FTrees Commandline Documentation

Marcus Gastreich

May 4, 2022

Contents

1 Introduction

FTrees is modern software for very fast, fuzzy similarity searching with query molecules. The search can span vast quantities of molecules ("chemical spaces") — or traditional collections of molecules ("libraries"). FTrees technology lies at the heart of our flagship tool infiniSee. This document is a brief documentation for the commandline package.

The basic principle behind FTrees is that molecules are represented by trees called "Feature Trees". Every tree is composed of nodes that represent molecular fragments, and vertices that describe their connectivity. Depending on whether the search is carried out on a standard library or in a chemical space, the algorithm differs slightly:

- Standard similarity: The query and the hit Feature Trees are aligned using a dynamic programming algoritm (see Further Reading at the end). The nodes of the trees, or fragments they represent, are aligned in such a way that the overall similarity of the trees, or molecules, is maximized. Both the overall similarity and the so-called local similarities are output. This alignment can be visualized together with the local similarities and is therefore of great benefit to the user: It shows *why* the computer considers parts of the two molecules to be similar or dissimilar.
- Chemical space navigation: The query Feature Tree is split up into fragments, and — obeying the original connectivity in the query — fragments are searched from the chemical space that have a similarity as close as possible to the target similarity (see further below). The next fragment is added in accordance with the connection/reaction rules encoded in the chemical space to optimize the similarity score. Overall, a new molecule is constructed from the chemical space that overlaps as well as possible with the query molecule (or rather, its Feature Tree), and minimizes the difference between the target similarity and actual similarity (default: identity).

FTrees reads input and writes output: Input consists of two files, one query file and one library (or chemical space) file. Input files can be in SD, SMILES, or mol2 file format, and the files may contain one or multiple molecules. Chemical spaces are *.space files, downloadable from biosolveit.de/download. Output is typically an SD file that contains the similarity values in SD data fields. Other types of output such as an ASCII list are possible.

Please note that FTrees is agnostic with regards to stereo chemistry and o-,m-,p- substitution patterns, due to the fuzziness of the descriptor. It is therefore a good idea to augment FTrees results with information about shape (3D align-

ment,...) or with docking experiments. A hit containing an R stereo center could just as well be an S isomer; cis could just as well be trans!

FTrees similarities are normalized from 0 (entirely different) to 1 (identity).

2 Technical Prerequisites

FTrees is a commandline application. (If you prefer a graphical tool, then please download infiniSee from biosolveit.com/download; infiniSee uses the same algorithms below the surface and will therefore deliver identical results.) FTrees needs the following to run:

- The FTrees package (from biosolveit.de/download); depending on your operating system, some libraries may have to be installed (get in touch with us if that is the case: support@biosolveit.com; and please mention any errors/warnings that you see in your mail)
- A shell (Linux/Unix) or a terminal (macos), or a commandline environment (Windows; e.g.: cmd.exe)
- A valid license (from license@biosolveit.com)

The license setup/install instructions will come with the license that we send out or have already sent out to you. A "test license" that you can request online and that is sent to you instantaneously can simply be placed next to the executable (FTrees.exe, ftrees, or FTrees — depending on your operating system).

macos Specialties On macos, the executable will typically reside inside the *.app package:

```
/Applications/FTrees.app/Contents/MacOS/FTrees
```

When you call FTrees for the first time, go to the Finder, and navigate to the Applications folder. Do a right(!) click on FTrees.app, and — if applicable — confirm that you want to open the program. It will flash up once, and you are good to go at the terminal prompt from there on.

To make the first step, call the FTrees within your shell/terminal/environment.

3 Commandline Options

An overview of all of FTrees's commmandline options is available by calling FTrees with `--help`:

```
./FTrees --help
 Available options:

 Program options:
  -i [ --input ] arg                  Input query molecule file or single input molecule as smiles.
                                      Supported file types are *.smi, *.smiles, *.mol, *.mol2 and *.sdf.
  -s [ --searchFiles ] arg            Paths to library input molecule files for similarity scoring or to
                                      Fragment Space FSF files or Fragment Spaces. Supported file types are
                                      *.smi, *.smiles, *.mol, *.mol2, *.sdf, *.fsf, *.space and *.zip.
                                      Note: The .flf and fragment files specified in the FSF have to be in
                                      the appropriate relative paths.
  -o [ --outputFiles ] arg            Output base files (suffixes are required). Supported file types are
                                      *.csv and *.sdf.
  -m [ --matchImageBaseFile ] arg     Output base file name for matching images (suffix required).
                                      Supported file types are *.pdf, *.png and *.svg.
                                      Note: For each match a separate file is created.
  --gen2dOutput arg (=0)              Generates 2d coordinates in case of SDF output files.
                                      Note: Can't be used together with '--gen3dOutput'.
  --gen3dOutput arg (=0)              Generates 3d coordinates in case of SDF output files.
                                      Note: Can't be used together with --gen2dOutput'.

 Configuration:
  --expandAlternativeResults arg (=0)  Write alternative results based on alternative reaction paths or
                                       duplicate matchings.
  --maxNofResults arg (=100)           Maximum number of top-ranking result molecules [1 to 1000000].
  --minSimilarityThreshold arg (=0.8)  Similarity threshold below which molecules are discarded [0.0 to 1.0].
  --targetSimilarity arg (=1)          Desired target similarity to the query molecule [0.5 to 1.0].
                                       Note: Must be >= '--minSimilarityThreshold'
  --totalDiversity arg (=1)            Required diversity between any two compounds in a solution set [0.9 to
                                       1.0].
                                       Note: Only available if --maxNofResults' is <= 500.
                                       WARNING: any value below 1.0 drastically extends the run time.

 Deprecated options:
  -l [ --library ] arg                Library input molecule files to calculate similarity score with.
                                      Note: Can't be used together with '--searchFiles'.
  -f [ --fragSpace ] arg              Paths to the Fragment Space FSF files or Fragment Spaces.
                                      Note: The .flf and fragment files specified in the FSF have to be in
                                      the appropriate relative paths.
                                      Note: Can't be used together with '--searchFiles'.

 General options:
  -h [ --help ]                       Prints this help message
  --license-info                      Prints license info
  --thread-count arg                  Maximum number of threads used for calculations. The default is to use
                                      all available cores.
  --version                           Prints version info
  -v [ --verbosity ] arg (=2)         Set verbosity level
                                           0 [silent]
                                           1 [error]
                                           2 [warning]
                                           3 [workflow]
                                           4 [steps]
```

Please note that the abbreviated, one-letter options are preceded with one dash - whereas the longer, named options are preceded with two dashes: --.

If an option needs an argument (arg), you can include or omit the equals sign. Options that are flags (they switch an option on or off) require 0 (off) or 1 (on) as an argument.

3.1 Basic Options

-i / ––input The input (query molecule) is given using the -i or ––input argument. The file can contain one or more queries in sdf, SMILES, or mol2 format.

Example: `FTrees -i myquery.sdf`

Instead of entering a filename, you can also enter a SMILES string directly. This must be surrounded by double quotes.

-s / ––searchFiles The molecules that are to be searched with the query/queries are given as arguments here. Also multiple files can be used. And even libraries (enumerated collections of molecules) and chemical spaces (cf. above) be mixed:

Examples:

```
FTrees -s mylibrary.sdf

FTrees -s myFIRSTlib.sdf mySECONDlib.sdf

FTrees -s myLIBRARY.sdf mySPACE.space

FTrees -s mylibrary.smi

FTrees -s mychemicalspace.space

FTrees -s mychemicalspace.fsf
```

-o / ––outputFiles Output will be written either as SDF or in CSV format — or both. Specify the type of output by the file extension. The SDF format will contain the similarities in respective SD data fields. The CSV file will contain the similarities in a comma separated ASCII file and include SMILES for the molecules.
NOTE: If you have multiple queries in your input file, then one csv file will be written per query!

Examples:

```
FTrees -o myoutput.sdf
```

```
FTrees -o myoutputtable.csv
```

```
FTrees -o myoutput.sdf myoutputtable.csv
```

The latter example writes both, an sdf and a csv file.

-m / --matchImageBaseFile This will generate (one per match, so potentially many!) output images that explain the matching of query versus hit in 2D pictures. Depending on the file extension, PNG, PDF, or SVG, the images will be written as png, pdf, or vector-based svg files.

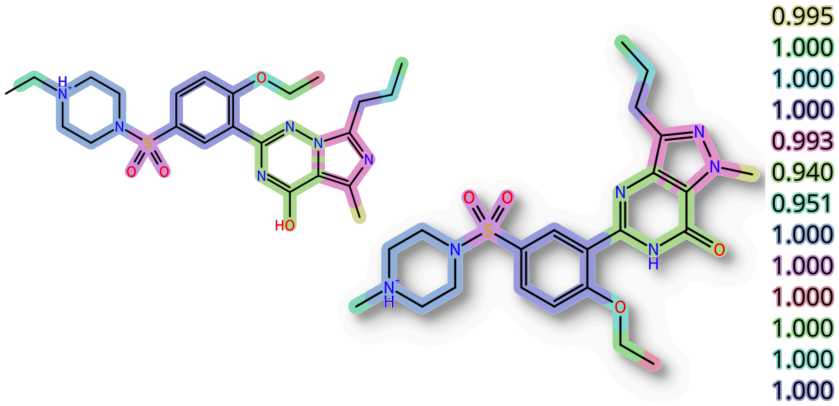
Example:

```
FTrees -m pix.png
```

This call will generate *one* png file per hit molecule(!), the file names will look like:

`pix_query_hitnumber_(someName).png`.

Here is a matching as it is displayed in the graphical interface infiniSee:



The local similarities are listed and color-coded on the right hand side. You can easily understand why the computer has computed these molecules to be similar: The ethyl-methylene on the left hand side matches a methyl, piperazinyl to piperazinyl-, sulfone to sulfone, and so on.

--gen2dOutput If you do not need 3D coordinates, but would still like 2D coordinates for 2D sketches for example, you can let FTrees generate 2D coordinates in your output SD file by switching this option on. If you set this value to zero (off, the default setting), the generation of 2D coordinates will be switched off, and all x, y, z coordinates will contain zeroes — in addition, you will save a little bit of time that may be relevant to you if you are dealing with very large quantities of molecules.

Examples:

```
FTrees --gen2dOutput 1

FTrees --gen2dOutput=1
```

--gen3dOutput If you would like to use the results for docking or 3D visualization or some other purpose that requires 3D input, FTrees can automatically generate 3D coordinates in the output SD files if you switch this option on..

Examples:

```
FTrees --gen3dOutput 1

FTrees --gen3dOutput=0
```

--thread-count This specifies the amount of parallelization across availble nodes on your machine during the calculation. This is useful if you work on a machine with other users and you would like to constrain FTrees to not use the entire machine. By default there is no need to change the parameters here, because FTrees will use all the nodes (threads) available on your computer.

Examples:

```
FTrees --thread-count 4

FTrees --thread-count=1
```

--maxNofResults This is often needed to limit the number of results that will be output. The default is 100. The results will always be sorted by the Feature Tree similarity, so the parameter controls the TOP number of results.

A maximum of one million results can be written out.

Examples:

```
FTrees --maxNofResults=1000

FTrees --maxNofResults 1
```

––minSimilarityThreshold This is also often a handy option as it lets you limit the results to those exceeding a minimum similarity. For example, if you are only interested in very highly similar molecules, then you may constrain the similarity to be above 0.95. Remember that Feature Trees are "late talkers", i.e., a similarity of 0.3 or 0.5 does NOT mean that half of the molecule is similar. Instead, since the similarity is fuzzy, a tangible similarity starts around 0.8; chemists will agree on a more "obvious" similarity in ranges around 0.85 and higher. You may want to play with this parameter to find the best range for your purposes.

Example:

```
FTrees --minSimilarityThreshold=0.95
```

––targetSimilarity This is more relevant for chemical space searches where the typical use case is to find more or less similar molecules from vast amounts of tangible molecules. For example, in later stages of projects, one would like to stay comparably close to a query molecule (say 0.95 or above), whereas in earlier stages one would like to enforce pronounced scaffold hops (say 0.85-0.9). With standard searches, i.e., query against a library of molecules, the parameter leads to analogous behavior: Those results that are closest to the target similarity will be at the top of the output list.

Note that '––targetSimilarity' has to be **>=** '––minSimilarityThreshold'

Example:

```
FTrees --targetSimilarity=0.9
```

3.2 Advanced Options

––expandAlternativeResults This may be relevant in cases where you search against a chemical space, i.e., a reaction-driven universe of potential molecules. Imagine two reactions can form a C-C bond, then setting this parameter to

1 will deliver the same results twice, one per reaction. Similarly, when matchings (cp. above) are found twice, both will be written out. Usually this flag is not modified.

Example:

```
FTrees --expandAlternativeResults=1
```

--totalDiversity This enforces diversity between all molecules in the set of results by allowing only a maximum similarity between any two molecules contained in the set. Consequently, the smaller the value, the more diverse the set will be in the end.

To do this, pairwise similarities must be computed between all molecules in the set of results. Therefore, use it only with small sets. For reasons of practicality, this parameter is limited to the range from 0.9 to 1.0.

Note: Only available if –maxNofResults' is **<=** 500. WARNING: any value below 1.0 will drastically affect the run time.

4 Examples

Below are several examples for typical use cases. Please not that the line breaks have been inserted for typesetting and readability reasons only; you must type in the full commands in one single line.

Run a similarity calculation for one query molecule versus an SD file library; write the top 200 most similar molecules into a new SD file:

```
FTrees  -i myquery.sdf -s mylibrary.sdf --maxNofResults 200
                                                    -o myoutput.sdf
```

Run a similarity calculation for one query molecule versus an SD file library; write the top 10 most similar molecules into a new SD file and create a set of png images to explain the similarity:

```
FTrees -i myquery.sdf -s mylibrary.sdf -o myoutput.sdf
               --maxNofResults 10  --matchImageBaseFile mypix.png
```

Navigate a chemical space with a query molecule and write out the top 10,000 compounds that are most similar into a SMILES file for further processing:

```
FTrees -i myquery -s MyChemical.space -o mytop1K.smi
                                              --targetSimilarity=1.0
```

The chemical space files can be obtained from

`https://www.biosolveit.de/infiniSee/#chemical_spaces`

If you should want to create your own chemical space as Pfizer, Boehringer Ingelheim, AstraZeneca, Merck and others do, please get in touch with us, and we will be happy to introduce you to our CoLibri suite of tools — or, alternatively, our Services department will be glad to do this for you.

5 Further Reading

- The original ideas behind the FTrees method are covered in the original publication by Matthias Rarey and J. Scott Dixon, JCAMD 12, 471-490, 1998;
- The dynamic match search is described by Marc Zimmermann et al., 2003, in: 14th European Symposium on Quantitative Structure-Activity Relationships, (Van de Waterbeemd, H. ed.), Blackwell Publishing.
- Chemical space navigation with Feature Trees descriptors in the FTrees program has been published by Matthias Rarey and Martin Stahl in JCAMD 15, 27-520, 2001; and
- A review on chemical space searching has been published by Torsten Hoffmann and Marcus Gastreich in DDT 2019 (https://doi.org/10.1016/j.drudis.2019.02.013).
- CoLibri is our software to generate chemical spaces from, for example, your own in-house synthons and reactions. `biosolveit.de/products/#CoLibri` has more information.

More on our other complementary tools, especially also the graphical platform infiniSee, can be obtained from the BioSolveIT website (biosolveit.com).

We wish you great success and much joy with FTrees!