Fast and accurate taxonomic domain assignment of short metagenomic reads using BBERT

Dmitry Alekhin[1,*], Moshe Alon[1,2,*], Tomer Sidi[3,*], Stav Perez[1,4], Gon Carmi[5], Omri M. Finkel[2,✉], Amir Erez[1,✉]

[1] Racah Institute of Physics, The Hebrew University of Jerusalem, Edmond J. Safra Campus - Givat Ram, Jerusalem, 9190401 Israel

[2] Department of Plant & Environmental Sciences, The Alexander Silberman Institute of Life Sciences, The Hebrew University of Jerusalem, Edmond J. Safra Campus - Givat Ram, Jerusalem, 9190401 Israel

[3] Department of Computer Science, University of Haifa, Haifa 3303221, Israel

[4] Benin School of Computer Science and Engineering, The Hebrew University of Jerusalem, Edmond J. Safra Campus - Givat Ram, Jerusalem, 9190401 Israel

[5] Institute of Plant Sciences, Newe Ya'ar Research Center,
Agricultural Research Organization - Volcani Institute,
Ministry of Agriculture, P.O.B. 1021 Ramat Yishay, 30095, ISRAEL.

* These authors contributed equally.
✉ Corresponding authors, omri.finkel@mail.huji.ac.il and amir.erez1@mail.huji.ac.il

Abstract

Biological diversity revealed by metagenomic sequencing far exceeds that of known or cultured organisms, yet much of this diversity remains inaccessible because most sequences from complex habitats, such as soil, lack database references. The majority of soil-derived reads cannot be assembled or annotated and are designated as “microbial dark matter.” Recent advances in large language models provide an opportunity to classify such unannotated sequences without relying on assembly or reference databases.

We present BBERT, a BERT-based language model trained to distinguish whether a metagenomic read originates from bacteria. Using an out-of-distribution detection framework, BBERT flags reads that diverge from learned bacterial patterns as non-bacterial. By focusing on this fundamental split in the tree of life and training only on bacterial sequences, BBERT achieves high accuracy and enables large-scale analyses of soil metagenomes. Testing across 1,971 soil metagenomes, BBERT reliably identified bacterial reads, predicted coding potential, and determined reading frames directly from short reads, without assembly. Applying BBERT enhances downstream analyses and assembly performance, enabling a more robust analysis of microbial processes and their environmental drivers.

Introduction

Microbial ecosystems embody immense complexity. Understanding this complexity is essential for elucidating the processes that sustain the biosphere, a task that relies on the analysis of massive datasets. Today, such datasets are abundant: thousands of publicly available metagenomes, spanning every biome on Earth, can be freely downloaded. Among these, soil metagenomes are extraordinarily important for us to understand. Soil is the substrate on which all terrestrial life, including humanity, thrives. Yet, despite centuries of study, we still have only a partial understanding of what makes a healthy soil [1]. Soil habitats contain significant and diverse representations of all domains of life: Bacteria, Archaea, Eukaryota, and their viruses. Since many species found in soil are absent from any database[2–5], current annotation tools are unable to taxonomically classify many, or in some cases most, of the DNA reads even at the domain level. This phenomenon is commonly referred to as "microbial dark matter" [6,7]. The partiality of the databases forces existing bioinformatic tools to employ heuristics to classify metagenomic reads[8,9] which causes many of the reads to be either unannotated or mis-annotated. Notably, a significant and variable portion of this "dark matter" in soil metagenomes appears to be non-bacterial, much of it eukaryotic in origin[10–12]. A major goal of metagenomic sequencing is comparing the abundance of microbial genes between differing environmental conditions. An accurate measurement of functional profiles from soil metagenomes requires correctly identifying and separating eukaryotic and bacterial DNA reads within a given metagenome. Otherwise, it is not possible to correctly normalize bacterial gene abundances and obtain a comparable set of values between different systems. This problem is exacerbated by the vast difference in genome structure and size between prokaryotes and eukaryotes. Without a reliable domain-level classification, the values of neither the numerator nor the denominator for functional abundance calculations are known[10–12].

To address the current uncertainty in read taxonomic assignment, as well as the large fraction of unannotated and misannotated reads in soil metagenomes quickly and accurately, we developed BBERT, a new database-independent method to classify short metagenomic reads to either bacterial or nonbacterial. BBERT is a BERT-based large language model that addresses a simple question: is a given metagenomic read bacterial in origin? To overcome the sparsity of non-bacterial reference genomes available, we used an out-of-distribution detection framework into BBERT, flagging reads that differ from the learned bacterial patterns as non-bacterial. By focusing on the primary divergence in the tree of life and training only on bacterial reads, we were able to achieve high accuracy and speed even with a modestly-sized model. Thus, BBERT overcomes both throughput and accuracy limitations of previous LLM-based methods[13]. Using BBERT, we were able to perform large-scale analyses essential for comprehensive exploration of short read soil metagenomes, including its unexplored biodiversity.

Results

Reference-based tools cannot classify most metagenomic reads at the domain level

Existing methods for analyzing soil metagenomes often include de-novo assembly and binning of contiguous regions (contigs)[14]. By placing metagenomic reads into longer genomic context, the classification of reads mapped to contigs becomes an easier task. Examples for such assembly-based tools include EukRep[15] and EukFinder[16], all of which perform well on data that can be assembled. Indeed, the inclusion of metagenomically-assembled genomes has vastly expanded the microbial diversity in databases[17]. But assembly-based approaches consider only what the algorithms are able to assemble, and this represents the minority of the data. We tested this assertion by assembling 478 soil metagenomes, and found that about 67%±16% of the read data could not be mapped to any of the contigs (Fig. S1). Therefore, assembly-based tools ignore the majority of biodiversity in soils.

Assembly-free methods attempt to directly classify individual metagenomic reads, allowing us to gain information from all available sequence data. Assembly-free approaches include direct alignment to reference protein databases (e.g., using DIAMOND[18] on the Kyoto Encyclopedia of Genes and Genomes KEGG[19], or NCBI's non-redundant protein database *nr*[20]), k-mer matching (Kraken2[21]), and the use of marker genes that provide only taxonomic classification (MetaPhlAn[22]). While these methods have proven instrumental to interpret metagenomes, their potential is capped by their dependence on external reference databases. Approaches that attempt to transcend the limitations of reference databases include the use of hidden Markov models (HMMs), that attempt to find probabilistic features in a collection of similar reads. However, even state-of-the-art HMM-based tools such as SingleM[23], that uses HMMs for single-copy phylogenetic marker genes, do not classify the vast majority of the reads, as they do not contain such markers. Thus, while SingleM may reliably estimate bacterial composition and the fraction of bacterial reads, it cannot classify all the individual reads, limiting its use to bulk averages.

To gauge the ability of existing tools to reliably partition metagenomes at the domain level, we curated a global soil dataset of publicly available metagenomes, featuring 1971 bulk soil samples (Fig.1a and SI Table S1). We then used two of the aforementioned computational tools to measure the domain-level nucleotide fractions within each of the 1971 samples. First, we used Kraken2[24], which uses kmer frequencies, matched to its reference dataset, to taxonomically classify reads. Using Kraken2's default standard database plus protozoa and fungi resulted in an average prokaryote fraction of less than 15.2% ± 10.5% with an unidentified reads comprising 82.6% ± 12.4% of the data, and only 0.1% ± 0.2% of the reads classified as eukaryotic (SI Fig. S2a). Since these results are not plausible, we reasoned that the default Kraken2 database is unsuitable for analyzing soil metagenomes. In order to customize kraken2 to the soil metagenomes we used here, we curated a custom reference database for it, consisting of 222,337 (SI Table S2) prokaryotic isolate and metagenomically-assembled genomes (MAGs), removing commonly sequenced clinically-relevant taxa which heavily skews the composition of the genome database (*cf*. Methods). Since the current database coverage of

eukaryote genomes is grossly insufficient (<9,000 reference genomes, overwhelmingly from animals, plants and fungi, out of millions of species), we did not include eukaryotes in our custom database, and simply used Kraken2 to estimate the number of prokaryotic reads in each sample. To validate the Kraken2 estimates, we used SingleM[25], which uses pre-trained profile-hidden-Markov-models (pHMM) of universal single-copy genes. Although SingleM does not classify all individual reads, but only the small fraction of reads that contain single-copy genes, it can be used to estimate the prokaryotic fraction within a metagenome. Had Kraken2 and SingleM yielded similar results, we would have concluded that Kraken2 is a practical solution for assigning taxonomy to soil metagenomic reads. We therefore compared the estimated prokaryote fraction generated by both tools (Fig. 1b and SI Fig. S2b). As is apparent, the two tools strongly disagree, thereby exposing a gap in our ability to accurately partition metagenomes at the domain level.

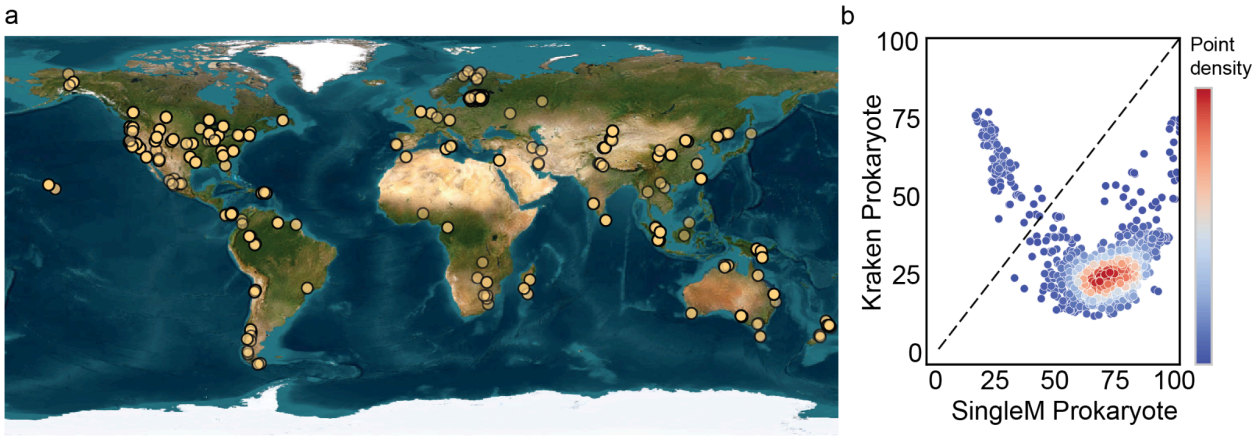

Fig. 1: Existing state-of-the-art tools show conflicting prokaryote fractions. a, The metagenomic dataset used in this study and its global distribution, compiled from 1971 soil samples. **b,** The imputed bacterial percent of each sample, calculated using SingleM[26] on the *x*-axis and Kraken2[24] on the *y*-axis using our custom 222,337 prokaryote database, shown color-coded according to the local density.

BBERT performs consistently well on data it was not trained on

LLMs have recently emerged as powerful tools for classifying metagenomic reads by capturing complex patterns in sequence data without relying on reference databases[27,28]. Models like Evo2[29], BERTax[13], DeepMicroClass[30] and Tiara[31], have used deep learning to improve taxonomic classification of long reads. However, these tools either perform inaccurately and inefficiently on short reads (SI Table S3), or not at all, as in the case of DeepMicroClass which has a 500 bp lower limit. Crucially, of the approximately 100 petabytes of sequencing bases in the SRA archive, about 20 petabytes are from metagenome samples[32]. Almost all existing metagenomic data, and most of the metagenomic data generated today, are composed of short reads, at most 300 bp long, with much of the earlier data comprising 100 bp-150 bp reads.

LLMs inherently capture complex dependencies within DNA sequences via their attention mechanisms, enabling them to generalize beyond the training data and to perform reference-free classification of all reads. Such functionality is particularly valuable for identifying

sequences absent from the training data, as LLMs can discern patterns characteristic of novel or previously unrepresented organisms[33]. However, current deep learning-based tools for DNA sequence analysis are prohibitively slow when applied to large numbers of short metagenomic reads (100–250 bp), restricting their use to small datasets or to long-read data[13,29]. This computational inefficiency substantially limits their utility for the large-scale metagenomic analyses needed to comprehensively explore soil biodiversity.

To overcome the limitations of existing LLM-based tools, we developed BBERT, an LLM built on Google's BERT architecture, that can quickly and accurately classify all short metagenomic reads without relying on reference databases. BERT is based on a self-attention mechanism[34,35], which considers the importance of each token (here, nucleotide) according to its context, and thus is well suited for DNA-read classification. DNA sequencing yields bidirectional fragments whose absolute genomic coordinates are unknown; BERT's bidirectional encoding therefore captures both upstream and downstream nucleotide dependencies. By exploiting these capabilities, BBERT enables efficient, large-scale classification of short metagenomic reads without reliance on extensive reference databases. Since the current database coverage for bacterial genomes is much more complete than for the eukaryotic-archaeal branches of the tree of life[36], we adopted an out-of-distribution (OOD) detection approach by training exclusively on bacterial sequences. The OOD framework is designed to distill the universal properties of the bacterial sequences. BBERT was trained and optimized for short reads, requiring only 100 bp to reach a conclusion quickly, opening up enormous archives.

BBERT identifies reads that do not conform to the learned bacterial sequence patterns, flagging them as non-bacterial. To this end, we pre-trained BBERT to infer masked nucleotides within reads, thus making it sensitive to the nucleotide context rather than pre-defined sequence patterns (see Methods). For the pre-training, we randomly selected two genomes out of every bacterial family from the ~200K bacterial genomes we curated, a total of 1519 genomes, and used InSilicoSeq to create a simulated metagenome composed out of all (see methods and SI Table S2). To discriminate bacteria from non-bacteria, we first used a cutoff based on the mean cross-entropy loss of the masked residues. To determine the cutoff, we ran BBERT on 10 validation datasets composed of species that do not appear in the training data (see Methods). The optimal cutoff threshold for separating bacterial and non-bacterial reads proved to be universal, holding true regardless of which species were included in the validation (Fig. S3a). Furthermore, the distribution of scores for bacterial and eukaryotic reads forms two distinct peaks (Fig. S3b). To assert that BBERT captures true signatures in DNA (and not GC content or similarly trivial markers), we shuffled the order of the bases in the reads in the validation dataset. This resulted in an identical distribution of scores for both bacterial and eukaryotic shuffled reads (Fig S3b). The fact that after shuffling, reads from the two domains produce an identical distribution of scores indicates that BBERT learns actual bacterial patterns and can discern between bacteria, eukaryota and random DNA. Having set the universal cut-point using the validation data, we proceeded to test the classifier on a test dataset. The reproducibility of our approach is underscored by near-identical ROC curves across 10 non-overlapping test sets (Fig. S3c, details in Methods).

BBERT embeddings correctly classify domain, coding vs. non-coding reads and identify open reading frames

Thus far, we have interrogated only one dimension of BBERT's classification capability for metagenomic reads: its penalty score, which quantifies the divergence of each read from the bacterial-sequence patterns learned by the model. Yet LLMs operate within a high-dimensional latent embedding space; upon classification, BBERT projects each read into this representation. We hypothesized that BBERT's high-dimensional embedding space encodes the bacteria/non-bacteria split, along with additional interpretable features of read content, which may manifest as distinct clusters. To test this, we generated a mock metagenomic dataset of synthetic sequencing reads derived from two reference genomes (see Methods), *Saccharomyces cerevisiae* (eukaryote) and *Pseudomonas aeruginosa* (bacterium). We then inspected the 78,336-dimensional space (size of hidden layer 768 x 102 bases including start and stop tokens) using Principal Component Analysis (PCA, Fig. 2a). As is apparent, PC1 separated the bacterial from eukaryotic reads, and moreover, there appears deeper structure within PC2. To capture non-linear relationships in BBERT's embedding space, we applied t-distributed stochastic neighbor embedding (t-SNE[37]), labeling reads as bacterial or non-bacterial and as protein-coding or non-coding according to their predominant nucleotide composition. t-SNE clearly separated the bacterial and non-bacterial reads, and furthermore, within each group, separated the protein coding and non-coding reads (Fig. 2b). Among the coding reads, there appeared six distinct clusters for each species. We hypothesized that these six clusters represent the six reading frames in protein-coding data. Therefore, we randomly selected several reads from each frame and labeled them (Fig. 2c). As is apparent, the different frames mapped almost perfectly to the different protein-coding clusters. These clustering patterns generalize beyond the two species examined in Fig. 2: embeddings from additional taxa exhibit analogous taxon-taxon separation and within-genome substructure (Fig. S4). BBERT outputs per-read embeddings (see User Manual on github), enabling users to reproduce these analyses or interrogate their own datasets with standard dimensional-reduction and clustering workflows that exploit this biologically interpretable latent space.

The results presented in Fig. 2a-c suggest that it would be useful to train a classifier on the BBERT embeddings. Indeed, when applying BERT for classification tasks, it is common practice to freeze the model weights and train a fully connected layer on the embeddings. We have trained three such classifiers on top of the BBERT embeddings (see Methods), used for determining: (i) bacterial vs. non-bacterial; (ii) coding vs. non-coding; (iii) open reading frame.

As expected, the accuracy of identifying bacterial reads increased from 76.4% ± 2.9% using BBERT's loss cutoff (Fig. S3), to an accuracy of 89.4% ± 2.0% when performed on the same 10 test datasets described above (Fig. S5). Furthermore, we have obtained 87.5% ± 0.3 accuracy for identifying coding vs. non-coding reads, and an accuracy of 95.5% ± 0.6% when identifying the open reading frame in protein-coding reads (Fig. 2d). Thus, we demonstrated that a simple classifier on the BBERT embeddings correctly identifies the domain, whether a read is protein-coding, and its correct reading frame.

BBERT outperforms state-of-the-art deep-learning classifiers

We compared BBERT's performance to two state-of-the-art deep-learning classification tools: BERTax[13] and Tiara[31]. Another recent classification tool, DeepMicroClass[30], only accepts reads that are 500bp long, and so could not be fairly evaluated. BBERT outperformed BERTax in both accuracy and speed (SI Table S3). BBERT's mean accuracy for domain classification on the 10 test sets is 89.4% ± 2.0%, compared with 71.9% ± 4.1% for BERTax and 72.1% ± 4.6% for Tiara[31] (Fig. S5, SI Table S3). Moreover, BBERT processed 13,400 reads per second when running on a dedicated Nvidia A100 GPU node, compared to 25 for BERTax and 408 for Tiara (SI Table S3). Running BBERT on a Windows 10 personal computer with an NVIDIA RTX 3090 GPU card clocked 8,480 reads per second, meaning it is usable without a high performance computing cluster, a feat unfeasible with most alternatives (BERTax: 17 reads per second, Tiara: didn't install on a Windows machine). Indeed, one may run BBERT freely on a Google Collab T4 machine (2390 reads per second) and still be able to process a typically sized metagenome in the free time window. BBERT ran on a Macbook Pro M3 at 560 reads per second. Taken together, these results demonstrate that BBERT is a fast and accurate tool to partition large amounts of short metagenomic reads at the domain level, including the large fraction of unannotated or unassembled reads.

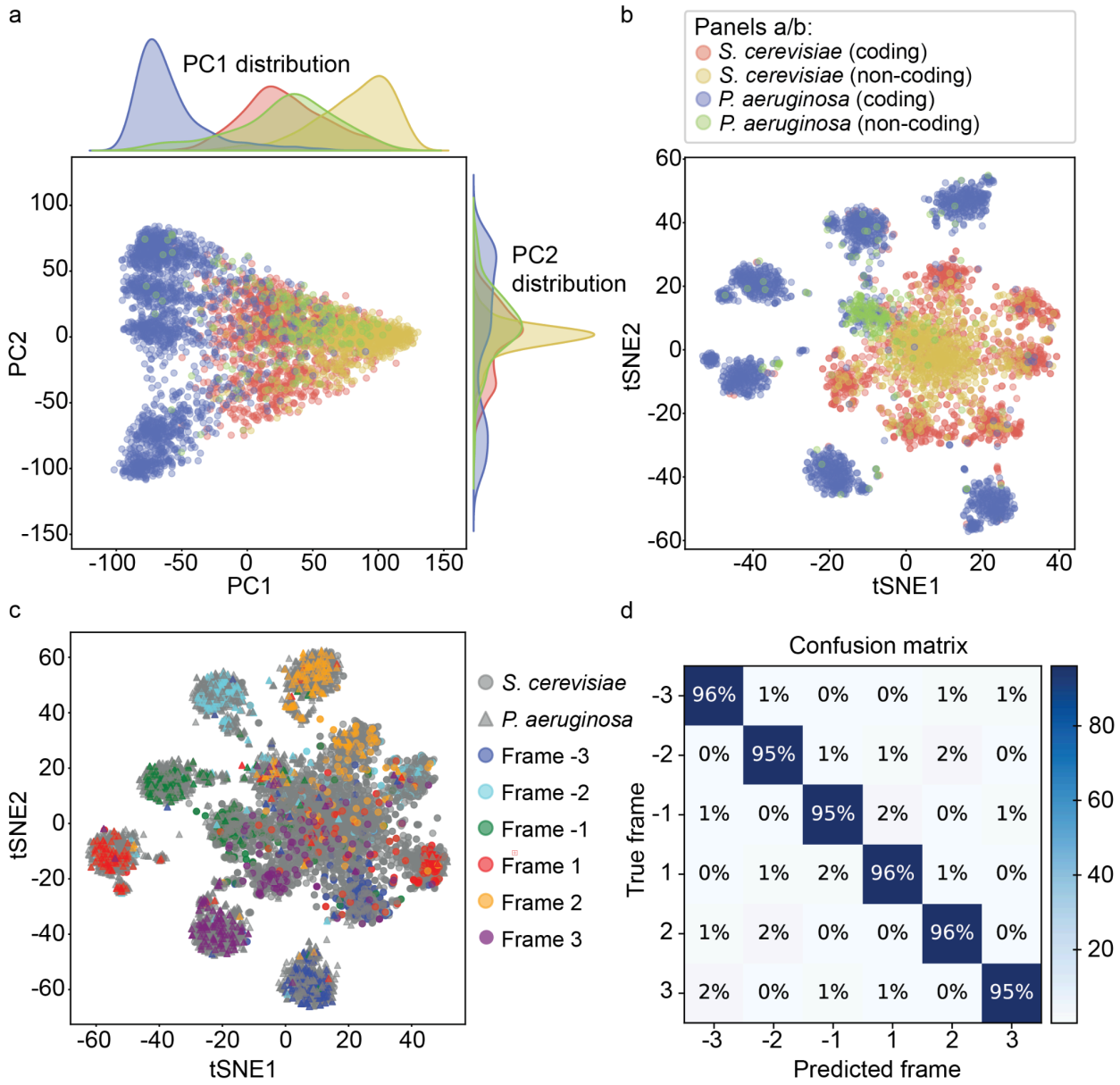

Fig. 2. BBERT differentiates between bacteria and non-bacteria, coding and non-coding reads and identifies the reading frame. Two-dimensional visualization of the 78,336-dimensional embedding by BBERT of short reads generated from *Pseudomonas Aeruginosa* (blue/green) and *Saccharomyces Cerevisiae* (orange/yellow) using **a,** principal component analysis (PCA) and **b,** t-SNE[37]. The colors represent the coding and non-coding reads. **c,** t-SNE representation of the embedding by BBERT of the same reads as in (b), labeling several reads from different reading frames (colors) and from the two species (circles and triangles), and plotting their location in the embedding space. **d,** Confusion matrix showing BBERT's reading frame classification accuracy on a test dataset of 100 bp reads taken from within annotated ORFs from RefSeq genomes (see SI Table S8).

BBERT sorts soil metagenomic reads without taxonomic bias and agrees with SingleM estimates

We applied BBERT to our curated set of 1971 publicly available soil metagenomes. To compare the BBERT results with SingleM, we plotted for each metagenome the number of prokaryotic bases estimated by SingleM, vs the total number of bases and the bases deduced as bacterial by BBERT. Whereas before filtering, the metagenomes contained 33% more bases than deemed prokaryotic by SingleM (slope 1.33 ± 0.01), after BBERT filtering, this number was reduced to 11% (slope 1.11 ± 0.00, Fig. 3a). SingleM, however, estimates the prokaryotic fraction only using single copy genes without assigning each read individually, in contrast to BBERT which classifies all reads. To verify that BBERT filtering is not taxonomically biased, we used SingleM to profile the community composition of all metagenomes, at the family level, before and after filtering. Remarkably, with the exception of a handful of metagenomes, this composition remains nearly unchanged after BBERT filtering, demonstrating that BBERT does not bias by taxonomy (Fig. 3b).

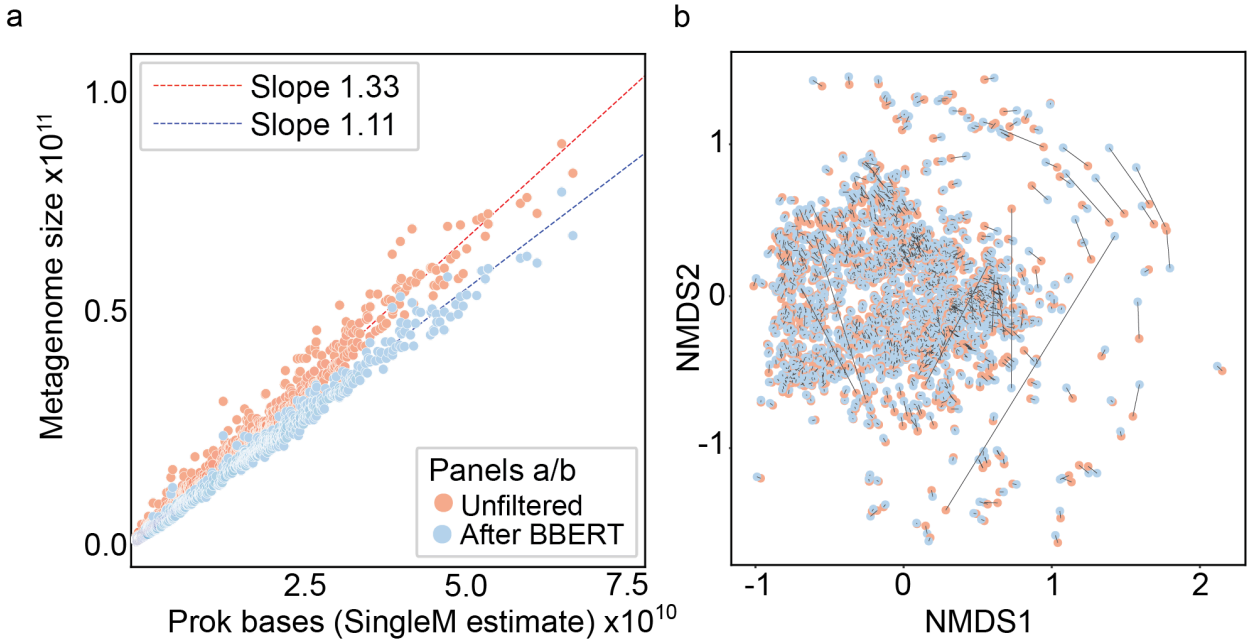

Fig. 3. BBERT does not bias bacterial taxonomic composition and identifies the bacterial DNA fraction in agreement with SingleM. a, Analysis of all 1971 soil samples reveals a striking agreement between the estimated number of prokaryotic nucleotides (by SingleM) to the actual nucleotides classified by BBERT as bacterial. **b,** MDS plot of the family-level taxonomic composition, as inferred by SingleM, before and after BBERT filtering.

The "dark-matter" in soil metagenomes is mostly non-bacterial

Soil metagenomes are notoriously difficult to functionally annotate[3,6–8,38], with only a fraction of total data reliably annotated. We hypothesized that the majority of unannotated reads, the so-called "dark matter", is nonbacterial. To test this, we mapped the BBERT-assigned bacterial and non-bacterial fractions from all 1971 metagenomes to the KEGG prokaryotic database, and found that at an E-value cutoff of 10^{-5}, 59.1% ± 8.0% of the bacterial reads were functionally

assigned, compared to 24.8% ± 7.5% in the non-bacterial fraction. This trend persists for a range of E-value cutoffs (Fig. 4a). Therefore, we propose that a significant portion of the "dark matter" consists of non-bacterial DNA, likely non-coding eukaryotic sequences. Moreover, when the percentage of reads mapped is plotted against $-\log_{10}(E)$, the data exhibit a linear relationship. The slope of this regression may serve as an indicator of read-to-database similarity, and the steeper slope observed for BBERT-sorted bacterial reads indicates a closer correspondence with KEGG. In summary, focusing exclusively on bacterial reads enables concise functional annotation for the majority of metagenomic sequences.

BBERT reveals functional differences between metagenomes from temperate and boreal-arctic environments

Measuring the abundance of a gene encoding a bacterial function in soil requires knowing the fraction of bacterial DNA that is present within it. Since BBERT assigns all reads at the domain level, we expect it to alter gene abundance estimation as it affects both the numerator and denominator in the abundance calculation. To test how BBERT affects the results of comparative metagenomics projects in soil, we considered the KEGG annotation of the prokaryote pathways from a recent global survey of soil metagenomes[38] (see Methods). This study compared metagenomes from tropical, temperate and boreal-arctic environments. When the abundances are normalized by the full metagenome size (as in the original publication), 27 pathways significantly differed between the boreal-arctic and tropical environment, and 30 pathways significantly differed between the temperate and tropical environment; but there appear no statistically significant differences in the relative abundance of KEGG pathways between the temperate and arctic regions, which is unexpected, considering how different these two biomes are from one another (Fig. 4b and Fig. S6a), details in SI Table S4). In contrast, applying BBERT filtering to partition the bacterial reads reveals 7 pathways that differ significantly between boreal-arctic and temperate environments (Fig. 4c and Fig. S6b). BBERT filtering therefore reshaped the biological interpretation of KEGG-based functional profiling, uncovering ecological differences between boreal-arctic and temperate environments that standard analyses failed to detect.

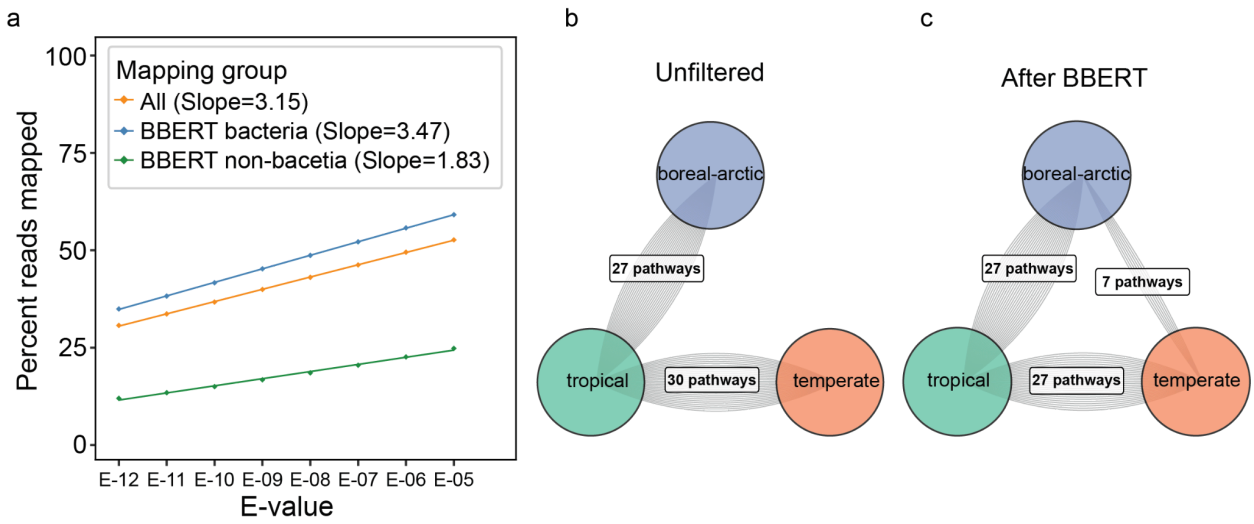

Fig. 4 **Functional implications of BBERT-sorting the bacterial reads in soil metagenomes. a,** Percentage of the reads mapped to KEGG prokaryote proteins from all soil metagenomes in our dataset. The values (dots) are shown for all reads (orange), and the reads sorted by BBERT as bacterial (blue) and non-bacterial (green). The dots are fitted by a straight line with the slopes reported on the figure. **b,c** Comparison of the statistical significance in KEGG prokaryote pathway abundances between metagenomes originating from tropical, temperate and arctic regions from Bahram et. al[38] when normalized by the full metagenome size (b) and by using only the reads sorted by BBERT as bacterial (c).

BBERT accelerates de-nono metagenomic assembly with fewer errors

Could BBERT be used to improve de-novo assembly of metagnomic reads? Since eukaryotic DNA is notoriously difficult to assemble, and de-novo assembly of bacterial metagnomes is a difficult task even without eukaryotic contamination, we reasoned that using BBERT to select only the bacterial reads would improve metagenomic assemblies without sacrificing coverage. To test this, we created four non-overlapping synthetic metagenomes with a mix of short reads from eukaryotic and bacterial genomes (see Methods), and assembled them before and after BBERT filtering using metaSPAdes[39]. We found that BBERT filtering resulted in 90% of the assembled genomic coverage obtained by the unfiltered data (Fig. 5a, SI Table S7). However, this reduction in coverage was not uniform across genomes. Of the 512 bacterial genomes used to construct the synthetic metagenomes, more than half exhibited less than 5% coverage reduction compared with the unfiltered data (Fig. 5b, Fig. S7). Moreover,this 10% reduction in coverage came with two significant benefits: (i) run time was reduced x2.7, from 66±8 hours to 25±2 hours per metagenome, which is beneficial since de-novo assembly is an extremely resource intensive process; (ii) when counting assembly errors compared to the ground-truth genomes, the BBERT-filtered data resulted in half of the gaps (Fig. 5c, Fig. S7) and 77% of the mismatches (Fig. 5d, Fig. S7) compared to the unfiltered data. In summary, using BBERT before de-novo assembly saves significant costs in compute time and generates truer contigs, while retaining almost the same coverage as unfiltered data.

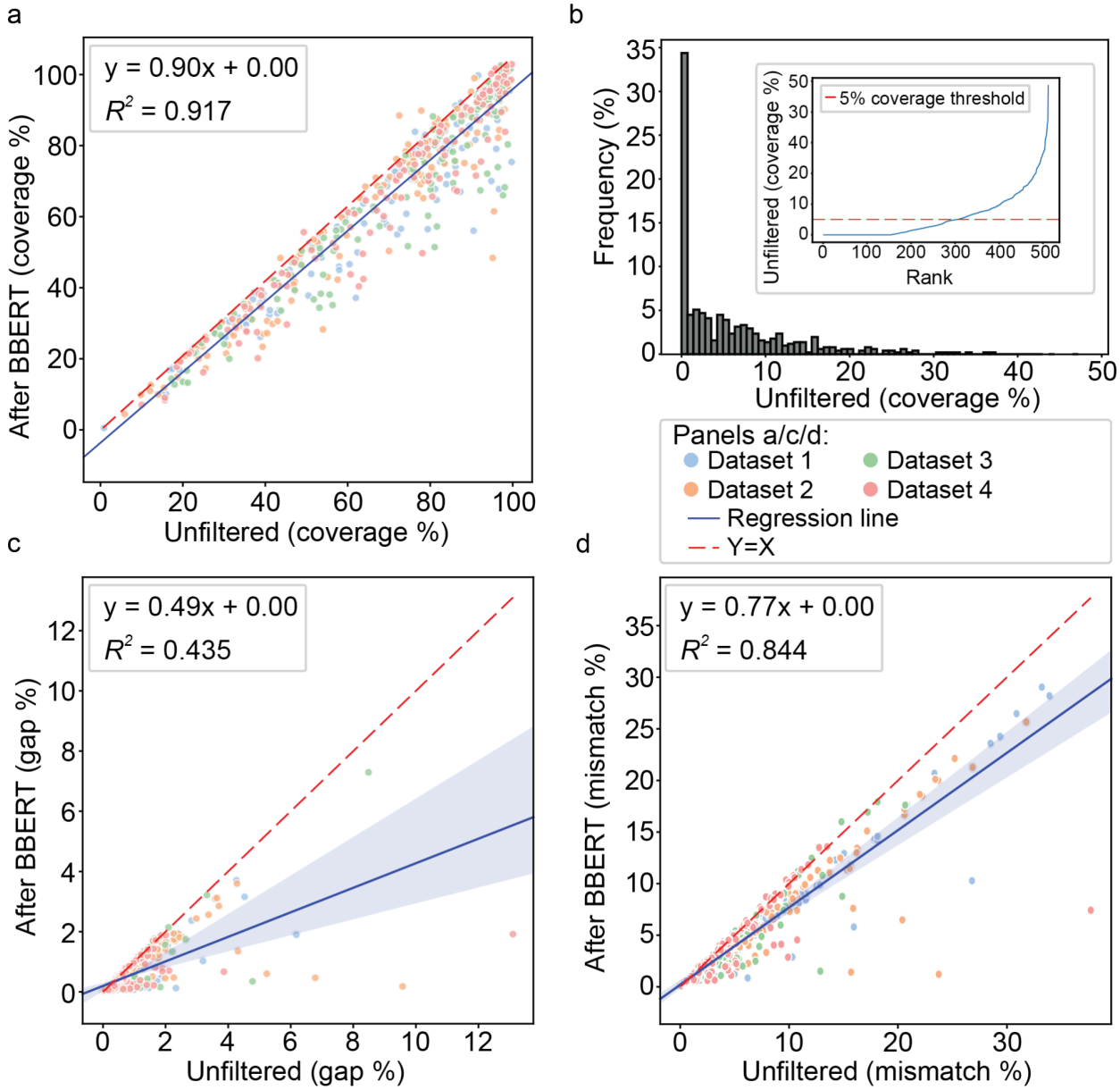

Fig. 5 BBERT-based filtering improves de novo assembly on synthetic datasets. a, Genome coverage of assembled contigs before and after BBERT classification. Each dot represents one of 128 bacterial genomes in each of four datasets (512 genomes total). The regression line indicates that, on average, 90% of the genomes covered in unfiltered assemblies are also covered in filtered assemblies. **b,** Histogram of the percentage of each genome covered exclusively in the unfiltered assembly. Inset: the same data ranked by coverage, showing that more than half of the genomes differ by <5% between filtered and unfiltered assemblies. **c,** Percentage of gaps relative to the assembled genome in unfiltered versus filtered assemblies. The regression line shows that filtering reduces gaps to 49% of those observed in unfiltered assemblies. **d,** The percentage of mismatched nucleotides in contigs, showing that filtered assemblies retain only 77% of the mismatches found in unfiltered assemblies.

Discussion

We used BBERT to address a fundamental question arising at the earliest split in the tree of life: whether a DNA sequence originates from a bacterial genome. We integrated an out-of-distribution detection framework into BBERT, flagging reads that differ from the learned bacterial patterns as non-bacterial. By focusing on the primary divergence in the tree of life and training only on bacterial reads, we were able to achieve high accuracy even with a modestly-sized model. Thus, BBERT overcomes the throughput limitations of previous LLM-based methods while preserving high accuracy, classifying a typically-sized soil metagenome within a couple of hours. As a result, we are able to perform large-scale analyses essential for comprehensive exploration of soil metagenomes, including its unexplored biodiversity.

Our analysis, using a newly published tool, SingleM[26], estimates that on average, about 32% of soil metagenomic sequence reads are not from prokaryotes (Fig. 1b). The presence of such large quantities of non-bacterial DNA has significant implications for the computation of functional profiles from metagenomic datasets as well as for comparative analyses. Yet SingleM, however powerful, is limited, since it only estimates prokaryotic fractions based on a small set of marker genes, and is unable to classify individual reads. We address this need with BBERT, our large-language-model trained to recognize bacterial reads. Remarkably, BBERT yields an estimate of the prokaryotic fraction that closely resembles SingleM's, while additionally classifying every read as bacterial or non-bacterial (Fig. 3).

In theory one could solve most of the problem of assigning taxonomy by correctly assembling and binning the short metagenomic reads. Such longer contigs are much easier to classify [13,29–31]. However, such an assembly-based approach is severely limited given current data and computational tools. We showed this by assembling 478 soil metagenomes, where 67% of the reads were not assembled at all (Fig. S1). Our tool, BBERT, innovates by using LLM technology to assign taxonomy to short reads. In comparison, existing tools have been trained on long reads which do not represent the many petabytes of existing sequence data[13,29].

Although binary classification of reads requires only a single score, BBERT's high-dimensional embedding space captures rich, multidimensional features, including properties beyond those it was trained for. Analysis of BBERT's embedding space in a mock dataset revealed not only clear domain-level separation but also seven discrete clusters of reads within each genome: one comprising non-coding fragments and six corresponding to the six possible coding reading frames (Fig. 2). This observation suggests substantial potential to enhance existing bioinformatic algorithms by leveraging these embedding-derived structural features. Training classifiers on the embedding space resulted in excellent inference not only of the taxonomic domain, but also whether a read is protein coding and its correct reading frame. Thus BBERT can improve both the speed and accuracy of many downstream tools.

One open question in the metagenomics community is the origin of the so-called “dark matter”, sequence reads with no assigned taxonomy or function. We found that most of the reads that could not be annotated by KEGG originate from the non-bacterial fraction, whereas the bacterial fraction is mapped successfully by KEGG (Fig. 4a). Therefore, it stands to reason that most of the “dark matter” in soil metagenomes is from eukaryotic genomes.The presence of large fractions of non-bacterial and unannotated sequences, at varying levels, among datasets, hinders our ability to correctly analyze and compare different habitats to one another. Comparison requires that the data is correctly normalized to sequencing depth. With a large and unknown amount of eukaryotic DNA in a sample, it is not possible to know the true sequencing depth of the bacterial genomes in a sample. Here, we demonstrated how this can influence ecological inferences, where we re-analyzed a global metagenomic survey[38]. Using BBERT revealed seven bacterial metabolic processes that are differentially abundant between soils from temperate and boreal-arctic habitats, where no significant differences were detected prior to BBERT-filtering. (Fig. 4b-c). Therefore, BBERT can reveal hidden ecological relationships in metagenomic data.

Assembly and binning of metagenomic data into putative genomes remains a major challenge in metagenomics. As we have seen here, most metagnomic reads do not assemble at all (Fig. S1). But also the contigs that are generated de-novo cannot be validated, in the absence of genomic references. Applying BBERT to the data prior to assembly, halves the number of erroneous gaps, and reduces mismatches by a quarter. This improvement in assembly quality comes with a cost of a 10% loss of alignment coverage but with an almost three-fold speedup in run time, which translates to saved costs (Fig. 5). An interesting direction to explore is how BBERT could be used to improve de-novo assemblies from metagenomes by adopting a combined approach: assemble the data before and after BBERT filtering. For regions that overlap between the two assemblies, use the assembly of the filtered data, and fill in gaps from the assembly of unfiltered data. BBERT is ready to be used to improve the reliability of metagenomic assemblies.

When sequencing metagenomes, especially from environmental samples, one is bound to encounter a large fraction of genomes not represented in existing references. Large-language-models are excellent at generalizing classification from partial training data and are therefore well-suited for analyzing environmental metagenomes. The research community has recognized this potential and we have seen an explosion of recent and high quality tools which utilize enormous computational power to derive deep insights into DNA[13,27,27,29–31]. Here, we showed that by selectively using an LLM on a sharp question, trained with an out-of-distribution approach, we can derive deep biological insight from reads that are 100 bases long, requiring only modest computational power, and making our tool accessible to everyone. Thus, LLM technology has the power to transform our understanding of microbial ecology and boost ecological genomics at large.

Methods

BBERT is a BERT-based model developed for out-of-distribution (OOD) detection of nucleotide sequences originating from bacterial genomes. Here, we outline a detailed description on BBERT's training and evaluation (Section 1). Then, the design of two synthetic scenarios used to implement BBERT within the context of downstream tasks, namely frame classification and metagenome assembly (Section 2). Finally, we describe the methodological approach for applying BBERT to complex environmental datasets, with a focus on soil metagenomes (Section 3).

1. BBERT Training and Evaluation

To create a nucleotide-based LLM for OOD detection of bacterial genomes, we adopted three phases of development: (i) pretraining the BERT architecture only on bacterial genomic reads, (ii) calibrating the classification threshold for the pretrained model scores, and (iii) evaluation on the bacterial read classification task with comparison to BERTax[13] and Tiara[31].

BBERT pretraining was performed on a bacterial dataset of sampled synthetic short nucleotide sequences, each 100 bp in length, designed to simulate metagenome reads and their errors (see *Datasets* and *Preprocessing* sections). For the training process, we used the Masked Language Model objective, i.e. masking 30% of tokens in the input and optimizing the model ability to infer those tokens from context, which enables the model to learn representation of missing nucleotides based on those reads. Notably, we use the average cross entropy loss calculated on the masked nucleotides only. The pretraining optimization process was conducted in sessions of 5 to 10 epochs until an optimal result was achieved, saving all intermediate checkpoints (model weights). For each checkpoint, we evaluated the classification accuracy on the calibration dataset (see the *Datasets* section) and selected the optimal checkpoint accordingly. Typically, we observed simultaneous decreases in training loss and increases in classification accuracy. Training was terminated when progress stagnated, with no obvious improvement in model performance.

Next, to used BBERT as an OOD tool, we treated the cross entropy loss of the masked nucleotides as classification score and used specialized datasets to calibrate it for bacterial read classification (*OOD Calibration of Cutoff Threshold* section). Assuming bacterial reads should have lower scores than non-bacterial reads as the model was pretrained only on bacteria. The cut-off is a fixed scalar, and calculated from the optimal accuracy point on a separated disjoint calibration dataset with equal number of bacterial and eukaryotic species, with no taxonomic overlap at the genus level to the pre-training dataset. We repeated this process 10 times, on 10 different disjoint sets (i.e., none have the same genus), thus creating 10 different cutoffs. The final cutoff for using at inference time is the average of those runs, namely - the universal cutoff. Dataset details can be found in sections *Datasets* and *Dataset Preprocessing.*

To evaluate the model performance, we constructed another 10 disjoint datasets, with no taxonomic overlap at the genus level with the calibration or the pretraining datasets, namely the

evaluation datasets. We calculated each read score and classified it as bacterial or eukaryotic according to the final cutoff, reporting the overall classification metrics on those datasets. The results on this benchmark were compared to BERTax[13].

Finally, as the major application use case for such a classification procedure is on big data samples, the model throughput is an important factor in making the model applicable. Thus, another aspect of the model performance that we monitored and optimized for is the inference throughput. To this end, we conducted a small ablation test, with several model architecture sizes and model quantization, and analyzed the tradeoffs in accuracy and throughput. Model hyper-parameters are found in sections *Hyper-Parameters*, and *Hardware and Software*.

Datasets

For BBERT development and evaluation we curated two types of datasets. One for the pretraining, that comprises 1416 bacterial species (for a good representation of the bacterial tree-of-life, we randomly sampled two genomes from each bacterial family in the Refseq database), and an additional dataset for calibration and evaluation that comprises 899 bacterial species from 396 different genera and 1200 eukaryotic species from 823 of different genera. Notably, the two datasets are separated at the genus-level, and share no genus among them. Additionally, as we compare the model performance with BERTax[13], we verified that the latter does not share any genus with the published BERTax training and finetuning sets.

To account for noise in species group selection, all accessions from the calibration and evaluation dataset were divided separately by genus into 20 equal groups for bacteria and 20 equal groups for eukaryotes, with random abundances assigned following a log-normal distribution within each group. These groups were then merged pairwise to form 20 combined subsets, each with a unique genus composition from both domains. From these, 10 subsets (datasets 1–10) were assigned for score calibration, while the remaining 10 (datasets 11–20) were reserved for evaluation.

Species abundance file for the pretraining dataset can be found in SI Table S5.

Species abundance files for score calibration and evaluation sets (group 1-10 for score calibration, groups 11-20 for evaluation) can be found in SI Table S6.

Dataset Preprocessing

BBERT's is to classify metagenomic reads: short nucleotide sequences that may contain errors. These errors typically include insertions, substitutions, and deletions in DNA sequences. Thus, to train and test BBERT on data resembling typical metagenomic sequencing, we generated synthetic short reads with typical sequencer errors. The genomes were downloaded from the NCBI database (in .fna format), and using InSilicoSeq[40], we simulated reads from each .fna file using the NextSeq error model.

BBERT Model and Pretraining

BBERT is based on the Huggingface implementation of the BERT architecture (`BertForMaskedLM`). During training, the data collation process plays a critical role in preparing raw DNA reads into consistent, model-ready batches. Specifically, the collation function performs token masking, label assignment, and sequence padding, ensuring that all inputs in a batch are of uniform length and properly annotated for training loss computation. To accommodate for nucleotide sequence input, we implemented a simple character-based tokenizer, with five core tokens: A, C, T, G, and N. The latter, N, represents ambiguous or error-prone bases. To support sequence structure and model training, we added special tokens: `<s>` (start-read), `</s>` (end-read), `<pad>` (padding), and `<msk>` (mask for the masked language modeling, MLM, objective). Reserved tokens like `<cls>` (read summary), `<sep>` (read separator for multi-read inputs), and `<unk>` (unknown nucleotide) were also included for extended flexibility. During training, only reads with a length of at least 100 nucleotides were used, truncating them to a fixed length of 100 tokens. Then, 30% of the tokens in each read were randomly selected and replaced with `<msk>`, while the remaining tokens were assigned a label of -100 to exclude them from the loss computation. The model was trained to predict the original identity of the masked tokens based solely on their surrounding context, using a cross-entropy loss computed over the masked positions. This self-supervised learning strategy allows BERT to capture the underlying structure and dependencies within DNA sequences, enabling effective downstream classification. Details of the dataset generation strategy are provided in the Datasets section.

Calibration of Score Cutoff for OOD Detection

To use BBERT for read classification, we compute a score for each input sequence based on the model's cross-entropy loss during inference. This score reflects how well the model can reconstruct a given read, provided its bacteria-only training data. The core assumption is that bacterial reads, being in-distribution for BBERT, are more predictable and thus yield lower loss, whereas non-bacterial or out-of-distribution (OOD) reads will lead to higher reconstruction errors (higher loss). To transform these scores into binary predictions, we defined a cutoff threshold: reads with a loss below the cutoff are classified as bacterial, and those with equal or higher loss as non-bacterial.

To determine an optimal threshold, we generated 10 score calibration datasets (see the Datasets section) and computed the cutoff that maximized the *accuracy*, $(TP + TN) / (TP + TN + FP + FN)$, where TP represents true positives, TN represents true negatives, FP represents false positives and FN represents false negatives. Strikingly, all our calibration datasets yielded very similar cutoffs for maximal accuracy, despite having no overlap with each other (Fig. S3a). This establishes that the cutoff does not depend on the specific taxonomic composition of the metagenome.

To obtain a single decision boundary, a universal cutoff for the model, we average the cutoff values for the 10 score calibration datasets. We then applied this universal cutoff to classify

reads in the evaluation datasets. Although we recommend using this cutoff since it has been robustly tested, a user of our tool can choose a different cutoff, depending on requirements.

Model Inference

This section outlines the procedure for applying BBERT to new metagenomic read datasets. The scoring pipeline supports common input formats, including *FASTA*, *FASTQ*, and compressed *gzip* files. During preprocessing, each read was truncated to 100 characters, all letters were converted to uppercase, and any character not belonging to the canonical nucleotide set [A,C,T,G] was replaced with `'N'`. Reads were tokenized using the same vocabulary employed during training. For each batch, a per-read score was computed using the cross-entropy loss function, consistent with the training objective.

To improve classification robustness in paired-end sequencing data, the inference pipeline processes both R1 and R2 reads independently, computing loss and classification scores for each. A separate script can then be used to compute the mean of the two as the final score for the read pair. If only one of the R1 or R2 reads is 100 bp or longer, we use its score directly without averaging, to maintain reliability. The data are processed in batches (default batch size = 1024 reads) for efficient computation within the model. For scalability, particularly when handling large input files, the data are divided into larger chunks, each containing 50 times the batch size by default. This chunking strategy ensures efficient CPU memory usage and enables parallel processing.

Hardware and software

All model training procedures were carried out on an NVIDIA A100 GPU with 80 GB of VRAM, using a batch size selected to maximize efficient resource utilization.

For inference and benchmarking, models were converted from 32-bit (FP32) to 16-bit floating-point precision (FP16) to achieve faster scoring, with a minimal trade-off in accuracy. On our evaluation datasets (11–20), the accuracy loss due to the transition from FP32 to FP16 was less than 0.1%.

All benchmarks were conducted across different hardware configurations to evaluate performance under diverse computational environments: a high-performance computing cluster node equipped with an NVIDIA A100 GPU (80 GB VRAM), optionally utilizing a multiprocessing DataLoader; cluster nodes with NVIDIA L40 (48GB VRAM) and NVIDIA L4 (24GB VRAM); a local workstation featuring an NVIDIA RTX 3090 GPU (24 GB VRAM) and 64 GB of RAM; and a cloud-based Google Colab instance running an NVIDIA Tesla T4 GPU (16 GB VRAM). This range of setups allowed us to assess the model’s efficiency from powerful cluster nodes to accessible cloud resources.

Throughput Benchmark

To assess the computational performance of BBERT, we conducted a comparative benchmark against existing approaches, BERTax[13] and Tiara[31]. The evaluation was performed under optimized conditions for each model, using the maximum batch size that fully utilized the available GPU memory. The benchmarks are presented in SI Table S3.

2. Synthetic data: using BBERT for important downstream tasks

2.a. Reading frame classification

Single case visual analysis - Low dimensional representation of the BBERT embedding space

Synthetic 150 bp reads were generated using a modified version of InSilicoSeq for two organisms: *Pseudomonas aeruginosa* PAO1 (RefSeq ID: GCF_000006765.1) and *Saccharomyces cerevisiae* S288C (RefSeq ID: GCF_000146045.2). The simulator appended to each read's description the start and end coordinates, as well as the percentage of coding sequence (CDS) nucleotides, computed using the corresponding .gtf file from NCBI.

BBERT was then run on the synthetic dataset to produce embeddings. Dimensionality reduction was performed using PCA and t-SNE (with t-SNE parameters: perplexity = 30, n_iter = 1000, random_state = 42), and the results were visualized using scatter plots. For PCA, an additional distribution plot was included.

Robust linear classification of sequence properties

To classify multiple properties of DNA reads, we employed three simple linear classifiers on top of the BBERT embeddings. Specifically, three separate classification heads were used: (i) bacterial versus non-bacterial reads, (ii) reading frame prediction (six classes), and (iii) coding versus non-coding regions. Each head consisted of a single fully connected layer, allowing us to evaluate whether a simple classifier on the BBERT embeddings performed sufficiently.

1. For bacterial/non-bacterial classification, we constructed the training dataset by combining the first 10 mixed calibration sets (Datasets 1–10), while evaluation was performed on the remaining 10 mixed subsets (Datasets 11–20), ensuring no overlap in genus composition between training and test sets. We trained a binary classifier on top of frozen embeddings extracted from a pre-trained BBERT. Input reads of length 100 bp were processed with the BERT encoder to obtain contextual representations from the final hidden layer. These embeddings were then passed to a lightweight classifier network that distinguished the bacterial from non-bacterial sequences. The BERT parameters remained fixed during training, ensuring that only the classifier was optimized.

2. The training dataset for frame classification was constructed using the same accession numbers from the NCBI database as those used for training the BBERT model; however, only

coding regions were included. From each coding region, a fragment of around 300 base pairs was randomly extracted, with minor variation in length introduced by sampling from a normal distribution with a mean of 300 bp and standard deviation of 5 bp. Reads of 100 bp length were then taken from the beginning and end of this fragment. To simulate sequencing errors, artificial mutations, insertions, and deletions were introduced using the error matrices from InSilicoSeq[40]. This procedure resulted in a dataset comprising 10M reads, which was used to train the frame classification model. The test dataset was generated in the same way, but based exclusively on accession numbers reserved for BBERT evaluation dataset (see Datasets).

The model achieved an accuracy of approximately 95% on the evaluation datasets. The confusion matrix (Fig. 2d) showed that of the few misclassifications, most occurred between forward and reverse orientations, while errors between adjacent frames were rare. These results suggest that the BERT-based embeddings capture both the direction and frame position of DNA reads with high fidelity.

3. For the task of distinguishing coding from non-coding sequences, we used the same synthetic dataset described below, which was generated with the modified InSilicoSeq pipeline and includes coding percentage information in the read annotations. As in the bacterial vs. non-bacterial classification task, input reads of length 100bp were first encoded with the frozen BBERT model to obtain contextual embeddings. These embeddings were then passed to a lightweight classifier trained to assign class 1 if more than 50% of the sequence was coding and class 0 otherwise.

2.b. Reduced assembly errors and run-time acceleration

To compare de-novo assembly of metagenomes before and after BBERT filtering, a synthetic dataset was generated using a modified version of InSilicoSeq[40], which was adapted to add to the read annotation the details: start, end, chromosome, organism source, % coding using ncbi's *gtf*. Four datasets were created by composing 128 non-overlapping prokaryotic and 128 eukaryotic genomes each. Genomes were downloaded from NCBI RefSeq, and had assembly-levels of either "chromosome" or "complete genome" and were unique at the genus level. Paired-end FASTQ files (R1 and R2) were simulated using the modified InSilicoSeq. Each organism contributed 35,000 paired-end reads. The resulting FASTQ files were mixed into combined R1 and R2 files per dataset, maintaining consistent order, to simulate a realistic metagenomic dataset. The so-called "mixed" data contain both eukaryotic and bacterial reads in equal abundance.

The BBERT model was then applied to the mixed data, generating classification scores. Based on these scores and the recommended predefined cut-off, reads were classified as bacterial or non-bacterial. We used metaSPAdes[39] to perform genome assembly on the full mixed dataset as well as on each of the classified subsets. Assembly run times were recorded, and the resulting contigs were saved in FASTA format.

To compare the assembly results, a BLAST nucleotide database was created from the 384

source genomes using the command *makeblastdb -dbtype nucl*. Each contig was aligned against this database using *blastn* with the following parameters: -outfmt 6 -max_target_seqs 10 -evalue 1e-5. Alignment errors were categorized into three types based on their position and frequency: single errors, isolated mismatches or gaps flanked by correctly aligned nucleotides; adjacent pairs, two consecutive mismatches or gaps between correctly aligned nucleotides; and longer runs, more than two consecutive mismatches or gaps.

For each organism, assembly coverage was calculated for both the mixed and classified assemblies using the formula:
(number of correctly aligned nucleotides in contigs - number of errors [mismatches or gaps]) / genome length × 100.

3. Soil metagenomic analyses

Metagenome dataset - To find relevant shotgun sequenced metagenomes we used the Integrated Microbial Genomes (IMG) system website. We searched for the term “soil” to get all records with this term in their descriptions. We then looked through each record to check for relevant SRA run accessions, and to verify that it was not an amplicon sequencing run. The SRA accessions were then used to acquire their respective metadata from the NCBI website. The SRA accessions for the 1971 soil metagenomes, as well as relevant metadata, can be found in SI Table S1.

Geographic information systems (GIS) - Metagenomes that had longitude and latitude coordinates available in their metadata were imported into ArcGIS Pro.

Kraken2
We used Kraken2 with the following reference databases: (i) Kraken2’s expanded database with protozoa, fungi and plants (PlusPFP version September 2024) in stringent settings (confidence 0.1) as well as less stringent conditions (confidence 0). (ii) A curated Kraken2 database: We curated 213,194 bacterial genomes downloaded from the NCBI RefSeq database. Out of all available bacterial genomes, we removed known human pathogens which represent the bulk of the database, but are not relevant for our purposes. In addition, we acquired 9,143 high quality prokaryote genomes from the GEMs database for a total of 222,337 genomes (SI Table S2). Reads were mapped to this database in stringent settings (confidence 0.1) as well as less stringent conditions (confidence 0), similarly to the PlusPFP database.

Read mapping and KEGG functional annotation

Reads were quality trimmed and mapped to the KEGG prokaryote database using DIAMOND[18] blastx with parameters --sensitive --max-target-seqs 1 --evalue 1e-5 --outfmt 6. We then assigned a KO number to read hits if available in the KEGG database for that reference hit (some reference proteins exist in the KEGG database without a corresponding KO number). We then summed up the coverage of all hits per KO in each metagenome and divided by the protein length to get an approximate number of copies of the gene in that sample.

MDS map of metagenomes before and after BBERT

To compare the taxonomic profile of the metagenomic samples before and after filtering by BBERT, we first used SingleM to get the taxonomic profile of each individual sample. We then combined them into one feature table that included the taxonomic profile of each sample before and after BBERT and used *metaMDS* and *vegdist* functions from the *vegan* R package to calculate MDS scores, and *ggplot* to create the ordination plot.

KEGG pathway annotation and significant differences between tropical, temperate, and arctic-boreal environments

To test the effects of BBERT filtering on the functional profile of metagenomic samples we used the soil metagenome dataset used in Bahram et al.[41]. We downloaded the metagenomes, quality-filtered them, and mapped to the KEGG prokaryote database as described above. We then used BBERT to separate the bacterial fraction of the reads to create two tables, one for prokaryotic and one for all reads each containing one row per KO identifier and one column per metagenomic run (e.g. *ERR1877650*). Cell values were then normalised as KO coverage in metagenome divided by total or bacterial reads respectively. Mapping of the KOs to pathways was performed using the KO to pathway lost file, downloaded from the KEGG website. Because about 20% of KOs are annotated to 2 or more pathways, raw summation inflates those pathways. We therefore computed a splitting weight as $\omega_i = \frac{1}{N_i}$, where N_i is the number of distinct map pathways containing KO i. The weighted abundance for KO *i* in sample *s* was $A_{is}^{split} = A_{is}^{norm} \times \omega_i$. For every sample *s* and pathways *p* we summed the corrected KO abundances: $P_{ps} = \sum_{i \in p} A_{is}^{split}$. For the 30 most abundant pathways (ranked by mean abundance in the total reads fraction) we tested for biome effects: Kruskal-Wallis test on raw (linear-scaled) abundances and for significant pathways, pairwise Dunn tests (Holm correction).

Data availability

BBERT is available to download from www.github.com/AmirErez/BBERT
The output from the BBERT runs on the 1971 soil metagenomes takes up approximately 3Tb when compressed. It is hosted on our server and is available upon reasonable request.

Author information and contributions

All authors analyzed results and wrote the manuscript. AE, OMF, and TS conceived the idea. DA with TS wrote the BBERT code. MA obtained and curated the metagenomic dataset, as well as the training genome set, and performed the KEGG analyses. AE and OF supervised the research.

Ethics declaration

The authors declare no competing interests.

Bibliography

1. Fierer, N., Wood, S. A. & Bueno de Mesquita, C. P. How microbes can, and cannot, be used to assess soil health. *Soil Biology and Biochemistry* **153**, 108111 (2021).
2. Williamson, K. E., Fuhrmann, J. J., Wommack, K. E. & Radosevich, M. Viruses in Soil Ecosystems: An Unknown Quantity Within an Unexplored Territory. *Annual Review of Virology* **4**, 201–219 (2017).
3. Bailly, J. *et al.* Soil eukaryotic functional diversity, a metatranscriptomic approach. *ISME J* **1**, 632–642 (2007).
4. Aslani, F., Geisen, S., Ning, D., Tedersoo, L. & Bahram, M. Towards revealing the global diversity and community assembly of soil eukaryotes. *Ecology Letters* **25**, 65–76 (2022).
5. Bates, S. T. *et al.* Examining the global distribution of dominant archaeal populations in soil. *The ISME Journal* **5**, 908–917 (2011).
6. Rinke, C. *et al.* Insights into the phylogeny and coding potential of microbial dark matter. *Nature* **499**, 431–437 (2013).
7. Marcy, Y. *et al.* Dissecting biological “dark matter” with single-cell genetic analysis of rare and uncultivated TM7 microbes from the human mouth. *Proceedings of the National Academy of Sciences* **104**, 11889–11894 (2007).
8. Fierer, N. Embracing the unknown: disentangling the complexities of the soil microbiome. *Nat Rev Microbiol* **15**, 579–590 (2017).
9. Graham, E. B. *et al.* A global atlas of soil viruses reveals unexplored biodiversity and potential biogeochemical impacts. *Nat Microbiol* **9**, 1873–1883 (2024).
10. Piton, G. *et al.* Life history strategies of soil bacterial communities across global terrestrial biomes. *Nat Microbiol* **8**, 2093–2102 (2023).
11. Osburn, E. D., McBride, S. G. & Strickland, M. S. Microbial dark matter could add uncertainties to metagenomic trait estimations. *Nat Microbiol* **9**, 1427–1430 (2024).

12. Piton, G. *et al.* Reply to: Microbial dark matter could add uncertainties to metagenomic trait estimations. *Nat Microbiol* **9**, 1431–1433 (2024).
13. Mock, F., Kretschmer, F., Kriese, A., Böcker, S. & Marz, M. Taxonomic classification of DNA sequences beyond sequence similarity using deep neural networks. *Proceedings of the National Academy of Sciences* **119**, e2122636119 (2022).
14. Diamond, S. *et al.* Mediterranean grassland soil C–N compound turnover is dependent on rainfall and depth, and is mediated by genomically divergent microorganisms. *Nat Microbiol* **4**, 1356–1367 (2019).
15. West, P. T., Probst, A. J., Grigoriev, I. V., Thomas, B. C. & Banfield, J. F. Genome-reconstruction for eukaryotes from complex natural microbial communities. *Genome Res.* **28**, 569–580 (2018).
16. Zhao, D. *et al.* Eukfinder: a pipeline to retrieve microbial eukaryote genome sequences from metagenomic data. *mBio* **16**, e00699-25 (2025).
17. Nayfach, S. *et al.* A genomic catalog of Earth's microbiomes. *Nat Biotechnol* **39**, 499–509 (2021).
18. Buchfink, B., Xie, C. & Huson, D. H. Fast and sensitive protein alignment using DIAMOND. *Nature Methods* **12**, 59–60 (2015).
19. Kanehisa, M. & Goto, S. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research* **28**, 27–30 (2000).
20. NCBI Resource Coordinators. Database resources of the National Center for Biotechnology Information. *Nucleic Acids Research* **42**, D7–D17 (2014).
21. Wood, D. E., Lu, J. & Langmead, B. Improved metagenomic analysis with Kraken 2. *Genome Biology* **20**, 257 (2019).
22. Blanco-Míguez, A. *et al.* Extending and improving metagenomic taxonomic profiling with uncharacterized species using MetaPhlAn 4. *Nat Biotechnol* **41**, 1633–1644 (2023).

23. Woodcroft, B. J. *et al.* SingleM and Sandpiper: Robust microbial taxonomic profiles from metagenomic data. 2024.01.30.578060 Preprint at https://doi.org/10.1101/2024.01.30.578060 (2024).
24. Wood, D. E., Lu, J. & Langmead, B. Improved metagenomic analysis with Kraken 2. *Genome Biol* **20**, 257 (2019).
25. Woodcroft, B. J. *et al.* SingleM and Sandpiper: Robust microbial taxonomic profiles from metagenomic data. 2024.01.30.578060 Preprint at https://doi.org/10.1101/2024.01.30.578060 (2024).
26. Woodcroft, B. J. *et al.* Comprehensive taxonomic identification of microbial species in metagenomic data using SingleM and Sandpiper. *Nat Biotechnol* 1–6 (2025) doi:10.1038/s41587-025-02738-1.
27. Zhou, Z. *et al.* DNABERT-2: Efficient Foundation Model and Benchmark For Multi-Species Genome. Preprint at http://arxiv.org/abs/2306.15006 (2024).
28. Zhou, Z. *et al.* DNABERT-S: Pioneering Species Differentiation with Species-Aware DNA Embeddings. Preprint at https://doi.org/10.48550/arXiv.2402.08777 (2024).
29. Brixi, G. *et al.* Genome modeling and design across all domains of life with Evo 2. 2025.02.18.638918 Preprint at https://doi.org/10.1101/2025.02.18.638918 (2025).
30. Hou, S. *et al.* DeepMicroClass sorts metagenomic contigs into prokaryotes, eukaryotes and viruses. *NAR Genom Bioinform* **6**, lqae044 (2024).
31. Karlicki, M., Antonowicz, S. & Karnkowska, A. Tiara: deep learning-based classification system for eukaryotic sequences. *Bioinformatics* **38**, 344–350 (2022).
32. Contributions of the Petabyte Scale Sequence Search Codeathon toward efforts to scale sequence-based searches on SRA. https://arxiv.org/html/2505.06395v1?utm_source=chatgpt.com.

33. Zhang, A., Xiao, T. Z., Liu, W., Bamler, R. & Wischik, D. Your Finetuned Large Language Model is Already a Powerful Out-of-distribution Detector. Preprint at https://doi.org/10.48550/arXiv.2404.08679 (2024).
34. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Preprint at https://doi.org/10.48550/arXiv.1810.04805 (2019).
35. Vaswani, A. *et al.* Attention is All you Need. in *Advances in Neural Information Processing Systems* vol. 30 (Curran Associates, Inc., 2017).
36. Lewin, H. A. *et al.* Earth BioGenome Project: Sequencing life for the future of life. *Proceedings of the National Academy of Sciences* **115**, 4325–4333 (2018).
37. Maaten, L. van der & Hinton, G. Visualizing Data using t-SNE. *Journal of Machine Learning Research* **9**, 2579–2605 (2008).
38. Braham, M. *et al.* Structure and function of the global topsoil microbiome. *Nature* **560**, 233–237 (2018).
39. Nurk, S., Meleshko, D., Korobeynikov, A. & Pevzner, P. A. metaSPAdes: a new versatile metagenomic assembler. *Genome Res.* **27**, 824–834 (2017).
40. Gourlé, H., Karlsson-Lindsjö, O., Hayer, J. & Bongcam-Rudloff, E. Simulating Illumina metagenomic data with InSilicoSeq. *Bioinformatics* **35**, 521–522 (2019).
41. Bahram, M. *et al.* Structure and function of the global topsoil microbiome. *Nature* **560**, 233–237 (2018). University of Haifa. We thank Arnout Schepers and Tenaces Biosciences Ltd. for their support of T.S.'s independent scientific research outside of his primary professional responsibilities.

Supplementary Information

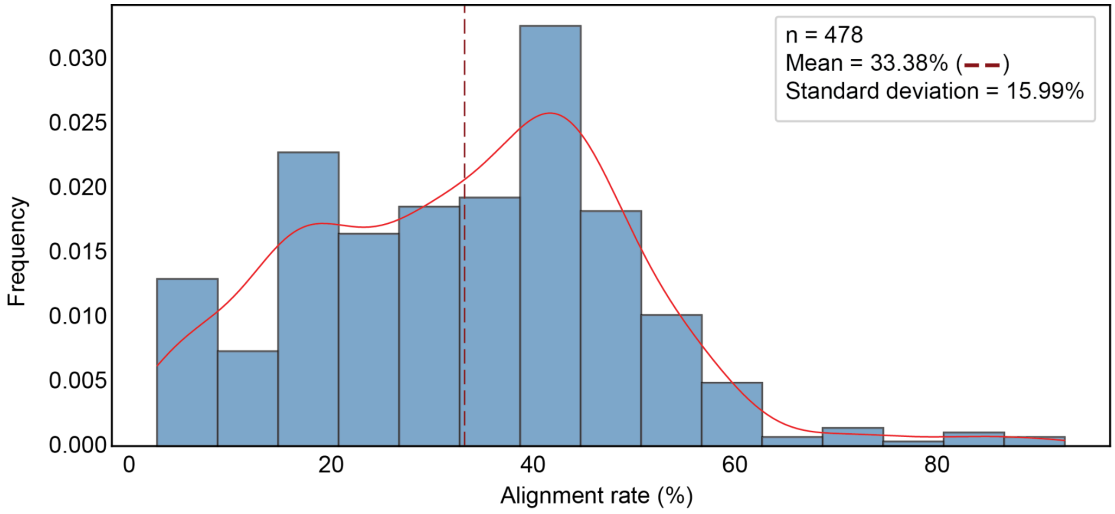

Fig. S1: **Percent of reads mapped to assemblies of 478 soil metagenomes.** The mean alignment rate is 33.38%, meaning that 66.62% of the reads do not align to any contig.

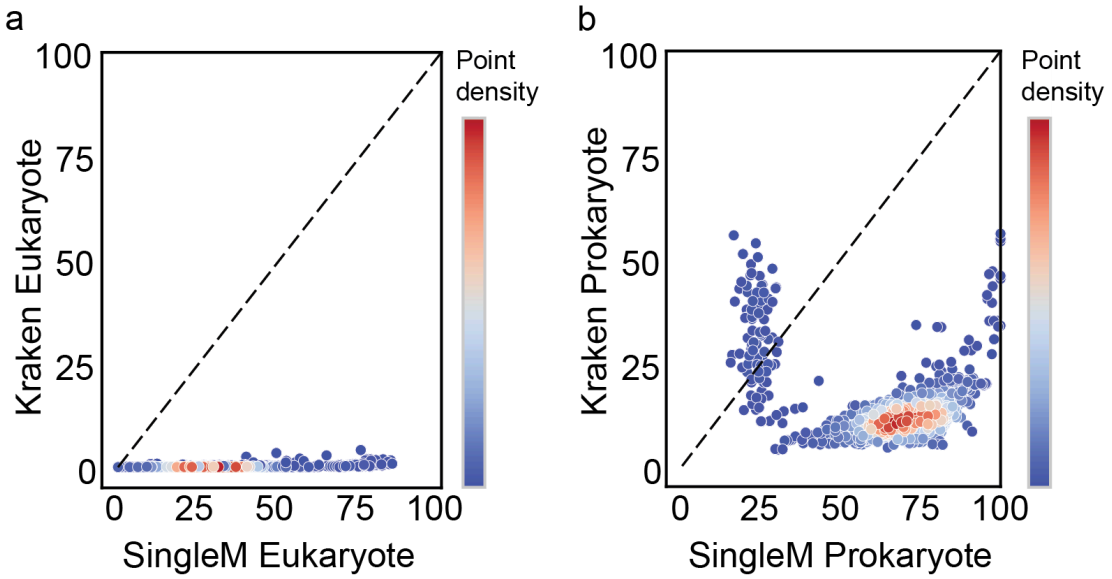

Fig. S2 Further evidence for conflicting eukaryote and prokaryote fractions between SingleM and Kraken2's default database. The imputed domain percent of each sample, calculated using SingleM on the *x*-axis and Kraken2 using its default database plus fungi, protozoa, and yeast (PlusPFP, September 2024). **a,** Imputed eukaryote fraction, where only positively identified eukaryotes are considered for Kraken2. **b,** The imputed prokaryote percent of each sample.

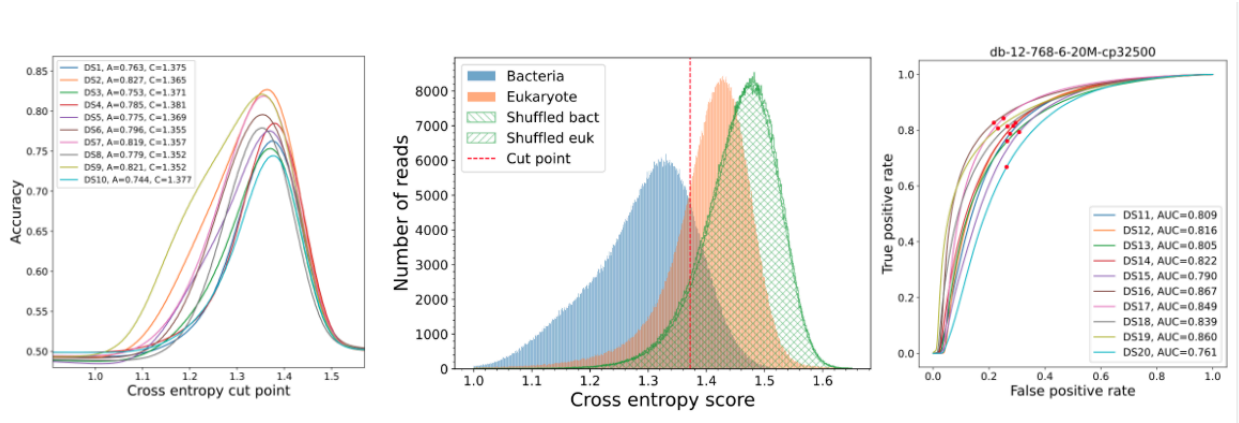

Fig. S3. BBERT Can use the cross-entropy loss to classify bacterial reads consistently well on data it was not trained on. a, Accuracy of classification on 10 validation sets shows that the optimal cut-point for separating bacterial and nonbacterial reads is independent of the specific taxa used, therefore a universal cutpoint can be used on unknown data. The expected accuracy when using the cross-entropy loss cutpoint to classify, is in the range of 70%-80%. We obtain higher accuracy by using a classifier on the embeddings (see main text). **b,** The distribution of loss function scores (cross entropy) for bacterial and eukaryotic reads shows two clear peaks for the two domains, well separated by the universal cutpoint (vertical dashed red). We also shuffled the nucleotide order in the reads (green) which produced an even higher score, agnostic to their eukaryote or prokaryote origin. **c,** ROC curves for the classification of the 10 test datasets, for a range of cutoffs. Red dots represent the universal cutpoint, showing remarkable agreement between the 10 non-overlapping test sets.

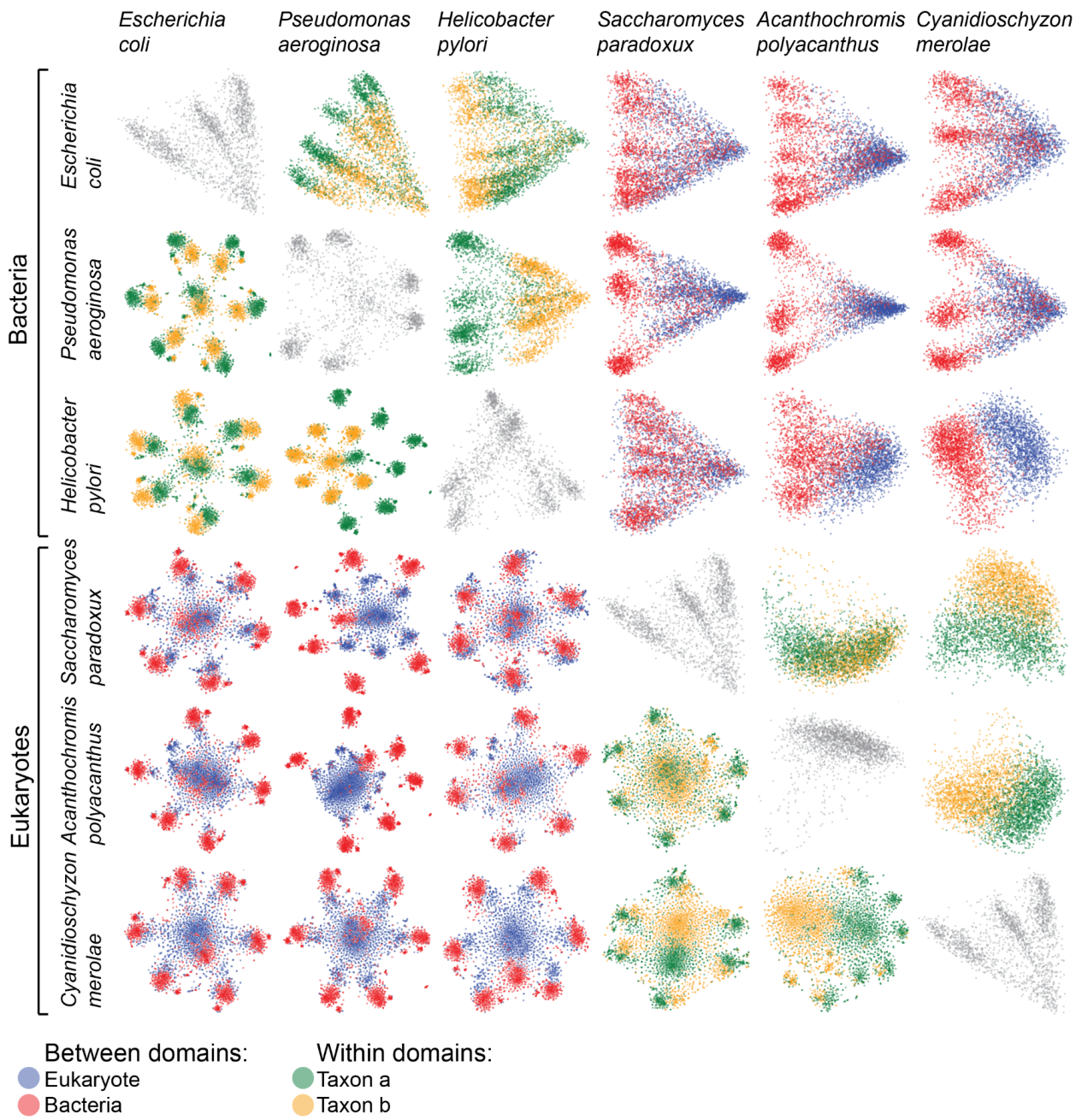

Fig. S4. Generalization of BBERT embedding structure across taxa within and between domains. Grid of pairwise taxon comparisons derived from per-read BBERT embeddings. Off-diagonal panels show joint embeddings of the two indicated taxa: the lower triangle uses t-SNE, and the upper triangle shows PCA (PC1 vs. PC2). Diagonal panels display single-taxon PCA to highlight within-taxon substructure. Across pairs, both methods robustly separate taxa, including across domains, while single-taxon views reveal finer organization consistent with content-based features (Fig. 2).

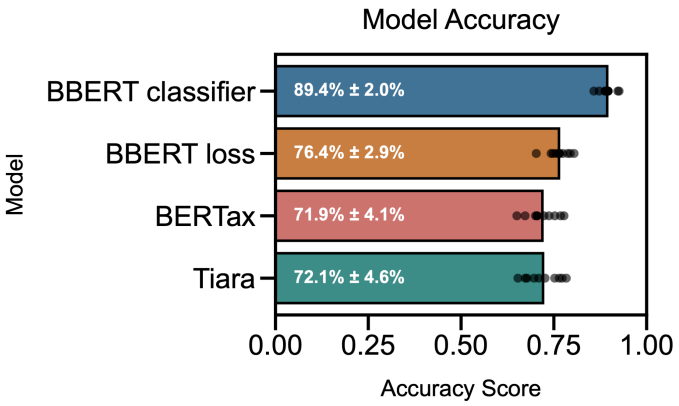

Fig. S5. Comparison of classification accuracy. We ran the bacterial/non-bacterial classification using BBERT on the embeddings, BBERT's loss cutoff, BERTax, and Tiara. The BBERT embeddings classifier was significantly the most accurate. All runs were on the same test dataset (SI Table S6).

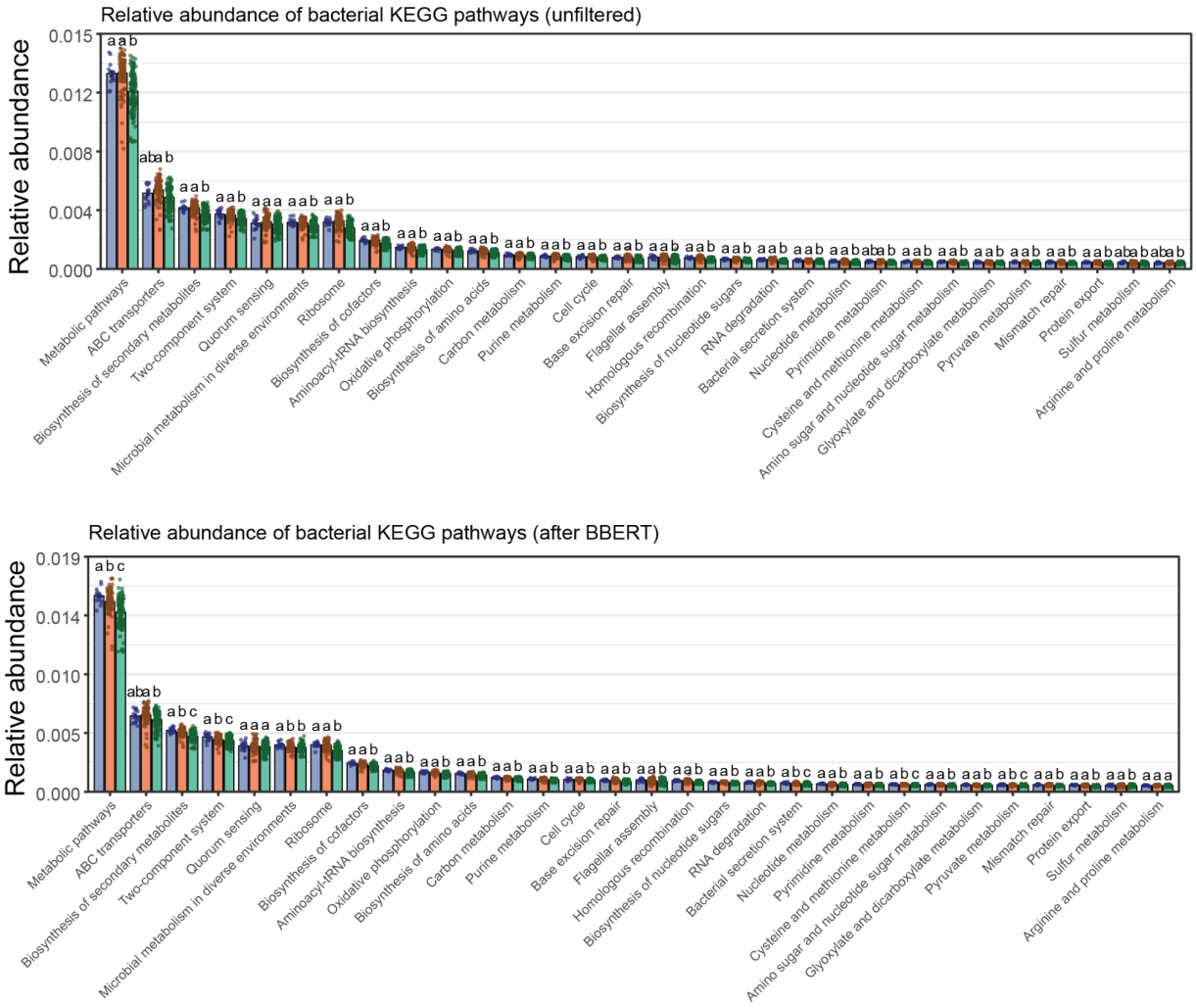

Fig. S6. Normalized KEGG pathway prevalence before and after BBERT filtering. Normalized KEGG pathway abundances across three biomes, reanalyzed from Barham et al[41]. **a,** KEGG abundances before filtering. **b,** KEGG abundances after filtering. Statistical significance for each pathway is shown using the compact letter display.

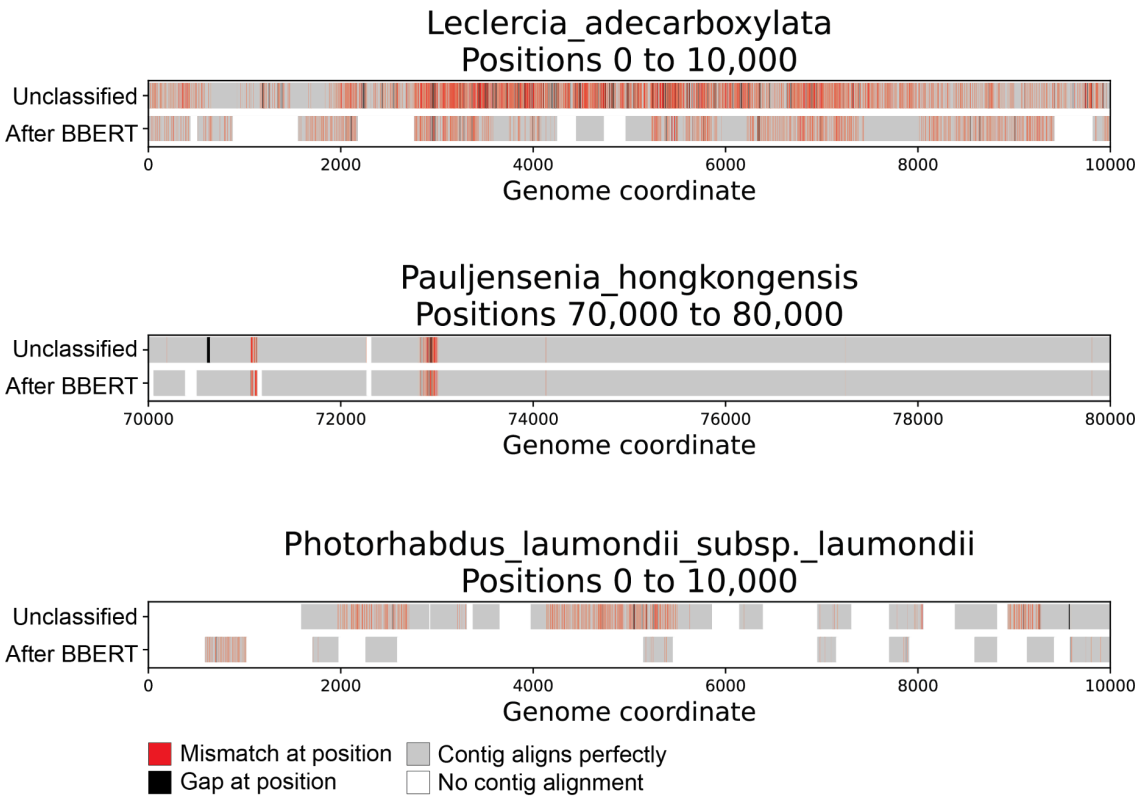

Fig. S7. Comparison of de-novo assembly errors before and after BBERT filtering.
Reads were assembled with metaSPAdes[39] using either unfiltered input or BBERT-filtered (bacterial) reads. Resulting contigs were aligned to the known bacterial source genomes used to generate the mixtures (see Methods), enabling per-base annotation of mismatches and gaps and identification of mis-assemblies. Shown are representative loci illustrating some outcomes: unfiltered data produces chimeric joins or dense local errors, whereas BBERT-filtered data assembles the region correctly or refrains from assembling (avoiding false contigs); in a minority of cases, the unfiltered assembly is correct while filtering prevents assembly, consistent with reduced coverage after removal of non-bacterial reads; some regions remain challenging under both conditions. Aggregate improvements in assembly quality and runtime are quantified in Fig. 5.