ARCHE: AN ADVANCED FLEXIBLE TOOL FOR HIGH-THROUPUT ANNOTATION OF FUNCTIONS ON MICROBIAL CONTIGS

Daniel Gonzalo Alonso-Reyes[1] and Virginia Helena Albarracín[1, 2]*

[1]Laboratorio de Microbiología Ultraestructural Y Molecular, Centro Integral de Microscopía Electrónica (CIME,), CONICET-Universidad Nacional de Tucumán, Camino de Sirga s/n, Finca El Manantial, Yerba Buena (4107), San Miguel de Tucumán, Tucumán, Argentina

[2]Facultad de Ciencias Naturales e Instituto Miguel Lillo, Universidad Nacional de Tucumán, Tucumán, Argentina

SUMMARY

The growing amount of genomic data has prompted a need for less demanding and user friendly functional annotators. At the present, it’s hard to find a pipeline for the annotation of multiple functional data, such as both enzyme commission numbers (E.C.) and orthologous identifiers (KO and eggNOG), protein names, gene names, alternative names, and descriptions. In this work, we provide a new solution which combines different algorithms (BLAST, DIAMOND, HMMER3) and databases (UniprotKB, KOfam, NCBIFAMs, TIGRFAMs, and PFAM), and also overcome data download challenges. The software framework, Arche, herein demonstrated competitive results over *Escherichia coli* K12 genome when compared to other annotators. Finally, Arche provides an analysis pipeline that can accommodate advanced tools in a unique order, creating several advantages regarding to other commonly used annotators.

1. INTRODUCTION

A crucial step in assessing the metabolic role of a microorganism is functional gene prediction and metabolic reconstruction [1]. The functional data “annotated” by means of bioinformatic tools can be used to infer metabolic differences with close relatives, characterize novel potential metabolisms, or detect the presence of antibiotic-resistance genes or toxins, among other genes of interest.[2,3].

Genome annotation is the process of identifying and labeling all the relevant features in a genome sequence [4]. This should include coordinates of coding regions and their predicted products, but it may also be desirable to identify non-coding RNAs, signal peptides, etc. The genome annotation starts with gene identification or gene calling (structural annotation), which can be automatically performed using several tools, including Prodigal and GeneMark among others[5–7]. The next annotation step relies on the use of reference protein databases (DBs) to assign functions to query (unknown) protein sequences based on homology or orthologues searches (functional annotation) [8,9]. Some of the DBs used are comprehensive, including protein sequences derived from complete and partial genomes and are usually updated periodically; widely used examples include the Uniprot[10], RefSeq[11], InterPro[12], Pfam[13] and SEED[14] DBs. On the other hand, specialized DBs aim at curating the entries to include only protein sequences related to specific functions or protein families of interest, e.g., CARD for antibiotic resistance proteins [15]. The search tools used against the DBs are variable, and in some cases optional, being BLAST[16], DIAMOND [17], SWORD [18] or HMMER [19] the most used.

The current annotation pipelines vary in complexity, comprehensiveness, speed, scalability, requirements and outputs. For example, the NCBI Prokaryotic Genome Annotation Pipeline (PGAP)[20], Prokka[21], RAST[22], and DRAM [23] start from genomic contigs and predict genes and proteins, tRNAs, rRNAs, and perform functional annotation of the predicted proteins. Others such as InterProScan[24], eggNOG-Mapper[8,25] and MicrobeAnnotator[26] start from already predicted protein sequences. The RAST service requires an account and email, with a turn-around time measured in less than a day. It's almost the same with PGAP, a web server that provides annotation results in several days [22]. On the other hand, Prokka, MicrobeAnnotator, DRAM, and eggNOG-Mapper can be run in a desktop computer but they have limitations. Prokka relies on a very short database (DB) and generates a constrained number of annotated proteins. At the opposite, MicrobeAnnotator in standard mode demands the download of a huge DB (250-300 GB) which requires a lot of disk space and fast internet connection. eggNOG-Mapper is circumscribed to the eggNOG DB[27]and fails to combine different search algorithms (e.g. BLAST and HMMER) in the same run. Finally, DRAM is designed to be run on high-performance computers.

As a result of the pros and cons of the above-mentioned tools, it is necessary to develop an annotation pipeline that simultaneously accounts for the following: 1) resolve trait prediction from vast amounts of genomic content; 2) combine diverse databases and algorithms in the same run; 3) be able to launch from a desktop or laptop; 4) be easy to download. Additionally, the software should produce highly informative annotation tables for constructing metabolic pathways, comparing orthologues, and screening single traits. We present Arche, a bash-based command-line tool designed for the coordinated and optimized annotation of prokaryotic contigs using a suite of existing software tools. With Arche, several genomes can be annotated simultaneously or the annotation of an individual genome can be sped up by taking advantage of multiple processing cores.

2. IMPLEMENTATION

a. Input

Arche expects previously assembled genomic or metagenomic DNA sequences in FASTA format. This sequence file is the only mandatory parameter to the software.

b. Annotation process

Before functional annotation can be made, coding sequences (CDS), ribosomal RNA genes (rRNAs), and transfer RNA genes (tRNAs) must be discovered inside the contigs. Arche relies on external prediction tools to identify those features within the contigs (Fig. 1). These tools are:

(1) barrnap (https://vicbioinformatics.com/software.barrnap.shtml) for rRNAs.
(2) tRNAscan-SE 2.0 [28] for tRNAs.

(3) GeneMarkS-2 [29] for coding sequences (CDS) in genomes.

(4) MetaProdigal[30] for coding sequences (CDS) in metagenomes.

Once the coordinates of candidate gene sequences have been identified, a description is required of the putative gene product. The traditional method of predicting a gene's function is to compare its sequence against a set of sequences (database) with already known functions. Then, the CDS is labeled with the name and function of the closest significant match. Arche uses this method, but in a hierarchical manner similar to Prokka[21]; it starts making comparisons by heuristic or Smith-Waterman algorithms over in-house made databases, then uses an hmm algorithm over databases of protein families, and finally the same hmm algorithm over a domain-specific database. In each step, those sequences that didn’t matched any homolog in the DB pass to the next step. In summary Arche queries the genomic CDS over the following series of DBs:

(1) A set of sequences picked from the UniProtKB DB and clustered in new redesigned DBs (NRDBs). These sequences have an ID number that is linked to functional information (description, name, alternative name, E.C. number, etc). Four optional NRDBs are provided by Arche; three of them have the ability to bias the annotation of one of three functional codes—Kegg Orthologues (KO[9]), eggNOG orthologues (eggNOG[27]), or Enzyme Commission numbers (E.C.[31]). The fourth NRDB is built from a trimmed Swiss-Prot DB, and enable the annotation of any of the three aforementioned codes with no bias. The search is made by a heuristic or Smith-Waterman algorithm with a default e-value threshold of 10^{-8}.

(2) A series of HMM profile DBs (HMMDBs) in this order: a prokaryotic version of KOfam[32], NCBIFAMs [33], TIGRFAMs [34] and Pfam[13]. The hmm algorithm of HMMER3 [19] is used for the search. By default, gathering score thresholds (implemented with the --cut_ga option) are used during the search over the HMMDBs except KOfam. In the case of KOfam search, the pre-computed adaptive score

thresholds[32] were used (incorporated in the hmm file and implemented with the --cut_ga option).

(3) If no matches can be found, the CDS is labelled as 'hypothetical protein'.

To provide greater flexibility to the user in the querying over the NRDBs clusters, Arche supports the use of three different popular search tools, i.e., BLASTp[16], DIAMOND[17] and SSEARCH[35]. The user can also modify the filtering thresholds used to select the best match found in the DBs (i.e., e-value, or query cover). Finally, multiprocessing is also supported with the -t (threads) option.

There are two modes of running Arche, "standard" and "heavy". The "standard" mode uses a reduced NRDB (based on a trimmed Swiss-Prot DB) that annotates E.C., KO or eggNOG codes, before starting using HMMDBs. The "heavy" mode has three options; it performs almost the same pattern as the "standard" scanning but replaces the Swiss-Prot DB with an NRDB that bias the annotation of E.C., KO or eggNOG codes, depending on the option provided by the user. This NRDB is subdivided into seven sequence clusters (Fig. 1). At the top of the scanning are those clusters of sequences that have the functional codes prioritized (e.g. KO), and at the bottom are the remaining clusters. The scanning will proceed through all seven clusters before starting with HMMDBs. For example, if the priority code is E.C., the clusters will be searched in this order: KO + eggNOG + E.C. (cluster 1), KO + EC (cluster 2), eggNOG + EC (cluster 3), EC (cluster 4), KO + eggNOG (cluster 5), KO (cluster 6), and eggNOG (cluster 7), after which HMMDBs will begin to run. If the code prioritized is KO, then the search will be as follows: KO + eggNOG + E.C. (cluster 1), KO + eggNOG (cluster 2), KO + EC (cluster 3), KO (cluster 4), KOfam(HMMDB), eggNOG + E.C (cluster 5), eggNOG (cluster 6), and EC (cluster 7). In the last example it's noticeable that KOfam scanning begins after the last cluster that contains KO (Fig. 1), thus truncating cluster scanning. Once KOfam search is over, truncated cluster scanning will be resumed, and then HMMDBs will run (except KOfam). This procedure is intended to maximize the annotation of KOs.

At the end of the annotation, query sequences are labeled with the database-specified code of the matching sequence. This code will be used by the pipeline to retrieve the metadata from each of mapping tables implemented in Arche (see Fig. 2). This final table will summarize all the functional information (names, KO identifiers, E.C. numbers, etc.) provided by the different DBs queried. However, some data is not linked directly. For example, the Uniprot-associated KO identifiers were obtained through the conversion of the KEGG identifiers (provided by the Uniprot database) to KO identifiers via script. This procedure involves the downloading of the KEGG entry and the finding of the associated KO (Fig. 2). Moreover, the E.C. numbers associated to the Pfam DB were obtained from the ECDomainMiner approach [36]. At the end, all annotated CDS will be grouped in a single table per genome, which includes all metadata associated with each best match.

c. Output

Arche produces a single main output folder per run, which contains several files described in Table 1.

3. RESULTS

3.1. Computing Requirements of Arche compared to other tools.

We compared Arche to other popular genomic annotation pipelines, including Prokka[21], RAST[22], eggNOG-Mapper[25], and MicrobeAnnotator light mode[26]; this last one can recover similar metabolic reconstructions to those obtained using MicrobeAnnotator standard mode.

The first comparison was in terms of download size of the required DBs to perform an annotation. These DBs include FASTA protein entries, HMM profiles, and mapping files depending on the tool. Table 2 shows that Arche (13 Gb) and eggNOG-Mapper (6,4 Gb) required downloads are bigger than the other tools. Arche’s download size is mostly driven by the inclusion of the NRDBs, four different HMMDBs, and their respective mapping tables.

Those DBs are already formatted for all Arche's modes and searching tools, and suitable for the annotation of both bacterial and archaeal genomes, and metagenomes. The Arche's NRDBs include only those sequences associated with E.C./KO/eggNOG data while most of the sequences and map files used by the other programs have not such data associated.

More important than the download size, we compared each tool's annotation performance in terms of speed, number of proteins annotated, and level of consistency between tools. For this, we annotated the genome of the well-known *Escherichia coli* K12 strain (Table S1) with all the aforementioned tools including in the testing two modes of Arche: Arche "standard" mode (AS) and Arche "kegg" mode (AK). Of the tools tested, Prokka and AS were the fastest, taking 122 and 378 seconds to annotate the *E. coli* genome, respectively (Table 2). These were followed by MicrobeAnnotator and eggNOG-Mapper which took 454 s and 1587 s to annotate the genome. In our tests and depending on the server load, the web version of RASTtk can take up to 5 h. Finally, AK required significantly more time to fully annotate the genome (2936 s).

Annotation completeness of Arche compared to other tools

In general, all tools annotated 92-99% of the CDSs of the genome of *E. coli*, except for Prokka, which annotated 82% (Table 2). Overall, AS and AK had high percentages of annotation, with 97% and 99% of CDSs annotated respectively. MicrobeAnnotator also gave good results in terms of CDSs annotated, with the 98% being classified with a function.

Out of all the tools tested, RAST had the lowest number of functional identifiers, including Enzyme Commission numbers which accounts for the 28% of the CDS of the *E. coli*'s genome (Table 2). On the other hand, eggNOG-mapper and Prokka included E.C. numbers in 31% and 37% of the CDS, respectively. At the top of performances are AS (40%), MicrobeAnnotator (54%), and AK (63%). In regards to KO and eggNOG ortholog DBs, RAST doesn't provide such data in the annotation. The remaining methods consistently included identifiers from such DBs to complement the text-based annotation descriptions. 69%, 75%, 76%, and 86% of the CDS include KO identifiers according to MicrobeAnnotator, AS, eggnog-mapper and AK results respectively. Egg-NOG identifiers are present in 67% of CDS according to Prokka annotation, 75% of CDS in AS results, 97% in AK results, and 100% of CDS in eggNOG-mapper annotations. Although AK run times are longer than those for the other tools (Table 2), the output is rich in annotated CDSs (Table 2) with both names and functional codes. Moreover, if the DIAMOND option in AK is used, the runtimes can be reduced, without a significant loss in annotation performance (data not shown).

We evaluated the quality of the annotation for each method through a comparison against the reference annotation of the *E. coli* K12's genome (NCBI NC_000913.3). We measured the number of exact coincidences of names in the first 100 CDS of the reference (Fig. 3, Table S1). We observe that AS has the same trend of Prokka, eggNOG-mapper, and RAST. On the other hand, AK have less than a half of the coincidences of AS (37 coincidences vs 81). This difference can be explained by the usage of a short curated protein DB (Swiss-Prot) by AS over a highly studied genome such as that of E. coli *K12*, whose majority of products have been characterized at experimental level. On the other hand, AK uses a very exhaustive DB made of unreviewed entries (TrEMBL) which could be more fit to annotate poorly characterized genomes.

3.2. Quantitative and qualitative advantages of Arche's annotations over the other tools.

Arche presents a set of exclusive tools and procedures, while some of them are shared with the other annotators (Fig. 4). Theoretically, Arche have advantage in regards to the number of annotation steps, procedure options, or quality of some steps. For example, Arche is a complete genomic annotation pipeline since it can accept as input a genome (complete or fragmented) or even a metagenome. These contigs are going to be undergo structural annotation through the appropriate tools for the prediction of RNAs and proteins. MicrobeAnnotator lacks this option and depends on the user directly providing the already predicted protein fasta files. eggNOG-Mapper, on the other hand, does not map RNAs and performs protein gene calling only optionally. Regarding to the programs used for structural annotation, both Arche and Prokka use

Barrnap (https://github.com/tseemann/barrnap) for the prediction of rRNAs. However, in the case of tRNAs, Arche exploits tRNAscan-SE 2.0[28], a more updated and improved program than ARAGORN[37] and tRNAscan-SE 1.0[22,38] implemented in Prokka and RAST respectively. With regards to the protein gene calling in genomes, Arche implements the amazing GeneMarkS-2[29] tool which has been demonstrated to give better results than Prodigal[6] (used by Prokka and optionally by eggNOG-Mapper) and Glimmer[39] (used by RAST). Furthermore GeneMarkS-2's output classifies coding regions in 'native' for species-specific genes and 'atypical' for harder-to-detect genes likely (horizontally transferred)[29]. In the case of metagenome annotation, both Arche and Prokka implement the MetaProdigal tool[30] which has been proven to be one of the best algorithms for metagenomic gene prediction. On the other hand, RAST has no option for metagenomic analysis at all.

When it comes to DBs used and search procedures, Arche benefits from summarizing other bigger DBs. The pipeline has four optional new redesigned DBs (NRDBs) made of picked sequences from the UniProtKB DB, and four different HMM profile DBs (HMMDBs) which are mandatory. All these DBs are run in a hierarchical strategy: first a search by sequence alignment (e. g. BLAST) is performed over a series DBs whose sequences have functional codes related, and then HMM searches (by using HMMER) are applied on the remaining sequences. This is based on the concept that HMM profile comparisons are more appropriate for detecting remote homologs[40].

Prokka uses the same logic although with less and shorter DBs. MicrobeAnnotator light mode reverses the procedure, launching an HMM search against the KOfam DB during the first step, and then applying a sequence alignment search against Swiss-Prot over the remaining sequences. eggNOG-Mapper fails to combine alignment search methods with HMM searches (they exist as alternative procedures), and RAST doesn't have any HMM search at all. When we consider the search tools used during the sequence alignment step(s), Arche has some advances. It’s possible to choose between both BLASTp or DIAMOND as the main heuristic algorithms, or the SSEARCH implementation of the S-W algorithm. The options provided by

MicrobeAnnotator are similar, although the SWORD tool replaces SSEARCH. However, in contrast to Arche, MicrobeAnnotator formats DBs according one of the three algorithms (e.g., BLAST), which would require a new DB download in case the user wants to run the annotation with another search tool (e.g., DIAMOND). eggNOG-Mapper has DIAMOND and MMseqs2 as alternatives, having MMseqs2 almost a similar sensitivity than BLAST, but still less at the end. Finally, both Prokka and RAST uses only BLAST in this step which makes them the least flexible of all the tools tested.

Another Arche's advantage to the user is the retrieval of additional text-based information from the NRDBs and KOfam, like the locus name(s), or the alternative name(s) of a protein and text about its function whenever available (Fig. 2, 4). This gives the user an extra benefit in case of looking for particular features of interest using text search tools.

4. DISCUSSION

We provide a new software tool ‘‘Arche’’ a bash-based command-line tool for annotation of prokaryotic genomes that provides a hierarchical workflow and incorporates state-of-the-art programs for both tRNA and protein prediction. The software demonstrated an innovative segmentation of steps towards a functional-oriented annotation of genes, and organized output of data within one tool. Overall, Arche provides an automated end-to-end solution for the entire annotation of prokaryotic contigs that is both flexible and suitable for extracting the kind of information that the user needs for a chosen downstream analysis. At present, only a small group of developed software produced sufficient quantity and quality of functional codes to allow adequate metabolic reconstruction. Some of these tools like DRAM demands high-performance computers; other require the download of huge datasets. For example, MicrobeAnnotator standard mode requires the download 200-300 Gb of DB only for a simple BLAST analysis. Some of the download blocks are very heavy like the TrEMBL dat files (>150 Gb), thus requiring a fast and uninterrupted connection to be downloaded successfully. These abovementioned tools also rely on DBs which are full of redundant, poorly characterized sequences, a fact that penalize data retrieval. The major advantages of excluding such useless sequences from a DB is that it can provide a platform to analyze the type of data that the user is more interested in. Thus, by combining the use of functional-oriented DBs and the proper methodology for genomic annotation, we created a universal tool that can potentially be utilized across different genomic sources (bacteria, archaea and metagenomes) and can annotate functional data in variety of flavors (E.C., KO, and eggNOG numbers/identifiers). We validated this last point by training our program with the genome of *E. coli* K12, where Arche showed very good results among all tools tested (Fig. 3).

Very recently, different annotation approaches were adopted for genomic and metagenomic analysis by our group, including the annotation of metagenomic reads through custom gene-based DBs[41], Prokka (with custom DBs) annotation of genomic contigs[42], and HMMER screening using specific gene profiles[43]. All our previous analysis lacked of a fast, standardized pipeline capable of giving rich outputs of useful metabolic data. That would require the right DBs combined with a carefully picking of all available functional codes (E.C., KO, eggNOG) that such DBs could provide. Arche overcomes the previous limitations by the implementation of NRBDs along with their associated mapping tables (Figs. 1, 2) to provide a more robust and high-throughput analysis of genomes and metagenomes.

Our genomic pipeline incorporates multiple and optional search algorithms, becoming an update of the hierarchical procedures used by Prokka and MicrobeAnnotator during searching steps. In addition, it employs GeneMarkS-2 to fundamentally learn of the horizontally transferred features across the genome[29].All these traits are also advances in regards to eggNOG-Mapper last version [25], which either can’t combine alignment search methods with HMM searches (mandatory in Arche pipeline) and doesn’t annotate RNAs. Arche also has the singularity that it powers the text-based analysis of the annotated features, as the output table can hold a bunch of description for a singular feature in the fields of “alternative names” and “function”. Because of these features and the other aforementioned, the Arche pipeline can become a good alternative to the remaining tested annotation packages (Figs. 3, 4).

In sum, the Arche bash-based command-line tool for genomic annotation offers an automated, competitive, and flexible workflow that can yield high-quality results for further metabolic studies.

5. METHOD DETAILS

The annotation tools tested include Prokka v1.14.6, RAST v2.0, eggNOG-Mapper v2.1.3, MicrobeAnnotator light v.2.0.5, and Arche v.1.0.1. All the essays were run with an E-value threshold of 10^{-8} and 20 threads, using an Intel(R) Core(TM) i9-10850K computer with Ubuntu v. 19.04. Both Arche and MicrobeAnnotator tools were run with BLAST search option. A fasta file with sequences of proteins predicted by GeneMarkS-2 MicrobeAnnotator and eggnog-mapper.

To test how comprehensive and consistent the annotations of each tool are, we quantified the number CDSs having attached a name denoting some functionality and functional identifiers (e. g. E.C. numbers) in the output. We considered a protein annotated if it had at least one match in any of its DBs, which would transfer both a functional name (e. g. “DNA polymerase”) and a functional ID/code/number (E.C., eggNOG and/or KO) to the queried CDS. Thus, all uninformative annotations like those labeled as “hypothetical protein”, “putative protein”, “uncharacterized protein”, “no match found”, “non supervised orthologous group”, “family of unknown function”, “domain of unknown function”, “protein of unknown function”, “domain protein”, “uncharacterized conserved protein”, or “DUF protein” with any clue of function, additional name or description will be considered as unannotated or nulls annotations. In the case of eggNOG-mapper a protein is considered annotated when the match had transferred a name that denotes some function, description or a KO identifier (e.g., K02313) to the queried CDS; in case that the annotation includes one of the uninformative labels mentioned before and doesn’t include additional functional information it will be discarded as unannotated or null annotation.

6. RESOURSE AVAILABILITY

Software code generated in this study and the manuals have been deposited to Github https://github.com/gundizalv/Arche. The full software with the all the preformatted DBs required for its proper functioning are available for download at the following link https://drive.google.com/file/d/1x9caXGPpYXCHUoodOdnuJI0tCDe9qtGG/view?usp=sharing. The following table describes the DBs used, their sources, and respective modifications:

Database(s)	Link of download	Modifications/observations
NRDBs	https://ftp.uniprot.org/pub/databases/uniprot/current_release/knowledgebase/complete/	All sequences whitout an associated KEGG, eggNOG or E.C. identifier were filtered out from the databases.
NCBIFAMs and TIGRFAMS	https://ftp.ncbi.nlm.nih.gov/hmm/current/	Both the NCBIFAMs and TIGRFAMS hmm profiles have been separated and merged into two different hmm files, so that both DBs can be scanned in separate stages.
Pfam	https://ftp.ebi.ac.uk/pub/databases/Pfam/current_release /	The hmm profiles were merged in one single hmm file.
KOfam	https://www.genome.jp/ftp/db/kofam/	The prokaryotic hmm profiles were split out from the main DB and merged in one hmm file which stands up for the KOfam DB used in ARCHE's workflow

7. ACKNOWLEDGMENTS

The authors acknowledge the generous financial support by PICT RAICES 2019–3216 projects. VHA as a Georg Forster Experienced Researcher acknowledges the Alexander von Humboldt Foundation, which has also financially supported this project. VHA is a staff researcher from the National Research Council (CONICET) in Argentina. DA is a postdoctoral researcher at the Center for Electron Microscopy (CIME). This manuscript has been released as a Pre-Print at bioRxiv.

8. AUTHOR'S CONTRIBUTION

DGAR designed the tool, wrote the code, performed the analyses, and wrote the manuscript, VHA provided the hardware for the analysis, and wrote the manuscript. All authors read and approved the final manuscript.

9. REFERENCES

1. Christoffels A, van Heusden P. Genome annotation: perspective from bacterial genomes. Encycl. Bioinforma. Comput. Biol. 2019; 152

2. Tsementzi D, Wu J, Deutsch S, et al. SAR11 bacteria linked to ocean anoxia and nitrogen loss. Nature 2016; 536:179–183

3. D. YN, Jacobo de la C-Z, H. RG, et al. Large-Scale Metagenome Assembly Reveals Novel Animal-Associated Microbial Genomes, Biosynthetic Gene Clusters, and Other Genetic Diversity. mSystems 2020; 5:e01045-20

4. Richardson EJ, Watson M. The automatic annotation of bacterial genomes. Brief. Bioinform. 2013; 14:1–12

5. Zhu W, Lomsadze A, Borodovsky M. Ab initio gene identification in metagenomic sequences . Nucleic Acids Res. 2010; 38:e132–e132

6. Hyatt D, Chen G-L, Locascio PF, et al. Prodigal: prokaryotic gene recognition and translation initiation site identification. BMC Bioinformatics 2010; 11:119

7. Noguchi H, Taniguchi T, Itoh T. MetaGeneAnnotator: Detecting Species-Specific Patterns of Ribosomal Binding Site for Precise Gene Prediction in Anonymous Prokaryotic and Phage

Genomes. DNA Res. 2008; 15:387–396

8. Huerta-Cepas J, Forslund K, Coelho LP, et al. Fast Genome-Wide Functional Annotation through Orthology Assignment by eggNOG-Mapper. Mol. Biol. Evol. 2017; 34:2115–2122

9. Kanehisa M, Sato Y, Kawashima M, et al. KEGG as a reference resource for gene and protein annotation. Nucleic Acids Res. 2016; 44:D457–D462

10. UniProt Consortium T. UniProt: the universal protein knowledgebase. Nucleic Acids Res. 2018; 46:2699

11. Haft DH, DiCuccio M, Badretdin A, et al. RefSeq: an update on prokaryotic genome annotation and curation. Nucleic Acids Res. 2018; 46:D851–D860

12. Mitchell AL, Attwood TK, Babbitt PC, et al. InterPro in 2019: improving coverage, classification and access to protein sequence annotations. Nucleic Acids Res. 2019; 47:D351–D360

13. El-Gebali S, Mistry J, Bateman A, et al. The Pfam protein families database in 2019. Nucleic Acids Res. 2019; 47:D427–D432

14. Overbeek R, Begley T, Butler RM, et al. The Subsystems Approach to Genome Annotation and its Use in the Project to Annotate 1000 Genomes. Nucleic Acids Res. 2005; 33:5691–5702

15. G. MA, Nicholas W, Fazmin N, et al. The Comprehensive Antibiotic Resistance Database. Antimicrob. Agents Chemother. 2013; 57:3348–3357

16. Camacho C, Coulouris G, Avagyan V, et al. BLAST+: architecture and applications. BMC Bioinformatics 2009; 10:421

17. Buchfink B, Xie C, Huson DH. Fast and sensitive protein alignment using DIAMOND. Nat. Methods 2014; 12:59

18. Vaser R, Pavlović D, Šikić M. SWORD—a highly efficient protein database search. Bioinformatics 2016; 32:i680–i684

19. Eddy SR. Accelerated Profile HMM Searches. PLOS Comput. Biol. 2011; 7:e1002195

20. Tatusova T, DiCuccio M, Badretdin A, et al. NCBI prokaryotic genome annotation pipeline. Nucleic Acids Res. 2016; 44:6614–6624

21. Seemann T. Prokka: rapid prokaryotic genome annotation. Bioinformatics 2014; 30:2068–2069

22. Aziz RK, Bartels D, Best AA, et al. The RAST Server: Rapid Annotations using Subsystems Technology. BMC Genomics 2008; 9:75

23. Shaffer M, Borton MA, McGivern BB, et al. DRAM for distilling microbial metabolism to automate the curation of microbiome function. Nucleic Acids Res. 2020; 48:8883–8900

24. Quevillon E, Silventoinen V, Pillai S, et al. InterProScan: protein domains identifier. Nucleic Acids Res. 2005; 33:W116–W120

25. Cantalapiedra CP, Hern□andez-Plaza A, Letunic I, et al. eggNOG-mapper v2: Functional Annotation, Orthology Assignments, and Domain Prediction at the Metagenomic Scale. Mol. Biol. Evol. 2021; 38:5825–5829

26. Ruiz-Perez CA, Conrad RE, Konstantinidis KT. MicrobeAnnotator: a user-friendly, comprehensive functional annotation pipeline for microbial genomes. BMC Bioinformatics 2021; 22:1–16

27. Huerta-Cepas J, Szklarczyk D, Heller D, et al. eggNOG 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. Nucleic Acids Res. 2019; 47:D309–D314

28. Chan PP, Lin BY, Mak AJ, et al. TRNAscan-SE 2.0: Improved detection and functional classification of transfer RNA genes. Nucleic Acids Res. 2021; 49:9077–9096

29. Lomsadze A, Gemayel K, Tang S, et al. Modeling leaderless transcription and atypical genes results in more accurate gene prediction in prokaryotes. Genome Res. 2018; 28:1079–1089

30. Hyatt D, LoCascio PF, Hauser LJ, et al. Gene and translation initiation site prediction in metagenomic sequences. Bioinformatics 2012; 28:2223–2230

31. Bairoch A. The ENZYME database in 2000. Nucleic Acids Res. 2000; 28:304–305

32. Aramaki T, Blanc-Mathieu R, Endo H, et al. KofamKOALA: KEGG Ortholog assignment based on profile HMM and adaptive score threshold. Bioinformatics 2020; 36:2251–2252

33. Li W, O'Neill KR, Haft DH, et al. RefSeq: expanding the Prokaryotic Genome Annotation Pipeline reach with protein family model curation. Nucleic Acids Res. 2021; 49:D1020–D1028

34. Haft DH, Selengut JD, Richter RA, et al. TIGRFAMs and Genome Properties in 2013. Nucleic Acids Res. 2013; 41:D387–D395

35. Lipman DJ, Pearson WR. Rapid and Sensitive Protein Similarity Searches. Science (80-.). 1985; 227:1435–1441

36. Alborzi SZ, Devignes M-D, Ritchie DW. ECDomainMiner: discovering hidden associations between enzyme commission numbers and Pfam domains. BMC Bioinformatics 2017; 18:107

37. Laslett D, Canback B. ARAGORN, a program to detect tRNA genes and tmRNA genes in nucleotide sequences. Nucleic Acids Res. 2004; 32:11–16

38. Lowe TM, Eddy SR. tRNAscan-SE: A Program for Improved Detection of Transfer RNA Genes in Genomic Sequence. Nucleic Acids Res. 1997; 25:955–964

39. Delcher AL, Harmon D, Kasif S, et al. Improved microbial gene identification with GLIMMER. Nucleic Acids Res. 1999; 27:4636–4641

40. Madera M, Gough J. A comparison of profile hidden Markov model procedures for remote homology detection. Nucleic Acids Res. 2002; 30:4321–4328

41. Alonso-Reyes DG, Farias ME, Albarracín VH. Uncovering cryptochrome/photolyase gene diversity in aquatic microbiomes exposed to diverse UV-B regimes. Aquat. Microb. Ecol. 2020; 85:141–154

42. Alonso-Reyes DG, Galván FS, Portero LR, et al. Genomic insights into an andean multiresistant soil actinobacterium of biotechnological interest. World J. Microbiol. Biotechnol. 2021; 37:166

43. Alonso-Reyes DG, Galván S, Irazoqui JM, et al. Dissecting Light Sensing and Metabolic Pathways on the Millimeter Scale in High-Altitude Modern Stromatolites. Microb. Ecol. 2022;

Figure and Table legends.

Figure 1. Arche workflow. The RNA prediction stage (orange boxes) is optional (broken lines) and only applied to genomic inputs. GeneMarkS-2 and MetaProdigal are used to perform protein prediction (blue boxes) from assembled contigs in genomes and metagenomes respectively. During the first search stage (green boxes), BLASTp, DIAMOND, or SSEARCH can be used alternatively to align the input proteins over the four alternative NRBDs (purple containers). Each of those NRBDs are split in several clusters of sequences which have functional information associated (KO identifiers, eggNOG identifiers or Enzyme Commission numbers). Except the “standard” NRBD, each one prioritizes the annotation of a particular functional identifier. During the second search stage (grey box) the remanent unannotated proteins serve as input for HMMER which scan over several HMMBDs (cyan containers). “KO priority” NRBD is interrupted by KOfam DB (cyan container) after the fourth cluster, which means that all the input sequences unnanotated by the previous clusters would be scanned on KOfam DB using HMMER. After this, the workflow will resume the alignment with the remaining clusters of the “KO priority” NRBD. The final output of the workflow is a summary table with biological information associated to the matches.

Figure 2. Picture showing the annotation process. Once a group of hits from the search step(s) is obtained, their respective identifications are matched to the functional data saved in tables (colored containers). Each table gathers all the useful information provided by the original source (Uniprot, KOfam, etc.). Finally, all the query identifiers, the query matches, and the

corresponding related data will be resumed in one single table. Inside the rectangles, we observe the number of *Exiguobacteriumsibiricum* 255_15's CDSs which are hits in each step.

Figure 3. Number of coincidences of annotations in the first 100 CDS of the reference annotation of the *E. coli* K12's genome (NCBI NC_000913.3) vs other methods.

Figure 4. Set of shared and unshared tools and procedures of five different annotators, including Arche.

Table 1. Description of Arche output files.

Table 2. Comparison of the performance of several annotators, including Arche. The datasets analyzed include 15 *Exiguobacterium* genomes, 15 *Citrobacter* genomes, and one metagenome. Best performances are highlighted in green while the worst in light orange.

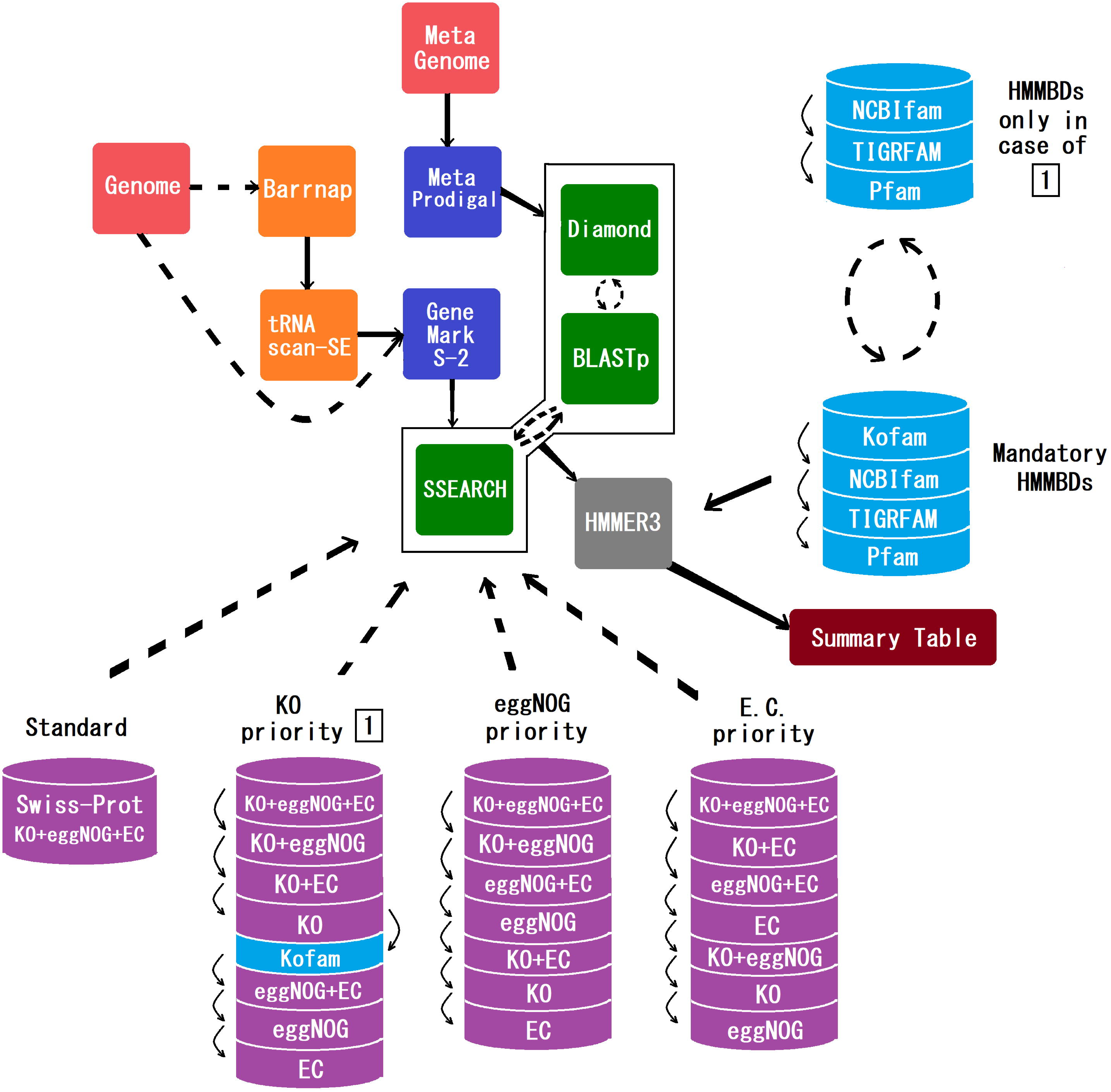
Table 1

File(s)	Description of file contents
rRNA.tsv	GFF v3 file containing rRNA annotations
rRNA.fna	FASTA file of all rRNA features
tRNA.tsv	Table with tRNA details (coordinates, isotype, anticodon, scores, etc)
tRNA.fna	FASTA file of all tRNA features
[]_struc_annot.fna	FASTA file of all genomic features (nucleotide)
[]_struc_annot.faa	FASTA file of translated coding genes (protein)
heuristic[]_out	Output matches of the search instance(s) performed with BLASTp, DIAMOND or SSEARCH36
heuristic[]_non_match.faa	FASTA file with the remaining non-matched sequences after the search instance(s) performed with BLASTp, DIAMOND or SSEARCH36
hmmscan_[]_out	HMMER3 output table of the search instance(s) performed against a specific hmmdb

[hmmdb]_non_match.faa	FASTA file with the remaining non-matched sequences after the search instance performed against a specific hmmdb
[]_omic_table.tbl	Feature table with fields separated by vertical bars
[]_omic_table.tsv	Feature table with tab-separated fields
arche_report	File which includes the parameters of the run and results

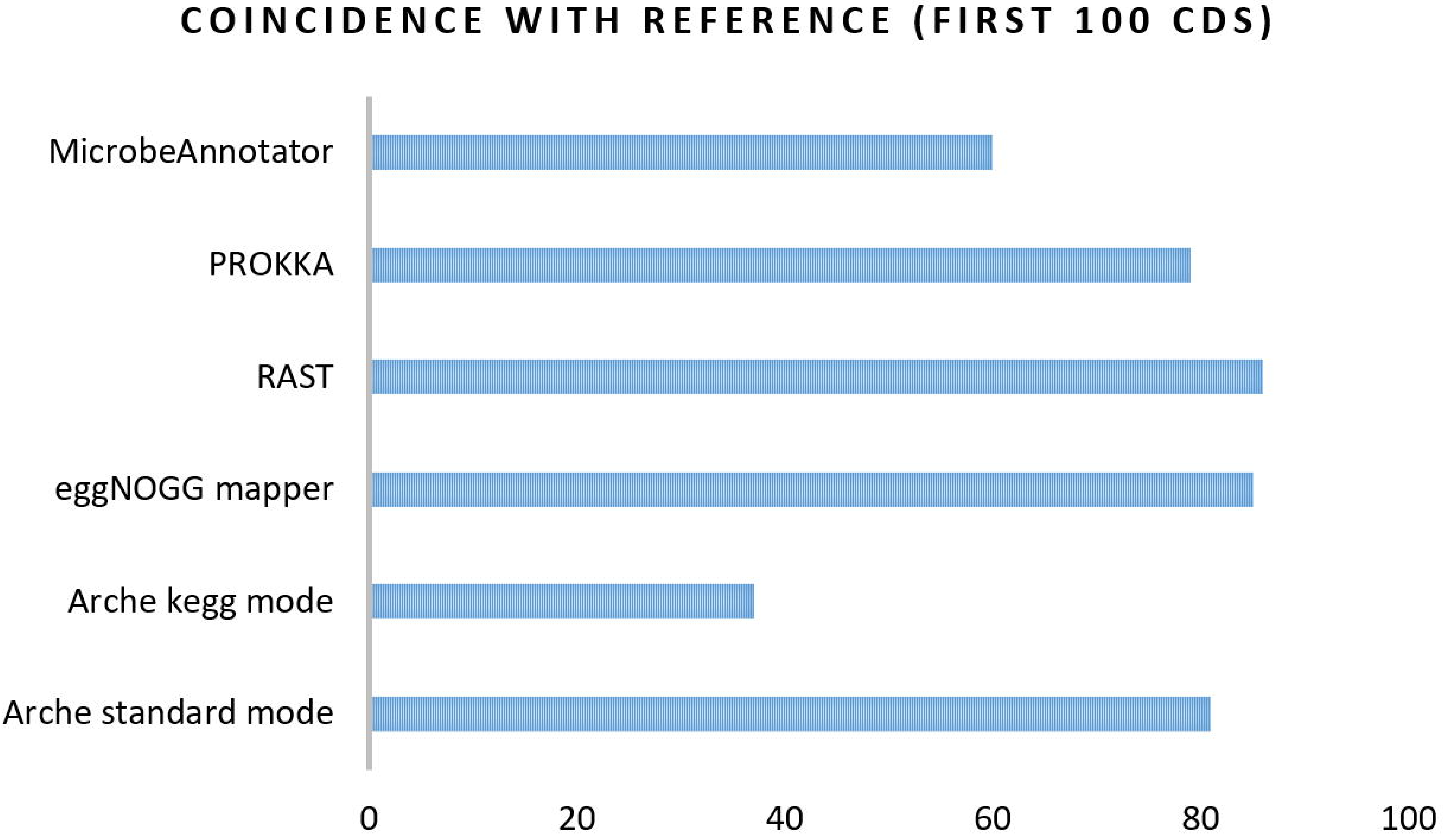
Table 2

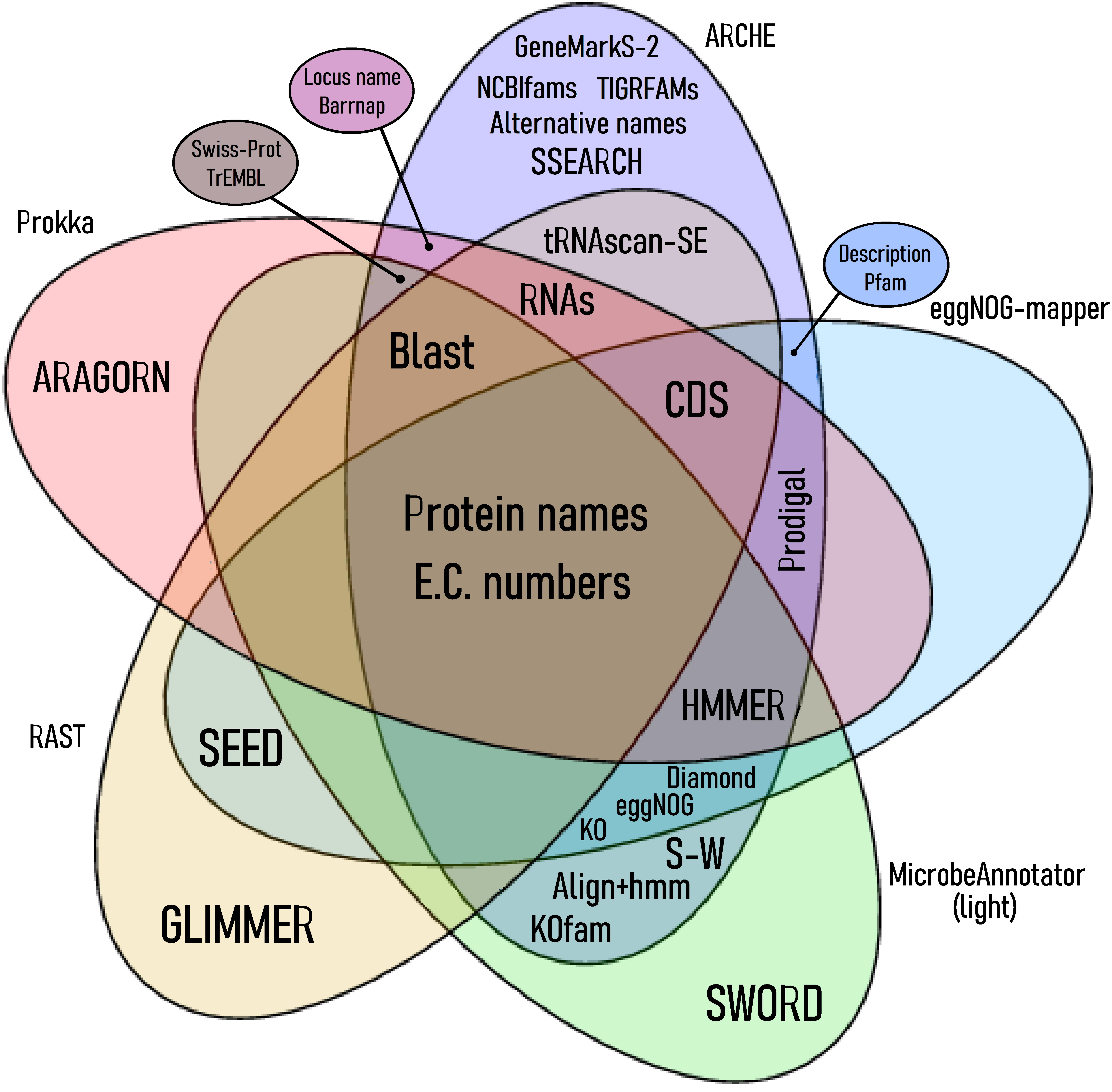
Method	CDS	Annotated proteins	E.C.	KO	eggNOG	Time (s)	Db download size (Gb)
			Escherichia coli K-12				
ARCHE (standard)	4269	4231	1720	3214	3218	378	13
ARCHE (kegg)	4269	4141	2728	3703	4173	2936	13
eggNOG-mapper	4269	3948	1361	3277	4269	1587	6,4
RAST	4513	4261	1275	0	0	0	0
MicrobeAnnotator (light)	4269	4217	2316	2971	0	454	2,11
Prokka	4308	3572	1633	0	2896	122	0


Meta Genome
Genome
Barrnap
Meta Prodigal
tRNA scan-SE
Gene Mark S-2
Diamond
BLASTp
SSEARCH
HMMER3
Summary Table
NCBIfam
TIGRFAM
Pfam
HMMBDs only in case of 1
Kofam
NCBIfam
TIGRFAM
Pfam
Mandatory HMMBDs
Standard
Swiss-Prot
KO+eggNOG+EC
KO priority 1
KO+eggNOG+EC
KO+eggNOG
KO+EC
KO
Kofam
eggNOG+EC
eggNOG
EC
eggNOG priority
KO+eggNOG+EC
KO+eggNOG
eggNOG+EC
eggNOG
KO+EC
KO
EC
E.C. priority
KO+eggNOG+EC
KO+EC
eggNOG+EC
EC
KO+eggNOG
KO
eggNOG

Optional NRDBs during search by sequence alignment step(s)

Uniprot entry
KEGG
abcd:A1234_56789
www.genome.jp/entry/abcd:A1234_56789
KO
3077
CDSs
Name
AltNam
LocNam
E.C.
eggNOG
Dscrp
Uniprot table
2076
KOfam database
Name
AltNam
E.C.
KO
KOfam table
237
NFAM database
Name
LocNam
E.C.
NFAM table
8
TIGRFAM database
Name
E.C.
TIGRFAM table
5
Pfam table
84
Name
E.C.
Pfam database
ECDomainMiner
List of Hits
Table with annotations
667
Hypotetical


COINCIDENCE WITH REFERENCE (FIRST 100 CDS)
MicrobeAnnotator
PROKKA
RAST
eggNOGG mapper
Arche kegg mode
Arche standard mode
0
20
40
60
80
100


ARCHE
GeneMarkS-2
NCBIfams
TIGRFAMs
Alternative names
SSEARCH
Locus name
Barrnap
Swiss-Prot
TrEMBL
Prokka
tRNAscan-SE
RNAs
Description
Pfam
eggNOG-mapper
ARAGORN
Blast
CDS
Prodigal
Protein names
E.C. numbers
HMMER
RAST
SEED
Diamond
eggNOG
KO
S-W
Align+hmm
KOfam
GLIMMER
MicrobeAnnotator
(light)
SWORD