Blast2GO CLI User Manual

Version 1.4.0

February, 2018

BioBam Bioinformatics S.L.

User Manual

Support: clisupport@blast2go.com
Website: https://www.blast2go.com

A Functional Annotation Pipeline: The Blast2GO Command Line is a professional solution for flexible, high-performance and automatic functional annotation tasks. This Annotation Pipeline allows you to integrate and automate your functional annotation task in a flexible way. generate high-quality results in a reproducible way directly integrated into your data analysis workflows. The Blast2GO Command Line (CLI) version allows to combine BLAST and InterPro XML results, to perform the mapping against the Gene Ontology database and to assign the most suitable functional labels to the un-characterized sequence dataset. The command line allows to run the GO analysis on a local database which can be easily setup from the command line. Additionally the CLI allows to perform GO-Slim and Annex, to generate over 25 different summary charts and a comprehensive PDF report of each analysis. The Command Line is based on the Blast2GO methodology, first published in 2005, [Conesa et al., 2005] for the functional annotation and analysis of gene or protein sequences. The method uses local sequence alignments (BLAST) to find similar sequences (potential homologous) for one or several input sequences. The program extracts all Gene Ontology (GO) terms associated to each of the obtained hits and returns an evaluated GO annotation for all query sequence(s). Enzyme codes are obtained by mapping to equivalent GOs and InterPro motifs can directly be added to the BLAST based annotation. A basic annotation process with Blast2GO consists of 4 steps: blasting and interpro-scan, GO mapping and functional annotation [Götz et al., 2008].

Copyright © 2017, BioBam Bioinformatics S.L.

Main Characteristics

- **High Performance**
  The command-line version of Blast2GO allows you to analyse large datasets on your own computing servers without a grafical user interface.
- **Flexible**
  Easily integrate your functional annotation tasks within your custom analysis pipeline and run different analysis scenarios in parallel.
- **Automatic Data Generation**
  Generate all the statistics charts Blast2GO offers in an automatic fashion. This includes a summary report in PDF as well as different images and text file formats.
- **Reproducibility**
  Control the whole analysis with a simple configuration file. This allows you to set up different analysis strategies and reproduce the multiple scenario for one or various datasets.
- **Secure**
  Run BLAST, InterProScan and the Blast2GO annotation offline on your own servers according to your security requirements. Take 100% control of data sources and versions you use throughout the analysis.

Copyright © 2017, BioBam Bioinformatics S.L.

Main Features

- Perform Blast (Cloud/Local) directly from the Blast2GO Command Line
- Perform InterProScan from the Command line (online feature)
- Semi-automatic local GO database setup and update
- Run Blast2GO on your own servers and control all analysis steps from the command line
- Automate your functional annotation
- Reproduce your results in a consistent manner
- Handle tens of thousand of sequences
- Design advanced annotation strategies
- Integrate Blast2GO into your existing analysis pipeline
- Work offline with your own resources
- Create your own local Blast2GO database
- Fast import of BLAST and InterProScan results
- Automatically generate PDF Reports
- Save all your results to specific project folders
- Work consistent and effective once you defined the right settings for your analysis

Copyright © 2017, BioBam Bioinformatics S.L.

Setup

System Requirements

Blast2GO Command Line (CLI) is a Java application and can be run on Mac, Linux and Windows 64-bit systems. It is always necessary to have Java 64-bit (version 1.8 from Sun/Oracle) installed, at least 1GB of RAM is recommended. The Blast2GO Command Line needs a Blast2GO database to perform the mapping step. This database can be imported from existing database dumps for MongoDB, those are provided on our webpage.

> In general this program works offline, however the CloudBlast and InterProScan need a working internet connection.

Product Activation

Blast2GO offers 2 types of product activation. Either bound to a specific hardware or via a floating license. Floating licenses are served by a RLM license server in the same network and the setup instructions are provided upon request. This section describes how to activate the product on a specific hardware (either for evaluation or perpetual). You will need your Activation Key for these steps. The license can be obtained automatically from www.blast2go.com/cli-activation. To do so, a signature of the workstation has to be generated first. The command line parameter -createkeyfile will generate such a file named information.lic.
Note: On MS Windows all the following commands starting with ./blast2go cli.run must be replaced with blast2go cli.exe.
Steps to activate the Blast2GO Command Line:

1. Generate the information.lic file by executing the following command:

```
./blast2go_cli.run -createkeyfile
```

2. Go to www.blast2go.com/cli-activation and upload information.lic, provide your Activation Key and your Email.
3. The Blast2GO support team will create a license.lic file which has to be placed in the same folder as the blast2go cli executable.
4. You can check the details of a license file with the option -showlicenseinfo You can now continue with the GO Mapping database installation(See section 2.4) and have a look at the example use cases.

Create a Properties File

The Blast2GO Command Line needs a properties file, that contains all the information of the different paramaters that can be changed for the analysis. The properties file can be created with this command:

```
./blast2go_cli.run -createproperties cli.prop
```

Once this file has been created it is possible to edit it with a text editor. An editor with syntax highlighting is recommended, since it allows to better distinguish between comments and parameters.

Setting up a local Blast2GO GO Database

In order to set up a local GO Mapping database, please install Mongo 3.4.X or higher (https://www.mongodb.com/download-center?jmp=nav#community) and import one of the data-dumps provided by us.

Dumps can be downloaded from:

http://resources.biobam.com/go_mapping_dumps/index.html

```
wget -c http://resources.biobam.com/go_mapping_dumps/2017.12.gz
wget -c http://resources.biobam.com/go_mapping_dumps/2017.12.gz.md5
md5sum -c 2017.12.gz.md5
```

After a successful download, the import can be done as follows:

Copyright © 2017, BioBam Bioinformatics S.L.

```
mongorestore --host 192.168.1.240 --db go_db --collection 2017.12 --gzip
--archive=2017.12.gz
```

This would result in the following settings for Blast2GO (cli.prop):

```
// ** GoMappingDirectParameters **
// Please specify your MongoDB connection URI:
// https://docs.mongodb.com/manual/reference/connection-string
GoMappingDirectParameters.mongoUri=mongodb://192.168.1.240/go_db

// This is the name of the MongoDB collection to be used.
GoMappingDirectParameters.collectionName=2017.12
```

- For more details on how to configure your mongo URI go to https://docs.mongodb.com/manual/reference/connection-string
- The configuration has been tested with mongod v3.4.6, it should work with any version equal or higher.
- Please consider the following for your mongod.conf:

```
storage:
  # Where and how to store data. You will need about 20 GB at least.
  dbPath: /var/lib/mongodb
  journal:
    enabled: false
  # Use this storage engine.
  engine: wiredTiger

net:
  # Change the IP address binding if necessary and take it into account for the
MongoURI.
  port: 27017
  bindIp: 192.168.1.240
```

Copyright © 2017, BioBam Bioinformatics S.L.

Command Line Parameters

This section gives a quick guide on the parameters used in Blast2GO CLI. Some command examples will be given in the end of the detailed description.

Load or import data commands:

- **loadannot** <path> Path to .annot file
- **loadb2g** <path> Path to Blast2GO .b2g file
- **loadblast** <path>Path to Blast .xml file (pre 2.2.31)
- **loadblast31** <path>Path to Blast .xml/.json/.zip file (2.2.31+)
- **loaddat** <path> Path to Blast2GO .dat file (legacy format - use loadb2g instead)
- **loadfasta** <path> Path to fasta file. Activate -protein option when working with amino acids.
- **loadips48** <path> Path to InterProScan 4.8 file or folder
- **loadips50** <path> Path to InterProScan 5.0 file or folder

Analysis commands:

- **annex** Run ANNEX to complement the Gene Ontology annotation based on existing molecular functions
- **mapping** Run the Gene Ontology mapping
- **annotation** Run the Blast2GO Annotation algorithm
- **ecmapping** Map Annotated GOs to their Enzyme Codes (Included when -annotation is set)
- **goslim** <path> Run goslim using an *.obo file. Possible subset obo files can either be downloaded from http://geneontology.org/page/go-slim-and-subset-guide, or customized by hand with OBO-Edit2.
- **cloudblast** <cloublastkey> Run CloudBlast via webservice. This requires a working internet connection and a valid CloudBlast key with a positive balance.
- **cloudblastbalance** <cloublastkey> Print the CloudBlast Computation Unit balance. This requires a working internet connection.
- **extractfasta** <path> Extract features from a fasta reference to a fasta file (path). Needs configuration in the properties file.
- **ips** <email> Run InterPro via webservice. This requires a working internet connection, a valid email address and that your data-set contains sequence data.
- **localblast** <path> Run Blast against a local database. This requires a working internet connection in order to download the necessary Blast executable (Alternatively you can specify a binary folder manually and place the binary there). Also necessary is a correctly configured local blast database (properties file).

Save or export commands:

- **saveannot** <path> Save the functional annotations (Gene Ontolgoy terms and Enzymes) as .annot
- **saveb2g** <path> Save the project as .b2g
- **savedat** <path> Save the project as .dat (legacy format - use saveb2g instead)
- **savelog** <path> Save the log in a specified file.
- **savelorf** <path> Convert nucleotide sequences (FASTA format) into amino acid sequences (longest Open Reading Frame, FASTA format). This function may be used to prepare a FASTA file for a local InterProScan run.
- **savereport** <path> Create .pdf report
- **saveseqtable** <path> Save your data as it would be shown in the Blast2GO GUI version (tab separated)
- **statistics** <charts> Provide a comma-separated list of desired statistical charts (try -statistics without options to get a list of all available charts). '-statistics all' will try to export all statistics that are available. The option -nameprefix will be ignored.
- **exportgeneric** <path> Export sequence data in tabular format for post processing

Other Options:

- **createproperties** <path> Path to where the default properties file should be created
- **createkeyfile** Create a file which contains a unique ID for your computer. This file is nessesary to issue license keys.
- help Display this message
- **nameprefix** <name> Prefix for any output files, if you do not specify any path for them (default: b2g project)
- **properties** <path> Path to properties file (mandatory)
- **protein** Set this flag if the fasta file contains protein sequences. This option only makes sense together with the -loadfasta option.
- **showlicenseinfo** Show details about the currently available license.
- **tempfolder** <path> Path to temporary folder (default: Systems temp folder)
- **useobo** <path> The obo file to use for annotation, some statistics and various file im- and exports. Download the latest version from http://data.biobam.com/b2g res/obo files/go latest.obo.gz
- **workspace** <path> Workspace folder, e.g. where the results will be saved if not specified (default: current folder)

> If a path is specified for a save option (e.g. -**saveannot**), the options **workstation** and **nameprefix** will be ignored for this particular option (see **Use Case Examples** for detailed information).

Copyright © 2017, BioBam Bioinformatics S.L.

Generic Export

The option **-exportgeneric** allows to export the data obtained for each sequence into a tabular text file. The reulting file contain one line per sequence and customizable columns. The corresponding settings in the properties file (**GenericExportParameters**) allow to decide how columns and items are separated and which data should be exported.
Column and item separators can defined as: comma, semicolon, tabulator, whitespace or the pipe symbol (|). There are 54 different items available for each sequence. These items can be grouped as follows:

- General Sequence information such as the sequence name or its length.
  seq_name, seq_desc, seq_data, seq_length
- Summarized Blast information like the total number of blast hits.
  blast_hit_count, blast_min_eval, blast_sim_mean
- Specific information about each Blast hit such as its hit description, length, e-value or alignment length.
  blast_hits_desc, blast_hits_tax, blast_hits_eval, blast_hits_length, blast_hits_alignlength
  blast_hits_pos, blast_hits_sim, blast_hits_hsphit, blast_hits_query, blast_hits_hspcount
  blast_hits_frame, blast_hits_geneid, blast_hits_acc, blast_hits_score
- Specific information about the top blast hit (highest bit score).
  blast_tophit_desc, blast_tophit_tax, blast_tophit_eval, blast_tophit_length
  blast_tophit_alignlength, blast_tophit_pos, blast_tophit_sim, blast_tophit_hsphit
  blast_tophit_query, blast_tophit_hspcount, blast_tophit_frame, blast_tophit_geneid
  blast_tophit_acc, blast_tophit_score
- All GO Mapping candidate GO terms and evidence codes.
  mapping_genename, mapping_tax, mapping_xref, mapping_xref_db
  mapping_goid, mapping_goname, mapping_gocategory
- Annotated GO terms and enzyme codes.
  annot_count, annot_goid, annot_goterm, annot_gocategory, enzyme_code, enzyme_name
- Detailed information about InterPro results like obtained domains, families as well as GO terms.
  ips_acc, ips_type, ips_name, ips_sig, ips_goid, ips_goname, ips_gocategory

Copyright © 2017, BioBam Bioinformatics S.L.

Use Case Examples

Before We Start

This section provides several example use cases for the Blast2GO Command Line. Please read the Setup chapter carefully and configure the GO Mapping database in your properties file.

Important things to consider:

- If you are using MS Windows all commands must be changed accordingly. Please replace **./blast2go_cli.run** with **blast2go_cli.exe**
- A properties file is always necessary, create it with:
  ./blast2go_cli.run -createproperties cli.prop
- GO Annotation, Enzyme Code Mapping, Statistics, GO Slim and various import and export functions make use of the obo file (-useobo). The CLI contains a default obo file. However, we recommend that to download the up-to-date version of the obo file.

  > The obo file should be from the same month as the GO Mapping database. http://resources.biobam.com/b2g_res/obo_files/index.html

  The latest version can always be found here: http://resources.biobam.com/b2g_res/obo_files/go_latest.obo.gz
  Just provide this file additionally when executing a command:
  ./blast2go_cli.run -useobo go_latest.obo.gz -properties cli.prop -annotation ...
- Make sure having a working MongoDB server installed, with the GO Mapping database dump imported (see setup chapter).

Examples

1. Load a DNA fasta file, add the corresponding BLAST results and perform GO Mapping and Annotation. Furthermore, we want to save the .b2g file and the PDF report at the current directory with the name "example".

   **./blast2go_cli.run -properties cli.prop -loadfasta **
   **example_data/1000_plant.fasta -loadblast example_data/1000_plant_blastResult.xml **
   -mapping -annotation -saveb2g example.b2g -savereport example.pdf

2. This example requires a local Swissprot Database installation. Simply download and extract the file from: ftp.ncbi.nlm.nih.gov/blast/db/swissprot.tar.gz Load nucleotide sequences, run local BLAST against the Swissprot database, GO Mapping and Annotation. We also create various statistics. Finally the whole project will be saved to the example data folder in .b2g format with the chosen name prefix together with the log file. The LocalBlastAlgoParameters have to be configured:

   **// ** LocalBlastAlgoParameters **
   LocalBlastAlgoParameters.blastProgram=blastx-fast**
   LocalBlastAlgoParameters.blastDbFile=/path/to/swissprot.pal
   LocalBlastAlgoParameters.blastXML2ResultEnable=true
   LocalBlastAlgoParameters.blastXML2Result=example_data/blast_xmls

   Command Line:
   **./blast2go_cli.run -properties cli.prop -loadfasta example_data/15_plant.fasta **
   **-workspace example_data -nameprefix localblastSwissprot **
   **-localblast -mapping -annotation -statistics **
   bspecdis,mdbresmap,aannotscore -saveb2g -savelog example_data/blast2go.log

3. Load nucleotide sequences, import BLAST results (.json or .xml2) from a zip file, run GO Mapping and Annotation. Save the whole project and its report as example json.b2g.

   **./blast2go_cli.run -properties cli.prop -loadfasta example_data/15_plant.fasta **
   **-loadblast31 example_data/json/02X9PD4T01R-Alignment.json.zip -mapping **
   **-annotation -savereport example_data/example_json_report.pdf -saveb2g **
   example_data/example_json.b2g

4. Load example.b2g from the second example and run InterProScan (online). We will save the project, as well as the InterProScan results. The following InterProScanAlgoParameters have to be configured:

   **// ** InterProScanAlgoParameters **
   InterProScanAlgoParameters.ipsXML2Result=example_data/ips_xmls**
   InterProScanAlgoParameters.ipsXML2ResultEnabled=true

   Command Line:

Copyright © 2017, BioBam Bioinformatics S.L.

**./blast2go_cli.run -properties cli.prop -loadb2g example_data/example.b2g **
-ips <valid_email_address> -saveb2g example_data/example_withIPS.b2g

5. Convert sequences to proteins and save them as fasta file.

   **./blast2go_cli.run -properties cli.prop -useobo go_latest.obo -loadfasta **
   example_data/15_plant.fasta -savelorf example_data/15_plant_protein

6. Load a .b2g file, apply plants GO Slim and save the results as .b2g, which will be saved with the default nameprefix “b2g project” into the current directory.

   **./blast2go_cli.run -properties cli.prop -useobo go_latest.obo -loadb2g **
   example.b2g -goslim example_data/goslim_plant.obo -saveb2g

7. To run this example, we need the results from the previous example. Run CloudBlast, GO Mapping, Annotation on the protein sequences and save the results as .b2g and customized annotation format. Please configure the following properties sections:

   **// ** CloudBlastAlgoParameters **
   CloudBlastAlgoParameters.blastProgram=blastp-fast**
   CloudBlastAlgoParameters.blastDB=nr_alias_viridiplantae
   CloudBlastAlgoParameters.blastXML2ResultEnable=true
   CloudBlastAlgoParameters.blastXML2Result=example_data/blast_xmls

   **// ** ExportAnnotParameters **
   ExportAnnotParameters.format=custom**
   ExportAnnotParameters.desc=true
   ExportAnnotParameters.go=category_and_id_and_term
   ExportAnnotParameters.goseparator=tabulator
   ExportAnnotParameters.column=tabulator
   ExportAnnotParameters.row=sequence

   Command Line:

   **./blast2go_cli.run -properties cli.prop -loadfasta **
   **example_data/15_plant_protein.fasta -protein -cloudblast B2G-CloudBlastKey **
   **-mapping -annotation -saveb2g example_data/15_plant_protein.b2g **
   -saveannot example_data/15_plant_annotation.txt

8. Load a protein fasta file, add the corresponding BLAST results and execute GO Mapping and Annotation. All files (.b2g, .pdf, .annot and .txt) will be saved with the nameprefix "p53" in the "working dir" in the current directory. Additionally, the data distribution pie chart and enzyme statistics will also be saved in the “work dir” folder.

   Command Line:

   **./blast2go_cli.run -properties cli.prop -loadfasta **
   **example_data/1000_seq_protein.fasta -protein -loadblast **
   **example_data/1000_plant_protein_blastResult.xml -mapping -annotation **
   **-workspace work_dir -nameprefix p53 -saveb2g -saveannot -savereport **
   -saveseqtable -statistics gdatadispie,aecdis

9. Load a fasta file, a BLAST result file and InterProScan 5.0 files, perform GO mapping, annotation and ANNEX. Then create all statistical charts. As a result we want to obtain the .b2g and the PDF report, which will be saved with the default nameprefix “b2g project” into the current directory.

   Command Line:

   **./blast2go_cli.run -properties cli.prop -loadfasta example_data/1000_plant.fasta **
   **-loadblast example_data/1000_plant_blastResult.xml -loadips50 **
   **example_data/1000_seq_protein_ips50.xml -mapping -annotation -annex **
   -statistics all -saveb2g -savereport

10. Load an example data-set and export user defined columns for each sequence. Please configure the following properties section:

    **// ** GenericExportParameters **
    GenericExportParameters.columnSeparator=tabulator**
    GenericExportParameters.itemSeparator=semicolon
    **GenericExportParameters.itemsToExport=seq_name,blast_hit_count, **
    mapping_genename,mapping_xref,mapping_goid,annot_goid,enzyme_code

    Command Line:

Copyright © 2017, BioBam Bioinformatics S.L.

```
./blast2go_cli.run -loadb2g example_data/example.b2g \
-exportgeneric example_data/blast_top_hit.txt
```

Copyright © 2017, BioBam Bioinformatics S.L.

Bibliography

[Conesa et al., 2005] Conesa, A., Götz, S., García-Gómez, J. M., Terol, J., Talón, M., and Robles, M. (2005). Blast2go: a universal tool for annotation, visualization and analysis in functional genomics research. Bioinformatics, 21(18):3674–3676.

[Götz et al., 2008] Götz, S., Garcia-Gomez, J. M., Terol, J., Williams, T. D., Nagaraj, S. H., Nueda, M. J., Robles, M., Talon, M., Dopazo, J., and Conesa, A. (2008). Highthroughput functional annotation and data mining with the blast2go suite. Nucl. Acids Res., pages gkn176+

Copyright © 2017, BioBam Bioinformatics S.L.