MAGE: Strain Level Profiling of Metagenome Samples

Vidushi Walia, Saipradeep V.G, Rajgopal Srinivasan, Naveen Sivadasan

TCS Research, Hyderabad, India
{vidushi.walia, saipradeep.v, rajgopal.srinivasan, naveen.sivadasan}@tcs.com

***Abstract*—Metagenomic profiling from sequencing data aims to disentangle a microbial sample at lower ranks of the taxonomy such as species and strains with precise measurement of abundances. State-of-the-art tools for metagenomic profiling often rely on homology based and marker based identification of marker sequences using complex and compute intensive methods. Such approaches often have difficulties in accurate strain level profiling due to limited read support. Furthermore, incorporation of updates to the reference data are expensive. We develop MAGE, which is a non-homology based, non-marker based and alignment free strain level profiling tool for accurate abundance estimation. MAGE works with a k-mer based index of the reference data. MAGE uses a combination of the well-known FM index and R-index, together with an efficient meta-index for constructing a compact and comprehensive index. Using the read mapping results MAGE performs strain level abundance estimations. In order to obtain accurate estimates, MAGE performs a novel local search based optimization which computes maximum likelihood estimates subject to constraints on read coverage. We performed several experiments for strain level profiling, where MAGE showed superior performance compared to the state-of-the-art on a wide range of performance metrics.**

***Index Terms*—computational metagenomics, microbial profiling, strain profiling, abundance estimation**

I. Introduction

Metagenomics research enables detailed characterization of environmental or host-associated microbial communities. This has several applications including clinical study, disease study, study of environmental habitats, characterizing an industrial product and study of host-pathogen interactions. Metagenomic sequencing [1] is a powerful way of examining the diversity and complexity of microbial communities. Advancements in Next Generation Sequencing (NGS) technologies have greatly facilitated culture independent analysis of the microbiome, which has led to dramatic increase in the number and scope of metagenomic studies [2], [3]. An essential prerequisite for any metagenomic analysis is to disentangle the microbial sample at lower ranks of taxonomy such as species/strain with precise measurements of their abundances. Diversity and abundance of the microbial population governs the dynamics of the microbial ecosystem. Within the same species, individual strains can elicit different function and metabolic response [4], [5], necessitating strain-level identification and quantification, that are key for diagnostics and therapeutics purposes.

High similarity of genome sequences present in metagenome samples makes estimation of strain level taxonomic composition tremendously challenging. A standard microbial profiling pipeline starts with alignment/mapping of metagenomic reads to a microbial reference database. Most state-of-the-art tools work with curated database of identified marker sequences generated after complex and costly pre-processing, including phylogeny analysis of reference sequences. Approaches that make use of specific marker sequences often have difficulty in accurate strain level profiling, especially due to limited read support and high inter-strain similarity. Furthermore, incorporating new sequences or modifying the reference data can be very expensive.

There are several tools available for microbial profiling, especially for species level profiling [6], [7], [8], [9], [10], [11], [12]. MetaPhlAn2 [6], [7] and Centrifuge [12] are tools that are capable of performing strain level profiling.

MetaPhlAn2 [6] is a marker based approach that uses a catalogue of around one million clade-specific marker sequences defined at different levels of taxonomy. MetaPhlAn2 is part of the Biobakery2 suite [13] and it makes use of the IMG (Integrated Microbial Genomes) database [14], [15]. The IMG database contains sequence data from multiple sources, including the Refseq database [16]. It then uses Bowtie2 [17] to map all reads against the marker set to detect the taxonomic clades present in a microbiome sample. Clade-specific markers used by MetaPhlAn2 are identified by performing homology based analysis of conserved genes at clade levels. Specifically, markers are screened based on their intra-clade universality and inter-clade uniqueness. The Biobakery suite also provides the StrainPhlAn2 [7] tool which can process the MetaPhlAn2 output to generate strain level MSA and phylogenetic tree for the reconstructed strains. Biobakery has recently updated its suite to Biobakery3 [8] which is based on the updated database, called ChocoPhlAn3. The marker sequences are tightly connected to the genome collection available while indexing. As a consequence, the marker sequences have to be recomputed when the genome collection is updated.

Centrifuge [12] classifies metagenomic reads based on a memory efficient FM-Index. Centrifuge pre-processes the input reference sequences to construct species-wise compressed genomic sequences to further reduce the size of the FM-Index. In order to construct species-wise compressed genomes, Centrifuge considers all the strains from a single species and uses jellyfish [18] to extract k-mers from each strain

in the species. Based on the k-mer composition of all the strains coming from a species, Centrifuge identifies two most identical strains (Centrifuge uses nucmer [19] to identify the most identical genomes). Centrifuge then compares the two most identical genomes and discards identical sequences from one of the genome and concatenates only the unique regions into the other genome. Centrifuge further continues to combine only the unique regions from the subsequent strains to build the final compressed sequence for each species in the database. These species-wise compressed sequences are then used to build the final memory-efficient FM-Index. Centrifuge considers only exact-matches of at least 16bp or more for read mapping and extends the match until a single nucleotide mismatch is observed. Centrifuge also reports abundance at various levels of taxonomy using maximum-likelihood estimation with EM heuristics.

Homology based microbial profiling approaches requires complex and compute intensive pre-processing of the reference database in order to index the reference genome collection. These approaches work on selected subset of markers which eliminates major portion of the database. This leads to estimation errors because of reduced read support while estimating the abundances. Furthermore, for such methods, significant amount of re-indexing and pre-processing is often required to incorporate updates to the reference genome collection.

We develop MAGE (Microbial Abundance GaugE), which is a non-homology based, non-marker based and alignment free profiling tool that can disentangle the metagenomic samples down to the level of strains. MAGE works with a k-mer based index of the reference data. It uses a combination of the well-known FM index [20] and R-index [21], together with an efficient meta index, for constructing a compact and comprehensive index. MAGE indexes the entire reference database without any marker identification or k-mer selection, allowing easy incorporation of modifications to the reference data. Using the MAGE index, reads are mapped to strains solely based on the k-mer composition of the reads without requiring any gapped alignment. Using the read mapping results, MAGE performs strain level abundance estimations. In order to obtain accurate estimates, MAGE performs a novel local search based optimization which computes maximum likelihood estimates subject to constraints on read coverage to estimate strain level composition and abundance. We benchmark MAGE against state-of-the-art methods using a variety of low complexity and high complexity datasets. MAGE showed superior performance compared to the state-of-the-art on a wide range of performance measures.

II. Methods

In this section, we discuss MAGE index construction, read mapping and abundance estimation.

A. MAGE Index

For a given collection of microbial genome sequences, MAGE builds a k-mer lookup index for the collection. The sequence collection comprise strain level genome sequences from across a set of species. Let $S = \{s_1, \ldots, s_N\}$ denote the collection of N input sequences. We will use the terms sequence and strain interchangeably. For each sequence in S, there is an associated internal unique reference ID. The mapping between the internal IDs and their corresponding external taxonomic IDs (strain ID and species ID) is maintained separately. The collection S is partitioned into T non-overlapping sub-collections $S_1 \cup \cdots \cup S_T$ where each sub-collection S_i is a subset S. The MAGE index is made of two level indices. In the index at level 2, denoted as L2 index, the T sub-collections are indexed separately using FM index based full text indexing that supports k-mer lookup. In particular, we use either the standard FM index or R index to index each of the T sub-collections. The partitioning is balanced in the sense that all sub-collections have comparable sizes and each sub-collection has no more than 4GB of total genomic data. This allows using efficient 32-bit implementations of FM index and R-index. The L2 index query returns the set of all sequence IDs from the sub-collection S_t that contain the query k-mer x.

The level 1 index, denoted as L1 index, is a meta-index which maps a query k-mer x to a subset of $\{1, \ldots, T\}$ denoting the subset of the level 2 sub-collections where k-mer x occurs. The k-mer lengths used in L1 index and L2 index are not required to be the same. The L1 index is constructed by creating a T length binary characteristic vector for each k-mer x that encodes its corresponding subset of $\{1, \ldots, T\}$.

B. Read Mapping

Using the two level indexes, MAGE performs read mapping purely based on k-mer hits and without any gapped alignment. For a read r, the L1 index is queried for each k-mer in the read r. The top h second level indices, for a threshold parameter h, are identified based on the cumulative k-mer hits they received from all the L1 index k-mer queries for read r. Each qualifying L2 index S_t is then queried for the read k-mers. The target strains identified by L2 index for read r are then randomly sampled to identify the final set of mapped strains $Q(r)$ for read r. The sampling is done based on the cumulative k-mer hits that each target strain received for the k-mers in read r. In order to do this, we use a probability distribution, which was constructed a priori using read simulations, that captures the probability for a read to have a given number of k-mer hits to its source sequence. The strains are randomly sampled using this distribution based on their respective cumulative hits and the final set of sampled strains constitute $Q(r)$.

C. Abundance Estimation and MLE

Before discussing the details of the optimization approach used by MAGE, we first discuss the standard MLE (Maximum Likelihood Estimation) formulation similar to [22], [23] for abundance estimation. Let $\mathcal{R}$ denote the set of mapped reads and S denote the reference strain collection. Each read r in $\mathcal{R}$ is assumed to be sampled independently and randomly from its source strain. The source strain is chosen with probability a_i, which is the relative abundance of s_i in the sample. We

make the simplifying assumption of uniform coverage, which implies that a read is generated from a location sampled uniformly at random from its source strain. Let ℓ'_j denote the effective length of strain s_j [22], [23]. Under the uniform coverage assumption, we approximate ℓ'_j as $\ell_j - \hat{r}$ where $\hat{r}$ is the average read length. More accurate estimates for ℓ'_j can also be used [23]. For a given sample, the relative abundances of all the strains is denoted by the vector $\mathbf{a}$ where the ith component denotes the relative abundance of the ith reference strain. Clearly, $\sum a_i = 1$. We recall from the previous section that for a read r, $Q(r)$ denotes the set of strains from S to which r maps. The likelihood for $\mathcal{R}$ is given by

$$\Pr(\mathcal{R} \mid \mathbf{a}) = \prod_{r \in \mathcal{R}} \Pr(r \mid \mathbf{a}) = \prod_{r \in \mathcal{R}} \sum_{s_j \in Q(r)} (x_j/\ell'_j) \quad (1)$$

where x_j and a_j are related as $x_j = (a_j \ell'_j)/\sum_{i=1}^{m}(a_i \ell'_i)$. Here, x_j is simply the length adjusted relative abundance of strain s_j. Clearly, $\sum x_i = 1$. For several reads, their corresponding sequence sets $Q(r)$ will be identical. In other words, the mapped sequence sets of the reads define a partition of $\mathcal{R} = \mathcal{C}_1 \cup \cdots \cup \mathcal{C}_R$ where $\mathcal{C}_i$ denotes the subset of reads that are mapped to the same set of sequences denoted as Q_i. Therefore, the log likelihood with respect to the strain collection S is given by

$$\mathcal{L}(S) = \sum_{i=1}^{R} n_i \log \left(\sum_{s_j \in Q_i} (x_j/\ell'_j) \right) \quad (2)$$

where n_i is the cardinality of $\mathcal{C}_i$. We solve for x_js that maximizes $\mathcal{L}$. From x_j, we solve for a_j using their relation. To reduce estimation noise, estimated abundance values below a small predefined threshold can be made zero.

The standard MLE approach usually works well for species level abundance estimation. However, MLE suffers from the problem of distributing abundance values across several related strains that are present in the reference leading to reduced specificity and strain level profiling quality. This is because, inter-strain sequence similarity within a species is much higher than inter-species similarity.

D. Constrained Optimization and Strain Level Coverages

To perform accurate strain level profiling, MAGE uses the following constrained optimization problem which is then solved using local search. EM based solution of the above MLE formulation also yields an estimate of the latent matrix $M_{R \times N}$, where $M(i,j)$ is the number of reads from the read partition $\mathcal{C}_i$ that have originated from strain s_j. We recall that all reads in $\mathcal{C}_i$ map to the same subset of strains Q_i. If strain s_j does not belong to Q_i then clearly $M(i,j) = 0$. From an optimal MLE solution, estimate for $M(i,j)$, where strain s_j belongs to Q_i, is obtained as

$$M(i,j) = n_i \cdot (x_j/\ell'_j) / \sum_{s_k \in Q_i} (x_k/\ell'_k) \quad (3)$$

where we recall that n_i is the total number of reads in the partition $\mathcal{C}_i$ and Q_j is the set of sequences to which each read in $\mathcal{C}_i$ maps to. Using matrix M, estimate for the total number of reads originating from strain s_j denoted by Y_j is given by

$$Y_j = \sum_{i=1}^{R} M(i,j) \quad (4)$$

Also, Y_j and x_j are related as

$$x_j = Y_j / N_r$$

where $N_r = \sum n_i = |\mathcal{R}|$ is the total number of reads in $\mathcal{R}$.

Using the estimate of Y_j, estimated coverage of strain s_i, denoted by $cov(j)$, is given by

$$cov(j) = Y_j \ \hat{r}/\ell'_j \quad (5)$$

where we recall that $\hat{r}$ is the average read length and ℓ'_j is the effective length of strain s_j. From the above set of equations, we obtain that optimal estimates for x_j and $cov(j)$ are related as

$$cov(j) = x_j N_r \hat{r}/\ell'_j \quad (6)$$

Coverage values $cov(j)$ for different strains can vary as it depends on both read coverage as well as the abundance of the strain in the sample. Using eq (6) and noting that N_r and $\hat{r}$ are constants, it is straightforward to verify that at an optimal MLE solution for (2), the objective value $\mathcal{L}^o$ is given by

$$\mathcal{L}^o(S) = \sum_{i=1}^{R} n_i \log \left(\sum_{s_j \in Q_i} cov(j) \right) - c \quad (7)$$

where the $c = N_r \log(\hat{r} N_r)$ is a constant. From the above expression, we see that the final optimum value depends on the cumulative coverages of the strain sets $Q_1, \ldots, Q_R$ where the cumulative coverage of set Q_i is the sum total of the strain coverages of strains in Q_i.

Suppose we wish to perform a constrained MLE where S is restricted to a subset say $S' \subset S$. When S' is specified, the original read mapping, namely, $Q_1, \ldots, Q_R$, undergoes a projection to obtain the modified mapping information $Q'_1, \ldots, Q'_R$ where Q'_i is simply $Q'_i = Q_i \cap S'$. That is, only the strains in S' are retained in Q'_i. In the rest of the paper, when a strain subset S' is specified, we assume without explicitly stating that the associated mapping information Q'_is are the projections of Q_is obtained with respect to S'. It is easy to see that restricting the strain set to S' amounts to constraining remaining strains to have zero abundance. MLE with respect to S' has its optimum value again given by $\mathcal{L}^o(S')$.

In the next section, we discuss how MAGE performs strain set restrictions (refinements) using local search, guided by strain coverages $cov(j)$ and $\mathcal{L}^o(S')$, in order to improve the specificity and profiling accuracy. We note that under strain set constraints, for a strain $s' \in S - S'$, the reads that are estimated to be from strain s' in the unconstrained MLE are redistributed to those strains in S' that occur along with s' in any of the original strain subset Q_i. This redistribution leads to an

increase in the estimated coverages of the strains in S'. From $\mathcal{L}^o(S')$, we observe that the final optimum objective value depends not only on the individual coverages $cov(j)$ of the strains in S' but more importantly the cumulative coverages with respect to the associated Q'_is. Hence, strain set restriction should be guided in a manner that increases the cumulative coverages of associated Q'_is.

One additional aspect to be considered for strain set refinement is that when the strain set S is restricted to S', some of the reads in $\mathcal{R}$ can become unmapped because none of the mapped strains for this read are present in S'. This is a low probability event if S' contains all the strains present in the sample. Let p_ϵ denote the probability of observing such an unmapped read. From the independence assumption, the probability of observing t such reads is p_ϵ^t. As a consequence, in the likelihood expression given by eq (1), each unmapped read contributes a multiplicative term p_ϵ. Lastly, an additional sparsity based penalty is introduced in the objective function to favour sparse solution. Consequently, the optimum objective value $\mathcal{L}^o$ in eq (7) is extended as

$$\mathcal{L}^*(S') = \sum_{Q_i \neq \emptyset} n_i \log \left(\sum_{s_j \in Q_i} cov(j) \right) + \lambda N_\epsilon - \beta |S'| - c \quad (8)$$

where S' is the restricted set, Q_is are the projected strain sets with respect to S', $\lambda = \log(p_\epsilon)$, β is the sparsity parameter, and N_ϵ is the number of unmapped reads with respect to S'.

E. Local Search Optimization and Strain Set Refinement

Suppose we know the ground truth set of strains S'. Then the abundance estimation is done simply by performing the constrained MLE ($\mathcal{L}^o(S')$) as discussed in the previous section. However, in practice we do not have such knowledge. Estimating $\mathcal{L}^*(S')$ for all possible subsets is also prohibitively expensive as there are exponentially many subsets. Additionally, if we have prior information on lower bounds for strain level coverages ($cov(j)$), we could further prune the search space to those S' for which the estimated coverages satisfy the additional constraints. However, this information is also hard to obtain, because, as discussed earlier, coverages $cov(j)$ depends on the strain abundances and coverage induced by the mapped reads, which can deviate due to strain mutations.

MAGE attempts to efficiently find an approximate solution S' that is close to the optimal set and having low false positives and false negatives using the following local search approach. Starting with the initial MLE solution on the on the full strain set S, MAGE iteratively removes strains from among the candidates that have low estimated coverages. The details are discussed later. Removal of such strains can induce only minimal changes to $\mathcal{L}^o(S')$ in each iteration as there are only few reads that gets redistributed in each iteration. Therefore, performing refinement in this manner helps the local search and refinement not to deviate from the desired trajectory in the solution space that leads to the true final S^*.

To start with, all strains whose coverage is below a low threshold are removed at once before performing the iterative refinement. In each iteration of strain set refinement, greedily removing the strain with the lowest estimated coverage can lead to false negatives and some of the true strains can have low estimated coverages in the current solution S' due to the presence of several similar false positive strains in the current solution S'. MAGE therefore considers the bottom b strains $\{s'_i, \ldots, s'_b\}$ having the lowest estimated coverages $cov(j)$, for some fixed b, and compute the modified $\mathcal{L}^*(S' \backslash s'_j)$ values for each of these b candidates. Among these b candidates, we finally remove, from the current solution, the strain s' whose corresponding $\mathcal{L}^*(S' \backslash s')$ is the largest. The refined set after this iteration is given by $S' \backslash s'$. The coverage values of the strains in the refined set are updated using the outcome of MLE performed (using eq(5)) for estimating $\mathcal{L}^*(S' \backslash s')$.

To reduce the overall number of MLE computations, MAGE further approximates $\mathcal{L}^*(S' \backslash s'_j)$ for a low coverage strain s'_j as follows. The current sets Q_i are first projected with respect to $S \backslash s'_j$ as discussed earlier. Using the projected Q'_is and using eq (3), $M(i,j)$ values are directly recomputed. Using these updated $M(i,j)$ values, modified coverage values $cov'(j)$ for the strains in $S \backslash s'_j$ are computed using equations (4) and (5). These modified coverage values and the modified count of unmapped reads N'_ϵ (as a consequence of removing s'_j) are used in eq (7) to approximate $\mathcal{L}^*(S \backslash s_j)$. The above approximation is performed for every t consecutive iterations, for a fixed t, followed by MLE and computation of $\mathcal{L}^*(S')$ without approximation in the subsequent iteration.

As discussed earlier, after each iteration, the coverages $cov(j)$ of the remaining strains in the solution set S' keeps increasing. However, excessive refinement will lead to significant decline in the $\mathcal{L}^*(S')$ value because of false negatives with several unmapped reads. MAGE stops iterative refinement at a step where $\mathcal{L}^*(S')$ does not improve. The associated abundance values of strains in the final S' are computed using MLE as discussed earlier. We also remark that the above local search based refinement induces a ranked ordering on the strains based on the iteration at which the strain was filtered out. The strain rank in a sense indicates the likelihood of the strain to be present in the sample.

III. Results

We compare MAGE against state-of-the-art metagenome strain level profiling tools Centrifuge [12] and MetaPhlAn2 [6].

For MetaPhlAn2, we used the tool accessed from https://github.com/biobakery/MetaPhlAn2, the strain tracking feature of MetaPhlAn2 enables profiling the samples down to the resolution of strains. MetaPhlAn2 [6] database has extended set of markers which enables strain identification and strain tracking of metagenomic samples. Centrifuge prefers uncompressed index for taxonomic profiling at the strain level [12]. Hence, we considered only those indexes that are constructed from uncompressed genomic sequences. For benchmarking MAGE's results with Centrifuge, we have considered two indexes provided by Centrifuge, which we denote as Centrifuge1 and Centrifuge2 in the results. Centrifuge1 uses "p+h+v" index for

read mapping. It includes reference genomes from Bacteria, Archae, Viruses and Human (2016 release). Centrifuge2 uses the massive "nt-database" (compressed index size 64GB) index built using NCBI nucleotide non-redundant sequences (2018 release). These indexes are available for download from https://ccb.jhu.edu/software/centrifuge/. The strain level outputs of MetaPhlAn2 and Centrifue 1&2 were mapped to the NCBI taxonomy to compare them with the ground truth.

For all experiments, the MAGE parameters were fixed as 21 for k-mer length in level 2 index, 16 for k-mer length in level 1 index and 3 for parameter h used in read mapping. In the abundance estimation phase, the parameter values used were $b = 5, t = 5, \beta = 10^6$.

We also compared our results against the standard MLE solution (described in the Section II-C). For the standard MLE, the pruning threshold was length adjusted abundance was set to 0.0001.

A. Reference Collection

MAGE index was constructed using reference genome sequences from NCBIs Refseq (complete genomes) [16], that were available at the time of download (2018). MAGE reference collection also includes genomes from Mosaic Challenge[24] that were not present in the Refseq collection. In total, MAGE reference database consisted of 24,146 strains spanning 13,168 species. The species were from 3,559 genus belonging to Virus, Bacteria, Archae, Fungi and Protozoa. The final MAGE index was $\sim$38 GB in size.

MetaPhlAn2 uses reference genomes from Integrated Microbial Genomes (IMG) system to build its database of clad-specific marker genes. The IMG database contains sequence data from multiple sources including Refseq.

The "nt-database" index of Centrifuge [12] used by Centrifuge2 contains prokaryotic sequences from NCBI BLAST's nucleotide database which contains all spliced non-redundant coding sequences from multiple database including Refseq. The "p+h+v" Centrifuge index used by Centrifuge1 consists of prokaryotic and human genome sequences downloaded from NCBI and contains sequences from Bacteria, Archae, Viruses and Human.

B. Datasets

We experimented with multiple datasets, each containing strains from across the various kingdoms of prokaryotes. We used ART [25] simulator to generate Illumina paired-end reads of 150bp length and with 20X coverage, to generate the synthetic read datasets.

We created three simulated datasets, namely, HC1, HC2 and LC. Both HC1 and HC2 datasets are high complexity metagenomic samples, where multiple strains from same species or species with same genus are present in a sample, such that the constituent strains bear high similarity. HC1 consists of 34 strains from 10 species spanning 8 genus and HC2 consists of 60 strains from 25 species spanning 25 genus. LC is a low complexity dataset most strains do not bear very high similarity with one another. The LC dataset consists of 50 strains from 41 species in total and spanning 38 genus.

To mimic the real world metagenomic samples, we also created simulated read datasets from reference strains with mutations to demonstrate the ability of MAGE to correctly identify and quantify the target strains from a genetically divergent metagenomic sample. We used two datasets, namely HC1-Mutated and HC2-Mutated. Both HC1-Mutated and HC2-Mutated are high complexity datasets consisting of 34 strains from 10 species and 60 strains from 25 species respectively. We used wgsim [26], [27] pipeline to create the mutated datatsets with 20X coverage. Wgsim introduces SNPs and insertion/deletion (INDEL) mutations in the input genome taken from the reference collection, and simulates reads from the mutated genomes. We used default parameters to generate mutated read dataset with 0.1% mutation, of which 15% of the mutations are INDELS.

We additionally generated datasets for strain knock-out experiments. Strain knock-put experiments were performed by blocking from MAGE index, some of the strains present in the sample. For knock-out experiments, we considered the above mentioned LC and the HC2 datasets. Of the 50 strains in the LC dataset, we knocked-out 10 strains from the MAGE index. Of the 60 strains in the HC2 dataset, we knocked-out 24 strains from MAGE index.

C. Metrics

We used a wide variety of metrics to benchmark the various tools used in our experiments. We used four well-known divergence measures for probability distributions to compare relative abundance distributions, the Jensen-Shannon divergence(JSD), Total variation distance(TVD), Hellinger distance(HD) and cumulative mass (CM). The Jensen-Shannon divergence (JSD) between two distributions P and Q, which is a smoothed and symmetric version of the KL-divergence [28], is given by $JSD(P\|Q) = KL(P\|M) + KL(Q\|M)$ where $M = \frac{1}{2}(P + Q)$. Square root of JS divergence is known to be a metric. Total variation distance (TVD) $\delta(P, Q)$ between distributions P and Q is a statistical distance metric and is given by $\frac{1}{2}\|P - Q\|_1$. Hellinger distance [29] $H(P, Q)$ between distributions P and Q is given by $\frac{1}{\sqrt{2}}\|\sqrt{P} - \sqrt{Q}\|_2$, where $\sqrt{P}$ denotes the component-wise square root of P. Hellingner distance is known to be a metric and is also related to Bhattacharya coefficient. JSD, TVD and HD are known to be special cases of f-divergences between distributions. The cumulative mass (CM) of the true positives for a given distribution P with respect to a true distribution Q is given by the sum $\sum_{i:Q_i \neq 0} P_i$. Additionally, we also compare the ability of the tools to correctly identify strains using the standard precision, recall and F1-score metrics. For JSD, TVD and HD, lower values indicate better performance. For CM, precision, recall and F1-scores, higher values indicate better performance.

D. Results on HC1, HC2 and LC Datasets

Table I, Table II and Table III shows the comparison of MetaPhlAn2, Centrifuge 1, Centrifuge 2, standard MLE and MAGE on the high complexity datasets HC1 and HC2 and the low complexity data LC. As seen in the tables, MAGE has significantly higher F1 score and lower distances.

Low values of distance measures indicates MAGE's ability to accurately assign abundance to the true positive strains. As results indicate, MetaPhlAn2, Centrifuge1&2 suffer from high number of false positives in the strain level profiling. Higher values of recall for MLE comes at the cost of lower precision and significantly lower F1 due to the presence of several false positives.

Tool	MetaPhlAn2	Centrifuge1	Centrifuge2	MLE	MAGE
JSD	0.8197	0.7468	0.8305	0.2829	**0.2240**
TVD	0.9714	0.8515	0.9988	0.1721	**0.0927**
HD	0.9845	0.8892	0.9923	0.3315	**0.2683**
CM	0.0331	0.1888	0.0012	0.8381	**0.9434**
Prec.	0.1111	0.0323	0.0233	0.2063	**0.9118**
Recall	0.0294	0.3235	0.2941	**0.9706**	0.9118
F1	0.0465	0.0587	0.0431	0.3402	**0.9118**

TABLE I: Strain level profiling performances on HC-1 data

Tool	MetaPhlAn2	Centrifuge1	Centrifuge2	MLE	MAGE
JSD	0.8139	0.7337	0.8294	0.3188	**0.2946**
TVD	0.9588	0.8061	0.9982	0.2033	**0.1507**
HD	0.9776	0.8783	0.9892	0.3706	**0.3524**
CM	0.0436	0.2167	0.0018	0.8308	**0.9043**
Prec.	0.0714	0.0342	0.0262	0.2624	**0.9091**
Recall	0.0333	0.2833	0.3000	**0.9667**	0.8333
F1	0.0455	0.0610	0.0481	0.4128	**0.8696**

TABLE II: Strain level profiling performances on HC-2 data

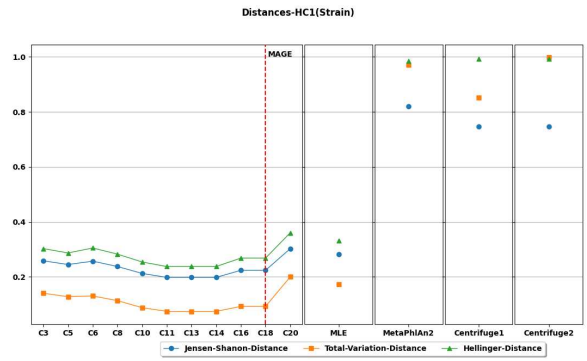
Tool	MetaPhlAn2	Centrifuge1	Centrifuge2	MLE	MAGE
JSD	0.7877	0.7157	0.8312	0.1666	**0.1587**
TVD	0.9154	0.7843	0.9993	0.0758	**0.0692**
HD	0.9453	0.8551	0.9936	0.1932	**0.1830**
CM	0.1353	0.3016	0.0006	0.9389	**0.9470**
Prec	0.2000	0.1552	0.1417	0.6757	**0.9091**
Recall	0.1000	0.3600	0.3400	**1.0000**	**1.0000**
F1	0.1333	0.2169	0.2000	0.8065	**0.9524**

TABLE III: Strain level profiling performances on LC data

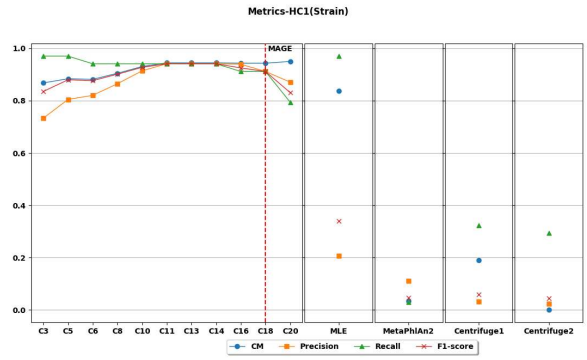
We provide plots, namely Figures 1, 2 and 3, showing the metrics computed on the intermediate results produced by MAGE for the HC1, HC2 and LC datasets. These plots provide insights into how the solution improves over the different iterations of MAGE. In particular, it compares the intermediate solutions to the final MAGE solution and the solutions given by the other methods. For the intermediate solutions of MAGE, the associated minimum coverage value of the strains in the solution is also shown in the X-axis. The metrics for the final solution output by MAGE is indicated by a red dotted line in each plot.

E. Experiments on Mutation Datasets

Table IV and Table V shows the comparison of MetaPhlAn2, Centrifuge 1, Centrifuge 2, standard MLE and MAGE the high complexity datasets HC1-Mutated and HC2-Mutated. In these experiments too, MAGE exhibits superior performance over a range of performance metrics.



(a)

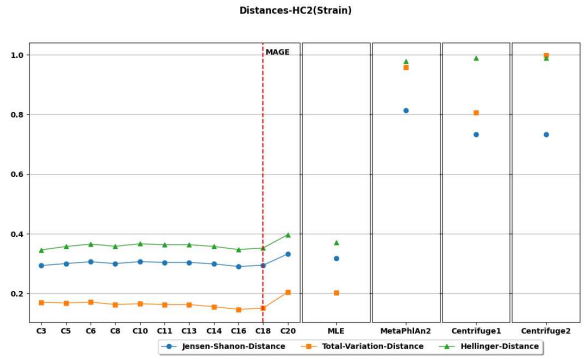


(b)

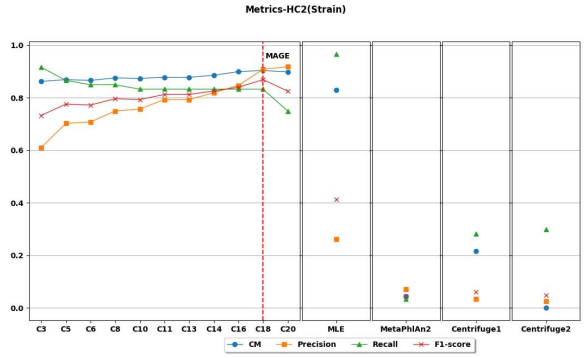
Fig. 1: Strain Level profiling performance of MAGE, MLE, MetaPhlAn2, Centrifuge1 and Centrifuge2 on HC1 dataset. The figures also show metrics on MAGE results for the intermediate iterations. The final MAGE solution is identified using a red dotted line. For (1a) lower values denote better performance while for (1b) higher values indicate better performance.

Similar to the previous experiments, we provide plots, namely Figures 4 and 5, showing the metrics computed on the intermediate results produced by MAGE and compare them the the final results produced by MAGE and the other tools.

Table VI and Table VII shows the comparison of MetaPhlAn2, Centrifuge 1, Centrifuge 2, standard MLE and MAGE on LC and HC2 datasets respectively. As described earlier, for MAGE and standard MLE, a subset of the ground truth strains were excluded from their indexes. We denote these modified indexes MAGE_X and MLE_X. For the other tools, we used their original indexes without any alteration. The tables show JSD, TVD and HD distances for all the tools. The objective here was to see how well MAGE recovers the strain abundances for the strains that were not excluded from its index, denoted as NE (non excluded) strains, in the presence of additional mutated strains (excluded strains) in the sample. Therefore, while comparing two abundance distributions, the

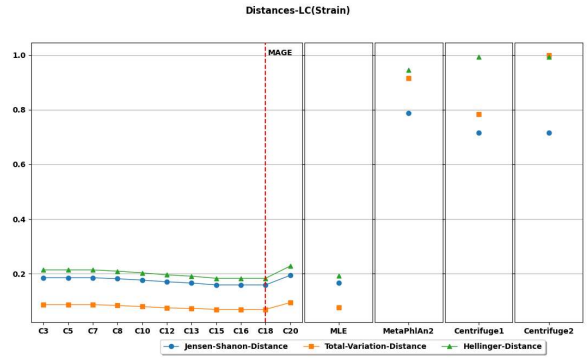


(a)

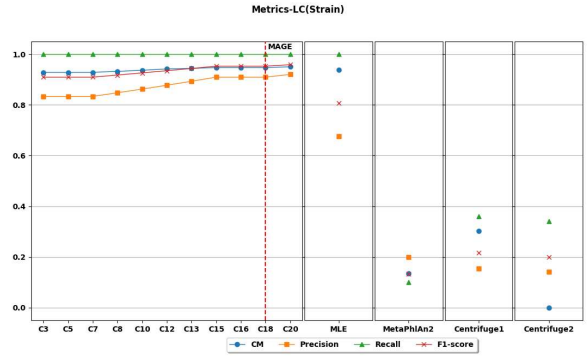


(b)

Fig. 2: Strain Level profiling performance of MAGE, MLE, MetaPhlAn2, Centrifuge1 and Centrifuge2 on HC2 dataset. The figures also show metrics on MAGE results for the intermediate iterations. The final MAGE solution is identified using a red dotted line. For (2a) lower values denote better performance while for (2b) higher values indicate better performance.



(a)



(b)

Fig. 3: Strain Level profiling performance of MAGE, MLE, MetaPhlAn2, Centrifuge1 and Centrifuge2 on the LC dataset. The figures also show metrics on MAGE results for the intermediate iterations. The final MAGE solution is identified using a red dotted line. For (3a) lower values denote better performance while for (3b) higher values indicate better performance.

abundances of strains other than NE strains were combined into a single cumulative value. The last row in these two tables show, for each tool, the number of NE strains recovered and the total number of strains reported (inside parenthesis).

Tool	MetaPhlAn2	Centrifuge1	Centrifuge2	MLE	MAGE
JSD	0.8170	0.7627	0.8308	0.5417	**0.4496**
TVD	0.9684	0.8805	0.9990	0.6088	**0.4450**
HD	0.9803	0.9084	0.9930	0.6276	**0.5285**
CM	0.0341	0.1475	0.0010	0.3912	**0.5559**
Prec	0.1538	0.0437	0.0272	0.0667	**0.5517**
Recall	0.0588	0.2941	0.3235	**0.9706**	0.9412
F1	0.0851	0.0760	0.0502	0.1248	**0.6957**

TABLE IV: Strain level profiling performances on HC1 Mutated dataset

Tool	MetaPhlAn2	Centrifuge1	Centrifuge2	MLE	MAGE
JSD	0.8029	0.7695	0.8295	0.5041	**0.4661**
TVD	0.9333	0.8771	0.9982	0.5191	**0.4359**
HD	0.9644	0.9214	0.9898	0.5869	**0.5525**
CM	0.0719	0.1328	0.0018	0.4954	**0.5867**
Prec	0.0968	0.0286	0.0290	0.1473	**0.4949**
Recall	0.0500	0.2000	0.2667	**0.9500**	0.8167
F1	0.0659	0.0501	0.0524	0.2550	**0.6164**

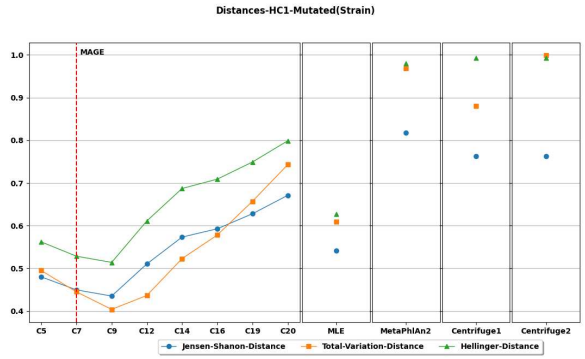
TABLE V: Strain level profiling performances on HC2 Mutated dataset

Tool	MetaPhlAn2	Centrifuge1	Centrifuge2	MLE_X	$MAGE_X$
JSD	0.5933	0.5592	0.6330	**0.0676**	0.0903
TVD	0.7026	0.6496	0.7761	**0.0564**	0.0985
HD	0.6809	0.6398	0.7198	**0.0681**	0.0905
NE (T)	4 (25)	11 (116)	12 (120)	40 (74)	39 (47)

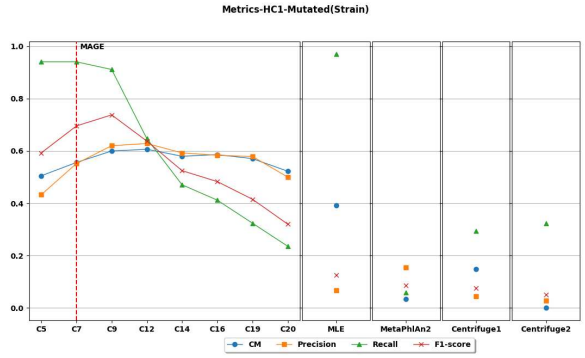
TABLE VI: Strain level profiling performances for strain exclusion on LC dataset

F. MAGE Implementation

MAGE was implemented in C++ as multi-threaded. The R-Index library [21], [30] and FM-index library [31] were used for indexing level 2 sequence sub-collections. The total size of the MAGE index was 38GB. MAGE takes around 1.9 hours using 64 threads for processing a sample of 1 million reads.

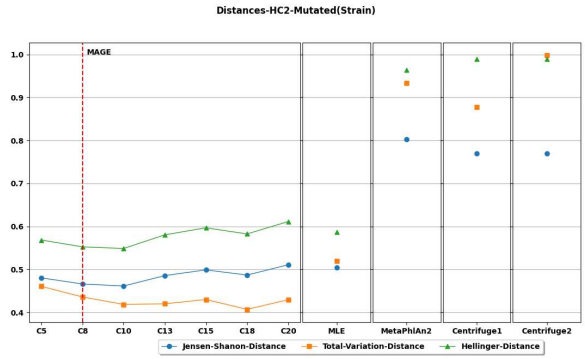


(a)

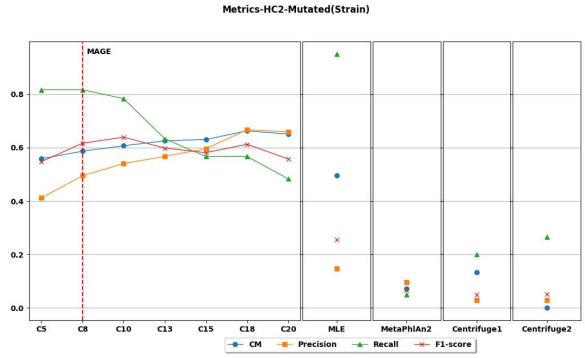


(b)

Fig. 4: Strain Level profiling performance of MAGE, MLE, MetaPhlAn2, Centrifuge1 and Centrifuge2 on the HC1-Mutated dataset. The figures also show metrics on MAGE results for the intermediate iterations. The final MAGE solution is identified using a red dotted line. For (4a) lower values denote better performance while for (4b) higher values indicate better performance.



(a)



(b)

Fig. 5: Strain Level profiling performance of MAGE, MLE, MetaPhlAn2, Centrifuge1 and Centrifuge2 on the HC2-Mutated dataset. The figures also show metrics on MAGE results for the intermediate iterations. The final MAGE solution is identified using a red dotted line. For (5a) lower values denote better performance while for (5b) higher values indicate better performance.

Tool	MetaPhlAn2	Centrifuge1	Centrifuge2	MLE_X	MAGE_X
JSD	0.5164	0.4568	0.5354	**0.1874**	0.1959
TVD	0.5838	0.4847	0.6237	0.1320	**0.1023**
HD	0.5973	0.5276	0.6106	**0.2093**	0.2325
NE (T)	4 (28)	13 (497)	14 (688)	36 (287)	30 (56)

TABLE VII: Strain level profiling performances for strain exclusion on HC2 dataset

IV. Conclusion

Shotgun metagenomic sequencing has enabled the exploration of metagenomic samples at lower ranks of taxonomy. We propose MAGE (Microbial Abundance GaugE), which is a non homology based, non marker based and alignment free tool, for accurate microbial profiling. MAGE enables investigation of metagenomic ecosystems at higher resolution of strains along with accurate estimates of their abundances. MAGE database is comprehensive, covering sequences from different kingdoms including Bacteria, Virus, Fungi, Protozoa and Archae. Our experiments showed that non-homology based and non-marker based MAGE indexing combined with alignment free mapping and MLE estimation can provide robust estimation of both strain level and species level abundances with high sensitivity and accuracy in the presence of sequencing errors. We have extensively benchmarked our results on both low complexity and high complexity datasets demonstrating the ability of MAGE for accurate profiling, including mutated strains. Absence of any complex or costly pre-processing such as homology based analysis or marker sequence identification allows easy incorporation of new sequences in the MAGE index. MAGE index is compact with size with significant data compression obtained by the use of FM index and R index. Future work include further reduction in the index size, mapping improvement, additional benchmarking, and improved optimization approaches for strain set refinement.

References

[1] H. Y. Simon, K. J. Siddle, D. J. Park, and P. C. Sabeti, "Benchmarking metagenomics tools for taxonomic classification," *Cell*, vol. 178, no. 4, pp. 779–794, 2019.

[2] M. Hamady and R. Knight, "Microbial community profiling for human microbiome projects: tools, techniques, and challenges," *Genome research*, vol. 19, no. 7, pp. 1141–1152, 2009.

[3] C. Anyansi, T. J. Straub, A. L. Manson, A. M. Earl, and T. Abeel, "Computational methods for strain-level microbial detection in colony and metagenome sequencing data," *Frontiers in Microbiology*, vol. 11, p. 1925, 2020.

[4] O. Balmer and M. Tanner, "Prevalence and implications of multiple-strain infections," *The Lancet infectious diseases*, vol. 11, no. 11, pp. 868–878, 2011.

[5] S. Alizon, J. C. de Roode, and Y. Michalakis, "Multiple infections and the evolution of virulence," *Ecology letters*, vol. 16, no. 4, pp. 556–567, 2013.

[6] D. T. Truong, E. A. Franzosa, T. L. Tickle, M. Scholz, G. Weingart, E. Pasolli, A. Tett, C. Huttenhower, and N. Segata, "Metaphlan2 for enhanced metagenomic taxonomic profiling," *Nature methods*, vol. 12, no. 10, pp. 902–903, 2015.

[7] D. T. Truong, A. Tett, E. Pasolli, C. Huttenhower, and N. Segata, "Microbial strain-level population structure and genetic diversity from metagenomes," *Genome research*, vol. 27, no. 4, pp. 626–638, 2017.

[8] F. Beghini, L. J. McIver, A. Blanco-Míguez, L. Dubois, F. Asnicar, S. Maharjan, A. Mailyan, P. Manghi, M. Scholz, A. M. Thomas, *et al.*, "Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with biobakery 3," *Elife*, vol. 10, p. e65088, 2021.

[9] D. E. Wood, J. Lu, and B. Langmead, "Improved metagenomic analysis with kraken 2," *Genome biology*, vol. 20, no. 1, pp. 1–13, 2019.

[10] D. E. Wood and S. L. Salzberg, "Kraken: ultrafast metagenomic sequence classification using exact alignments," *Genome biology*, vol. 15, no. 3, pp. 1–12, 2014.

[11] J. Lu, F. P. Breitwieser, P. Thielen, and S. L. Salzberg, "Bracken: estimating species abundance in metagenomics data," *PeerJ Computer Science*, vol. 3, p. e104, 2017.

[12] D. Kim, L. Song, F. P. Breitwieser, and S. L. Salzberg, "Centrifuge: rapid and sensitive classification of metagenomic sequences," *Genome research*, vol. 26, no. 12, pp. 1721–1729, 2016.

[13] L. J. McIver, G. Abu-Ali, E. A. Franzosa, R. Schwager, X. C. Morgan, L. Waldron, N. Segata, and C. Huttenhower, "biobakery: a metaomic analysis environment," *Bioinformatics*, vol. 34, no. 7, pp. 1235–1237, 2018.

[14] V. M. Markowitz, I.-M. A. Chen, K. Palaniappan, K. Chu, E. Szeto, Y. Grechkin, A. Ratner, I. Anderson, A. Lykidis, K. Mavromatis, *et al.*, "The integrated microbial genomes system: an expanding comparative analysis resource," *Nucleic acids research*, vol. 38, no. suppl_1, pp. D382–D390, 2010.

[15] I.-M. A. Chen, K. Chu, K. Palaniappan, A. Ratner, J. Huang, M. Huntemann, P. Hajek, S. Ritter, N. Varghese, R. Seshadri, *et al.*, "The img/m data management and analysis system v. 6.0: new tools and advanced capabilities," *Nucleic acids research*, vol. 49, no. D1, pp. D751–D763, 2021.

[16] N. A. O'Leary, M. W. Wright, J. R. Brister, S. Ciufo, D. Haddad, R. McVeigh, B. Rajput, B. Robbertse, B. Smith-White, D. Ako-Adjei, *et al.*, "Reference sequence (refseq) database at ncbi: current status, taxonomic expansion, and functional annotation," *Nucleic acids research*, vol. 44, no. D1, pp. D733–D745, 2016.

[17] B. Langmead and S. L. Salzberg, "Fast gapped-read alignment with bowtie 2," *Nature methods*, vol. 9, no. 4, p. 357, 2012.

[18] G. Marçais and C. Kingsford, "A fast, lock-free approach for efficient parallel counting of occurrences of k-mers," *Bioinformatics*, vol. 27, no. 6, pp. 764–770, 2011.

[19] S. Kurtz, A. Phillippy, A. L. Delcher, M. Smoot, M. Shumway, C. Antonescu, and S. L. Salzberg, "Versatile and open software for comparing large genomes," *Genome biology*, vol. 5, no. 2, pp. 1–9, 2004.

[20] P. Ferragina and G. Manzini, "Opportunistic data structures with applications," in *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pp. 390–398, IEEE, 2000.

[21] T. Gagie, G. Navarro, and N. Prezza, "Optimal-time text indexing in bwt-runs bounded space," in *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1459–1477, SIAM, 2018.

[22] N. L. Bray, H. Pimentel, P. Melsted, and L. Pachter, "Near-optimal probabilistic rna-seq quantification," *Nature biotechnology*, vol. 34, no. 5, pp. 525–527, 2016.

[23] A. Roberts and L. Pachter, "Streaming fragment assignment for real-time analysis of sequencing experiments," *Nature methods*, vol. 10, no. 1, pp. 71–73, 2013.

[24] "Mosaic challenge." https://platform.mosaicbiome.com/, 2017.

[25] W. Huang, L. Li, J. R. Myers, and G. T. Marth, "Art: a next-generation sequencing read simulator," *Bioinformatics*, vol. 28, no. 4, pp. 593–594, 2012.

[26] H. Li, B. Handsaker, A. Wysoker, T. Fennell, J. Ruan, N. Homer, G. Marth, G. Abecasis, and R. Durbin, "The sequence alignment/map format and samtools," *Bioinformatics*, vol. 25, no. 16, pp. 2078–2079, 2009.

[27] H. Li, "wgsim - simulating sequence reads from a reference genome." https://github.com/lh3/wgsim, 2011.

[28] G. E. Sims, S.-R. Jun, G. A. Wu, and S.-H. Kim, "Alignment-free genome comparison with feature frequency profiles (ffp) and optimal resolutions," *Proceedings of the National Academy of Sciences*, vol. 106, no. 8, pp. 2677–2682, 2009.

[29] M. S. Nikulin *et al.*, "Hellinger distance," *Encyclopedia of mathematics*, vol. 78, 2001.

[30] A. Kuhnle, T. Mun, C. Boucher, T. Gagie, B. Langmead, and G. Manzini, "Efficient construction of a complete index for pan-genomics read alignment," *Journal of Computational Biology*, vol. 27, no. 4, pp. 500–513, 2020.

[31] M. Petri, "Fm-index-compressed full-text index." https://github.com/mpetri/FM-Index, 2015.