**User’s Guide to PAW\_BLAST program “db\_to\_db\_blaster.py”**

**Written by Phillip Wilmarth**

**David Lab, OHSU**

**9/23/2011**

**(minor update 3/9/2018)**

**Requirements:**

A local, stand-alone copy of NCBI BLAST program must be installed on the computer running “db\_to\_db\_blaster.py”. BLAST is available for several platforms and downloadable installers are available at this URL:

<ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/LATEST/>

There is a program variable called “blast\_path” which needs to be set to the location where BLAST is installed on your computer. If you get an error message, search the source code for “blast\_path” and change to match the location of BLAST on your computer. The program requires a 3.X version of Python which can be obtained from [www.python.org](http://www.python.org) or from a scientific python distribution like anaconda.

The program does not require any modules other than the standard libraries included in the main Python distribution.

**Running Python programs:**

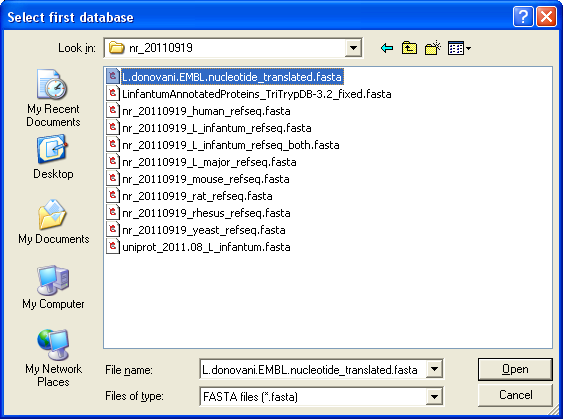
The program can be launched in a few ways. The program’s icon can be double clicked on Windows systems. The program can be opened with IDLE, an integrated development environment included with the Python distribution. This is easiest by using context menus (right-clicking). Programs in IDLE are run using the F5 key or the “Run” menu. Python can also be executed from a command line with “db\_to\_db\_blaster.py” as the second parameter.

**Program description:**

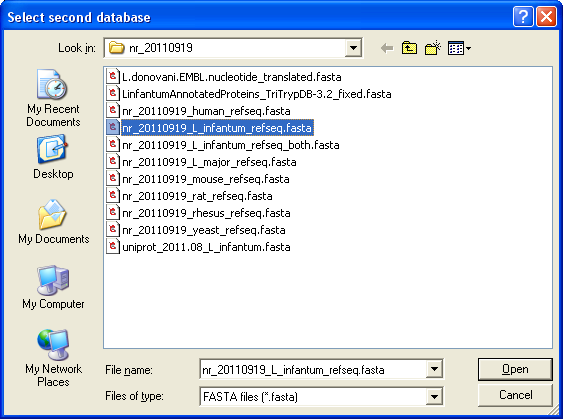
Program logic is simple. It takes every sequence in the first database (called the query database) and performs BLAST alignments to all proteins in a second database (called the hit database). The databases should be protein sequences in FASTA format. The databases will be converted to BLAST formatted files (3 files with extensions of “.phr”, “.pin”, and “.psq”). If those files exist, the database conversion is skipped to save time. Be warned that BLAST may not like valid FASTA characters such as stop codons (\*) or gaps (-) and input databases may need to be processed to remove such characters. The BLAST run is saved in a (large) XML file and may take several hours to complete. If the XML file already exists, the BLAST run is skipped. If the BLAST run needs to be repeated make sure the XML file is deleted before running “db\_to\_db\_blaster.py”. After the BLAST run has finished, the XML file is parsed and the top-matching hit protein for each query protein is reported in a tab-delimited text file that can be opened with Excel or any text editor. Details about the output file contents are given below.

**Program execution:**

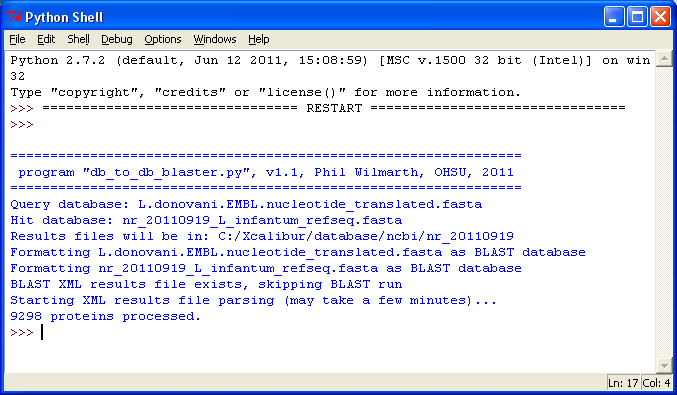
When the program starts, a dialog box will ask the user to browse to the first (query) database file:



The databases should be protein FASTA files. The second (hit) database does not have to be in the same location as the first database, however, the results files will be written to the folder containing the first database. After selecting the first database file, a second dialog box appears for selecting the second database:



Several lines of status information will be written to the console during program execution and a sample is shown below:



After program has finished the results files will be located in the same folder as the first database. The results files will be two files with “.xml” and “.txt” extensions. The filename will be a combination of the query and hit database names. The XML file is the full results from the BLAST run, including alignments and all hits per query. The TXT file is a tab-delimited summary of the top matches that shows query protein accession and description, top-matching hit accession and description, BLAST scores, and status information. Details are given below.

**Database definitions:**

The reference database (the first database selected) is denoted as "query" and the database that is being BLASTed against (the second database selected) is denoted as "hit". The output file will have an entry for each protein in the query database. The highest scoring "hit" (top match) for a given query sequence is always reported unless there was nothing above some minimum e-value threshold that is specified in the blast call. The cutoff is pretty low, something like 10, but e-values depend on the size of the database and are calibrated for the 13.5 million sequence nr database not smaller single organism sequence databases. A status of "No\_match" means that nothing was returned by BLAST.

**TXT file Column keys:**

query\_number ==> sequential number of the proteins in the query database.

query\_acc ==> the protein accession for the query sequence.

query\_desc ==> the protein description string for the query sequence.

hit\_acc ==> the accession for the best matching hit protein.

hit\_desc ==> the protein description string for the best-matching hit sequence.

blast\_score ==> a collection of information from the BLAST run (see below).

status ==> classification of the match quality (OK, Better, Partial, Poor, or No).

**The BLAST scores are defined as:**

ident:X/Y => number of identical amino acids (X) out of the total query length (Y).

pos:X/Y => number of identical or similar amino acids, same definitions for X and Y.

hit: length (*aa* count) of hit sequence.

align: subset of hit that encompasses the alignment region (gaps, deletions, etc.).

bit: the BLAST bit score (sort of an information content score, better than e-value for small databases).

**Status code explanations:**

If the alignment region length is less than half of the hit sequence length, the status will be “Partial\_match\_hit”. If the alignment region length is less than half of the query sequence length, the status will be “Partial\_match\_query”. If the alignment region length is less than half of the hit sequence length AND the alignment length is less than half of the query sequence length, the alignment is called "Partial\_match\_both".

For all of the hit proteins, all bit scores for query matches are saved. If a query protein matches to a given hit protein but that hit protein matched to a different query protein with a higher bit score, then the match is called "Better\_match". If you search the "hit\_acc" column for an accession associated with a "Better\_match", you should find at least one different query protein (different row) where the bit score is higher. If there are gene duplications and such then there may be multiple matches with nearly the same bit scores.

The mean and standard deviation of percent identity scores are tabulated for all matches. If the identity is less than the mean identity minus one standard deviation, the match is called a “Poor\_match”.

If no hits were above the BLAST e-value cutoff, then the status will be “No\_match”.

The status will be “OK” when none of the above exceptions occur.