User's guide

1. Detailed installation instructions
2. Interpretation of the output and Discussion

These instructions were written for MacOS X but should work fine with little of no modification for all UNIX-type systems.

1. Detailed installation instructions
(the user with previous UNIX experience may find the below to be annoyingly detailed)

- In order to run this script, you will need to have sudo/superuser privileges ("root / Admin password") for your Mac. You will similarly have to install the Apple Xcode package available on your MacOS X System DVD in order to be able to compile source code. Please talk to your system administrator if you feel unsure about these steps. Note that they are mandatory and that you should not proceed unless these criteria are fulfilled.
- The below will use the directory `/Users/henrik/temp/` as a temporary directory for various file operations. You will probably find that you don't have such a directory on your computer since it is user specific (and my username is `henrik`). Open a Terminal, go to your home directory with `cd ~` and create a folder called `temp` there with `mkdir temp` followed by enter. Enter the directory with `cd temp` followed by enter. Type `pwd` followed by enter. It will show you something like `/Users/your_account/temp/`. So when it says `/Users/henrik/temp/` below, think `/Users/your_account/temp/` instead.
- To install the HMMER package, I went to `http://hmmer.janelia.org/#documentation` and downloaded `ftp://selab.janelia.org/pub/software/hmmer/CURRENT/hmmer-2.3.2.tar.gz` which I put in `/Users/henrik/temp/` and unpacked: `tar xvfz hmmer-2.3.2.tar.gz` followed by enter.
- I enter the new directory with `cd hmmer-2.3.2` and follow the instructions in the file INSTALL such that I type `./configure` and press enter; then I type `make` and press enter; then I type `make check` and press enter; and finally I type `sudo make install` and press enter. The HMMER package should now have been compiled and installed on your computer; you can check this by typing `hmmpfam -h` and press enter - you should now see HMMER output. Note that you might have to change the above file names if you're installing another version of HMMER than 2.3.2 (the script remains untested with the experimental generation 3 of HMMER). [On one Mac I tried this on, the `make install` step failed for some obscure reason. As long as the preceding steps worked fine, this should be no problem, though you will have to copy the `hmmpfam` binary to `/usr/bin/` by hand: `sudo cp src/hmmpfam /usr/bin/` followed by enter.]
- The zip archive `http://www.emerencia.org/FungalITSextractor.zip` contains the fungal pipeline. Download it to, say, `/Users/henrik/program/` , go there by `cd /Users/henrik/program/` and extract the zip file with `unzip FungalITSextractor.zip` followed by enter. A folder called `FungalITSextractor` will be created. Enter it with `cd FungalITSextractor` followed by enter.
- The file `indata/indata.fasta` is where you place your query sequences; by default this is the only admissible file name for input sequences (note that the software comes bundled with a demonstration file with that very name). Use the FASTA format (see example below). It is advisable to stick with the generic FASTA format and to avoid, e.g., overly long or identical sequence names (and names with characters such as spaces, tabs , ; . | - and the like). The parser

included is tolerant but there are obviously limits to what it can do. If you would just like to take the script for a spin, you could very well use the `indata/indata.fasta` that comes bundled with the script - that is, no further action needed. The bundled FASTA file contains some 1600 environmental fungal ITS sequences.

Example of the FASTA file format:

```
>FN396389_uncultured_ectomycorrhizal_fungus
CCGAATTGTCAAACACGGGTTGTTGCTGGTCCCCAGATGGGGACACGTGCACGCTCTGTTTACACATCCACTTACACCTGTGC
ACCCTCTGTAGTTCTATGGTCTGGGAGACTCTGTCTTCCTTCTGTAGTCCTGCGTCTTTACACGTACGCCGTAAAAAAGTCTT
ATGGAATGTTTGCCGCGTTTAACGCAATACAATACAACTTTCAGCAACGGATCTCTTGGCTCTCGCATCGATGAAGAACGCAG
CGAAATGCGATAAGTAATGTGAATTGCAGAATTCAGTGAATCATCGAATCTTTGAACGCACCTTGCGCCCCTTGGCTATTCCG
AGGGGCATGCCTGTTTGAGTATCATGAACACCTCAACTCTCATGGTTTGCCATGATGAGCTTGGACTCTGGAGGTTTTGCTGG
TCTTTCCGGTCAGCTCCTCTCAAATGAATCAGCCTGCCGGTATCTGGTGGCATCATGGGTGTGATAACTATCTACGCTCAGAG
CCGTCCACTGGGCAACCTCCGGCGATGGAGGTTTGCTGGGGCTCACAAACGTCTCTCTTCAGTGGGGACAGCTTTTTGAACGT
TCGATCTCAAATCAGGTAGGACTACCCGCTGAACTTAAGCATATCAATAAGCGGAGGAAAAGAAACTAACAAGGATTCCCCTA
GTAACTGCGAGTGAAGCGGGAAGAGCTCAAATTTAAAATCCGACGTGCCTCTGGTCGTCCGAGTTGTAGTCTGAAGAAGCGTT
TTCCGTGTTGGGCCGTGTATAAGTCTCTTGGAACAGGGCGTCATAGAGGGTGAGAATCCCGTCTTTGACACGGACTACCAATG
CTTTGTGATGCGCTCTCAAAGAGTCGAGTTGTTTGGGAATGCAGCTCAAAATGGGTGGTAAATTCCATCTAAAGCTAAATATA
GGCGAGAGACCGATAGCGAACAAGTACCGTGAGGGAAAGATGAAAAGCACTTTGGAAAGAGAGT

>DQ309238_uncultured_fungus
AAGAATACCGGCTTCGCAGCCAATATTCTCGCCCTCTGTATACCAAACTTTGTTGCTTTGGCAGGCCCCTGGCTTCAGCTGGA
CTGTGCCTGCCAGAGGACCCCTAACTCTTGAATGCCTGCTGTCTGAGTACTATCTAAATAGTTAAAACTTTCAACAACGGATC
TCTTGGTTCTGGCATCGATGAAGAACGCAGCGAAATGCGATAAGTAATGTGAATTGCAGAATTTAGTGAATCATCGAATCTTT
GAACGCACCTTGCACCCTTTGGTATTCCGAAGGGTACACCCGTTTGAGTGTCATTGTAATCTCACCTCTTCGACTTTTTTTAT
GGTCGATCAGTGTGGACTTGGATGTTGCTGTGTAACAAACGGCTCGTCTGAAATGCCTTAGTGTACCCTGCTTTGCGGCGTAT
TCGGGTGTGATAAATCTTCACCGGAGTTTCGACTTTCGGGTCGGCGCTTGTAATGTTTTGGCTCTATGCTTCGAACCGTCCCT
AACAAGGACAAACTATACTCTGACAATTTGACCTCAAATCGGGGTGGGACTACCCGCTGAACTTTAA

>EU444545_Tomentellopsis_sp_EDM59
CTTGGCGGGTTAAGCGCCCGCCAGAGGACCCAACATTCCTGATTTTAATGTTTTTTTGGGACCTTATAATATAGTTAAAACTC
TCAACCGCGGATCTTTTGTTTGGGGTATCGAAGGAGAGCACAGAGAAATGATATAAATTATGTGAATTAGAGAATACAGAGTA
ATATAGTATTTGAGAACACACTTTGCGCCCCTTGGTTTTCAGAGGGGCACCCCTTTTAGTGAGTCATTACCCCCCTCAAACAA
TGCTTGGTTGGGGGTTGCGGCGCTCACCCAGGGGGCTTAAAAGACAGGGGGGGCGCTGTCGGCCCGTGAGTGTAATAAACCTT
CTCGACATAGGGACGATAGGGGCGGGGTCCCTCAACCCCCTCACTTCTAAGTTGACCTC
...
```

- Run the script: `perl FungalITSextractor.pl`
- You will now see output on your screen. The output is intended for the user to get a sense of progress of the script – all important information is saved to files. See below for more information on the output files.

2. Interpretation of the output and Discussion

- Once you have put your query sequences [or if you feel happy with the default test sequences] in `indata/indata.fasta`, you can start the script with `perl FungalITSextractor.pl`. Various information will be output: whether `hmmpfam` was found, the location of the output files, an entry-by entry brief summary of the results, and a summary of the results of the entire run. It is nevertheless the files that are the true end product of the run – the screen output is primarily meant for convenience.

```
Nilsson, Abarenkov et al. 2009. An open source software package for automated ITS1 and ITS2 extraction from fungal ITS sequences.
 <OK> Found HMMER.

Located infile as indata/indata.fasta.
All extracted ITS1 sequences will be saved as FASTA to outdata/2009-9-25-9/ITS1.fasta
All entries for which ITS1 could not be detected will be saved as FASTA to outdata/2009-9-25-9/NoITS1.fasta
All extracted ITS2 sequences will be saved as FASTA to outdata/2009-9-25-9/ITS2.fasta
All entries for which ITS2 could not be detected will be saved as FASTA to outdata/2009-9-25-9/NoITS2.fasta
Entries for which both of ITS1 nor ITS2 were extracted will be summarized as outdata/2009-9-25-9/Both.txt
Entries for which neither ITS1 nor ITS2 were extracted will be summarized as outdata/2009-9-25-9/None.txt
```

Fig. 1. Header of the output. Here the locations of the destination files are shown. In this case HMMER was detected – the program will abort if it cannot find the HMMER component `hmmpfam`.

```
(1 of 54) EF634387Penicillium_... 1144 bp.      ITS1    ITS2                  ITS1: 0-186     ITS2: 344-509
(2 of 54) AY436479Amanita_yuan... 495 bp.       ITS1    ITS2                  ITS1: 0-167     ITS2: 326-end
(3 of 54) AY014233Microbotryum... 542 bp.       ITS1    ITS2                  ITS1: 0-134     ITS2: 292-end
(4 of 54) AB071801Coprinopsis_... 716 bp.       ITS1    ITS2                  ITS1: 55-294    ITS2: 453-658
(5 of 54) EU266113Cladonia_ran... 747 bp.       ITS1    ITS2                  ITS1: 248-489   ITS2: 648-end
(6 of 54) EU798704Mucor_hiemal... 321 bp.       ----    ITS2                  -----           ITS2: 85-263
(7 of 54) AF453845Cladonia_cer... 562 bp.       ITS1    ITS2                  ITS1: 0-240     ITS2: 399-end
(8 of 54) AM269811Stereum_rugo... 649 bp.       ITS1    ITS2                  ITS1: 54-252    ITS2: 411-608
(9 of 54) AF325574Cortinarius_... 652 bp.       ITS1    ITS2                  ITS1: 18-253    ITS2: 412-614
(10 of 54) AY781270Sistotrema_b... 556 bp.      ITS1    ITS2                  ITS1: 0-160     ITS2: 319-534
(11 of 54) AY762364Fusarium_sub... 863 bp.      ----    ITS2                  -----           ITS2: 96-248
(12 of 54) AB249009Emericella_c... 593 bp.      ITS1    ITS2                  ITS1: 56-208    ITS2: 367-533
(13 of 54) DQ394387Parmelia_lae... 659 bp.      ITS1    ITS2                  ITS1: 37-231    ITS2: 390-540
(14 of 54) AB354787Phakopsora_e... 402 bp.      ----    ITS2                  -----           ITS2: 157-end
(15 of 54) AM882745Inocybe_giac... 351 bp.      ITS1    ITS2                  ITS1: 0-20      ITS2: 179-end
(16 of 54) EF197077Podospora_fi... 598 bp.      ITS1    ITS2                  ITS1: 33-234    ITS2: 393-560
(17 of 54) AY180261Stachybotrys... 546 bp.      ITS1    ITS2                  ITS1: 32-190    ITS2: 349-524
(18 of 54) EU623641Gerronema_st... 661 bp.      ITS1    ITS2                  ITS1: 0-270     ITS2: 429-end
(19 of 54) EU669207Clavariadelp... 885 bp.      ITS1    ITS2                  ITS1: 67-184    ITS2: 344-529
(20 of 54) S75443Scedosporium_p... 231 bp.      ----    ----                  -----           ----
(21 of 54) AF141631Phlebia_subs... 1203 bp.     ----    ITS2                  -----           ITS2: 86-263
```

Fig. 2. The query sequences are processed sequentially and are numbered according to their order in the input file. **The first column** is a simple progress report that shows what query sequence is currently being processed out of the total number of query sequences. The **second column** shows the name of the query sequence (truncated to fit the screen format, if applicable). The third column shows the length of the query sequence. If the **fourth column** says "ITS1" then ITS1 was extracted (in whole or in part) from the query; similarly, if the **fifth column** says "ITS2" then ITS2 was extracted (in whole or in part) from the query. In analogy, for both of these columns, "----" means that ITS1 and/or ITS2 were not extracted from the query. The **sixth column** shows the absolute position of ITS1 in the query sequence and the **seventh column** shows the absolute position of ITS2. Note that if a query features only half of ITS1 (ie starts in the middle of ITS1), its position will be give as 1-(say) 240. If the extremes of the boundaries coincide with those of the query sequence, then it is very likely that there would have been more sequence data for ITS1 and ITS2, had the query sequence been longer to begin with. The **eight column**, where present, is used to indicated that the query sequence is reverse complementary (note: the 5' end of 5.8S is needed for this detection). The sequence will be given in the correct way in all output files (if, that is, the 5' end of 5.8S was available in the sequence to begin with.)

```
------------------------------------------------------------
Total number of sequences in input file: 54

ITS1 and ITS2 were extracted from 47 entries.
Only ITS1 was extracted from 0 entries.
Only ITS2 was extracted from 6 entries.
Neither ITS1 nor ITS2 were extracted from 1 entries.

End of output.
Run started: Fri Sep 25 18:03:39 2009
Run ended: Fri Sep 25 18:03:43 2009
```

Fig. 3. Summary statistics for the run (when completed).

- A unique output directory will be created for each run in the directory `outdata`. This unique directory will bear the name of the date (with an appended number: 1 for the first run that day, 2 for the second etc). So for the first run you did on August 10, 2009 you will find the output files in `outdata/2009-8-10-1/`. Enter this directory.
- The file `outdata.csv` is a tab-separated file that contains the most important parts of the screen output. Open it in Excel or OpenOffice Calc - you will probably have to specify that tabs were used as field delimiter.
- The file `Both.txt` contains the names of all query sequences for which both ITS1 and ITS2 were found (in full or in part).
- The file `Both.fasta` contains the ITS1, 5.8S, and ITS2 (ITS1 and ITS2: in full or in part) of the above sequences in the FASTA format. Any portions of nSSU and nLSU have been removed, but the 5.8S is still in the sequence.
- The file `None.txt` contains the names of all sequences for which neither ITS1 nor ITS2 were found.
- The file `ITS1.FASTA` is a FASTA format file of all ITS1 sequences found.
- The file `ITS2.FASTA` is a FASTA format file of all ITS2 sequences found.
- The file `NoITS1.FASTA` is a FASTA format file of all sequences (full sequence data) for which ITS1 was not found.
- The file `NoITS2.FASTA` is a FASTA format file of all sequences (full sequence data) for which ITS2 was not found.

Some observations:

- The script will normally be able to find nSSU and nLSU as short as 18 bp. It will not find segments shorter than that. And it will have problems with such short segments if they are of poor read quality or if they come from deviant taxa. The longer HMMs are less sensitive [though not entirely immune] to such problems.
- It is not a good idea to use whole genomes or very large sequences as indata to the script. This increases the chances of false positives (particularly for the shorter HMMs). Though the script takes measures against false positives, some cases will probably be impossible to spot in an automated way.
- The HMMs will not perform very well on sequences of poor read quality / many IUPAC ambiguities. Poorly read base-pairs are unfortunately not uncommon near the 5' and 3' ends

of sequences – which also are the regions you are (probably) after.

- The default settings of the script were tailored to suit full-length ITS sequences – 500-800 bp. They seem to work fine on shorter sequences too. But before you run it in your sharp data, you should convince yourself that the settings are fine for your purposes. Do this by running the script on the first 10 of your sequences and see what comes out. **Please take the time to look at how the default settings perform on your sequences!**
- Hibbett et al., 1995, *Mycologia* 87: 618-638 shows the arrangement of the fungal ITS region in a detailed way. Compare the results with this publication to see if ITS1 and ITS2 were extracted correctly.
- If you find that ITS1/ITS2 are not located when in fact they should have been, you may need to adjust the HMMER E-values to be more allowing. Run `hmmpfam` by hand for a few of your sequences to see at what E level the boundaries in question are found, and then change the script accordingly. Note that you may still want the E-values to be as unforgiving as possible in the interest of a low number of false positives.
- Some taxa, such as *Cantharellus*, *Craterellus*, and *Tulasnella*, have very deviant ribosomal genes. The script can be expected to perform suboptimally on these. If you find this to be the case, you may have to compile special HMMs for these only, and use these HMMs only for those taxa. In our experience it is not a good idea to force deviant sequences into wide-scope HMMs since you will detract from the usability of the HMMs on other taxa (much like you distort phylogenetic signal when you force such sequences into otherwise satisfactory multiple alignments; Moncalvo et al., Mycologia 98, 2006).
- Please note that the software performs poorly on *Microsporidia*. These have very deviant ITS region and must be analysed with tailor-made HMMs (not "all of fungi"-HMMs).
- In our primary evaluation, we detected 4/488 (=0.82%) false negatives and 5/2000 (0.25%) false positives. We ask the user to understand that we have made every effort to ensure an as good-as-possible performance of the HMMs on "all-of-fungi" datasets, but that there will be some cases of incorrect extractions (though the script tries to warn against false positives insofar as their detection is amenable to algorithmic interpretation).
- Again, please keep in mind that it is important to evaluate how the default settings perform on your sequences. The defaults work fine for the average fungal ITS sequence in GenBank (=40 bp SSU, ITS1, 5.8S, ITS2, 40 bp LSU), that much is true, but if your sequences differ in length or coverage, you might have to tweak the settings to get the most out of the software.