Current Comparative Table (CCT) automates customized searches of dynamic biological databases

Benjamin R. Landsteiner, Michael R. Olson and Robert Rutherford*

Department of Biology, St Olaf College, 1520 St Olaf Avenue, Northfield, MN 55057, USA

Received February 14, 2005; Revised and Accepted March 24, 2005

ABSTRACT

The Current Comparative Table (CCT) software program enables working biologists to automate customized bioinformatics searches, typically of remote sequence or HMM (hidden Markov model) databases. CCT currently supports BLAST, hmmpfam and other programs useful for gene and ortholog identification. The software is web based, has a BioPerl core and can be used remotely via a browser or locally on Mac OS X or Linux machines. CCT is particularly useful to scientists who study large sets of molecules in today’s evolving information landscape because it color-codes all result files by age and highlights even tiny changes in sequence or annotation. By empowering non-bioinformaticians to automate custom searches and examine current results in context at a glance, CCT allows a remote database submission in the evening to influence the next morning’s bench experiment. A demonstration of CCT is available at http://orb.public.stolaf.edu/CCTdemo and the open source software is freely available from http://sourceforge.net/projects/orb-cct.

INTRODUCTION

A constant flood of new genomic information has brought a new age of discovery to biology. Unfortunately, this deluge of new data is rarely fully utilized, in part because bench scientists find it increasingly challenging to maintain a current, integrated picture of the latest data. This problem is particularly pronounced for scientists who study large numbers of potentially interesting molecules, a common result of microarray-based or proteome-based experiments. Regularly updating such data by hand can be extremely burdensome and is rarely done. As a result, many scientists work unaware of novel features of CCT include automated highlighting of even small changes in data files, a simple interface for scientists interested in multiple sequences and the ability to monitor any number of databases to which the user has access.

MATERIALS AND METHODS

CCT was developed on a Gateway E-6100 series computer running RedHat 9 Linux. The computer has a 3 GHz processor, 200 GB of hard drive space and 2 GB of RAM. Mac OS X compatibility was tested on a Dual 2 GHz G5 Tower with 1.5 GB of RAM, a 160 GB hard drive and Mac OS X 10.3.4. CCT is implemented in Perl and makes heavy use of the BioPerl toolkit (4). CCT is freely available and open source. A demonstration of CCT and an installation guide are available at http://orb.public.stolaf.edu/CCTdemo, and the software is freely available from http://sourceforge.net/projects/orb-cct.

*To whom correspondence should be addressed. Tel: +1 507 646 3804; Fax: +1 507 646 3968; Email: robruth@stolaf.edu

© The Author 2005. Published by Oxford University Press. All rights reserved.

The online version of this article has been published under an open access model. Users are entitled to use, reproduce, disseminate, or display the open access version of this article for non-commercial purposes provided that: the original authorship is properly and fully attributed; the Journal and Oxford University Press are attributed as the original place of publication with the correct citation details given; if an article is subsequently reproduced or disseminated not in its entirety but only in part or as a derivative work this must be clearly indicated. For commercial re-use, please contact journals.permissions@oupjournals.org
A scientist typically begins using CCT by adding three types of data through a web interface: (i) a file containing sequences of interest, (ii) database location(s) to monitor for updates and (iii) searches to perform. Although CCT can be run at will, it is typically run automatically. In this mode, CCT periodically (e.g. once per day, such as at 1 a.m.) checks user-selected databases and downloads updated versions as they become available. CCT then runs user-selected searches on these data and builds a table with one row for each sequence and one column for each database searched (Figure 1). Each cell contains links to the results of each search, color-coded by the length of time since a change in data has affected the search results. In addition, when a result is updated, the new result is compared with the previous one, and differences are highlighted in the new data file (Figure 2). Taken together, link coloring and difference highlighting allow numerous search results to be quickly scanned and evaluated for novelty (or for stability over time). This feature is valuable for scientists engaged in ongoing projects as well as those deciding when to commit limited laboratory resources to the characterization of a set of interesting but preliminary sequences.

CCT can be run locally or remotely via the Internet and comes bundled with wrappers for six different bioinformatics search programs. hmmpfam is a wrapper for the hmmpfam tool and is useful for automating searches of sequences against ever changing protein domain databases such as Pfam (5). tblastn and blastp are the software’s wrappers for NCBI’s BLAST searches (6). These programs use the local tool blastall to run searches against a sequence database. The seq program is specific to CCT and is useful for isolating regions of the genome, e.g. for finding genes and open reading frames. seq takes blastn or blastp output, captures the sequences for the target BLAST high-scoring pairs and extends them out to a user-specified delimiter. For example, if a stop codon were selected, seq would capture the sequence of the flanking open reading frame. revblast uses seq output as a query to BLAST search another database, permitting reciprocal blasting, a common method for finding ortholog pairs (7). Finally, the homolog program takes revblast output and uses Clustal-W (8) to generate a pairwise alignment if sequence pairs meet
user-specified parameters. When used together, these programs are an effective tool for finding orthologs and can be customized in a way that sets CCT apart from other comparative genomics tools (9).

The Mac OS X version of CCT can be installed in ~60 s in a few simple steps from a double clickable install package. It is fully self-contained and includes code for BioPerl, Clustal-W, blastall and hmmpfam. The Linux version does not include this code in the expectation that Linux users may want to integrate CCT into existing bioinformatics resources on their servers.

CCT’s user manual is part of every installation and can also be found at the software’s demo site. The user manual contains an installation guide, a beginner’s guide, the addresses of sample databases, screenshots and other useful information. In addition, CCT is installed with a link to extensive code documentation to make its customization as easy as possible for users with any level of programming experience.

Figure 2. Example of a result file revised to reflect a new release of the Pfam database. Red highlighting indicates new data; black highlighting shows removed data. The regions above and below the horizontal line show different parts of the same result file. This hmmpfam search result shows that the ‘Conserved Hypothetical Protein’ Rv2030c matches a new domain in Pfam, specifically the ‘erythromycin esterase’ domain. CCT’s highlighting also shows a second finding: the model for the ‘phosphoribosyl’ domain has been changed subtly (note the changed amino acids in the subject line). Taken together with the change in database size for the new release, the $E$-value for this search has changed somewhat. To view the newest results only (without highlighting to show changes) a user can click on the ‘Unhighlighted File’ link.
Programmers can construct new program modules to interact seamlessly with CCT using an included template file. The design of CCT falls into two main parts: the web interface and the script runCCT.pl. The web interface uses the Perl CGI to interact with the browser. Users can add and delete tables, searches and databases, and can view their data by browsing a CCT web page. runCCT.pl controls most of CCT’s daily work, such as downloading databases, running searches and updating tables. This script can also be called manually from the command line and can be manipulated to perform only certain steps of its process or run only specified searches.

CCT is freely available to all and it will continue to be developed (http://sourceforge.net/projects/orb-cct). Users who find it especially valuable may cite this publication.

CCT can be a very useful tool for scientists who study large sets of genes in today’s evolving genomic landscape. By empowering non-bioinformaticians to automate custom searches and examine current results at a glance, CCT allows a remote database submission in the evening to influence the next morning’s bench experiment.

ACKNOWLEDGEMENTS

The authors thank the BioPerl consortium and SourceForge for support, Bob Bried for system administration, E. Savage and E. Zweifel and many others for testing. The Institute for Genomic Research and the Sanger Center for access to incomplete genomes and our referees and colleagues for constructive feedback. M.R.O. and B.R.L. were supported by a Howard Hughes Medical Institute Undergraduate Science Education grant. Funding to pay the Open Access publication charges for this article was provided by St Olaf College.

Conflict of interest statement. None declared.

REFERENCES

1. Hokamp, K. and Wolfe, K. (1999) What’s new in the library? What’s new in GenBank? Let PubCrawler tell you. Trends Genet., 15, 471–472.
2. Pagni, M., Ioannidis, V., Cerutti, L., Zahn-Zabal, M., Jongeneel, C.V. and Falquet, L. (2004) MyHits: a new interactive resource for protein annotation and domain identification. Nucleic Acids Res., 32, W332–W335.
3. Hokamp, K. and Wolfe, K. H. (2004) PubCrawler: keeping up comfortably with PubMed and GenBank. Nucleic Acids Res., 32, W16–W19.
4. Stajich, J. E., Block, D., Boulez, K., Brenner, S. E., Chervitz, S. A., Dagdigian, C., Fuellen, G., Gilbert, J. G., Korf, I., Lapp, H. et al. (2002) The BioPerl toolkit: Perl modules for the life sciences. Genome Res., 12, 1611–1618.
5. Bateman, A., Coin, L., Durbin, R., Finn, R. D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E. L. et al. (2004) The Pfam protein families database. Nucleic Acids Res., 32, D138–D141.
6. Altschul, S. F., Gish, W., Miller, W., Myers, E. W. and Lipman, D. J. (1990) Basic Local Alignment Search Tool. J. Mol. Biol., 215, 403–410.
7. Hish, A. E. and Fraser, H. B. (2001) Protein dispensability and rate of evolution. Nature, 411, 1046–1049.
8. Thompson, J. D., Higgins, D. G. and Gibson, T. J. (1994) Clustal-W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Res., 22, 4673–4680.
9. Frazer, K. A., Elnitski, L., Church, D. M., Dubchak, I. and Hardison, R. C. (2003) Cross-species sequence comparisons: a review of methods and available resources. Genome Res., 13, 1–12.
10. Voskuil, M. I., Schnappinger, D., Visconti, K. C., Harrell, M. I., Dolganov, G. M., Sherman, D. R. and Schoolnik, G. K. (2003) Inhibition of respiration by nitric oxide induces a Mycobacterium tuberculosis dormancy program. J. Exp. Med., 198, 705–713.