annot8r_blast2GO - A tool for blast based GO-annotation

(last updated 27/09/04)

Ralf Schmid and Mark Blaxter

with assistance from Ann Hedley

The Institute of Evolutionary Biology

University of Edinburgh

for the Natural Environment Research Council

Environmental Genomics Thematic Programme Data Centre

Ashworth Laboratories, King's Buildings, Edinburgh, EH9 3JT, UK

p +44 131 650 6760 +44 131 650 6761 f +44 131 650 7489

nematode.bioinf@ed.ac.uk

http://www.nematodes.org/
http://www.earthworms.org/
http://www.tardigrades.org/

-Overview

annot8r_blast2GO is a tool to annotate a set of peptide sequences derived from EST datasets with GO-terms. This tool is part of the EST-bioinformatics pipeline developed from EGTDC Edinburgh. It is suitable for medium to high throughput data processing of peptide and protein sequences. annot8r_blast2GO is freely available for download from http://www.nematodes.org/PartiGene/ and from the EGTDC webpage.
The GO-annotation in annot8r_blast2GO is based on a BLAST search against a GO annotated BLAST database like “uniprot” (available for download from ftp://ftp.ebi.ac.uk/pub/databases/GO/goa/). The GO-annotations are stored in a postgresql database. annot8r_blast2GO also prepares output needed for the creation of “GO pie-charts” based on GO_slim annotations.

annot8r_blast2GO step-by-step

-requirements

annot8r_blast2GO runs on Linux/Unix operating systems. It has been tested on RedHat 7.2, RedHat 9.0 and Bio-Linux 3.0 systems. annot8r_blast2GO requires perl, bioperl, blast and postgresql. A short introduction how to setup and run a postgresql server can be found on the following webpage: http://envgen.nox.ac.uk/envgen/software/archives/000447.html. If you are using Bio-Linux 3.0 you don't need to install any additional software. For other Linux/Unix systems you have to make sure that the mentioned programs are available.

-getting started

After unpacking the annot8r_blast2GO.tar.gz file, copy the executable annot8r_blast2GO.pl into /usr/software/annot8r/annot8r_blast2GO and create a softlink to /usr/software/exec, or just put it into a directory which is in your path. Before using annot8r_blast2GO the user has to make sure that postgresql is installed, properly set up and running and that he or she has a postgresql user account.

-the main menu

The main menu gives the user four options. These options or sub-menus have to be performed in consecutive order. However, annot8r_blast2GO runs can be interrupted after any submenu.

1.Create GO-database from flatfile

In this submenu an annotation flat file is read and the relevant information exported into a postgresql database for more efficient downstream data processing. In this document and in the program we call this database GO-database. We recommend to create this database as an unique database for holding GO-information linked to sequence identifiers.

Unless the user decides otherwise all the necessary files are downloaded automatically. This makes sure that the most uptodate files are used. The file gene_association.goa_uniprot which links UNIPROT-sequence entries to GO-terms is available from ftp://ftp.ebi.ac.uk/pub/databases/GO/goa/UNIPROT/. The files “goaslim.map” and “GO.terms_and_ids” are required for GO-slim annotation. “goaslim.map” relates GO-terms with GO-slim terms and can be downloaded from ftp://ftp.ebi.ac.uk/pub/databases/GO/goa/goslim. The file “GO.terms_and_ids” holds associated information for each GO-term and is available from http://www.geneontology.org/doc/ .
Then the user is asked for the name of the GO-database to create. If the GO-database already exists it can be updated, otherwise a new database is created. All the downloaded files will then be processed and the relevant data imported into the GO-database. The GO-database contains a table “go” (for details see ftp://ftp.ebi.ac.uk/pub/databases/GO/goa/UNIPROT/README) complying with the format for annotation files described by the GeneOntology consortium and also the tables “go_slim” and “go_id” required for the GO_slim annotation.
We recommend running this option overnight, since this step is rather time consuming.

2. GO-BLAST preparations

Option 2 prepares the relevant files for the actual BLAST search. The user's input is the name of the GO-database as defined in option 1. Unless the user decides otherwise the most uptodate uniprot fasta-sequence files (uniprot-SWISS-PROT and uniprot-trembl) are downloaded automatically from ftp://ftp.ebi.ac.uk/pub/databases/uniprot/knowledgebase/. All sequence entries in these files are looked up for corresponding entries in the GO-database. A new BLAST database, "GO_blast.fsa", solely consisting of entries which have corresponding GO-terms is then created, formatted and saved in the working directory.

3. GO-BLAST run

In this step the actual BLAST search against the BLAST database created in the previous step is performed. The user is expected to set an e-value cutoff. We recommend setting the cut-off conservatively (eg. 1e-25) to avoid further contributions to the plague of wrong annotations. However, if only the most general level of GO-annotation is of interest (*ie* GO-slim annotation) higher e-values might be considered (eg. 1e-6). Please note: automated GO-annotations based on BLAST hits to sequences which have been automatically annotated with GO-terms are not necessarily correct - in case of doubt, use your scientific common sense.
A single file holding all the protein sequences to be blasted (as for example produced by prot4EST) is required as input file for the BLAST search. The sequences in the inputfile have to be in FASTA format. The user will be asked for the name and location of this file, then a BLAST run is started. BLAST searches can be extremely time consuming, therefore this step can also be

performed outside of the main program, for example using a 'BLASTfarm'. If you don't have access to a 'BLASTfarm' you might consider asking the EGTDC helpdesk how to run your blast search.

4. GO-annotation

In submenu 4, each BLAST hit for every query sequence is parsed to extract relevant information from the BLAST output and the related GO-information from the GO-database linked to the respective hit. For each query this information is then exported into a postgresql database, for example a database created from PartiGene, and stored in a table called blast_go. This table contains entries for protein identity, GO-BLAST database used, BLAST hit, blast e-value, GO-term, GO-category and GO-evidence.

How to run annot8r_blast2GO - an example

In this example user input is printed in ***italics***. Please note, if you exit the program before the run is completed, you can carry on from where you have exited, but you might be asked to reenter file locations or database names.

1. Start ***annot8r_blast2GO.pl***.
2. Select the "**Create GO-database**" option (***1***).
3. Enter ***y*** for downloading all the necessary data files.
4. Enter ***go_data*** as name for the database.
5. After replying ***y*** to the query "Create godbtest?" the database is created and the data are processed automatically. Please note this step can take a few hours, but it has to be done only once

=> so far you have created a GO-database holding a set of sequence identifiers with the respective GO-terms, and the data required for GO-slim annotation.

6. Type ***y*** to continue and select the "**Prepare your BLAST**" option (***2***).
7. Enter ***y*** for downloading the BLAST data files.
8. Answer ***y*** to use godbtest.

=> in this step you have prepared and formatted the files necessary for a BLAST search against a dataset consisting solely of sequences which have associated GO-terms.

For the next step you need a sequence input file in FASTA format such as "translations_xtn.fsa" from prot4EST:

9. Type ***y*** to continue and select the "**BLAST your sequences**" option (***3***).
10. Enter ***translations_xtn.fsa*** for your file holding the query protein sequences to be used.
11. Enter ***1e-25*** as e-value, this will start the BLAST search.

=>now you have blasted your sequences of interest against the dataset prepared in step 2.

12. Type ***y*** to continue and select the "**Annotate your sequences**" option (***4***).
13. Type ***y*** to select godata as your GO-term datase.
14. If you have used PartiGene previously, you are asked whether you want to use the PartiGene database as defined in your PartiGene configuration file for GO-annotation. For this test run answer ***n***.
15. Enter ***antest*** as database to be created or used.
16. Reply ***y*** to the query to create "antest". This will create the database and start the processing of the BLAST output.

=> in the final step of annot8r_ blast2GO you have annotated your sequences of interest with GO-terms and stored them in the postgresql database "antest".

17. Enter ***n*** to quit annot8r_blast2GO.
18. type ***psql antest*** to enter your results postgresql database.
19. type ***select * from blast_go;*** (don't forget the semicolon) to view your results.
20. type ***\q*** to exit postgresql.

The data necessary to create GO pie-charts based on GO-slim annotation are stored in the files piedata_C, piedata_F and piedata_P.

For further questions, comments, problems or bug reports please contact nematode.bioinf@ed.ac.uk

If you are working in an EG-Awardee lab and have questions or problems please contact the helpdesk helpdesk@envgen.nox.ac.uk