ShadowCaster Documentation

Release 0.9.2

Daniela Sanchez, Aminael Sanchez

Jun 10, 2020

Contents

ShadowCaster implements an evolutionary model to calculate Bayesian likelihoods for each 'alien genes' with an unusual sequence composition according to the host genome background to detect HGT events in prokaryotes.

CHAPTER 1

ShadowCaster analysis workflow

1. The user defines a query genome (by providing two fasta files, see *Usage*) in which HGT events will be detected.
2. ShadowCaster uses a list of proteomes from phylogenetically related species to the query genome (one proteome FASTA file per species) to construct a phylogenetic shadow.

   The list of proteomes could be either: -Provided by the user (a collection of FASTA files).

   -Automatically retrieved by ShadowCaster from the NCBI ftp site by using `script/get_proteomes.py`, see *get_proteomes.py*.
3. A prioritized list of potential 'alien genes' present in the query genome is generated by the analysis of compositional features i.e. 4mers and codon usage. An unsupervised one-class support vector machine is used for the prioritization task.
4. Orthology relationships among the query genome and its phylogenetically related species are obtained by the third-party algorithm ORTHOMCL. This information is used to calculate the 'probability of orthology' between the query genome and each other genome in the phylogenetic shadow.
5. BLAST is used to calculate the identity between each alien gene and the rest of genes in the genomes of the phylogenetic shadow.
6. A likelihood is calculated for each alien genes in the list from step 3. The likelihood expresses how likely is that the pattern of identity across genomes in the phylogenetic shadow for this alien gene derives from vertical inheritance.

CHAPTER 2

Installation

For a comprehensive guide on how to install ShadowCaster and its prerequisites, see *Installation*.

CHAPTER 3

Contribute

- Issue Tracker: issues
- Source Code: code

CHAPTER 4

Support

Send additional enquiries to asanchez2@utpl.edu.ec

CHAPTER 5

License

GNU General Public License Version 3

CHAPTER 6

Contents

6.1 Installation

6.1.1 Prerequisites

Fundamental prerequisites

```
python => v2.7.10
R => v3.4
perl => v5.10
```

These items are prerequisites for the installation of ShadowCaster as described below.

Packages

- Install python packages using:

```
pip install numpy scipy biopython pandas ete3 scikit-learn matplotlib
```

- R packages: `cluster` v2.0.*
- Install figlet. For Linux (Mint 18.3) use:

```
apt install figlet
```

Other dependencies

For using the phylogenetic component, some programs are required and should be in the PATH:

- BLAST 2.2.26+ (blastp needed).

If you have a newer version of BLAST already installed, you just can copy the path to the executable file for blastp in the configuration file of ShadowCaster.

- OrthoMcl pipeline

OrthoMcl needs blast v2.2.26(blastall and formatdb), these binary files can be found here. Download blast-2.2.26-*.tar.gz.

For Linux (Mint 18.3) this can be installed through:

```
apt-get install blast2
```

More information, see suggestions.

- Package emboss.

For Linux (Mint 18.3) this is installed through:

```
apt-get install emboss
```

To check that these dependencies were correctly added in the path, run:

```
type orthomcl-pipeline blastall formatdb
```

6.1.2 Installation of ShadowCaster

To use ShadowCaster, download it from the GitHub repository and extract the files. If you have git installed, you can install ShadowCaster by running:

```
cd
git clone https://github.com/dani2s/ShadowCaster.git
```

Resolve all dependencies, see above and then execute:

```
cd ShadowCaster/
python setup.py install
```

This will install ShadowCaster under your home folder.

6.2 Usage

ShadowCaster uses a configuration file (args.ini) to manage all the options needed. This file must be specified on the command line and will supply the following arguments:

6.2.1 Files

- **Query genome** *Only fasta files.*
- **Query proteome** *Only fasta files*
- **OrthoMcl configuration file**, previously obtained with the installation of OrthoMcl-pipeline(orthomcl.conf). This file looks like:

```
coOrthologTable=CoOrtholog
dbConnectString=dbi:mysql:orthomcl:localhost:mysql_local_infile=1
dbLogin=orthomcl
dbPassword=orthomcl
dbVendor=mysql
evalueExponentCutoff=-5
inParalogTable=InParalog
interTaxonMatchView=InterTaxonMatch
oracleIndexTblSpc=NONE
orthologTable=Ortholog
percentMatchCutoff=50
similarSequencesTable=SimilarSequences
```

6.2.2 Path

- **Proteomes folder, contain the proteomes(*Only fasta files*) of each species to construct the shadow.**
  -Provided by the user (a collection of FASTA files)

  or

  -Automatically retrieved by ShadowCaster from the NCBI ftp site by using `script/get_proteomes.py`, see *get_proteomes.py*.
- **Blastp26** ShadowCaster uses blastp 2.2.26, specify the binary file or the shell command used
- **Formatdb** Specify the binary file or the shell command used. Ex. formatdb

6.2.3 Parametric

- **nuSVM** A bound between the fraction of training errors and the fraction of support vectors. Should be in the interval (0, 1].

A template of the args.ini file can be found in the bin folder.

Specifications of fasta files

The id of each gene in the GENOME fasta file SHOULD not contain any character like `|, #, %, /, \, *, &, $, !, :`. It is preferred only one identification number.

All the proteomes files provided by the user MUST follow these specifications:

- Fasta file name ONLY can be the binomial name of the species (Rhodanobacter_denitrificans.fasta).
- Each id record of a fasta file MUST have only one id number or the following structure:

```
>AGG91012.1 ribosomal protein L34 [Rhodanobacter denitrificans]
MKRTFQPSKLKRARTHGFRARMATADGRKVLNARRAKGRKRLIP
```

Examples of the input files can be found in the test data repository of ShadowCaster, see here

6.2.4 Run ShadowCaster

```
cd $ShadowCaster/bin/
shadowcaster --config_file args.ini
```

6.3 Example

This documentation aims to be a complete example walk through for the usage of ShadowCaster. It assumes you have successfully gone through the *Installation*.

6.3.1 Downloading test data

Download and extract the test data repository of ShadowCaster in a suitable location, see here

6.3.2 Software specifications

The results provided in the test data repository were obtained running ShadowCaster with the following software versions.

O.S Linux Mint 18.03 (Sylvia)

Package base Ubuntu Xenial

Python v2.7.15

R v3.4.4

Perl v5.22.1

mysql v5.7.23

Python packages

- `pip` v18.0
- `numpy` v1.14.3
- `biopython` v1.72
- `matplotlib` v2.2.2
- `pandas` v0.23.0
- `scipy` v1.1.0
- `scikit-learn` v0.19.1
- `ete3` v3.1.1

R package

- `cluster` v2.0.7-1

6.3.3 Run ShadowCaster

All fasta files are in the `shadowcaster-input` and `proteomes-output` folders. The arguments needed in the args.ini file are:

- query_genome = /home/user/path/to/shadowcaster-input/Rdenitrificans_genome.fasta
- query_proteome = /home/user/path/to/shadowcaster-input/Rhodanobacter_denitrificans.fasta
- proteomes_folder = /home/user/path/to/proteomes-output/proteomes/
- orthomcl_config = /home/user/orthomcl-pipeline/orthomcl.conf
- blastp26 = Binary file or your command line used of blastp v2.2.26.

- formatdb26 = Binary file or your command line used of blastall v2.2.26.
- nuSVM = 0.4

MUST use the full path of the files or directory.

Run ShadowCaster through:

```
cd $ShadowCaster/bin/
shadowcaster --config_file args.ini
```

When ShadowCaster has finished the message "ShadowCaster finished" is printed. The program generates a number of files in the output directory (called with the date and time of the running).

6.3.4 Output description

ShadowCaster generates the following files in the output directory.

log.txt Contains used parameters

Parametric folder

- `data4mer_chi2.csv` Compositional difference measured with chi2 between each gene and the genome based on 4mers .
- `kl_codonUsage.csv` Measure compositional difference of codon usage with Kullback-Leibler.
- `plot_aliens.png` Plot of the two compositional features.
- `alien_genes.fasta` Fasta file with the alien genes identified by one-class support vector machine.

Phylogenetic folder

- Orthomcl folder contains files of orthologs groups found by OrthoMCL.
- `orthologs_probabilities.csv` the probabilities of orthology between the query proteome and each other genome in the phylogenetic shadow.
- `alien_likelihoods.csv` The log likelihood calculated for each alien gene found in the parametric component.
- `histogram_alienLikelihoods.png` Histogram of likelihoods from alien genes.
- `hgt_candidates.csv` List of HGT candidate genes found with the software with their likelihood.
- `shadowcaster_predictions.fasta` Fasta file with genes predicted as HGT by ShadowCaster.

A copy of all the output files can be found in the `shadowcaster-output` folder.

6.4 get_proteomes.py

`get_proteomes.py` implements a method that we apply before finding HGT candidates with ShadowCaster. This script retrieves a list of proteomes from phylogenetically related species to the query species (fasta files) from the NCBI ftp. ShadowCaster needs these proteomes to construct a phylogenetic shadow used in its phylogenetic component.

6.4.1 Prerequisites

- EDirect UNIX command line of NCBI.

Before using the script, check that the commands esearch and xtract work correctly in a new shell window.

```
type esearch xtract
```

6.4.2 Usage

The usage and help documentation of `get_proteomes.py` can be seen by running `python get_proteomes.py -h`:

6.4.3 Example

An example of how to run `get_proteomes.py` on the test data:

```
cd ShadowCaster/scripts
python get_proteomes.py -n Rhodanobacter_denitrificans -sp 25
```

This results in the following output files in the folder named with the species name provided:

- `log.txt` Name of the downloaded species and its ftp address.
- `proteomes folder` Proteomes (fasta file) used to construct the shadow.

The results should be similar to those found in the `proteomes-output` folder of the test data repository, see here