Tax-Filter: Taxonomy Filter Tool
ACeS Software Package

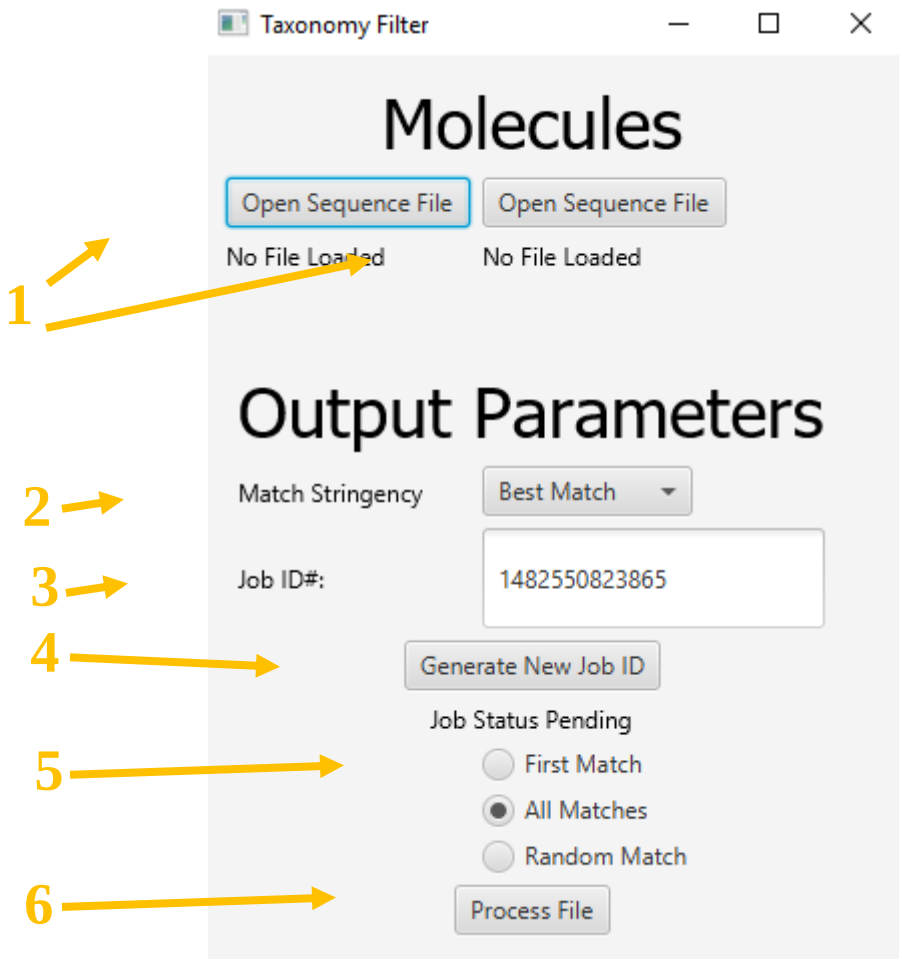
User Manual
-Professor Camenares, Kingsborough Community College

Purpose:
This tool will take two separate FASTA files, each representing a collection of sequences, and filter both such that the only remaining sequences are those that are from an organism represented in both collections. One purpose of this tool is to prepare a batch of sequences for mutual information analysis; the result are two files, one for each gene, with each row representing a single genus or species. The resulting files can then be aligned separately and compared with the mutual information tool to determine co-evolution. 'Discarded' sequences are not lost, but rather moved to a separate file for comparison or further analysis.

Input:
This tool requires two FASTA files, each with one or more sequences. They do not need to represent the same gene, or even the same type of macromolecule, but should ideally contain sequences from at least some of the same species or genus.

In order to be processed properly, each new sequence in the FASTA file must start with a ">" character. The header line following this can contain information about the gene, such as reference number or name, but must end with the taxonomical identification of the source organism; the organism name must be found between brackets ("[" and "]"). Following the closing bracket should be the gene or protein sequence.

Interface:



1. **File Open Buttons:** Use these to select the appropriate files to compare / filter. Each button loads a different sequence file, which must adhere to the input specifications mentioned above.

2. **Matching Stringency:** This selection controls how strict the matching is; in a rough approximation, it will successfully match any two genes that are from the same genus, or species, or subspecies, or strain. In reality, this represents a matching of either the first, second, third, or fourth separate words or names contained within the brackets that flank the organism name (respectively). There are two additional settings: “Best Match” and “Fast Match”.

   “Best Match” will match the taxonomy of sequence 1 word by word for as many words are available for the taxonomy of sequence 2, checking to make sure there is a match at each step. Thus, *Escherichia coli* and *Escherichia coli O157H* would match, as the first taxonomic ID does not specific which strain, while *Escherichia coli Nissle 1917* would not match with the O157H strain. The search depth is not fixed in this case, but is determined on a case-by-case basis.

   “Fast Match” takes a different, but similar approach. For taxonomic names 1 and 2, the entire word or string will be checked to see if it matches, or is contained in, the string / name of the other. This process is reciprocal, and will generate a similar result to “Best Match” in the *E. coli* example above.

3. **Job ID:** This textfield, with a timestamp generated number, will be used to name the directory in which the output files will be placed. You may rename the folder by changing this value. Note that a new ID is automatically

generated each time the program is loaded anew.

4. **Generate New ID:** As the name implies, this button will generate a new ID (so that previous work is not overwritten). The new Job ID is displayed in the textfield (item #3)

5. **Match Action:** In the event that there are multiple matches between different sequences (for example, on "Best Match", a sequence from *E. coli* would match both *E. coli O157H and E. coli Nissle 1917*), this option determines which sequences will be used; either the first encountered, a random selection, or all of them. Selecting "All Matches" can generate a large file with many duplicates (in the above example, there would be two instances of *E. coli* in one file, matched by the *O157H* and the *Nissle 1917* strains in the other file).

6. **Process Button:** Selecting this button will process the inputs; an error message will be displayed if the one or more inputs are missing.

Output and Additional Notes:

As a sample, two FASTA sequence collections are provided: this includes files tRNA_Ala_test.txt (collection of Alanine tRNA sequences) and AARS_Ala_test.txt (collection of Alanine tRNA synthetase sequences). Running these, or any other two sequences, will generate a folder with the following files:

1. **Runtime Information:** Summarizes the run of the program, including the amount of processing time required, number of sequences in the input and the output.

2. ***Job_name*_keep1.txt:** The collection of sequences from file 1 that are from an organism represented in both files.

3. ***Job_name*_waste1.txt:** The collection of sequences from file 1 that are from organisms unique to that file alone.

4. ***Job_name*_keep2.txt:** The collection of sequences from file 2 that are from an organism represented in both files.

5. ***Job_name*_waste2.txt:** The collection of sequences from file 2 that are from organisms unique to that file alone.

The 'waste' files are provided as to inform a broader search and make your analysis more inclusive. For example, you may find that you have a tRNA sequence from species X, but lack an AARS sequence. You can expand your analysis by searching the NCBI database (http://www.ncbi.nlm.nih.gov/) for the AARS sequence from species X – it might have a cryptic name, be heretofore unannotated, or otherwise require this more rigorous, individual search.