PULL User Manual

1. Program description

PULL is the abbreviation of Plant Full-length lncRNA Identity. Which take advantages of sequencing for terminal signal (cap analysis gene expression, CAGE-seq or polyA-seq) to get a relatively complete transcript collection. Additionally, long read sequencing data can also be added to form a high credible collection. After get the full-length transcriptome, we combined several popular tools to evaluate the coding potential in order to identify the widespread long non-coding RNAs (lncRNAs).

The PULL pipeline accepts four types of high-throughput sequencing data, including second-generation sequencing (ssRNA-seq), 5 'cap sequencing (CAGE-seq), 3' polyA sequencing (polyA-seq), and third-generation sequencing (pacbio-seq). Among them, CAGE-seq, polyA-seq and pacbio-seq are not required inputs. In addition, the CAGE-seq or polyA-seq data provided must be consistent with the ssRNA-seq sample. In our process, transcripts assembled from ssRNA-seq are modified to precise transcription start site (TSS) or transcription end site (TES) based on CAGE-seq or polyA-seq.

To match a transcript with a TSS/TES supported by CAGE-Seq/PolyA-seq, we considered two factors in PULL pipeline. The first one is "dis" (Hon et al.), which means the distance between TSS/TES supported by CAGE-seq/PolyA-seq and the 5' end/3' end of the transcript. The second one is "cor" (Kawaji et al.), which means coefficient between Transcripts Per Million (TPM) of TSS from CAGE-seq and fragments per kb of exonic sequence per million mapped reads (FPKM) of the transcripts from ssRNA-seq across samples. For each assembled transcript, we kept all CAGE peaks between 500bp upstream of the 5 'end and the first exon. CAGE peak with the highest "cor" was selected as the TSS of the transcript. If there were multiple CAGE peaks with the same "cor," CAGE peak with the smallest "dis" was selected as the TSS of the transcript. The method to match a transcript with a TES supported by PolyA-seq followed the similar way.

All assembled transcripts were submitted to the following processing steps to get a set of lncRNA transcriptome: the known PCGs annotated in the genome and known ncRNAs in Rfam (tRNAs, rRNAs, snoRNAs, and snRNAs) were removed; the transcripts with lengths less than 200bp or ORF lengths more than 100 amino acids(aa) were removed; the transcripts encoding known protein in Swiss-Prot or Pfam were removed; the transcripts with Coding-Non-Coding Index (CNCI) score >0 (Sun et al.) or Coding Potential Calculator 2 (CPC2) score >0.5 (Kang et al.) were removed; to avoid the possible false-positive long noncoding natural antisense transcripts (lnc-NATs) caused by the contamination from sense transcripts or background from the ssRNA-seq protocol were removed; For lncNATs not having TSS or TES supported by CAGE-seq or PolyA-seq, we made use of intersectBed program in Bedtools with parameter -S -f 0.5 -F 0.5, which means that we will exclude the lnc-NATs , if only the overlap between lnc-NATs and PCG account for more than 50%.

2. Work environment preparation

The whole pipeline is worked on python3 environment, so we recommend a convenient tool called anaconda help you to build the work environment.

Step1: Download anaconda

```
wget https://repo.anaconda.com/archive/Anaconda2-2018.12-Linux-x86_64.sh
sh Anaconda2-2018.12-Linux-x86_64.sh
```

Step2: Create a new python3 work environment

```
conda create -n Py3_pull -c bioconda --file requirements_py3.txt python=3
```

In addition to the requirements satisfied by conda, here are also some softwares you are supposed to installed already.

ncbi-blast (v2.7.1+)

```
ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/2.7.1/ncbi-blast-2.7.1+-x64-linux.tar.gz
```

StringTie (v1.3.4d)

```
http://ccb.jhu.edu/software/stringtie/dl/stringtie-1.3.4d.Linux_x86_64.tar.gz
```

R (v3.4.2)

https://mirrors.tuna.tsinghua.edu.cn/CRAN/src/base/R-3/R-3.4.2.tar.gz

CAGEr (v1.22.3)

http://bioconductor.org/packages/release/bioc/html/CAGEr.html

3. Installation of PULL

```
git clone https://github.com/chenyanj/PULL.git
```

Unpack the compressed file, then PULL is ready for use!

4. Usage of PULL

Firstly, you are supposed to activate the work environment for Pull:

```
source activate Py3_pul
```

Then use the absolute path of Pull to call the tool

Pull.py 1.0.0 -- Plant full length lncRNA identity

Usage: Pull.py [options]

Options:

```
-h                          Show this screen.
-v                          Show version.
-g GENO                     Genome fasta file
-r REF                      Genome annotation file in gtf format.
-c CAGE                     A directory contain cage sample.ctss files
-p PAS                      A directory contain pas.ctss files
--ngs NGS_BED               A bed12 format file assembled by NGS sequencing data
--ngs_bam NGS_BAM           A directory contain all bam files for different samples.
--pb PACBIO                 A bed12 format file assembled by Pacbio sequencing data
```

5. Input Files needed

Reference files for target genome are obligatory, including sequence fasta file and annotation file in gtf format.

-g reference genome sequence in fasta format, we recommend you only keep chromosomal sequence except for other scaffold sequence.

-r reference genome annotation file in gtf format, the same as -g, only genes in chromosomal are supposed to be kept in order to save computation space.

-c A directory contain cage sample.ctss files, data for information of transcript start site(tss), such as CAGE-Seq. The input file is optional, if not provided, accurate 5'end will not be calculated. If provided, you are supposed to rename files in this directory as sample.ctss .

-p A directory contain pas.ctss files, data for information of transcript end site(tes), such as PAS-Seq. The input file is optional, if not provided, accurate 3'end will not be calculated. If provided, you are supposed to rename files in this directory as sample.ctes .

--ngs A bed12 format file assembled by NGS sequencing data, you should assemble all sample to a final merged.bed. Raw sequencing data after quality control are mapped to reference genome, then transcriptome file for every sample were merged to get the final input bed12 file.

--ngs_bam A directory contain all bam files for different samples. In addition to the merged file, alignment results for every sample in bam format are also necessary for further dispose. The parameter accept a directory path in which bam files are named as sample.bam. Finally, the library for NGS-Seq only be strand-specific library.

--pb A bed12 format file assembled by Pacbio sequencing data, the input file is optional. Raw sequencing data from pacbio is adjusted by NGS data, then fixed data is mapped to reference genome to get transcriptome file in bed12 format.

Provide tss or tes data should corresponding to ngs data, the sample name should be same. Ctss file are exampled bellow,

```
Chr01    27833    +    1
Chr01    28743    +    1
Chr01    28745    +    3
Chr01    28754    +    1
Chr01    28865    +    1
```

6. Explanation for output files

This pipeline will produce full-length lncRNAs collection in bed12 format which is classified by structural integrity to four types, including with-cap-and-polya, only-cap, only-polyA, other.

Reference

Hon, Chung-Chau, et al. “An Atlas of Human Long Non-Coding RNAs with Accurate 5′ Ends.” *Nature*, vol. 543, no. 7644, Nature Publishing Group, Mar. 2017, pp. 199–204, doi:10.1038/nature21374.

Kang, Yu-Jian, et al. “CPC2: A Fast and Accurate Coding Potential Calculator Based on Sequence Intrinsic Features.” *Nucleic Acids Research*, vol. 45, no. W1, Oxford University Press, July 2017, pp. W12–16, doi:10.1093/nar/gkx428.

Kawaji, Hideya, et al. “Comparison of CAGE and RNA-Seq Transcriptome Profiling Using Clonally Amplified and Single-Molecule next-Generation Sequencing.” *Genome Research*, vol. 24, no. 4, Cold Spring Harbor Laboratory Press, Apr. 2014, pp. 708–17, doi:10.1101/gr.156232.113.

Sun, Liang, et al. “Utilizing Sequence Intrinsic Composition to Classify Protein-Coding and Long Non-Coding Transcripts.” *Nucleic Acids Research*, vol. 41, no. 17, Narnia, Sept. 2013, pp. e166–e166, doi:10.1093/nar/gkt646.