TEAM 2; GROUP 2

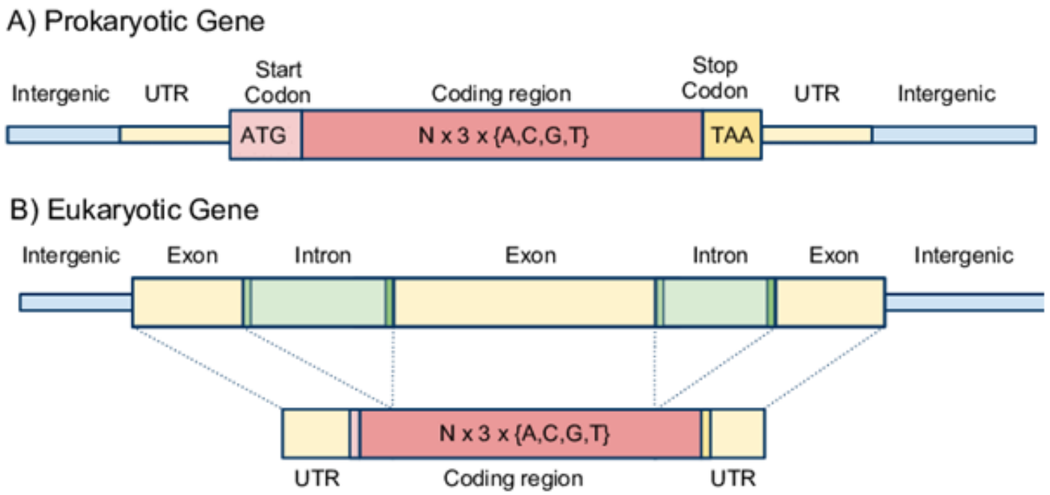
GENE PREDICTION

BACKGROUND AND STRATEGY

Gautham Krishna Sankar Ramalaxmi
Jay Ayr Wroe
Jyothi Guruprasad
Kaize H Ali
Lee Ellen Mullins
Varsha Srinivasan

WHAT IS GENE PREDICTION?

- The process of identifying gene-encoding regions
- One of the first and most important steps in understanding the genome once it has been sequenced
- One of the key steps in genome annotation following sequence assembly, filtering of non-coding regions and repeat masking



WHAT DOES GENE PREDICTION DO?

- Annotates large, contiguous sequences
- Distinguishes between coding and non-coding sequences of a genome
- Identifies fundamental elements of a genome such as functional genes, intron, exon, splicing sites, regulatory sites, protein-coding genes, motifs, etc.
- Predicts complete intron-exon structures of protein-coding regions
- Applied in structural genomics, functional genomics, metabolomics, transcriptomics, proteomics, genome studies and other genetic related studies including genetics disorders detection, treatment and prevention.

METHODS

There are two methods of gene prediction:

- Homology-based prediction
  - Approach: Finding similarity in gene sequences
- Ab-initio prediction
  - Approach: Gene structure and signal-based searches

HOMOLOGY-BASED PREDICTION

- Based on finding similarity in gene sequences between the target genome and ESTs, mRNA, and protein products
- Based on the assumption that functional regions (exons) are more conserved
- Similarity information can be used to infer gene structure or function of that region
- There are two methods of performing similarity-based prediction:
  - Local alignment (Ex. BLAST)
  - Global alignment (Ex. Genewise)

HOMOLOGY-BASED PREDICTION TOOLS

There are two commonly-used tools that employ this method of gene prediction:

- BLAST
- FASTA

BLAST

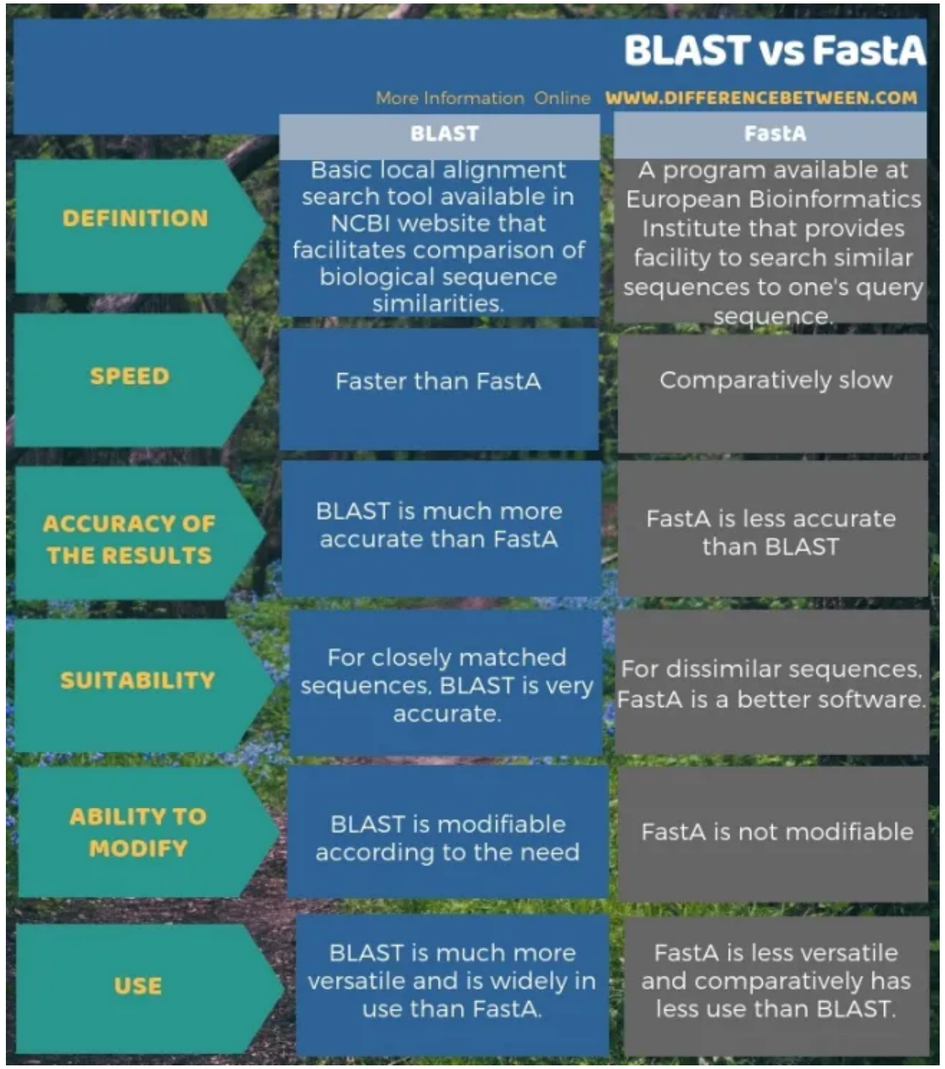
- Identifies homologous sequences by searching one or more databases
- Returns a collection of local alignments called high-scoring segment pairs (HSPs)
- Parameters used – word size, match/mismatch scores, matrix, gap costs
- Disadvantage – Seldom identifies multiple short similar sequences while searching for coding sequences/proteins
- To tackle these issues, the following tools have been implemented:
  - GenBlastG
  - GeMoMa
  - Exonerate

Option	Query Type	DB Type	Comparison	Note
blastn	Nucleotide	Nucleotide	Nucleotide-Nucleotide	
blastp	Protein	Protein	Protein-Protein	
tblastn	Protein	Nucleotide	Protein-Protein	The database is translated into protein
blastx	Nucleotide	Protein	Protein-Protein	The queries are translated into protein
tblastx	Nucleotide	Nucleotide	Protein-Protein	The queries and database are translated into protein

FASTA

- Compares input sequences with the existing database using local sequence alignment
- Calculates statistical significance between the sequences
- Parameters used – threshold, true homology, e-value, putative conserved domains
- Provides local and global alignment search options using SSEARCH and GGSEARCH respectively

BLAST VS. FASTA

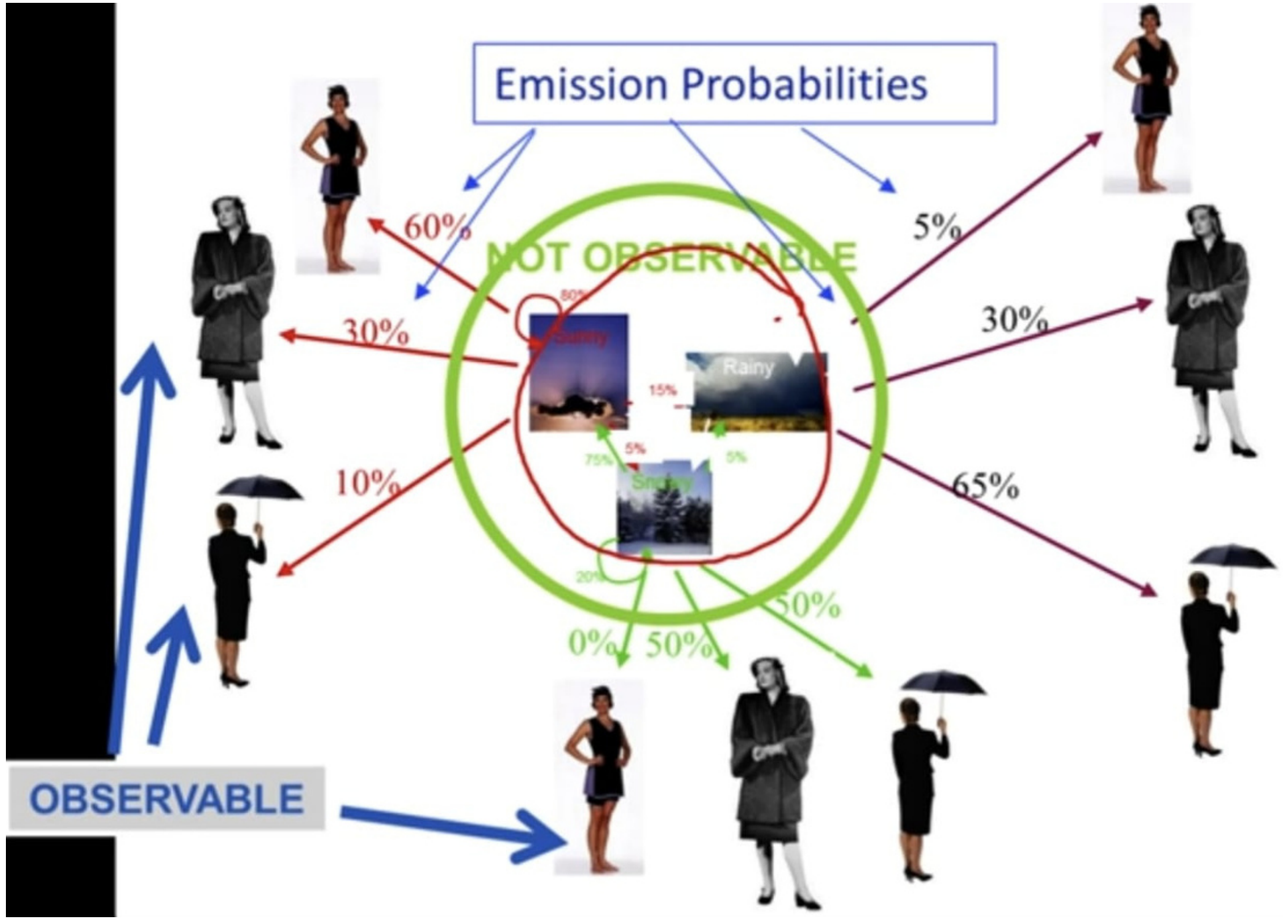


MATCHING PROGRAMS IN BLAST & FASTA

Table 1: Comparison programs in the FASTA36 package

FASTA program	BLAST equiv.	Description
`fasta36`	`blastp`/ `blastn`	Compare a protein sequence to a protein sequence database or a DNA sequence to a DNA sequence database using the FASTA algorithm [15, 17]. Search speed and selectivity are controlled with the *ktup*(wordsize) parameter. For protein comparisons, *ktup* = 2 by default; *ktup* =1 is more sensitive but slower. For DNA comparisons, *ktup*=6 by default; *ktup*=3 or *ktup*=4 provides higher sensitivity.
`ssearch36`		Compare a protein sequence to a protein sequence database or a DNA sequence to a DNA sequence database using the Smith-Waterman algorithm [21]. `ssearch36` uses SSE2 acceleration, and is only 2 - 5X slower than `fasta36` [5].
`ggsearch36`/ `glsearch36`		Compare a protein sequence to a protein sequence database or a DNA sequence to a DNA sequence database using an optimal global:global (`ggsearch36`) or global:local (`glsearch36`) algorithm.
`fastx36`/ `fasty36`	`blastx`	Compare a DNA sequence to a protein sequence database, by comparing the translated DNA sequence in three frames and allowing gaps and frameshifts. `fastx36` uses a simpler, faster algorithm for alignments that allows frameshifts only between codons; `fasty36` is slower but can produce better alignments because frameshifts are allowed within codons [25].
`tfastx36`/ `tfasty36`	`tblastn`	Compare a protein sequence to a DNA sequence database, calculating similarities with frameshifts to the forward and reverse orientations [25].
`fastf36`/ `tfastf36`		Compares an ordered peptide mixture, as would be obtained by Edman degradation of a CNBr cleavage of a protein, against a protein (`fastf`) or DNA (`tfastf`) database [10].
`fasts36`/ `tfasts36`		Compares set of short peptide fragments, as would be obtained from mass-spec. analysis of a protein, against a protein (`fasts`) or DNA (`tfasts`) database [10].
`lalign36`		Calculate multiple, non-intersecting alignments using the sim2 implementation of the Waterman-Eggert algorithm [22] developed by Xiaoqui Huang and Web Miller [7]. Statistical estimates are calculated from Smith-Waterman scores of shuffled sequences.

HIDDEN MARKOV MODEL (HMM)

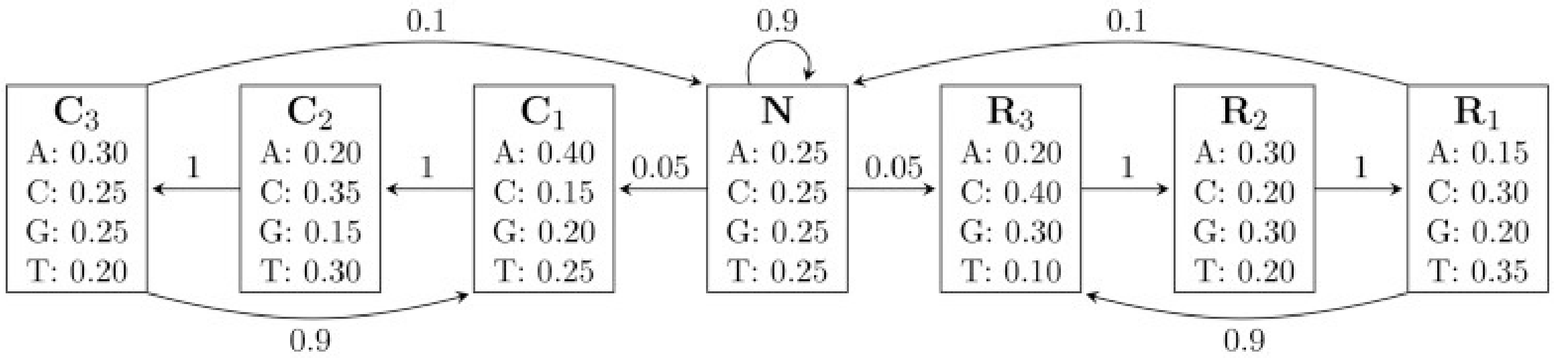


- Uses a stochastic method for randomly changing systems
- Here, the next state is dependent only on the current state

HIDDEN MARKOV MODEL (HMM)

- Observation probability estimation
  - Estimate the probability of observation sequence given the model
- Optimal hidden state sequence
  - Determine the optimal sequence of the hidden states
- Parameter estimation
  - Get the model parameters that maximize the probability of specific observations given specific states

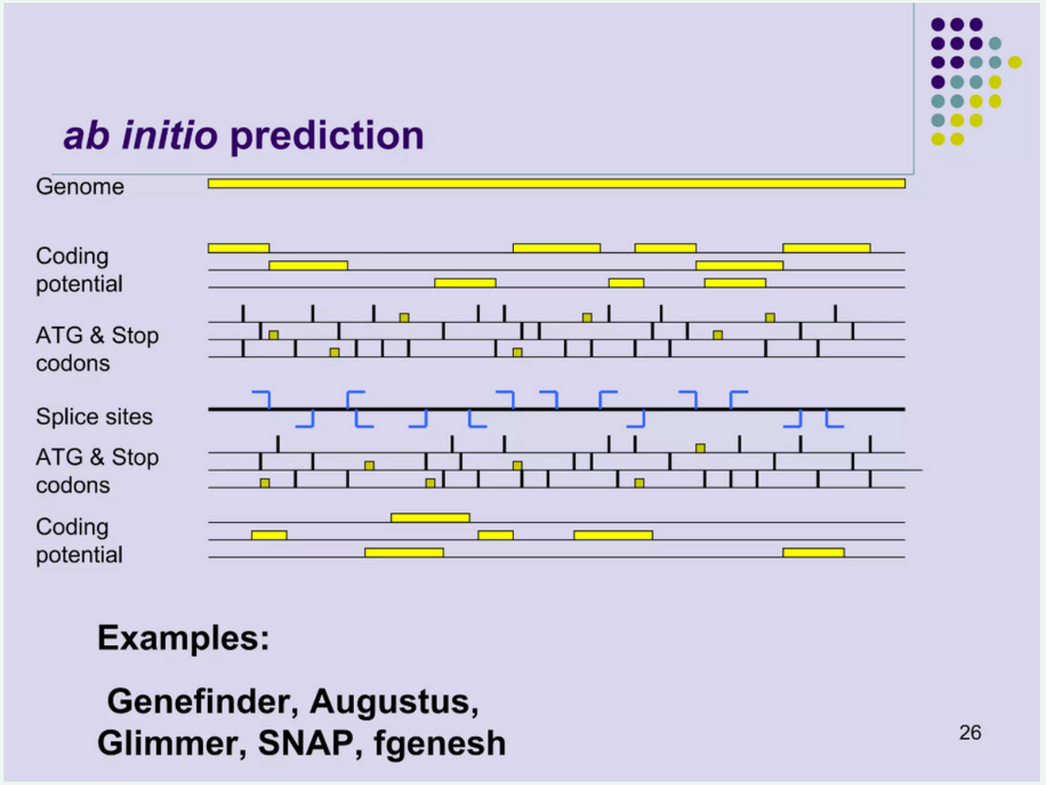
HMM FOR GENE PREDICTION



An HMM for gene prediction. Each box represents a hidden state, and the numbers inside are the emission probabilities of each nucleotide. Numbers on arcs are transition probabilities between hidden states.

AB-INITIO PREDICTION

- Predicts protein structures using only sequence information and no templates
- Utilizes component detection algorithms for promoter sequences and start and stop codons
- Relies on signal and content sensors
  - Signal sensors – short-sequence motifs
    - Splice sites and start and stop codons
  - Content sensors – Sequence analysis based on coding and non-coding regions
- Built on probabilistic models
  - Dynamic programming, neural networks, hidden Markov model (HMM), etc.



AB-INITIO PREDICTION TOOLS

There are three commonly-used tools that employ this method of gene prediction:

- GeneMarkS-2
- Glimmer
- Prodigal

GENEMARKS-2

- Latest iteration was developed to increase the prediction accuracy of where the gene starts
- Software is built using Hidden Markov Model
- The model which yields the highest log-odds score is selected
- Robust method of detection
- Disadvantage – Frameshifts and pseudogenes lead to a tendency to distract and provide issues when predicting

Genome Res. 2018 Jul; 28(7): 1079–1089.
doi: 10.1101/gr.230615.117

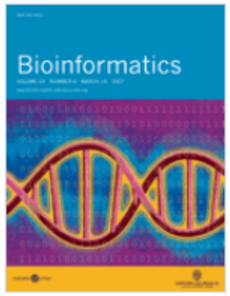
PMCID: PMC6028130
PMID: 29773659

Modeling leaderless transcription and atypical genes results in more accurate gene prediction in prokaryotes

Alexandre Lomsadze,[1,2,6] Karl Gemayel,[3,6] Shiyuyun Tang,[4] and Mark Borodovsky[1,2,3,4,5]

GLIMMER

- Has two built-in programs:
  - Build imm – Takes input set of sequences, then outputs an Interpolated Markov Model
  - Glimmer: Using the model from build imm, it identifies the presumed gene in genome
- Disadvantages:
  - High false positives compared to its counterparts
  - Has a unique output file format



Volume 23, Issue 6
march 2007

JOURNAL ARTICLE

Identifying bacterial genes and endosymbiont DNA with Glimmer

Arthur L. Delcher ✉, Kirsten A. Bratke, Edwin C. Powers, Steven L. Salzberg
Author Notes

Bioinformatics, Volume 23, Issue 6, march 2007, Pages 673–679,
https://doi.org/10.1093/bioinformatics/btm009

PRODIGAL

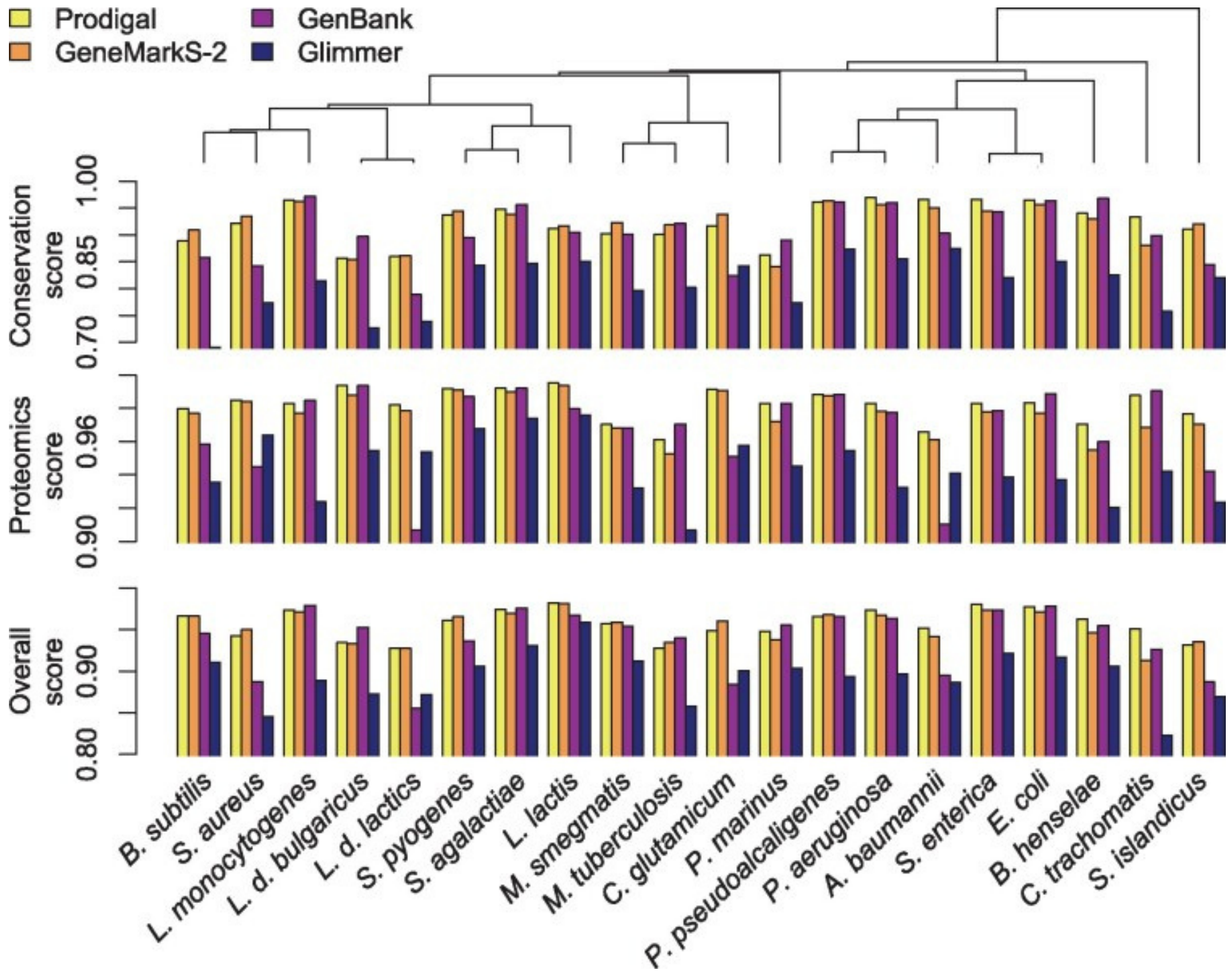
- Pros:
  - Superbly fast, lightweight and easy to use
  - Performs exceptionally well in GC rich content
  - Handles partial gaps and partial genes well
  - Uses dynamic programming
  - Lower false positives (<5%)
- Cons:
  - Since there is heavy emphasis on reducing false positives, potential negation of some predictions occur

Software | Open Access | Published: 08 March 2010

Prodigal: prokaryotic gene recognition and translation initiation site identification

Doug Hyatt, Gwo-Liang Chen, Philip F LoCascio, Miriam L Land, Frank W Larimer & Loren J Hauser

AB-INITIO TOOLS COMPARISON



HOMOLOGY VS. AB-INITIO METHODS

HOMOLOGY (PROS)

- Accurate
- Works for an increasing number of sequences
- Works for eukaryotes
- Reliable for known genes

AB-INITIO (PROS)

- Fast
- Not limited by existing knowledge
- Inexpensive
- Easy and lightweight tools

HOMOLOGY (CONS)

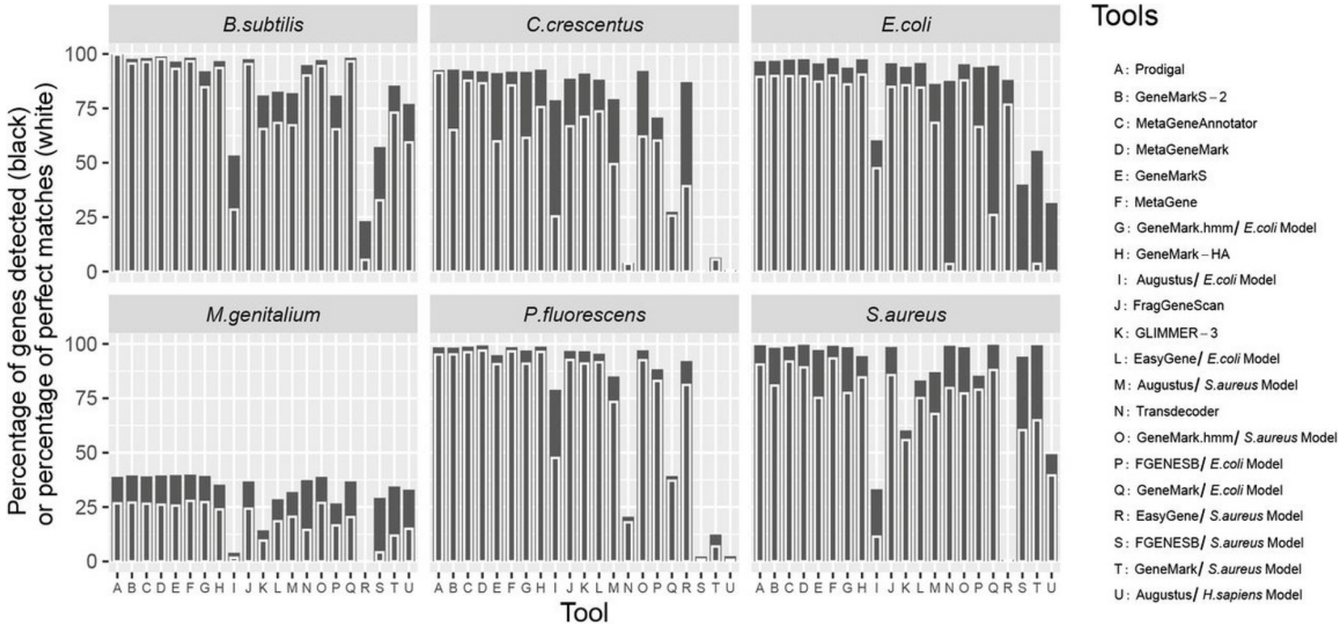
- Experimental errors could propagate and affect the whole process
- Limited by existing knowledge
- Requires large databases
- Does not have as many tools

AB-INITIO (CONS)

- Possibility of false positives
- Cannot be experimentally verified
- Preferred for prokaryotes
- Not as robust

ALGORITHM COMPARISON

- The diversity of prokaryotic genomes mean no single algorithm performs best
- Ab-initio approaches more reliable cross-species, but performance is still variable (note MetaGeneMark, col. D)
- Pooling CDSs from multiple algorithms increases false positives with negligible improvements in detection



Tools

A: Prodigal
B: GeneMarkS – 2
C: MetaGeneAnnotator
D: MetaGeneMark
E: GeneMarkS
F: MetaGene
G: GeneMark.hmm/ *E.coli* Model
H: GeneMark – HA
I: Augustus/ *E.coli* Model
J: FragGeneScan
K: GLIMMER – 3
L: EasyGene/ *E.coli* Model
M: Augustus/ *S.aureus* Model
N: Transdecoder
O: GeneMark.hmm/ *S.aureus* Model
P: FGENESB/ *E.coli* Model
Q: GeneMark/ *E.coli* Model
R: EasyGene/ *S.aureus* Model
S: FGENESB/ *S.aureus* Model
T: GeneMark/ *S.aureus* Model
U: Augustus/ *H.sapiens* Model

ALGORITHM COMPARISON

- The diversity of prokaryotic genomes mean no single algorithm performs best
- Ab-initio approaches more reliable cross-species, but performance is still variable (note MetaGeneMark, col. D)
- Pooling CDSs from multiple algorithms increases false positives with negligible improvements in detection

Model Organism	CEA CDS	Best Tool	Best Tool Detected [**Partial Matches**]	Agg' Detected [**Partial Matches**]	Best Tool CDSs	Agg' Extra CDSs [**Per' Increase**]
B. subtilis	4,011	MetaGeneAnnotator	99.85% [**1.40%**]	100% [**0.37%**]	4,058	1,692 [**41.09%**]
C. crescentus	3,737	MetaGeneMark	92.83% [**31.62%**]	93.66% [**23.17%**]	3,770	1,304 [**34.59%**]
E. coli	4,052	Prodigal	98.05% [**5.94%**]	98.82% [**1.57%**]	4,253	1,635 [**38.44%**]
M. genitalium	476	Prodigal	39.92% [**32.63%**]	40.13% [**30.89%**]	995	426 [**42.81%**]
P. fluorescens	5,178	GeneMarkS	99.29% [**12.97%**]	99.92% [**3.05%**]	5,513	1,891 [**34.03%**]
S. aureus	2,478	GeneMark.hmm (*S. aureus* model)	99.60% [**4.58%**]	99.84% [**0.28%**]	2,582	774 [**29.98%**]

Table 11: Numbers of additional CDSs predicted by Prodigal that can be added to Ensembl gene annotations. Additional CDSs are chosen if there are no fewer than 50 nucleotides overlapping with an Ensembl gene.

ORFORISE

- ORForise is an open-source pipeline that compares the CDS regions marked by a gene prediction algorithm to current ensemble annotations.
- Runs in python 3
- Produces a score for each algorithm applied to particular batch of sequencing data
- GeneMarkS-2, MetaGeneAnnotator, Prodigal frequently among best performing algorithms

- **M1** Percentage of Genes Detected
- **M2** Percentage of Predicted CDSs that Detected a Gene
- **M3** Percentage Difference of Number of Predicted CDSs
- **M4** Percentage Difference of Median Predicted CDS Length
- **M5** Percentage of Perfect Matches
- **M6** Median Start Difference of Matched Predicted CDSs
- **M7** Median Stop Difference of Matched Predicted CDSs
- **M8** Percentage Difference of Matched Overlapping Predicted CDSs
- **M9** Percentage Difference of Matched Short Predicted CDSs
- **M10** Precision
- **M11** Recall
- **M12** False Discovery Rate

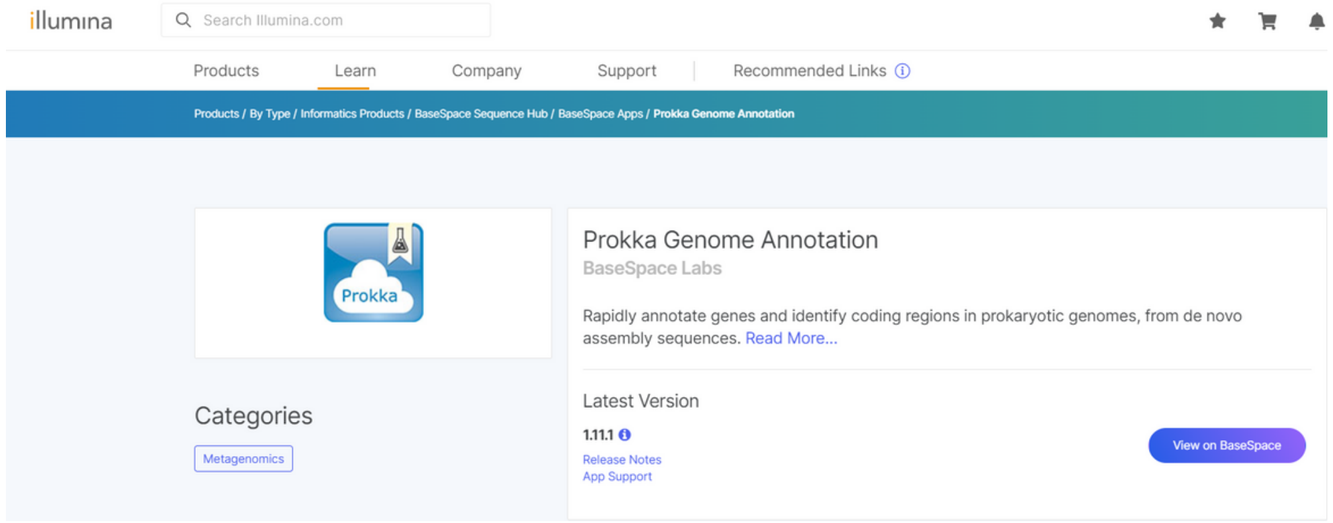
< Primary features measured and compared by ORForise

ANNOTATION PIPELINES

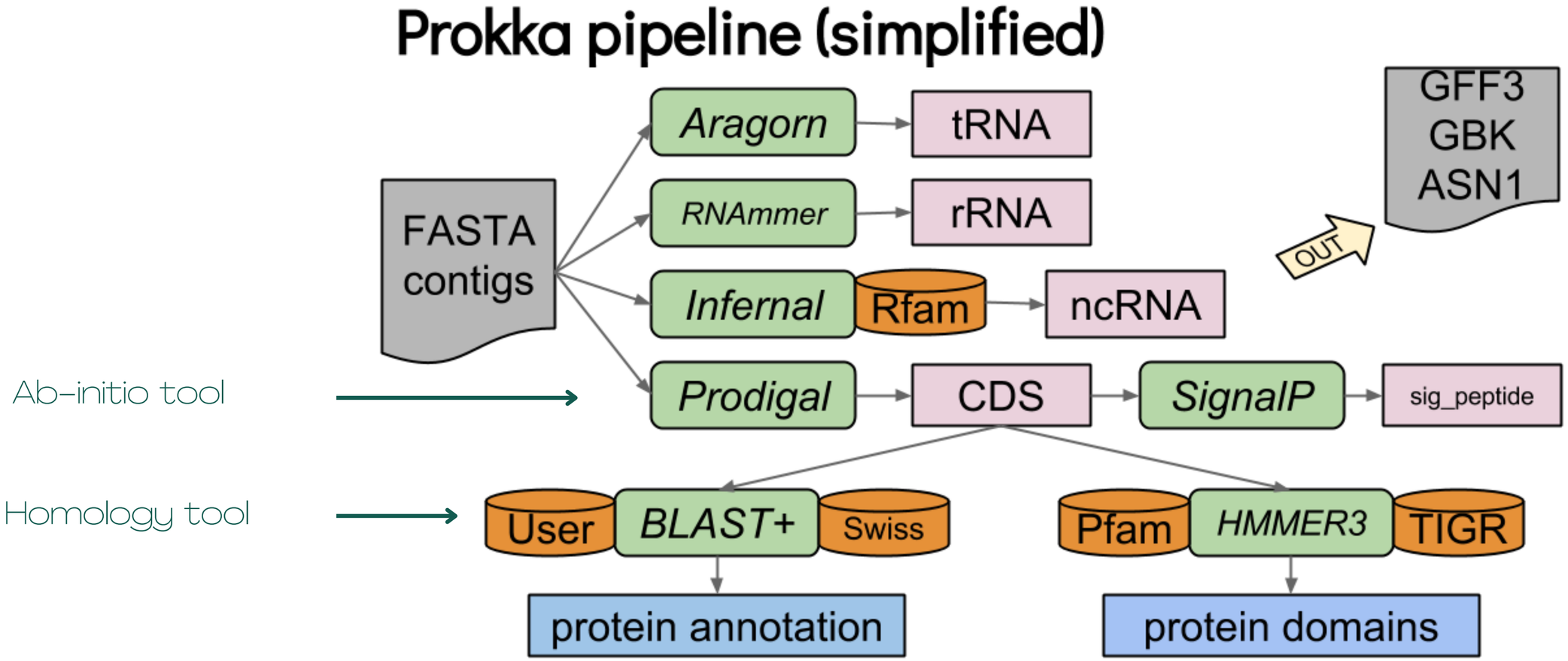
- Combination of tools
  - Homology (BLASTs), Ab Initio (HMMs), rRNA predictors, frameshift detectors
- Homology-based:
  - rRNA, tRNA highly conserved; classify clades
  - Orients HMM models, specifically PGAP, by providing “hints”
- Ab-initio / HMM models:
  - Prodigal, GeneMark tools
  - Predict CDSs / proteins

PROKKA

- Prokaryotic rapid genome annotation
- Uses Prodigal (ab initio) and BLAST (homology) tools, as well as others
- Runs in ~ 10 mins for 4MB assembly
- "… the NCBI provides a Prokaryotic Genomes Automatic Pipeline service via email, with a turn-around time measured in days." – Torsten Seemann, Prokka: rapid prokaryotic genome annotation, *Bioinformatics*, Volume 30, Issue 14, July 2014, Pages 2068–2069

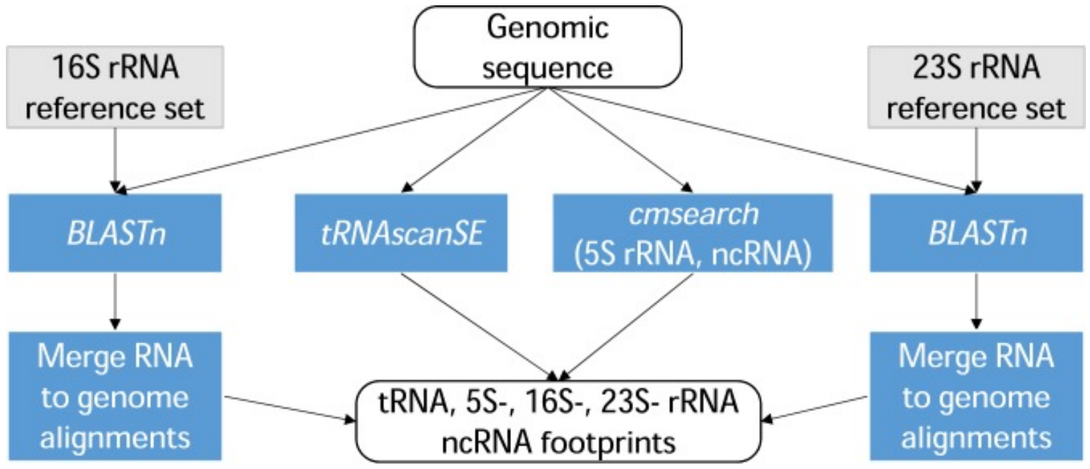


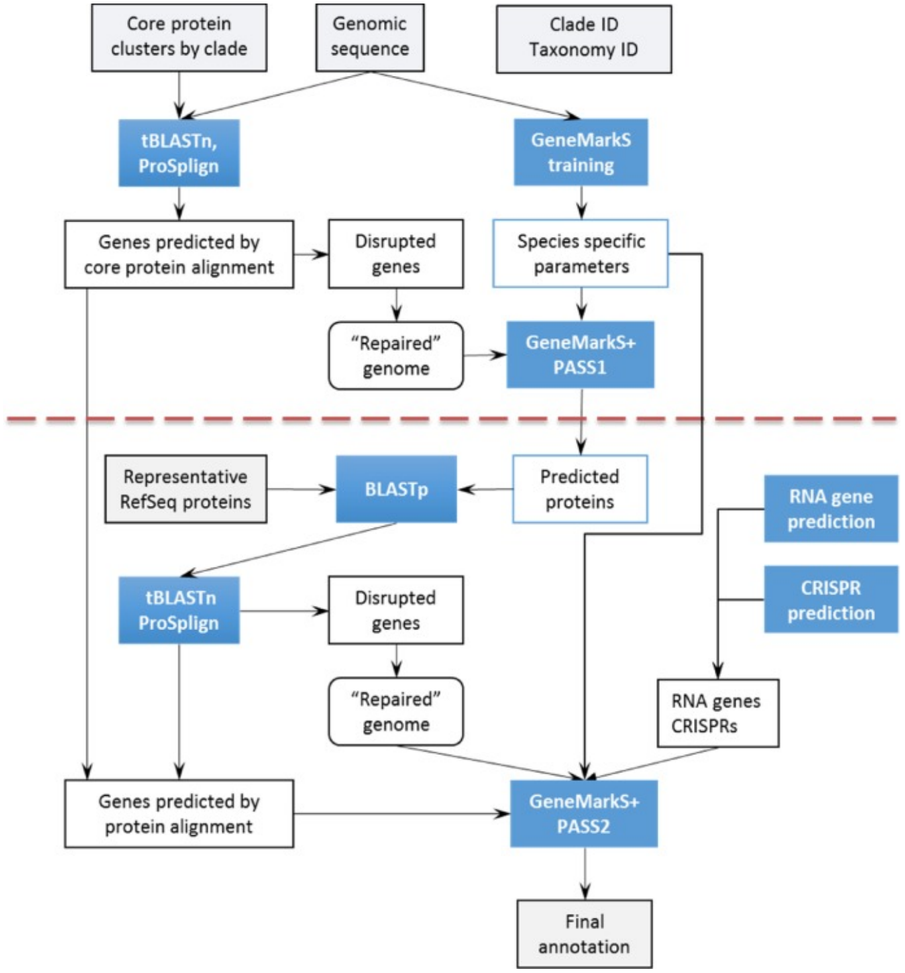
PROKKA


Prokka pipeline (simplified)
FASTA contigs
Aragorn
tRNA
RNAmmer
rRNA
Infernal
Rfam
ncRNA
Prodigal
CDS
SignalP
sig_peptide
OUT
GFF3
GBK
ASN1
User
BLAST+
Swiss
protein annotation
Pfam
HMMER3
TIGR
protein domains
Ab-initio tool
Homology tool

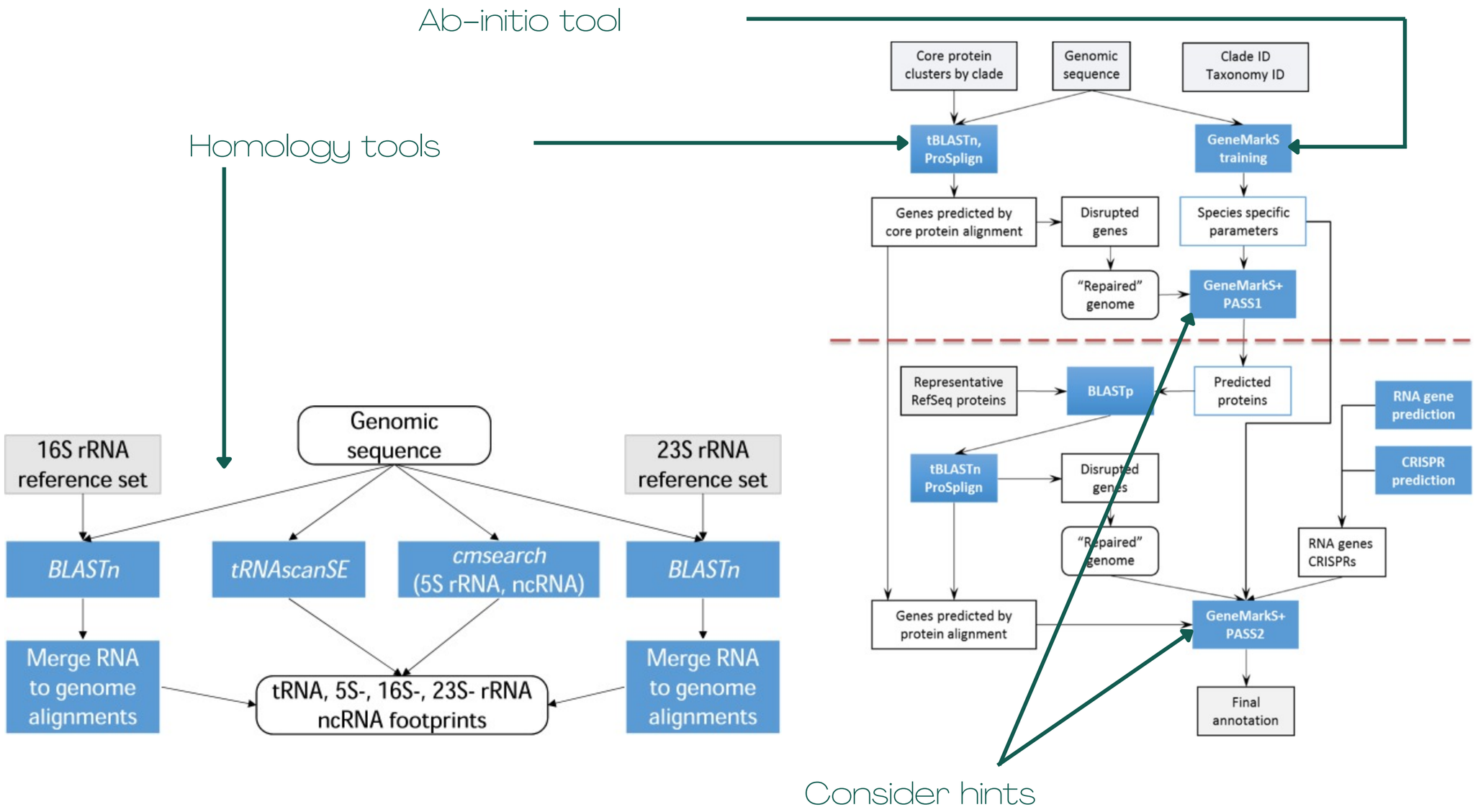
PGAP

- Prokaryotic genome annotation pipeline
- Widely used (“standard”?)
- More computationally intensive, improved accuracy
- Developed as collaboration between NCBI and Georgia Tech





PGAP


Ab-initio tool
Homology tools
Core protein clusters by clade
Genomic sequence
Clade ID Taxonomy ID
tBLASTn, ProSplign
GeneMarkS training
Genes predicted by core protein alignment
Disrupted genes
Species specific parameters
"Repaired" genome
GeneMarkS+ PASS1
Representative RefSeq proteins
BLASTp
Predicted proteins
RNA gene prediction
CRISPR prediction
tBLASTn ProSplign
Disrupted genes
"Repaired" genome
RNA genes CRISPRs
Genes predicted by protein alignment
GeneMarkS+ PASS2
Final annotation
Genomic sequence
16S rRNA reference set
23S rRNA reference set
BLASTn
tRNAscanSE
cmsearch (5S rRNA, ncRNA)
BLASTn
Merge RNA to genome alignments
tRNA, 5S-, 16S-, 23S- rRNA ncRNA footprints
Merge RNA to genome alignments
Consider hints

PROPOSED PREDICTION PIPELINE

- Gene prediction performed in parallel with:
  - GeneMarkS-2, Prodigal, MetaGeneAnnotator
- Outputs of each evaluated using ORForise & CEA of the organism
  - CEAs retrieved from Ensembl Bacteria
  - Rare organisms may not have a good CEA
- Coding Sequence predictions from algorithm with best ORForise score will be passed downstream to annotation pipeline
  - If we implement a pre-built pipeline, GeneMarkS-2 performing best would favor use of PGAP, Prodigal performing best would favor use of PROKKA
  - MetaGeneAnnotator is not part of a pre-built pipeline, and will require a more bespoke pipeline

REFERENCES

[1] Wang Z, Chen Y, Li Y. A brief review of computational gene prediction methods. Genomics Proteomics Bioinformatics. 2004;2(4):216–221. doi:10.1016/s1672-0229(04)02028-5

[2] Tataru P, Sand A, Hobolth A, Mailund T, Pedersen CN. Algorithms for hidden markov models restricted to occurrences of regular expressions. Biology (Basel). 2013;2(4):1282–1295. Published 2013 Nov 8. doi:10.3390/biology2041282

[3] Lomsadze A, Gemayel K, Tang S, Borodovsky M. Modeling leaderless transcription and atypical genes results in more accurate gene prediction in prokaryotes. Genome Res. 2018;28(7):1079–1089. doi:10.1101/gr.230615.117

REFERENCES

[4] Arthur L. Delcher, Kirsten A. Bratke, Edwin C. Powers, Steven L. Salzberg, Identifying bacterial genes and endosymbiont DNA with Glimmer, Bioinformatics, Volume 23, Issue 6, march 2007, Pages 673–679, https://doi.org/10.1093/bioinformatics/btm009

[5] Belliardo, C., Koutsovoulos, G.D., Rancurel, C. et al. Improvement of eukaryotic protein predictions from soil metagenomes. Sci Data 9, 311 (2022). https://doi.org/10.1038/s41597-022-01420-4

[6] Korandla DR, Wozniak JM, Campeau A, Gonzalez DJ, Wright ES. AssessORF: combining evolutionary conservation and proteomics to assess prokaryotic gene predictions. Bioinformatics. 2020;36(4):1022-1029. doi:10.1093/bioinformatics/btz714

REFERENCES

[7] Nicholas J Dimonaco, Wayne Aubrey, Kim Kenobi, Amanda Clare, Christopher J Creevey, No one tool to rule them all: prokaryotic gene prediction tool annotations are highly dependent on the organism of study, Bioinformatics, Volume 38, Issue 5, March 2022, Pages 1198–1207, https://doi.org/10.1093/bioinformatics/btab827

[8] Tatusova T, DiCuccio M, Badretdin A, Chetvernin V, Nawrocki EP, Zaslavsky L, Lomsadze A, Pruitt KD, Borodovsky M, Ostell J. NCBI prokaryotic genome annotation pipeline. Nucleic Acids Res. 2016 Aug 19;44(14):6614-24. doi: 10.1093/nar/gkw569. Epub 2016 Jun 24. PMID: 27342282; PMCID: PMC5001611.

[9] Torsten Seemann, Prokka: rapid prokaryotic genome annotation, Bioinformatics, Volume 30, Issue 14, July 2014, Pages 2068–2069, https://doi.org/10.1093/bioinformatics/btu153