

resolveS: rapid genome-independent inference of RNA-seq library strandedness using universal rRNA alignments

Dalang Yu^{1#*}, Xi Long^{2#} and Ting Zhao^{2*}

¹ Zhejiang Key Laboratory of Signaling Mechanisms in Tumor Progression and Intervention, School of Life Science, Hangzhou Institute for Advanced Study, University of Chinese Academy of Sciences, Hangzhou 310024, China

² Zhejiang Provincial Key Laboratory of Crop Genetic Resources, Institute of Crop Science, Plant Precision Breeding Academy, College of Agriculture and Biotechnology, Zhejiang University, Hangzhou 310058, China

Authors contributed equally: Dalang Yu, Xi Long

* Correspondence: yudalang@ucas.edu.cn (Yu D); 0618041@zju.edu.cn (Zhao T)

Abstract

Accurate specification of RNA-seq library strandedness is essential for read counting, transcript assembly, and antisense transcription analysis, yet this information is frequently missing from public metadata. Here, we present resolveS, a rapid and lightweight statistical method for inferring RNA-seq library strandedness without requiring species-specific reference genomes or gene annotations. resolveS maps a fixed-size subset of reads to a compact universal rRNA database and infers library type using a robust strand-bias statistic, relative difference, combined with adaptive mapping-quality filtering and a per-rRNA majority-vote framework. This genome-independent design enables strandedness inference across both model and non-model organisms without requiring species-specific genomic resources, while avoiding the computational burden of conventional genome- or transcriptome-based workflows. Across benchmark datasets with 1,091 known library types, resolveS provided robust strandedness inference while maintaining computational efficiency, typically completing analysis in seconds per sample with < 0.5 GiB memory usage. In addition to strandedness classification, resolveS reports detection confidence levels to aid interpretation of ambiguous cases. Together, these features make resolveS a practical and scalable solution for strandedness inference in RNA-seq workflows, particularly for public data reanalysis and studies lacking high-quality genomic references.

Keywords: RNA-sequencing, Fast strandedness inference, Quality control, Non-model organisms

Citation: Yu D, Long X, Zhao T. 2026. resolveS: rapid genome-independent inference of RNA-seq library strandedness using universal rRNA alignments. *Genomics Communications* 3: e022 <https://doi.org/10.48130/gcomm-0026-0020>

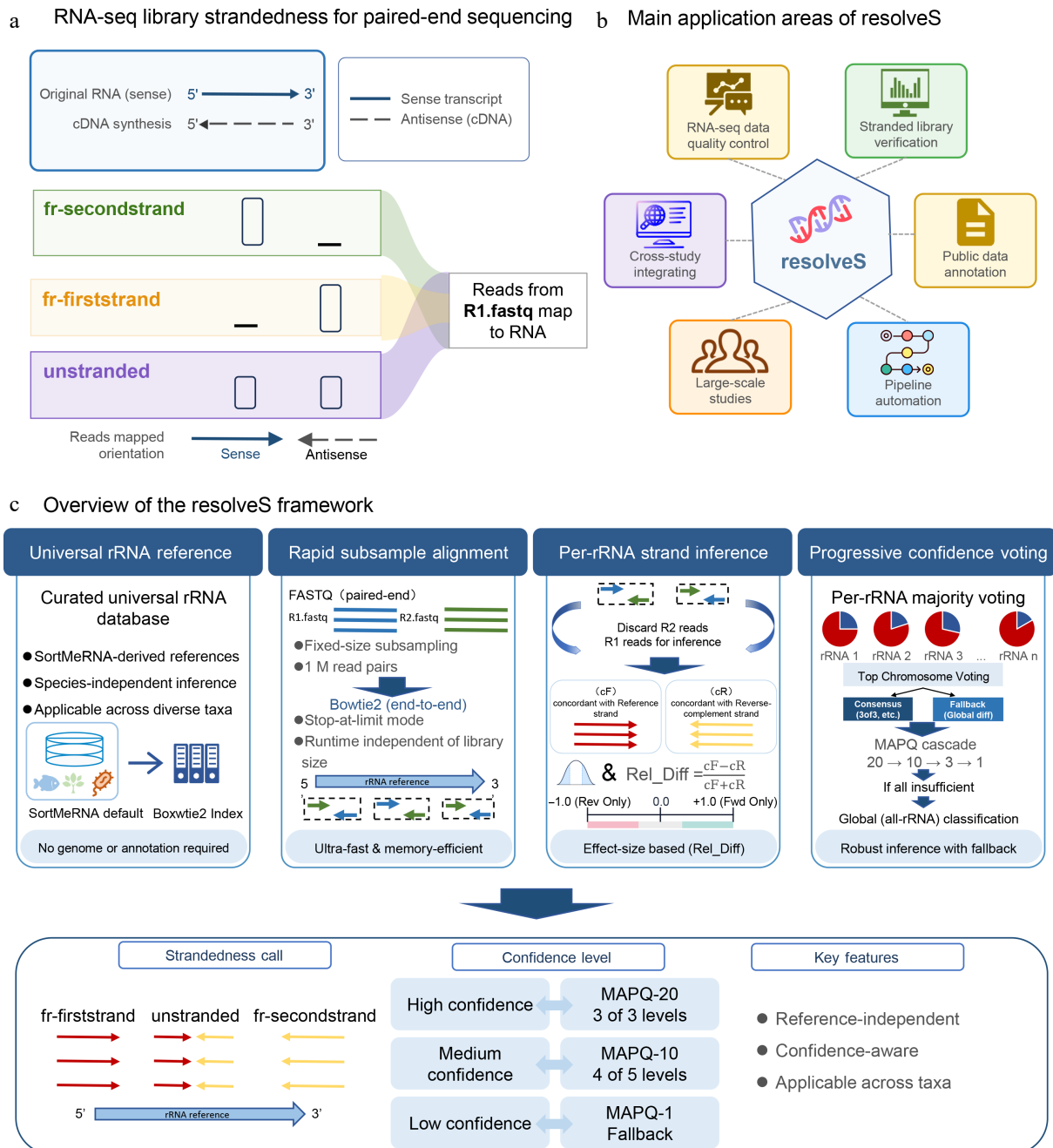
Introduction

High-throughput RNA sequencing (RNA-seq) has revolutionized our understanding of transcriptomes, enabling precise quantification of gene expression, alternative splicing, and novel transcript discovery^[1–5]. Modern RNA-seq protocols frequently employ strand-specific library preparation^[5], which is essential for distinguishing transcripts from overlapping genes encoded on opposite strands^[6] (Fig. 1a). Such overlap is common: approximately one quarter of human protein-coding genes overlap with genes on the opposite strand, and this organization is even more prevalent in compact genomes^[7]. Depending on the preparation strategy, libraries are classified as unstranded, fr-firststrand, or fr-secondstrand. Incorrect specification of the strandedness parameter in downstream tools^[8] (Fig. 1b), such as featureCounts^[9] or Trinity^[10], can severely bias analytical results. Despite its importance, this parameter is frequently omitted from public repository metadata and from the corresponding publications; a systematic investigation found that approximately 44% of studies failed to report the library preparation method or strandedness^[11].

Existing tools face significant limitations. RSeQC's infer_experiment.py requires a genome-aligned BAM file and a gene annotation converted to BED format^[12], and how_are_we_stranded_here^[11] accelerates inference by pseudoaligning a subset of reads but still depends on species-specific reference resources. This reliance on reference genomes and annotations poses a substantial challenge for newly sequenced or non-model organisms, where high-quality genomic resources are unavailable. It can be only partially

circumvented by a cumbersome *de novo* workflow assembly with Trinity^[10], coding-region prediction with TransDecoder, annotation conversion with BEDOPS^[13], realignment, and inference with RSeQC^[12], or by manually inspecting strand patterns in a genome browser such as IGV^[14], an *ad hoc* approach that is subjective, poorly suited to large-scale analysis, and whose general validity has not been rigorously established. In low-quality, low-rRNA, or ambiguous datasets, residual rRNA reads may be sparse, unevenly distributed, or ambiguously aligned, making such manual inference unreliable (See https://github.com/yudalang3/resolveS/releases/tag/IGV_demo_data to run the demo and [Supplementary File 1](#) to see the intuitive IGV snapshot).

In this work, we developed resolveS, an ultra-fast, genome-independent solution (Fig. 1c). Its principal novelty lies in independence from species-specific reference genomes and gene annotations: it instead maps reads to a compact, curated database of universal rRNA sequences, leveraging the evolutionary conservation of ribosomal RNA across the tree of life and the ubiquity of residual rRNA in RNA-seq libraries to make rRNA an ideal universal proxy. We further address a common statistical pitfall in NGS analysis, the sensitivity of *p*-values to large sample sizes, with a progressive inference framework that combines adaptive mapping-quality filtering with per-rRNA evaluation and reports confidence estimates for ambiguous and low-evidence cases. Combined with the stop-on-max-reads functionality of Bowtie^[15], this design achieves remarkable speed and resource efficiency, establishing resolveS as a practical tool for universal RNA-seq library strandedness inference.



Materials and methods

The test datasets

Benchmark datasets with known library strandedness were collected from both plant and animal studies. Plant datasets were obtained from

Shi et al.^[16], the animal cell line data were obtained from Naphade et al.^[17] and Sigurgeirsson et al.^[18]. To evaluate large-scale performance, we also included the supplementary collection of Signal and Kahlke^[11]. Obviously, broken raw data (SRR7759712/SRR7759714/SRR7890157) is not used for precise quantification (Supplementary File 2). Please see Supplementary File 3 for the detailed SRA accession number.

The universal rRNA reference strategy

We employed a lightweight, universal reference database derived from the SortMeRNA database^[19]. The default database version was selected as the standard reference based on empirical benchmarking, while a high-sensitivity version is provided as an alternative. Please see [Supplementary File 4](#) for detailed taxonomic composition of the universal rRNA reference database (by domain and phylum).

Database setup and indexing instructions are provided in the Data availability section.

Aligner choice and subsampling optimization strategy

Speed and memory efficiency were primary design goals of resolveS. We compared three representative short-read aligners, BWA (0.7.19-r1273)^[20], Bowtie2 (2.5.4)^[15], and LexicMap (v0.8.0)^[21], for rapid strandedness inference against the universal rRNA reference. Bowtie2 showed the most favorable balance of runtime, memory usage, and practical functionality. Its key advantage is the -u option, which stops alignment after a user-defined number of reads and thereby enables inference from a fixed-size subset of the library; BWA provides no equivalent, and LexicMap was less well suited to direct alignment of large RNA-seq FASTQ datasets. Bowtie2 also required approximately half the memory of BWA, aligned two to three times faster, and used a substantially smaller index (214 vs. 521 MB).

Bowtie2 was therefore selected as the default aligner. resolveS uses the -u option to align only the first subset of reads to the rRNA index, with a default of 5,000,000 reads per library, a practical compromise between detection robustness and computational efficiency. Runtime therefore depends on the preset subsampling depth rather than total sequencing volume, making the method effectively constant time with respect to library size and returning results within seconds per sample even for very large inputs. Because rRNA sequences are highly repetitive, multi-mapping is handled explicitly: resolveS counts only primary alignments (Bowtie2 reports one primary alignment per read in its default end-to-end mode), so a multi-mapping read contributes at most one orientation observation. Reads with ambiguous placement receive low mapping qualities and are progressively removed by the adaptive MAPQ filter (MAPQ ≥ 20 down to 1), while per-rRNA majority voting further limits the influence of any single repetitive locus. Please see [Supplementary File 5](#) for the mapping ratio of the rRNA library.

Permutation test for non-random strand bias

For each rRNA reference sequence, resolveS records the numbers of primary alignments mapped in the same orientation as the reference and in the reverse-complement orientation, denoted cF and cR, respectively. Several candidate strand-bias metrics were evaluated on these counts ([Supplementary File 6](#)), and the signed relative difference (Rel_Diff) was selected as the primary inference metric:

$$\text{Forward Ratio} = \frac{cF}{cF + cR}$$

$$\text{Reverse Ratio} = \frac{cR}{cF + cR}$$

$$\text{Rel_Diff} = \frac{cF - cR}{cF + cR}$$

Here, Forward Ratio and Reverse Ratio quantify the proportions of retained R1 alignments assigned to each orientation. Rel_Diff summarizes both the direction and magnitude of strand bias on a scale from -1 to +1, and the chi-square statistic tests deviation from the null expectation of a 50:50 strand distribution.

To address sample-size-dependent variation and mild baseline orientation bias, we characterised the signed relative difference $\text{Rel_Diff} = (cF - cR)/(cF + cR)$ under an explicit null model. cF was modeled as Binomial ($n, 0.5$) across n in {20, 50, 100, 500, 1,000}, and exact two-sided Binomial critical values were derived on the Rel_Diff scale. To examine mild baseline orientation probabilities, cF was further modeled as Binomial (n, p_{true}) for p_{true} from 0.4 to 0.6, and critical Rel_Diff boundaries were computed using an exact two-sided tail criterion of $p \leq 0.001$. Finally, for $p_{\text{true}} = 0.5$, the Binomial acceptance region at $\alpha = 0.01$ was evaluated across sequencing depths, with the fixed $|\text{Rel_Diff}| = 0.3$ threshold as an effect-size reference. Please see [Supplementary File 7](#) for detailed reproducible pipelines.

Statistical framework for strandedness inference

Unlike the chi-square statistic, which is highly sensitive to sample size and becomes inflated at large read counts, Rel_Diff is a direct measure of effect size and is therefore robust to variation in mapped counts. Strandedness is inferred independently for each informative rRNA reference before sample-level consensus voting, combining hypothesis testing with effect-size filtering: $\text{Rel_Diff} > +0.3$ indicates fr-secondstrand, $\text{Rel_Diff} < -0.3$ is fr-firststrand, and $|\text{Rel_Diff}| \leq 0.3$ is unstranded.

Targets with insufficient support, defined as $F + R < 18$ after filtering, are labelled insufficient-data and excluded from voting. The final sample-level call is determined by a progressive majority-vote framework across the top-supported informative references combined with MAPQ-based filtering, which limits the influence of weakly supported targets while retaining an interpretable confidence level for each sample.

Strandedness detection confidence levels

Strandedness is inferred using a per-rRNA majority-vote framework combined with adaptive MAPQ filtering, rather than from global forward/reverse fragment counts alone. For each sample, informative rRNA references are ranked in descending order by valid read count, and inference proceeds progressively across MAPQ thresholds of 20, 10, 3, and 1, beginning with the most stringent. At each threshold, only alignments at or above the cutoff are retained.

A progressive majority vote is then applied to the top-ranked targets using the sequence 3/3, 4/5, 6/7, and 7/8, representing progressively relaxed consensus criteria. The first level at which a valid majority is achieved is reported as the detection level; for example, MAPQ-20/3-of-3 indicates that the top three informative targets agree unanimously under stringent filtering, whereas MAPQ-0/7-of-8 indicates that consensus was reached only after relaxed filtering.

If no majority consensus is obtained at any MAPQ threshold, resolveS assigns a fallback detection level and determines the class from the global forward/reverse alignment ratio across all retained alignments. Fallback calls should therefore be interpreted with lower confidence than calls supported by successful per-rRNA voting.

For strandedness inference, only R1 alignments are retained after paired-end mapping. R1 reads are classified according to their orientation relative to each rRNA reference. Reads aligned in the same orientation as the rRNA reference are assigned to the cF class (concordant with reference orientation), whereas reads aligned in the opposite orientation are assigned to the cR class (concordant with reverse-complement orientation). For each informative rRNA reference, the numbers of reads assigned to the cF and cR classes

are denoted as cF and cR, respectively, and used to calculate the relative difference statistic: $Rel_Diff = (cF - cR)/(cF + cR)$. The strandedness specificity judgement combines hypothesis testing (binomial distribution of cF) with Rel_Diff filtering to achieve statistical robustness.

The resulting Rel_Diff value quantifies strand bias for each rRNA reference and serves as the basis for downstream consensus voting and confidence-aware strandedness inference.

Per-rRNA strandedness inference is first performed independently for each informative rRNA reference and subsequently integrated through adaptive MAPQ filtering and consensus voting. For paired-end data, both reads are used for proper-pair validation and mapping confidence, whereas only R1 is counted for orientation to avoid double-counting the same fragment. Consensus levels such as '3 of 3' indicate complete agreement among the highest-supported rRNA references. This hierarchical decision framework improves robustness in low-signal, noisy, or ambiguous datasets while simultaneously providing confidence-aware strandedness classification.

Results

Efficient pipeline and practical robustness across library conditions

The resolveS pipeline was designed for rapid and robust strandedness inference (Fig. 1c). Benchmarking of candidate aligners showed that Bowtie2 provided the most favorable balance of speed, memory usage, and practical functionality (Table 1), and was therefore selected as the default (see Materials and Methods). Evaluation of the SortMeRNA-derived reference versions showed that the default database performed comparably to the high-sensitivity version while requiring less computation, supporting its use as the standard

reference. Importantly, library input amount and RNA enrichment strategy, including Ribo-Zero-based protocols, did not materially affect accuracy (Supplementary Files 6 and 8). Although poly(A) enrichment and rRNA depletion are designed to reduce ribosomal RNA content, these procedures are rarely complete; resolveS exploits the residual rRNA fraction as a stable and broadly available signal. Intuitive informal IGV^[14] screenshots are provided for manual checking (Supplementary File 1).

Rel_Diff combines with binomial hypothesis testing and detection levels to enable robust and interpretable inference

A major challenge in high-depth sequencing data is that conventional significance tests become overly sensitive to trivial deviations when read counts are large; a chi-square test may reach significance even when strand imbalance is biologically negligible. To reduce this sample-size dependence, we evaluated multiple strand-bias statistics (Supplementary File 6: Fig. S1) and found that Rel_Diff provided the clearest separation between stranded and unstranded libraries (Supplementary File 6: Fig. S2), offering the widest effective dynamic range while remaining interpretable as a direct measure of strand bias. We therefore adopted it as the primary classification statistic. Simulation (Supplementary File 7) confirmed the cutoff: the Binomial ($n, 0.5$) null distribution contracted rapidly around $Rel_Diff = 0$ as n increased, with the $p \leq 0.001$ critical $|Rel_Diff|$ falling from 0.800 at $n = 20$ to 0.106 at $n = 1,000$, and the $\alpha = 0.01$ acceptance region narrowing from ± 0.700 to ± 0.084 over the same range. Statistical null behavior can thus be evaluated explicitly while preserving $|Rel_Diff| = 0.3$ as a separate biological effect-size criterion rather than a universally optimized constant.

In addition to a strandedness class, resolveS reports a detection level indicating the confidence of each call. The highest-confidence

Table 1. Benchmarking of resolveS runtime and peak memory usage under different execution settings.

Input type	Approach	Threads number	Maximum align reads	Time cost (s)	Memory cost (GiB)
5 samples	Apptainer_image	15	1,000,000	18.7733 (0.3099)	0.3240 (0.0017)
5 samples	Apptainer_image	15	4,000,000	68.0333 (4.3625)	0.3277 (0.0025)
5 samples	Apptainer_image	8	1,000,000	21.1567 (0.4267)	0.2813 (0.0021)
5 samples	Apptainer_image	8	4,000,000	75.1567 (2.7356)	0.2833 (0.0006)
5 samples	Bash_script	15	1,000,000	21.9400 (0.5011)	0.3280 (0.0017)
5 samples	Bash_script	15	4,000,000	79.6300 (2.5363)	0.3310 (0.0000)
5 samples	Bash_script	8	1,000,000	19.8900 (0.5862)	0.2833 (0.0021)
5 samples	Bash_script	8	4,000,000	72.6667 (4.3824)	0.2857 (0.0006)
5 samples	Singularity_image	15	1,000,000	19.9367 (0.1464)	0.3267 (0.0012)
5 samples	Singularity_image	15	4,000,000	65.6567 (0.4693)	0.3293 (0.0006)
5 samples	Singularity_image	8	1,000,000	21.7067 (0.5609)	0.2800 (0.0017)
5 samples	Singularity_image	8	4,000,000	70.7333 (0.9570)	0.2857 (0.0029)
Single sample	Apptainer_image	15	1,000,000	3.6867 (0.0252)	0.3227 (0.0006)
Single sample	Apptainer_image	15	4,000,000	10.8867 (0.2371)	0.3250 (0.0017)
Single sample	Apptainer_image	8	1,000,000	4.1033 (0.1159)	0.2783 (0.0006)
Single sample	Apptainer_image	8	4,000,000	11.7100 (0.2706)	0.2817 (0.0023)
Single sample	Bash_script	15	1,000,000	2.9900 (0.0300)	0.3243 (0.0006)
Single sample	Bash_script	15	4,000,000	9.7700 (0.6791)	0.3283 (0.0029)
Single sample	Bash_script	8	1,000,000	2.9767 (0.0231)	0.2810 (0.0000)
Single sample	Bash_script	8	4,000,000	10.7567 (0.6309)	0.2850 (0.0000)
Single sample	Singularity_image	15	1,000,000	3.6133 (0.1041)	0.3240 (0.0026)
Single sample	Singularity_image	15	4,000,000	11.5267 (0.8363)	0.3283 (0.0051)
Single sample	Singularity_image	8	1,000,000	3.8433 (0.0379)	0.2807 (0.0023)
Single sample	Singularity_image	8	4,000,000	11.5700 (0.1609)	0.2790 (0.0000)

The CPU is a 13th Gen Intel Core™ i7-13700, and storage is a WD Green SN350 1TB. The output style is Mean (Standard Deviation). 5 samples means that in many sequencing-based biological experiments, samples within the same batch are often prepared using the same library construction protocol. Therefore, it is valuable to analyze them together to obtain a more reliable conclusion.

calls are those supported under stringent mapping-quality filtering with unanimous agreement among the top-ranked informative references (MAPQ-20/3-of-3); The weakest, MAPQ-0/multi-of-8-fallback, indicates that even at the lowest mapping quality, the top eight rRNA transcripts still yielded more than two candidate types. Lower-confidence outcomes therefore arise when evidence is recovered only after progressively relaxed MAPQ thresholds, or when no stable per-rRNA consensus is achieved, and the call falls back to pooled strand counts. The detection level thus summarizes both mapping quality and agreement consistency across informative targets.

High concordance across benchmark datasets

To assess accuracy, we applied resolveS to benchmark datasets with known library types (Supplementary File 8). Without additional read cleaning or species-specific genomic resources, resolveS achieved 98.81% concordance (249/252) with the reported labels, covering both stranded and unstranded datasets. Notably, the three discordant cases all carried fallback-type detection levels, indicating that the confidence framework flagged them as low-evidence rather than producing over-confident misclassifications. resolveS therefore combines high accuracy with substantially improved usability and reference independence (Table 2). All procedures have been documented in Supplementary Files 5 and 8.

We examined the three discordant samples in detail, as they delineate the boundary conditions of the method (See Supplementary File 2 for details). All three carried fallback detection levels and sat immediately adjacent to the $|\text{Rel_Diff}| = 0.3$ decision boundary. In two *Saccharomyces cerevisiae* libraries (SRR6357071, SRR6357072) of very poor sequencing quality, reported as stranded but called unstranded, residual rRNA was sparse and taxonomically scattered, so per-rRNA votes split across weakly supported targets and no progressive consensus was reached. The third (SRR8368636, *Arabidopsis thaliana*), reported as unstranded but called stranded, was dominated by a single organellar rRNA target carrying an extreme

single-orientation pile-up that tipped the global dominance fallback just past the threshold, whereas its sibling replicates were correctly classified. None reflected genuine biological antisense transcription; each arose from low or atypically distributed residual rRNA. In practice, fallback-flagged, near-threshold calls should be treated as low-confidence and, where needed, resolved by increasing the subsampling depth or inspecting the batch mode running results.

To further evaluate resolveS on an independent, curated benchmark, we compiled 817 samples from 42 RNA-seq projects (BD4 dataset, Supplementary File 9), comprising 401 single-end and 416 paired-end libraries. resolveS correctly recovered the expected strandedness in 94.9% of samples (775/817). Stratified by layout, consensus-match rates were 99.0% for paired-end (412/416) but 90.5% for single-end (363/401) (Fisher's exact test, $p = 9.5 \times 10^{-9}$). Project-level majority voting confirmed that this is not a systemic single-end bias: 41 of 42 projects (97.6%) achieved a majority vote consistent with the curated annotation, and 40 of 41 majority-YES projects were unanimous (100% agreement). The entire single-end deficit traces to one study, GSE103492 (14/52 matched; majority vote = NO), a dataset of unstranded libraries; excluding it, single-end consensus rose to 100% (349/349).

Excellent runtime and memory efficiency

We next evaluated computational performance across execution modes, thread counts, and read volumes (Table 1). For comparison with 'how_are_we_stranded_here', both tools were run on the same FASTQ files with the same read limit, specified by -u in resolveS and -n/--reads in how_are_we_stranded_here (Supplementary File 9 for details). resolveS consistently completed single-sample analyses within a few seconds while maintaining memory usage below 0.5 GiB across all tested conditions (Tables 2 and 3). Raising the read limit from one million to five million predictably increased runtime but had little effect on the final classification in routine cases, and increasing the thread count reduced runtime modestly compared with the effect of read depth.

Table 2. Comparison of resolveS with representative approaches for RNA-seq strandedness inference.

Software	Reference requirement	Primary alignment target	Speed/efficiency	Installation friendliness	Key advantage
RSeQC (infer_experiment.py)	Requires reference genome and annotations	Genome/transcriptome	Slow (requires full alignment/BAM)	Conda/mamba and pip	Established method.
how_are_we_stranded_here	Requires reference transcriptome/GTF	Transcriptome (Pseudoalignment via Kallisto)	Fast (seconds/sample)	Conda/mamba and pip	Fast for organisms with high-quality reference.
Trinity/TransDecoder Workflow	No (but requires massive compute)	De novo assembly	Extremely slow/high resource	Conda/mamba and extra configuration	For discovery in totally de novo context.
resolveS (Ours)	No reference genome required	Universal rRNA database	Ultra-fast (< 10 s) and low memory (< 0.5 GiB)	Portable binary and one-stop container images	Applicability across nearly all species, including non-model ones.

Table 3. Runtime performance comparison of two strand-orientation detection tools.

Tool	Cores	Runtime (s)			Peak memory (KB)			CPU utilization (%)		
		Avg	Min	Max	Avg	Min	Max	Avg	Min	Max
how_are_we_stranded_here	10	90.14	76.66	150.76	7,290,165	7,261,828	7,503,544	417.8	290.0	567.0
resolveS	10	25.38	18.05	50.51	358,330	354,076	368,756	1,118.4	1,107.0	1,124.0
how_are_we_stranded_here	20	77.53	69.59	119.89	7,590,248	7,548,824	7,993,380	559.8	417.0	834.0
resolveS	20	33.55	22.52	40.89	479,081	468,592	495,364	2,093.4	2,086.0	2,098.0

All tests were performed on 50 human paired-end RNA-seq samples from Naphade et al. (SRR13786231–SRR13786280), using the same FASTQ files and a limit of 5,000,000 read pairs for both tools, as specified by their respective built-in read-number parameters. how_are_we_stranded_here performs strand-orientation inference using Kallisto pseudoalignment (v0.50.1) together with RSeQC infer_experiment.py; resolveS is a standalone C/C++ program that detects strand orientation directly from FASTQ files. Runtime and peak memory were collected using GNU time, and CPU utilization was obtained from the %P metric. All tests were completed on the same compute node. See Supplementary File 9 for the reproducible pipeline and 1,000,000 read pairs running performance.

Discussion

Practical guidelines and limitations

resolveS infers strandedness from the orientation of reads aligned to residual rRNA. Because this signal is limited, occasional misclassification cannot be excluded, particularly for samples with sparse or atypically distributed rRNA, and several practical considerations improve interpretation. First, when a high-quality reference genome and reliable annotation are available, inference based on protein-coding genes should be preferred or used as independent validation; the advantage of resolveS lies in settings where such resources are unavailable or impractical. Second, for low-confidence, fallback, or near-threshold calls, users may raise the read count with the `-u` parameter, for example to 10 or 20 million, balancing the gain in evidence against computational time, and inspect the per-rRNA detail reported by `-d`. Third, samples from the same experimental batch are commonly prepared with the same protocol and are expected to share strandedness, so batch-mode results should be examined jointly. Ambiguous calls should not be interpreted in isolation or treated as definitive solely on the reported class.

Future directions

Future work could explore pseudoalignment methods (e.g., Salmon) to gain better performance and develop strandedness-specific scoring strategies instead of relying on MAPQ. In addition, although the current rRNA database provides broad taxonomic coverage, its composition may introduce bias for highly divergent or underrepresented organisms such as extremophilic archaea and rare protists, where reduced representation of homologous sequences could decrease alignment sensitivity and affect accuracy. Future versions should therefore expand and periodically update the reference database, with systematic benchmarking of the effects on efficiency, robustness, and accuracy before adoption.

Extension to RNA secondary structure profiling

To evaluate downstream utility, we benchmarked resolveS on four public chemical-probing datasets (30 samples, BD5 in [Supplementary File 8](#)^[8]). These methods map RNA secondary structures via reverse-transcription signals on short cDNA fragments^[22,23]. Established strand-specific protocols like icSHAPE serve as ideal positive controls for validation. Because these libraries yield orientation-sensitive reads, they provide a stringent test for strand-aware inference. resolveS recovered the expected strandedness in all 30 samples (100%), including perfect agreement in the single-end datasets (20/20).

The nominally lower single-end consensus rate in our benchmark was driven entirely by a single unstranded study (GSE103492; [Supplementary File 8](#)). Excluding it restored single-end agreement to 100% (349/349), confirming that strandedness is dictated by library-construction chemistry rather than read architecture.

Installation and documentation

resolveS is freely available under the MIT license at <https://github.com/yudalang3/resolveS>, which provides the source code, installation instructions, example commands, and documentation covering database setup, indexing, and routine usage. Portable distributions and ready-to-use singularity/apptainer container images are available from the release page. To serve both zero-code biologists and pipeline developers, resolveS emits machine-readable tab-separated results (one row per sample, with the inferred library type, detection level, MAPQ tier, and Rel_Diff) alongside human-readable logs, plus an optional per-rRNA breakdown in debug mode.

Conclusions

We developed resolveS, a rapid and genome-independent method for inferring RNA-seq library strandedness using residual rRNA reads and a universal rRNA reference. Across benchmark datasets, resolveS achieved high concordance with known library types while typically completing analysis within seconds per sample using less than 0.5 GiB of memory. Its confidence-aware inference framework further enables ambiguous or low-evidence cases to be identified for cautious interpretation. By eliminating the requirement for species-specific reference genomes and annotations, resolveS provides a practical and scalable solution for RNA-seq quality control, public data reanalysis, and studies of non-model organisms.

Author contributions

The authors confirm their contributions to the paper as follows: study conception and design, supervision: Yu D, Zhao T; methodology, software, formal analysis, investigation, resources, data curation, draft manuscript preparation: Yu D, Long X; validation, visualization, writing – review & editing: Yu D, Long X, Zhao T; project administration, funding acquisition: Yu D. All authors reviewed the results and approved the final version of the manuscript.

Data availability

All benchmark datasets are publicly available from the Sequence Read Archive; the complete list of run accessions for the 252 benchmark samples is provided in [Supplementary File 3](#). Plant datasets are from Shi et al., animal cell-line datasets from Naphade et al. and Sigurgeirsson et al., and the large-scale evaluation collection from Signal and Kahlke. Per-sample resolveS outputs and intermediate result tables are included in the supplementary materials. resolveS is freely available under the MIT license at <https://github.com/yudalang3/resolveS>. The repository provides the source code, installation instructions, example commands, demonstration data, and documentation for database setup and indexing (<https://github.com/yudalang3/resolveS/tree/main/docs>). Portable releases and ready-to-use Singularity/Apptainer container images are available at <https://github.com/yudalang3/resolveS/releases>. Scripts reproducing the analyses in this revision are provided in the Supplementary materials.

Acknowledgments

We thank Professor Lin Li for helpful discussions on RNA-seq analysis of cell signal transduction. This work was supported by the Youth Fund of the National Natural Science Foundation of China (Grant No. 32400511).

Conflict of interest

The authors declare that they have no conflict of interest.

Supplementary information accompanies this paper online at: <https://doi.org/10.48130/gcomm-0026-0020>.

Dates

Received 2 June 2026; Revised 12 August 2026; Accepted 24 August 2026; Published online 29 September 2026

References

- [1] Sahraeian SME, Mohiyuddin M, Sebra R, Tilgner H, Afshar PT, et al. 2017. Gaining comprehensive biological insight into the transcriptome by performing a broad-spectrum RNA-seq analysis. *Nature Communications* 8:59
- [2] Conesa A, Madrigal P, Tarazona S, Gomez-Cabrero D, Cervera A, et al. 2016. A survey of best practices for RNA-seq data analysis. *Genome Biology* 17:13
- [3] Hong M, Tao S, Zhang L, Diao LT, Huang X, et al. 2020. RNA sequencing: new technologies and applications in cancer research. *Journal of Hematology & Oncology* 13:166
- [4] Gondane A, Itkonen HM. 2023. Revealing the history and mystery of RNA-seq. *Current Issues in Molecular Biology* 45:1860–1874
- [5] Stark R, Grzelak M, Hadfield J. 2019. RNA sequencing: the teenage years. *Nature Reviews Genetics* 20:631–656
- [6] Pust MM, Mohamed AMT, Stražar M, Arias-Rojas A, Cunningham-Oakes E, et al. 2026. Antisense transcription reveals disease-associated adaptations in the human gut microbiome. *Nature Microbiology* 11:2464–2477
- [7] Chen CH, Pan CY, Lin WC. 2019. Overlapping protein-coding genes in human genome and their coincidental expression in tissues. *Scientific Reports* 9:13377
- [8] Zhou T, Hong Y, Li P, Liu X, Zhang J, et al. 2026. Modtector: ultra-fast modification signal mining on mapped sequencing reads. *Bioinformatics* 42:btg627
- [9] Liao Y, Smyth GK, Shi W. 2014. featureCounts: an efficient general purpose program for assigning sequence reads to genomic features. *Bioinformatics* 30:923–930
- [10] Haas BJ, Papanicolaou A, Yassour M, Grabherr M, Blood PD, et al. 2013. De novo transcript sequence reconstruction from RNA-seq using the Trinity platform for reference generation and analysis. *Nature Protocols* 8:1494–1512
- [11] Signal B, Kahlke T. 2022. How_are_we_stranded_here: quick determination of RNA-Seq strandedness. *BMC Bioinformatics* 23:49
- [12] Wang L, Wang S, Li W. 2012. RSeQC: quality control of RNA-seq experiments. *Bioinformatics* 28:2184–2185
- [13] Neph S, Kuehn MS, Reynolds AP, Haugen E, Thurman RE, et al. 2012. BEDOPS: high-performance genomic feature operations. *Bioinformatics* 28:1919–1920
- [14] Robinson JT, Thorvaldsdóttir H, Winckler W, Guttman M, Lander ES, et al. 2011. Integrative genomics viewer. *Nature Biotechnology* 29:24–26
- [15] Langmead B, Salzberg SL. 2012. Fast gapped-read alignment with Bowtie 2. *Nature Methods* 9:357–359
- [16] Shi W, Wang L, Zhou N, Zhang M, Hao Y, et al. 2026. Nanopore sequencing and multiomics reveal predictable non-coding RNA activation in DNA methylation deficient *Arabidopsis thaliana*. *Communications Biology* 9:175
- [17] Naphade S, Bhatnagar R, Hanson-Smith V, Choi I, Zhang A. 2022. Systematic comparative analysis of strand-specific RNA-seq library preparation methods for low input samples. *Scientific Reports* 12:1789
- [18] Sigurgeirsson B, Emanuelsson O, Lundeberg J. 2014. Analysis of stranded information using an automated procedure for strand specific RNA sequencing. *BMC Genomics* 15:631
- [19] Kopylova E, Noé L, Touzet H. 2012. SortMeRNA: fast and accurate filtering of ribosomal RNAs in metatranscriptomic data. *Bioinformatics* 28:3211–3217
- [20] Li H, Durbin R. 2009. Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics* 25:1754–1760
- [21] Shen W, Lees JA, Iqbal Z. 2026. Efficient sequence alignment against millions of prokaryotic genomes with LexicMap. *Nature Biotechnology* 44:1383–1390
- [22] Spitale RC, Flynn RA, Zhang QC, Crisalli P, Lee B, et al. 2015. Structural imprints in vivo decode RNA regulatory mechanisms. *Nature* 519:486–490
- [23] Flynn RA, Zhang QC, Spitale RC, Lee B, Mumbach MR, et al. 2016. Transcriptome-wide interrogation of RNA secondary structure in living cells with icSHAPE. *Nature Protocols* 11:273–290



Copyright: © 2026 by the author(s). Published by Maximum Academic Press, Fayetteville, GA. This article is an open access article distributed under Creative Commons Attribution License (CC BY 4.0), visit <https://creativecommons.org/licenses/by/4.0/>.