Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Application Note

Sequence analysis

MMseqs2 desktop and local web server app for fast, interactive sequence searches

Mirdita M. [1], **Steinegger M.** [1,2,*] **and Söding J.** [1,*]

[1]Quantitative and Computational Biology Group, Max Planck Institute for Biophysical Chemistry, Göttingen, Germany and [2]Department of Chemistry, Seoul National University, Seoul, Korea.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Summary: The MMseqs2 desktop and web server app facilitates interactive sequence searches through custom protein sequence and profile databases on personal workstations. By eliminating MMseqs2's runtime overhead, we reduced response times to a few seconds at sensitivities close to BLAST.
Availability and implementation: The app is easy to install for non-experts. GPLv3-licensed code, prebuilt desktop app packages for Windows, macOS and Linux, Docker images for the web server application, and a demo web server are available at https://search.mmseqs.com.
Contact: martin.steinegger@mpibpc.mpg.de or soeding@mpibpc.mpg.de
Supplementary information: Supplementary data is available at *Bioinformatics* online.

1 Introduction

The most popular sequence similarity search tool, BLAST (Altschul *et al.*, 1990, 1997), has garnered ~7000 citations per year during the last 5 years, attesting to the unremitting importance of sequence searches for biology. This popularity may be largely owed to the excellent web services with short response times despite fast-growing databases provided by the NCBI/NIH, which requires a huge compute infrastructure. The distributed approach of running searches *locally* on personal computers or IT platforms of companies and research groups allows for custom databases, high availability, and protects sensitive data. But web server applications for local homology searches are slow as they mostly rely on BLAST (e.g. Deng *et al.* (2007); Priyam *et al.* (2015)). Here, we present an application software to search with protein and nucleotide sequences through custom protein sequence and profile databases using MMseqs2 (Steinegger and Söding, 2017), achieving response times of seconds instead of minutes at a similar sensitivity as BLAST.

2 Methods

Reduced runtime overhead. MMseqs2 owes its sensitivity and speed mainly to its prefiltering stage, which rejects ~99.99% of sequences. The prefilter uses a reverse k-mer index table for the target database and also requires matrices with similarity scores between 2-mers and between 3-mers to generate the lists of similar 7-mers (Steinegger and Söding, 2017). Reading in the index table and computing these matrices on-the-fly takes ~0.5 min of runtime overhead for each search. We reduced this to 0.05 s by (1) writing the index table, the matrices, and other precomputable data into a file if it does not yet exist, memory mapping the file to take advantage of the system page cache (for detailed memory requirements see Supplementary Materials), and (3) optimizing I/O operations.

Optimized sequence-to-profile search mode. The index table for profile databases stores, for each position in a profile, all k-mers with a profile similarity score above a threshold set by `-s`. The number of similar k-mers grows exponentially with k. To save memory, we chose a short $k = 5$ as default for this mode. We also added to MMseqs2 utilities for creating profiles from multiple sequence alignments (MSAs) and converting between profile formats.

Desktop and web server app. Based on the same code base, the application can be either deployed through Docker containers to be accessed through web browsers or packaged as a desktop GUI application with the Electron framework (electronjs.org). In either case, the backend part of the application provides a RESTful API and worker scheduling. The server supports protein, translated nucleotide and nucleotide sequence searches and iterative and reverse profile searches.
The application takes a list of either protein or nucleotide sequences in FASTA/FASTQ format as query input. To generate a target search database, the application takes a FASTA/FASTQ file for protein sequence searches or a STOCKHOLM multiple sequence alignment file for protein profile searches. Search results are shown with a customized feature-viewer (github.com/calipho-sib/feature-viewer) (Figure 1A) and can be downloaded in tabular BLAST format.

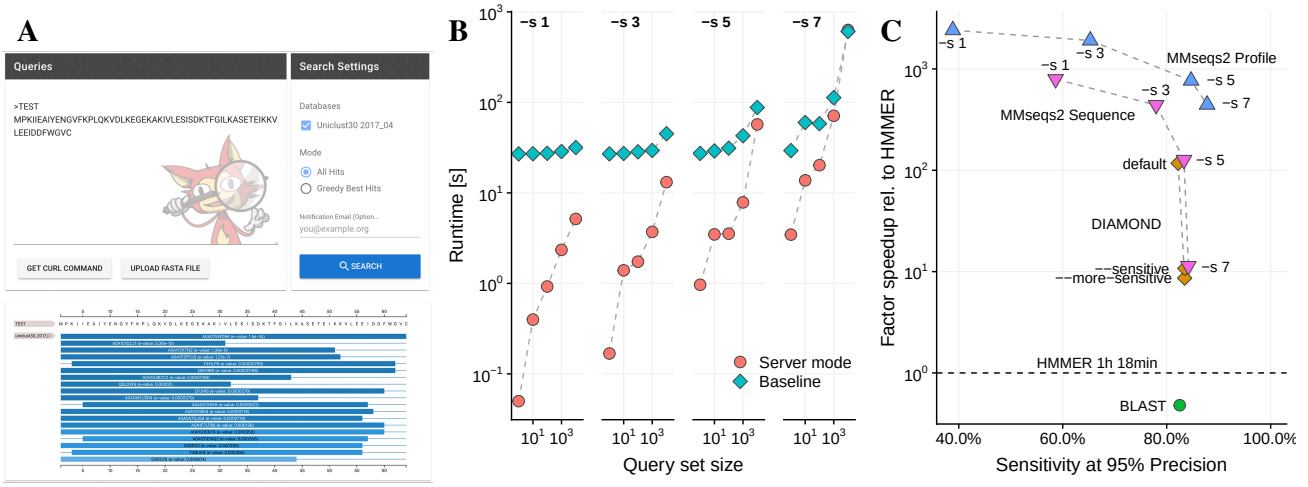

Fig. 1. (A) Screenshots of the search interface and result visualization. (B) Runtime of searches with the baseline MMseqs2 (teal) and the new server mode (red) at four sensitivity settings (`-s`). (C) Domain annotation: Speedup versus sensitivity at 95% precision for MMseqs2 (blue: sequence-profile search, magenta: sequence-sequence search; sensitivity settings: `-s 1, 3, 5, 7`), DIAMOND (brown; default, `--sensitive`, `--more-sensitive`) and BLAST (green). HMMER3 matches to Pfam domains are used as ground truth. The speed-ups exclude the times to format the databases.

3 Results

Figure 1B demonstrates the reduction of runtime overhead by comparing the runtimes of the MMseqs2 version without ("baseline") to the new version with precomputations and memory mapping ("server mode"). Runtimes refer to searches with amino acid query sets of 1, 10, 100, 1000, and 10 000 sequences of average length 350 (sampled from the Uniclust30 database) through the Uniclust30 2017_10 database (Mirdita *et al.*, 2017) with 13.5 million sequences, measured on a server with 2 Intel Xeon E5-2680 v4 CPUs with 14 cores each. The index table and matrix precomputation (~3 min 40 s) is not included in the runtimes.

To test the quality and speed of annotating Pfam domains on genes assembled from metagenomics data, we built a test set by sampling 100 000 full-length sequences longer than 150 residues from our Marine Eukaryotic Reference Catalogue (Steinegger *et al.*, 2018), clustering this set to 30 % maximum pairwise sequence identity with MMseqs2, and sampling 10 000 sequences from the redundancy-reduced set. We annotated these sequences with PfamA 31.0 domains (Finn *et al.*, 2014) using HMMER3 (Finn *et al.*, 2011).

We then compared how well the sequence-sequence searches of MMseqs2, BLAST, and DIAMOND (Buchfink *et al.*, 2015) and the sequence-to-profile searches of MMseqs2 could find the correct domain annotations. For the sequence-sequence search methods, we built a database from all sequences in `PfamA.full` MSAs and reported as E-value of a Pfam domain the E-value for the best-matching sequence from its MSA. We defined a search as true positive (TP) if the top match was annotated by HMMER3 with an E-value better than 10^{-3} and as false positive (FP) if the top match was not annotated with an HMMER3 E-value below 1. All other searches were considered ambiguous and ignored. For each method, we determined the E-value at which the precision TP/(TP+FP) is 95% and measured the sensitivity at that E-value.

As Figure 1C shows, MMseqs2 sequence-to-profile searches are ~30 times faster than sequence-sequence searches with DIAMOND, MMseqs2 and BLAST and ~300 times faster than HMMER3. MMseqs2 sequence-to-profile searches reach 87% relative sensitivity at 95% precision, making them an attractive alternative to HMMER3 when speed is critical.

4 Conclusion

The desktop and web server app for MMseqs2 performs fast sequence searches at unprecedented speed-to-sensitivity trade-off on local computers. 1000 queries take only a minute to search through 15 million sequences of the Uniclust30 database, much faster than NCBI's BLAST website. We hope the MMseqs2 app will also empower users unfamiliar with command line interfaces to perform fast and sensitive searches with their own sequence and profile databases.

Acknowledgements

We thank Yuna Kwon for crafting the "little Marv" mascot.

Funding

This work was supported by the European Research Council in the framework of its Horizon 2020 Framework Programme for Research and Innovation (grant "Virus-X", project no. 685778).

Conflict of Interest: none declared

References

Altschul, S.F. et al (1990). Basic local alignment search tool. *J. Mol. Biol.*, **215**(3), 403–410.

Altschul, S.F. et al (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**(17), 3389–3402.

Buchfink, B. et al (2015). Fast and sensitive protein alignment using DIAMOND. *Nat. Methods*, **12**(1), 59–60.

Deng, W. et al (2007). Viroblast: a stand-alone blast web server for flexible queries of multiple databases and user's datasets. *Bioinformatics*, **23**(17), 2334–2336.

Finn, R.D. et al (2011). HMMER web server: interactive sequence similarity searching. *Nucleic Acids Res.*, **39**, W29–37.

Finn, R.D. et al (2014). Pfam: the protein families database. *Nucleic Acids Res.*, **42**(D1), D222–D230.

Mirdita, M. et al (2017). Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Res.*, **45**(D1), D170–D176.

Priyam, A. et al (2015). Sequenceserver: a modern graphical user interface for custom BLAST databases. *bioRxiv*, page 033142.

Steinegger, M. and Söding, J. (2017). MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.*, **35**(11), 1026–1028.

Steinegger, M. et al (2018). Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *bioRxiv*.