G&I Genomics & Informatics

eISSN 2234-0742
*Genomics Inform* 2014;12(2):76-78
http://dx.doi.org/10.5808/GI.2014.12.2.76

APPLICATION NOTE

# SFannotation: A Simple and Fast Protein Function Annotation System

Dong Su Yu[1]*, Byung Kwon Kim[2]

[1]Korean BioInformation Center (KOBIC), Korea Research Institute of Bioscience and Biotechnology (KRIBB), Daejeon 305-806, Korea,
[2]BioNano Health Guard Research Center, Korea Research Institute of Bioscience and Biotechnology (KRIBB), Daejeon 305-806, Korea

Owing to the generation of vast amounts of sequencing data by using cost-effective, high-throughput sequencing technologies with improved computational approaches, many putative proteins have been discovered after assembly and structural annotation. Putative proteins are typically annotated using a functional annotation system that uses extant databases, but the expansive size of these databases often causes a bottleneck for rapid functional annotation. We developed SFannotation, a simple and fast functional annotation system that rapidly annotates putative proteins against four extant databases, Swiss-Prot, TIGRFAMs, Pfam, and the non-redundant sequence database, by using a best-hit approach with BLASTP and HMMSEARCH.

**Keywords:** bioinformatics, gene product, protein annotation

**Availability:** SFannotation system is available at https://code.google.com/p/axxa76/wiki/SFannotation.

## Introduction

Functional annotation of putative proteins is a fundamental and essential practice in the postgenomics era [1]; it allows us to analyze genomic and genetic features, such as physiological activity and metabolism, as well as to discover medically and industrially relevant enzymes. Since large numbers of putative proteins were discovered from a vast amount of sequencing data generated using high-throughput sequencing technologies, including those of the next and third generation, many automated functional annotation systems have contributed greatly to the annotation of them with minimal manual effort [2]. However, their runtime performance of functional annotation against large extant databases often causes a bottleneck, and especially, standalone tools, such as AutoFACT [3] and BLANNOTOR [4], demand high-performance hardware resources for fast annotation from users.

From the user's perspective, a web-based annotation server system would be a useful tool to bypass the demands of high-performance computer resources, and besides, they offer user-friendly interfaces. The RAST server system is particularly popular and can be used to rapidly annotate many microbial proteins against a specially curated subsystem database [5]. Web server systems, however, may be undesirable because of critical obstacles, such as the limitation of usable server resources, a long waiting time by many queries, a low-bandwidth network or unstable traffic flow associated with the upload of query data and download of outputs, and data security problems. Thus, some users prefer standalone systems to web-based systems in spite of the demand for high-performance resources. Although standalone and web-based systems have good and bad points, slow runtime performance in themselves cannot be avoided because of the exponential increase in database sizes, without controlling some aspect of the annotation workflow.

We developed SFannotation, which rapidly annotates putative proteins by using single or bidirectional best-hit approach with sequence-based methods—BLASTP [6] and

Received May 9, 2014; Revised May 21, 2014; Accepted May 23, 2014

***Corresponding author:** Tel: +82-42-879-8521, Fax: +82-42-879-8519, E-mail: axxa76@kribb.re.kr

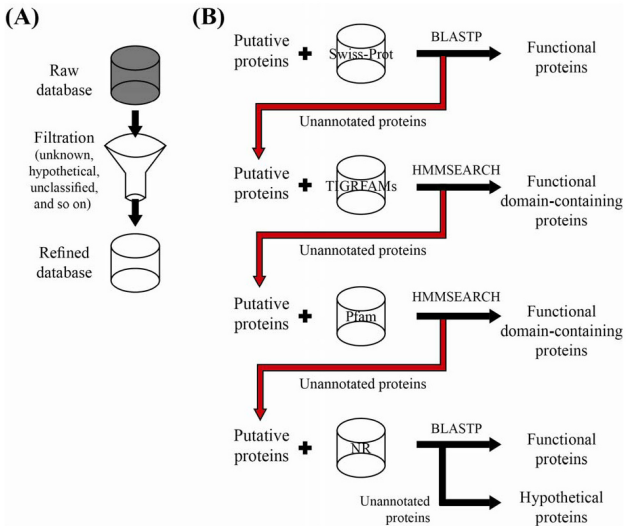


**Fig. 1.** Database filtration (A) and workflow of the SFannotation annotation system (B). Black arrows represent putative proteins that are annotated by the best-hit approach, and red arrows represent the conversion of unannotated proteins to query putative proteins to search homologs against other databases.

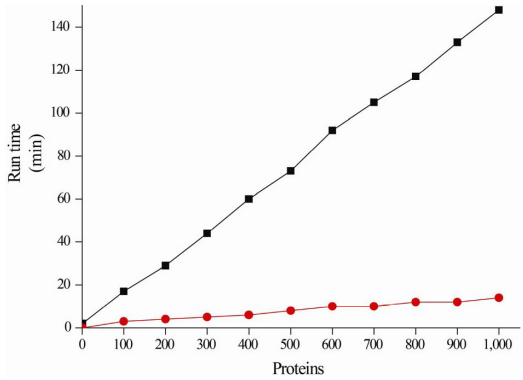


**Fig. 2.** Runtime of the SFannotation system (red) and a best-hit approach without the hierarchical SFannotation workflow (black). Randomly selected proteins from *Escherichia coli* MG 1655 (GenBank accession number: U00096) were tested using a 64-bit Linux system (Ubuntu) possessing 20 CPU threads.

HMMSEARCH [7]—against big extant databases: Swiss-Prot [8], TIGRFAMs [9], Pfam [10], and the non-redundant sequence database (NR) of NCBI [11]. As best-hit approaches, especially bidirectional best-hit [12], have been widely utilized in searching reliable homologous protein sequences, such as orthologs, as well as functional annotation systems [13-16], SFannotation can reliably annotate putative proteins. Remarkably, SFannotation can rapidly annotate proteins against large extant databases by our hierarchical workflow.

## Methods and Results

Before annotating putative proteins against Swiss-Prot, TIGRFAMs, Pfam, and the NR database, SFannotation filters out all proteins described in the databases by terms, such as "unknown," "hypothetical," "unclassified," "uncharacterized," "putative," "predicted," and "conserved" (Fig. 1A), because some putative proteins may be misannotated by their inclusion. Then, using BLASTP and HMMSEARCH, SFannotation searches homologous proteins and domains in each refined database using a default threshold ($\leq 10^{-5}$ E-value) and selects the highest-scoring homolog to annotate putative proteins as the best-hit approach, such as single best hit and bidirectional best hit [12, 16].

Putative proteins are hierarchically annotated using the following database priority: Swiss-Prot → TIGRFAMs → Pfam → NR, which is ordered according to their reliability (Fig. 1B). Once annotated, the putative proteins are no longer queried using homology searches against the other databases. For example, if a putative protein is annotated against Swiss-Prot, it is excluded from annotation against the other databases, while the remaining unannotated putative proteins continue to be annotated against the other databases. Therefore, the runtime performance can be reduced, because the number of unannotated putative proteins gradually decreases (Fig. 2).

## Implementation

SFannotation is written in Perl and bash shell and is implemented on a Linux/Unix system on which BLASTP and HMMSEARCH are able to function. SFannotation automatically annotates putative proteins with downloading of all four databases, as well as BLASTP and HMMSEARCH. SFannotation is implemented by a command line on the Linux/Unix system: "perl SFannotation --download --fasta <input fasta file> --speedup" (Supplementary Fig. 1).

## Supplementary material

Supplementary data including one figure can be found with this article online at http://www.genominfo.org/src/sm/gni-12-76-s001.pdf.

## Acknowledgments

We thank the members of the Korean BioInformation Center (KOBIC). This project was supported by a grant from "KRIBB Research Initiative Program" and the Korean Ministry of Science, ICT & Future Planning (MSIP) under grant numbers NRF-2010-0029345 and NRF-2011-0019745.

## References


1. Beckloff N, Starkenburg S, Freitas T, Chain P. Bacterial genome annotation. *Methods Mol Biol* 2012;881:471-503.
2. Radivojac P, Clark WT, Oron TR, Schnoes AM, Wittkop T, Sokolov A, *et al*. A large-scale evaluation of computational protein function prediction. *Nat Methods* 2013;10:221-227.
3. Koski LB, Gray MW, Lang BF, Burger G. AutoFACT: an automatic functional annotation and classification tool. *BMC Bioinformatics* 2005;6:151.
4. Kankainen M, Ojala T, Holm L. BLANNOTATOR: enhanced homology-based function prediction of bacterial proteins. *BMC Bioinformatics* 2012;13:33.
5. Overbeek R, Olson R, Pusch GD, Olsen GJ, Davis JJ, Disz T, *et al*. The SEED and the Rapid Annotation of microbial genomes using Subsystems Technology (RAST). *Nucleic Acids Res* 2014;42:D206-D214.
6. Camacho C, Coulouris G, Avagyan V, Ma N, Papadopoulos J, Bealer K, *et al*. BLAST+: architecture and applications. *BMC Bioinformatics* 2009;10:421.
7. Finn RD, Clements J, Eddy SR. HMMER web server: interactive sequence similarity searching. *Nucleic Acids Res* 2011; 39:W29-W37.
8. Kiefer F, Arnold K, Kunzli M, Bordoli L, Schwede T. The SWISS-MODEL Repository and associated resources. *Nucleic Acids Res* 2009;37:D387-D392.
9. Haft DH, Selengut JD, Richter RA, Harkins D, Basu MK, Beck E. TIGRFAMs and Genome Properties in 2013. *Nucleic Acids Res* 2013;41:D387-D395.
10. Punta M, Coggill PC, Eberhardt RY, Mistry J, Tate J, Boursnell C, *et al*. The Pfam protein families database. *Nucleic Acids Res* 2012;40:D290-D301.
11. Johnson M, Zaretskaya I, Raytselis Y, Merezhuk Y, McGinnis S, Madden TL. NCBI BLAST: a better web interface. *Nucleic Acids Res* 2008;36:W5-W9.
12. Overbeek R, Fonstein M, D'Souza M, Pusch GD, Maltsev N. The use of gene clusters to infer functional coupling. *Proc Natl Acad Sci U S A* 1999;96:2896-2901.
13. Li L, Stoeckert CJ Jr, Roos DS. OrthoMCL: identification of ortholog groups for eukaryotic genomes. *Genome Res* 2003;13: 2178-2189.
14. Ostlund G, Schmitt T, Forslund K, Köstler T, Messina DN, Roopra S, *et al*. InParanoid 7: new algorithms and tools for eukaryotic orthology analysis. *Nucleic Acids Res* 2010;38:D196-D203.
15. Richardson EJ, Watson M. The automatic annotation of bacterial genomes. *Brief Bioinform* 2013;14:1-12.
16. Moriya Y, Itoh M, Okuda S, Yoshizawa AC, Kanehisa M. KAAS: an automatic genome annotation and pathway reconstruction server. *Nucleic Acids Res* 2007;35:W182-W185.