Title

transXpress: a Snakemake pipeline for streamlined de novo transcriptome assembly and annotation

Permalink

https://escholarship.org/uc/item/40m1r17j

Journal

BMC Bioinformatics, 24(1)

ISSN

1471-2105

Authors

Fallon, Timothy R
Čalounová, Tereza
Mokrejš, Martin
et al.

Publication Date

2023

DOI

10.1186/s12859-023-05254-8

SOFTWARE

Open Access

transXpress: a Snakemake pipeline for streamlined de novo transcriptome assembly and annotation

Timothy R. Fallon[1], Tereza Čalounová[2], Martin Mokrejš[2], Jing-Ke Weng[3,4*] and Tomáš Pluskal[2*]

*Correspondence:
wengj@wi.mit.edu;
tomas.pluskal@uochb.cas.cz

[1] Scripps Institution of Oceanography, UC San Diego, 9500 Gilman Dr, La Jolla, CA 92093, USA
[2] Institute of Organic Chemistry and Biochemistry of the Czech Academy of Sciences, Flemingovo náměstí 2, 16000 Prague 6, Czech Republic
[3] Whitehead Institute for Biomedical Research, 455 Main Street, Cambridge, MA 02142, USA
[4] Department of Biology, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

Abstract

Background: RNA-seq followed by de novo transcriptome assembly has been a transformative technique in biological research of non-model organisms, but the computational processing of RNA-seq data entails many different software tools. The complexity of these de novo transcriptomics workflows therefore presents a major barrier for researchers to adopt best-practice methods and up-to-date versions of software.

Results: Here we present a streamlined and universal de novo transcriptome assembly and annotation pipeline, transXpress, implemented in Snakemake. transXpress supports two popular assembly programs, Trinity and rnaSPAdes, and allows parallel execution on heterogeneous cluster computing hardware.

Conclusions: transXpress simplifies the use of best-practice methods and up-to-date software for de novo transcriptome assembly, and produces standardized output files that can be mined using SequenceServer to facilitate rapid discovery of new genes and proteins in non-model organisms.

Keywords: De novo transcriptome assembly, RNA-seq, Non-model organisms, Transcriptome annotation, Differential expression analysis, Reproducible software, High-performance computing

Background

De novo transcriptome assembly of short-read RNA-seq data followed by prediction of open reading frames (ORFs) and automated annotation of predicted proteins is widely used for studying non-model eukaryotic organisms without a reference genome [1, 2]. The NCBI Sequence Read Archive (SRA) database currently contains over 3 million RNA-seq datasets, including hundreds of thousands from non-model eukaryotes [3]. These datasets represent a rich and continuously growing resource for diverse biological research across the tree of life. In contrast, only ~ 6900 eukaryotic transcriptome assemblies have been uploaded to the NCBI Transcriptome Shotgun Assembly (TSA) database to date, reflecting the difficulties in producing and uploading high-quality assemblies [4]. Generating and annotating a de novo transcriptome assembly requires numerous bioinformatic tools that can be difficult to install, and best practices are not always followed [5].

We surveyed existing pipelines for RNA-seq data analysis, including de novo transcriptome assembly and gene annotation tasks (Table 1). To date, four pipelines have been published for de novo transcriptome assembly, two of which (Rnnotator [6] and themira [7]) have been discontinued since their publication. Several other pipelines are available for aligning RNA-seq reads to a reference genome. Only a few of them support alignment of raw reads to a de novo assembled or reference transcriptome, depending mostly on the read aligner used. However, such pipelines generally were not designed to assist with gene discovery in non-model organisms. Presently, Pincho [8] is the only maintained pipeline that supports both de novo transcriptome assembly and transcript annotation using a variety of tools. However, Pincho does not support distributed computing on high-performance computational clusters (HPCs), and therefore has limited utility for processing large sequencing datasets.

Here, we present a new de novo transcriptome assembly pipeline, transXpress, which streamlines reproducible assembly of transcripts, quantification of transcript expression levels, and gene and protein prediction and annotation. transXpress also supports parallel execution on heterogeneous cluster computing hardware.

Implementation

Workflow engine

Older RNA-seq pipelines were typically implemented as shell scripts with the use of Perl, Python or R to execute the relevant downstream analyses. Recently, there is a strong tendency to employ bioinformatic workflow engines such as Snakemake, Nextflow or Galaxy [20–22]. Owing to its general simplicity and ease of use, we selected Snakemake to handle the dependencies between the executed tasks, to avoid repeated computations upon pipeline re-execution, and to support cluster computing [20]. The users of transXpress are advised to install required dependencies using Conda [23] and Python's PIP package management systems, as described on the transXpress GitHub page [24].

The transXpress pipeline (Fig. 1) performs parallel execution of the underlying tools whenever possible. Furthermore, it splits the input datafiles (e.g., for the Trimmomatic and the FASTA annotation steps) into multiple partitions (batches) to speed up even single-threaded tasks by parallelization. The partial results files from such split tasks are then merged automatically back into a single output file. In the case of the Trinity assembler, the individual jobs generated within Trinity by the 'Chrysalis' phase as input for the 'Butterfly' phase, are automatically parallelized by transXpress [25, 26]. The output files from all the underlying tools, including their graphical results, are retained in the project folder.

Data pre-treatment

The quality of the input sequencing reads has a major impact on the quality of the final transcriptome assembly [27]. To assess the quality of the provided reads, transXpress uses the FastQC tool [28]. Its wrapper add-on MultiQC [29] further aggregates and summarizes FastQC reports of all samples into a single report, providing an easy overview of

Table 1 Overview of existing pipelines for RNA-seq data analysis

Pipeline	Platform	Preprocessing	Assembly	Read mapping	Expression analysis	Functional annotation
transXpress	Snakemake	trimmomatic, FastQC, MultiQC	Trinity, rnaSPAdes	bowtie2 (optional)	kallisto, edgeR	BLAST, TargetP, SignalP, TMHMM, BUSCO
Pincho [8]	Bash, python3	trimmomatic, Rcorrector, TransRate, CD-HIT	Trinity, rnaSPAdes, BinPacker, IDBA-tran, Velvet-Oases, Shannon, Trans-AbySS, TransLig	HISAT2	kallisto, RSEM	BLAST, BUSCO, TransRate
RNAflow [9]	Nextflow	FastQC, MultiQC, fastp, SortMeRNA	Trinity	HISAT2	DESeq2	BUSCO, dammit
Rnnotator (unavailable) [6]	Unknown		Velvet, AMOS			
themira (unavailable) [7]	Unknown	FastXtoolkit, FastX, CAP3	Velvet-Oases			Blast2GO
nf-core/rnaseq [10]	Nextflow	FastQC, TrimGalore	(None)	STAR, HISAT2	RSEM, Salmon, DESeq2	
Pipeliner [11]	Nextflow	FastQC, MultiQC, TrimGalore	(None)	STAR, HISAT2	StringTie, HTSeq, featureCounts	
VIPER [12]	Snakemake	RSeQC	(None)	STAR	Picard, Cufflinks, RSeQC, ComBat, DESeq2, PCA	VarScan, Gostats, GAGE, Pathview, ClusterProfiler, STAR-fusion, TRUST, TIMER, virus contamination detection
RASflow [13]	Snakemake	TrimGalore, FastQC, MultiQC	(None)	Salmon, HiSAT2	featureCounts or htseq-count, Qualimap, edgeR, DESeq2	
hppRNA [14]	Snakemake	cutadapt, FastQC, PRINSEQ, FASTX-toolkit	(None)	Tophat, bowtie, subread, STAR, HiSAT	Cufflinks, featureCounts, RSEM, eXpress, kallisto, StringTie, ngs.plot, Cuffdiff, DESeq2, EBSeq, edgeR, sleuth, Ballgown	GATK, FusionCatcher
TRAPLINE [15]	Galaxy	FastxClipper, FastQC, FASTQ, FASTX-toolkit	(None)	Tophat, bowtie	Picard, Cufflinks, Cuffdiff	DAVID, miRanda, BioGRID
QuickRNAseq [16]	bash, Perl, R	RSeQC	(None)	STAR	featureCounts, RSeQC, edgeR	VarScan
ARMOR [17]	Snakemake	TrimGalore, FastQC + MultiQC	(None)	Salmon, STAR	edgeR, DRIMSeq	

Table 1 (continued)

Pipeline	Platform	Preprocessing	Assembly	Read mapping	Expression analysis	Functional annotation
BISR-RNAseq [18]	PBS, bash, shiny, R	FastQC + MultiQC	(None)	HiSAT2	Picard, featureCounts, RSeQC, limma, edgeR	
RNAseq123 [19]	Bioconductor		(None)		edgeR, limma, glimma	

The table summarizes the architecture and individual tools used in the pipelines for the main steps of data processing. Five of the pipelines (transXpress, Pincho, RNAflow, themira, Rnnotator) include a step of de novo transcriptome assembly, while the others require a reference genome or transcriptome

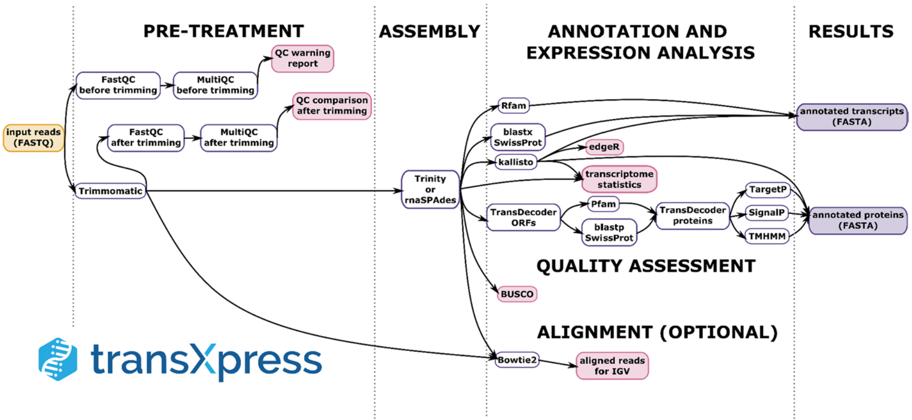


Fig. 1 A schema of the data processing steps performed by the transXpress pipeline. The input data are on the very left in a yellow-colored frame. Initial data pre-treatment tasks are on the left, followed by assembly and tasks executed largely in parallel (annotation and expression analysis). Output data types are in a purple background on the very right. This is a manually simplified version of the directed acyclic graph (DAG) of Snakemake tasks. The DAG can be automatically generated by Snakemake for each transXpress run

the quality of sample preparation, library construction, and sequencing across all samples. Such a report is fundamental for the subsequent interpretation of the data.

Sequencing adapters and poor quality reads are removed using Trimmomatic [30]. Trimming the reads is very important for de novo assembly, since artificially introduced sequences (various types of adapters and their dimers, multimers, partial copies, or PCR-based artifacts) may interfere with the extension of contigs. After read trimming, transXpress performs another round of FastQC/MultiQC quality assessment and checks the generated report for potential warnings.

de novo transcriptome assembly

Roughly ten de novo transcriptome assemblers for short RNA-seq reads have been developed and are in common use [31]. Among them, Trinity [25], rnaSPAdes [32] and TransAbyss [26], are the most widely used tools, and a recent evaluation indicated these three assemblers generally outperformed other tools [33]. All three utilize kmer-based De Bruijn graph assembly, which often requires a large amount of memory for the kmer frequency counting step. transXpress pools the sequencing reads for all provided samples and performs de novo assembly either using Trinity or rnaSPAdes, depending on the configuration settings provided by the user. Since these assemblers were primarily developed for high-quality short-read sequences, the range of supported sequencers includes Illumina, DNBSEQ, MGISEQ, or BGISEQ platforms, as well as older Roche/454 instruments [34]. transXpress does not support assembly from long-read sequencers such as PacBio or Nanopore. The assembled transcripts are further processed with TransDecoder [26] to identify likely protein-coding regions (ORFs). In case multiple potential ORFs are identified within a single transcript, TransDecoder reports all of them, leading to multiple protein sequences being subject to downstream annotation tasks in transXpress.

For each assembled transcriptome, transXpress reports simple statistics using scripts provided by the Trinity assembler (e.g., the number of assembled isoforms and genes, median contig length, contig Nx and ExN50 values) [35]. Further, transXpress runs the Benchmarking Universal Single-Copy Orthologs (BUSCO) tool to assess the completeness of the transcriptome by estimating completeness and redundancy in terms of expected gene content [36].

Expression analysis and transcriptome annotation

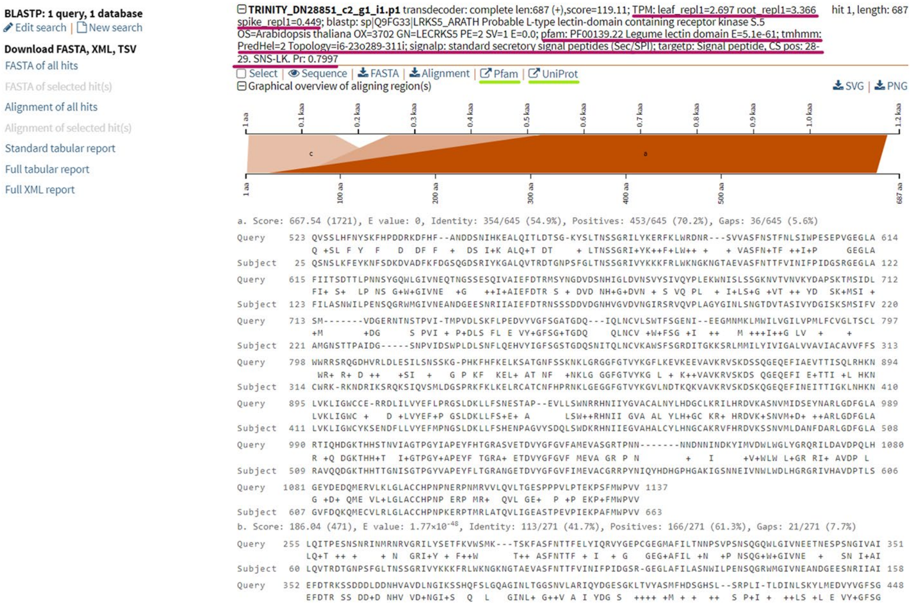
The underlying RNA-seq reads used for the transcriptome assembly are also used to estimate transcript expression levels (transcript-per-million or TPM values) using kallisto, a fast alignment-free method for near-optimal expression quantification at the transcript isoform level [37]. As an optional step, full read-to-transcript local alignments can also be performed using Bowtie2 [38], to allow for troubleshooting and manual inspection of read coverage, for example in Integrated Genomics Viewer [39]. If multiple samples are included, transXpress performs differential expression analysis using edgeR [40]. This step also generates graphical output in the form of heat maps with hierarchical clustering analysis, using Perl and R scripts provided by the Trinity assembler [26]. The information about sample groups for differential expression analyses is obtained automatically from the transXpress main input file *samples.txt*, which defines the sample groups, replicates, and paths to raw sequencing reads (FASTQ files) for each sample.

The assembled transcriptome is further decorated with automated annotations. NCBI BLAST + [41] searches (blastx and blastp) are performed against the curated UniProtKB/Swiss-Prot database [42]; hmmer3 [43] is used to search through the Pfam-A database of protein domains [44]; and cmscan from the Infernal package [45] is used to search the Rfam database of non-coding RNA sequences [46]. Moreover, transXpress uses SignalP 6.0 and TargetP 2.0 to predict N-terminal signaling and targeting peptides [47, 48]. A Python re-implementation of the widely used TMHMM algorithm is employed for prediction of transmembrane helices [49].

The resulting flat files are parsed via custom Python scripts and the collected annotations are used to decorate the output FASTA files with transcripts and predicted protein coding sequences.

Transcriptome mining

The most user-friendly way to mine the annotated FASTA files generated by transXpress is to use SequenceServer [50], which enables performing BLAST + [51] searches against custom FASTA sequence databases. For every hit, SequenceServer displays its alignment to the query and also the FASTA headers of each sequence, which include functional



BLASTP: 1 query, 1 database
Edit search | New search

Download FASTA, XML, TSV
FASTA of all hits
FASTA of selected hit(s)
Alignment of all hits
Alignment of selected hit(s)
Standard tabular report
Full tabular report
Full XML report

⊟ **TRINITY_DN28851_c2_g1_i1.p1** transdecoder: complete len:687 (+),score=119.11; TPM: leaf_repl1=2.697 root_repl1=3.366 spike_repl1=0.449; blastp: sp|Q9FG33|LRKS5_ARATH Probable L-type lectin-domain containing receptor kinase S.5 OS=Arabidopsis thaliana OX=3702 GN=LECRKS5 PE=2 SV=1 E=0.0; pfam: PF00139.22 Legume lectin domain E=5.1e-61; tmhmm: PredHel=2 Topology=i6-23o289-311i; signalp: standard secretory signal peptides (Sec/SPI); targetp: Signal peptide, CS pos: 28-29. SNS-LK. Pr: 0.7997 hit 1, length: 687

□ Select | Sequence | FASTA | Alignment | Pfam | UniProt
⊟ Graphical overview of aligning region(s) SVG | PNG

```
a. Score: 667.54 (1721), E value: 0, Identity: 354/645 (54.9%), Positives: 453/645 (70.2%), Gaps: 36/645 (5.6%)
Query     523 QVSSLHFNYSKFHPDDRKDFHF--ANDDSNIHKEALQITLDTSG-KYSLTNSSGRILYKERFKLWRDNR---SVVASFNSTFNLSIWPESEPVGEGLA 614
              Q +SL F Y  F   D  DF F  +  DS I+K ALQ+T DT    + LTNSSGRI+YK++F+LW++ +    + VASFN+TF ++I+P     GEGLA
Subject    25 QSNSLKFEYKNFSDKDVADFKFDGSQGDSRIYKGALQVTRDTGNPSFGLTNSSGRIVYKKKFRLWKNGKNGTAEVASFNTTFVINIFPIDGSRGEGLA 122
Query     615 FIITSDTTLPNNSYGQWLGIVNEQTNGSSESQIVAIEFDTRMSYNGDVDSNHIGLDVNSVYSIVQYPLEKWNISLSSGKNVTVNVKYDAPSKTMSIDL 712
              FI+ S+   LP NS G+W+GIVNE  +G   ++I+AIEFDTR S + DVD NH+G+DVN + S VQ PL  + I+LS+G +VT ++ YD  SK+MSI +
Subject   123 FILASNWILPENSQGRWMGIVNEANDGEESNRIIAIEFDTRNSSSDDVDGNHVGVDVNGIRSRVQVPLAGYGINLSNGTDVTASIVYDGISKSMSIFV 220
Query     713 SM-------VDGERNTNSTPVI-TMPVDLSKFLPEDVYVGFSGATGDQ---IQLNCVLSWTFSGENI--EEGMNMKLMWILVGILVPMLFCVGLTSCL 797
              +M       +DG     S PVI + P+DLS FL E VY+GFSG+TGDQ    QLNCV +W+FSG +I  ++   M +++I++G LV  +    +
Subject   221 AMGNSTTPAIDG-----SNPVIDSWPLDLSNFLQEHVYIGFSGSTGDQSNITQLNCVKAWSFSGRDITGKKSRLMMILYIVIGALVVAVIACAVVFFS 313
Query     798 WWRRSRQGDHVRLDLESILSNSSKG-PHKFHFKELKSATGNFSSKNKLGRGGFGTVYKGFLKEVKEEVAVKRVSKDSSQGEQEFIAEVTTISQLRHKN 894
               WR+ R+ D ++   +SI  +   G P KF  KEL+ AT NF  +NKLG GGFGTVYKG L + K++VAVKRVSKDS QGEQEFI E+TTI +L HKN
Subject   314 CWRK-RKNDRIKSRQKSIQVSMLDGSPRKFKLKELRCATCNFHPRNKLGEGGFGTVYKGVLNDTKQKVAVKRVSKDSKQGEQEFINEITTIGKLNHKN 410
Query     895 LVKLIGWCCE-RRDLILVYEFLPRGSLDKLLFSNESTAP--EVLLSWNRRHNIIYGVACALNYLHDGCLKRILHRDVKASNVMIDSEYNARLGDFGLA 989
              LVKLIGWC +   D +LVYEF+P GSLDKLLFS+E+ A      LSW++RHNII GVA AL YLH+GC KR+ HRDVK+SNVM+D+ ++ARLGDFGLA
Subject   411 LVKLIGWCYKSENDFLLVYEFMPNGSLDKLLFSHENPAGVYSDQLSWDKRHNIIEGVAHALCYLHNGCAKRVFHRDVKSSNVMLDANFDARLGDFGLA 508
Query     990 RTIQHDGKTHHSTNVIAGTPGYIAPEYFHTGRASVETDVYGFGVFAMEVASGRTPNN-------NNDNNINDKYIMVDWLWGLYGRQRILDAVDPQLH 1080
              R +Q DGKTHH+T  I+GTPGY+APEYF TGRA+ ETDVYGFGVF MEVA GR P N        +   I     +V+WLW L+GR RI+ AVDP L
Subject   509 RAVQQDGKTHHTTGNISGTPGYVAPEYFLTGRANGETDVYGFGVFIMEVACGRRPYNIQYHDHGPHGAKIGSNNEIVNWLWDLHGRGRIVHAVDPTLS 606
Query    1081 GEYDEDQMERVLKLGLACCHPNPNERPNMRVVLQVLTGESPPPVLPTEKPSFMWPVV 1137
              G +D+ QME VL+LGLACCHPNP ERP MR+  QVL GE+  P +P EKP+FMWPVV
Subject   607 GVFDQKQMECVLRLGLACCHPNPKERPTMRLATQVLIGEASTPEVPIEKPAFMWPVV 663
b. Score: 186.04 (471), E value: 1.77×10⁻⁴⁸, Identity: 113/271 (41.7%), Positives: 166/271 (61.3%), Gaps: 21/271 (7.7%)
Query     255 LQITPESNSNRINMRNRVGRILYSETFKVWSMK---TSKFASFNTTFELYIQRVYGEPCGEGMAFILTNNPSVPSNSQGQWLGIVNEETNESPSNGIVAI 351
              LQ+T ++ +    + N  GRI+Y + F++W      T++ ASFNTTF + I  + G   GEG+AFIL +N  +P NSQG+W+GIVNE  +   SN I+AI
Subject    60 LQVTRDTGNPSFGLTNSSGRIVYKKKFRLWKNGKNGTAEVASFNTTFVINIFPIDGSR-GEGLAFILASNWILPENSQGRWMGIVNEANDGEESNRIIAI 158
Query     352 EFDTRKSSDDDLDDNHVAVDLNGIKSSHQFSLGQAGINLTGGSNVLARIQYDGESGKLTVYASMFHDSGHSL--SRPLI-TLDINLSKYLMEDVYVGFSG 448
              EFDTR SS DD+D NHV VD+NGI+S  Q  L   GINL+ G++V A I YDG S  ++++ +M + +  ++  S P+I +  ++LS +L E VY+GFSG
```

Fig. 2 An example entry of a transcript annotated by the transXpress pipeline and rendered through SequenceServer 2.0.0 [50]. A number of annotations including TPM (expression quantification) values, protein domain and transmembrane domain predictions, subcellular localization, and signaling peptide predictions are annotated (underlined in purple). Auto-generated external hyperlinks are added as well (underlined in green). The example protein sequence was shortened for clarity

Table 2 Descriptive statistics of the *P. longum* transcriptomes assembled with transXpress using the Trinity and rnaSPADES assemblers

	Trinity (v2.13.2)	rnaSPADES (v3.13.0)
Number of raw sequencing reads (input data)	16,901,456 (leaf) + 22,900,035 (spike) + 27,496,748 (root) = 67,298,239 total reads	
Number of assembled transcripts (isoforms)	268,313	296,600
Number of reconstructed genes (Trinity estimate)	132,944	–
Min / median / mean / max transcript lengths	185/577/914/15,159	112/363/832/15,665
Number of predicted protein ORFs (TransDecoder)	131,098	118,984
% of full-length ORFs (TransDecoder estimate)	54.7	60.4
Min / median / mean / max ORF lengths	85/200/282/4982	85 / 191 / 255 / 5091
Transcriptome completeness (BUSCO, embryophyta_odb10 lineage)	C: 95.2% [S: 10.5%, D: 84.7%], F: 2.7%, M: 2.1%	C: 84.1% [S: 18.6%, D: 65.5%], F: 11.1%, M: 4.8%
% of reads aligned to the transcriptome (Bowtie2)	87.5%	83.3%

The estimate of the number of reconstructed genes is only generated by Trinity, by grouping the transcript isoforms that likely originated from the same gene

annotations created with transXpress—expression levels in different samples, the best BLAST hit in SwissProt, identified Pfam domains, topology prediction for transmembrane proteins, subcellular localization and prediction of targeting peptides, and auto-generated external hyperlinks to relevant Pfam and UniProt entries (Fig. 2).

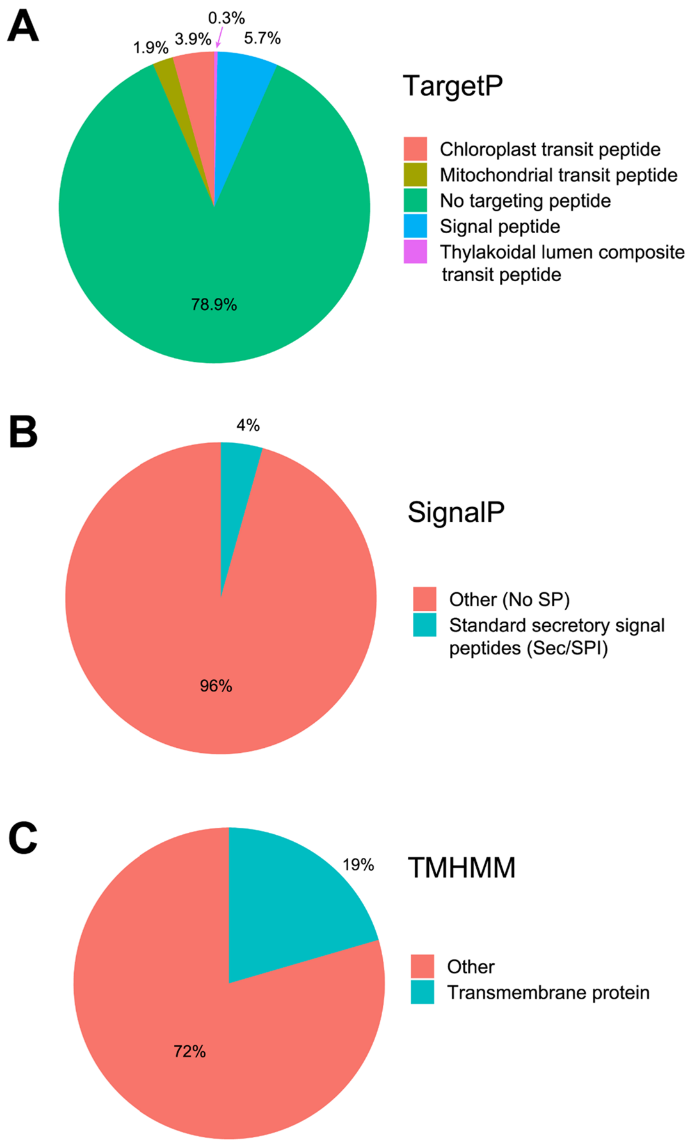


Fig. 3 Statistics of the predicted *P. longum* protein sequences (n = 131,098) generated automatically using the tools included in transXpress. Data from the Trinity assembly is shown, as the results for the rnaSPADES assembly were very similar. **A** N-terminal targeting peptides predicted by TargetP. **B** N-terminal signaling peptides predicted by SignalP. **C** Transmembrane proteins predicted by TMHMM

Results and discussion

To demonstrate the utility of the transXpress pipeline, we processed RNA-seq reads from long pepper (*Piper longum*), also known as pippali, a non-model plant used in Indian Ayurvedic medicine [52]. *P. longum* plants have been used in traditional medicine from ancient times and are known to produce biochemically interesting alkaloids with anticancer and nootropic effects in humans [53, 54]. The RNA-seq data were downloaded from NCBI Sequence Read Archive (SRA) and contained Illumina stranded, paired-end 2 × 150 bp reads from *Piper longum* leaf, spike and root samples. The *transXpress* pipeline was run on a computational cluster with either Trinity or rnaSPADES as the assembler of choice. Notably, both de novo assemblers generated over 200 thousand

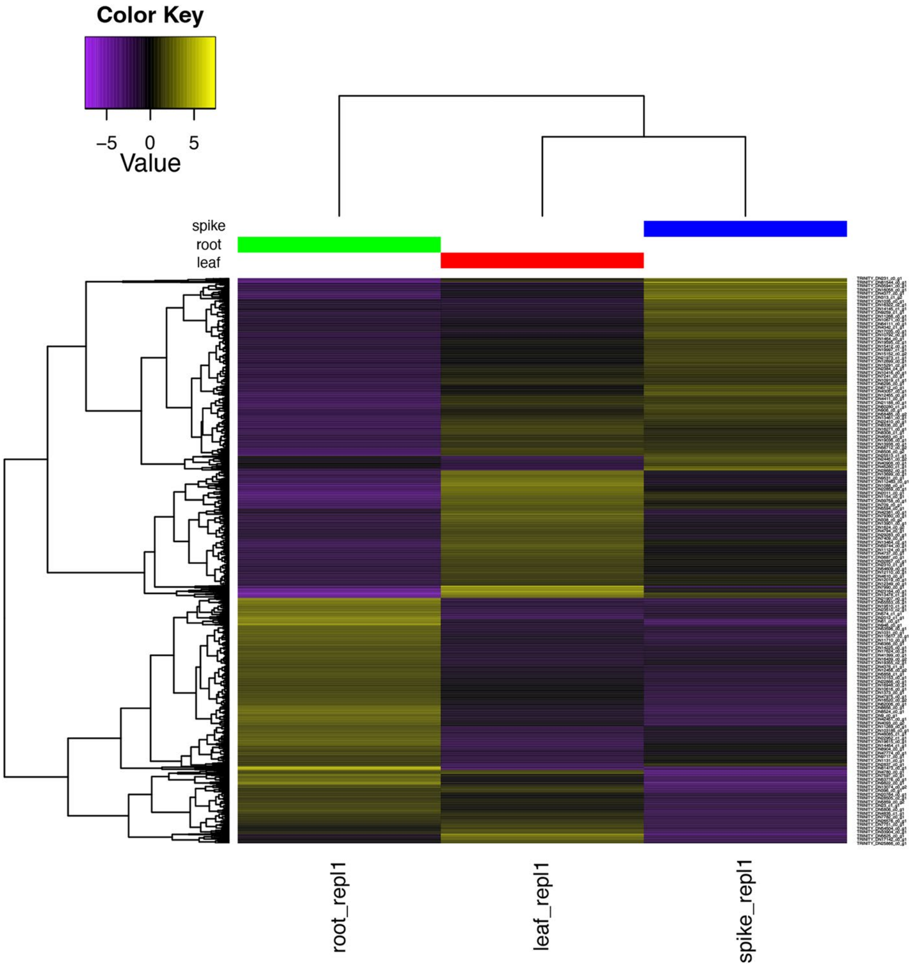


Fig. 4 A hierarchically-clustered heatmap showing differential expression between root, leaf and tissue RNA-Seq samples from *Piper longum* [52]. This figure was automatically generated by the differential expression analysis step of transXpress from the transcriptome assembled with Trinity

unique transcripts with an average predicted ORF length of 282 and 255 amino acids, respectively (Table 2). In comparison, a recent genome assembly of the closely related black pepper (*Piper nigrum*) [55] contains 63,466 genes with the average protein coding sequence length 1347 nt (449 amino acids). This difference is likely related to the large proportion (22%) of 5′-partial transcripts, possibly caused by incomplete PCR amplification using oligo (dT) primers, as commonly performed in RNA-seq protocols. It is worth noting that for such 5′-partial protein sequences, targeting peptide prediction is not possible.

Targeting peptides were found in 11.8% of the protein sequences using TargetP. The most common targeting sequence was a signal peptide for endoplasmic reticulum, followed by a chloroplast transit peptide (Fig. 3A, B). About 19% of all protein sequences were predicted to contain transmembrane domains (Fig. 3C). Differential expression analysis of the three tissue samples was performed using edgeR [40] (Fig. 4).

Conclusions

The transXpress pipeline is an easy-to-install, integrated tool that generates reproducible, annotated FASTA files ready for downstream mining. With this, transXpress facilitates rapid discovery of new genes and proteins in non-model organisms. The pipeline is actively maintained and is already used by many labs. For experienced users, transXpress can provide a good starting point to develop customized workflows.

Availability and requirements

Project name: transXpress.

Project home page: https://github.com/transXpress/transXpress

Operating system(s): Linux.

Programming language: Snakemake (Python), bash.

Other requirements: Dependencies installed via Conda or pip.

License: GNU GPLv3.

Any restrictions to use by non-academics: none.

Acknowledgements

We thank Brian Hass for his support with numerous issues and questions related to the Trinity assembler. The transXpress logo was designed by the Whitehead Institute Bioinformatics & Research Computing group.

Author contributions

TRF and TP developed the initial version of the pipeline. TČ added edgeR and documentation. TP wrote the draft of the manuscript. JKW supervised the initial development of the pipeline and edited the manuscript. MM provided testing and functionality improvements of the pipeline and contributed to the manuscript.

Funding

T.R.F. is supported by the National Institute of Environmental Health Sciences, Kirschstein-NRSA postdoctoral fellowship (grant number F32-ES032276). This work is supported by the Family Larsson-Rosenquist Foundation (J.K.W.), the National Science Foundation (MCB-1818132, J.K.W.), Chan Zuckerberg Foundation (2020-221485, J.K.W.), Gordon and Betty Moore Foundation (9331, J.K.W.), the Czech Science Foundation—GA CR (21-11563M, T.P.), and the European Union's Horizon 2020 research and innovation programme (Marie Skłodowska-Curie grant 891397, T.P.). The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding providers.

Availability of data and materials

The datasets analyzed during the current study are available in the NCBI SRA repository, containing ***Piper longum*** leaf (SRR10362954), spike (SRR10362953) and root (SRR10583928) RNA-seq datasets [52]. Two archives with the output files produced by the transXpress runs using Trinity and rnaSPADES on the ***Piper longum*** sequencing datasets were deposited into Zenodo under https://doi.org/10.5281/zenodo.7380017 [56].

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

J.K.W. is a member of the Scientific Advisory Board and a shareholder of DoubleRainbow Biosciences, Galixir, and Inari Agriculture, which develop biotechnologies related to natural products, drug discovery and agriculture. All other authors have no competing interests.

Received: 21 March 2022 Accepted: 24 March 2023
Published online: 04 April 2023

References

1. Torrens-Spence MP, Fallon TR, Weng JK. A workflow for studying specialized metabolism in nonmodel eukaryotic organisms. In: O'Connor SE, editor. Methods in enzymology. Academic Press; 2016. p. 69–97.
2. Stark R, Grzelak M, Hadfield J. RNA sequencing: the teenage years. Nat Rev Genet. 2019;20:631–56.

3. RNA-Seq datasets in NCBI SRA. https://www.ncbi.nlm.nih.gov/sra/?term=TRANSCRIPTOMIC%5BSource%5D. Accessed 24 Oct 2022.
4. NCBI TSA. https://www.ncbi.nlm.nih.gov/Traces/wgs/?view=TSA. Accessed 24 Oct 2022.
5. Conesa A, Madrigal P, Tarazona S, Gomez-Cabrero D, Cervera A, McPherson A, et al. A survey of best practices for RNA-seq data analysis. Genome Biol. 2016;17:13.
6. Martin J, Bruno VM, Fang Z, Meng X, Blow M, Zhang T, et al. Rnnotator: an automated de novo transcriptome assembly pipeline from stranded RNA-Seq reads. BMC Genom. 2010;11:663.
7. Melicher D, Torson AS, Dworkin I, Bowsher JH. A pipeline for the de novo assembly of the *Themira biloba* (Sepsidae: Diptera) transcriptome using a multiple k-mer length approach. BMC Genom. 2014;15:188.
8. Ortiz R, Gera P, Rivera C, Santos JC. Pincho: a modular approach to high quality de novo transcriptomics. Genes. 2021;12:953.
9. Lataretu M, Hölzer M. RNAflow: an effective and simple RNA-Seq differential gene expression pipeline using nextflow. Genes. 2020;11:1487.
10. Ewels PA, Peltzer A, Fillinger S, Patel H, Alneberg J, Wilm A, et al. The nf-core framework for community-curated bioinformatics pipelines. Nat Biotechnol. 2020;38:276–8.
11. Federico A, Karagiannis T, Karri K, Kishore D, Koga Y, Campbell JD, et al. Pipeliner: a nextflow-based framework for the definition of sequencing data processing pipelines. Front Genet. 2019;10:614.
12. Cornwell M, Vangala M, Taing L, Herbert Z, Köster J, Li B, et al. VIPER: visualization pipeline for RNA-seq, a snakemake workflow for efficient and complete RNA-seq analysis. BMC Bioinform. 2018;19:135.
13. Zhang X, Jonassen I. RASflow: an RNA-Seq analysis workflow with snakemake. BMC Bioinform. 2020;21:110.
14. Wang D. hppRNA—a snakemake-based handy parameter-free pipeline for RNA-Seq analysis of numerous samples. Brief Bioinform. 2018;19:622–6.
15. Wolfien M, Rimmbach C, Schmitz U, Jung JJ, Krebs S, Steinhoff G, et al. TRAPLINE: a standardized and automated pipeline for RNA sequencing data analysis, evaluation and annotation. BMC Bioinform. 2016;17:21.
16. Zhao S, Xi L, Quan J, Xi H, Zhang Y, von Schack D, et al. QuickRNASeq lifts large-scale RNA-seq data analyses to the next level of automation and interactive visualization. BMC Genom. 2016;17:39.
17. Orjuela S, Huang R, Hembach KM, Robinson MD, Soneson C. ARMOR: an automated reproducible modular workflow for preprocessing and differential analysis of RNA-seq data. G3. 2019;9:2089–96.
18. Gadepalli VS, Ozer HG, Yilmaz AS, Pietrzak M, Webb A. BISR-RNAseq: an efficient and scalable RNAseq analysis workflow with interactive report generation. BMC Bioinform. 2019;20(Suppl 24):670.
19. Law CW, Alhamdoosh M, Su S, Dong X, Tian L, Smyth GK, et al. RNA-seq analysis is easy as 1–2–3 with limma, Glimma and edgeR. F1000Res. 2016;5.
20. Köster J, Rahmann S. Snakemake—a scalable bioinformatics workflow engine. Bioinformatics. 2012;28:2520–2.
21. Di Tommaso P, Chatzou M, Floden EW, Barja PP, Palumbo E, Notredame C. Nextflow enables reproducible computational workflows. Nat Biotechnol. 2017;35:316–9.
22. Goecks J, Nekrutenko A, Taylor J, Galaxy Team. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. Genome Biol. 2010;11:R86.
23. Grüning B, Dale R, Sjödin A, Chapman BA, Rowe J, Tomkins-Tinch CH, et al. Bioconda: sustainable and comprehensive software distribution for the life sciences. Nat Methods. 2018;15:475–6.
24. transXpress GitHub page. https://github.com/transXpress/transXpress. Accessed 30 Nov 2022.
25. Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, et al. Full-length transcriptome assembly from RNA-Seq data without a reference genome. Nat Biotechnol. 2011;29:644–52.
26. Haas BJ, Papanicolaou A, Yassour M, Grabherr M, Blood PD, Bowden J, et al. De novo transcript sequence reconstruction from RNA-seq using the Trinity platform for reference generation and analysis. Nat Protoc. 2013;8:1494–512.
27. Smith-Unna R, Boursnell C, Patro R, Hibberd JM, Kelly S. TransRate: reference-free quality assessment of de novo transcriptome assemblies. Genome Res. 2016;26:1134–44.
28. Babraham bioinformatics—FastQC A quality control tool for high throughput sequence data. https://www.bioinformatics.babraham.ac.uk/projects/fastqc. Accessed 11 Oct 2021.
29. Ewels P, Magnusson M, Lundin S, Käller M. MultiQC: summarize analysis results for multiple tools and samples in a single report. Bioinformatics. 2016;32:3047–8.
30. Bolger AM, Lohse M, Usadel B. Trimmomatic: a flexible trimmer for Illumina sequence data. Bioinformatics. 2014;30:2114–20.
31. Geniza M, Jaiswal P. Tools for building de novo transcriptome assembly. Curr Plant Biol. 2017;11–12:41–5.
32. Bushmanova E, Antipov D, Lapidus A, Prjibelski AD. rnaSPAdes: a de novo transcriptome assembler and its application to RNA-Seq data. Gigascience. 2019;8:100.
33. Hölzer M, Marz M. De novo transcriptome assembly: a comprehensive cross-species comparison of short-read RNA-Seq assemblers. Gigascience. 2019;8:039.
34. Ren X, Liu T, Dong J, Sun L, Yang J, Zhu Y, et al. Evaluating de Bruijn graph assemblers on 454 transcriptomic data. PLoS ONE. 2012;7: e51188.
35. Trinity Wiki—assembly statistics. https://github.com/trinityrnaseq/trinityrnaseq/wiki/Transcriptome-Contig-Nx-and-ExN50-stats. Accessed 24 Oct 2022.
36. Manni M, Berkeley MR, Seppey M, Simão FA, Zdobnov EM. BUSCO update: novel and streamlined workflows along with broader and deeper phylogenetic coverage for scoring of eukaryotic, prokaryotic, and viral genomes. Mol Biol Evol. 2021;38:4647–54.
37. Bray NL, Pimentel H, Melsted P, Pachter L. Near-optimal probabilistic RNA-seq quantification. Nat Biotechnol. 2016;34:525–7.
38. Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. Nat Methods. 2012;9:357–9.
39. Robinson JT, Thorvaldsdóttir H, Winckler W, Guttman M, Lander ES, Getz G, et al. Integrative genomics viewer. Nat Biotechnol. 2011;29:24–6.
40. Robinson MD, McCarthy DJ, Smyth GK. edgeR: a bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics. 2010;26:139–40.

41. Camacho C, Coulouris G, Avagyan V, Ma N, Papadopoulos J, Bealer K, et al. BLAST+: architecture and applications. BMC Bioinform. 2009;10:421.
42. UniProt Consortium. UniProt: the universal protein knowledgebase in 2021. Nucleic Acids Res. 2021;49:D480–9.
43. Eddy SR. Profile hidden Markov models. Bioinformatics. 1998;14:755–63.
44. Sonnhammer EL, Eddy SR, Durbin R. Pfam: a comprehensive database of protein domain families based on seed alignments. Proteins. 1997;28:405–20.
45. Nawrocki EP, Eddy SR. Infernal 1.1: 100-fold faster RNA homology searches. Bioinformatics. 2013;29:2933–5.
46. Kalvari I, Nawrocki EP, Ontiveros-Palacios N, Argasinska J, Lamkiewicz K, Marz M, et al. Rfam 14: expanded coverage of metagenomic, viral and microRNA families. Nucleic Acids Res. 2021;49:D192-200.
47. Teufel F, Almagro Armenteros JJ, Johansen AR, Gíslason MH, Pihl SI, Tsirigos KD, et al. SignalP 6.0 predicts all five types of signal peptides using protein language models. Nat Biotechnol. 2022;40:1023–5.
48. Almagro Armenteros JJ, Salvatore M, Emanuelsson O, Winther O, von Heijne G, Elofsson A, et al. Detecting sequence signals in targeting peptides using deep learning. Life Sci Alliance. 2019;2:5.
49. Sonnhammer EL, von Heijne G, Krogh A. A hidden Markov model for predicting transmembrane helices in protein sequences. Proc Int Conf Intell Syst Mol Biol. 1998;6:175–82.
50. Priyam A, Woodcroft BJ, Rai V, Moghul I, Mungala A, Ter F, et al. Sequenceserver: a modern graphical user interface for custom BLAST databases. Mol Biol Evol. 2019. https://doi.org/10.1093/molbev/msz185.
51. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. J Mol Biol. 1990;215:403–10.
52. Dantu PK, Prasad M, Ranjan R. Elucidating biosynthetic pathway of piperine using comparative transcriptome analysis of leaves, root and spike in *Piper longum* L. bioRxiv. 2021; 2021.01.03.425108.
53. Salehi B, Zakaria ZA, Gyawali R, Ibrahim SA, Rajkovic J, Shinwari ZK, et al. Piper species: a comprehensive review on their phytochemistry. Biol Act Appl Mol. 2019;24:1364.
54. Choudhary N, Singh V. A census of *P. longum's* phytochemicals and their network pharmacological evaluation for identifying novel drug-like molecules against various diseases, with a special focus on neurological disorders. PLoS ONE. 2018;13:e0191006.
55. Hu L, Xu Z, Wang M, Fan R, Yuan D, Wu B, et al. The chromosome-scale reference genome of black pepper provides insight into piperine biosynthesis. Nat Commun. 2019;10:1–11.
56. Čalounová T. Piper longum transcriptomes generated using transXpress. https://doi.org/10.5281/zenodo.7380017. 2022.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.