MADRe: Strain-Level Metagenomic Classification Through Assembly-Driven Database Reduction

Josipa Lipovac[1], Mile Šikić[1,2], Riccardo Vicedomini[3], and Krešimir Križanović[1]

[1]Laboratory for Bioinformatics and Computational Biology, Faculty of Electrical Engineering and Computing, University of Zagreb, Zagreb, Croatia
[2]Laboratory of AI in Genomics, Genome Institute of Singapore, A*STAR, Singapore, Singapore
[3]Univ Rennes, CNRS, Inria, IRISA - UMR 6074, F-35000 Rennes, France

Abstract

Strain-level metagenomic classification is essential for understanding microbial diversity and functional potential, but remains challenging, particularly in the absence of prior knowledge about the composition of the sample. In this paper we present MADRe, a modular and scalable pipeline for long-read strain-level metagenomic classification, enhanced with **M**etagenome **A**ssembly-Driven **D**atabase **Re**duction. MADRe combines long-read metagenome assembly, contig-to-reference mapping reassignment based on an expectation-maximization algorithm for database reduction, and probabilistic read mapping reassignment to achieve sensitive and precise classification. We extensively evaluated MADRe on simulated datasets, mock communities, and a real anaerobic digester sludge metagenome, demonstrating that it consistently outperforms existing tools by achieving higher precision with reduced false positives. MADRe's design allows users to apply either the database reduction or read classification step individually. Using only the read classification step shows results on par with other tested tools. MADRe is open source and publicly available at https://github.com/lbcb-sci/MADRe.

Keywords: metagenomics, strain-level, metagenomic classification, database reduction

1 Background

Metagenomics enables the study of genetic material from complex microbial communities found in environments such as human gut, soil, or marine ecosys-

tems. It provides a comprehensive view of microbial diversity and interactions within these environments [1, 2]. A central challenge in metagenomic analysis is the accurate identification of organisms present in a sample, typically performed by comparing sequencing reads to reference genome databases [3].

A wide range of metagenomic classification tools have been developed, which can be broadly categorized into marker-based, DNA-to-protein and DNA-to-DNA approaches, as described in [4]. Marker-based tools, such as MetaPhlAn [5, 6], StrainPhlAn, mOTUs [7], and Melon [8], classify taxa using conserved, clade-specific marker genes. DNA-to-protein tools, including Kaiju [9], DIAMOND [10], MMseqs2 [11], and MEGAN-LR [12] translate reads into amino acid sequences before aligning them to protein databases. DNA-to-DNA tools compare reads directly against genomic sequences and are commonly divided into k-mer-based and mapping-based tools [13]. K-mer-based tools such as Kraken2 [14], KrakenUniq [15], Bracken [16], Centrifuge [17], CLARK/CLARK-S [18, 19], Ganon [20, 21], Taxor [22], and Sylph [23] are known for their speed and scalability to large databases, but often trade precision for speed. In contrast, mapping-based tools such as MetaMaps [24], PathoScope2 [25], EMU [26] and MORA [27], which rely on read alignments and reassignment algorithms, offer higher precision at a greater computational cost.

Although k-mer-based tools, especially Kraken2, perform well at the species level, strain-level classification becomes increasingly challenging when sequences originate from closely related genomes [28]. However, resolving strain-level diversity is essential, as even closely related strains can exhibit substantial differences in gene content and function, with implications for microbial ecology, pathogenesis, and treatment outcomes [29, 30, 31, 32, 33].

While most existing tools are optimized for short reads due to their low cost and high accuracy, long-read sequencing technologies such as Oxford Nanopore and PacBio HiFi are rapidly improving. Longer read lengths provide advantages for genome assembly, structural-variant detection, and improved strain-level resolution.

Several short-read-based tools are designed for strain-level classification within a single species, such as StrainGE [34], StrainEST [35], and StrainSeeker [36], as well as the long-read-based ORI [37]. Other tools, including PanTax [4], MetaMaps [24], Centrifuge [17], PathoScope2 [25], and MORA [27], are suitable for more complex, multi-species datasets and support short- and long-read strain-level metagenomic classification. PanTax is a pangenome-based approach that, while supporting multi-species datasets, faces scalability limitations when applied to very large reference databases. MetaMaps is a mapping-based tool capable of high-resolution classification but is known to be extremely computationally demanding [13]. Centrifuge is a k-mer-based tool designed to perform strain-level classification, but like PanTax, it encounters limitations when constructing indexes for very large reference databases. PathoScope2 is an older tool that is no longer maintained and cannot be reliably executed due to outdated dependencies and software incompatibilities. Originally developed for strain-level classification of short reads, it is based on an expectation-maximization (EM) algorithm for read reassignment [38]. As part of the MORA study, the authors

introduced a continuation of PathoScope2, referred to as AugPatho, which includes a modified version of the original algorithm adapted for use with long reads [27]. MORA extends this approach by combining the EM algorithm from Agamemnon [39] with a read reassignment strategy based on the Weapon-Target Assignment (WTA) problem. According to its authors, MORA represents the current state-of-the-art in mapping-based long-read metagenomic classification.

Reference databases often contain multiple assemblies of the same strain and typically lack consistent organization. To address this, some strain-level classification tools perform database pre-clustering according to Average Nucleotide Identity (ANI) scores. Previous studies have shown that there is no universal ANI threshold for defining strains [34, 35, 40, 41, 29]. Setting the threshold too high may erroneously separate assemblies of the same strain, whereas setting a threshold too low may incorrectly group different strains together.

Using large and diverse databases is important for accurate strain-level classification [42], but it also makes the analysis much more demanding to run. MetaAlign [43] uses containment MinHash [44] to reduce the reference database prior to alignment, improving runtime while maintaining high species-level precision. However, it is primarily designed for short reads, and strain-level resolution is not its main focus.

In this work, we introduce MADRe, a pipeline for long-read, strain-level metagenomic classification enhanced with Metagenome Assembly-Driven Database Reduction, consisting of two main phases: database reduction and read classification. In the database reduction step, MADRe combines long-read assembly with an EM algorithm that assigns assembled contigs to one or more references, reducing the reference database.

In the second, read classification step, MADRe performs mappings-based read reassignment. It resolves ambiguous read mappings by assigning each read to the most likely reference, based on mapping scores and probabilistic support.

We conducted an extensive evaluation of MADRe using simulated datasets, Zymo mock communities, and a real anaerobic digester sludge metagenome. The results demonstrate that MADRe achieves high precision and strain-level resolution while maintaining lower memory usage and runtime compared to existing tools. Additionally, the two steps of the MADRe pipeline can be run independently, and our results show that the read classification module (MADRe_RC) alone performs competitively. MADRe's approach enables the use of large, diverse reference databases spanning multiple taxonomic levels, making it well-suited for scenarios where no prior knowledge about the sample is available. Using assembled contigs to detect potentially present strains, MADRe focuses on confidently represented organisms. As a result, compared to state-of-the-art tools, it significantly reduces the number of false positives while maintaining high resolution strain-level classification.

2 Results

2.1 MADRe - method overview

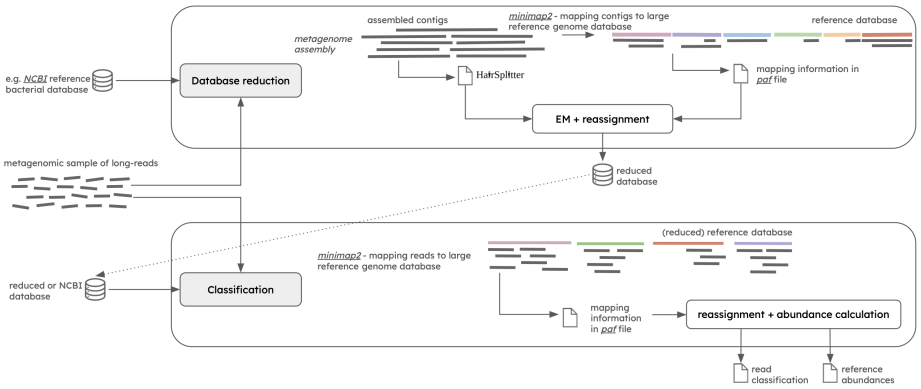

Figure 1: **MADRe overall pipeline** The first step of the pipeline performs database reduction using an EM-based contig-to-reference mapping procedure to identify organisms present in the sample. The second step involves read classification, which applies probabilistic read reassignment based on mapping information.

The MADRe pipeline is designed for strain-level metagenomic classification, particularly in scenarios where no prior knowledge of the sample composition is available. Its primary goal is to enable accurate strain identification while reliably distinguishing truly abundant strains from false positives.

As illustrated in Figure 1, the MADRe pipeline consists of two main steps: database reduction and read classification. It takes as input a large bacterial reference database and raw long metagenomic reads, and produces two main outputs: a read classification file and a reference abundance file.

In the database reduction step, long reads are first assembled, and the resulting contigs are mapped to the large reference database. MADRe employs HairSplitter [45] to estimate the number of collapsed strains represented by each contig. Using both the contig-to-reference mappings and the collapsed strain estimates, an EM algorithm is applied, followed by additional read reassignment steps. This process generates a reduced reference database that highlights strains likely present in the sample. Although this reduction effectively narrows the search space, it may still include false positives introduced during the assembly process. These are further filtered during the read classification step.

In the second step, reads are mapped to the reduced reference database, and ambiguous mappings are resolved through a reassignment procedure that leverages mapping scores and probabilistic support.

Both steps of the pipeline can be run independently so when prior knowledge about the sample exists, or when a reduced reference set is already available, the read classification step can be used on its own.

2.2 Benchmarking details

We benchmarked the MADRe pipeline against state-of-the-art tools developed for the same purpose: handling large reference databases while enabling strain-level classification. These tools include MORA [27] and AugPatho (PathoScope2) [25, 46]. We also evaluated Kraken2 [14], one of the most widely used metagenomic classification tools, which is often considered the standard for species-level classification. Although Kraken2 is capable of assigning reads at the strain level, its evaluation is complicated by the use of taxonomic identifiers (taxIDs) that may refer to either species or strain ranks. Additionally, we included MADRe_RC, a variant of MADRe that performs only the second step (*i.e.*, read classification) without prior database reduction.

We did not include MetaMaps, PanTax, or Centrifuge in our benchmarking analysis. In the case of MetaMaps, previous studies have reported crashes when attempting to build an index for the full Genome Taxonomy Database (GTDB), highlighting its scalability limitations [22, 27]. Similarly, our attempts to construct the same reference database for PanTax and Centrifuge, used successfully with other benchmarking tools, also failed due to crashes during the indexing process.

All commands used to run the benchmarking tools are available in the Supplementary file (Tools versions and commands).

2.3 Datasets

As part of the benchmarking process we evaluated mentioned tools on simulated metagenome datasets, Zymo mock communities, and a real anareobic digester sludge metagenome.

For the simulated datasets, we selected a smaller subset of genomes representing species commonly found in the human gut microbiome. Reference genomes were required to be labeled as "complete" or "chromosome" in NCBI, and to have a strain-level taxID distinct from their species-level taxID. This criterion ensured that Kraken2 could be included in the evaluation.

We simulated three different metagenomic datasets:

1. **sim_small (4 strains)** – This dataset includes four different strain references: two strains from *Adlercreutzia equolifaciens* and two from *Streptococcus anginosus*, with varying relative abundances.

2. **sim_medium (15 strains)** – This dataset contains 15 strain references distributed across five bacterial species (*i.e.*, *Helicobacter pylori*, *Cutibacterium acnes*, *Streptococcus intermedius*, *Streptococcus mutans*, *Lactococcus lactis*), with each species represented by three strains. At species level the abundances are different while strains of one species are equally abundant.

3. **sim_expanded (30 strains)** – An extension of the sim_medium dataset, incorporating 15 additional strains from distinct species and maintaining

variable abundance levels across species. Newly added species are listed in Supplementary Table 4.

Exact genome information including accession numbers, strain and species taxIDs, genome lengths, and number of simulated reads can be found in the Supplementary Tables (ST1-ST4).

We also tested MADRe using three Zymo mock communities: D6322 (ONT), D6331 (ONT) [47], and D6331 (PacBio HiFi) [48]. We included both D6311 ONT and D6311 PacBio HiFi datasets to demonstrate the pipeline's capability across different sequencing technologies.

To evaluate performance on a complex real-world dataset, we analyzed an anaerobic digester sludge metagenome dataset sequenced using ONT R10.4.1. reads [49]. This real metagenome represents the type of scenario for which MADRe is designed, where a highly diverse sample is analyzed without prior knowledge of its taxonomic composition.

2.4 Database

To thoroughly evaluate strain-level classification and ensure sufficient taxonomic divergence for accurate strain-level detection, we used a database obtained via the Kraken2 interface by selecting the bacterial database. This database consists of 102,639 sequences, encompassing all RefSeq [50] complete bacterial genomes. The database was downloaded in December 2024. The exact command used for downloading the database is provided in the Supplementary file (Tools versions and commands).

For consistency, the same database was used across all tools and experiments.

2.5 Experiment 1 - database reduction

To assess the effectiveness of strain identification and database reduction, we evaluated the output of MADRe's database reduction step on simulated datasets by comparing it to two baseline models. More precisely, assembled contigs were mapped to the large reference database and the identification of organisms was carried out using the following strategies:

- Baseline Model 1 (BM1): For the reduced database, each contig's top reference genome, determined by the highest summarized harmonic mean mapping value (Section 5, Equation 2), was included without performing any reassignment steps. BM1 represents the ideal reduction level under the assumption of a perfect assembly, where each contig corresponds to a single strain reference.

- Baseline Model 2 (BM2): For the reduced database, each contig's top three reference genomes, based on the highest summarized harmonic mean mapping values (Section 5, Equation 2), were included without performing any reassignment steps. BM2 defines an upper bound on the number of references expected in the reduced database. Since our simulated metagenomes

contain at most three strains per species, we assume that at most three strains could be collapsed into a single contig. This approach allows us to determine how many references should be retained in the reduced database to ensure that no true reference from the sample is missed.

The results, presented in Table 1, demonstrate that MADRe's database reduction achieves a high level of reduction while successfully retaining all expected strains. Additionally, when compared to BM2, MADRe reduces the number of false positive species, further highlighting its effectiveness.

Table 1: **Database reduction results.** Comparison of MADRe's performance (RDB) with two baseline models on simulated datasets: baseline model 1 (BM1), which includes only the top-1 mapping for each contig, and baseline model 2 (BM2), which includes the top-3 mappings.

	Metric	**sim_small**	**sim_medium**	**sim_expanded**
BM1	# in reduced	7	34	68
	# missing strains	0	1	3
	# FP strains	3	19	38
	# FP species	1	0	6
BM2	# in reduced	14	107	234
	# missing strains	0	0	0
	# FP strains	10	92	204
	# FP species	5	11	78
MADRe	# in reduced	7	42	84
	# missing strains	0	0	0
	# FP strains	3	27	54
	# FP species	1	0	6

2.6 Experiment 2 - classification of simulated reads

We used the Badread tool to simulate reads with known reference origins and compared the classification performance of MADRe, MADRe_RC, MORA, Aug-Patho, and Kraken2. Since the exact source of each read is known, we define classification outcomes as follows: true positive (TP) if the read is classified under the expected strain (or expected cluster), true negative (TN) if the read is not classified and its Badread label is *random* or *junk*, false positive (FP) if the read is classified under the incorrect strain (or incorrect cluster), and false negative (FN) if the read is not classified but its Badread label is different from *random* or *junk*.

We performed classification with and without post-clustering. The post-clustering method described in the Section 5, based on read mappings to reference genomes, groups closely related strains and assigns reads to clusters instead of specific strains. The same clustering approach was applied to all tools, utilizing read mappings to the full reference database. Kraken2's clustering results

are not included, as taxID alone does not allow for an accurate evaluation of post-clustering performance.

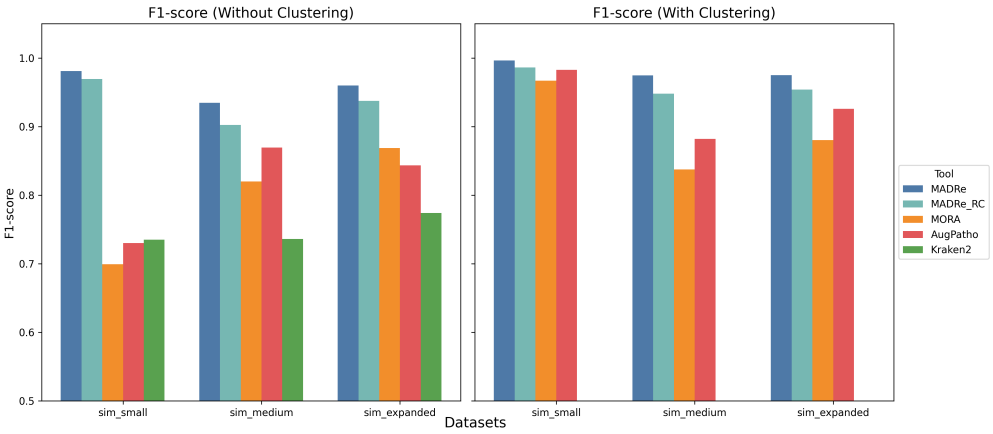

Figure 2: **Performance on simulated datasets.** F1 scores of strain-level classification on simulated reads, shown with and without post-clustering (grouping highly similar strains).

The classification results are presented in Figure 2, which shows the F1 scores for classification with and without post-clustering. The results demonstrate that both MADRe and MADRe_RC outperform all other approaches. Interestingly, on the sim_small dataset, Kraken2 achieves better results than AugPatho and MORA. In the sim_expanded dataset (without clustering), MORA outperforms AugPatho, but in all other cases, including all clustering scenarios, AugPatho performs significantly better than MORA. Post-clustering improves AugPatho's performance, bringing it close to MADRe_RC. Across all experiments, MADRe_RC performs nearly as well as MADRe. However, this can be drastically changed in case of more complicated datasets.

The exact counts of TPs, TNs, FPs, FNs, along with additional performance metrics such as accuracy, precision, and recall for both strain and species level are provided in Supplementary Table 5.

In addition, a Supplementary Figure S4 displays Bray–Curtis (BC) distances [51] between the observed read count abundances and the ground-truth abundances, offering further insight into the similarity between predicted and true community compositions. BC distance is one of the most commonly used distances to calculate microbial abundance differences more described in the Section 5. From these distances, it is evident that MADRe achieves the closest match to the ground truth, while AugPatho reports the lowest scores among the strain-level tools. Kraken2, although efficient, shows the poorest performance in terms of Bray–Curtis distance.

The number of identified organisms for each tool across all datasets is presented in Figure 3. Kraken2 is again excluded, as taxID alone does not allow for precise strain-level identification. In these datasets, all tools successfully identified all expected organisms (*i.e.*, all true positives), so a higher number

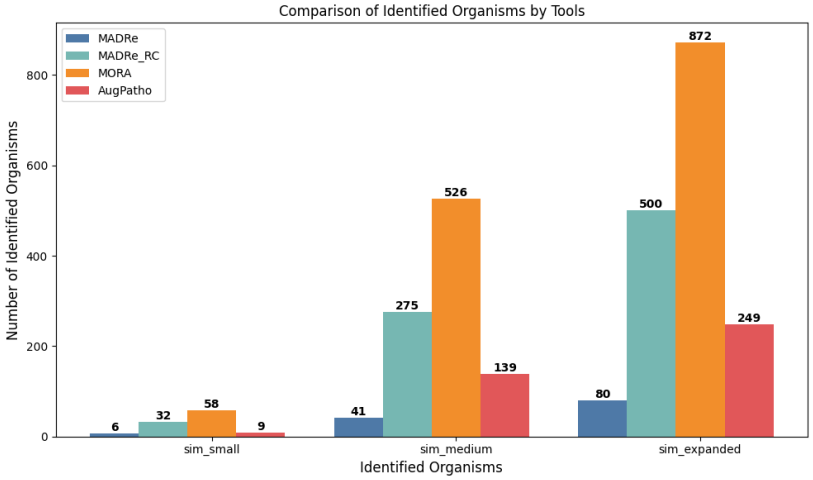

Figure 3: **Number of identified organisms by different tools on simulated reads.** An organism is considered identified if at least one read is classified under it. The sim_small dataset contains 4 strains, sim_medium contains 15 strains, and sim_expanded contains 30 strains. All tools successfully identified all expected strains, resulting in no false negatives.

of identified organisms indicates the presence of false positives. The results indicate that MORA reports the highest number of identified organisms, but this also results in the highest number of false positives. While MADRe_RC achieves high classification accuracy, it still identifies a considerable number of false positive strains, though fewer than MORA. In contrast, AugPatho identifies significantly fewer strains than MADRe_RC. However, the full MADRe pipeline demonstrates the strongest ability to reduce false positive identifications. These results, combined with the F1 classification scores, highlight the true power of MADRe, as it achieves high accuracy while minimizing false positive identifications.

2.7 Experiment 3 - classification of Zymo mock communities reads

To evaluate MADRe on real sequencing data, we conducted experiments on three different Zymo mock community datasets: ONT ZymoD6322, which consists of eight organisms (seven bacterial species and one fungus), and both the ONT and HiFi versions of ZymoD6331, which contain 21 organisms, including two fungi and five different strains of *Escherichia coli*. The primary challenge in the ZymoD6331 dataset is the ability to distinguish between these closely related *E. coli* strains. In this analysis, we excluded fungal genomes, focusing solely on bacterial classifications.

For Zymo mock communities, exact reference genomes of the strains present in the sample are available, along with their theoretical relative abundances provided by ZymoBIOMICS [52]. We supplemented our database with the Zymo

reference genomes, assigning them separate labels. However, we did not use provided theoretical abundances in our analysis, as they may deviate from the expected values due to variations in library preparation [8, 26]. Instead, we established ground-truth read classifications.

We mapped all reads to the expected bacterial reference genomes using Minimap2 and assigned true labels based on the best hit. These assignments were also used to determine the relative abundances. However, this process was not straightforward for the five *Escherichia coli* strains, as their high similarity led to ambiguous mappings. To address this, we leveraged our clustering method (explained in Section 5.2.2), which grouped these five strains into three clusters. Specifically, strains B766 and B3008 each formed separate clusters, while the remaining three strains were grouped into a single cluster, indicating that they were too similar to be reliably distinguished at the strain level. This clustering result aligns with previous findings from metagenome assembly procedures [53], where B766 and B3008 were successfully assembled, while the other three strains were not.

Using this information, we incorporated the clustering results into our ground-truth labeling: reads originating from the three indistinguishable strains were assigned the same label, ensuring a more accurate classification.

To evaluate performance, we calculated the BC distances (eq.9) between the observed read count abundances and the ground-truth abundances, both with and without post-clustering.

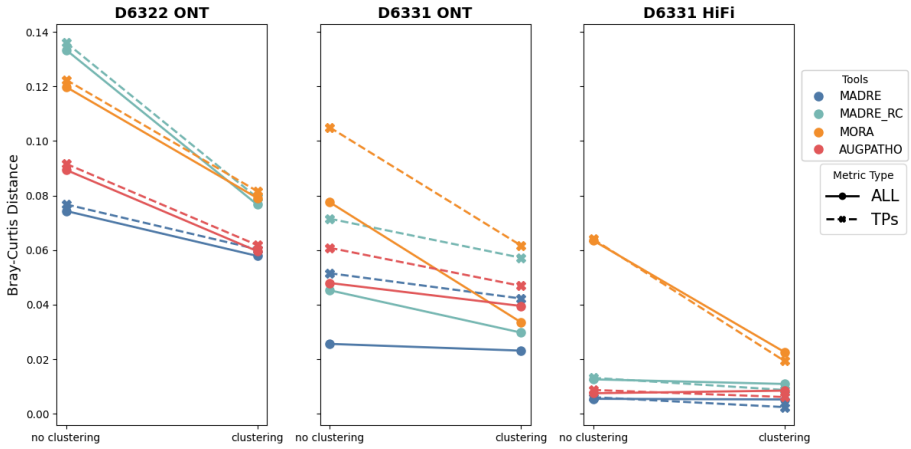

Figure 4: **Bray–Curtis distances for Zymo datasets.** The plots show changes in BC distance with and without post-clustering of similar strains. Solid lines represent distances based on all classified read counts, while dashed lines show distances calculated using only true positive (TP) read counts.

Figure 4 depticts the BC distances for the zymo D6322 ONT, zymo D6331 ONT, and zymo D6331 HiFi datasets. Dotted lines and "x" markers indicate BC distances computed using only true positive classifications based on the ground truth. In the first panel, which reports results for the D6322 dataset,

MADRe clearly outperforms all other tools. Although the performance gap between MADRe and AugPatho narrows after clustering, MADRe still maintains the lowest BC distance. Before clustering, MADRe_RC exhibits the highest BC distance among all tools, while, after clustering, its performance surpasses that of MORA. The second and third panels display BC distances for the D6331 ONT and HiFi datasets, respectively. For the ONT dataset, when evaluating all classified reads, both MADRe and MADRe_RC outperform the other approaches. However, when considering only true positive (TP) read count abundances, MADRe_RC produces a higher number of false positives (FPs) in the overall read classification. MADRe still achieves the lowest BC distance, confirming its superior accuracy. For the HiFi dataset, MADRe again achieves the best performance, although the difference compared to AugPatho is relatively small. Interestingly, BC distances increase for both AugPatho and MADRe after clustering. This suggests that additional false positive classifications occur when strains within the same cluster as the expected strain are incorrectly assigned. This interpretation is further supported by the observation that BC distances are lower when considering only TP classifications.

The exact read counts, used to calculate BC distance, are listed in the Supplementary Tables (ST6-ST8).

Table 2: **False positive (FP) species and strains detected by different tools on Zymo datasets.** An organism is considered a false positive if at least one read is classified under it, but it is not in the true community.

	Tool	**D6331 ONT**	**D6331 HiFi**	**D6322 ONT**
FP Species	MADRe	18	6	9
	MADRe_RC	324	52	260
	MORA	650	166	532
	AugPatho	3	1	17
FP Strains	MADRe	177	136	58
	MADRe_RC	3255	1218	6363
	MORA	6278	4036	10368
	AugPatho	8344	4699	11589

Table 2 presents the number of false positive species and strain identifications. In the D6331 dataset, AugPatho reports the lowest number of false positive species, whereas in the D6322 dataset, MADRe performs better. This difference could be attributed to MADRe's reliance on assembly, which may introduce misassemblies or shorter contigs that resemble genomes from different species, leading to their misidentification. However, the second part of the Table 2 shows that AugPatho reports a significantly higher number of false positive strains, even exceeding MORA. This may result in a greater number of false positives within the species, including strains that are highly dissimilar to the expected ones. Since this dataset contains organisms with extremely low abun-

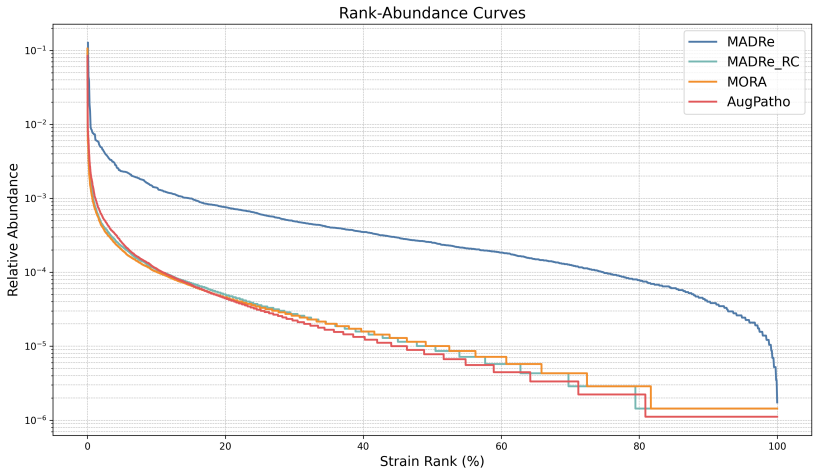

Figure 5: **Percentile-normalized rank-abundance curves.** The x-axis shows strain ranks expressed as percentiles, while the y-axis represents the relative abundance of each strain on a logarithmic scale.

dance (0.01% or less), the MADRe pipeline was unable to detect them. However, MADRe_RC demonstrated high accuracy in classifying these low-abundance organisms (Supplementary Tables ST7-ST8).

2.8 Experiment 4 - anaerobic digester sludge metagenome

While Zymo mock communities represent real metagenomic data, they do not fully capture the complexity typically found in environmental or host-associated microbial communities. To better reflect realistic classification scenarios, we evaluated MADRe and the other competing tools on a real anaerobic digester sludge metagenome. As this dataset lacks ground truth, we focused on comparative analysis of classification outputs. All results presented here include post-clustering.

In this dataset, MADRe identified 1,294 reference strains (1,498 without clustering), while MADRe_RC reported 13,185 (20,038 without clustering), MORA 14,379 (23,517 without clustering), and AugPatho 11,285 (16,488 without clustering). Out of 3,646,771 total reads, MADRe classified 575,052 ($\sim$ 16%), MADRe_RC 696,961 ($\sim$ 19%), MORA 696,839 ($\sim$ 19%) and AugPatho 898,906 ($\sim$ 25%) reads.

Figure 5 illustrates percentile-normalized rank-abundance curves, highlighting differences in strain-level classification across the tools. The underlying read count abundance data used to generate this figure is provided in Supplementary Table 9.

The curve for MADRe displays a consistent, moderately steep gradient throughout, with notable deviations at the beginning and end. The sharp rise at the beginning indicates the presence of a highly abundant strain, significantly

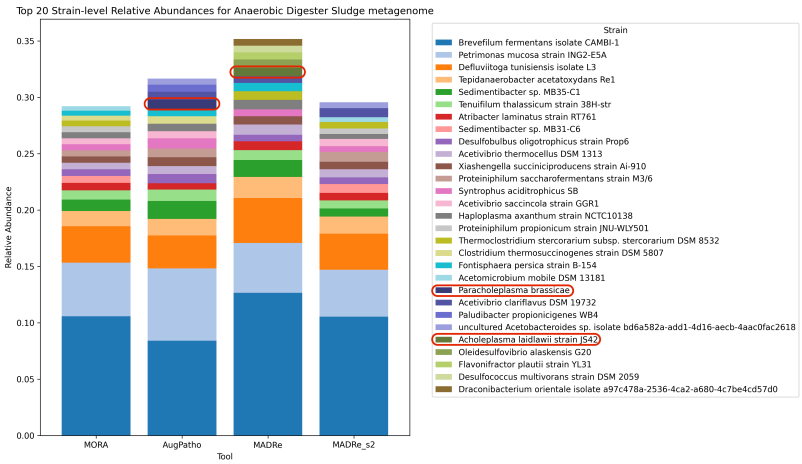

Figure 6: **Real data strain-level abundances of the top 20 most abundant strains identified by each tool.** Two strains are highlighted in red to illustrate cases where different tools classified reads originating from an unrepresented reference to distinct false positives that share similar genomic regions.

more dominant than the others. This can be seen for the other tools as well. Toward the end, the curve drops sharply, likely reflecting false positives or low-confidence strain assignments. Compared to the other tools, MADRe shows a smoother and more gradual decline in the abundances of lower-ranked strains. In contrast, MADRe_RC, MORA, and AugPatho report a larger number of low-abundance strains, resulting in a more stepwise decline. The flat tail in their curves suggests that many strains are assigned near-zero abundances, which could reflect either false positives or low-confidence classifications.

Figure 6 presents the relative abundances of the top 20 most abundant strains reported by each tool, calculated relative to the total number of classified reads. We also generated the same visualization at the species level (Supplementary Figure S5), which includes Kraken2 results. Among all tools, MADRe showed the highest cumulative relative abundance for the top 20 strains, followed by AugPatho, while MADRe_RC and MORA had similar and lower overall contributions from their top strains.

Figure 6 highlights two notable strain-level discrepancies. The strain *Paludibacter propionicigenes* (accession number NC_022549.1, taxID 6135), appeared only in the top 20 rankings of AugPatho, while *Acholeplasma laidlawii* strain JS42 (accession number NZ_CP103951.1, taxID 2148) was reported only in MADRe's top 20. To investigate these discrepancies, we compared the reads classified under these strains. We found that the majority of reads classified with taxID 6135 by AugPatho were classified with taxID 2148 by MADRe, and vice versa. Since these strains both belong to the *Acholeplasmataceae* family, this suggests potential shared genomic regions between them.

We additionally mapped the relevant reads to both references and observed that neither produced strong, confident alignments, indicating that the true

source strain is likely absent from the reference database (Supplementary Figure S6). We then assembled the reads into contigs and classified them using Kraken2 against the original large database. In six out of nine contigs, the highest number of k-mers aligned to strain 2148, although k-mer counts were low, again supporting the hypothesis of a missing true reference.

Interestingly, strain 6135 is longer than 2148, and prior work on MORA has shown that AugPath's scoring tends to favor longer, more complete reference sequences. This likely explains AugPatho's preference for 6135. The corresponding reads in MORA were scattered across several references coming from the same family. In the case of MADRe_RC, approximately half of the reads agreed with MADRe's classification, while the remaining reads were scattered across a broader set of low-abundance strains.

2.9 Time and Memory Resources

Table 3: **Memory (RSS peak in GB) and CPU time (in seconds)** for different tools, split between Minimap2 mapping and other processing steps.

Tool	Minimap2 RSS (GB)	Minimap2 CPU (s)	Other RSS (GB)	Other CPU (s)	Max RSS (GB)	Total CPU (s)
MADRe	**40.22**	**61 360.38**	3.25	94 977.23	**40.22**	**156 337.61**
MADRe_RC	76.42	2 424 656.39	3.41	**150.97**	76.42	2 424 807.36
MORA	76.42	2 424 656.39	17.03	436.17	76.42	2 425 092.56
AugPatho	76.42	2 424 656.39	**1.89**	328.71	76.42	2 424 985.10
Kraken2	0.00	0.00	70.11	1 537.11	70.11	1 537.11

Table 3 presents the runtime and peak memory usage of the benchmarking tools on the ZymoD6331 ONT dataset.

Since all tools, except Kraken2, rely on Minimap2 for read mapping, we categorized peak memory usage into three components: memory used by Minimap2, memory used by other operations, and overall memory usage. Similarly, CPU time was divided into time spent by Minimap2 and time spent on all other processing steps.

In the case of MADRe_RC, MORA, and AugPatho, the "other operations" category solely consists of the read reassignment algorithm. In contrast, for MADRe it includes assembly, HairSplitter, database reduction, and read reassignment. The role of Minimap2 also differs across the tools. In MADRe_RC, MORA, and AugPatho, it is used for mapping reads to the large reference database, whereas in MADRe it is used both to map contigs to the large database and reads to the reduced one.

For HiFi reads, Minimap2 uses different parameters, and the MADRe pipeline employs metaMDBG instead of metaFlye for assembly. To account for these differences, Supplementary Table 10 reports the same performance metrics for HiFi data.

3 Discussion

In this work, we introduced MADRe, a metagenomic classification pipeline based on assembly-driven database reduction, followed by read classification through mapping and reassignment. This approach enables accurate strain-level classification without requiring any prior knowledge about the sample composition.

Many existing strain-level tools either require single-species input or cannot scale to large reference sets. While tools like Kraken2 perform well at the species level and can handle large databases, extending them to strain-level classification typically requires additional database preparation steps. These steps can be computationally expensive and impractical for complex metagenomic samples.

MADRe addresses this by enabling strain-level classification from large, multi-species databases without need of prior assumptions. The first phase of MADRe reduces the reference database by identifying likely candidate strains. Based on assembly and an EM soft clustering algorithm, this step aims to preserve the references truly present in the sample while eliminating unrelated ones. To take advantage of the longer contigs produced by standard assemblers, we avoided using strain-aware metagenome assemblers such as Strainberry [54], MetaBooster [55], HyLight [55], stRainy [56], and HairSplitter [45], which are known to yield shorter contigs. Instead, we used HairSplitter's functionality to estimate the number of collapsed strains per contig and integrated this information with the mapping data of the initially strain-collapsed contigs.

The evaluation shows that MADRe achieves effective database reduction while maintaining high recall. Compared to two baseline models, MADRe retained all expected strains and introduced fewer false positives, highlighting the value of incorporating reassignment based on mapping evidence and collapsed strain estimation.

Our benchmarking analysis compared MADRe to MADRe_RC, MORA, and AugPatho. On simulated datasets and Zymo mock communities, MADRe consistently outperformed the others. While AugPatho performed well in some cases, especially when post-classification clustering was applied, MADRe achieved higher overall F1 score with fewer false positives. For AugPatho, we used the updated SAM file produced by its reassignment step, in which individual reads can be associated with multiple references. This format may improve the detection of expected references but can also introduce ambiguity, contributing to higher false positive rates. MADRe_RC was more successful in detecting low-abundance organisms but also reported many false positives, similar to MORA and AugPatho. It is difficult to determine whether such low abundance organisms are truly present in real samples. By leveraging assembly, MADRe focuses on strains with sufficient abundance to be reliably identified, reducing the risk of false positives. The strong performance of MADRe_RC supports the reliability of the database reduction step, so classification remains effective if relevant references are retained. Also, MADRe_RC achieved better performance on simulated datasets compared to real Zymo datasets, likely because in real data, which contain shorter reads, the algorithm benefits from some prior reduction of database, whereas in simulated data, the reads are longer allowing

more confident classification even without prior reduction.

In the case of the real metagenomic dataset, MADRe classified fewer strains and focused on a confident subset of dominant organisms. In contrast, MADRe_RC, MORA, and AugPatho reported many more low-abundance strains. While this reflects higher sensitivity, it also indicates that many of these may be spurious strain-level assignments, especially when exact strain resolution is not supported by the data. In such cases, applying filtering thresholds for low-abundance organisms is common [57], although this can lead to the removal of truly present organisms.

In cases where the correct strain was not present in the reference database, MADRe assigned reads to the most similar available references, while AugPatho favored longer ones, and MORA and MADRe_RC dispersed these reads across many low-abundance strains. These behaviors reveal a future research direction by suggesting that analyzing patterns of ambiguous mappings could help detect novel or missing strains.

Computational efficiency is a critical consideration in metagenomic classification. Although MADRe incorporates an assembly step, which is typically memory- and time-intensive, it still consumes less memory than mapping raw reads to a large database or even running Kraken2 with the same database. Additionally, MADRe executes faster than full read mapping, though Kraken2 remains by far the fastest tool. The memory and runtime footprint of Minimap2 when mapping to a large database could be reduced by using a smaller reference database. However, this would require prior knowledge of the sample composition or risk missing important strains. Among the strain-aware tools, MADRe is the fastest, offering an effective balance between computational efficiency and classification accuracy, making it well-suited for high-precision, strain-level metagenomic analysis.

To address the issue of closely related strains in the database, we employed a reference clustering method during the evaluation. This approach groups references based on shared mapping profiles rather than ANI thresholds, demonstrating that mapping-based similarity can reveal biologically meaningful clusters. Although used here primarily for evaluation, this strategy opens up a future direction in which integration of clustering information into the classification process itself could enhance resolution, particularly when distinguishing between highly similar strains.

4 Conclusion

In this study, we introduced MADRe, a novel pipeline for strain-level metagenomic classification of long-read sequencing data. MADRe combines long-read assembly, EM-based contig-to-reference mapping reassignment for database reduction, and probabilistic read reassignment to deliver accurate and efficient classification, even without prior knowledge of sample composition. Unlike many existing tools, MADRe is designed to operate with large, diverse databases spanning multiple taxonomic levels, enabling high-resolution classification while

minimizing false positives.

The pipeline consists of two distinct steps: database reduction and read classification, both of which can be executed independently. If general insight into the strains present in a sample is required, the first step can be used alone. Conversely, when prior knowledge about the sample exists, or when a reduced reference set is already available, the read classification step can be applied independently. MADRe provides a practical, scalable, and modular solution for strain-level classification in complex microbial communities and contributes to advancing strain-resolved metagenomics in the era of long-read sequencing.

5 Materials and Methods

5.1 MADRe Database Reduction

The database reduction step, shown in Figure 1 and illustrated in more detail in Supplementary Figure S1, consists of two main phases: input file preparation and the database reduction process.

In the input file preparation phase, raw long metagenomic reads are first assembled using metaFlye for ONT reads or metaMDBG for HiFi reads. When multiple strains of the same species are present in a sample, the assembly process can lead to strain collapse, producing contigs that represent a blend of closely related strains rather than distinct strain-specific sequences. Instead of using strain-aware metagenome assemblers, which typically generate shorter contigs, we chose to retain the longer contigs and infer strain-level complexity using HairSplitter which estimates the number of collapsed strains per contig.

Assembled contigs are mapped to the reference database using Minimap2 with the *asm5* parameter preset, generating a PAF file as output. We chose this preset because, compared to *asm10* and *asm20*, it provides higher sensitivity, which is crucial for capturing more accurate and complete alignments of contigs to highly similar reference genomes. The MADRe database reduction process takes two key inputs: the estimated number of collapsed strains per contig determined by HairSplitter and the contig-to-reference mappings from Minimap2.

The database reduction process is based on the EM algorithm, which reassigns contigs to different references while performing soft clustering, allowing a single contig to be assigned to multiple references with different probabilities. The EM algorithm is widely used for handling ambiguous mappings in metagenomic classification [27, 46, 25, 26, 58, 39]. The implementation of the EM algorithm in MADRe is inspired by PathoScope2 [46] and EMU [26].

The database reduction process consists of three main steps. We can define a set of contigs as $C = \{c_1, c_2, \ldots, c_x\}$, where x is the number of contigs in the assembly. The set of references to which at least one contig is mapped is defined as $R = \{r_1, r_2, \ldots, r_g\}$, where g is the number of references. Additionally, let M represent the set of all of the mappings. In the first step, we compute a mapping score H for each mapping in the PAF file using the equation:

$$\mathrm{H} = 2 * \frac{\mathrm{N} \times ml}{\mathrm{N} + ml} \tag{1}$$

which represents the harmonic mean between the exact number of matches N and the mapping length ml. The ml is defined as the maximum value between the query mapping length and the reference mapping length. Applying the harmonic mean allows us to emphasize the smaller value, ensuring that a mapping does not receive an inflated score due, for example, to a very long but low-quality alignment.

The summarized mapping value S is then calculated for each contig-reference pair using:

$$\mathrm{S}(c_i, r_j) = \sum_{m \in M(c_i, r_j)} \mathrm{H}(p) \tag{2}$$

where $M(c_i, r_j)$ represents the set of mappings of contig c_i to reference r_j. This ensures that $\mathrm{S}(c_i, r_j)$ is computed by summing the mapping values of all instances where c_i maps to r_j, thus capturing all possible alignments between the contig and the reference. Following this, we divided mappings into *unique* and *non-unique*. Unique mappings occur when a contig maps exclusively to a single reference, whereas non-unique mappings represent ambiguous cases that require further resolution.

In the second step, non-unique mappings are processed using the EM algorithm, which iteratively refines contig assignments based on mapping probabilities. The E-step updates the expected assignments of contigs, while the M-step re-estimates the parameters using the newly computed assignment probabilities from the previous iteration.

The probability of selecting a reference y_i is given by:

$$P(r_i) = \frac{1}{G}, \quad \text{where} \quad G = |R| \tag{3}$$

The conditional probability of c_i given r_i is expressed as:

$$P(c_i | r_i) = \frac{\mathrm{S}(c_i, r_i)}{\max_{r_j \in R} \mathrm{S}(c_i, r_j)} \tag{4}$$

The log-likelihood function $L(X)$ is given by:

$$L(C) = \sum_{i=1}^{X} \log \left(\sum_{j=1}^{G} P(c_i \mid r_j) \cdot P(r_j) \right) \tag{5}$$

The expectation step (E-step) updates the posterior probability $P(r_i \mid c_i)$ as follows:

$$P(r_i \mid c_i) = \frac{P(c_i \mid r_i) \cdot P(r_i)}{\sum_{j=1}^{G} P(c_i \mid r_j) \cdot P(r_j)} \tag{6}$$

The maximization step (M-step) updates the prior probability $P(y_i)$ as follows:

$$P(y_i) = \frac{\sum_{j=1}^{X} P(r_i \mid c_j)}{|M|} \tag{7}$$

The EM algorithm runs iteratively until it converges or reaches the maximum number of iterations set by the stopping criteria. Once the algorithm outputs posterior probabilities, these values are used to determine which references will be included in the reduced database.

Before selecting references, we first classify each contig at the species level. This is done by summing the posterior probabilities across all references belonging to a species and assigning the contig to the species with the highest total probability. After determining the species classification, we retain only posterior probabilities associated with references belonging to the selected species. Finally, for each contig, we select the top N references to be included in the reduced database, where N corresponds to the number of collapsed strains, as determined by HairSplitter.

5.2 MADRe Read Classification

The read classification step in MADRe is designed to work both with and without prior database reduction. The only requirement is a PAF file - Minimap2 output file containing read-to-database mappings, where the database sequences must include NCBI taxonomic information. The MADRe read classification pipeline is illustrated in more detail in Supplementary Figure S2.

The process begins by computing the mapping scores for each alignment, where the mapping score is defined as N (number of exact matches) divided by ml (mapping length).

Mappings are then categorized into unique and non-unique. Since a read can be mapped multiple times, only the alignment with the highest mapping score is retained for each read-reference pair. If a read has a single best-scoring mapping, it is classified as a unique mapping. However, if multiple mappings share the same highest score, they are considered non-unique mappings and the read must be reassigned to a single reference.

Before reassigning non-unique mappings, reads are first clustered at the species level. This clustering is performed by counting the number of mappings a read has to different species and assigning the read to the species with the highest number of mappings.

To reassign non-unique mappings, a species-specific algorithm is applied. This algorithm analyzes the number of unique and non-unique mappings for each reference and indirectly identifies which references share a significant number of mappings, suggesting that they likely represent overlapping genomic regions and should form clusters. The fundamental assumption is that a reference truly present in the sample should have the highest total number of mappings, including both unique and high-confidence non-unique mappings. Based on this

principle, expected references are identified, and all non-uniquely mapped reads are reassigned to the most probable reference within their respective cluster.

5.2.1 Abundances calculation

At this stage, each read is assigned to a single reference genome. Based on these assignments, MADRe calculates the abundance of each detected strain. The primary abundance output file reports the number of reads assigned to each reference. However, MADRe also provides an option to compute a length-normalized abundance, which accounts for both read and reference lengths. This alternative abundance metric is calculated as:

$$\text{Abundance}(r) = \frac{\sum\limits_{i \in \text{Reads}_r} Length(i)}{\text{Length}(r)} \quad \text{for } r \in R \tag{8}$$

where R is the set of the references and $Reads_r$ is set of the reads classified under reference r.

5.2.2 Similar strains clustering

The high similarity between closely related strains and the lack of a clear threshold for defining when two sequences represent the same strain, makes it difficult to ensure that a reference database contains only unique strain sequences [29]. Some entries may correspond to highly similar strains or even to multiple assemblies of the same strain. To address this, MADRe includes an optional reference clustering step within the read classification process, designed to group similar references based on shared read mappings.

This clustering step operates on the same mapping file used in the classification stage and produces two output files: one reporting the abundances of identified clusters, and another listing the representative reference for each cluster.

The clustering procedure is illustrated in Supplementary Figure S3. It begins by using the calculated mapping scores and species-level labels. For each species-specific subset, the algorithm identifies the highest-quality mapping for each read. A binary vector is the constructed for each reference, where each bit indicates whether a given read strongly supports that reference. These binary vectors are then clustered using DBSCAN with Jaccard distance as the similarity metric. Cluster-level abundances are computed accordingly. As a result, the post-clustering abundance files report only the representative references for each cluster.

5.3 Evaluation details

Our evaluation is primarily focused on exact read-level taxonomic assignments and read count-based abundance estimates.

To ensure a fair comparison, all tools were benchmarked using the same large reference database. For tools requiring mapping files as input, namely

MORA, AugPatho, and MADRe_RC, we used a unified set of read-to-reference alignments generated by Minimap2. All reads were mapped to the full database, and the resulting SAM file was used directly for MORA and AugPatho. This SAM file was subsequently converted to PAF format using Paftools, as required by MADRe_RC.

Simulated reads were generated with the Badread tool [59], applying the whole-metagenome simulation mode with default values for chimeric, junk, and random reads. The simulation was performed using the *nanopore2023* model, which corresponds to the ONT R10.4.1. The exact command used is provided in the Supplementary file.

In case of simulated datasets, ground truth was available for every read, including its corresponding strain-level taxID and reference accession. Using this information, we computed true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) by comparing the assigned strain-level taxIDs with the expected ones. For Kraken2, we extracted read IDs and assigned taxIDs from its output. If a read was assigned to a higher taxonomic level, even if it was taxonomic correct, we treated it as a false positive, as the evaluation strictly focused on strain-level classification.

For the simulated datasets and Zymo mock communities we calculated Bray-Curtis (BC) distance as:

$$\mathrm{BC}(x, y) = \frac{\sum_{i=1}^{n} |x_i - y_i|}{\sum_{i=1}^{n} (x_i + y_i)} \tag{9}$$

Where $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$ are the abundance vectors for two samples or profiles, n is the number of strains, x_i and y_i are the abundances of the ith strain in samples x and y, respectively. $\mathrm{BC}(x, y)$ is the Bray-Curtis dissimilarity or distance, ranging from 0 (identical composition) to 1 (completely disjoint).

In the case of real metagenomic datasets, the evaluation was also performed at the species level (results presented in Supplementary Figure S5). All strain-level classifications were uplifted to their corresponding species, and read count abundances were calculated. For Kraken2, we used the species-level abundances reported in its summary file, limited to entries labeled with an "S" (species rank).

MADRe, MADRe_RC, and MORA each produce read-level classification files that associate each read with a reference genome. Since all tools shared the same mapping input, for AugPatho we ran only the ID step, which outputs an updated SAM file and a report containing reference abundance estimates. In this updated SAM file, a single read can be associated with multiple references. For evaluation purposes, we allowed such multi-reference assignments, which may slightly benefit AugPatho by increasing the number of true positives, while also increasing the risk of false positives. These trade-offs are largely neutralized when clustering is applied, as similar strains typically end up grouped in the same cluster.

We used the same clustering across all tools to ensure consistency in cluster-

based evaluation. Our clustering is based on read-to-reference mapping profiles, which can differ depending on the size and composition of the database. For example, when reads are mapped to a reduced database, the absence of certain similar references can make ambiguous mappings more resolvable. To avoid such inconsistencies, clustering was performed only once on the PAF file used for MADRe_RC, which contains read mappings to the complete reference database. During cluster-level evaluation, a classification was considered a true positive if the read was assigned to a reference that belongs to the same cluster as the ground truth reference.

All commands used to perform classification with the evaluated tools are listed in the Supplementary file.

6 Data availability

The source code for MADRe is avaliable at https://github.com/lbcb-sci/MADRe. Simulated data can be accessed via Zenodo. Zymo D6322 ONT dataset is obtained from BioProject PRJNA1240873, zymo D6331 ONT dataset if obtained from [47], and zymo D6331 PacBio HiFi dataset from [48].

7 Contributions

K.K. and M.Š. conceived the study. J.L. designed and implemented the pipeline. K.K. supervised database reduction implementation. R.V. supervised read classification implementation. J.L. drafted the manuscript. K.K., R.V. and M.Š. revised the manuscript. All authors read and approved the manuscript.

8 Funding

This work was supported by the Croatian Science Foundation under grants IP-2018-01-5886 (SIGMA) and MOBDOK-2023-2941, and by the Singapore Ministry of Health's National Medical Research Council, Singapore, under the grant MOH-000649-01 (Rapid diagnostic of infectious diseases based on nanopore sequencing and AI methods) – Individual Research Grant (NMRC/OFIRG/MOH-000649-00).

9 Competing interests

M.Š. has been jointly funded by Oxford Nanopore Technologies and AI Singapore for the project AI-driven De Novo Diploid Assembler. The remaining authors declare no competing interests.

References

[1] Jack A Gilbert, Martin J Blaser, J Gregory Caporaso, Janet K Jansson, Susan V Lynch, and Rob Knight. Current understanding of the human microbiome. *Nature medicine*, 24(4):392–400, 2018.

[2] Losee L Ling, Tanja Schneider, Aaron J Peoples, Amy L Spoering, Ina Engels, Brian P Conlon, Anna Mueller, Till F Schäberle, Dallas E Hughes, Slava Epstein, et al. A new antibiotic kills pathogens without detectable resistance. *Nature*, 517(7535):455–459, 2015.

[3] Jennifer Lu, Natalia Rincon, Derrick E Wood, Florian P Breitwieser, Christopher Pockrandt, Ben Langmead, Steven L Salzberg, and Martin Steinegger. Metagenome analysis using the kraken software suite. *Nature protocols*, 17(12):2815–2839, 2022.

[4] Wenhai Zhang, Yuansheng Liu, Jialu Xu, Enlian Chen, Alexander Schönhuth, and Xiao Luo. Pantax: Strain-level taxonomic classification of metagenomic data using pangenome graphs. *bioRxiv*, pages 2024–11, 2024.

[5] Duy Tin Truong, Eric A Franzosa, Timothy L Tickle, Matthias Scholz, George Weingart, Edoardo Pasolli, Adrian Tett, Curtis Huttenhower, and Nicola Segata. Metaphlan2 for enhanced metagenomic taxonomic profiling. *Nature methods*, 12(10):902–903, 2015.

[6] Aitor Blanco-Míguez, Francesco Beghini, Fabio Cumbo, Lauren J McIver, Kelsey N Thompson, Moreno Zolfo, Paolo Manghi, Leonard Dubois, Kun D Huang, Andrew Maltez Thomas, et al. Extending and improving metagenomic taxonomic profiling with uncharacterized species using metaphlan 4. *Nature Biotechnology*, 41(11):1633–1644, 2023.

[7] Hans-Joachim Ruscheweyh, Alessio Milanese, Lucas Paoli, Anna Sintsova, Daniel R Mende, Georg Zeller, and Shinichi Sunagawa. motus: profiling taxonomic composition, transcriptional activity and strain populations of microbial communities. *Current Protocols*, 1(8):e218, 2021.

[8] Xi Chen, Xiaole Yin, Xianghui Shi, Weifu Yan, Yu Yang, Lei Liu, and Tong Zhang. Melon: metagenomic long-read-based taxonomic identification and quantification using marker genes. *Genome Biology*, 25(1):226, 2024.

[9] Peter Menzel, Kim Lee Ng, and Anders Krogh. Fast and sensitive taxonomic classification for metagenomics with kaiju. *Nature communications*, 7(1):11257, 2016.

[10] Benjamin Buchfink, Chao Xie, and Daniel H Huson. Fast and sensitive protein alignment using diamond. *Nature methods*, 12(1):59–60, 2015.

[11] Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

[12] Daniel H Huson, Benjamin Albrecht, Caner Bağcı, Irina Bessarab, Anna Gorska, Dino Jolic, and Rohan BH Williams. Megan-lr: new algorithms allow accurate binning and easy interactive exploration of metagenomic long reads and contigs. *Biology direct*, 13:1–17, 2018.

[13] Josip Marić, Krešimir Križanović, Sylvain Riondet, Niranjan Nagarajan, and Mile Šikić. Comparative analysis of metagenomic classifiers for long-read sequencing datasets. *BMC bioinformatics*, 25(1):15, 2024.

[14] Derrick E Wood, Jennifer Lu, and Ben Langmead. Improved metagenomic analysis with kraken 2. *Genome biology*, 20:1–13, 2019.

[15] Florian P Breitwieser, Daniel N Baker, and Steven L Salzberg. Krakenuniq: confident and fast metagenomics classification using unique k-mer counts. *Genome biology*, 19:1–10, 2018.

[16] Jennifer Lu, Florian P Breitwieser, Peter Thielen, and Steven L Salzberg. Bracken: estimating species abundance in metagenomics data. *PeerJ Computer Science*, 3:e104, 2017.

[17] Daehwan Kim, Li Song, Florian P Breitwieser, and Steven L Salzberg. Centrifuge: rapid and sensitive classification of metagenomic sequences. *Genome research*, 26(12):1721–1729, 2016.

[18] Rachid Ounit, Steve Wanamaker, Timothy J Close, and Stefano Lonardi. Clark: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers. *BMC genomics*, 16:1–13, 2015.

[19] Rachid Ounit and Stefano Lonardi. Higher classification sensitivity of short metagenomic reads with clark-s. *Bioinformatics*, 32(24):3823–3825, 2016.

[20] Vitor C Piro, Temesgen H Dadi, Enrico Seiler, Knut Reinert, and Bernhard Y Renard. ganon: precise metagenomics classification against large and up-to-date sets of reference sequences. *Bioinformatics*, 36(Supplement_1):i12–i20, 2020.

[21] Vitor C Piro and Knut Reinert. ganon2: up-to-date and scalable metagenomics analysis. *bioRxiv*, pages 2023–12, 2023.

[22] Jens-Uwe Ulrich and Bernhard Y Renard. Fast and space-efficient taxonomic classification of long reads with hierarchical interleaved xor filters. *Genome Research*, 34(6):914–924, 2024.

[23] Jim Shaw and Yun William Yu. Metagenome profiling and containment estimation through abundance-corrected k-mer sketching with sylph. *bioRxiv*, pages 2023–11, 2023.

[24] Alexander T Dilthey, Chirag Jain, Sergey Koren, and Adam M Phillippy. Strain-level metagenomic assignment and compositional estimation for long reads with metamaps. *Nature communications*, 10(1):3066, 2019.

[25] Changjin Hong, Solaiappan Manimaran, Ying Shen, Joseph F Perez-Rogers, Allyson L Byrd, Eduardo Castro-Nallar, Keith A Crandall, and William Evan Johnson. Pathoscope 2.0: a complete computational framework for strain identification in environmental or clinical sequencing samples. *Microbiome*, 2:1–15, 2014.

[26] Kristen D Curry, Qi Wang, Michael G Nute, Alona Tyshaieva, Elizabeth Reeves, Sirena Soriano, Qinglong Wu, Enid Graeber, Patrick Finzer, Werner Mendling, et al. Emu: species-level microbial community profiling of full-length 16s rrna oxford nanopore sequencing data. *Nature methods*, 19(7):845–853, 2022.

[27] Andrew Zheng, Jim Shaw, and Yun William Yu. Mora: abundance aware metagenomic read re-assignment for disentangling similar strains. *BMC bioinformatics*, 25(1):161, 2024.

[28] Lorian Schaeffer, Harold Pimentel, Nicolas Bray, Páll Melsted, and Lior Pachter. Pseudoalignment for metagenomic read assignment. *Bioinformatics*, 33(14):2082–2088, 2017.

[29] Thea Van Rossum, Pamela Ferretti, Oleksandr M Maistrenko, and Peer Bork. Diversity within species: interpreting strains in microbiomes. *Nature Reviews Microbiology*, 18(9):491–506, 2020.

[30] Chengwei Luo, Seth T Walk, David M Gordon, Michael Feldgarden, James M Tiedje, and Konstantinos T Konstantinidis. Genome sequencing of environmental escherichia coli expands understanding of the ecology and speciation of the model bacterial species. *Proceedings of the National Academy of Sciences*, 108(17):7200–7205, 2011.

[31] Nadav Kashtan, Sara E Roggensack, Sébastien Rodrigue, Jessie W Thompson, Steven J Biller, Allison Coe, Huiming Ding, Pekka Marttinen, Rex R Malmstrom, Roman Stocker, et al. Single-cell genomics reveals hundreds of coexisting subpopulations in wild prochlorococcus. *Science*, 344(6182):416–420, 2014.

[32] Siegfried Schloissnig, Manimozhiyan Arumugam, Shinichi Sunagawa, Makedonka Mitreva, Julien Tap, Ana Zhu, Alison Waller, Daniel R Mende, Jens Roat Kultima, John Martin, et al. Genomic variation landscape of the human gut microbiome. *Nature*, 493(7430):45–50, 2013.

[33] Moran Yassour, Eeva Jason, Larson J Hogstrom, Timothy D Arthur, Surya Tripathi, Heli Siljander, Jenni Selvenius, Sami Oikarinen, Heikki Hyöty, Suvi M Virtanen, et al. Strain-level analysis of mother-to-child bacterial transmission during the first few months of life. *Cell host & microbe*, 24(1):146–154, 2018.

[34] Lucas R van Dijk, Bruce J Walker, Timothy J Straub, Colin J Worby, Alexandra Grote, Henry L Schreiber IV, Christine Anyansi, Amy J Pickering, Scott J Hultgren, Abigail L Manson, et al. Strainge: a toolkit to track and characterize low-abundance strains in complex microbial communities. *Genome biology*, 23(1):74, 2022.

[35] Davide Albanese and Claudio Donati. Strain profiling and epidemiology of bacterial species from metagenomic sequencing. *Nature communications*, 8(1):2260, 2017.

[36] Märt Roosaare, Mihkel Vaher, Lauris Kaplinski, Märt Möls, Reidar Andreson, Maarja Lepamets, Triinu Kõressaar, Paul Naaber, Siiri Kõljalg, and Maido Remm. Strainseeker: fast identification of bacterial strains from raw sequencing reads using user-provided guide trees. *PeerJ*, 5:e3353, 2017.

[37] Grégoire Siekaniec, Emeline Roux, Téo Lemane, Eric Guédon, and Jacques Nicolas. Identification of isolated or mixed strains from long reads: a challenge met on streptococcus thermophilus using a minion sequencer. *Microbial genomics*, 7(11):000654, 2021.

[38] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1):1–22, 1977.

[39] Giorgos Skoufos, Fatemeh Almodaresi, Mohsen Zakeri, Joseph N Paulson, Rob Patro, Artemis G Hatzigeorgiou, and Ioannis S Vlachos. Agamemnon: an accurate metagenomics and metatranscriptomics quantification analysis suite. *Genome biology*, 23(1):39, 2022.

[40] Chirag Jain, Luis M Rodriguez-R, Adam M Phillippy, Konstantinos T Konstantinidis, and Srinivas Aluru. High throughput ani analysis of 90k prokaryotic genomes reveals clear species boundaries. *Nature communications*, 9(1):5114, 2018.

[41] David Koslicki, Stephen White, Chunyu Ma, and Alexei Novikov. Yacht: an ani-based statistical test to detect microbial presence/absence in a metagenomic sample. *Bioinformatics*, 40(2):btae047, 2024.

[42] Christine Anyansi, Timothy J Straub, Abigail L Manson, Ashlee M Earl, and Thomas Abeel. Computational methods for strain-level microbial detection in colony and metagenome sequencing data. *Frontiers in Microbiology*, 11:1925, 2020.

[43] Nathan LaPierre, Mohammed Alser, Eleazar Eskin, David Koslicki, and Serghei Mangul. Metalign: efficient alignment-based metagenomic profiling via containment min hash. *Genome biology*, 21:1–15, 2020.

[44] David Koslicki and Hooman Zabeti. Improving minhash via the containment index with applications to metagenomic analysis. *Applied Mathematics and Computation*, 354:206–215, 2019.

[45] Roland Faure, Dominique Lavenier, and Jean-François Flot. Hairsplitter: haplotype assembly from long, noisy reads. *Peer Community Journal*, 4, 2024.

[46] Owen E Francis, Matthew Bendall, Solaiappan Manimaran, Changjin Hong, Nathan L Clement, Eduardo Castro-Nallar, Quinn Snell, G Bruce Schaalje, Mark J Clement, Keith A Crandall, et al. Pathoscope: species identification and strain attribution with unassembled sequencing data. *Genome research*, 23(10):1721–1729, 2013.

[47] Lei Liu, Yu Yang, Yu Deng, and Tong Zhang. Nanopore long-read-only metagenomics enables complete and high-quality genome reconstruction from mock and complex metagenomes. *Microbiome*, 10(1):209, 2022.

[48] Daniel M Portik, C Titus Brown, and N Tessa Pierce-Ward. Evaluation of taxonomic classification and profiling methods for long-read shotgun metagenomic sequencing datasets. *BMC bioinformatics*, 23(1):541, 2022.

[49] Mantas Sereika, Rasmus Hansen Kirkegaard, Søren Michael Karst, Thomas Yssing Michaelsen, Emil Aarre Sørensen, Rasmus Dam Wollenberg, and Mads Albertsen. Oxford nanopore r10. 4 long-read sequencing enables the generation of near-finished bacterial genomes from pure cultures and metagenomes without short-read or reference polishing. *Nature methods*, 19(7):823–826, 2022.

[50] Nuala A O'Leary, Mathew W Wright, J Rodney Brister, Stacy Ciufo, Diana Haddad, Rich McVeigh, Bhanu Rajput, Barbara Robbertse, Brian Smith-White, Danso Ako-Adjei, et al. Reference sequence (refseq) database at ncbi: current status, taxonomic expansion, and functional annotation. *Nucleic acids research*, 44(D1):D733–D745, 2016.

[51] J Roger Bray and John T Curtis. An ordination of the upland forest communities of southern wisconsin. *Ecological monographs*, 27(4):326–349, 1957.

[52] Zymo Research. ZymoBIOMICS: Microbiomics Solutions, 2025. Accessed: 2025-03-14.

[53] Xiaowen Feng, Haoyu Cheng, Daniel Portik, and Heng Li. Metagenome assembly of high-fidelity long reads with hifiasm-meta. *Nature methods*, 19(6):671–674, 2022.

[54] Riccardo Vicedomini, Christopher Quince, Aaron E Darling, and Rayan Chikhi. Strainberry: automated strain separation in low-complexity metagenomes using long reads. *Nature Communications*, 12(1):4485, 2021.

[55] Xiao Luo, Xiongbin Kang, and Alexander Schönhuth. Enhancing long-read-based strain-aware metagenome assembly. *Frontiers in genetics*, 13:868280, 2022.

[56] Ekaterina Kazantseva, Ataberk Donmez, Maria Frolova, Mihai Pop, and Mikhail Kolmogorov. Strainy: phasing and assembly of strain haplotypes from long-read metagenome sequencing. *Nature Methods*, 21(11):2034–2043, 2024.

[57] H Ye Simon, Katherine J Siddle, Daniel J Park, and Pardis C Sabeti. Benchmarking metagenomics tools for taxonomic classification. *Cell*, 178(4):779–794, 2019.

[58] Nicolae Sapoval, Yunxi Liu, Kristen D Curry, Bryce Kille, Wenyu Huang, Natalie Kokroko, Michael G Nute, Alona Tyshaieva, Alexander Dilthey, Erin K Molloy, et al. Lightweight taxonomic profiling of long-read metagenomic datasets with lemur and magnet. *bioRxiv*, 2024.

[59] Ryan R Wick. Badread: simulation of error-prone long reads. *Journal of Open Source Software*, 4(36):1316, 2019.