**Before you start… Garbage in, garbage out**

-Assembly quality must first be high. Complete and contiguous.

-Quality measured by N50 and average scaffold gap size (diagram)

**ANNOTATION**

Bare minimum annotate protein coding genes

Pipeline = collection of bioinformatic tools

**Phase 1: Computation**

**1. Repeat identification and masking**

-low complexity sequences, eg. homopolymeric runs of nt

-transposable elements eg. viruses, LINEs, SINEs

Why remove?

-BLAST may align to repeats -> false gene annotations

-transposon ORFs look like true exons to gene predictors; additional exons tacked onto true genes

Challenges:

-a lot of them

-borders hard to define

-often poorly conserved -> make repeat library for genome

-How? homology-based or de novo (de novo risks removing true protein coding genes with repetitive elements)

Steps:

a. create repeat library

b. use repeat masker to id stretches of sequences in target genome homologous to known repeats (eg. BLAST or Crossmatch)

c. mask to allow downstream programs know these are repeats

hard mask: replace repeating nts with N

soft mask: write repeating nts in lower case

**2. Evidence alignment:**

-align protein, EST, RNA-seq sequences to genome assembly

-from previous work on animal or homologous proteins from other animals

-homologous proteins obtained from databases eg. UniProtKB/SwissProt, NCBI

How?

A. Alignment:

a. BLAST and BLAT (explain algorithm quickly) - align seqs to genome to ID homologous regions (metrics like % similarity and % identity used to determine ‘true’ alignments)

b. Cluster the evidence by gene. Why? Organizes and amalgamates the many types of data, removes redundant evidence eg. many identical ESTs for a highly expressed gene

B. Polishing: identify splice sites and exon boundaries

Use splice-aware programs to realign ESTs, mRNAs, and protein to the genome.

RNA-seq data most effectively identifies exons, splice sites, and alternatively spliced exons (seqs are mRNA). Two way to use RNAseq data:

1. assemble a transcriptome de novo and align to genome

2. align reads directly to genome, assemble alignments intro transcripts

(How to choose tools)

**3. Ab initio gene prediction**

-Use mathematical models to identify genes

Pros: no external data needed

Cons:

-report the single most likely CDS, does not include any alternatively spliced transcripts

-does not find UTRs

-require a very high quality assembly

-sensitivity near 100% but accuracy of intron-exon bounaries 60-70%

-Must train the program. How?

-genes are characterized via organism-specific traits (eg. codon frequency, distribution of intron-exon lengths)

-use precalculated paramter files; not good for distantly related animals

-train the program on the genome you’re working on - input is alignment of ESTs, RNAseq and protein seqs

Increasingly programs combine ab initio and evidence-driven gene prediction

**Phase 2: Annotation**

Approach 1

-Run several different gene finders (i.e. the evidenced-based or ab initio gene prediction programs) and use a ‘chooser’ or ‘combiner’ algorithm to select the one prediction whose intron-exon structure is closest to consensus overlap of all the gene models proposed by the programs

-Can also estimate the type and frequency of errors made by each type of evidence and choose a combination of the evidence that minimizes these kinds of errors before making the best choice

Approach 2

Feed evidence driven prediction alignments to gene predictors at run time to improve prediction accuracy (i.e. combining evidence-driven and ab initio gene finders?), then use the ‘chooser’ to select the most representative prediction (I became a little lost here, should check paper again.)

Before or after the chooser you can add UTRs (as identified by RNA-seq and ESTs)

Choice of approach depends on computational burdens; better to use a more thorough annotation approach for genomes of distantly related animals

**VISUALIZING ANNOTATION**

Output of annotation pipelines are:

-FASTA files of transcript and protein sequences from gene models

-Genome organized in genome browser and genome database

-File formats (GenBank, GFF3, GTF, EMBL) that describe: intron-exon boundaries, start/stop codons, UTRs, alternative transcripts, documentation about the sequence alignments and gene predictions supporting each model

**QUALITY CONTROL**

-incorrect annotations can incorrectly inform the annotations of other genomes

-difficult to measure quality without a reference

How?

a. Experimental verification - not practical for an entire genome

b. Overall estimates: Find teh percentage of annotations that encode proteins with known protein domains. This is usually pretty consistent at ~57-75%. Poor results are at 5-25%.

c. Gene-specific estimates: Check whether protein, EST, and RNA-seq evidence support or contradict the intron-exon boundaries (see AED score)

Error correction

-can edit intron-exon coordinates manually - via browser

-Annotation Jamborees: widescale collaboration on manual curation of a genome’s annotations; infrastructure eg. internet required

**MAKING DATA PUBLICLY AVAILABLE**

How?

-build your own genome database and publish

-submit to GenBank or Ensembl

-submit to theme-baesd genome databases

Updating annotations: merge new annotations with old ones.

.

Functional annotation

**Functional annotation** is defined as the process of collecting information about and describing a **gene's** biological identity—its various aliases, molecular **function**, biological role(s), subcellular location, and its expression domains within the plant.