23]) across operon sizes using a set of 45 genomes with a total of 5,126 annotated operons obtained from Operon DB [48] as ground truth. Line and boxes represent median and interquartile range (IQR) respectively; whiskers extend 1.5 × IQR beyond Q1 and Q3; random classifier AUROC (red dashed line). **(C)** *Bacformer* zero-shot operon predictions (assessed by AUROC) for five bacterial species where we generated experimental ground truth data from long-read RNA sequencing. **(D)** *Bacformer* learns protein-protein interactions across different interaction types from *STRING DB* (blue) [49], outperforming non-contextual protein embeddings (*ESM-2*, green). The interaction scores were calculated using cosine similarity between protein representations on held-out genomes ($N = 2,088$). Line and boxes represent median and interquartile range (IQR) respectively; whiskers extend 1.5 × IQR beyond Q1 and Q3; random classifier AUROC (red dashed line). **(E)** Finetuning *Bacformer* significantly boosts performance on protein-protein prediction task across genomes. We used *STRING DB* combined score as labels. Line and boxes represent median and interquartile range (IQR) respectively; whiskers extend 1.5 × IQR beyond Q1 and Q3. **(F)** ROC curves of finetuned *Bacformer* performance across diverse bacterial species. **(G)** Proteome-wide interaction scores for *Pseudomons aeruginosa* PAO1 predicted from finetuned *Bacformer* (line thickness indicates prediction strength; only predictions with probability > 0.5 were visualised). **(H)** Predicted structures obtained from *AlphaFold3* [50] for top scoring protein complexes from *P. aeruginosa* PAO1 predicted by *Bacformer* (ipTM score indicates the probability of the two proteins binding together and pTM score is a measure of overall predicted structure confidence; range for both 0-1, higher is better). **(I)** Protein-protein interaction network extracted using *Leiden* clustering (resolution= 40) on *Bacformer* contextualised protein embeddings. The edge weights with high importance (cosine similarity > 0.8) are highlighted in dark blue.

Predicting protein essentiality and functions

To explore how kingdom-wide evolutionary signals and their genome-wide context learnt by *Bacformer* might reveal protein characteristics, we first examined its ability to predict protein essentiality (Fig. 3A). To create a ground truth dataset, we compiled genomes from 51 bacterial genomes across 37 species (from the *DEG* database [55]), providing a total of 169,408 genes labelled as *essential* or *non-essential*. We then examined the performance of both zero-shot prediction and after fine-tuning; the latter significantly outperforming finetuned protein language (*ESM-2*) and DNA language (*Evo* [31]) models (Fig. 3B-C; Fig. S11). The finetuned *Bacformer* model was able to clearly separate essential from non-essential genes based on predicted probability distributions, achieving 0.81 ± 0.07 median AUROC across diverse genomes, and was able to identify many genes not previously annotated as essential which we scored as false positives, but which had orthologs in other species that are known to be essential, such as ribosomal proteins [56, 57] and tRNA synthetases [58, 59], indicating that our model made biologically plausible predictions.

We then evaluated the ability of *Bacformer* to predict protein function. To mimic the real-world setting where labels are often scarce, we used a limited set of 188 *KEGG* function labels [60] for

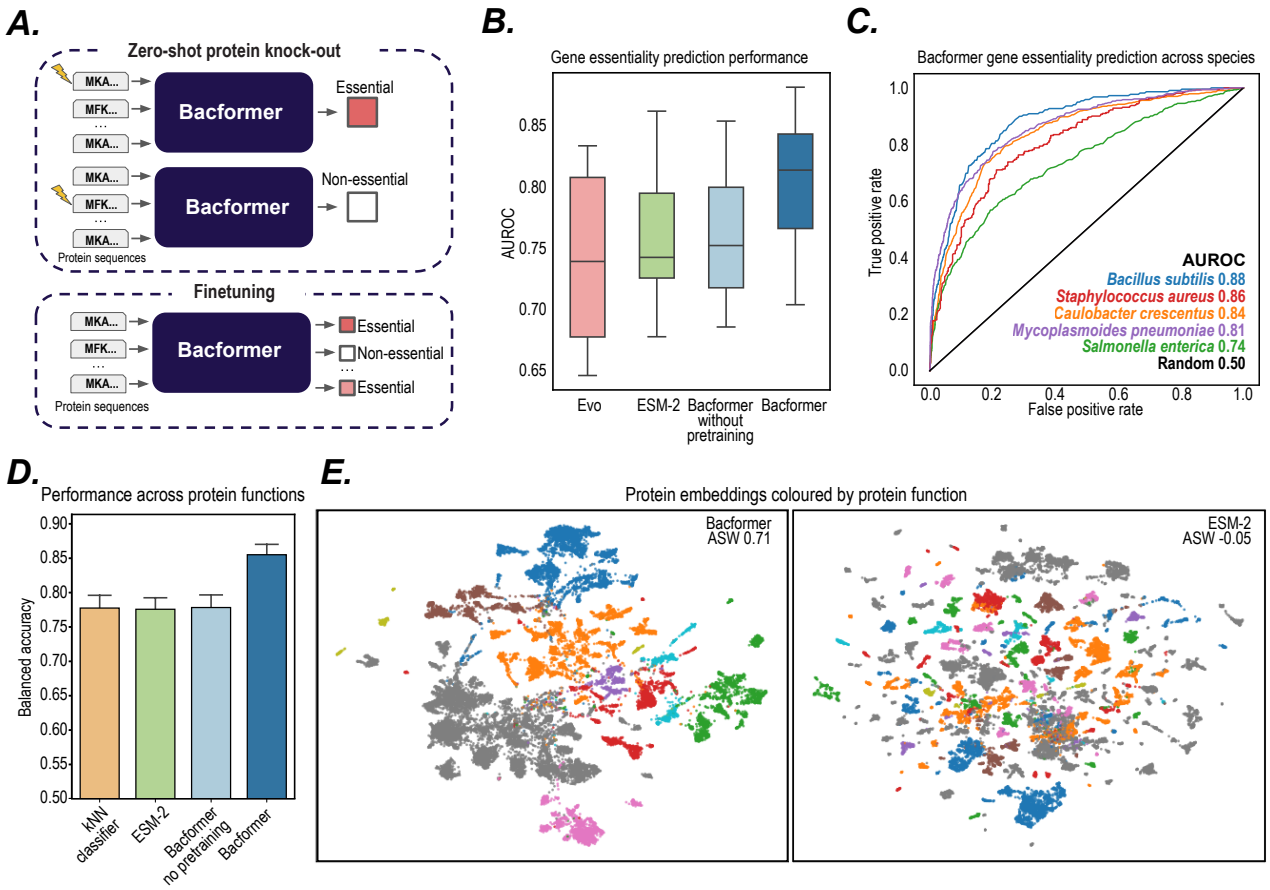

FIGURE 3. **Bacformer leverages genomic context to predict gene essentiality and protein function.** **(A)** Predicting gene essentiality and function using *Bacformer*. Top: In the zero-shot setting, we knocked-out the gene in silico by zeroing its representation at input level. We then measured the effect of the knock-out on the whole genome comparing the effect of essential and non-essential genes on genome embedding. Bottom: *Bacformer* is finetuned to predict protein essentiality or function label. **(B)** Finetuned *Bacformer* significantly outperforms competing methods on gene essentiality prediction across genomes on the held-out evaluation set (N = 10) evaluated by area under the receiver operating characteristic curve (AUROC). Line and boxes represent median and interquartile range (IQR) respectively; whiskers extend 1.5 × IQR beyond Q1 and Q3. **(C)** Receiver operating characteristic curves (ROC curve) for finetuned *Bacformer* predictions of gene essentiality across diverse species (showing AUROC values). **(D)** Pretrained *Bacformer* significantly outperforms competing methods (kNN classifier, orange; *ESM-2*, green; zero-shot *Bacformer*, light blue; finetuned *Bacformer*, dark blue) on protein function prediction of 503, 295 proteins across 188 functional classes. The protein function labels were taken from the *MGnify* dataset [21] which used *KEGG* [60] protein function labels (level 4). Error bars represent standard error. **(E)** UMAP plots of protein embeddings coloured by protein function predictions from *Bacformer* (left) and *ESM-2* (right) for a set of 10 *KEGG* protein functions from the *MGnify* dataset. Clustering was evaluated using Average Silhouette Width (ASW; range -1 to 1; higher is better).

210, 975 proteins from 666 genomes (from the *MGnify* dataset [21]) for training, with a much larger held-out test set of 503, 295 proteins from 1, 931 genomes used for evaluation. *Bacformer* was able to significantly outperform *ESM-2* and a *k*NN baseline model after training on the full dataset (Fig. 3D) and within low-data (few-shot) regimes (Fig. S9). Examining the protein embeddings of held-out genomes, *Bacformer* generated more cohesive groupings of proteins sharing the same function than *ESM-2* (ASW scores of 0.71 and −0.05 respectively; Fig. 3E). Surprisingly, *Bacformer* was still able to make confident predictions even when the query protein itself was masked (0.56 balanced accuracy across held-out genomes; Fig. S10), indicating how important the learned contextual information is to model performance.

Predicting bacterial phenotypes from genomes

We next assessed whether *Bacformer*, having learned the structure of protein co-organisation and co-evolution, could be used to predict bacterial phenotypes. We found that *Bacformer* genome embeddings intrinsically reflect phenotypic traits even without fine-tuning. For example, isolate embeddings clustered by gram stain (Fig. S11) and by host environment, even when comparing closely-related species (for example within the *Bifidobacterium* genus; Fig. 4A).

Encouraged by these observations, we compiled a dataset of 24, 462 genomes from 15, 477 species that had been annotated with 139 distinct phenotypic traits [61, 62, 63] and used this to train separate classifiers for each phenotype, using *Bacformer* genome-level embeddings (Fig. 4B). *Bacformer* outperformed all benchmarks and accurately predicted phenotypes across a range of diverse traits including carbon source utilisation, optimal growth temperature, motility, and toxin production (Fig. 4C).

We then employed those phenotype-specific models with high predictive performance (balanced accuracy > 85%) to annotate the entire 1.3M genomes within the MAG corpus across 32 diverse phenotypes spanning metabolic, ecological, and physiological characteristics (Fig.4D; see *Supplementary Methods*). We provide these models, annotations, and accompanying metadata, which represent the first large-scale resource for kingdom-wide phenotype inference of bacteria for researchers to explore, filter, or extend for their own studies.

The accuracy of *Bacformer* in predicting phenotypes motivated us to further leverage its contextual genomic representations to discover the genes that are likely causally associated with specific traits by employing a gradient-based attribution method [64] that assigns importance scores to genes within a finetuned phenotype classifier.

We first examined the ability of this method to identify genes that were identified as related to sporulation, motility, and anaerobic growth (three phenotypes with some pre-existing ground truth knowledge of responsible genes [65, 66, 67, 60]), and found that phenotype-associated genes were assigned higher importance scores than other genes (Mann-Whitney test, $p < 10^{-10}$; Fig. S11).

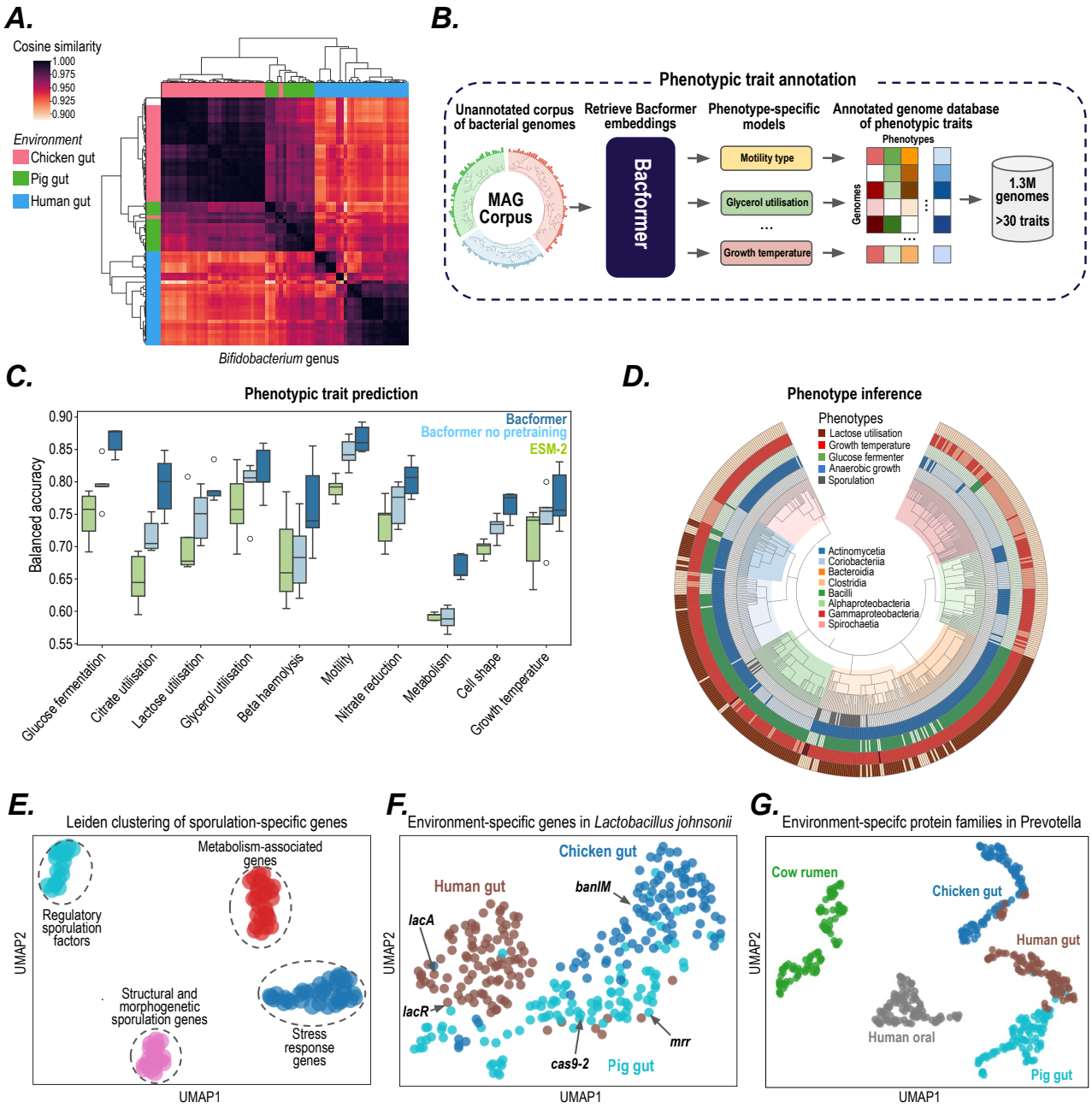
A.
Cosine similarity
1.000
0.975
0.950
0.925
0.900
Environment
Chicken gut
Pig gut
Human gut
Bifidobacterium genus
B.
Phenotypic trait annotation
Unannotated corpus of bacterial genomes
Retrieve Bacformer embeddings
Phenotype-specific models
Annotated genome database of phenotypic traits
MAG Corpus
Bacformer
Motility type
Glycerol utilisation
Growth temperature
Phenotypes
Genomes
1.3M genomes
>30 traits
C.
Phenotypic trait prediction
Balanced accuracy
Bacformer
Bacformer no pretraining
ESM-2
Glucose fermentation
Citrate utilisation
Lactose utilisation
Glycerol utilisation
Beta haemolysis
Motility
Nitrate reduction
Metabolism
Cell shape
Growth temperature
D.
Phenotype inference
Phenotypes
Lactose utilisation
Growth temperature
Glucose fermenter
Anaerobic growth
Sporulation
Actinomycetia
Coriobacteria
Bacteroidia
Clostridia
Bacilli
Alphaproteobacteria
Gammaproteobacteria
Spirochaetia
E.
Leiden clustering of sporulation-specific genes
Regulatory sporulation factors
Metabolism-associated genes
Structural and morphogenetic sporulation genes
Stress response genes
UMAP2
UMAP1
F.
Environment-specific genes in Lactobacillus johnsonii
Chicken gut
Human gut
banIM
lacA
lacR
cas9-2
Pig gut
mrr
UMAP2
UMAP1
G.
Environment-specifc protein families in Prevotella
Cow rumen
Chicken gut
Human gut
Human oral
Pig gut
UMAP2
UMAP1

FIGURE 4. **Bacformer enables phenotypic traits prediction and discovery of phenotype-specific genes. (A)** *Bacformer* genome embeddings capture environment niche in an unsupervised manner. Cosine similarity heatmap between strains from the *Bifidobacterium* genus showing higher similarity between strains from the same animal niche (chicken gut, pink; pig gut, green; human gut, blue). The dendrogram was created by performing hierarchical clustering on *Bacformer* genome embeddings. **(B)** *Bacformer* annotates genomes with diverse phenotypic traits. We trained phenotype-specific models using *Bacformer* genome embeddings as inputs to annotate genomes in the MAG corpus with more than 30 diverse phenotypic traits. **(C)** Phenotype prediction performance of pretrained *Bacformer* (dark blue), zero-shot *Bacformer* (light blue), and *ESM-2* (green) shown for a range of phenotypes. Line and boxes represent median and interquartile range (IQR) respectively; whiskers extend 1.5 × IQR beyond Q1 and Q3. **(D)** Example of phenotype inference across bacterial classes. Inner circle: phylogenetic tree of a selection of bacterial genomes (sampled from the MAG corpus), colour-coded by class. Outer circles: a subset of phenotypes (from a total of 139) inferred using phenotype-specific models trained using *Bacformer* genome embeddings. **(E)** UMAP plot of the 100 genes with the highest importance scores for the sporulation phenotype across genomes (using their embeddings extracted from the last layer of the *Bacformer* model and averaged across genomes) and annotated by *Leiden* clustering label and function (function labels were established by analysing the genes within each *Leiden* cluster and leveraging the phenotype labels from the GIDEON dataset [62]). **(F)** The top 100 genes from *Lactobacillus johsonii* genomes with the highest importance scores for each environmental niche (human gut, brown; chicken gut, dark blue; pig gut, light blue; i.e. isolation source) and plotted as a UMAP (using gene embeddings extracted from the last layer of the *Bacformer* model and averaged across genomes). Some genes with known association to the environmental niche are labelled. Genomes and environmental niche labels provided by the *MGnify* dataset [21]. **(G)** UMAP plot of the top 100 protein families with highest importance scores for each environmental niche (cow rumen, green; human oral, grey; chicken gut, dark blue; human gut, brown; pig gut, light blue) across *Prevotella* genus genomes using protein family embeddings were extracted from the last layer of the *Bacformer* model and averaged across genomes. Genomes and environmental niche labels provided by the *MGnify* dataset [21].

We then examined in detail the sporulation-associated genes identified by *Bacformer* and found that: the canonical sporulation genes, *spolllAG* and *spolllAF* [68, 69], had been ascribed the highest importance scores; and that clustering the top 100 sporulation-associated genes revealed functionally distinct groups including known regulatory factors (*sigG*, *spollAB* [70, 71]) and structural genes (*spoVE*, *spoVB* [72, 73]), as well as previously uncharacterised, but plausible, candidates, such as *katB*, *galE*, and *timB* (Fig. 4F).

We next tested whether we could use *Bacformer* to identify niche-specific genes. Within *Lactobacillus johnsonii*, a gut bacteria found in several animal species [74, 75], *Bacformer* was able to identify known genes involved in host-specific adaptation, such as *lacA* and *lacR* (human, [74]), *cas9* genes in livestock [76, 77], type III RM gene (*mmr*) (pig [76]), and *banIM* (chicken, [78]), as well as a credible set of additional novel genes (Fig. 4F). Finally, applying a similar approach to the *Prevotella*, a genus for which gene annotations and experimental validations are sparse, *Bacformer* was able to identify novel protein families specifically associated with gastrointestinal environments in different animal species (Fig. 4G).

Thus, by combining large-scale phenotype prediction with high-resolution gene-level attribution, *Bacformer* can not only assign specific traits across bacteria but can also discover potential causally-associated genes.

Generating bacterial genomes *de novo*

Since *Bacformer* has learnt genomic representations of bacteria that encompass evolutionary relationships, reflect protein functional organisation, and allow phenotype inference, we next looked to evaluate its potential for synthesising bacterial genomes *de novo*.

We first evaluated *Bacformer*'s generative capabilities by analysing its performance on unseen genomes. *Bacformer* achieved low perplexity (a measure of uncertainty when predicting the next token) across diverse genomes (Fig. 5A; Fig. S12), suggesting it captures the genomic architecture also of strains not encountered in the training set. We then prompted *Bacformer* with a short sequence of initial proteins from *Escherichia coli* and *Staphylococcus aureus*, and asked it to generate a complete genome (see *Supplementary Methods*). We showed that the resulting ordered sequence of protein families within synthesised genomes was comparable in distribution and ontological diversity to those of true genomes (Fig. 5A & B; Fig. S12), demonstrating *Bacformer*'s capacity to create genomes that span the full spectrum of protein functions. To ensure that the model did not simply recapitulate genomes seen during training, we generated multiple genomes from identical protein prompts while varying the temperature (a scaling factor in autoregressive sampling that controls randomness) and observed that *Bacformer* generated a variety of distinct genomes, none of which were replicas of any within the training corpus (Fig. S12).

We then tested whether *Bacformer* could accurately retrieve the full genome from an incomplete set of proteins, a valuable capability particularly for partial genomes typically recovered from metagenomic sequencing approaches. We therefore artificially removed a percentage of proteins from a collection of held-out complete genomes and computed their genome embeddings compared to the complete genomes, calculating the mean reciprocal rank (MRR). *Bacformer* was able to extrapolate the full genomic context even when only 50% of proteins were present, demonstrating the ability to confidently complete genomes from sparse protein sets (Fig. 5C; Fig. S13).

Since *Bacformer* was trained to predict protein families rather than specific proteins, we augmented its training with a contrastive loss, encouraging the model's contextual embeddings to align closely with actual protein embeddings present in our database (see *Supplementary Methods*), thereby enabling retrieval of specific protein sequences from generated contextual protein embeddings (Fig. 5D). To align the protein sequence embedding with the generated contextual embeddings, we finetuned *Bacformer* on a non-causal protein prediction task (where 15% of proteins were masked), combining the masked objective with a contrastive loss. Resulting embeddings were significantly more similar to their true counterparts than to other proteins (Fig. S13), and demonstrated the capability to retrieve the underlying protein sequence from a contextualised protein representation (Fig. 5E).

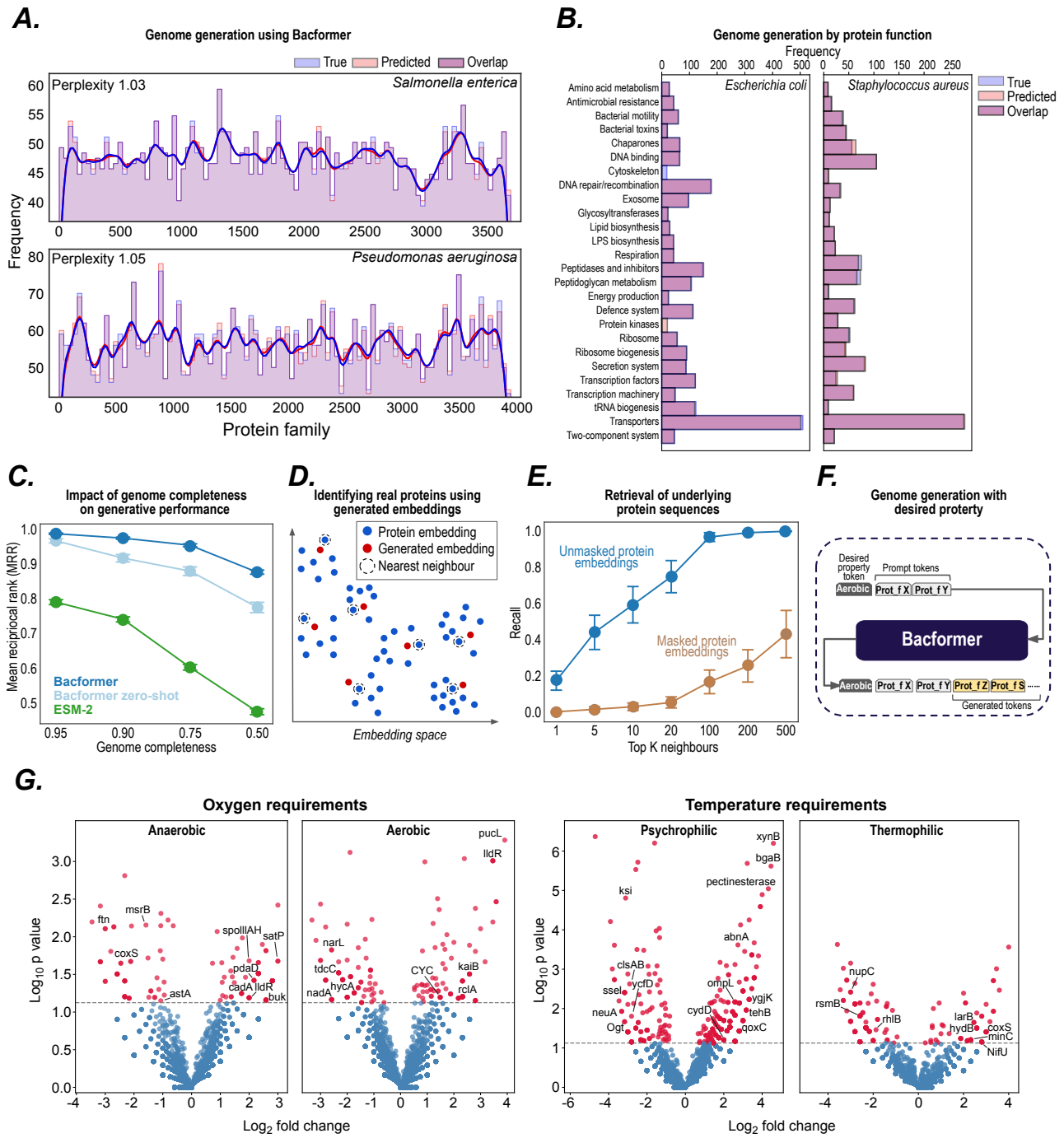
A.
Genome generation using Bacformer
True
Predicted
Overlap
Perplexity 1.03
Salmonella enterica
Perplexity 1.05
Pseudomonas aeruginosa
Frequency
Protein family
B.
Genome generation by protein function
Frequency
Escherichia coli
Staphylococcus aureus
Amino acid metabolism
Antimicrobial resistance
Bacterial motility
Bacterial toxins
Chaperones
DNA binding
Cytoskeleton
DNA repair/recombination
Exosome
Glycosyltransferases
Lipid biosynthesis
LPS biosynthesis
Respiration
Peptidases and inhibitors
Peptidoglycan metabolism
Energy production
Defence system
Protein kinases
Ribosome
Ribosome biogenesis
Secretion system
Transcription factors
Transcription machinery
tRNA biogenesis
Transporters
Two-component system
C.
Impact of genome completeness on generative performance
Mean reciprocal rank (MRR)
Bacformer
Bacformer zero-shot
ESM-2
Genome completeness
D.
Identifying real proteins using generated embeddings
Protein embedding
Generated embedding
Nearest neighbour
Embedding space
E.
Retrieval of underlying protein sequences
Unmasked protein embeddings
Masked protein embeddings
Recall
Top K neighbours
F.
Genome generation with desired proterty
Desired property token
Prompt tokens
Aerobic
Prot_f X
Prot_f Y
Bacformer
Prot_f Z
Prot_f S
Generated tokens
G.
Oxygen requirements
Anaerobic
Aerobic
Log10 p value
Log2 fold change
Temperature requirements
Psychrophilic
Thermophilic

FIGURE 5. **Bacformer generates whole bacterial genomes with desired properties. (A)** *Bacformer* successfully generates genomes not seen during training. Next protein family prediction (orange) on *Salmonella enterica* (top) and *P. aeruginosa* (bottom) compared to true (blue) protein family distributions. **(B)** *Bacformer* generates genomes encompassing a realistic range of protein functions (predicted, orange; true, blue) for *E. coli* (left) and *S. aureus* (right). We prompted *Bacformer* with a short sequence of initial proteins (500) and evaluated the generated genomes against true labels. **(C)** *Bacformer* retrieves true complete genome from incomplete genomes. Mean reciprocal rank (MRR, 0-1, higher is better) of genomes generated from inputs with ablated proteins by zero-shot (light blue) and non-zero-shot finetuned *Bacformer* (dark blue) and *ESM-2* (green). Error bars represent 95% confidence intervals. **(D)** Visualisation showing the process of identifying existing proteins from *Bacformer* generated contextual protein embeddings using next-protein family prediction. **(E)** Retrieval of underlying protein sequences using unmasked contextualised protein embeddings (blue) and masked (brown) embeddings across a range of k neighbours. Error bars represent standard error. **(F)** *Bacformer* generates sequences of protein families with desired properties. We finetune *Bacformer* to generate protein family sequences conditional on a property. The property is represented as a token appended to the beginning of the sequence. The prompt includes the property token and a set of initial protein families present in the genome which is used by the model to generate the rest of the genome. **(G)** We generated sequences of protein families with models finetuned for different genome properties (in this case oxygen requirement and optimal growth temperature). We then compared sequences produced with and without the property token, calculated the log_2 fold change for *KEGG orthology* (KO) groups [60], and assessed statistical significance with Fisher's exact test [79].

Finally, we explored how *Bacformer* could be used to generate entire bacterial genomes endowed with specific properties, a core ambition of synthetic biology [80, 81, 82]. Leveraging our phenotypic trait analysis (Fig. 4), we introduced a property token at the start of each protein-family sequence (Fig. 5F), and then finetuned *Bacformer* to generate protein-family sequences conditional on a particular phenotype, selecting two different traits (oxygen requirement and optimal growth temperature) to evaluate its generative capacity.

We then used each phenotype-specific model to generate full genomes from an initial set of protein families drawn from held-out species, and observed functional enrichment of proteins with biological roles linked to the specified phenotype (Fig. 5G). For example: conditioning on aerobic growth produced significant positive enrichment of urate oxidase (*pucL*) and cytochrome c family genes (*CYC*), both of which catalyse key oxidative steps in purine catabolism and electron transport during aerobic respiration [83, 84], and significant depletion of proteins that promote anaerobic metabolism (such as the nitrate-responsive regulator *narL* and formate-hydrogen lyase component *hycA* [85, 86]). Conditioning on psychrophilic growth yielded significant enrichment of cold-active xylanases (*xynB*) and β-galactosidases (*bgaB*), enzymes commonly isolated from polar and deep-sea bacteria where they facilitate polysaccharide degradation at low temperature [87, 88], and significant depletion of genes (such as steroid Delta-isomerase (*ksi*) [89]). These findings indicate the potential utility of *Bacformer* for rational genome design.

Discussion

Bacformer, a genomic foundation model trained on millions of diverse bacterial strains, offers a unified representation of genomes, modelled as an ordered sequence of interacting proteins, and generalises across species to incorporate the constraints and relationships that shape proteomes during evolution.

During pretraining, *Bacformer* generates highly informative whole-genome embeddings that identify biologically important clusters of bacteria and their associated gene content (providing important biological and/or clinical insights), and can be used to rapidly annotate unknown genomes by relating them to existing isolates with known phenotypes. Importantly *Bacformer* exhibits emergent properties by learning both protein-protein interactions and protein functional organisation, thereby enabling accurate prediction of operon organisation, gene essentiality, and protein function, and acquiring the syntactic rubric underpinning bacterial evolution and functional diversification, thereby driving biological discovery.

Leveraging these emergent properties, we have trained *Bacformer* to predict bacterial phenotypes from genomes and identify the putative causal gene modules responsible, revealing important biological processes, permitting phenotype inference within or across species at microbiome or ecological scale, and potentially enabling state prediction following perturbations.

Bacformer can also be used for generative tasks, inferring complete genomes from an incomplete set of proteins and creating a road map for de novo genome synthesis with desired properties; with clear applications to synthetic biology and bioengineering. In summary, *Bacformer* provides both important insights into the evolutionary language of bacterial genomic organisation and function, and a generative approach to whole genome design.

Code and data availability

Code and models related to this study, together with tutorials are publicly available at `https://github.com/macwiatrak/Bacformer` under a permissive `Apache 2.0` License.

The genomic sequences and assemblies are all publicly available and downloaded from *MGnify* [21], *SPIRE* [22] and NCBI RefSeq [36]. The long-read RNA sequencing operon annotations are available at `https://huggingface.co/datasets/macwiatrak/operon-identification-long-read-rna-sequencing`.

Additional details on the datasets used in this study and processing are provided in *Methods*.

Acknowledgements

We thank Vitor Mendes, Chris Ruis, Stephane d'Ascoli, Äicha Bentaieb, Alma Andersson, Maxwell Berk Monconis, Kristina Kordova, Charlie Harris, Samuel Horsfield, John Lees, Oliver Hernandez, Timon Schneider, Paulina Kulyte, Chris Brown, Audra Devoto and Daniela Goltsman for helpful

dicussions on the study design and manuscript. We thank Severine Habert, Nikolai Solmsdorf and Anja Hagting for assistance with computational infracturcture. We thank Julia Staniszewska for help with visualisations. We thank the *SPIRE* authors for help in accessing the genomic sequences.

Author Contributions

M.W., A.W. and R.A.F. conceived the project. R.A.F., A.W. and M.B. supervised the project. M.W. designed the model architecture. M.W., A.W. and M.N. curated and preprocessed pretraining datasets. M.W., A.W., R.V.T. curated and preprocessed datasets for finetuning and downstream tasks evaluation. M.W. implemented and models and infrastructure for pretraining. M.W. evaluated the pretraining model and conducted the finetuning. M.W. and A.W. conducted experiments for downstream tasks analysis. D.A. prepared the samples for long-read RNA sequencing. A.W., M.W. and D.C.A. preprocessed and analysed the long-read RNA sequencing results. R.V.T. conducted experiments on phenotypic traits prediction. A.D. contributed to protein-protein interaction prediction analysis. M.W., R.A.F., A.W. and M.B. contributed to study design and analysis. D.C.A. conducted the ethics and safety investigation and discussion. M.W. implemented the public codebase. R.A.F., M.W. and A.W. wrote the first draft of the manuscript. All authors wrote the final draft of the manuscript.

Funding

This work was supported by: The Wellcome Trust (Discovery Award 226602/Z/22/Z (R.A.F., A.W., A.D., D.A., D.C.A., M.N.)); LifeArc (CF Innovation Hub grant IH001) (R.A.F., A.D., D.A.), INEOS Oxbridge AMR Doctoral training programme (M.N.), NIHR Cambridge Biomedical Research Centre (R.A.F.); Cambridge Centre for AI in Medicine (CCAIM) doctoral training programme (M.W.); Swiss National Science Foundation (SNSF) starting grant TMSGI2_226252/1 and SNSF grant IC00I0_23192 (M.B.). We gratefully acknowledge the support of the Peter und Traudl Engelhorn Foundation to R.V.T. This work was performed using resources provided by the Cambridge Service for Data Driven Discovery (CSD3) operated by the University of Cambridge Research Computing Service, provided by Dell EMC and Intel using Tier-2 funding from the Engineering and Physical Sciences Research Council (capital grant EP/T022159/1), and DiRAC funding from the Science and Technology Facilities Council.

Competing Interests

Authors declare no competing interests.

References

[1] Locey KJ, Lennon JT. Scaling laws predict global microbial diversity. Proceedings of the National Academy of Sciences. 2016;113(21):5970-5.

[2] Fenchel T. Biogeography for bacteria. Science. 2003;301(5635):925-6.

[3] Falkowski PG, Fenchel T, Delong EF. The microbial engines that drive Earth's biogeochemical cycles. science. 2008;320(5879):1034-9.

[4] Lozupone CA, Knight R. Global patterns in bacterial diversity. Proceedings of the National Academy of Sciences. 2007;104(27):11436-40.

[5] Kuypers MM, Marchant HK, Kartal B. The microbial nitrogen-cycling network. Nature Reviews Microbiology. 2018;16(5):263-76.

[6] Wu HJ, Wang AH, Jennings MP. Discovery of virulence factors of pathogenic bacteria. Current opinion in chemical biology. 2008;12(1):93-101.

[7] Diacovich L, Gorvel JP. Bacterial manipulation of innate immunity to promote infection. Nature Reviews Microbiology. 2010;8(2):117-28.

[8] Davies J, Davies D. Origins and evolution of antibiotic resistance. Microbiology and molecular biology reviews. 2010;74(3):417-33.

[9] Osbourn AE, Field B. Operons. Cellular and Molecular Life Sciences. 2009;66:3755-75.

[10] Balleza E, Lopez-Bojorquez LN, Martínez-Antonio A, Resendis-Antonio O, Lozada-Chavez I, Balderas-Martínez YI, et al. Regulation by transcription factors in bacteria: beyond description. FEMS microbiology reviews. 2008;33(1):133-51.

[11] Mendoza-Vargas A, Olvera L, Olvera M, Grande R, Vega-Alvarado L, Taboada B, et al. Genome-wide identification of transcription start sites, promoters and transcription factor binding sites in E. coli. PloS one. 2009;4(10):e7526.

[12] Rocha EP. The replication-related organization of bacterial genomes. Microbiology. 2004;150(6):1609-27.

[13] Hu XP, Brahmantio B, Bartoszek K, Lercher MJ. Most bacterial gene families are biased toward specific chromosomal positions. Science. 2025;388(6743):186-91.

[14] Blin K, Shaw S, Kloosterman AM, Charlop-Powers Z, Van Wezel GP, Medema MH, et al. antiSMASH 6.0: improving cluster detection and comparison capabilities. Nucleic acids research. 2021;49(W1):W29-35.

[15] Cimermancic P, Medema MH, Claesen J, Kurita K, Brown LCW, Mavrommatis K, et al. Insights into secondary metabolism from a global analysis of prokaryotic biosynthetic gene clusters. Cell. 2014;158(2):412-21.

[16] Jia B, Raphenya AR, Alcock B, Waglechner N, Guo P, Tsang KK, et al. CARD 2017: expansion and model-centric curation of the comprehensive antibiotic resistance database. Nucleic acids research. 2016:gkw1004.

[17] Feldgarden M, Brover V, Gonzalez-Escalona N, Frye JG, Haendiges J, Haft DH, et al. AMRFinderPlus and the Reference Gene Catalog facilitate examination of the genomic links among antimicrobial resistance, stress response, and virulence. Scientific reports. 2021;11(1):12728.

[18] Wiatrak M, Weimann A, Dinan A, Brbić M, Floto RA. Sequence-based modelling of bacterial genomes enables accurate antibiotic resistance prediction. bioRxiv. 2024:2024-01.

[19] Blackwell GA, Hunt M, Malone KM, Lima L, Horesh G, Alako BT, et al. Exploring bacterial diversity via a curated and searchable snapshot of archived DNA sequences. PLoS biology. 2021;19(11):e3001421.

[20] Hunt M, Lima L, Anderson D, Hawkey J, Shen W, Lees J, et al. AllTheBacteria-all bacterial genomes assembled, available and searchable. bioRxiv. 2024:2024-03.

[21] Richardson L, Allen B, Baldi G, Beracochea M, Bileschi ML, Burdett T, et al. MGnify: the microbiome sequence data analysis resource in 2023. Nucleic Acids Research. 2023;51(D1):D753-9.

[22] Schmidt TSB, Fullam A, Ferretti P, Orakov A, Maistrenko OM, Ruscheweyh HJ, et al. SPIRE: a Searchable, Planetary-scale mIcrobiome REsource. Nucleic Acids Research. 2023 10;52(D1):D777-83. Available from: https://doi.org/10.1093/nar/gkad943.

[23] Lin Z, Akin H, Rao R, Hie B, Zhu Z, Lu W, et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. bioRxiv. 2022.

[24] Su J, Ahmed M, Lu Y, Pan S, Bo W, Liu Y. Roformer: Enhanced transformer with rotary position embedding. Neurocomputing. 2024;568:127063.

[25] Traag VA, Waltman L, Van Eck NJ. From Louvain to Leiden: guaranteeing well-connected communities. Scientific reports. 2019;9(1):1-12.

[26] Elnaggar A, Heinzinger M, Dallago C, Rehawi G, Wang Y, Jones L, et al. ProtTrans: Towards Cracking the Language of Life's Code Through Self-Supervised Deep Learning and High Performance Computing. bioRxiv. 2020. Available from: https://www.biorxiv.org/content/early/2020/07/21/2020.07.12.199554.

[27] Hayes T, Rao R, Akin H, Sofroniew NJ, Oktay D, Lin Z, et al. Simulating 500 million years of evolution with a language model. Science. 2025:eads0018.

[28] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is All you Need. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, et al., editors. Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc.; 2017. Available from: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[29] Zhou Z, Ji Y, Li W, Dutta P, Davuluri R, Liu H. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. arXiv preprint arXiv:230615006. 2023.

[30] Dalla-Torre H, Gonzalez L, Mendoza-Revilla J, Lopez Carranza N, Grzywaczewski AH, Oteri F, et al. Nucleotide Transformer: building and evaluating robust foundation models for human genomics. Nature Methods. 2024:1-11.

[31] Nguyen E, Poli M, Durrant MG, Kang B, Katrekar D, Li DB, et al. Sequence modeling and design from molecular to genome scale with Evo. Science. 2024;386(6723):eado9336.

[32] Huerta-Cepas J, Szklarczyk D, Heller D, Hernández-Plaza A, Forslund SK, Cook H, et al. eggNOG 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. Nucleic acids research. 2019;47(D1):D309-14.

[33] Steinegger M, Söding J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. Nature biotechnology. 2017;35(11):1026-8.

[34] Kaplan J, McCandlish S, Henighan T, Brown TB, Chess B, Child R, et al. Scaling laws for neural language models. arXiv preprint arXiv:200108361. 2020.

[35] Henighan T, Kaplan J, Katz M, Chen M, Hesse C, Jackson J, et al. Scaling laws for autoregressive generative modeling. arXiv preprint arXiv:201014701. 2020.

[36] O'Leary NA, Wright MW, Brister JR, Ciufo S, Haddad D, McVeigh R, et al. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. Nucleic acids research. 2016;44(D1):D733-45.

[37] Segata N, Waldron L, Ballarini A, Narasimhan V, Jousson O, Huttenhower C. Metagenomic microbial community profiling using unique clade-specific marker genes. Nature methods. 2012;9(8):811-4.

[38] Lan Y, Rosen G, Hershberg R. Marker genes that are less conserved in their sequences are useful for predicting genome-wide similarity levels between closely related prokaryotic strains. Microbiome. 2016;4:1-13.

[39] Jain C, Rodriguez-R LM, Phillippy AM, Konstantinidis KT, Aluru S. High throughput ANI analysis of 90K prokaryotic genomes reveals clear species boundaries. Nature communications. 2018;9(1):5114.

[40] Fouts DE, Brinkac L, Beck E, Inman J, Sutton G. PanOCT: automated clustering of orthologs using conserved gene neighborhood for pan-genomic analysis of bacterial strains and closely related species. Nucleic acids research. 2012;40(22):e172-2.

[41] Hall BG, Ehrlich GD, Hu FZ. Pan-genome analysis provides much higher strain typing resolution than multi-locus sequence typing. Microbiology. 2010;156(4):1060-8.
[42] Tonkin-Hill G, MacAlasdair N, Ruis C, Weimann A, Horesh G, Lees JA, et al. Producing polished prokaryotic pangenomes with the Panaroo pipeline. Genome biology. 2020;21:1-21.
[43] Heller DM, Tavag M, Hochschild A. CbtA toxin of Escherichia coli inhibits cell division and cell elongation via direct and independent interactions with FtsZ and MreB. PLoS genetics. 2017;13(9):e1007007.
[44] Melton-Celsa AR. Shiga toxin (Stx) classification, structure, and function. Microbiology spectrum. 2014;2(4):10-1128.
[45] Karmali MA. Prospects for preventing serious systemic toxemic complications of Shiga toxin–producing Escherichia coli infections using Shiga toxin receptor analogues. The Journal of infectious diseases. 2004;189(3):355-9.
[46] Gaudu P, Weiss B. Flavodoxin mutants of Escherichia coli K-12. Journal of Bacteriology. 2000;182(7):1788-93.
[47] Moyano AJ, Tobares RA, Rizzi YS, Krapp AR, Mondotte JA, Bocco JL, et al. A long-chain flavodoxin protects Pseudomonas aeruginosa from oxidative stress and host bacterial clearance. PLoS genetics. 2014;10(2):e1004163.
[48] Okuda S, Katayama T, Kawashima S, Goto S, Kanehisa M. ODB: a database of operons accumulating known operons across multiple genomes. Nucleic acids research. 2006;34(suppl_1):D358-62.
[49] Szklarczyk D, Kirsch R, Koutrouli M, Nastou K, Mehryary F, Hachilif R, et al. The STRING database in 2023: protein–protein association networks and functional enrichment analyses for any sequenced genome of interest. Nucleic acids research. 2023;51(D1):D638-46.
[50] Abramson J, Adler J, Dunger J, Evans R, Green T, Pritzel A, et al. Accurate structure prediction of biomolecular interactions with AlphaFold 3. Nature. 2024:1-3.
[51] Evans R, O'Neill M, Pritzel A, Antropova N, Senior A, Green T, et al. Protein complex prediction with AlphaFold-Multimer. biorxiv. 2021:2021-10.
[52] Bottomley MJ, Muraglia E, Bazzo R, Carfi A. Molecular insights into quorum sensing in the human pathogen Pseudomonas aeruginosa from the structure of the virulence regulator LasR bound to its autoinducer. Journal of Biological Chemistry. 2007;282(18):13592-600.
[53] Bazire A, Shioya K, Soum-Soutéra E, Bouffartigues E, Ryder C, Guentas-Dombrowsky L, et al.. The sigma factor AlgU plays a key role in formation of robust biofilms by nonmucoid Pseudomonas aeruginosa. American Society for Microbiology; 2010.
[54] Baraquet C, Murakami K, Parsek MR, Harwood CS. The FleQ protein from Pseudomonas aeruginosa functions as both a repressor and an activator to control gene expression from the pel operon promoter in response to c-di-GMP. Nucleic acids research. 2012;40(15):7207-18.
[55] Zhang R, Ou HY, Zhang CT. DEG: a database of essential genes. Nucleic acids research. 2004;32(suppl_1):D271-2.
[56] Grazziotin AL, Vidal NM, Venancio TM. Uncovering major genomic features of essential genes in Bacteria and a methanogenic Archaea. The FEBS journal. 2015;282(17):3395-411.
[57] Zhang X, Peng C, Zhang G, Gao F. Comparative analysis of essential genes in prokaryotic genomic islands. Scientific Reports. 2015;5(1):12561.
[58] Deng J, Deng L, Su S, Zhang M, Lin X, Wei L, et al. Investigating the predictability of essential genes across distantly related organisms using an integrative approach. Nucleic acids research. 2011;39(3):795-807.
[59] Ravishankar S, Ambady A, Swetha RG, Anbarasu A, Ramaiah S, Sambandamurthy VK. Essential-

ity assessment of cysteinyl and lysyl-tRNA synthetases of Mycobacterium smegmatis. PLoS One. 2016;11(1):e0147188.

[60] Kanehisa M, Sato Y, Kawashima M, Furumichi M, Tanabe M. KEGG as a reference resource for gene and protein annotation. Nucleic acids research. 2016;44(D1):D457-62.

[61] Madin J, Nielsen D, Brbic M, Corkrey R, Danko D, Edwards K, et al.. A synthesis of bacterial and archaeal phenotypic trait data. Sci Data 7: 170; 2020.

[62] Weimann A, Mooren K, Frank J, Pope PB, Bremges A, McHardy AC. From genomes to phenotypes: Traitar, the microbial trait analyzer. MSystems. 2016;1(6):10-1128.

[63] Brbić M, Piškorec M, Vidulin V, Kriško A, Šmuc T, Supek F. The landscape of microbial phenotypic traits and associated genes. Nucleic acids research. 2016:gkw964.

[64] Shrikumar A, Greenside P, Kundaje A. Learning important features through propagating activation differences. In: International conference on machine learning. PMlR; 2017. p. 3145-53.

[65] Galperin MY, Mekhedov SL, Puigbo P, Smirnov S, Wolf YI, Rigden DJ. Genomic determinants of sporulation in Bacilli and Clostridia: towards the minimal set of sporulation-specific genes. Environmental microbiology. 2012;14(11):2870-90.

[66] Rajagopala SV, Titz B, Goll J, Parrish JR, Wohlbold K, McKevitt MT, et al. The protein network of bacterial motility. Molecular systems biology. 2007;3(1):128.

[67] King GM, Weber CF. Distribution, diversity and ecology of aerobic CO-oxidizing bacteria. Nature Reviews Microbiology. 2007;5(2):107-18.

[68] Guillot C, Moran Jr CP. Essential internal promoter in the spoIIIA locus of Bacillus subtilis. Journal of bacteriology. 2007;189(20):7181-9.

[69] Zeytuni N, Flanagan K, Worrall L, Massoni S, Camp A, Strynadka N. Structural and biochemical characterization of SpoIIIAF, a component of a sporulation-essential channel in Bacillus subtilis. Journal of Structural Biology. 2018;204(1):1-8.

[70] Chary VK, Meloni M, Hilbert DW, Piggot PJ. Control of the expression and compartmentalization of σG activity during sporulation of Bacillus subtilis by regulators of σF and σE. Journal of bacteriology. 2005;187(19):6832-40.

[71] Duncan L, Losick R. SpoIIAB is an anti-sigma factor that binds to and inhibits transcription by regulatory protein sigma F from Bacillus subtilis. Proceedings of the National Academy of Sciences. 1993;90(6):2325-9.

[72] Miyao A, Theeragool G, Takeuchi M, Kobayashi Y. Bacillus subtilis spoVE gene is transcribed by sigma E-associated RNA polymerase. Journal of bacteriology. 1993;175(13):4081-6.

[73] Popham DL, Stragier P. Cloning, characterization, and expression of the spoVB gene of Bacillus subtilis. Journal of bacteriology. 1991;173(24):7942-9.

[74] Pridmore RD, Berger B, Desiere F, Vilanova D, Barretto C, Pittet AC, et al. The genome sequence of the probiotic intestinal bacterium Lactobacillus johnsonii NCC 533. Proceedings of the National Academy of Sciences. 2004;101(8):2512-7.

[75] Guinane CM, Kent RM, Norberg S, Hill C, Fitzgerald GF, Stanton C, et al. Host specific diversity in Lactobacillus johnsonii as evidenced by a major chromosomal inversion and phage resistance mechanisms. PLoS One. 2011;6(4):e18740.

[76] Zhang W, Wang J, Zhang D, Liu H, Wang S, Wang Y, et al. Complete genome sequencing and comparative genome characterization of Lactobacillus johnsonii ZLJ010, a potential probiotic with health-promoting properties. Frontiers in Genetics. 2019;10:812.

[77] Johnson A, Miller EA, Weber B, Figueroa CF, Aguayo JM, Johny AK, et al. Evidence of host specificity in

Lactobacillus johnsonii genomes and its influence on probiotic potential in poultry. Poultry science. 2023;102(9):102858.

[78] Dertli E, Colquhoun IJ, Gunning AP, Bongaerts RJ, Le Gall G, Bonev BB, et al. Structure and biosynthesis of two exopolysaccharides produced by Lactobacillus johnsonii FI9785. Journal of Biological Chemistry. 2013;288(44):31938-51.

[79] Fisher RA. On the interpretation of χ 2 from contingency tables, and the calculation of P. Journal of the royal statistical society. 1922;85(1):87-94.

[80] Hutchison III CA, Chuang RY, Noskov VN, Assad-Garcia N, Deerinck TJ, Ellisman MH, et al. Design and synthesis of a minimal bacterial genome. Science. 2016;351(6280):aad6253.

[81] Stephanopoulos G, Aristidou AA, Nielsen J. Metabolic engineering: principles and methodologies. 1998.

[82] Cameron DE, Bashor CJ, Collins JJ. A brief history of synthetic biology. Nature Reviews Microbiology. 2014;12(5):381-90.

[83] Wilson DF, Harrison DK, Vinogradov A. Mitochondrial cytochrome c oxidase and control of energy metabolism: measurements in suspensions of isolated mitochondria. Journal of Applied Physiology. 2014;117(12):1424-30.

[84] Schultz AC, Nygaard P, Saxild HH. Functional analysis of 14 genes that constitute the purine catabolic pathway in Bacillus subtilis and evidence for a novel regulon controlled by the PucR transcription activator. Journal of bacteriology. 2001;183(11):3293-302.

[85] Goh EB, Bledsoe PJ, Chen LL, Gyaneshwar P, Stewart V, Igo MM. Hierarchical control of anaerobic gene expression in Escherichia coli K-12: the nitrate-responsive NarX-NarL regulatory system represses synthesis of the fumarate-responsive DcuS-DcuR regulatory system. Journal of bacteriology. 2005;187(14):4890-9.

[86] Pinske C, Sawers RG. Anaerobic formate and hydrogen metabolism. EcoSal Plus. 2016;7(1):10-1128.

[87] Chen S, Kaufman MG, Miazgowicz KL, Bagdasarian M, Walker ED. Molecular characterization of a cold-active recombinant xylanase from Flavobacterium johnsoniae and its applicability in xylan hydrolysis. Bioresource technology. 2013;128:145-55.

[88] Sun J, Yao C, Wang W, Zhuang Z, Liu J, Dai F, et al. Cloning, expression and characterization of a novel cold-adapted β-galactosidase from the deep-sea bacterium Alteromonas sp. ML52. Marine drugs. 2018;16(12):469.

[89] Pinney MM, Mokhtari DA, Akiva E, Yabukarski F, Sanchez DM, Liang R, et al. Parallel molecular mechanisms for enzyme temperature adaptation. Science. 2021;371(6533):eaay2784.

A contextualised protein language model reveals the functional syntax of bacterial evolution

Supplementary Material

A. Safety and ethics discussion 24
B. Methods 25
B.1. Model architecture . . . 25
B.1.1 Input embeddings . . . 25
B.1.2 Transformer block . . . 26
B.2. Pretraining . . . 27
B.3. Data . . . 29
B.3.1 MAG corpus . . . 29
B.3.2 Complete-genome corpus . . . 30
B.3.3 Embedding genomes with a protein language model . . . 30
B.3.4 Long-read RNA sequencing for operon identification . . . 30
B.4. Downstream tasks & analysis . . . 31
B.4.1 Strain clustering . . . 31
B.4.2 Operon identification . . . 33
B.4.3 Protein-protein interaction prediction . . . 36
B.4.4 Protein annotation . . . 37
B.4.5 Phenotypic traits prediction . . . 39
B.4.6 *De novo* genome generation . . . 41
C. Supplementary figures 46

A. Safety and ethics discussion

Tools like *Bacformer* provide genome-wide insights into bacteria, holding the potential to drive progress in sustainable bio-production, antimicrobial discovery, and live-bacterial therapeutics. These same capabilities, however, could also enable the creation of strains with increased virulence, immune evasion, or drug resistance. As DNA synthesis becomes more accessible and bio-foundries expand, the gap between digital genome design and the creation of real-world organisms continues to narrow.

We have chosen to make *Bacformer* open-source to promote scientific advancement and equitable access. At the same time, we would like to stress the importance of ensuring the safe use of such tools. Because *Bacformer* is a computational tool, it is the subsequent potential knowledge gained from its output to produce live bacteria that falls under regulatory oversight.

In the United States, oversight of live bacteria and microbial engineering is primarily provided by the Centres for Disease Control (CDC) and National Institute of Health (NIH) through guidelines into Recombinant or Synthetic Nucleic Acid Molecules [1], Biosafety in Microbiological and Biomedical Laboratories (BMBL) [2], the Federal Select Agent Program [3], and the May 2025 Executive Order on gain of function research [4]. In Europe, the European Food Safety Authority (EFSA) oversees genetically modified organisms [5] under Directive 2001/18/EC [6] and related regulations. Internationally, the Cartagena Protocol on Biosafety under the Convention on Biological Diversity [7] offers a framework for the safe handling of living modified organisms.

These regulatory bodies focus on physical organisms, gain-of-function research and laboratory practices. Ongoing engagement with regulatory and public health bodies is essential as computational tools for genome engineering evolve beyond existing frameworks. All publications and uses of *Bacformer* and similar tools should clearly acknowledge the limitations of open-source distribution and any current regulatory gaps. Users are responsible for ensuring compliance with all applicable regulations.

As whole-genome models grow in capability, engineering entire organisms will become faster and more accessible. It is crucial to consider integrating safeguards as these tools evolve. For example, blocking high-risk queries (such as creating highly pathogenic bacteria) may help prevent misuse. Data sets with potential for harm should undergo multi-stakeholder review involving academia, industry, and public health experts before public release. All organism design must comply with existing regulations, with frameworks adapting as technology develops. Embedding these measures throughout development ensures that bacterial genome tools advance responsibly, supporting public health and the bioeconomy while reducing inequity, ecological harm, and security risks.

B. Methods

B.1. Model architecture

The *Bacformer* architecture is designed to mirror how bacterial genomes are structured. It represents complete genomes consisting of different replicons, i.e. the primary chromosome alongside any chromids or plasmids, as well as fragmented isolate genomes and metagenome assembled genomes (MAGs) distributed across multiple contigs, while explicitly capturing the relationships between neighbouring genes.

Bacformer utilises a pretrained protein language model (pLM) to generate embeddings for each protein present in the bacterial genome. These protein-level vectors are then passed to a genome-scale transformer that contextualises every representation across the bacterial genome.

B.1.1. Input embeddings

Every input sample to *Bacformer* is an ordered set of proteins from a bacterial genome (an ordered sequence of genes on one or multiple contigs).

Bacformer represents each bacterial genome as a set of protein sequences in context of its ordered location on the genome. To do so, each protein is treated as a token, and tokens combine information from (1) a protein embedding, (2) a contig (also referred to as chromosome or plasmid) embedding, and (3) a relative position embedding to encode relative distance between proteins, using rotary positional embeddings [8].

To create these, each genome $G = (P_1, P_2, \ldots, P_N)$ is first converted into a list of proteins ordered by their coordinates on the bacterial chromosome, plasmid(s) or contigs.

Protein embeddings. A pretrained pLM maps every protein sequence P_i to a matrix of residue embeddings; to get a single vector for a protein we average the residue embeddings,

$$\mathbf{e}_i^{\text{prot}} = \frac{1}{L_i} \sum_{j=1}^{L_i} \mathbf{h}_{i,j} \in \mathbb{R}^D,$$

yielding a fixed-length vector of dimension D. We use ESM-2 (12 layers, 35M parameters) [9] as the base protein language model. We opted for a relatively small pLM due to the computational costs of embedding almost 3B protein sequences.

Contig embeddings. To encode the order of proteins and their location on the genome, we represented contigs using a *contig embedding* $\mathbf{e}_k^{\text{contig}}$ for each of the K contigs that are part of the bacterial genome. As each contig embedding ($\mathbf{e}_k^{\text{contig}}$) is the same for all proteins on the contig, the model can therefore distinguish, for instance, plasmid-specific gene clusters.

For a protein P_i residing on contig c_i we form a token vector

$$\mathbf{x}_i = \mathbf{e}_i^{\text{prot}} + \mathbf{e}_{c_i}^{\text{contig}}.$$

Special learnable tokens are appended to mark global structure: `[CLS]` at the beginning, `[SEP]` between contigs, and `[END]` at the genome terminus. The ordered sequence

`[CLS]`, $\mathbf{x}_1, \ldots, \mathbf{x}_N$, `[END]`

preserves native synteny, allowing the transformer to represent functional dependencies such as operons [10].

Rotary positional embeddings. The rotary position embeddings [8] are incorporated in the attention mechanism to encode relative distances between proteins, enabling the model to capture genomic context. This is essential in bacteria, where genes are often organized in operons—compact, co-transcribed clusters whose function and regulation depend on precise ordering and spacing. Rotary embeddings are used instead of sinusoidal schemes to avoid length limitations.

Below we show an example of order in a genomic sequence for complete genomes and MAGs. `chr`, `plasm` and `contig` refer to chromosome, plasmid, and contig, respectively.

Complete genome:
[CLS] chr1_P_1 ... chr1_P_{N_1} [SEP] plasm1_P_{N_1+1} ... plasm1_P_N [END]

MAG:
[CLS] cont1_P_1 ... cont1_P_{N_1} [SEP] cont2_P_{N_1+1} ... cont2_P_{N_2} [SEP] ... contZ_P_N [END]

During pretraining on MAGs, we randomize the order of contigs within every bacteria for every epoch to prevent the model from learning spurious signals from the arbitrary contig ordering produced by the metagenomic assembly.

Finally, each token is processed by a shared linear layer, followed by layer normalization and dropout (with a rate of 0.2). The resulting sequence of embeddings for each bacterial genome is then fed back into the transformer, which updates each protein representation by attending to all other proteins in the genome, thereby capturing their genomic context and relationships.

B.1.2. Transformer block

Each bacterial genome, consisting of N proteins, is fed into a 12-layer, 8-head transformer with a hidden size of 480. This produces protein embeddings conditioned on all other proteins in the genome, yielding contextualised representations that capture protein interactions within their genomic context.

We employ rotary positional embeddings (RoPE) [8] within the self-attention mechanism and use `FlashAttention` in `PyTorch` for memory-efficient training and inference. The transformer is

trained from scratch. For downstream tasks, we use the final hidden state of each protein, while the mean of all token representations provides a fixed-length embedding of the whole genome.

B.2. Pretraining

We sought to learn the relationships between proteins present in the genome, leveraging billions of years of bacterial evolution. All large language models (LMs), including protein LMs and DNA LMs, rely on a discrete vocabulary set for pre-training. In *Bacformer*, each token is a protein, and the universe of bacterial proteins is effectively unbounded.

To construct a finite protein-family vocabulary, we generated ESM-2 protein embeddings [9] for millions of proteins across diverse organisms. These embeddings were clustered using k-means clustering [11] to define a finite protein-family vocabulary. After experimenting with various values of k, we set it to 50,000 centroids, corresponding to the size of our protein dictionary. The CuPy library [12] was used to perform the clustering efficiently.

We evaluated our k-means clustering approach by comparing it with MMseqs2, a widely used protein-clustering method [13]. We randomly sampled 12.3 million proteins from species-representative genomes, clustered them using MMseqs2, and assessed concordance between the MMseqs2 and k-means clusters. The normalized mutual information (NMI) score was 0.88, indicating strong agreement between the two methods. A key advantage of k-means in *Bacformer* is that it allows direct control over the number of clusters, enabling us to define a fixed-size vocabulary for pre-training.

Pre-training relies on the quality of these unsupervised clusters, which should reflect evolutionary relationships. Therefore, after clustering, we performed quality checks by evaluating distances to centroids and the number of proteins per cluster, and removed labels for proteins exceeding a distance threshold.

To further validate that our clusters reflected evolutionary relationships, we used the eggNOG 5.0 database [14], which provides protein orthology annotations across a wide range of organisms. We obtained eggNOG annotations from *MGnify* [15] for 12,630 representative genomes, extracting annotations for all proteins with an available eggNOG label. We then compared these orthology-based annotations to our k-means cluster assignments and observed strong concordance, with an NMI score of 0.74 (where 0 indicates no mutual information and 1 indicates perfect agreement). This high NMI demonstrates that our k-means clusters capture biologically significant groupings of proteins.

Having established the quality and biological relevance of our k-means clusters (hereafter "protein clusters"), we used these 50,000 groups—derived from clustering ESM-2 embeddings—as supervision targets during pre-training. While distinct from curated families such as those in Pfam, these clusters offer an alternative, unsupervised representation of protein families.

During pre-training, *Bacformer* receives amino acid sequences as input and learns to predict the protein cluster label for a masked protein using a two-layer multi-layer perceptron (MLP) trained

jointly with the transformer. The protein cluster labels are used only for supervision; *Bacformer*'s contextual protein embeddings (henceforth, *Bacformer* embeddings) are learned directly from sequence and are influenced by both single-residue mutations and genomic context, including the arrangement of surrounding proteins and contigs. The initial protein clusters serve as informative labels but do not restrict the flexibility of the learned representations.

Bacformer embeddings can be used for a variety of downstream tasks. The MLP classification head can be retrained or replaced to predict different labels, such as bacterial phenotypes. As *Bacformer* integrates both sequence and genomic context, it can distinguish between proteins that are similar in sequence but differ in their genomic neighborhoods, and hence can capture the functional impact of allelic variation.

Training objective. We trained two versions of *Bacformer* with different self-supervised objectives: (1) an autoregressive next-protein-family prediction (generative) objective and (2) a masked-prediction objective. In practice these approaches confer complementary strengths on downstream tasks. Each model contains 12 transformer layers followed by a classification head that predicts the protein-family label from the final layer's output.

The masking procedure resembles the standard BERT-style scheme [16], adapted to our setting: (*i*) 15 % of the proteins are selected for masking. (*ii*) In 87.5 % of those cases the selected tokens are replaced by `[MASK]`. (*iii*) In the remaining 12.5 % the selected tokens are left unchanged.

Pre-training details. We first pre-trained both models on a corpus of almost ~1.3M metagenome-assembled genomes (MAGs). Training was performed on a mixture of NVIDIA A100 and Intel Ponte Vecchio (PVC) GPUs for up to 1.2 M steps; running on four A100s required approximately two weeks. The effective batch size was 32. We used a linear learning-rate schedule with a 10 % warm-up, ramping to a peak of 0.00015 per GPU; the rate was then scaled by the square root of the total number of GPUs. Optimisation employed Adam [17] with weight decay 0.01 and $\beta = (0.9, 0.999)$. All tensors used the `bfloat16` format. Models were implemented in `PyTorch` [18] and trained with the `HuggingFace Transformers` library [19].

After this stage we continued pretraining on a held-out set of $48,932$ complete genomes. We separated the two stages because MAGs usually consist of multiple contigs whose order is unknown, whereas complete genomes comprise an uninterrupted chromosome (or set of replicons) with a well-defined gene order, which can be fed to the model as a single contiguous sequence. For every genome we capped the maximum number of proteins at $6,000$, a value that exceeds the protein count of more than 99% of genomes in our corpus.

All pre-trained models are available for zero-shot use or fine-tuning via `HuggingFace`[1]. Code for

[1] https://huggingface.co/collections/macwiatrak/bacformer-681a17d6a77a928a1531def2

pre-training is provided in our GitHub repository: https://github.com/macwiatrak/Bacformer.

Pre-training performance analysis. During training, we monitored the validation loss and selected the checkpoint with the lowest value. We then evaluated pre-training performance by running the model on the validation set at regular intervals, recording accuracy for the masked-prediction objective and perplexity for the autoregressive objective. In both cases performance improved steadily with additional steps, demonstrating that the models generalise to unseen proteins (i.e. zero-shot; Fig. S1).

To probe this further we conducted an ablation study across taxonomic ranks in the zero-shot setting. *Bacformer* retains high accuracy on unseen species, but performance declines for genomes that belong to entirely unseen genera, families or orders (Fig. S1) as the greater evolutionary distance of these taxa reduces overlap in k-mer composition, gene content and synteny between the training set and the target genomes.

We next examined the effect of genome completeness. MAGs are often fragmented and may lack sizeable genomic segments; therefore, the model must be robust to varying completeness levels. Pre-training results show that *Bacformer* benefits from more complete assemblies. However, it remains relatively robust until completeness drops below 75 % (Fig. S1). The same trend appears when comparing MAGs with complete genomes (Fig. S2): access to the full sequence and an intact chromosome without contig breaks markedly boosts accuracy.

Finally, we investigated scaling. A variant trained on only 25% of the data and with 6 (rather than 12) layers performed noticeably worse, indicating that *Bacformer* benefits from both additional training data and model capacity; further scaling is therefore likely to yield significant gains (Fig. S2).

B.3. Data

We collated a diverse set of bacterial genomes spanning the tree of life and a wide range of environments. Our guiding principle was to maximise phylogenetic and ecological diversity so that the pretrained model would be broadly applicable to downstream tasks in bacterial genomics.

B.3.1. MAG corpus

For pre-training we assembled the MAG corpus from two complementary resources: *MGnify* v2.0 [15] and *SPIRE* [20]. *MGnify* provides seven biome-specific genome catalogues that together contain 306, 596 non-redundant metagenome-assembled or isolate genomes representing 11, 048 species. *SPIRE* provides 99, 146 consistently processed whole-genome shotgun metagenomic samples drawn from 739 studies worldwide and 1, 008, 689 MAGs.

We downloaded the assemblies containing protein annotations and associated metadata from both resources. Protein order within each contig was preserved so that positional encodings could capture synteny. Lineage labels (species, genus, family) and contextual metadata (sample identifier, biome, etc.) were processed as provided by *MGnify* and *SPIRE*.

The final MAG corpus comprises 1, 314, 344 genomes, totalling 2, 700, 709, 306 proteins, 220, 207, 650 contigs, 24, 663 species and 70 distinct environments.

B.3.2. Complete-genome corpus

Many downstream applications—e.g. comparative genomics, gene-context analysis and operon prediction—benefit from access to complete isolate genomes. Unlike MAGs, which are reconstructed *in silico* from a meta-genome-wide assembly and are often fragmented, incomplete or contaminated, complete genomes are obtained from clonal cultures and typically consist of a single circular chromosome (and any additional chromids or plasmids) with high continuity and accuracy [21]. To capture these characteristics, we downloaded available bacterial assemblies from NCBI RefSeq [22] and GenBank [23]. After removing duplicates we retained 48,932 unique complete genomes, each with its translated proteome and annotation files. The metadata (i.e assembly accessions) were stored alongside the sequences. The NCBI collection is biased towards clinically important pathogens—for example, *Salmonella enterica*, *Staphylococcus aureus* and *Mycobacterium tuberculosis*. The combined corpus, consisting of the MAG and complete-genome corpora therefore encompasses a heterogeneous yet complementary training set that balances ecological breadth with assembly quality, positioning the model to generalize across both environmental and clinical bacterial genomics, despite the current available datasets being biased with broad sections of the global bacterial ecosystem under-represented.

B.3.3. Embedding genomes with a protein language model

Bacformer represents each genome by the average of its protein embeddings computed with a pretrained protein language model (pLM). To avoid the cost of re-embedding every protein at every training epoch, we generated all embeddings once, off-line, and cached the resulting protein embeddings.

We chose the 12-layer, 35 M-parameter variant of *ESM-2* [9] because it offers a good balance between accuracy and computational efficiency when compared with larger pLMs such as *ESM-2-650M* or *ProtT5-XL* [9, 24]. Embedding the entire corpus of ~1.3 million genomes required approximately 30 days on a cluster equipped with 16 Intel Ponte Vecchio GPUs, and produced ~5.6 TB of embedding files.

B.3.4. Long-read RNA sequencing for operon identification

To test *Bacformer*'s ability to predict operons, we performed long-read RNA sequencing to annotate operons across five distinct strains, processing three independent biological replicates for each strain.

Bacterial strains and culture conditions. Five bacterial strains were used in this experiment: *Staphylococcus aureus RN450* (*S. aureus* RN450), *Mycobacterium abscessus* ATCC 19977 (*M. ab*), ΔleuD ΔpanCD *Mycobacterium tuberculosis* H37Rv 102J23 (ΔleuD ΔpanCD *M. tb*), *Pseudomonas aeruginosa* PAO1, and *Escherichia coli DH5α*. Each strain was cultured in triplicate under nutrient-rich conditions until mid-exponential phase (OD_{600} = 0.4–0.6) was reached.

RNA isolation. Cells were harvested by centrifugation and processed for total RNA extraction using the MasterPure Complete DNA and RNA Purification Kit (Lucigen) with strain-specific modifications: For *S. aureus* RN450, cell pellets were pre-treated with lysostaphin (Tris buffer, pH 8.0) at 37 °C for 30 min to aid lysis. For the mycobacterial strains (*M. ab* and ΔleuD ΔpanCD *M. tb*), cells were mechanically disrupted by bead beating in lysis buffer, followed by extraction with the standard kit protocol. Isolated RNA was treated twice with TURBO DNase (Invitrogen) to remove residual genomic DNA and purified using RNA Clean & Concentrator columns (Zymo Research). RNA integrity was assessed on an Agilent TapeStation RNA ScreenTape system, and concentrations were measured with a Qubit fluorometer (Invitrogen).

Library preparation and sequencing. For each replicate, 1,000 ng of total RNA underwent rRNA depletion with riboPOOLs (siTOOLs Biotech). The depleted RNA was polyadenylated with poly(A) polymerase (PAP) in the presence of a manganese catalyst, adding 50–90 adenosines per molecule. cDNA libraries were prepared with the Nanopore cDNA-PCR kit, pooled and sequenced on a PromethION device equipped with R10 flow cells (Oxford Nanopore Technologies).

B.4. Downstream tasks & analysis

B.4.1. Strain clustering

We tested *Bacformer*'s ability to define strains in an unsupervised manner from within the most common bacterial species (n=20), genera (n=10) and families (n=7) in our MAG corpus. Because *Bacformer* takes only the ordered proteins present in a genome as input — without access to any species labels — it can be used in a fully unsupervised manner to cluster within each pre-defined set.

To cluster assemblies into strains we averaged all vector contextural protein embeddings for each MAG (assembly) from our *Bacformer* model (pre-trained with autoregressive training). We then performed *Leiden* clustering of these average embedding vectors [25] , at resolution 0.5, with `scanpy` to identify near-identical genomes. This yielded 199 195, 314 148 and 647 437 clusters in the species-, genus- and family-level data sets, respectively which we henceforth refer to as strains.

We benchmarked the results against an *ESM-2* baseline [9], where protein embeddings are non-contextual. For both *Bacformer* and ESM, three clustering metrics were computed:
(1) Average Silhouette Width (ASW; range $[-1, 1]$),

(2) Adjusted Rand Index (ARI; range $[0, 1]$), and
(3) Normalised Mutual Information (NMI; range $[0, 1]$).

For all three metrics, a higher score reflects better performance and across all taxonomic ranks and on all three metrics *Bacformer* significantly outperformed *ESM-2* (Fig. 1C) and Fig. S3).

As expected, the accuracy of strain clustering deteriorated at coarser taxonomic levels (ASW decreased) when including assemblies from wider genomic distributions of genus and family, than from within species alone.

Evolutionary relationships. We proceeded to test if *Bacformer* representations recognise similarity of species by their families and genus. To do so, we took assemblies in the most common order of bacteria in *MGnify* [15], *Bacteroidales* ($N = 187,402$). A species representation was obtained by averaging the embeddings of all MAGs belonging to that species and calculated cosine similarities between the 100 most prevalent species (Fig. 1D). Species representations from *Bacformer* retained high similarity within taxonomic groups, achieving an average cosine similarity of 0.96 (SD = 0.02) for species in the same genus and 0.93 (SD = 0.05) for species in the same family, versus 0.81 (SD = 0.03) and 0.80 (SD = 0.03) for species from different genera and families, respectively.

Revealing sub-populations in *Escherichia coli*. To test *Bacformer*'s ability to identify subpopulations within a species, we clustered embeddings from all 7,585 *Escherichia coli* strains in *MGnify*. For each strain, we computed an average embedding across its assemblies. Clustering these strain-level embeddings using the *Leiden* algorithm revealed six distinct groups (Fig. 1E). In contrast, embeddings from *ESM-2* did not form clear clusters (Fig. S4).

To exclude the possibility that the observed structure arose from technical batch effects, we asked whether strains grouped by sample or study identifier (Fig. S4). The ASW for both metadata labels was negative, indicating no such grouping and confirming that *Bacformer* captures genuine biological structure.

Having identified the sub-populations, we sought the genes specific to each cluster. Using *Panaroo* [26], which provides a binary gene-presence matrix, we performed Fisher's exact tests [27] comparing each cluster against all others. Genes significantly enriched in a cluster ($p < 0.05$) were considered differentially abundant; the top five per cluster are shown in (Fig. S4). Examples include *cbtA* (cluster 2; inhibits cell division and elongation [28]), *stxA* (cluster 5; Shiga toxin production [29]) and *fldP* (cluster 3; adaptation to low oxygen [30, 31]). This demonstrates that *Bacformer* embeddings can uncover sub-populations with functionally distinct gene complements.

Variation within *E. coli* is often driven by the accessory genome [32]—genes present in only a subset of strains. Defining rare genome as all genes present in $< 5\,\%$ of all strains, we found that 41% of

cluster-specific genes were rare. Given that the median genome of *Escherichia coli* contains 112 rare genes versus 3,718 core genes (those present in> 95% of all strains), this represents a 23-fold enrichment ($p < 10^{-20}$), highlighting the role of the accessory genome in shaping sub-populations and the utility of *Bacformer* for their discovery.

Scalable search for similar bacteria. Identifying similar bacterial genomes from unannotated sequences is a fundamental challenge in microbiology, underpinning our understanding of microbial diversity, evolutionary relationships, and ecological functions. Traditional methods struggle with the scale and complexity of bacterial genomes, particularly given the prevalence of horizontal gene transfer and the often limited functional annotation of newly sequenced organisms.

Bacformer directly addresses this challenge by learning genome representations that embed all 1,314,344 genomes from the MAG corpus—along with their metadata—into a unified, scalable database. This enables robust similarity search and comprehensive taxonomic classification, allowing for rapid identification of phylogenetically related bacteria and inference of environmental origins, which are critical for comparative genomics and environmental microbiology.

To rigorously validate that *Bacformer*'s representations capture true biological structure and enable reproducible classification, we combined unsupervised clustering with k-nearest-neighbour (kNN) classification across taxonomic ranks. By holding out 10% of genomes (with $k = 10$ and filtering for species with at least five representatives), we demonstrated that *Bacformer* achieves balanced accuracies of 0.83 ± 0.32, 0.94 ± 0.17, and 0.95 ± 0.16 at the species, genus, and family levels, respectively. These results show that *Bacformer*'s embeddings naturally reflect taxonomic groupings and generalize to unseen genomes. This validated functionality is openly available on github (`https://github.com/macwiatrak/Bacformer`), enabling researchers worldwide to analyze unannotated bacterial genomes with unprecedented scale and accuracy.

B.4.2. Operon identification

Attention-weight analysis. To evaluate how the model utilises gene-neighbourhood information during pre-training, we analysed the attention weights of a model trained with a masked-language objective on complete genomes. Attention scores were collected for 45 diverse organisms with known operons recorded in OperonDB [33].

In *Pseudomonas aeruginosa* PAO1 the model assigns its highest attention to the five closest neighbouring genes (Fig. S5). and layer-wise inspection revealed distinct patterns: weights in the initial layers are distributed relatively uniformly across neighbouring proteins, mid-layers concentrate on the immediately adjacent genes, and the final layers still focus on neighbours but shift the strongest weights slightly farther away (Fig. S5). These patterns indicate that successive transformer blocks capture complementary contextual information.

We next examined how *Bacformer* updates neighbouring protein representations. The cosine

similarity between adjacent gene embeddings follows an oscillatory pattern reminiscent of a cosine curve (Fig. S5), demonstrating that the model makes strong use of the rotary positional embeddings that encode genomic position. This observation is consistent with the layer wise attention patterns and underscores the importance of gene neighbourhood.

Zero-shot operon identification with OperonDB. To evaluate *Bacformer*'s zero-shot operon identification capabilities, we used OperonDB [33], a resource containing experimentally verified operons from diverse bacterial genomes. OperonDB genomes are often only partially annotated, with missing or inconsistent information regarding gene coordinates, strand orientation, or gene order. As a result, when calculating the operon score, we restricted ourselves to features that were reliably available for all genes—specifically, those derived from protein sequences—while excluding genomic features such as intergenic distance.

To predict operons, we calculated an 'operon score'. The first step in this score was computing the pairwise cosine similarity between the final-layer embeddings of all gene pairs. For a set of colocated genes of size n, the operon score then defined as the average of these pairwise similarity scores across the n genes, providing a single summary value that reflects the overall similarity among proteins encoded within the colocated genes.

For evaluation, operons annotated in OperonDB were used as positive examples. However, because OperonDB's genome annotations are incomplete, regions without operon annotation cannot be confidently assumed to be true negatives. To address this, negative examples were generated by randomly sampling contiguous gene sets from the same genomes, using a 1:10 ratio of positive (annotated operon) to negative (random gene set) samples. To further minimize the likelihood of including true, but unannotated, operons among the negatives, this random sampling was repeated five times with different random seeds, and results were averaged across these replicates.

Evaluation of prediction performance was stratified by operon size (number of genes) (Fig. S6) and five species with the highest number of annotated operons in OperonDB (Fig. S6). Bacfomer outperformed ESM on both levels of stratification and across all strata. Performance varied considerably depending on operon size and species, which likely reflects the differing numbers of verified operons of various lengths and across organisms that are incompletely annotated in OperonDB (Fig. S6).

Zero-shot operon prediction using long-read RNA sequencing. To further test *Bacformer*'s ability to predict operons directly from experimental data, we leveraged the long-read RNA sequencing datasets generated in our study. These datasets, described in detail in the previous section, provide high-resolution transcriptomic profiles across five diverse bacterial strains, each processed in three independent biological replicates (Fig. S6).

For each sequenced strain, the following genome assemblies and gene annotations from NCBI RefSeq [ref] were used: `GCF_000013425.1` (*Staphylococcus aureus RN450*), `GCF_000069185.1` (*Mycobacterium abscessus ATCC 19977*), `GCF_000195955.2` (ΔleuD ΔpanCD *Mycobacterium tuberculosis H37Rv 102J23*), `GCF_000006765.1` - (*Pseudomonas aeruginosa PAO1*), and `GCF_002899475.1` (*Escherichia coli DH5α*).

ONT reads from each replicate were polished with Pychopper (v2.7.10), polyA tails longer than 10 bases and sequencing adapters were trimmed using cutadapt and mapped against the genome assemblies using Minimap2 (v2.29)[34] and Samtools (v1.22) [35]. Candidate operons were identified from read alignments spanning at least two genes on the same strand and then extended by combining overlapping candidates at most 50 base pairs apart. Operons were then collated from the triplicates for each strain and used as our operon annotations.

For long read data, we formulated operon prediction as a binary boundary classification problem, evaluating whether two contiguous genes form an operon. To address this, we developed a method that incorporates three features: (1) *Bacformer* contextualized protein embeddings, (2) the distance between the two genes, and (3) the strand of each gene.

First, we used the contextualized protein embeddings to compute the cosine similarity between adjacent genes. These embeddings were extracted from the last hidden layer of the *Bacformer* transformer encoder, which was pretrained with a masked objective on complete genomes. Second, the distance between the two genes was calculated as the number of base pairs between the end of one gene and the start of the next, effectively measuring the intergenic region. Third, information on gene start and end positions, as well as gene-strand, was extracted from the genome assemblies.

We then constructed a model that takes these features as input and outputs a binary value indicating whether the gene pair forms an operon, based on:

$$s_i = c_i \, e^{-d_i/\lambda} \, I_{\text{strand}} \, I_{\text{gap}},$$

where

$$c_i = \tfrac{1}{2}\big(1 + \cos(\hat{h}_i, \hat{h}_{i+1})\big), \qquad \lambda = 500 \text{ bp},$$

and

$$I_{\text{strand}} = \begin{cases} 1 & \text{if the two genes are on the same strand,} \\ 0 & \text{otherwise,} \end{cases} \qquad I_{\text{gap}} = \begin{cases} 1 & \text{if } d_i \le 500 \text{ bp,} \\ 0 & \text{otherwise.} \end{cases}$$

Here $\hat{h}_i$ and $\hat{h}_{i+1}$ are the ℓ_2-normalised *Bacformer* embeddings of genes i and $i{+}1$, d_i is their intergenic distance, and $s_i \in [0, 1]$ serves as the operon-membership score; a pair is classified as belonging to the same operon when s_i exceeds a threshold. The exponential factor $e^{-d_i/\lambda}$ reduces the score of pairs separated by larger intergenic distances, while the two indicator terms I_{strand} and I_{gap} act as

a hard veto, nullifying the score if genes are transcribed in opposite directions or if the gap exceeds 500 bp. This reflects the empirical observation that bacterial operons almost always consist of closely spaced, co-oriented genes.

B.4.3. Protein-protein interaction prediction

Protein–protein interaction prediction using STRING DB. We downloaded the complete data archive from the *STRING DB* (v12.0) [36] download site (https://string-db.org/cgi/download). Using the species-metadata file, we selected only bacterial organisms and retrieved their protein sequences together with all available protein–protein interaction scores. After running the download scripts, we obtained 10,533 unique strains spanning 6,956 species. *STRING DB* provides several evidence-channel scores; here we used the *combined*, *co-occurrence*, *neighbourhood* and *co-expression* channels. To binarize the interaction scores we applied a threshold of 0.6 ($\geq$ 0.6 = interaction, < 0.6 = no interaction), resulting in ~10% of all interactions having a positive label. This cut-off was selected empirically by maximising the average AUROC and AUPRC on a validation set. The data were randomly split into training, validation and test sets in a 70/10/20% ratio.

For zero-shot *Bacformer* predictions and evaluation, we computed pairwise cosine-similarity scores between contextualised protein embeddings for every strain in the test set. *Bacformer* embeddings were taken from the last hidden layer of the transformer, pretrained on complete genomes with a masked objective. We benchmarked *Bacformer* predictions for interactions against *ESM-2* [9], whose non-contextual embeddings serve as input to the *Bacformer* transformer (AUROC in Fig. 2D; AUPRC in Fig. S7). Metrics were calculated per interaction type across genomes.

After confirming that *Bacformer* captures protein–protein interactions in a zero-shot setting, we fine-tuned the model by stacking a two-layer multilayer perceptron (MLP) on top of the pretrained encoder. The MLP takes as input the mean of the two contextual protein embeddings and is preceded by a dropout layer (frequency=0.2), followed by a GELU activation [37], a normalisation layer ($\epsilon = 1 \times 10^{-12}$) and another dropout layer (frequency=0.2). The resulting vector for each protein pair is passed through a final binary-classification layer, again preceded by dropout 0.2. Training minimises binary cross-entropy loss; we trained for at most 10 epochs with no early stopping, clipped gradients at 2.0, and used Adam [17] with weight decay 0.01, monitoring validation loss across genomes. Hyperparameters were tuned on the validation set. The fine-tuned model was initialised from the *Bacformer* weights pretrained on complete genomes.

We visualised AUROC (Fig. 2E) and AUPRC (Fig. S7) curves for representative strains from distinct species in the test set using the predicted interaction scores from the fine-tuned *Bacformer* model.

Identifying protein–protein interactions in *Pseudomonas aeruginosa*. To systematically identify potential protein–protein interactions in Pseudomonas aeruginosa PAO1, we employed the fine-tuned *Bacformer* interaction prediction model. To prevent data leakage, the training data excluded

strains from the same genus as *P. aeruginosa* (see Fig. 2G). Interaction probabilities were computed for all possible protein pairs, and only those with a predicted probability of at least 0.5 were retained.

The top 100 scoring protein pairs were then evaluated for direct physical interactions using *AlphaFold3* (Fig. 2H & Fig. S7; [38]). For each pair, we extracted the inter-protein TM-score (ipTM), a metric that quantifies the confidence and quality of the predicted protein complex structure—higher ipTM values indicate a greater likelihood of a true physical interaction.

To benchmark the *Bacformer* predictions, we also randomly sampled 100 neighbouring gene pairs (within five genes of each other) and 100 non-neighbouring gene pairs (separated by more than five genes). All sets—*Bacformer*-predicted and random—were evaluated with *AlphaFold3*, and their ipTM scores were extracted.

The analysis of ipTM scores was stratified by gene proximity: pairs were classified as either "neighbours" (within five genes of each other) or "non-neighbours" (separated by more than five genes). Within each category, the ipTM scores of *Bacformer*-predicted pairs were compared to those of randomly selected pairs using Welch's *t*-test to determine whether *Bacformer*'s predictions were significantly enriched for high-confidence interactions (Fig. S7).

The results demonstrate that *Bacformer* has strong predictive capacity for generating hypotheses about gene-gene interactions and gene networks, offering a valuable guide for experimental validation. This is particularly notable given that the model relies on *STRING DB*, which integrates diverse evidence sources and may include interactions lacking high-quality experimental support, leading to some false positives. *AlphaFold3*'s state-of-the-art computational predictions enable further screening of these predictions.

Uncovering gene networks from contextual embeddings. To test whether *Bacformer*'s contextualised representations predict gene co-expression and co-regulated genes, we compared zero-shot clusters obtained from pairwise cosine similarities between *Bacformer* embeddings with those from experimental co-expression data. Normalised RNA-seq data from 411 samples of P. aeruginosa were obtained from iModulonDB [39, 40] and used to compute a gene–gene co-expression matrix. Equivalent matrices were produced from *Bacformer* and from ESM-embeddings to benchmark *Bacformer* results. Each matrix served as input to the *Leiden* clustering algorithm [25] across multiple resolutions. Cluster similarity between co-expression and model-predicted partitions was quantified using normalised mutual information (NMI, range 0–1; Fig. S8). For qualitative inspection, we extracted clusters at resolution 40 containing key regulators such as lasR (Fig. 2I), fleQ (Fig. S8), and algU (Fig. S8).

B.4.4. Protein annotation

Gene essentiality prediction. Gene-essentiality annotations were downloaded from the *Database of Essential Genes* (*DEG*, `http://origin.tubic.org/deg`) [41]. Corresponding genomes (DNA

and protein) were retrieved from NCBI GenBank. Of the 66 DEG genomes, those with > 98% redundant annotations were removed, leaving 51 genomes across 37 species. Data were randomly split into training, validation and test sets in a 60/20/20% ratio.

In zero-shot knock-out experiments with *Bacformer*, the embedding of the protein of interest was set to zero and the genome-level representation recomputed as the mean of all embeddings, after which cosine similarity to the original genome vector served as the knock-out metric.

The *Bacformer* predictions were benchmarked against *ESM-2* and *Evo* [42]. For *Evo* we followed Nguyen *et al.* (2024) by introducing a premature stop codon into each gene.

A single linear layer preceded by a dropout layer (frequency=0.2) was trained with binary cross-entropy loss to predict gene essentiality (1 = essential, 0 = non-essential) from the protein embeddings; encoder weights were frozen. Training lasted up to 100 epochs with early-stopping patience 10, monitoring macro-AUROC. Adam weight decay was set to 0.02.

For *Evo* embeddings we compared (i) log-likelihood vectors ($z \in \mathbb{R}^N$) produced by the official scoring script and (ii) averaged hidden-state vectors ($\mathbf{x} \in \mathbb{R}^D$, $D = 4,096$) obtained using the *Evo* codebase. Option (i) performed better on the validation set and was used in final benchmarks.

We note that *Evo* outperformed *Bacformer* in the zero-shot case. We hypothesize that this is due to Evo, a DNA-based model, being able to capture the non-coding genome, in contrary to the *Bacformer* which only considers the coding sequence.

AUROC and AUPRC curves per species were generated by selecting five strains from the test set and plotting predicted versus true labels per gene using the fine-tuned *Bacformer* model (Fig. 3C & Fig. S9). To assess the contribution of context alone, we repeated the experiment after masking all essential genes and 20% of non-essential genes, training the same linear layer only on the remaining masked embeddings and comparing results to the unmasked baseline.

To evaluate the non-essential genes with highest predicted probability of gene essentiality, we extracted the binary predictions on two strains from the test set (1) *Staphylococcus aureus*, (NC_002952) and (2) *Salmonella enterica* (NC_003197) using models trained on 3 different random seeds (Fig. S9).

Protein function prediction. We extracted protein function annotations from the *MGnify* catalogue [15], which provides labels for representative strains of each species. Proteins in *MGnify* are annotated with *KEGG* identifiers [43], linking genes to metabolic pathways and other biological processes. Because most of these proteins can be labelled accurately by simple sequence homology methods, we reduced the training set to make the task more challenging and re-assigned the corresponding genomes to the test split. The final data set contained 666 training, 1,331 validation and 1,931 test genomes spanning 188 distinct functions, with splits chosen at random. We used the non-BRITE *KEGG* hierarchy as labels and framed the task as multi-class classification at *KEGG* level 3.

A single linear layer, preceded by a dropout layer (frequency=0.2), was stacked on top of the protein embeddings and trained with cross-entropy loss. Training ran for up to 100 epochs with early stopping (patience 10) based on macro-accuracy on the validation set. The learning rate was tuned per model; optimisation employed Adam [17] with weight decay 0.01.

To quantify the benefit of genomic context, we fine-tuned both the encoder and the linear head for *Bacformer* and compared the result with *ESM-2* embeddings coupled to a linear head. The *Bacformer* encoder was initialised from a model pre-trained on MAGs with a masked objective. For a k-nearest-neighbour baseline we used *ESM-2* embeddings, choosing $k = 3$ on the validation set.

Few-shot performance was assessed by selecting all protein functions in the test set with ≤ 20 training examples, binning them by sample count and plotting accuracy per function (Supp. Fig. S10). We then identified functions showing the largest performance gap between *Bacformer* and *ESM-2*, projected their embeddings to two dimensions with UMAP and computed the average silhouette width (ASW; range –1 to 1, higher is better; Fig. 3E).

Finally, to test prediction using context alone, we randomly masked 50% of proteins at input in each epoch and fine-tuned the model to predict labels for both masked and unmasked proteins. Results were compared with the baseline model trained without masking on the held-out test set.

B.4.5. Phenotypic traits prediction

Unsupervised clustering of bacterial genomes by phenotype. To evaluate whether bacterial genomes cluster by (host) phenotype without supervision, we considered two traits: (1) environmental niche and (2) Gram-stain. For (1) we focused on the *Bifidobacterium* genus, chosen because its members colonise a range of mammalian gastrointestinal tracts—spanning infant, adult and ruminant guts—and must adapt to the distinct nutrient profiles and immune pressures found in each host. Strains were extracted from the *MGnify* data set [15] and 70 were randomly sampled across three environments. For every strain we computed a *Bacformer* genome embedding (model pre-trained on MAGs with a masked objective) and calculated pairwise cosine similarity between all strains.

For (2) we used the annotations from [44] to cluster strains by Gram-stain. After filtering for genomes with a Gram-stain label (see paragraph below for details), 14,052 genomes remained. Each genome was embedded with *Bacformer* pre-trained on complete genomes, the genome-level vector being the mean of all protein embeddings. The embeddings were then projected with UMAP and the two-dimensional plot (Fig. S11) was coloured by Gram-stain.

Phenotypic traits prediction and annotation. Phenotypic traits were compiled from two major sources [44, 45]. Each trait label was prefixed with its source so that provenance is explicit and name collisions are avoided. Duplicate traits present in both catalogues were retained because they enlarge the pool of labelled genomes without forcing the merger of measurements obtained

with different protocols. Using the taxonomy IDs and assembly accessions supplied in the trait data sets, we downloaded the corresponding genome sequences from NCBI RefSeq and GenBank. We restricted ourselves to categorical phenotypes and discarded traits with fewer than 500 genomes, as well as classes with fewer than 50 samples. This produced 139 unique phenotypes across 24,462 genomes. For every phenotype the data were split randomly into 60% training, 20% validation and 20% test sets. Because many traits are rare, results were averaged over five independent splits to obtain stable estimates.

All genomes were embedded with *Bacformer* pre-trained on complete genomes; the frozen genome embeddings served as input to phenotype-specific logistic-regression classifiers, trained with cross-entropy loss for up to 1,000 iterations. Class imbalance was handled by weighting each class according to

$$w_c = \frac{N}{K\, n_c},$$

where n_c is the number of training samples in class c, K the number of classes and N the total number of training samples.

To apply these models to the full MAG corpus we first examined potential distribution shifts, since the classifiers were trained on complete genomes whereas MAGs are fragmented and sometimes incomplete. *Bacformer* embeddings were generated for every MAG and each phenotype model was re-trained on the MAG-derived vectors, revealing only a minor decrease in balanced accuracy. Thus the embedding space learned on MAGs generalised well to the original tasks.

We then selected thirty-two phenotypes whose labels could be transferred with reasonable confidence via strain lineage, annotated the entire MAG corpus and evaluated performance where ground-truth labels were available. Balanced accuracy remained above 94% across diverse phenotypes, indicating robustness to genome type.

Phenotypes were grouped by held-out balanced accuracy (bAcc): five exceeded 93% (high confidence), thirteen had bAcc between 80% and 93% (medium) and fourteen between 75% and 80% (low). Predictions for each confidence tier were generated for the full MAG collection and released with accompanying confidence flags. The prediction files are available for download at https://drive.google.com/drive/folders/1PSR0mcE2fTexP9ADDVgYz3lkQ_qcyEsy?usp=sharing.

Identifying phenotype-specific genes. We identified genes associated with four phenotypic traits: sporulation, motility type, anaerobic growth, and environmental niche. The first three analyses used the complete-genome trait data described above, while environmental-niche prediction required a new data set: we selected families, genera, and species found in multiple habitats to prevent the model from relying on broad taxonomic signals. From this pool, we focused on *Lactobacillus johnsonii* and the genus *Prevotella* for detailed study, as they occupy distinct ecological niches

but share close taxonomic ranks, allowing us to disentangle lineage effects from habitat-specific adaptations.

A separate *Bacformer* model was fine-tuned for each phenotype. Each data set was randomly split into 60% training, 20% validation, and 20% test sets. For sporulation, motility, and anaerobic growth, we initialized from the *Bacformer* checkpoint pre-trained on complete genomes; for environmental-niche models, we used the MAG-pre-trained checkpoint. Input × Gradient attribution [46] was applied to produce an importance score for each gene in every test genome, by summing over the contextualized protein-embedding vectors. To assess biological plausibility, we averaged scores across all occurrences of each annotated protein, discarded genes lacking annotations, and compared mean scores of proteins matching relevant keywords (e.g., "flagellum," "spore formation," "anaerobe") to the remainder.

Sporulation served as a convenient qualitative benchmark due to its well-characterized genetic machinery. Ranking proteins by attribution successfully recovered known sporulation factors—especially the *spo* gene set—among the highest-scoring hits. After excluding proteins lacking locus tags in at least three strains, we averaged the top 100 ranked proteins across strains, clustered them using the *Leiden* algorithm (resolution 0.1, $k = 5$), and identified four functionally coherent modules corresponding to successive stages of spore formation.

We repeated this procedure to investigate environment-specific adaptation in *L. johnsonii*. For each host gut, the 100 most important proteins were averaged across strains and visualized with UMAP, with points colored by environment. Manual inspection highlighted several proteins previously implicated in host-specific colonization, confirming that the model identified meaningful determinants of niche preference.

Because many genomes lack reliable gene names or product descriptions due to fragmentation and annotation errors, we repeated the analysis using the protein-family labels assigned during pre-training. As all proteins carry such a label, this approach ensured no sequences were discarded. Within the *Prevotella* test set, we computed mean importance for each family, averaged the top 100 families per environment, and visualized the results with UMAP, again revealing clear environment-specific clusters.

B.4.6. *De novo* genome generation

Using a pre-trained *Bacformer* model with autoregressive objective on complete genomes, we have computed perplexity, which is a metric that indicates how well the model predicts the next element in a sequence. We then analysed the performance across test genomes, including species not seen during training.

Generating genomes with broad range of protein functions. We then assessed whether *Bacformer* can generate genomes that encompass a broad range of essential functions—an impor-

tant advance over existing generative models that are limited to designing individual molecules [47, 48, 49, 50, 51] or relatively simple multi-component systems [42].

In its original formulation *Bacformer* represents each protein with a continuous embedding produced by a protein language model from the raw amino-acid sequence. This fine-grained representation allows residue-level reasoning, yet it renders generation impractical: a decoder (the autoregressive component that generates sequences in transformers),, which takes a vector in $\mathbb{R}^D$ as input (where D is the dimension of the embedding), would also have to predict an exact vector output in $\mathbb{R}^D$, an unbounded task, rather than producing a symbol from a finite vocabulary.

To enable genome generation we discretise the input. A vocabulary $\mathcal{F} = \{1, \ldots, F\}$ of $F = 50,000$ discrete protein clusters is obtained by leveraging the results of unsupervised clustering used for pretraining. Each protein is mapped to its cluster index so that a genome f becomes a sequence

$$\mathbf{f} = (f_1, f_2, \ldots, f_N), \qquad f_i \in \mathcal{F}.$$

The model now looks up a learnable token vector $\mathbf{x}_i = E_{f_i}$ from an embedding matrix $E \in \mathbb{R}^{F \times D}$, exactly as in word-piece language models [16]. All other components of the architecture remain unchanged.

We initialise E with the rows of the protein-cluster *classification head* $W \in \mathbb{R}^{F \times D}$ learned during the autoregressive MAG pretraining. As W already captures the distinguishing features of each cluster in the same latent space, this transfer (i) aligns the new discrete-token model with the previously learned semantic manifold, (ii) speeds up convergence, and (iii) avoids catastrophic forgetting of cluster-specific information.

Starting from this initialisation we further pretrained the discrete-token model on the complete-genome corpus, following the identical optimisation schedule used for the continuous-embedding *Bacformer*. This procedure leverages uninterrupted chromosomal context while retaining compatibility with sequence generation tasks.

We prompted pre-trained autoregressive protein cluster *Bacformer* to generate a set of protein sequences ordered by their chromosomal positions from a prompt of 500 proteins starting from the origin of replication for *Staphylococcus aureus* (NCBI RefSeq ID `GCF_001019415.2`) and *Escherichia coli* (NCBI RefSeq ID `GCF_003627195.1`). The origin of replication represents the natural starting point for bacterial chromosome replication, providing a biologically meaningful context for genome generation

We assessed functional diversity by mapping protein families to *KEGG* BRITE ontology categories [43]. We extracted *KEGG* annotations for bacterial genes by downloading gene annotations and genome assembly links from NCBI RefSeq and GenBank via the *KEGG* website. We processed the downloaded genomes and embedded them using the *ESM-2* model (12 layers, 35M parameters) [9].

After discarding proteins without *KEGG* annotations, we obtained over 600,000 annotated proteins. We assigned *KEGG BRITE* functions to our protein clusters using k-nearest neighbor classification (k=7) based on centroid embeddings with Euclidean distance. We validated the annotation quality by examining a random sample of protein clusters and their k-nearest neighbor assignments, and verified that the overall distribution of *KEGG* protein functions across protein families resembled real-world distributions. For generated genomes (excluding the 500-protein prompt), we assigned each protein family a *KEGG BRITE* function and compared the distribution of true versus predicted *KEGG* labels. This analysis confirmed that *Bacformer* generates genomes with functional repertoires that closely match those of true genomes (Fig. 5B; Fig. S12).

To ensure that the model can generate diverse genomes and does not simply recapitulate genomes seen during training set, we generated multiple genomes using the same prompt while varying the temperature parameter to introduce controlled diversity, and we computed the distance between the generated genomes and those in the pretraining corpus. The results (Fig. S12) show that *Bacformer* generates diverse genomes that are not merely replicas of any single training genome.

Genome completion. To test whether *Bacformer* ability to recognise genomes with missing sections, we used the complete genome held-out set ($N = 9,688$) and systematically removed 5%, 10%, 25% or 50% of the proteins by trimming sequentially from the genome's end. Both truncated and intact genomes were embedded with *Bacformer* (trained on complete genomes that were not part of the held-out set, using the autoregressive objective) and placed in the same embedding space for comparison. For each truncated genome we measured how highly its intact counterpart ranked among all genomes in the embedding space, reporting the mean reciprocal rank (MRR), where 1.0 indicates perfect match.

To visualize the effect of missing data, we projected *Bacformer* embeddings of full genomes and their 50%-ablated versions onto two dimensions using UMAP (10 neighbors). As a non-contextual baseline, we performed the same procedure with averaged *ESM-2* protein embeddings, omitting the *Bacformer* encoder.).

Retrieving protein sequences from contextual embeddings. *Bacformer* generates contextual protein representations by associating each protein with one of 50,000 pre-defined protein clusters, rather than encoding each unique protein sequence individually. This clustering approach addresses the challenge posed by the immense diversity of protein sequences, enabling the model to generalize across related proteins while maintaining computational tractability.

By inverse mapping each embedding to its corresponding protein cluster, it becomes possible to retrieve the native protein sequence—or a close homolog—from a large candidate pool. This is particularly valuable for applications such as genome completion and synthetic biology, where identifying the most likely protein sequence from an embedding is essential.

To facilitate this mapping, *Bacformer*'s output embedding for each gene $\mathbf{h}_i \in \mathbb{R}^d$ is trained using a contrastive loss. This loss encourages the *Bacformer* embedding to closely match the *ESM-2* embedding of the same protein within a batch, while remaining distinct from embeddings of other proteins. As a result, the learned embedding space supports accurate retrieval of protein cluster identities from contextual representations.

To achieve this, both sets of embeddings are ℓ_2-normalised and similarity is measured across protein embeddings using the cosine similarity, where $\mathrm{sim}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u}^\top \mathbf{v}}{\|\mathbf{u}\| \, \|\mathbf{v}\|}$ and temperature $\tau = 1$, the noise-contrastive loss for a positive pair $(\mathbf{h}_i, \mathbf{e}_i)$ is

$$\ell_i^{\mathrm{ctr}} = -\log \frac{\exp\big(\mathrm{sim}(\mathbf{h}_i, \mathbf{e}_i)/\tau\big)}{\sum_{j=1}^{N} \exp\big(\mathrm{sim}(\mathbf{h}_i, \mathbf{e}_j)/\tau\big)}, \qquad \mathcal{L}_{\mathrm{ctr}} = \frac{1}{N} \sum_{i=1}^{N} \ell_i^{\mathrm{ctr}}.$$

Masked-family prediction uses the usual cross-entropy with the true family labels at masked positions and the logits from *Bacformer*'s classification head. The combined training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{mask}} + \alpha \mathcal{L}_{\mathrm{ctr}}, \qquad \alpha = 0.5.$$

During fine-tuning we initialised from the MAG-pretrained *Bacformer* with masked objective, computed $\mathcal{L}_{\mathrm{mask}}$ on masked tokens, computed $\mathcal{L}_{\mathrm{ctr}}$ on all non-special tokens, and back-propagated the combined loss. At inference time we measured cosine similarity and recall@K between contextual embeddings and their ground-truth sequence embeddings, observing strong retrieval for unmasked proteins and a marked dependence on the transformer's ability to refine input vectors for masked positions.

We trained the model on complete genomes, initializing from a *Bacformer* pretrained on MAGs with the masked objective and following the same method for complete genome training. We computed $\mathcal{L}_{\mathrm{mask}}$ on masked tokens, $\mathcal{L}_{\mathrm{ctr}}$ on all proteins, and back-propagated the combined loss.

Using a trained model we ran inference on test genomes and calculated the cosine similarity between the real and predicted embeddings. For both masked and unmasked proteins, we measured the cosine similarity between the positive (true) and negative (false) protein embeddings. Furthermore, we assessed the model performance using the recall@K metric, which measures how often the correct embedding is found among the top K most similar embeddings. While the model performed strongly on unmasked proteins, it achieved significantly worse results in the task of retrieving the true embedding at low K values for masked proteins. This highlights the importance of the *Bacformer* transformer backbone's capacity to update protein embeddings, as the contextualized representations differ significantly from the initial protein embeddings used as input.

Generating genomes with desired properties with conditional finetuning. To explore how *Bacformer* could be applied to generate entire bacterial genomes endowed with specific properties, we selected two categorical traits from, the phenotypic dataset described above: oxygen requirements (specifically *aerobic*, *anaerobic* and *facultative*); and optimal growth temperature (*thermophilic* (>80°C), *psychrophilic* (<20°C) and *mesophilic* (20-40°C)).

We randomly split each dataset into train, validation and test splits with 60/20/20% ratio split. For training we initialized the model weights with the protein cluster *Bacformer* model pre-trained on complete genomes described in the *Generating genomes with broad range of protein functions* section above. For each phenotype, we introduced a property token at the start of each protein-family sequence (Fig. 5F). The property token was intitalized randomly and trained from scratch during training. We then finetuned *Bacformer* to generate protein-family sequences conditional on a phenotype token monitoring the validation loss separately for each trait.

Following finetuning, we used each trained model to generate a representative subset of genomes ($N = 20$) from the test set given a prompt of a property token and a set of 500 protein families. For comparison, we did the same, however, without the property token. We then compared the generated protein families with and without the property token, excluding the 500 prompt tokens. To evaluate the difference between the protein family sequences generated with and without the property token, we associated the protein family tokens with *KEGG Orthology* (KO) groups [43]. Thus, allowing us to identify specific orthologous protein groups which differ between the generated sequences. To do this, we used the annotated proteins from *KEGG* for bacterial genomes (see *Generating genomes with broad range of protein functions*). As a protein may belong to multiple orthologous groups, we assigned protein cluster tokens, using its centroid, to an orthologous group if 2 out of 10 nearest neighbors from the space of annotated proteins had the group label. We used euclidean distance to compute the kNN. The threshold was selected after experimenting with different values and evaluating the distribution of the number of KO groups assigned to protein family tokens. After assigning each protein family token its KO groups, we aggregated the KO groups based on the generated protein families for the genomes generated with and without the property token and computed the $log2$ fold-change to evaluate the difference between the two. Finally, to evaluate the statistical significance, we performed Fisher's exact test [27] using the aggregated KO counts from the genomes with and without the property token. We then qualitatively analysed the positively and negatively enriched statistically significant KO groups ($p < 0.075$).

We would like to highlight that full validation of synthetic genome viability remains outside the scope of this study, requiring complex genome engineering [52], these findings underscore the promise of contextual, evolution-informed models for rational genome design, laying the groundwork for more sophisticated generative designs in microbial genomics.

C. Supplementary figures

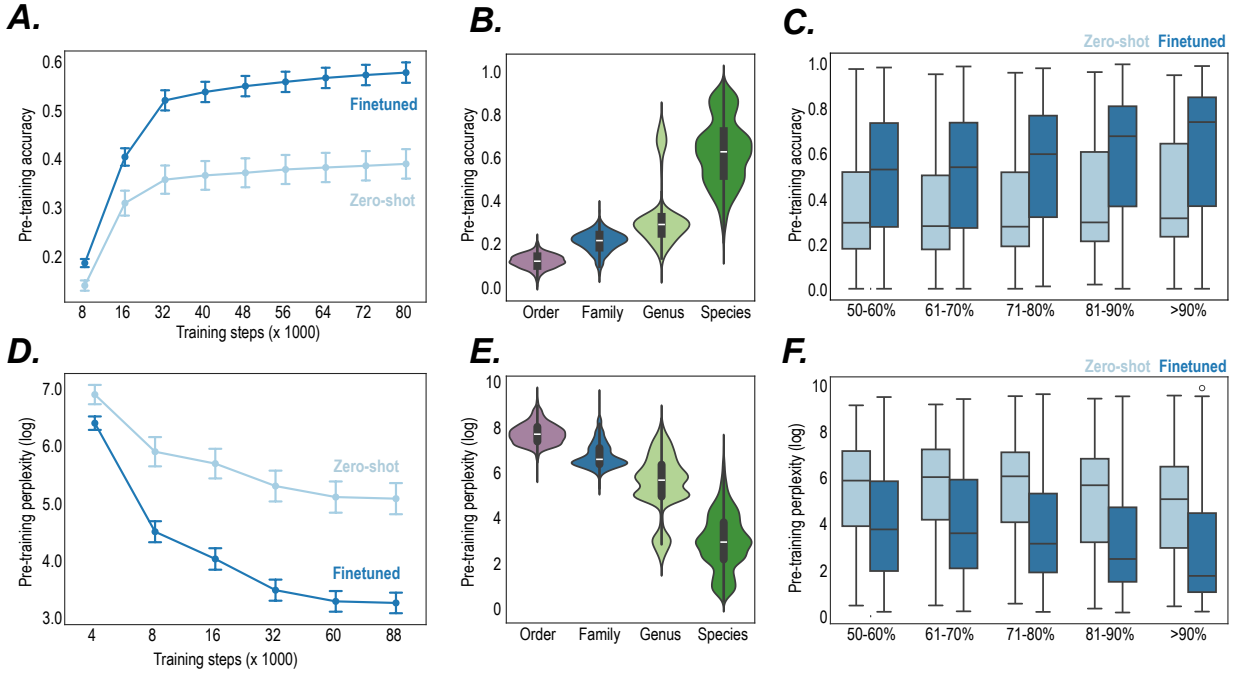

FIGURE S1. **Pretraining performance of the Bacformer. (A-C)** Pretraining performance of *Bacformer* on the held-out evaluation set ($N = 49,213$) on the masked protein family prediction objective. Error bars represent standard error computed from 15 bootstrap resamples. **(A)** Accuracy across training steps on zero-shot and non-zero-shot genomes. **(B)** Accuracy across taxonomic ranks ablated from training (median and interquartile range (IQR) shown). **(C)** Accuracy across genome completeness bins, stratified by zero-shot label (median and IQR shown; whiskers extend 1.5 × IQR). **(D-F)** Pretraining performance of *Bacformer* on the held-out evaluation set ($N = 49,213$) with autoregressive protein family prediction objective. Error bars represent standard error computed from 15 bootstrap resamples. **(D)** Perplexity across training steps on zero-shot and non-zero-shot genomes. **(E)** Perplexity across taxonomic ranks ablated from training (median and IQR shown). **(F)** Perplexity across genome completeness bins, stratified by zero-shot label (median and IQR shown; whiskers extend 1.5 × IQR).

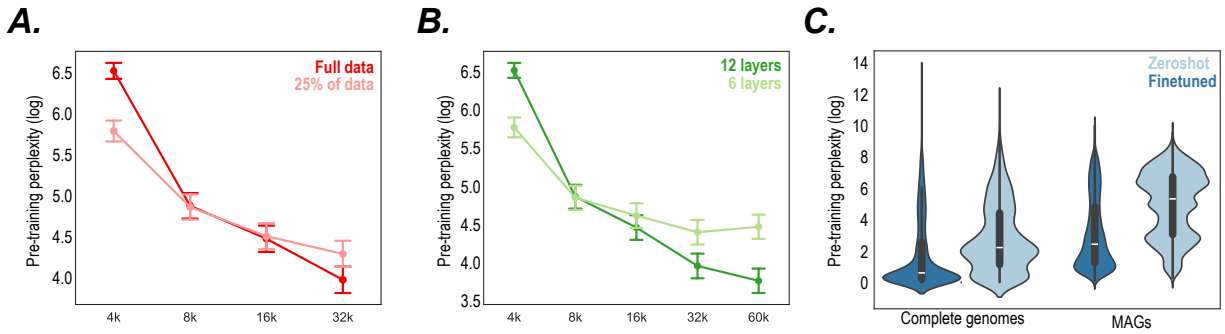

FIGURE S2. **Bacformer scales with model & dataset size. (A-C)** Pretraining performance of *Bacformer* on the held-out evaluation set (N = 49,213) with next protein family prediction objective. **(A)** Perplexity across training steps for model with 25% and 100% of data. Error bars represent standard error computed from 15 bootstrap resamples. **(B)** Perplexity on the held-out evaluation set across training steps for model with 6 layers and 12 layers. Error bars represent standard error computed from 15 bootstrap resamples. **(C)** Perplexity on complete vs metagenome assembled genomes (MAGs), including zero-shot and non-zero shot genomes (median and interquartile range shown) using *Bacformer* model pretrained on MAGs and finetuned on complete genomes.

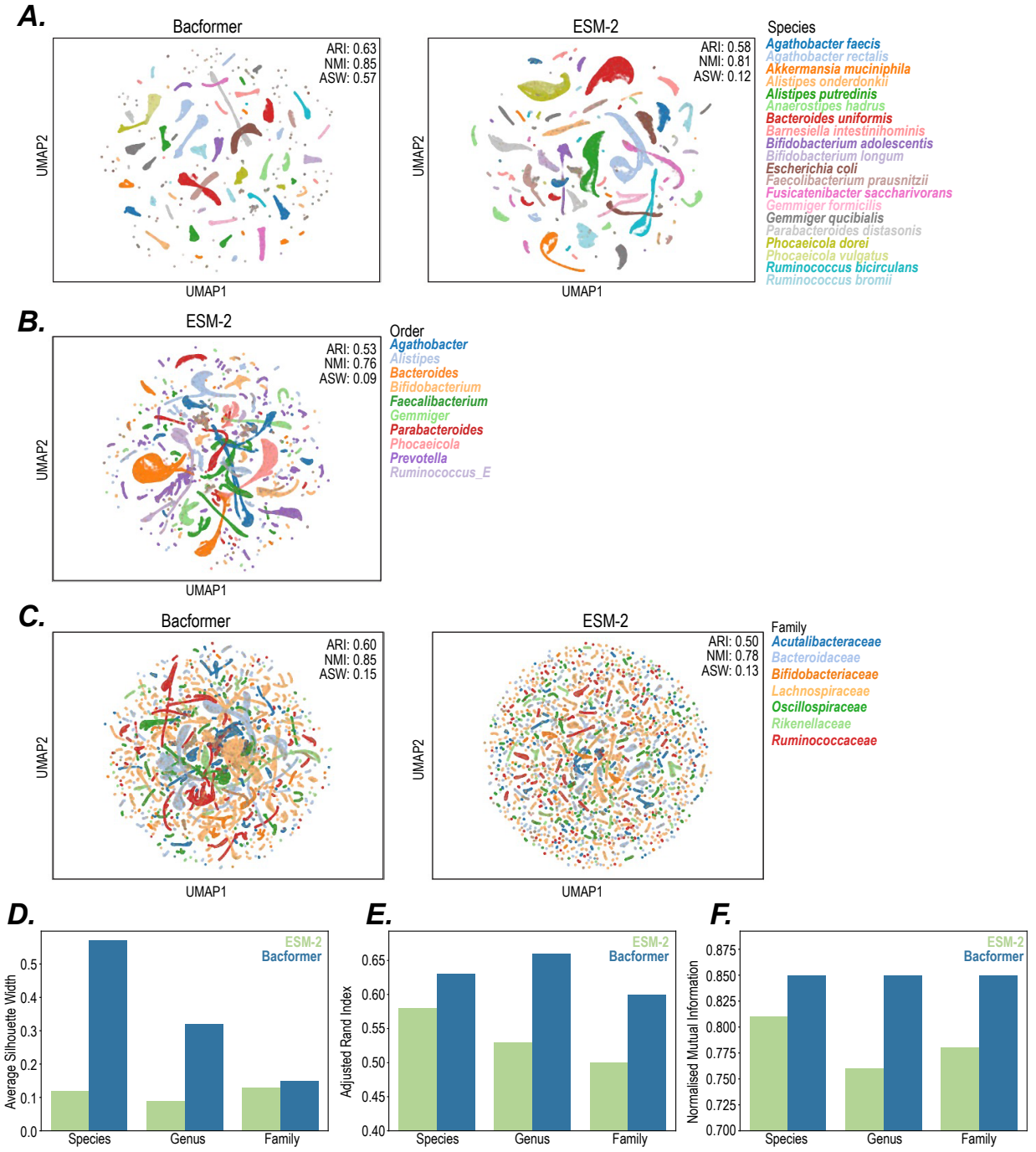

FIGURE S3. **Analysis of strain clustering using Bacformer. (A-C)** UMAP plots showing bacterial strain clustering of **(A)** The 20 most frequent species in our corpus, *Bacformer* (left) and *ESM-2* (right) **(B)** 10 most frequent genera by *ESM-2* (*Bacformer* in Fig. 1C), and **(C)** 7 most frequent family in our corpus, *Bacformer* (left) and *ESM-2* (right). Each is evaluated using three clustering metrics: Adjusted Rand Index (ARI), Normalised Mutual Information (NMI) and Average Silhouette Width (ASW). All metrics vary from 0 to 1; higher is better. **(D-F)** Clustering metrics for *Bacformer* and *ESM-2* across the top 20 species, 10 genera, and 7 families from our corpus, setting *Leiden* resolution of 1.0 for evaluation.

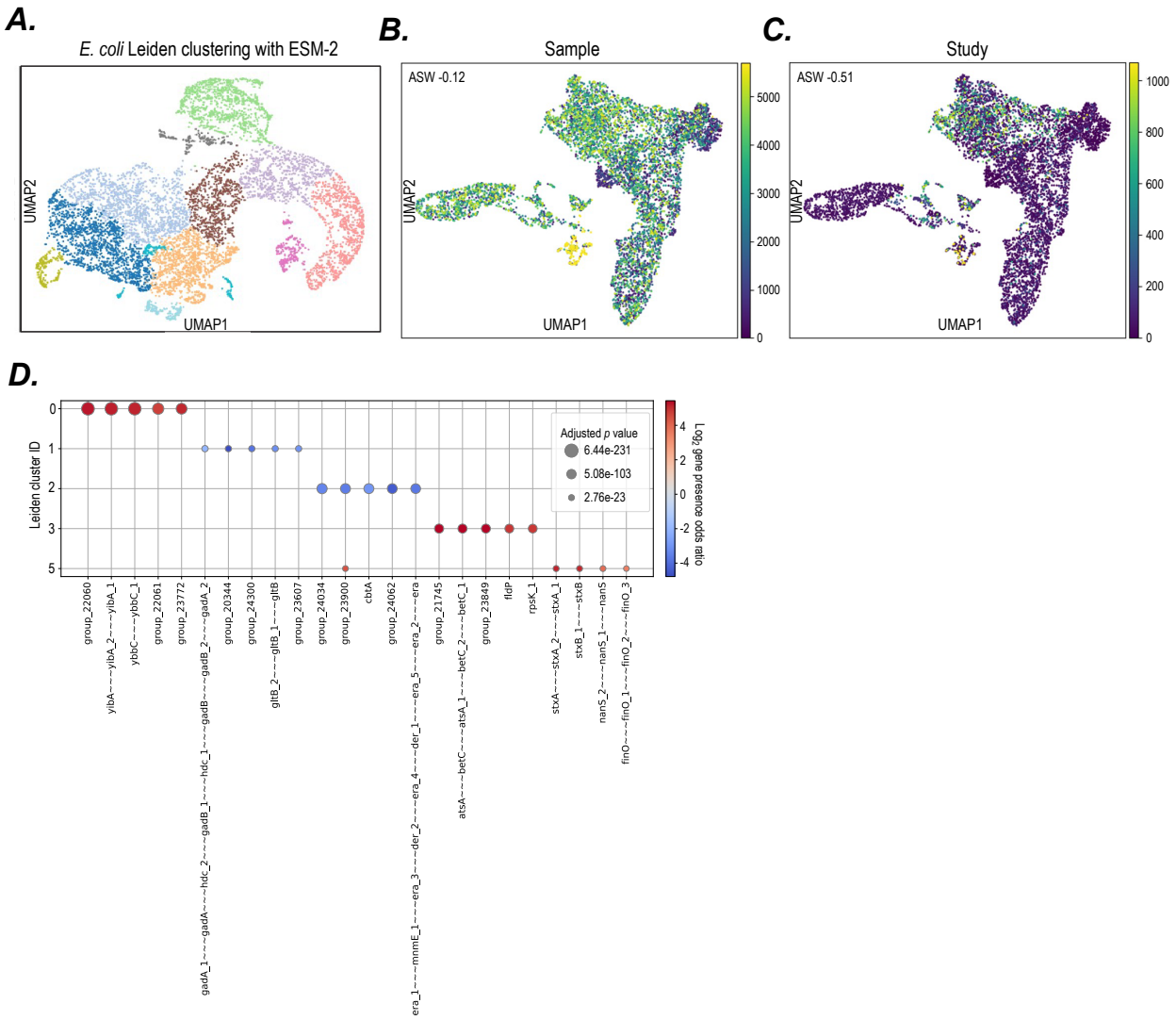

FIGURE S4. **Bacformer clustering results for *Escherichia coli*. (A)** UMAP plot of *E. coli* strains from the *MGnify* dataset [15] using genome embeddings from *ESM-2* annotated by their *Leiden* cluster labels. The comparison cluster map for *Bacformer* in Fig. 1E shows six distinct groups and has been re-annotated by the features **(B)** Sample, and **(C)** Study, from which the genome derived. The low results ASW (Average Silhouette Width (0-1, higher is better)) for B) and C) demonstrate that the identified clusters are unlikely an artifact of these technical batch effects. **(D)** Top 5 cluster-specific genes based on gene presence/absence according to the *p*-value for 5 *Leiden* clusters from the *MGnify E. coli* population. The color represents the log_2 gene presence odds ratio, showing whether the gene presence or absence is enriched in the subpopulation.

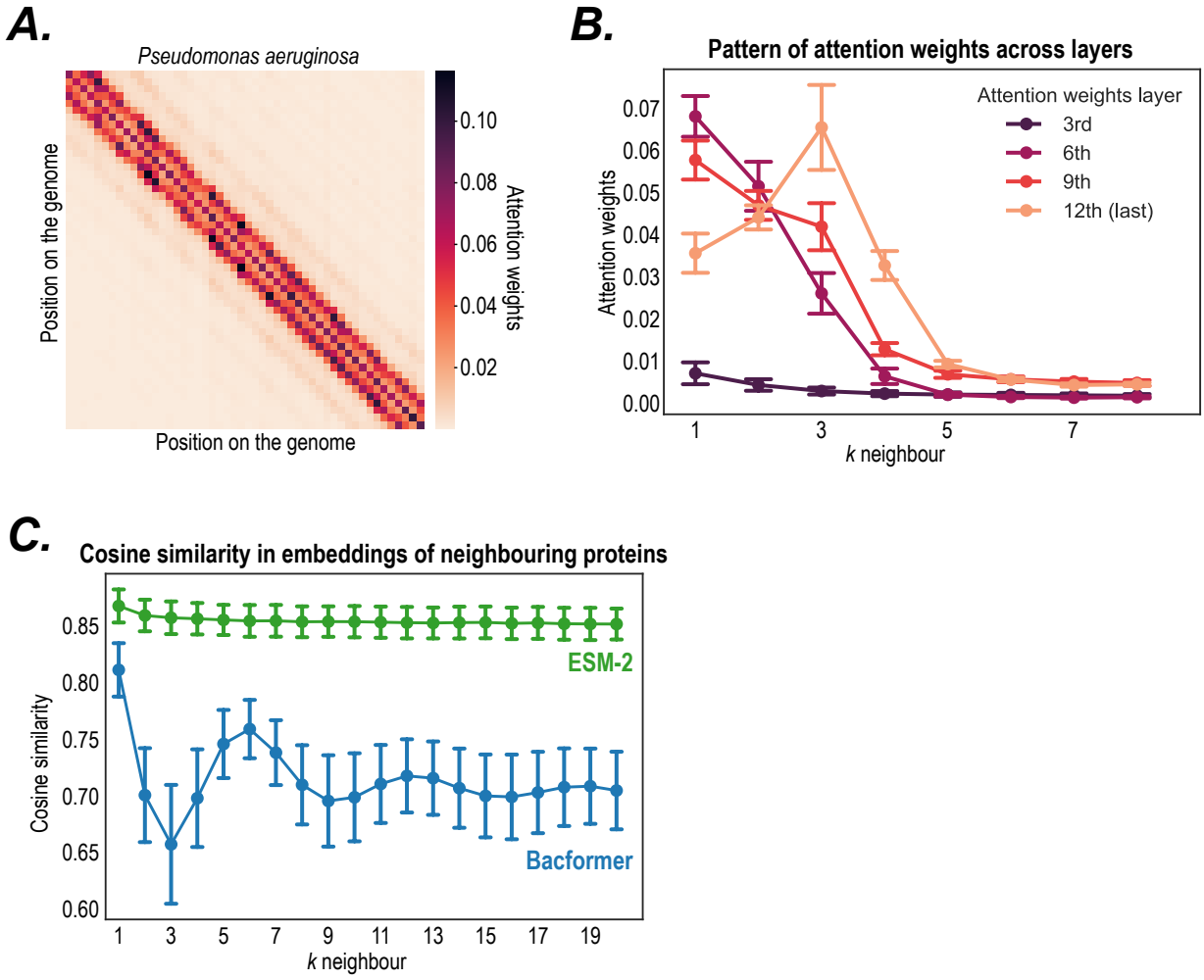

FIGURE S5. **Bacformer leverages genomic neighborhood to compute contextual protein representations. (A)** Heatmap of attention weights from the last *Bacformer* layer showing that the model pays considerable attention to neighbouring proteins (heatmap generated from data for the first 100 proteins on the *Pseudomonas aeruginosa* PAO1 genome). **(B)** Attention weights across layers and neighbours. The attention weights were computed by averaging attention heads across proteins from $N = 45$ genomes. Error bars represent standard error. **(C)** Cosine similarity between embeddings of proteins and their neighbours across the genome (evaluated in $N = 45$ genomes) for *ESM-2* (green) and *Bacformer* (blue), the latter showing cosine patterns (reflecting the use of rotary positional embeddings [8] in the model. Error bars represent standard deviation.

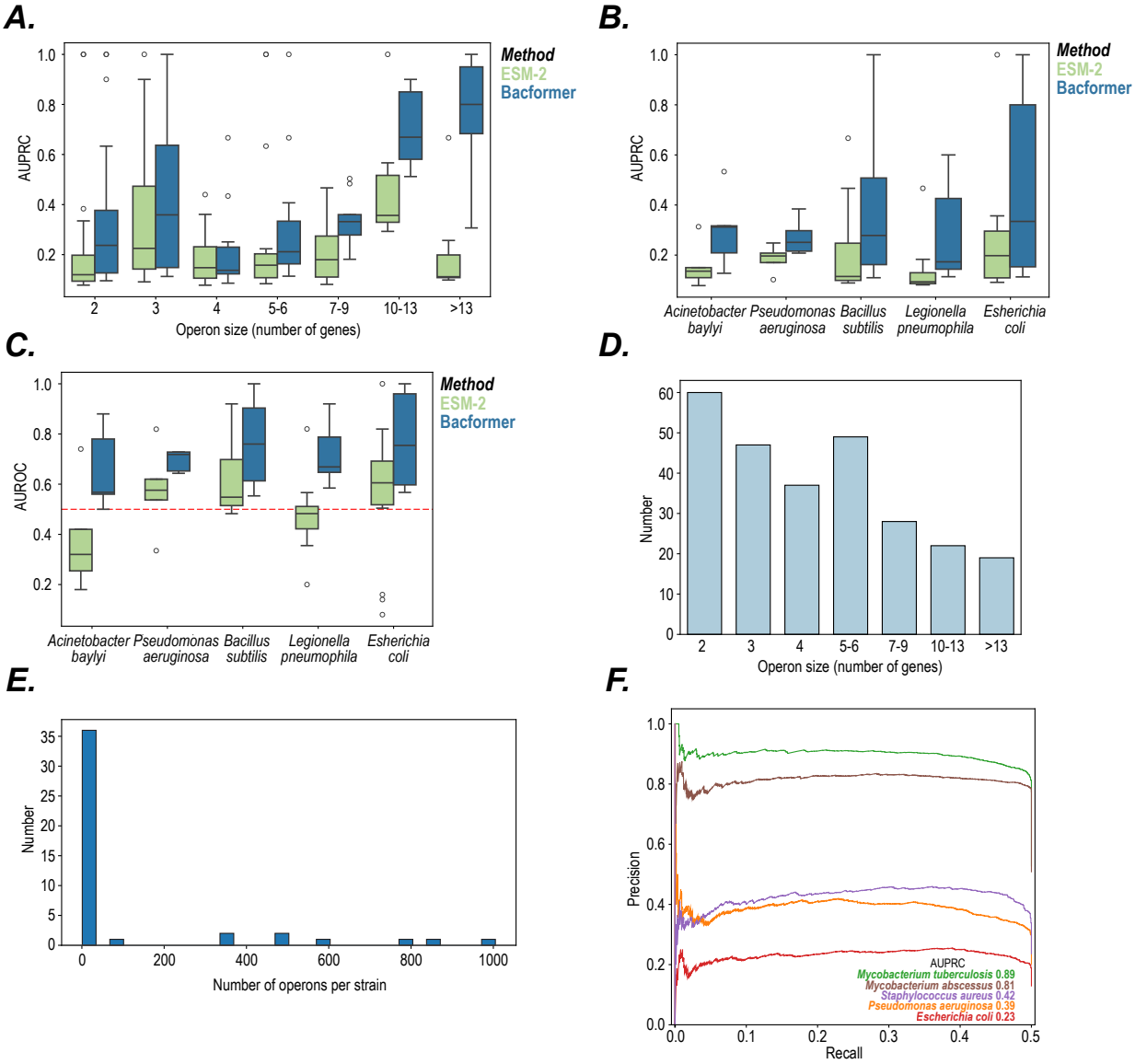

FIGURE S6. **Zero-shot operon predictions using Bacformer. (A-C)** Operon prediction performance of *Bacformer* (blue) and *ESM-2* (green) for operon annotations from Operon DB [33] across different **(A)** operon sizes and **(B & C)** different bacterial species. Performance is measured in **(A & B)** by area under precision recall curve (AUPRC) and in **(C)** area under receiver operator characteristic curve (AUROC) (red dashed line indicates random AUROC (0.5)). In each plot, lines and boxes represent median and interquartile range (IQR) respectively, with whiskers extending 1.5 × IQR beyond Q1 and Q3. **(D & E)** Count of operons available from Operon DB and used for predictions by **(D)** operon size and **(E)** strain in the preprocessed Operon DB dataset. **(F)** Performance for *Bacformer* zero-shot operon predictions of operons for five bacterial species in which we generated experimental ground truth data by long read RNA sequencing, measured by Precision recall curves (PRC) and AUPRC.

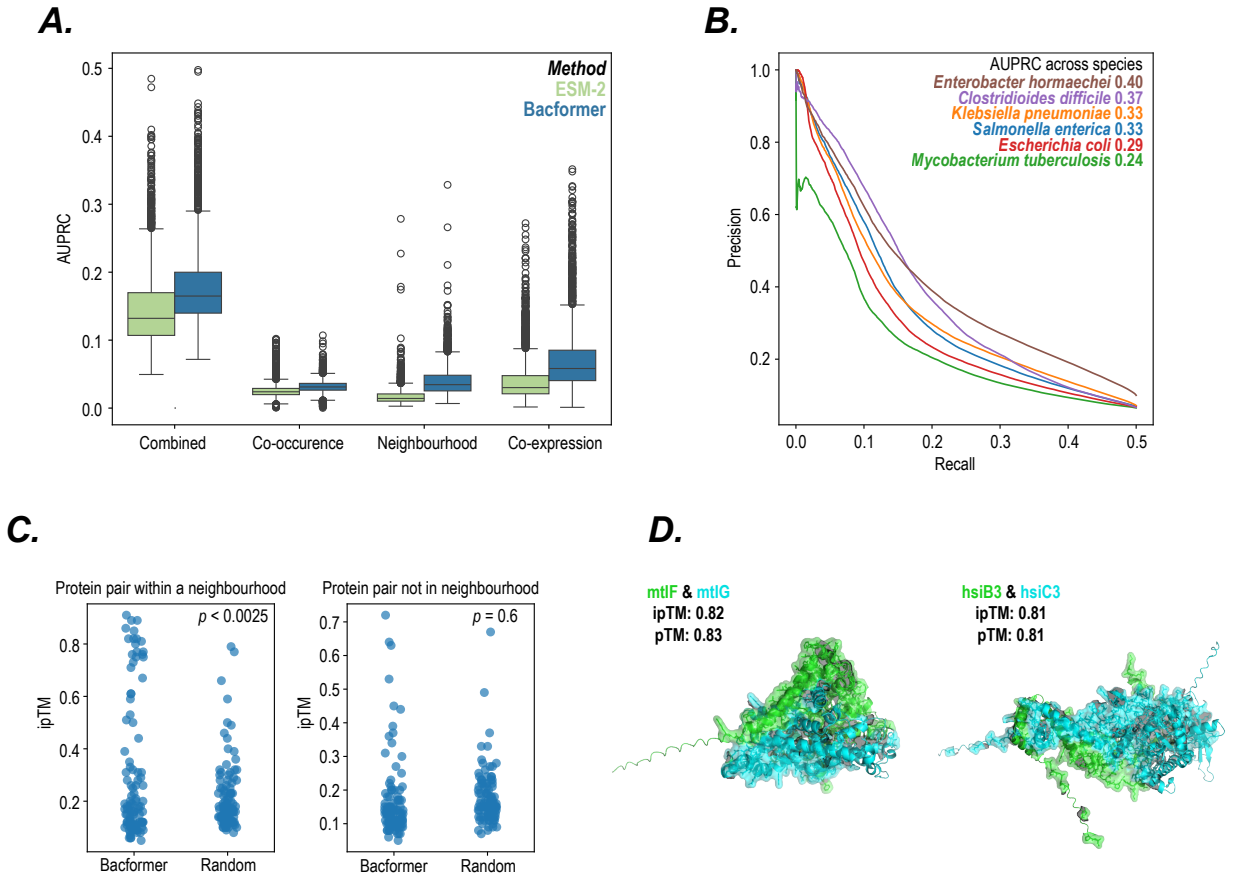

FIGURE S7. **Bacformer predictions of protein-protein interaction across species. (A)** Zero-shot predictions (evaluated using the area under the precision-recall curve; AUPRC) for *Bacformer* (blue) and *ESM-2* (green) across different *STRING DB* interaction types [36] on held-out genomes ($N = 2,088$). The interaction scores were calculated using cosine similarity between protein representations. Lines and boxes represent the median and interquartile rang (IQR) respectively, with whiskers extending 1.5 × IQR beyond Q1 and Q3. **(B)** Precision-recall curves for *Bacformer* finetuned on protein-protein interaction task with data from *STRING DB* across diverse species. **(C & D)** Using *AlphaFold3* to validate protein-protein interaction predictions from *Bacformer*. **(C)** ipTM scores of 100 top scoring protein pairs in *Pseudomonas aeruginosa* PAO1 genome using finetuned *Bacformer* vs ipTM scores of 100 randomly selected protein pairs for (left) protein pairs located within the same neighborhood and (right) protein pairs located not in the same neighborhood. The protein neighborhood was defined as $k = 5$ nearest gene from the gene of interest. The p-values were calculated using Welch's t-test. **(D)** Predicted protein complex structures with *AlphaFold3* [38] between two protein pairs with a high score from *Bacformer* model finetuned on the protein-protein interaction task. The ipTM score indicates the probability of the two proteins binding together and pTM score is a measure of overall predicted structure confidence. Both metrics are bound between 0-1, higher is better.

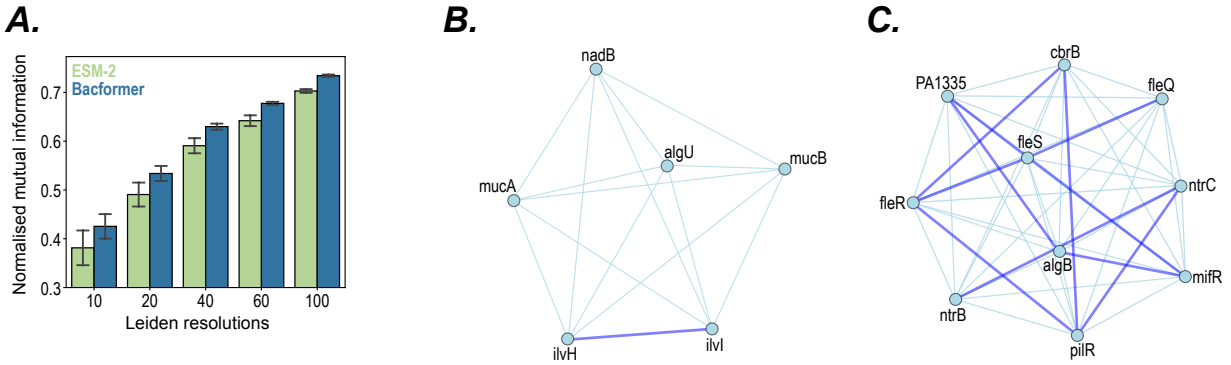

FIGURE S8. **Bacformer reveals gene networks in *Pseudomonas aeruginosa*. (A)** Evaluation of zero-shot gene clustering on *P. aeruginosa* PAO1 using co-expression and protein representations from *Bacformer* and *ESM-2* across *Leiden* resolutions. Higher normalised mutual information indicates higher overlap between clusterings. The error bars represent standard error across different number of neighbours used for *Leiden* clustering. **(B & C)** Gene networks from *P. aeruginosa* PAO1 genome extracted using zero-shot *Bacformer* contextual protein embeddings at *Leiden* resolution of 40 revealing **(B)** the *algU* gene network consisting interacting genes such as *mucA* and *mucB* and **(C)** the *fleQ* gene network consisting interacting genes such as *fleS*, *fleR* and *algB*.

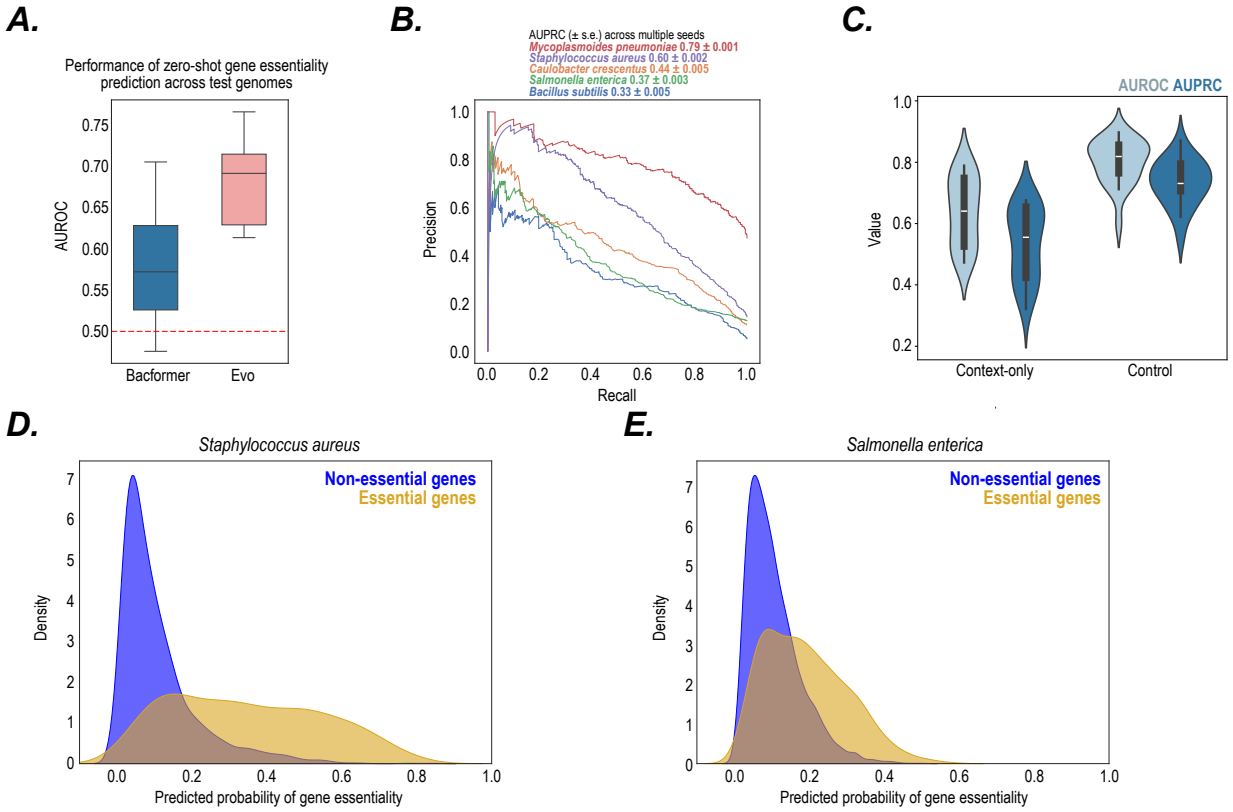

FIGURE S9. **Bacformer gene essentiality predictions across species. (A)** Zero-shot gene essentiality prediction performance by *Bacformer* and *Evo* on held-out genomes (N = 10) evaluated by area under ROC curve (AUROC) metric. **(B)** Precision-recall curves of finetuned *Bacformer* gene essentiality prediction for diverse species (reporting AUPRC and standard deviation computed by using different random seeds for finetuning *Bacformer*). **(C)** Finetuned *Bacformer* predicts gene essentiality based on context-only across diverse genomes. We masked the essential protein and asked the model to predict essentiality based only on the other proteins present in the genome (median and interquartile range shown). **(D & E)** Distribution of predicted probabilities for essential and non-essential genes from finetuned *Bacformer* in **(D)** *Staphylococcus aureus* and **(E)** *Salmonella enterica*.

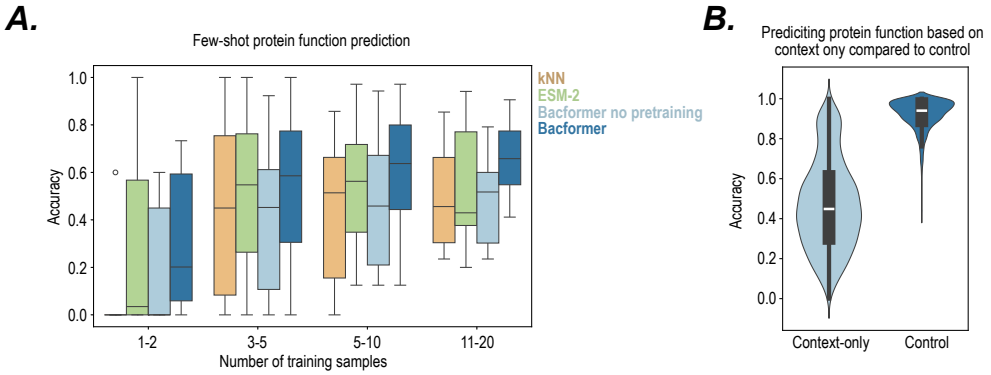

FIGURE S10. **Bacformer protein function prediction. (A)** Performance on the few-shot protein function prediction task across classes with different number of training samples (lines and boxes represent the median and interquartile range (IQR), with whiskers extending 1.5 × IQR beyond Q1 and Q3). **(B)** *Bacformer* can predict protein function based on genomic context only. We masked the protein embedding at the input level and asked the model to predict protein function using the contextual protein embedding across genomes ($N = 1,931$), median and IQR shown).

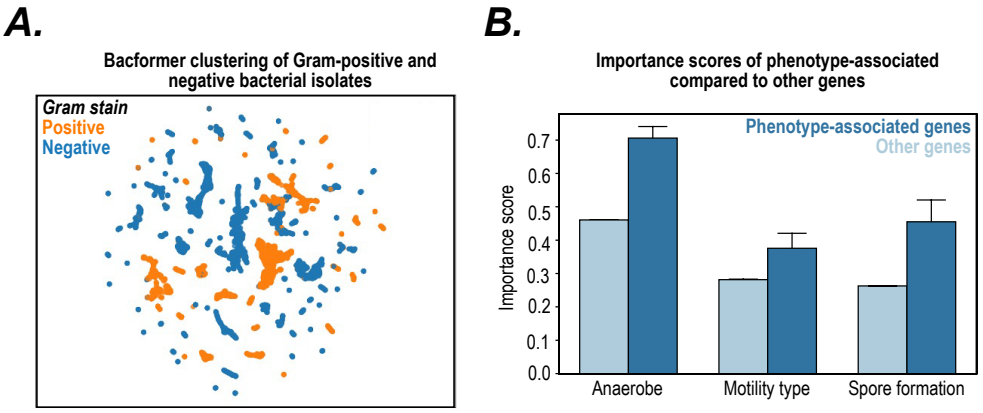

FIGURE S11. **Bacformer phenotypic traits prediction. (A)** UMAP plot of gram-positive (orange) and negative (blue) strains. The genome embeddings were computed in a zero-shot manner using aggregated contextualised protein embeddings from the last layer of the pretrained *Bacformer* model. Annotations were extracted from published datasets [44, 45]. **(B)** *Bacformer* identifies phenotype-specific genes by computing importance scores per gene using phenotype-specific models. We used a trained phenotype-specific *Bacformer* model to extract gradient-based attribution scores [46] per gene across samples. The genes with highest importance scores across samples are labelled *phenotype-specific genes*.

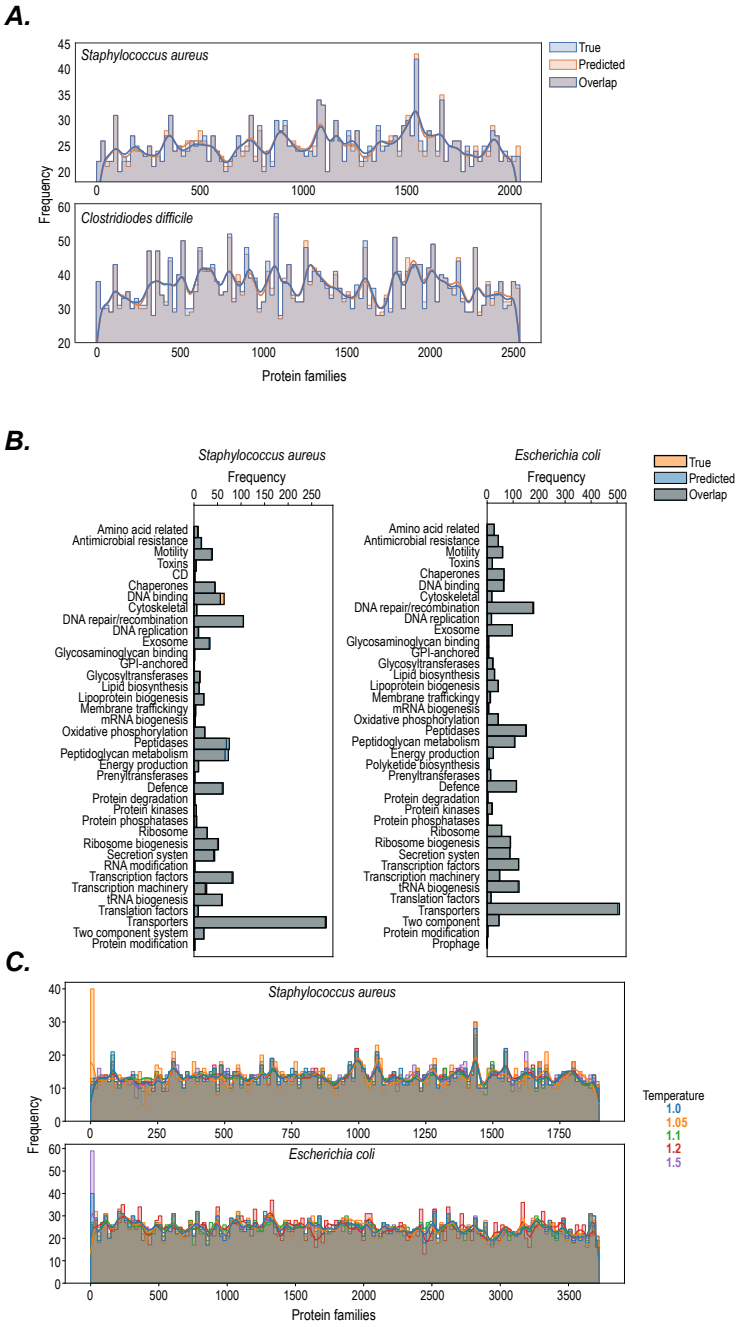

FIGURE S12. **Generative modelling capacity of Bacformer. (A)** *Bacformer* successfully generates genomes unseen during training. True (blue) and predicted (orange) protein family distributions shown for *Staphylococcus aureus* and *Clostridioides difficile*. **(B)** *Bacformer* generates genomes encompassing a broad range of functions. True (blue) and predicted (orange) protein function distribution on *S. aureus* (left) and *Escherichia coli* (right). **(C)** *Bacformer* generates diverse protein family sequences. We prompted *Bacformer* with a set of 500 initial proteins on two different bacteria. The generated genomes varies across different temperature levels, which control the diversity of the generation process.

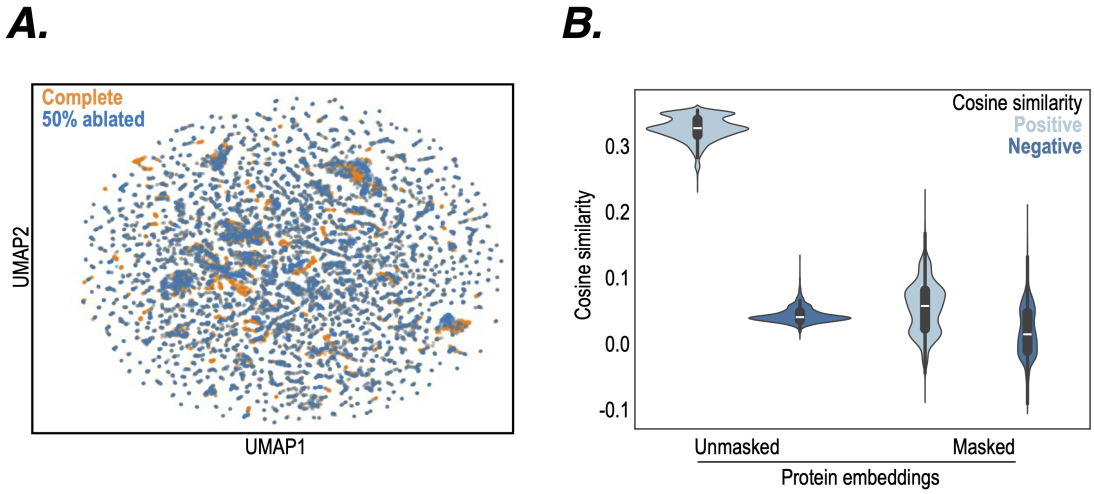

FIGURE S13. **Retrieving complete genomes & real proteins from contextual Bacformer embeddings. (A)** UMAP plot of *Bacformer* embeddings of complete genomes (orange) vs the same genomes with 50% of the proteins artificially ablated (blue). **(B)** Cosine similarity between *Bacformer* contextual protein embedddings and *ESM-2* protein embeddings for masked and unmasked protein embeddings. The *ESM-2* embeddings are used as input to the *Bacformer* transformer block which computes contextual embeddings. The positive (light blue) represents the true protein used at input-level, while negatives (dark blue) represent other proteins in the genome (median and interquartile range shown).

References

[1] National Institutes of Health. NIH Guidelines for Research Involving Recombinant or Synthetic Nucleic Acid Molecules. Bethesda, MD; 2024. NIH Office of Science Policy. Available from: https://osp.od.nih.gov/wp-content/uploads/NIH_Guidelines.pdf.

[2] U S Department of Health and Human Services, Centers for Disease Control and Prevention, and National Institutes of Health. Biosafety in Microbiological and Biomedical Laboratories. Atlanta; 2020. CDC. Available from: https://www.cdc.gov/labs/pdf/CDC-BiosafetyMicrobiologicalBiomedicalLaboratories-2020-P.pdf.

[3] Centers for Disease Control and Prevention and U S Department of Agriculture. Federal Select Agent Program; 2024. CDC/USDA. Available from: https://www.selectagents.gov.

[4] Executive Office of the President. Executive Order 14292 of May 5, 2025. Improving the Safety and Security of Biological Research; 2025. Federal Register Vol. 90, No. 88. Available from: https://www.presidency.ucsb.edu/documents/executive-order-14292-improving-the-safety-and-security-biological-research.

[5] European Food Safety Authority. Genetically Modified Organisms; 2025. EFSA. Available from: https://www.efsa.europa.eu/en/topics/genetically-modified-organisms.

[6] European Parliament and Council. Regulation (EC) No 1829/2003 of 22 September 2003 on genetically modified food and feed; 2003. Official Journal of the European Union, L 268, 1–23. Available from: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32003R1829.

[7] Cartagena Protocol on Biosafety to the Convention on Biological Diversity. Montreal, 29 January 2000; 2000. United Nations Treaty Series, Registration No. 30619. Available from: https://www.cbd.int/doc/legal/cartagena-protocol-en.pdf.

[8] Su J, Ahmed M, Lu Y, Pan S, Bo W, Liu Y. Roformer: Enhanced transformer with rotary position embedding. Neurocomputing. 2024;568:127063.

[9] Lin Z, Akin H, Rao R, Hie B, Zhu Z, Lu W, et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. bioRxiv. 2022.

[10] Osbourn AE, Field B. Operons. Cellular and Molecular Life Sciences. 2009;66:3755-75.

[11] Lloyd S. Least squares quantization in PCM. IEEE transactions on information theory. 1982;28(2):129-37.

[12] Okuta R, Unno Y, Nishino D, Hido S, Loomis C. CuPy: A NumPy-Compatible Library for NVIDIA GPU Calculations. In: Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Thirty-first Annual Conference on Neural Information Processing Systems (NIPS); 2017. Available from: http://learningsys.org/nips17/assets/papers/paper_16.pdf.

[13] Steinegger M, Söding J. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. Nature biotechnology. 2017;35(11):1026-8.

[14] Huerta-Cepas J, Szklarczyk D, Heller D, Hernández-Plaza A, Forslund SK, Cook H, et al. eggNOG 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. Nucleic acids research. 2019;47(D1):D309-14.

[15] Richardson L, Allen B, Baldi G, Beracochea M, Bileschi ML, Burdett T, et al. MGnify: the microbiome sequence data analysis resource in 2023. Nucleic Acids Research. 2023;51(D1):D753-9.

[16] Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers); 2019. p. 4171-86.

[17] Kingma DP. Adam: A method for stochastic optimization. arXiv preprint arXiv:14126980. 2014.

[18] Paszke A. Pytorch: An imperative style, high-performance deep learning library. arXiv preprint arXiv:191201703. 2019.

[19] Wolf T, Debut L, Sanh V, Chaumond J, Delangue C, Moi A, et al. Huggingface's transformers: State-of-the-art natural language processing. arXiv preprint arXiv:191003771. 2019.

[20] Schmidt TSB, Fullam A, Ferretti P, Orakov A, Maistrenko OM, Ruscheweyh HJ, et al. SPIRE: a Searchable, Planetary-scale mIcrobiome REsource. Nucleic Acids Research. 2023 10;52(D1):D777-83. Available from: https://doi.org/10.1093/nar/gkad943.

[21] Bowers RM, Kyrpides NC, Stepanauskas R, Harmon-Smith M, Doud D, Reddy T, et al. Minimum information about a single amplified genome (MISAG) and a metagenome-assembled genome (MIMAG) of bacteria and archaea. Nature biotechnology. 2017;35(8):725-31.

[22] O'Leary NA, Wright MW, Brister JR, Ciufo S, Haddad D, McVeigh R, et al. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. Nucleic Acids Research. 2016;44(D1):D733-45.

[23] Sayers EW, Cavanaugh M, Clark K, Pruitt KD, Schoch CL, Sherry ST, et al. GenBank. Nucleic Acids Research. 2022;50(D1):D161-4.

[24] Elnaggar A, Heinzinger M, Dallago C, Rehawi G, Wang Y, Jones L, et al. ProtTrans: Towards Cracking the Language of Life's Code Through Self-Supervised Deep Learning and High Performance Computing. bioRxiv. 2020. Available from: https://www.biorxiv.org/content/early/2020/07/21/2020.07.12.199554.

[25] Traag VA, Waltman L, Van Eck NJ. From Louvain to Leiden: guaranteeing well-connected communities. Scientific reports. 2019;9(1):1-12.

[26] Tonkin-Hill G, MacAlasdair N, Ruis C, Weimann A, Horesh G, Lees JA, et al. Producing polished prokaryotic pangenomes with the Panaroo pipeline. Genome biology. 2020;21:1-21.

[27] Fisher RA. On the interpretation of χ 2 from contingency tables, and the calculation of P. Journal of the royal statistical society. 1922;85(1):87-94.

[28] Heller DM, Tavag M, Hochschild A. CbtA toxin of Escherichia coli inhibits cell division and cell elongation via direct and independent interactions with FtsZ and MreB. PLoS genetics. 2017;13(9):e1007007.

[29] O'Brien AD, Newland JW, Miller SF, Holmes RK, Smith HW, Formal SB. Shiga-like toxin-converting phages from Escherichia coli strains that cause hemorrhagic colitis or infantile diarrhea. Science. 1984;226(4675):694-6.

[30] Sancho J. Flavodoxins: sequence, folding, binding, function and beyond. Cellular and Molecular Life Sciences CMLS. 2006;63:855-64.

[31] Unden G, Bongaerts J. Alternative respiratory pathways of Escherichia coli: energetics and transcriptional regulation in response to electron acceptors. Biochimica et Biophysica Acta (BBA)-Bioenergetics. 1997;1320(3):217-34.

[32] Touchon M, Perrin A, De Sousa JAM, Vangchhia B, Burn S, O'Brien CL, et al. Phylogenetic background and habitat drive the genetic diversification of Escherichia coli. PLoS genetics. 2020;16(6):e1008866.

[33] Okuda S, Katayama T, Kawashima S, Goto S, Kanehisa M. ODB: a database of operons accumulating known operons across multiple genomes. Nucleic acids research. 2006;34(suppl_1):D358-62.

[34] Li H. Minimap2: pairwise alignment for nucleotide sequences. Bioinformatics. 2018;34(18):3094-100.

[35] Danecek P, Bonfield JK, Liddle J, Marshall J, Ohan V, Pollard MO, et al. Twelve years of SAMtools and BCFtools. GigaScience. 2021 02;10(2). Giab008. Available from: https://doi.org/10.1093/gigascience/giab008.

[36] Szklarczyk D, Kirsch R, Koutrouli M, Nastou K, Mehryary F, Hachilif R, et al. The STRING database

in 2023: protein–protein association networks and functional enrichment analyses for any sequenced genome of interest. Nucleic acids research. 2023;51(D1):D638-46.

[37] Hendrycks D, Gimpel K. Gaussian error linear units (gelus). arXiv preprint arXiv:160608415. 2016.

[38] Abramson J, Adler J, Dunger J, Evans R, Green T, Pritzel A, et al. Accurate structure prediction of biomolecular interactions with AlphaFold 3. Nature. 2024:1-3.

[39] Rychel K, Decker K, Sastry AV, Phaneuf PV, Poudel S, Palsson BO. iModulonDB: a knowledgebase of microbial transcriptional regulation derived from machine learning. Nucleic acids research. 2021;49(D1):D112-20.

[40] Rajput A, Tsunemoto H, Sastry AV, Szubin R, Rychel K, Chauhan SM, et al. Advanced transcriptomic analysis reveals the role of efflux pumps and media composition in antibiotic responses of Pseudomonas aeruginosa. Nucleic acids research. 2022;50(17):9675-88.

[41] Zhang R, Ou HY, Zhang CT. DEG: a database of essential genes. Nucleic acids research. 2004;32(suppl_1):D271-2.

[42] Nguyen E, Poli M, Durrant MG, Kang B, Katrekar D, Li DB, et al. Sequence modeling and design from molecular to genome scale with Evo. Science. 2024;386(6723):eado9336.

[43] Kanehisa M, Sato Y, Kawashima M, Furumichi M, Tanabe M. KEGG as a reference resource for gene and protein annotation. Nucleic acids research. 2016;44(D1):D457-62.

[44] Madin J, Nielsen D, Brbic M, Corkrey R, Danko D, Edwards K, et al.. A synthesis of bacterial and archaeal phenotypic trait data. Sci Data 7: 170; 2020.

[45] Weimann A, Mooren K, Frank J, Pope PB, Bremges A, McHardy AC. From genomes to phenotypes: Traitar, the microbial trait analyzer. MSystems. 2016;1(6):10-1128.

[46] Shrikumar A, Greenside P, Kundaje A. Learning important features through propagating activation differences. In: International conference on machine learning. PMlR; 2017. p. 3145-53.

[47] Madani A, Krause B, Greene ER, Subramanian S, Mohr BP, Holton JM, et al. Large language models generate functional protein sequences across diverse families. Nature Biotechnology. 2023;41(8):1099-106.

[48] Watson JL, Juergens D, Bennett NR, Trippe BL, Yim J, Eisenach HE, et al. De novo design of protein structure and function with RFdiffusion. Nature. 2023;620(7976):1089-100.

[49] Hayes T, Rao R, Akin H, Sofroniew NJ, Oktay D, Lin Z, et al. Simulating 500 million years of evolution with a language model. Science. 2025:eads0018.

[50] Lal A, Garfield D, Biancalani T, Eraslan G. regLM: Designing realistic regulatory DNA with autoregressive language models. In: International Conference on Research in Computational Molecular Biology. Springer; 2024. p. 332-5.

[51] DaSilva LF, Senan S, Patel ZM, Reddy AJ, Gabbita S, Nussbaum Z, et al. DNA-Diffusion: Leveraging Generative Models for Controlling Chromatin Accessibility and Gene Expression via Synthetic Regulatory Elements. bioRxiv. 2024.

[52] Lee SY, Kim HU, Chae TU, Cho JS, Kim JW, Shin JH, et al. A comprehensive metabolic map for production of bio-based chemicals. Nature catalysis. 2019;2(1):18-33.