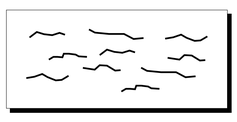
Evaluation of *de novo* Transcriptome Assemblies from RNA-Seq Data

Nathanael Fillmore, based on work with Bo Li and Colin Dewey
University of Wisconsin, Madison, Computer Sciences

Introduction

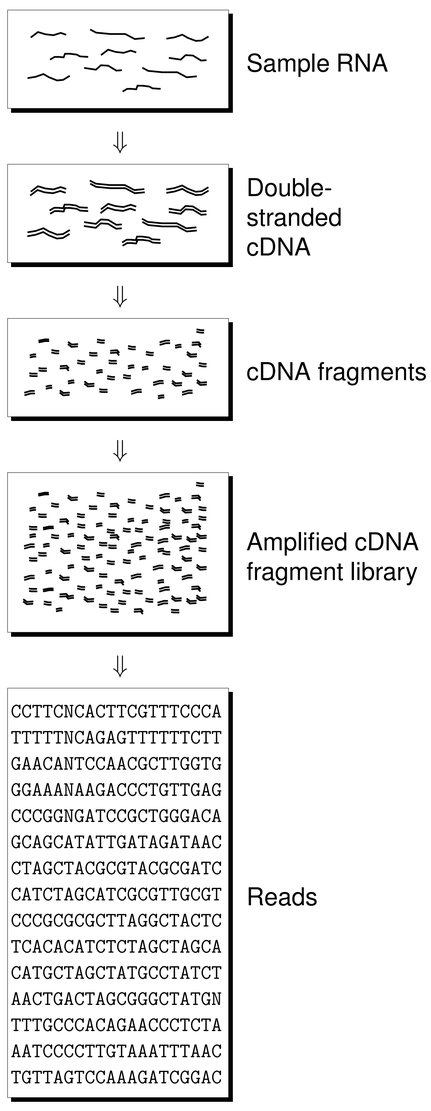
Transcriptome

The transcriptome is the collection of all RNA transcripts in a cell or sample.



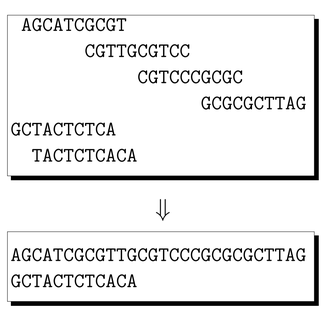
RNA-Seq

RNA-Seq is a method to sequence the transcriptome.



De novo assembly

In de novo assembly, one attempts to reconstruct the original transcripts, based only on the reads.



It is not necessarily possible to recover all the transcripts. Thus, the elements of an assembly are called contigs, contiguous (putative) subsequences of the original transcripts.

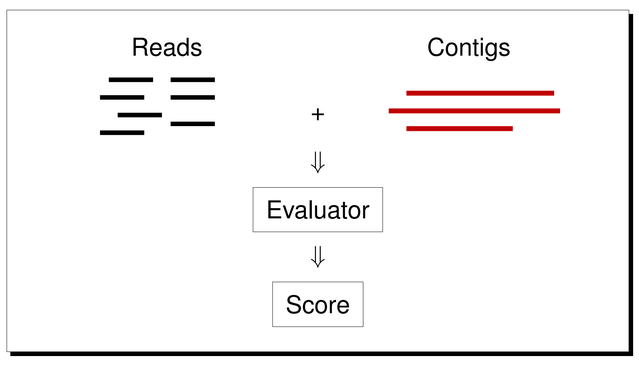
Status quo

Several *de novo* assembly programs exist, but it is difficult to evaluate the quality of the assemblies they produce. Different programs, or even different parameter settings for the same program, often produce substantially different assemblies, given the same read set as input. Complicating factors:

- Non-uniform expression - hence, one cannot assume that all transcript sequences have equal coverage.
- Alternative splicing - hence, different transcripts will often share large subsequences.

Our goal

Our goal is to evaluate *de novo* transcriptome assemblies, without a ground truth reference.



Related work

- Reference-based metrics:
  - Transcript/nucleotide level sensitivity/specificity.
  - But we do not have true transcript sequences.
- Reference-free metrics:
  - Crude metrics such as N50, median contig length.
  - N50 can be misleading.
- *De novo* genome assembly evaluation:
  - Rahman and Pachter 2013, Genome Biology.
  - A simpler task, since chromosomes are sequenced at the same level of coverage.

Nathanael Fillmore has been supported by NLM training grant 5T15LM007359. Bo Li has been supported by the Morgridge Institute for Research support for Computation and Informatics in Biology and Medicine. Colin Dewey has been partially funded by NIH grant 1R01HG005232.

Methods

Our contribution

Our contribution is a transcriptome assembly scoring function, which can be used to choose the best assembly from a collection of candidate *de novo* assemblies when no ground-truth reference is available. The score is based on a statistical model of the process of RNA-Seq read generation and of ideal transcriptome assembly. A software implementation has been developed and will be released in the near future.

Our score

Our score is defined as the joint probability of the assembly and the reads:

$$\text{score(assembly)} = P(\text{assembly, reads}).$$

If the read set is held fixed, the score is proportional to the posterior probability of the assembly, given the reads, since $P(\text{assembly}, \text{reads}) \propto P(\text{assembly}|\text{reads})$.

The score is decomposed into prior and likelihood components, as follows:

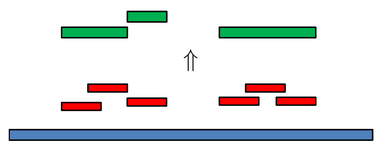
$$P(\text{assembly}, \text{reads}) = \int \underbrace{P(\text{assembly}, \text{coverage})}_{\text{prior}} \underbrace{P(\text{reads}|\text{assembly}, \text{coverage})}_{\text{likelihood}} d\text{coverage}$$

A contig's "coverage" is the expected number of reads generated from each position of the contig's original transcript.

The prior

The prior distribution is based on the following assumptions:

- Transcript lengths follow a negative binomial distribution, iid.
- Given the transcript lengths:
  - Transcript sequences follow a uniform distribution, iid.
  - The number of reads starting at each position of a transcript follows a Poisson distribution (mean = coverage), iid.
- The ideal assembly is formed by joining reads whose true positions (within the transcript set) overlap or are contiguous.



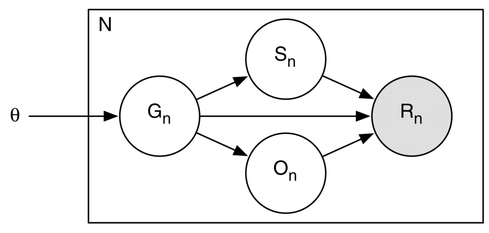
Based on the above, one can work out a recurrence for the prior probability of the assembly and coverage.

Practical contribution of the prior:

- The prior on transcript lengths penalizes contigs with aberrant lengths.
- The prior on transcript sequences penalizes contigs with too many nucleotides.

The likelihood

Previous work, RSEM, introduced a generative model of reads, given transcripts and their expression:*



where

- θ_j is the expression of transcript j.
- N is the number of reads.
- G_n is the transcript read n comes from.
- S_n is the start position of read n within its transcript.
- O_n is the orientation of read n within its transcript.
- R_n is read n.

Key observation:

- Generating from contigs $\equiv$ generating from transcripts, except that contigs are guaranteed to be covered by reads.

Therefore, we define the likelihood to be the probability of the reads given the contigs, according to RSEM's model, divided by the probability that the contigs are covered by reads.

Practical contribution of the likelihood:

- On one hand, the likelihood penalizes contigs that are not well-supported by reads.
- On the other hand, the likelihood penalizes assemblies that do not make use of all the reads.

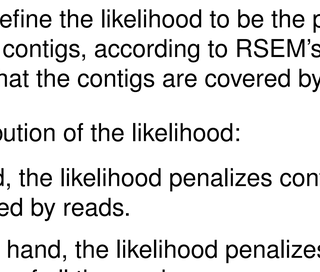
* The actual model is more complicated than this picture shows.

The integral

We approximate the integral over coverage using the Bayesian information criterion (BIC):

$$\begin{aligned}&\log P(\text{assembly}, \text{reads}) \\ &= \log \int P(\text{assembly}, \text{coverage}, \text{reads})\, d\text{coverage} \\ &\approx \log P(\text{assembly}, \text{reads}|\text{coverage}^*) - \frac{1}{2} M \log N\end{aligned}$$

where M = number of contigs, N = number of reads, coverage* = maximum likelihood estimate.



Practical contribution of the BIC term:

- The BIC term penalizes assemblies with too many contigs.

Figure from Bishop, *Pattern Recognition and Machine Learning*, Springer, 2009.

Results

Experiment 1 - Setup

This experiment demonstrates that if we perturb the ideal assembly, the score decreases, on average.

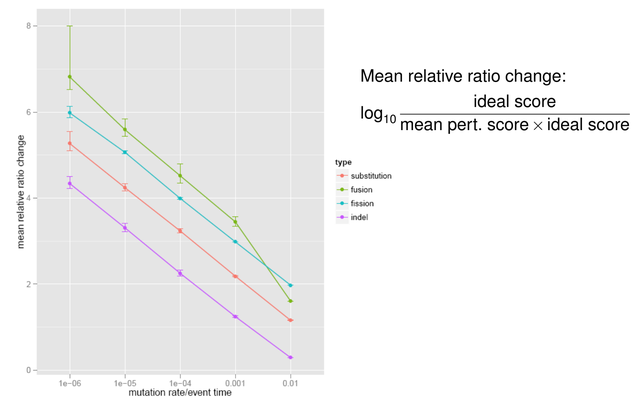
Procedure:

- Construct the ideal assembly, based on simulated data.
- Perturb this assembly:
  - Substitution - substitute a base.
  - Fusion - join two contigs into one contig.
  - Fission - split one contig into two contigs.
  - Indel - insert or delete a fragment from a contig.
- Compute score for ideal and perturbed assemblies.

Experiment 1 - Results

For all four types of perturbation, at all rates of perturbation, the mean score of the perturbed assemblies is lower than the score of the ideal assembly.

The following plot shows the mean relative ratio change of the ideal versus perturbed scores, at different rates of perturbation. As the rate of perturbation increases, the perturbed assemblies' score decreases, on average.



Experiment 2 - Setup

This experiment demonstrates that on both real and simulated data, our reference-free score has high correlation with several simple reference-based scores.

Procedure:

- For each dataset (real mouse and real simulated):
  - Create ∼100 assemblies, by running the *de novo* assemblers Trinity, Oases, SOAPdenovo-trans, and Trans-ABySS with different parameter settings.
  - For each assembly, compute both our reference-free score and the three reference-based scores described below.

We compare to the following reference-based scores:

- Nucleotide F1 - The nucleotide recall is the fraction of nucleotides in the reference transcript set that are correctly predicted in the assembly. The nucleotide precision is the fraction of nucleotides in the assembly that correctly predict a nucleotide in the reference. The nucleotide F1 is the harmonic mean. All nucleotides are weighted by nucleotide expression.
- Transcript F1 - The transcript recall is the fraction of transcripts in the reference transcript set that are correctly predicted in the assembly. The transcript precision is the fraction of contigs in the assembly that correctly predict a transcript in the reference. The transcript F1 is the harmonic mean. All transcripts are weighted by transcript expression. A transcript is correctly predicted if $\geq 95\%$ of its nucleotides are correctly predicted. A contig correctly predicts a transcript if $\geq 95\%$ of the contig's bases correctly predict a nucleotide.
- *k*mer JS divergence - Let $K = \{\mathtt{A}, \mathtt{T}, \mathtt{C}, \mathtt{G}\}^k$ be the set of all possible *k*mers. Each assembly induces a probability distribution over K by counting how many times each *k*mer occurs in the assembly and normalizing. The oracleset also induces such a probability distribution. The Jensen-Shannon divergence measures how close two such distributions are to each other.

Experiment 2 - Results

The Spearman correlation between our score and the three reference-based scores is high on both the real and simulated datasets.