Data and Methodology

1.1 Methodology Overview

The general processing steps performed in this work can be grouped into four processing stages, or workflows. A simplified overview of these stages and the flow of data between them is shown in Figure 1.1. In ascending order, each processing stage is discussed in more detail in Sections 1.2, 1.4, 1.9, and **??**.

1.2 Sequence Processing and Assembly Workflow

The sequence processing and assembly workflow is shown in Figure 1.2. Processing steps are represented by boxes, results of interest generated by this work are represented by folders, and datasets that were not produced by this work are represented by cylinders. Other workflows that use datasets produced by this workflow are depicted as arrows outside of the brown sub-graph.

Illumina©™[**?**] sequences were first trimmed and filtered using Trimmomatic[**?**] to remove adapter content and low-quality sequences, as low-quality sequences may affect the assembly process, resulting in fragmented and erroneous assemblies. The processed Illumina©™sequences and Nanopore©™[**?**] sequences were then supplied to the hybrid genome assembly tool NextDenovo[**?**]. As Ilumina sequences are of extremely high accuracy, especially after trimming, the processed Illumina©™data was then re-used to polish the assemblies with NextPolish[**?**]. Polishing is used because long-read sequences used in the assembly process are less accurate, at approximately 90-95%, than Illumina sequences at 99.9%, so the Illumina data can be used to correct any assembled sequences that may contain base-call errors. The resulting assemblies were then analyzed to identify AT-rich genomic sequence, analyzed with QUAST for general assembly metrics, and then used as input in the gene prediction workflow.

The processing steps Trimmomatic filtering, NextDenovo assembly, NextPolish Polishing and QUAST assembly assessment are described in Section 1.3. The AT-rich sequence identification process is described later in Section 1.15.

1.3 Sequence Pre-processing and Assembly

Genomic sequences used for assembly were generated for DC1 and Tsth20 using Nanopore©™[**?**] and Illumina©™[**?**] sequencing technologies by Brendan Ashby at the Global Institute for Food Security at the University of Saskatchewan. Short-read sequences were first examined with the FastQC tool[**?**] to evalute sequencing quality and general metrics associated with the sequencing run. FastQC was run with default parameters. The short-read sequences were then trimmed to remove any adapter content or low quality sequences as they may lead to fragmented assemblies and erroneous base calls. Illumina©™sequencing data

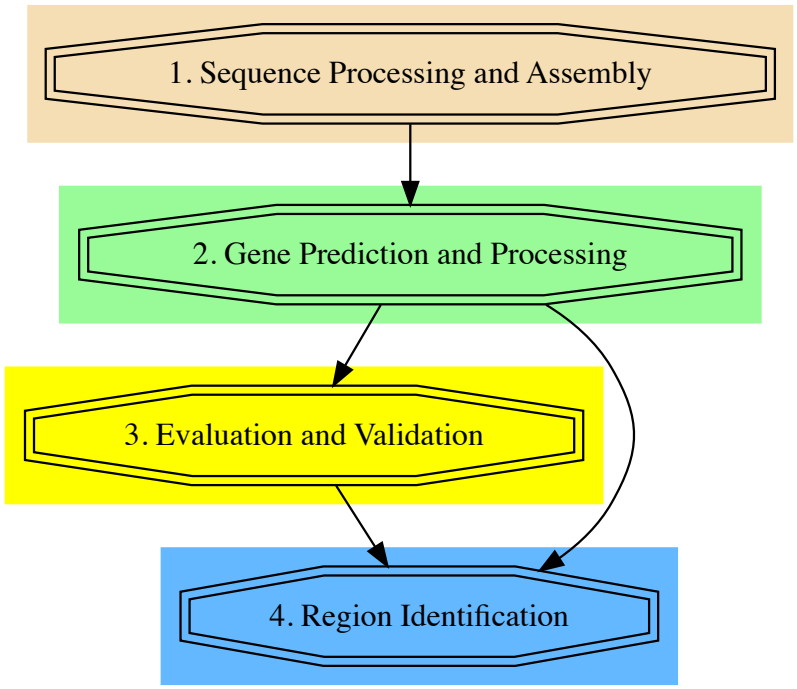


Figure 1.1: A simplified workflow used in this work. Each processing stage is is represented by an octagon, and the flow of data are indicated by directed edges between them. In this figure and subsequent figures, the sequence processing stage is coloured brown, the gene prediction and processing stage is coloured green, the evaluation and validation stage is coloured yellow, and the region identification process is coloured blue.

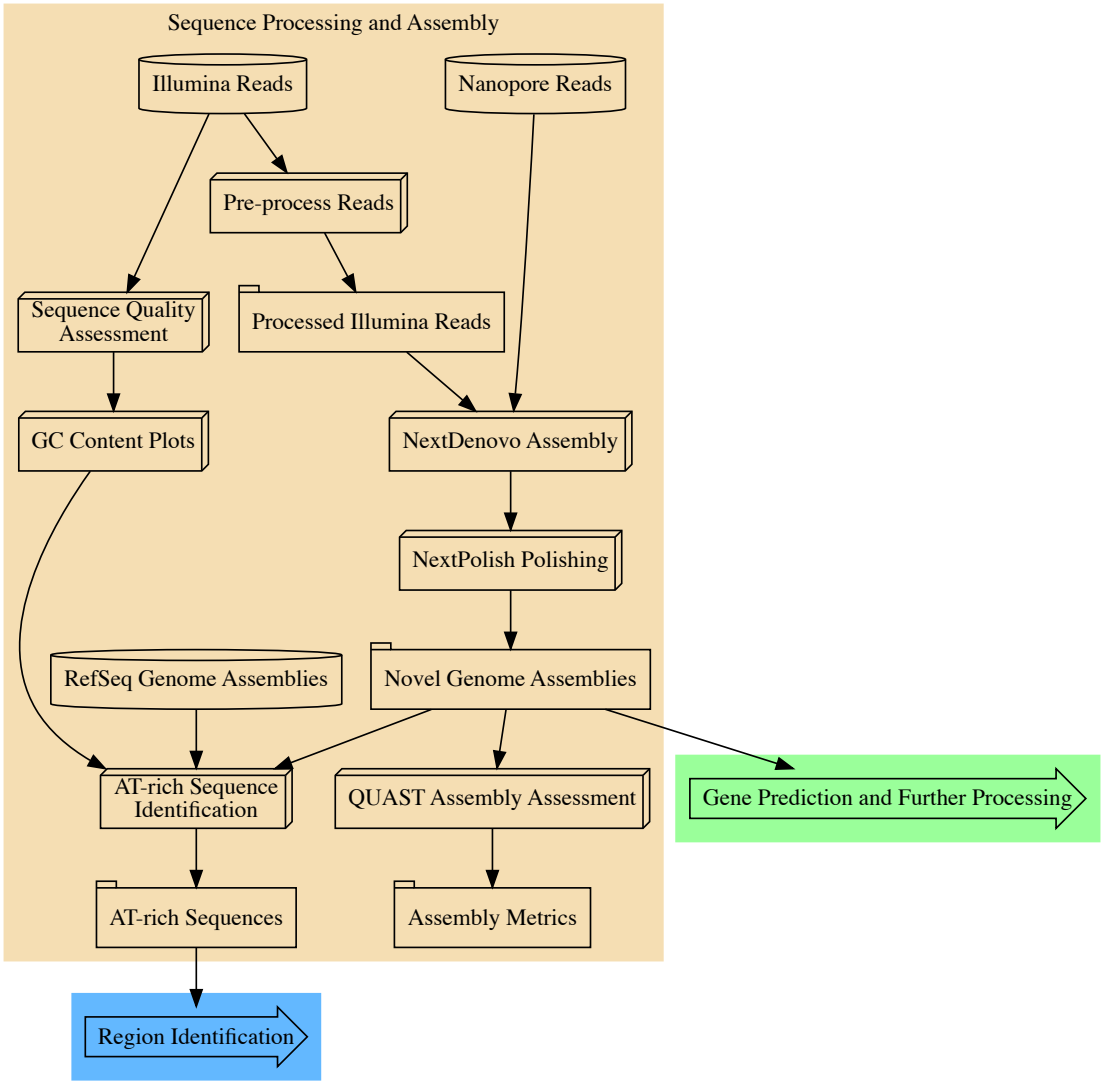


Figure 1.2: A workflow (in brown) depicting the steps taken to process input sequences, generate assemblies, and post-process those assemblies for use in other workflows. Processing steps are represented by boxes, datasets of interest that were produced by this work are represented by folders, and datasets that were not produced in this work are represented by cylinders. Other workflows are represented by green and blue arrows outside of the brown graph. Directed edges indicate the flow of data.

was filtered using Trimmomatic v0.38[**?**] with filtering criteria as follows: SLIDINGWINDOW:4:28, LEADING:28, TRAILING:28, MINLEN:75. Only surviving paired-end reads were used for further analysis. The Nanopore©™data were not processed prior to assembly.

It has been shown that assemblies that use both short-read and long-read sequencing data during the assembly are of higher quality and less fragmented than assemblies using only one form of sequencing. Genomic Nanopore©™and Illumina©™sequences from DC1 and Tsth20 were assembled in to contigs using the hybrid assembly tool NextDenovo©™[**?**] v2.5.0 with default parameters. Although hybrid genome assemblies are of higher quality, they may still contain base calling errors after assembly. The highly accurate short-reads can be used to correct base calling errors after assembly. To do this, the hybrid assemblies were polished with Illumina©™sequences using NextPolish[**?**] v1.4.1 and default parameters. Final assembly metrics were calculated using QUAST v5.0.2[**?**] with default parameters.

1.4 Overview of Gene Prediction and Further Processing

To better understand the methods used in the gene prediction and CDS analysis workflow, the steps have been grouped together in Figure 1.3. Processing steps are represented by boxes, results of interest generated by this work are represented by folders, and datasets that were not produced in this work are represented by cylinders. Other workflows that use datasets produced by this workflow are depicted as arrows outside of the green sub-graph.

First, RefSeq assemblies and their associated annotations of closely related *Trichoderma* species were selected from NCBI for training and comparison. Both and *ab initio* (GeneMark[**?**]) and evidence-based (Braker2[**?**]) gene finders were selected for comparison to evaluate gene prediction behaviour of both methods in the context of *Trichoderma* genomes. Gene prediction was performed using both gene finders on the RefSeq genomes and the genomes of DC1 and Tsth20. Coding sequences (CDS) output by the gene finders were then examined to better understand distributions of CDS lengths produced by different gene finders. Finally, all gene predictions were then supplied to the evaluation and validation stage as well as the region identification stage.

Selection of NCBI datasets for training and comparison are further discussed in Section 1.5. The GeneMark gene prediction process is described in Section 1.6, while the Braker2 gene prediction process is described in Section 1.7. The CDS processing step is described in Section 1.8.

1.5 Selection of Datasets from NCBI

When evaluating gene-finders, it is important compare gene predictions to a standard, or control dataset, as a method of ground-truthing. Such datasets include reference genome sequences and their associated gene predictions from NCBI as used in this work. In addition to comparing against control datasets, RNAseq data can also be used as training data for various tools as well, such as the evidence-based training used by Braker2.

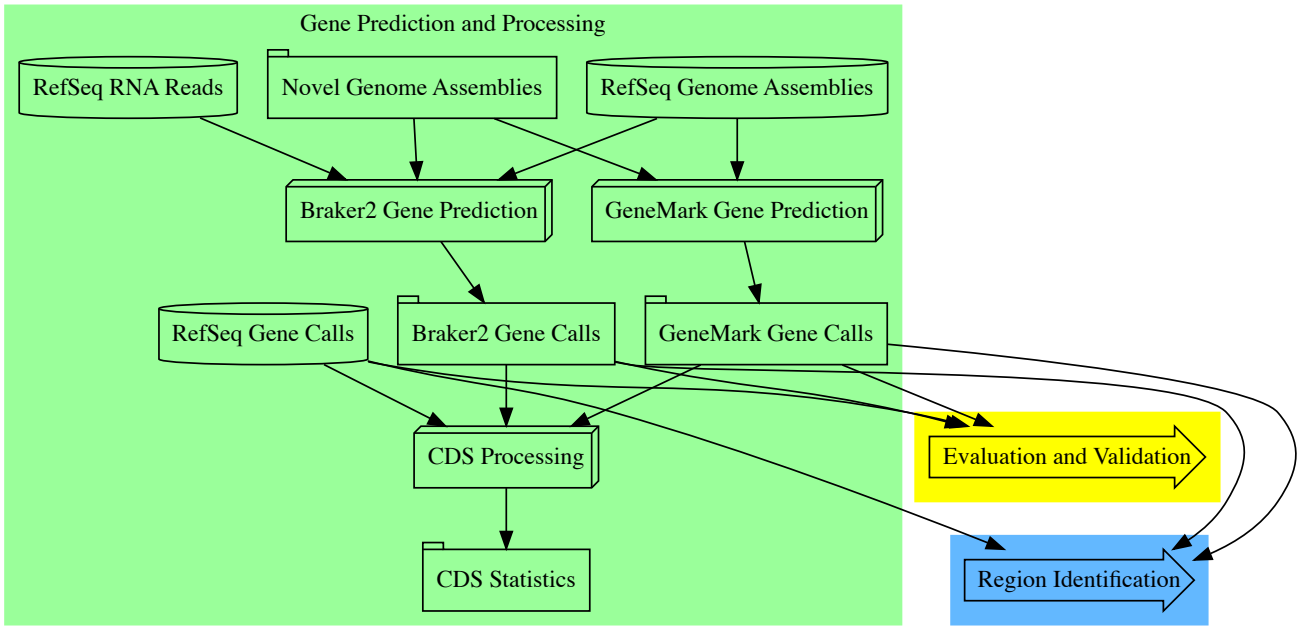


Figure 1.3: A workflow (green) depicting the steps taken to predict genes for use in other workflows and process CDS sequence lengths. Processing steps are represented by boxes, datasets of interest that were produced by this work are represented by folders, and datasets that were not produced in this work are represented by cylinders. Other workflows are represented by blue and yellow arrows outside of the green graph. Directed edges indicate the flow of data.

In this work, four forms of these inputs were selected from NCBI for comparison and training. The first two datasets selected were assemblies and associated gene predictions of three RefSeq *Trichoderma* accessions, with those accessions being *T. virens*, *T. harzianum*, and *T. virens*. NCBI RefSeq accession IDs for those assemblies are GCF_000167675.1_v2.0, GCF_003025095.1_Triha_v1.0, and GCF_000170995.1_TRIVI_v2.0, respectively. *T. reesei* was selected as it is the most well-studied *Trichoderma* species, and widely considered to be the gold-standard for *Trichoderma* genomes. *T. harzianum* and *T. virens* were selected as additional datasets due to their prevalence in literature as well.

The second set of selected inputs were RNAseq datasets from *T. reesei*, used as input in the training process for Braker2. The use of training data in gene finding applications is to produce a predictive model that factors in experimental evidence for a 'tailored' gene prediction model. These models can predict genes with more confidence and predict the correct gene structure more often[**?**]. The SRA accession IDs for sequences used are: SRR5229930, SRR8329344, SRR8329345, SRR8329346, and SRR8329347. Limited RNAseq datasets were used as there were few RNAseq datasets available at the beginning of this project.

The final set of inputs are protein sequences from RefSeq individuals *T. atroviride*, *Fusarium graminarium*, and *Saccharomyces cerevisiae*, which were for downstream validation of gene predictions. The NCBI accession IDs for these assemblies and associated protein sequences are GCF_000171015.1, GCF_000240135.3, and GCF_000146045.2, respectively.

1.6 Gene Prediction Using GeneMark-ES

Ab initio gene finding was performed using GeneMark-ES v4.71[**?**] with DC1, Tsth20 and selected RefSeq *Trichoderma* assemblies used as input. Default parameters were used in all cases except for the fungal option, which allowed the use of a GeneMark model specific to fungal genomes.

1.7 Gene Prediction Using Braker2

For evidence-based gene finding, Braker2 v3.0.2[**?**] was used with *Trichoderma reesei* selected as the reference organism for training. Illumina©™[**?**] RNAseq datasets from *T. reesei* were downloaded from the NCBI short-read archive using the sra-toolkit[**?**] and were not trimmed or filtered prior to their use in Braker2 training. Default parameters were used to train a Braker2 model on *Trichoderma reesei* data except in the case of the fungal option, which was used for this analysis for improved gene-finding performance in fungi. The trained Braker2 *T. reesei* prediction model was then applied to all assemblies with default parameters.

Braker2 also provides a UTR option to predict upstream sequence features, but that option is experimental and was left off for this work.

1.8 Statistical Analysis of CDS Lengths

The distribution of gene lengths predicted by a gene-finder is an important feature to consider when selecting a gene finding tool, as genes of a particular length may be of interest to researchers, but more importantly, gene-finders may be biased to certain distributions based on their underlying models. This is important as the distribution of predicted gene lengths may also play a role in *Trichoderma's* role as an avirulent plant symbiont. These biased distributions may differ from the true distribution of CDS lengths leading to propagation of sub-optimal gene-finding results and results that may not reflect the true distribution of gene lengths. To determine, if differences exist between distributions of predicted gene lengths from each tool, the $\log_{10}$ lengths of coding sequences (CDS) were subject to a two-sample Kolmogorov-Smirnov test[**?**].

1.9 Overview of Evaluation and Validation of Gene Predictions

Following the gene prediction process, prediction results can then be evaluated and queried against exiting datasets as a form of validation. The workflow for evaluation and validation of gene predictions is shown in Figure 1.4. Processing steps are represented by boxes, results of interest generated by this work are represented by folders, and datasets that were not produced in this work are represented by cylinders. Other workflows that use datasets produced by this workflow are depicted as arrows outside of the yellow sub-graph. The flow of data is represented by directed edges.

Coding sequences from the previous stage were evaluated and validated using several tools. Firstly, protein products from predicted CDSs were processed with InterProScan to identify Pfam domains as evidence for the validity of gene predictions. Secondly, CDSs were analyzed with BUSCO to assess prediction of conserved fungal genes. Finally, RefSeq proteins from related fungal species were used in tblastn searches of the five *Trichoderma* assemblies selected for the main body of this work.

BUSCO evaluation, InterProScan Annotation, and BLAST alignments processes are described in more detail in Sections 1.10, 1.11, and 1.12, respectively. Results from these processes are also used as inputs in the region identification workflow later.

1.10 Evaluating Gene-Finder Performance Using BUSCO

While novel gene predictions are of great interest in new subjects, it is also important to confirm that gene-finders are predicting genes that are expected to be present in the organism of interest[**?**]. To determine if this is true, BUSCO v5.7.1[**?**] was selected as a tool to determine a gene-finder's ability to predict a set of conserved orthologous genes. The BUSCO Metaeuk pipeline was run using the BUSCO fungi_Odb10 fungal lineage dataset to capture highly conserved genes in fungal genomes. This pipeline was applied to all genome assemblies used in this work.

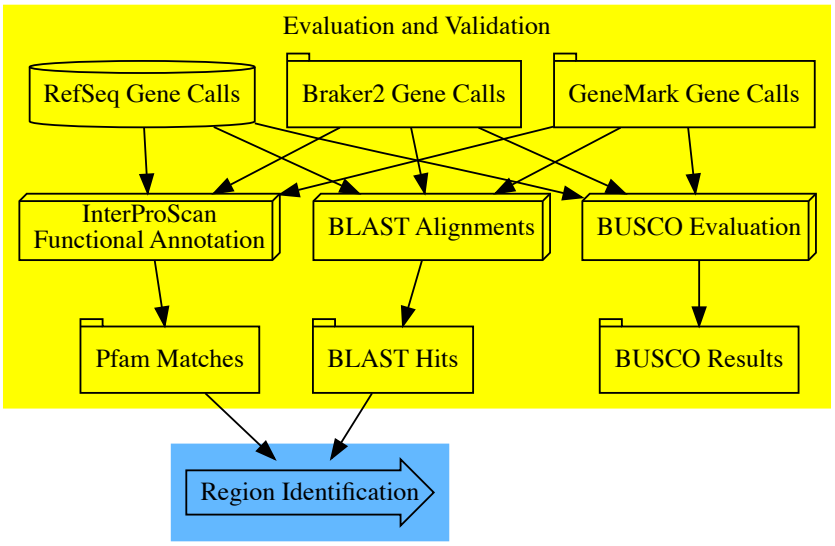


Figure 1.4: A workflow (yellow) depicting the steps taken to evaluate and validate predicted genes. Processing steps are represented by boxes, datasets of interest that were produced by this work are represented by folders, and datasets that were not produced in this work are represented by cylinders. Other workflows using results from these processing steps are represented by a blue arrow outside of the yellow graph. Directed edges indicate the flow of data.

1.11 Functional Annotation Using InterProScan

The presence of a functional component in a predicted gene's protein product is good evidence in favour of a gene-finder's performance. To identify functional components, the protein products from each gene finder's predictions were functionally annotated using InterProScan v5.65-97.0[**?**] without the match lookup service and with the following analyses included: AntiFam-7.0, CDD-3.20, Coils-2.2.1, FunFam-4.3.0, Gene3D-4.3.0, Hamap-2023_01, MobiDBLite-2.0, NCBIfam-13.0, PANTHER-18.0, Pfam-36.0, PIRSF-3.10, PIRSR-2023_05, PRINTS-42.0, ProSitePatterns-2022_05, ProSiteProfiles-2022_05, SFLD-4, SMART-9.0, SUPERFAMILY-1.75. InterProScan was run on all predicted proteins using default parameters. The resulting protein annotations were mapped back to their genome assemblies and processed using the region identification algorithm in section 1.14.

1.12 Ground-truthing with BLAST

The *Trichoderma* assemblies chosen for comparison were used as the subject sequences in tblastn searches with the query proteins discussed in paragraph three of Section 1.5. Default parameters were used in the tblastn searches. Hits from the tblastn alignments were then filtered for alignments with a minimum of 30% query coverage and 30% identity. These criteria were selected as they filter out spurious alignments while also retaining short alignments that capture functional or conserved protein coding sequence. Remaining alignments were then processed using the region identification algorithm in 1.14

1.13 Region Identification Overview

A major portion of this work was the process of identifying regions of overlapping features as a method of comparing gene calls from different assemblers. The definition of a region and the algorithm used to identify them is described in detail in Section 1.14, and an overview of the region identification workflow is shown in Figure 1.5. The top of the graph shows the gene calls produced by the workflow described in Section 1.4, which are used as the main input to the region identification process along with other previously generated datasets. These additional datasets may contain other information such as blast alignments or functional annotations and can be included as features in the region identification process. Datasets outlined with rectangles indicate that the data within were processed in the context of the main gene calls and are discussed in their own results sections. Datasets output by the region identification process were then visualized with the Integrative Genomics Viewer, described in Section 1.16

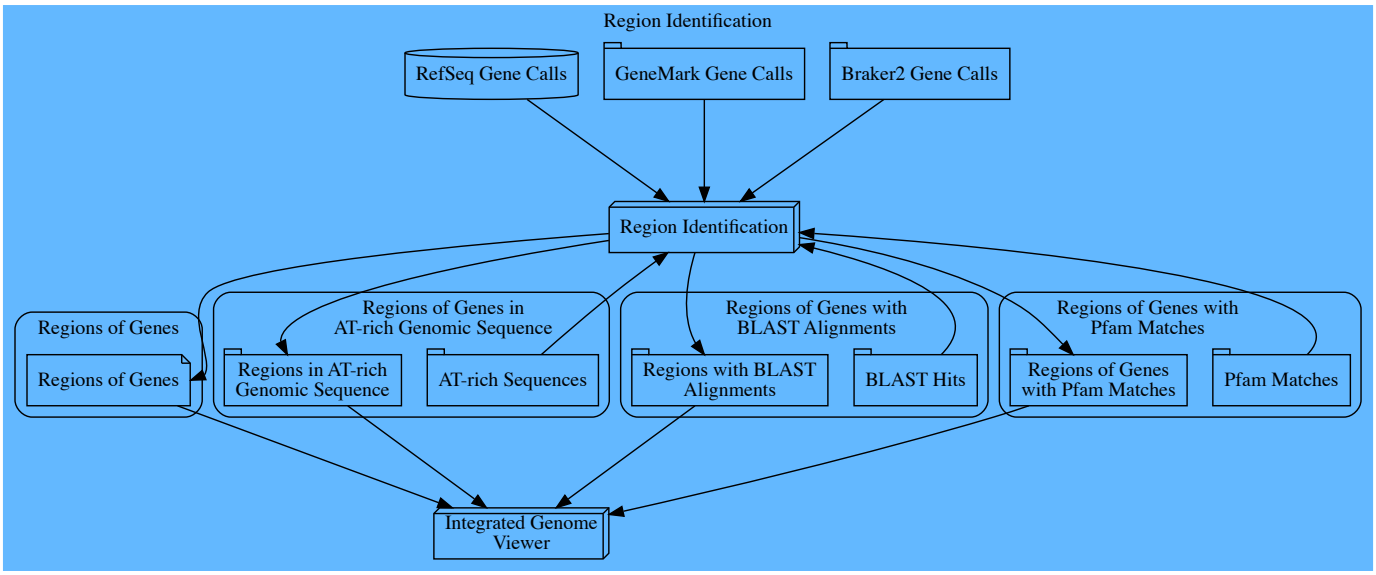


Figure 1.5: An overview of the region identification process for different datasets. Sections of the graph are outlined with rectangles, indicating that the datasets within were processed along with gene calls and discussed in their own results sections. Processing steps are represented by boxes, datasets of interest that were produced by this work are represented by folders, and datasets that were not produced by this work are represented by cylinders. Directed edges indicate the flow of data.

1.14 Region Identification Algorithm

Predictions and results from 1.6 and 1.7 were processed into regions of overlapping gene predictions and annotations. A region is defined as a pair of genomic coordinates containing one start and one stop coordinate, that contains one or more overlapping features. We define a feature as a pair genomic coordinates containing one start and one stop coordinate that associated with a gene prediction, annotation, or other data point of interest.

Overlapping features were processed into regions using a concept similar to Allen's interval algebra[**?**]. The algorithm first sorts all features on each contig by start coordinate and then iterates over each feature, identifying overlaps with previous features to identify regions of overlapping features. We approached this as a graph problem, where features make up a set of nodes V, and spatial overlaps between features make up a set of edges E, forming the graph $G = \{V,\ E\}$. This algorithm performs a transitive closure R^+ of G, resulting in sub-graphs, or regions of features in which each node has a path to each other node in that sub-graph. Pseudo-code for the algorithm is shown in Algorithm 1. This processing was performed with Python v3.12.4[**?**] and the GFF package from BCBio[**?**] for easy GFF parsing. The resulting regions were then further processed to identify trends and behaviours of gene finders in the context of other annotated information and features.

1.15 Gene predictions in AT-rich Genomic Sequence

Initial exploration of the Illumina sequence data generated for DC1 and Tsth20 showed bi-modal distributions of GC content in the sequences. An example of the GC distribution of Illumina sequences for DC1 is shown in Figure 1.6 compared to a theoretical normal distribution. To differentiate AT-rich sequences from normal sequences, a cutoff value was selected to isolate AT-rich sequences from the rest. The cutoff value was chosen as 28% GC content, which is roughly the point where the AT-rich sequence counts begin increasing in all assemblies. The details of this process are explained below.

AT-rich genomic sequences were identified using sliding windows of GC content over each *Trichoderma* assembly. Sliding windows were generated using the isochore program from the EMBOSS tool suite v6.6.0[**?**]. The sliding window used was 250bp wide with a 50bp shift. A window was classified as AT-rich if the sequence contained lower than 28% GC content. The genomic sequences from AT-rich windows were mapped to a GFF file and processed using the region identification algorithm described in section 1.14.

1.16 Visualization of Identified Regions

Results from the region identification process were visualized using the IGV genome browser v2.19.1[**?**]. An example of an IGV screenshot is shown in Figure 1.7

Algorithm 1 The general algorithm underlying the region identification process.

INPUT: list of GFF files, $GFFList$

RETURNS: list of regions, $regionList$

for `<each GFF in GFFList>` **do**

  $allFeatures \leftarrow GFF.features$

end for

$sortedFeatures \leftarrow sortStartCoord(allFeatures)$

$regionList \leftarrow []$

$firstFeature \leftarrow firstItem(sortedFeatures)$

$lastFeature \leftarrow lastItem(sortedFeatures)$

for `<each feature in sortedFeatures>` **do**

  if $feature = firstFeature$ **then**

    $currentRegion \leftarrow newRegion(feature)$

    $currentRegion.start \leftarrow feature.start$

    $currentRegion.end \leftarrow feature.end$

  else if $feature = lastFeature$ **then**

    $regionList.append(currentRegion)$

    $return\ regionList$

  else if $feature\ overlaps\ currentRegion$ **then**

    $currentRegion.updatePositions(currentRegion, feature)$

    $currentRegion.addFeature(feature)$

  else

    $regionList.append(currentRegion)$

    $currentRegion \leftarrow newRegion(feature)$

    $currentRegion \leftarrow feature.start$

    $currentRegion \leftarrow feature.end$

  end if

end for

$return\ regionList$

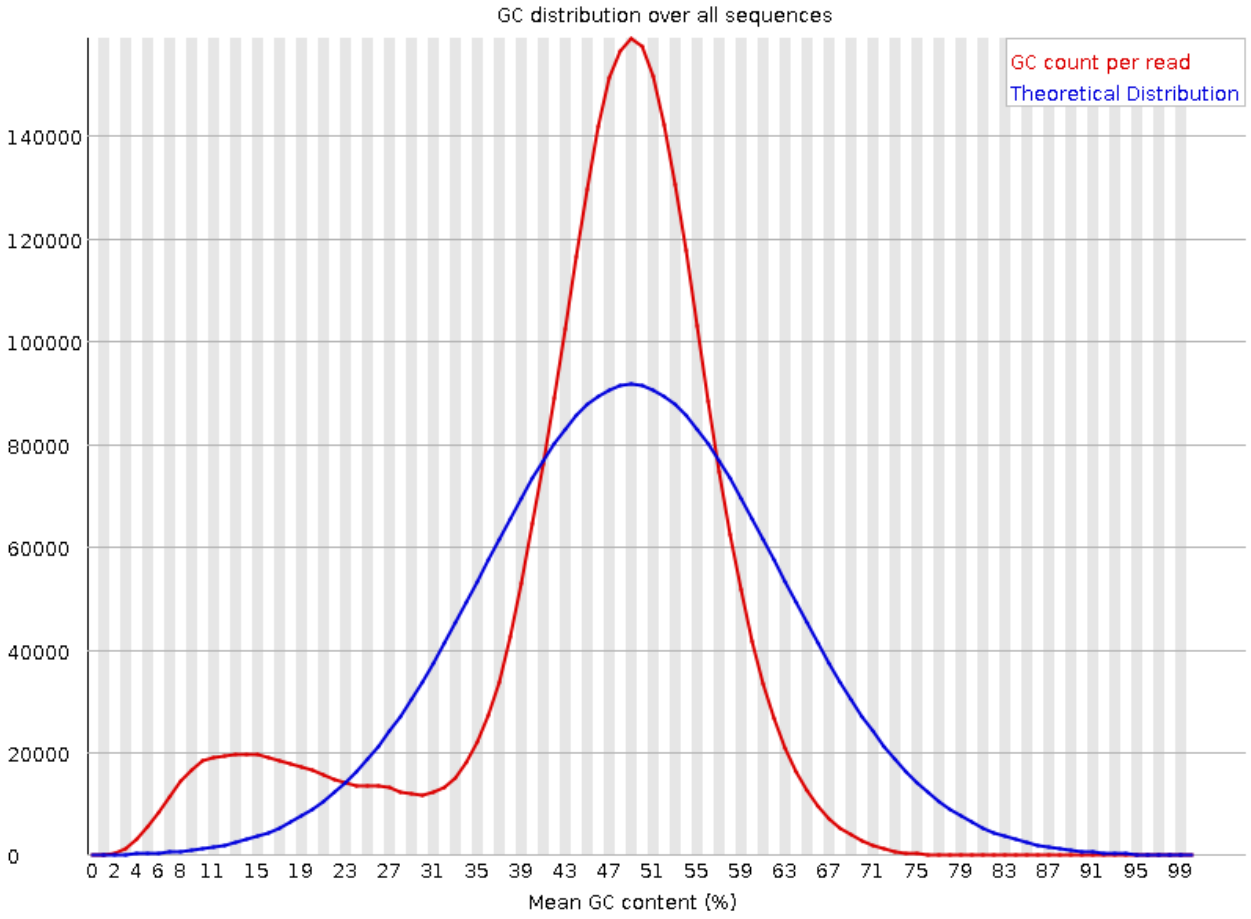


Figure 1.6: A plot of GC content of Illumina sequences generated for DC1. The X-axis indicates mean GC content of sequences while the Y-axis indicates the total number of sequences for a given mean GC content. A theoretical normal distribution is overlaid in blue.

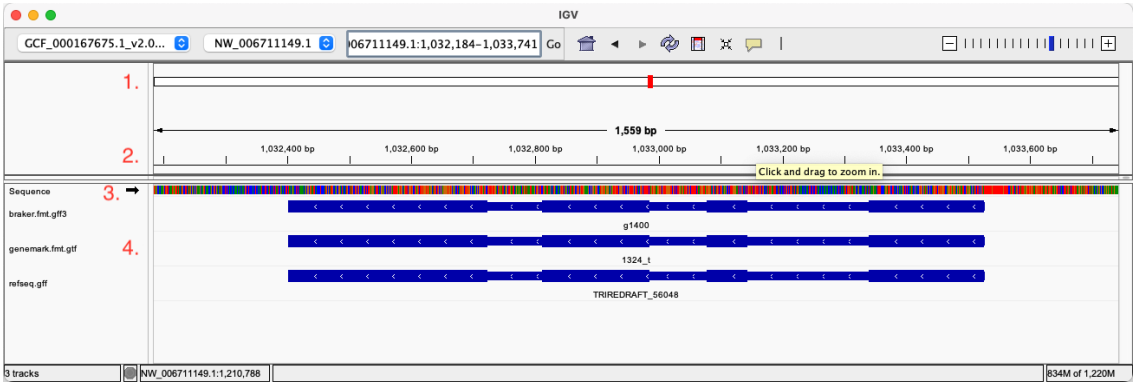


Figure 1.7: An example of an IGV screenshot used in portions of the results from this work. The top half of the image displays information about the reference sequence with track one showing the position of the viewer relative to the reference sequence, and track two showing the numerical positions of the reference. The lower half of the image displays more tracks in the form of additional layers of information mapped to the reference sequence. Track three shows a track displaying the nucleotide at each position if the scale of the viewer is small enough. Track four comprises several tracks, each containing features, the regions in blue, from a GFF file. The tracks in this example are limited to gene predictions, although tracks in other IGV screenshots may include features from other tools such as InterProScan and will be discussed in their respective sections. Individual tracks will be referred to in increasing numerical order in the track list beginning from the top. The nucleotide track is not included in the order.

¿¿

Results and Discussion

2.1 Overview

Results from this work are ordered in a particular manner relative to the order in which data was processed. Since there are novel *Trichoderma* assemblies included in this work, the results begin with an overview of the assemblies of DC1 and Tsth20 in Section 2.2. Following this, is Section 2.3, which discusses the installation, application and other features associated with the gene finders selected for this work. Results of the initial gene finding process are then presented in Section 2.4 followed by more detailed analysis of predicted CDS lengths in Section 2.5. Gene predictions from each gene finder and assembly combination are supplied to the BUSCO metric process in Section 2.6. Results from the initial spatial comparison of gene predictions using the region identification process are then presented in Section 2.7. The region identification process is then applied with additional information from tblastn alignments, InterProScan annotation, and GC sequence composition in Sections 2.8, 2.9 and 2.10, respectively. A final comparison of all results and selection of an appropriate gene finder are then discussed in the Conclusion 3.1.

RefSeq assemblies in these results are referred to by their scientific name for context and not by their associated NCBI accession IDs. Instead, NCBI accessions for the RefSeq assemblies and their associated annotations are listed here: *T. reesei* - GCF_000167675.1_v2.0, *T. harzianum* - GCF_003025095.1_Triha_v1.0, GCF_000170995.1_TRIVI_v2.0.

2.2 Assemblies of DC1 and Tsth20

For general assembly metrics of DC1 and Tsth20, the QUAST[**?**] tool was used. Results from QUAST are shown in Table 2.1, from which we can make several observations. In DC1 and Tsth20, the total contig counts are an order of magnitude smaller when compared to the other NCBI RefSeq assemblies, indicating highly contiguous assemblies from nextDenovo[**?**] and nextPolish[**?**]. This is likely due to the use of Nanopore sequencing in the assemblies of DC1 and Tsth20. The total assembled lengths of DC1 and Tsth20 are similar when compared to RefSeq assemblies, ranging from 38Mb to 42Mb, except in the case of *T. reesei*, which is known to have a significantly smaller genome length[**?**], at roughly 33Mb.

The largest contig size for each assembly varies greatly. DC1 and Tsth20 have the largest contigs of all assemblies being considered, which is again likely due to the inclusion of long-read sequencing data in the assembly process. The N50 values for all assemblies are above 1Mb, with DC1 and Tsth20 N50s being at minimum two times larger than others assemblies. N50 lengths nearing chromosomal lengths indicates that assemblies of DC1 and Tsth20 are better assembled than the other *Trichoderma* assemblies. While chromosome-scale contigs are not the sole indicator of genome quality, it does provide confidence in the quality of the input data and resulting assemblies. In general, the assemblies of DC1 and Tsth20 are of

Strain	Total Contigs	Total Length	Largest Contig	GC%	N50	L50
DC1	8	38.6 Mb	11.49 Mb	47.97	5.69 Mb	3
Tsth20	7	41.58 Mb	8.02 Mb	47.33	6.52 Mb	3
T. harzianum	532	40.98 Mb	4.08 Mb	47.61	2.41 Mb	7
T. virens	93	39.02 Mb	3.45 Mb	49.25	1.83 Mb	8
T. reesei	77	33.39 Mb	3.75 Mb	52.82	1.21 Mb	9

Table 2.1: General assembly metrics produced by QUAST[**?**] (a genome quality assessment tool).

similar length to existing *Trichoderma* assemblies and the number of contigs reported match the number of 'chromosome' scale contigs reported in other work[**?**].

During initial investigation of the inputs to the assembly process, we observed that the Illumina reads have a bimodal distribution of GC content as shown in Figure2.1. To see if this observation extended to the assemblies as well, 250 bp sliding windows were used to calculate GC content for all assemblies included in this analysis. The results of this analysis are shown in Figure 2.1. AT-rich sequences are visualized on the left peak of the distributions sequences containing 90 percent AT content. Of the included assemblies, AT-rich sequences were identified in DC1, Tsth20, *T. reesei* and *T. harzianum*, with *T. virens* deviating from the other assemblies showing very few AT-rich windows. This lack of AT-rich sequence in *T. virens* nucleotide composition may indicate potential assembly issues or even misclassification of the organism. In addition to the confirmation of increased AT-rich sequence content in most assemblies, it appears that the distribution of GC content in *T. reesei* differs from the other assemblies. The curve of GC content for *T. reesei*, visualized in green in Figure 2.1, is shifted to the right of the other *Trichoderma* assemblies, indicating fewer AT-rich windows and more balanced nucleotide composition in its assembly. While the left tail of the curve also shows an increase in AT-rich sequence composition, with it's peak is again located farther right on the X-axis than other *Trichoderma* assemblies. We can not explain exactly why *T. reesei*'s curve differs so much at this time, but it is likely due to the much smaller *T. reesei* assembly and it's properties. Investigation of these AT-rich sequences is continued in Section 2.10, where only sequences containing less than greater than 72% AT nucleotide content are considered, as the distributions begin to deviate from the normal distribution at that point.

2.3 Installation and Profiling Gene Finding Tools

While Braker2, GeneMark and RefSeq all predict coding sequences of possible genes for a provided input, the implementation of each tool is different, requiring more or less effort to install and run than other tools. This section discusses the implementations, installation procedures and execution of GeneMark and Braker2 in more detail. For more information on the versions and parameters used in this processing, please see Section

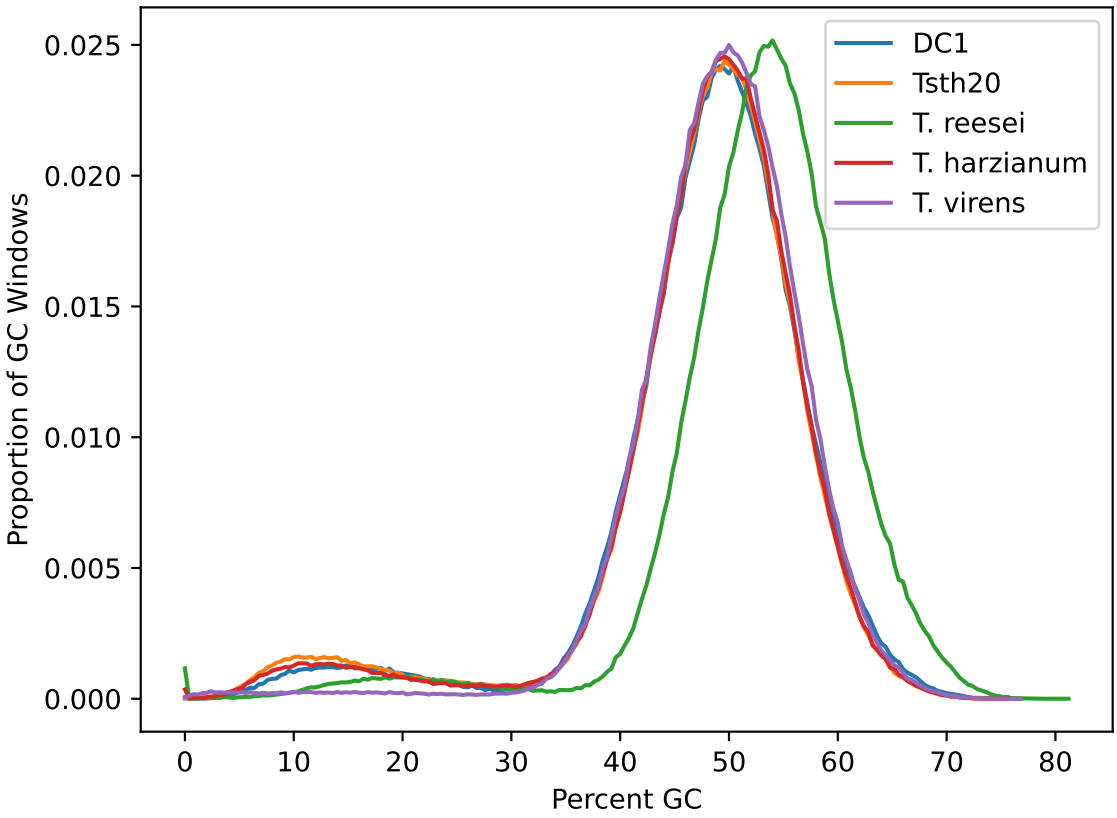


Figure 2.1: Plots showing proportions of sliding windows of GC content for each *Trichoderma* genome assembly.

1.7. The computing platform used in this research is hosted and managed by University of Saskatchewan services, which is modelled around the HPC platform used by the Digital Research Alliance of Canada and software is managed similarly.

GeneMark[**?**] is a gene finding tool developed by the Georgia Institute of Technology with packages prepared for Linux and MacOS. It is provided as licensed product in the form of a package which can be downloaded from their website after submitting a form. Once the necessary information is submitted, the user is provided with a key that must be placed in the appropriate location once the software is downloaded and unpacked. The core controlling methods of GeneMark are written in Perl, accompanied by several Python scripts and compiled executables. GeneMark was tested by the developers with Perl version 5.10, and Python 3.3. A number of Perl dependencies are also required, which can be installed via CPAN. The user will have to know which implementation of GeneMark they are wanting to use for their application, as GeneMark has several variations it can run depending on the desired application. In this work, the GeneMark-ES variant of GeneMark was executed, as it is the self-training *ab initio* GeneMark method for eukaryotic organisms. Options required by GeneMark at runtime are documented in the help message, and simple enough that any user with familiarity of bioinformatics tools should be able to run GeneMark, although documentation for use is only provided by the help message when running the program and not online. In regards to run-time, running the GeneMark-ES pipeline on DC1 with 56 threads finished in 16 minutes. GeneMark's outputs consist of a GTF or GFF file of predicted genes as well as a number of other outputs related to the run.

Braker2[**?**] is hosted on GitHub as a repository that receives relatively frequent updates with Braker3 being released while working before the completion of this work. Braker is maintained by Katharina Hoff from the University of Greifswald and is available under the Open Source Artistic License. Installation of the repository is a straightforward pull from GitHub. As with GeneMark, Braker2 uses a combination of Perl, Python and other exectuables in its regular use. Downloading the repository itself is not enough for execution, as Braker2 relies on a number of dependencies and bioinformatics tools including Perl and (Perl dependencies), Augustus[**?**], BamTools[**?**], BedTools[**?**], GeneMark[**?**], StringTie[**?**], GFFRead[**?**] and several others. Manual installation of these dependencies would be difficult, time consuming and in general advised against. In this case, many of Braker2's requirements are satisfied by modules already included in the environment, making installation relatively simple if one is familiar with the Digital Research Alliance of Canada's software stack. This case still required installation of some Perl modules in addition to loading necessary modules. Alternatively, one could use a package manager like Conda to handle installation of packages. This is perfectly reasonable, but also requires knowledge specific to Conda, which can be complicated and frustrating for users with little software management experience. Once installation is finished, the Braker2 pipeline is relatively straightforward to run as well, with excellent documentation included both online and through the built in help message. The training, including RNAseq alignments, and gene finding pipeline in this case took 1 hour and 17 minutes using 60 threads. Applying the Braker2 trained gene finding model to DC1 with 60 threads took 21 minutes to complete. Run times will of course vary depending on processing power available

to the end user, but in the case of *Trichoderma* genomes, users can expect quick results with relatively little computing power. Once annotation is complete, Braker2 produces a GFF file containing predicted genes along with CDS sequences and amino acid sequences their protein products.

In summation, Braker2 and GeneMark are not direct plug-and-play software packages. Users should expect to encounter issues when getting these programs running in addition to normal downloading and unpacking of software packages so some expertise is recommended. Neither Braker2 nor GeneMark require users to compile software, however Braker2's dependencies may require additional compilation and attention. Once installed, both tools are relatively simple to use with documentation available for both on the command-line and excellent documentation available for Braker2 on their GitHub page. Outputs from both tools are similar although Braker2 has the ability to output coding and amino acid sequences for downstream processing. Both tools run in reasonable amounts of time, where in the case of smaller genomes such as *Trichoderma*, users can expect results within a few hours to a day depending on number of computing cycles available to them.

2.4 Initial Gene Finding Results

Prior to discussing gene finding results, we will first define the terms gene and coding sequence in the context of this work. We refer to a gene as a set of start and stop coordinates in a genomic sequence. This definition of a gene simplifies processing, although the definition could be expanded in the future to include introns, exons and potential up and downstream sequences as well. Gene finders may also predict isoforms, or alternative splice variants of a gene based on evidence provided in training, resulting in multiple potential coding sequences for a gene. In this work, we refer to coding sequences as the set of all coding sequences predicted by a gene finder.

Counts of genes and coding sequences predicted by Braker2, GeneMark and RefSeq are shown in Table 2.2. Immediately we see that Braker2 predicts far fewer genes in all assemblies, except in the case of *Trichoderma reesei.* This is possibly due to the effects of training the Braker2 gene model using data from *Trichoderma reesei*, which has a significantly smaller genome in comparison to other *Trichoderma* assemblies, although genome size is not always indicative of gene content. Regardless, we observe a difference in the number of genes predicted by Braker2 in comparison to GeneMark and RefSeq. The number of genes predicted by GeneMark and RefSeq are similar, except in the case of *T. harzianum*, in which RefSeq predicts roughly 17% more genes than GeneMark. Braker2 consistently predicts more coding sequences than GeneMark and RefSeq. RefSeq also appears to predict multiple coding sequences for each gene but in fewer numbers than Braker2. Coding sequence prediction counts in *T. harzianum* from RefSeq are also interesting, with RefSeq predicting fewer coding sequences than genes. Why this occurs is unknown but may warrant further investigation.

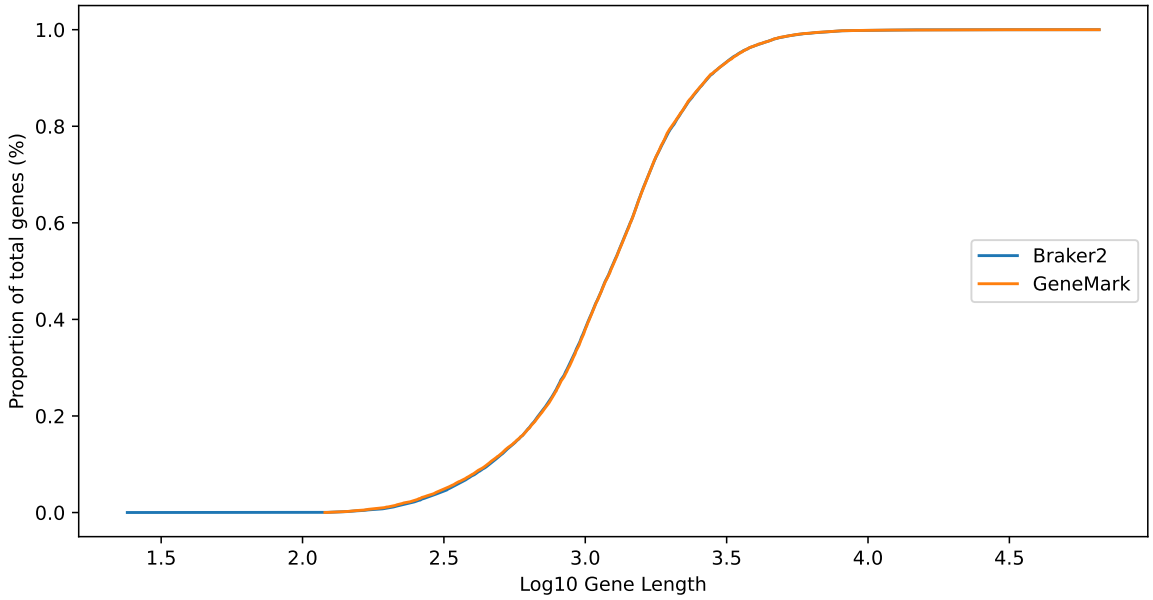
Assembly	Braker2		GeneMark		RefSeq	
	Genes	CDS	Genes	CDS	Genes	CDS
DC1	8546	8637	11353	11353	N/A	N/A
Tsth20	8784	8858	12362	12362	N/A	N/A
T. reesei	9659	10175	9196	9196	9109	9118
T. harzianum	8314	8385	12164	12164	14269	14090
T. virens	7801	7863	11866	11866	12405	12406

Table 2.2: Number of genes and coding sequences (isoforms) predicted by each gene finder for each *Trichoderma* genome.

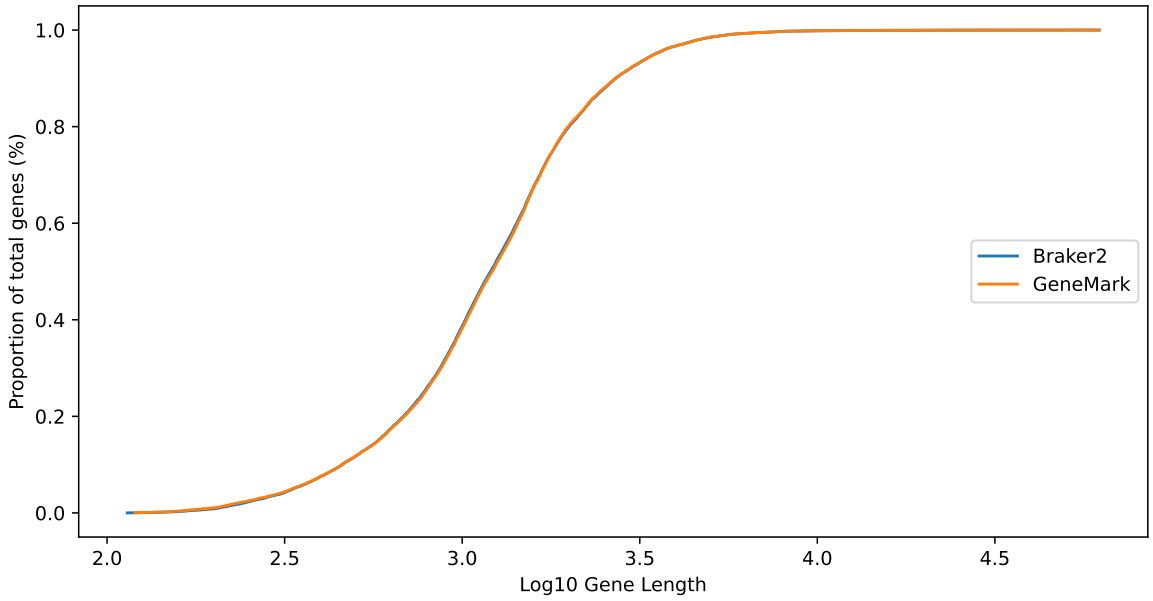
2.5 Distribution of Predicted Coding Sequence Lengths

To better understand and compare the distributions of CDS sequences predicted by Braker2, GeneMark, and RefSeq, the cumulative distribution function for the lengths of CDS sequences for each gene finding tool are shown in Figures 2.2, 2.3 and 2.4. The $\log_{10}$ values of gene lengths were used as the abscissa for a better visulazation of the distributions. In DC1, the curves from Braker2 and GeneMark follow each other closely, with the only variation being genes of short length, where Braker2's curve extends beyond that of GeneMark, indicating that Braker2 predicts the shortest genes in all assemblies except *T. virens*. In the case of Tsth20, the curves are nearly identical. In *T. reesei*, we see disagreement in the curves for shorter genes, with Braker2 appearing to predict a larger fraction of shorter genes than GeneMark and RefSeq. The right sides of the curves trend toward similar predicted gene lengths. In *T. harzianum*, RefSeq deviates from GeneMark and Braker2, predicting more genes of short length, while Braker and GeneMark appear to be in near-complete agreement except in the case of very short genes. Finally, in *T. virens*, we see the RefSeq curve predicting a larger fraction of shorter genes once again, although the deviation is not as drastic as in *T. harzianum*.From these plots, we can say that visually, gene finding tools appear to predict different lengths of genes. We also observe that Braker2 typically predicts the shortest genes of all three gene finding methods.

To confirm that CDF curves differ, two-sided two-sample Kolmogorov-Smirnov[**?**] tests were performed using the $\log_{10}$ transformed gene lengths, with the null hypothesis being that frequency of genes predicted in AT-rich and normal genomic sequence are same. Results are presented in Table 2.3. In the cases of DC1 and Tsth20, we see that in agreement with Figure 2.2, Braker2 and GeneMark do not produce statistically different lengths of genes, which is interesting considering that the Braker2 includes experimental evidence for another assembly while GeneMark does not. In *T. reesei*, Braker2's predicted gene lengths are significantly different from both RefSeq and GeneMark, which is also evident in the CDF plots. The same cannot be said for *T. harzianum* and *T. virens*, where RefSeq is significantly different from both GeneMark and Braker2, which are not significantly different from each other. It is also notable that RefSeq and GeneMark predict similar gene lengths in *T. reesei*, but not in *T. harzianum* and *T. virens.*

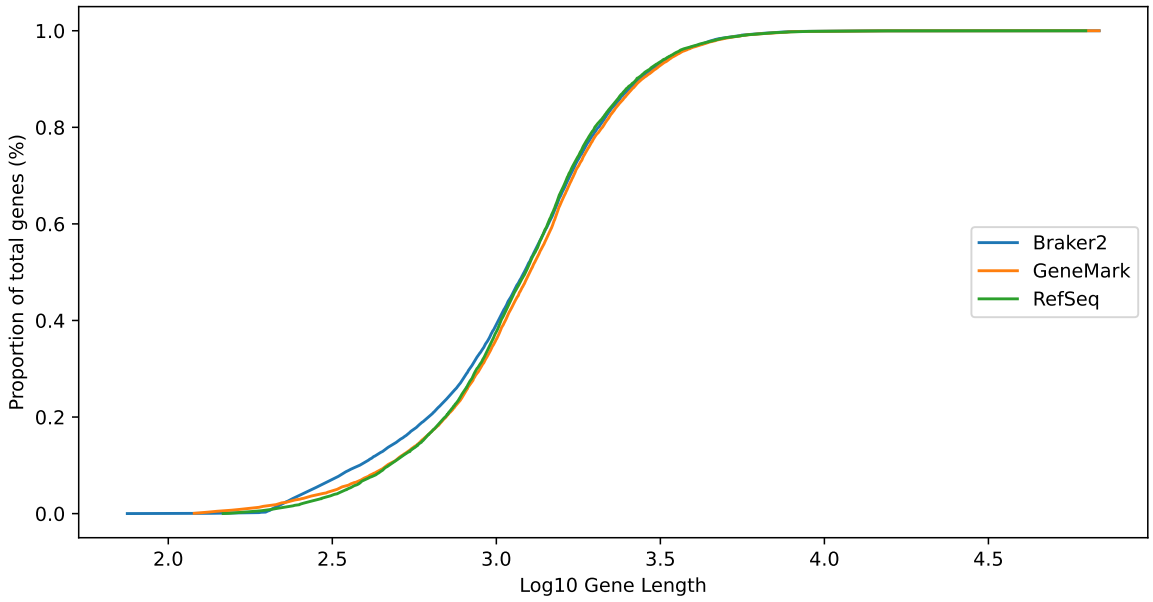


(a) DC1

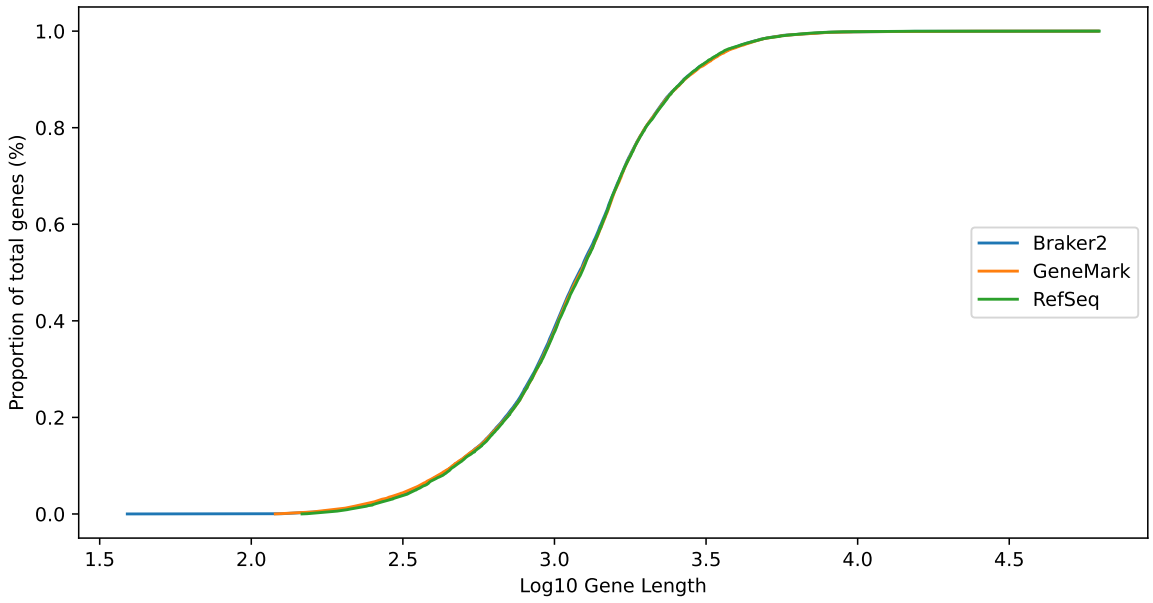


(b) Tsth20

Figure 2.2: Plots of the cumulative distribution function for CDS lengths produced be each gene finding tool applied to DC1 and Tsth20

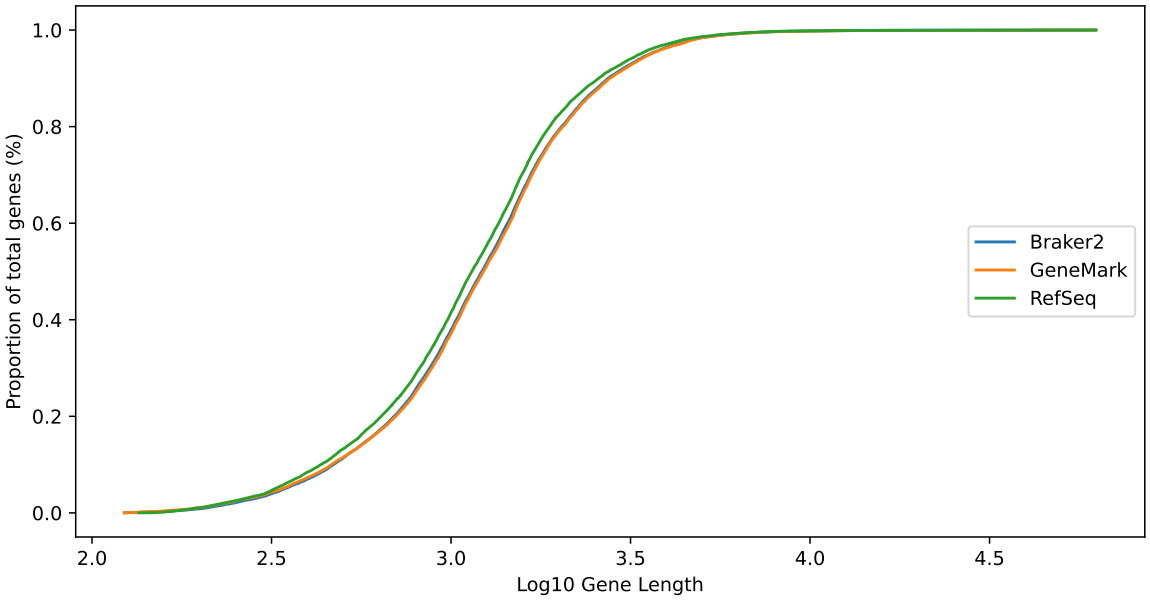


(a) *T. reesei*



(b) *T. harzianum*

Figure 2.3: Plots of the cumulative distribution function for CDS lengths produced be each gene finding tool applied to *T. reesei* (GCF_000167675.1_v2.0) and *T. harzianum.* (GCF_003025095.1_Triha_v1.0)



(a) *T. virens*

Figure 2.4: Plots of the cumulative distribution function for CDS lengths predicted by each gene finding tool when applide to *T. virens* (GCF_000170995.1_TRIVI_v2.0).

Genome	Tool #1	Tool #2	*P*-value
DC1	Braker2	GeneMark	0.999
Tsth20	Braker2	GeneMark	0.965
T. reesei	Braker2	GeneMark	$9.481 * 10^{-07}$
T. reesei	GeneMark	RefSeq	0.002
T. reesei	Braker2	RefSeq	$1.340 * 10^{-07}$
T. harzianum	Braker2	GeneMark	0.863
T. harzianum	GeneMark	RefSeq	$4.313 * 10^{-52}$
T. harzianum	Braker2	RefSeq	$4.674 * 10^{-55}$
T. virens	Braker2	GeneMark	0.635
T. virens	GeneMark	RefSeq	$7.352 * 10^{-12}$
T. virens	Braker2	RefSeq	$1.794 * 10^{-09}$

Table 2.3: Table of *P*-values from two-sided two-sample Kolmogorov-Smirnov tests between gene finding tools.

It can clearly be stated that these gene finding tools predict different distrbutions of gene lengths, paritcularly in *T. reesei, T. harzianum* and *T. virens*. Why that may be the case is difficult to answer without deeper invesitgation. There is clearly an underlying difference between RefSeq and the other gene finding tools. Braker2 and GeneMark tend to be in agreement, except in the case of *T. reesei*, for which Braker2 was specifically trained. We observe that when Braker2 is applied to an assembly from which the training data originated, Braker2 predicts a larger fraction of shorter genes. Conversly, we observe that when Braker2 is trained with experimental evidence and applied to a *Trichoderma* assembly from which the training data did not originate, the distributions of predicted genes lengths do not significantly differ from the *ab initio* gene finder GeneMark.

2.6 BUSCO Results

The results of BUSCO analysis using the fungal subset provided by BUSCO are presented in Table 2.4. Results from BUSCO indicate that all gene sets considered in this analysis have a BUSCO completeness of 99.2% or higher, with a maximum completeness of 99.9% for some gene sets. In general, Braker2 and RefSeq have the most BUSCO complete sets of gene predictions of the three tools considered. Interestingly, Braker2 produces far more duplicated BUSCO matches than both GeneMark and RefSeq. Examining the BUSCO output logs, this appears to be due to Braker2 predicting more than one coding sequence for some genes predictions, resulting in multiple similar proteins. In general, all gene finders perform exceptionally well in regards to BUSCO performance. While these results do not capture the entire set of genes possibly present in these *Trichoderma* assemblies, they do confirm that the gene finders are at minimum predicting many evolutionarily conserved fungal genes.

While BUSCO matches are a good metric for general performance of gene finders, it is also important to investigate BUSCO proteins without matching gene predictions. Table 2.4, shows breakdowns of genes missed by each gene finder across the *Trichoderma* assemblies. Braker2 misses four unique proteins across the five *Trichoderma* assemblies, with only one protein missing in more than one assembly. This protein represents a formyl transferase protein. GeneMark predictions miss six unique BUSCO proteins, with two proteins missing in more than one assembly. These proteins are the same formyl transferase missed by Braker2, which was missed in four of the five assemblies, and a ubiquitin-conjugating enzyme, which was missed in all five assemblies. RefSeq, being the gene finder with the most BUSCO complete set of gene predictions, misses only three unique BUSCO proteins in the three assemblies. Those missing proteins represent a YEATS protein domain and Midasin protein. Of those three proteins, the Midasin and YEATS proteins are missed in two of the three assemblies from NCBI. Those missing proteins represent a YEATS protein domain and Midasin protein.

Braker2, GeneMark and RefSeq all demonstrate excellent coverage of the BUSCO fungal protein set,

Strain	Complete	Single	Duplicated	Fragmented	Missing
DC1	99.5	80.2	19.3	0.1	0.4
Tsth20	99.9	81.7	18.2	0.0	0.1
T. harzianum	99.7	80.2	19.5	0.0	0.3
T. virens	99.8	79.0	20.8	0.1	0.1
T. reesei	99.9	85.5	14.4	0.1	0.0

(a) Braker2

Strain	Complete	Single	Duplicated	Fragmented	Missing
DC1	99.2	98.8	0.4	0.3	0.5
Tsth20	99.8	99.1	0.7	0.0	0.2
T. harzianum	99.6	98.9	0.7	0.0	0.4
T. virens	99.7	99.2	0.5	0.1	0.2
T. reesei	99.6	99.5	0.1	0.0	0.4

(b) GeneMark

Strain	Complete	Single	Duplicated	Fragmented	Missing
T. harzianum	99.9	99.2	0.7	0.0	0.1
T. virens	99.5	98.8	0.7	0.3	0.2
T. reesei	99.8	99.5	0.3	0.0	0.2

(c) RefSeq

Table 2.4: Results from BUSCO using the fungal analysis option organized by gene finding tool. The selected BUSCO dataset contains 758 markers. For more information on the categories assigned by BUSCO, please refer to the documentation.

Tool	BUSCO ID	Annotation	DC1	Tsth20	*T. reesei*	*T. harzianum*	*T. virens*
Braker2	195619at4751	Pyridoxal phosphate-dependent transferase		✓	✓	✓	✓
Braker2	285254at4751	Aminoacyl-tRNA synthetase	✓	✓	✓		✓
Braker2	348020at4751	Formyl transferase			✓		✓
Braker2	497024at4751	Zinc finger C2H2-type		✓	✓	✓	✓
GeneMark	195619at4751	Pyridoxal phosphate-dependent transferase		✓	✓	✓	✓
GeneMark	285254at4751	Aminoacyl-tRNA synthetase	✓	✓	✓		✓
GeneMark	348020at4751	Formyl transferase					✓
GeneMark	438731at4751	LSM domain	✓	✓		✓	✓
GeneMark	470813at4751	Ubiquitin-conjugating enzyme					
GeneMark	497024at4751	Zinc finger C2H2-type		✓	✓	✓	✓
RefSeq	494at4751	Midasin	N/A	N/A			✓
RefSeq	315802at4751	tRNA dimethylallyltransferase	N/A	N/A	✓	✓	
RefSeq	352224at4751	YEATS	N/A	N/A		✓	

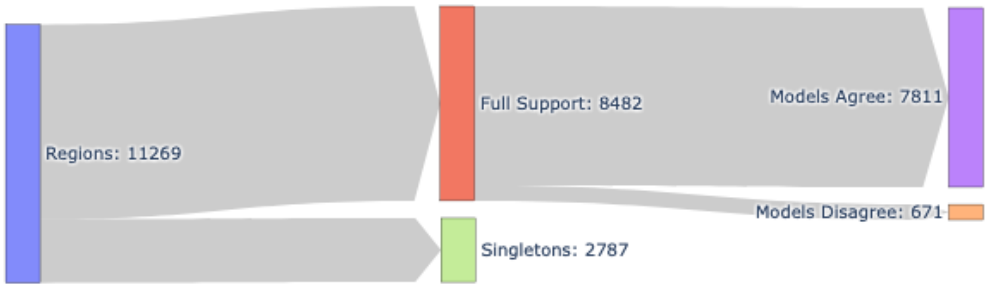
Table 2.5: The presence (✓) or absence of all BUSCO IDs missed by Braker2, GeneMark and RefSeq in each *Trichoderma* assembly.

indicating that these gene finders are capable of predicting genes that are expected to be present in these assemblies. From this we can say that the foundations of the underlying gene models used by each gene finder are solid. Braker2 produces more duplicate matches than GeneMark and RefSeq, but this is likely due to multiple isoforms of possible genes being present in the input data. Despite excellent coverage of the BUSCO fungal proteins, all three gene finders miss some BUSCO proteins in their predictions. GeneMark misses the most proteins and in particularly struggles with predicting a formyl transferase and a ubiquitin-conjugating enzyme. Braker2 also appears to have difficulty predicting a formyl transferase just as GeneMark did. RefSeq misses the fewest BUSCO proteins and does not appear to systematically miss certain proteins, although it is hard to draw a conclusion with only three assemblies considered. It is also worth noting that RefSeq misses completely different proteins than the other gene finders while Braker2 and GeneMark do share some missed proteins. Finally, we note the possibility that human curation of RefSeq datasets is responsible for these differences, but this requires further investigation.

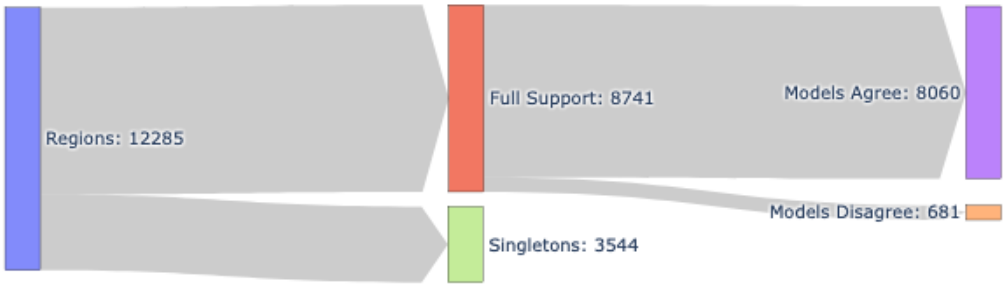
2.7 Region Identification

Visual breakdowns of the types of regions identified and their counts for each *Trichoderma* assembly are shown in Figure 2.5. We will discuss the types of regions, beginning with fully supported regions. These regions, labelled 'Full Support', are sections of genomic sequence where all three gene finders agree that a gene is present in some form. These fully supported regions are then broken down into two categories, based on whether or not the models from each gene finder agree on the start and/or stop positions of the gene model. Regions that have support from more than one gene finder, but not all, are labelled regions with 'Partial Support'. These regions are also broken down in to two subtypes based on whether or not the gene predictions agree on the start and/or stop positions of the gene. Regions with support from only one gene finder are labelled as singletons.

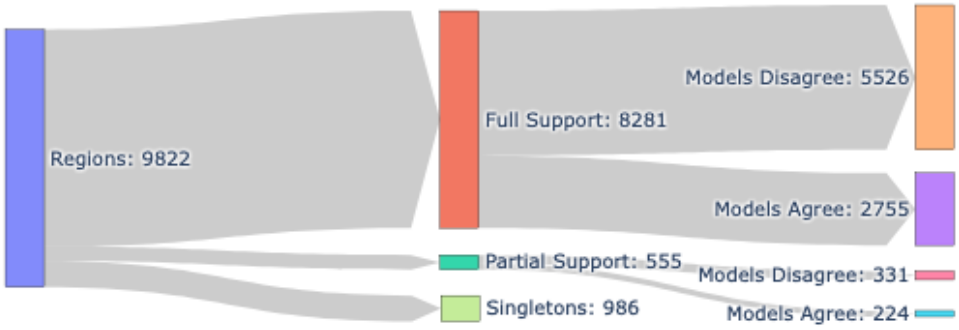
Looking at DC1 and Tsth20 in Figure 2.5, it appears that Braker2 and GeneMark predict genes in the same regions in the 69% and 65% of cases, respectively. For regions with full support in DC1 and Tsth20, the models also tend to agree on the start and stop positions of the gene in that region. This is not generally the case as we will see later when RefSeq is also considered. There are no regions with partial support for DC1 and Tsth20, as there are only two gene finders considered. *Trichoderma reesei* contains the fewest number of regions, which seems to scale appropriately with the total number of predicted genes and assembly length. Eighty-four percent of the regions are fully or partially supported by Braker2, GeneMark and RefSeq, with only 10 percent of the regions being single tool predictions. The most interesting observation here is the extent to which fully supported regions disagree on start and stop positions of the gene(s) in each region. There is clearly a difference in the gene models being produced by these gene finders. If there is also more disagreement on the start and stop positions of a gene, then there is likely disagreement on the number and location of exons and introns within the gene model as well. In the case of partially supported regions, there



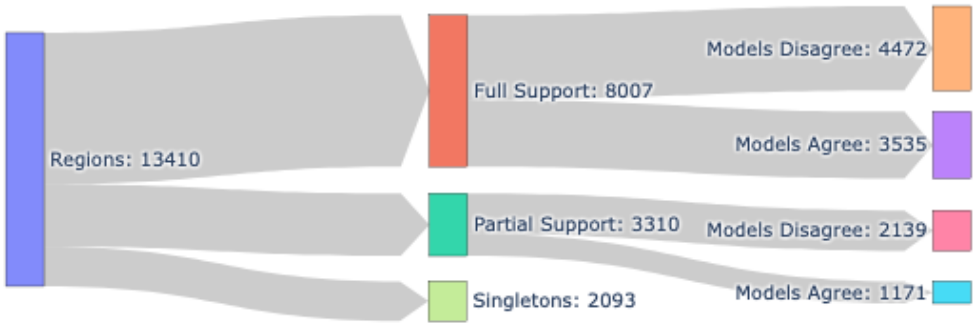
(a) DC1



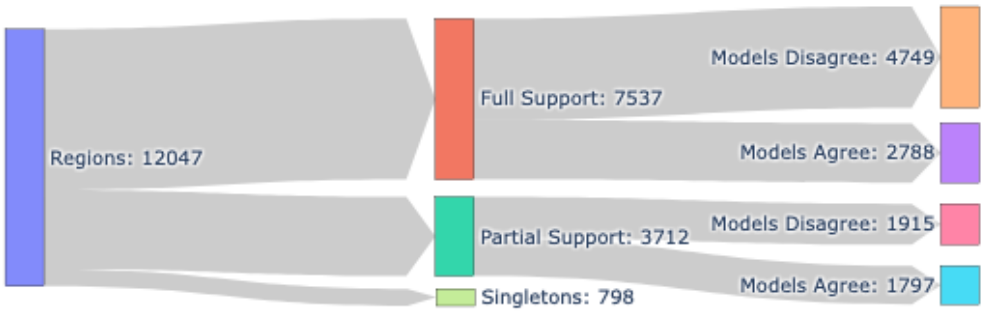
(b) Tsth20



(c) *T. reesei*



(d) *T. harzianum*



(e) *T. virens*

Figure 2.5: Figures showing breakdowns of genomic regions identified by the region finding process. Regions, in blue, are categorized based on support from gene finders. Regions with supporting predictions from all gene finders included in this analysis are labelled with 'Full Support' and colored red. Fully supported regions are then broken down into regions where gene models agree on the start and stop positions of the genes (purple), and those that do not (orange). Regions labelled with 'Partial Support' (turquoise) are regions with supporting gene predictions from two gene finders. Partially supported regions are also broken down into regions in which gene finders agree on the start and stop positions of the gene (cyan), and those that do not agree (pink). Regions with gene predictions from only one gene finder are labelled as singletons (green).

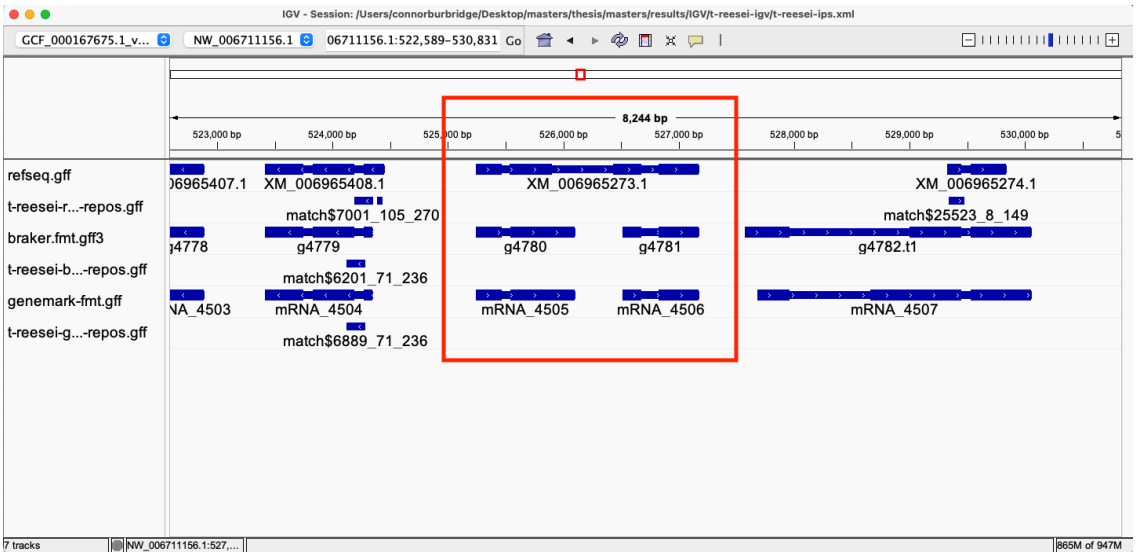


Figure 2.6: An IGV screenshot from *T. reesei* showing a region containing five gene predictions in which one of the gene finders disagrees with the other two, highlighted in the red box.

is roughly 50% to 60% agreement on start and stop positions, but that may come as less of a surprise as there is already disgreement on presence by definition. Gene finding behaviour in regions from *T.harzianum* and *T. virens* differ from the other assemblies in the split between fully and partially supported regions. There seems to be fewer regions with full support from all gene finders and the reason why is unclear. The gene models in each region, whether fully or partially supported, still tend to disagree on start and stop positions of the genes more often than agree.

It is also worthwhile investigating some potentially interesting cases of strange gene calls in regions. One such case is when a region contains more than three individual predicted genes. The null hypothesis, so to speak, is that if all gene finders agree exactly on the presence and position of a gene in a region, then one should expect exactly three predictions in that region. We observe that this is not always true. Cases of more than three gene predictions in a region were present in regions from all processed assemblies. Figure 2.6 shows an example of one of these regions, in which the single gene predicted by RefSeq spans multiple gene predictions from both Braker2 and GeneMark. In the case of DC1 and Tsth20, there were very few regions that matched this scenario, with 19 and 6 regions containing more than 3 gene predictions, respectively. Conversly, *T. reesei, T.harzianum and T.virens* contain many such regions, with assemblies reporting 546, 899 and 521 regions with greater than three gene predictions, respectively. While having predictions from each gene finding tool present in a region is a strong indicator for the presence of a gene, having more than one gene prediction per gene finder raises questions about which model (or models) is correct and why disagreement exists.

Another interesting set of criteria to investigate is whether or not gene finders always predict genes on the same strand within a region. We observe that regions containing predictions on different strands do exist, and are observed in predictions from all assemblies included in this analysis. Figure 2.7 shows an example

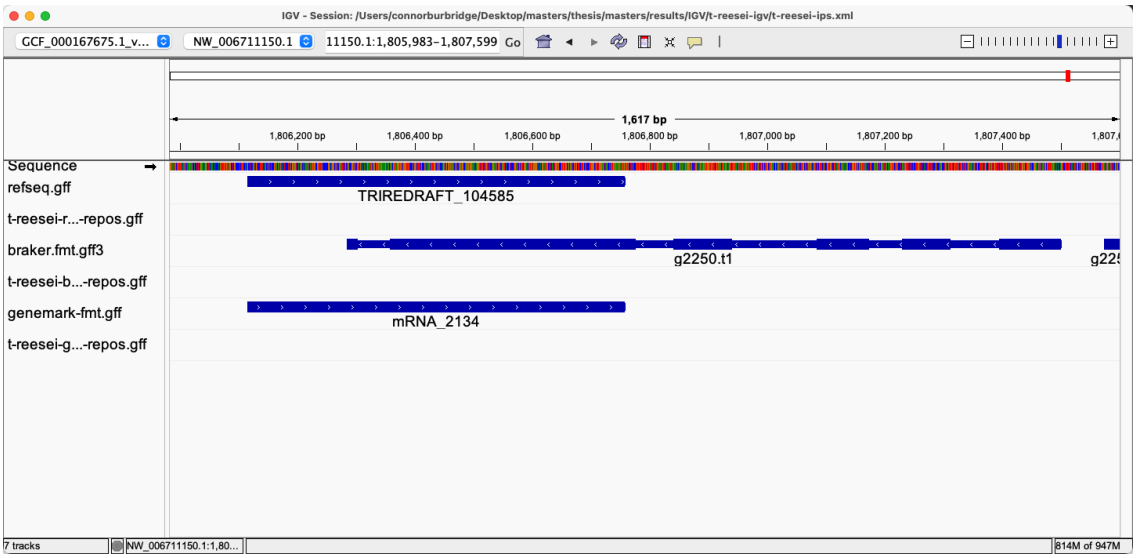


Figure 2.7: An IGV screenshot of *T. reesei* showing a region which contains gene predictions on opposite strands.

of a region containing gene predictions on opposing strands. As in the case of regions with more than three gene predictions, DC1 and Tsth20 have fewer mixed strand regions with 29 regions identified in DC1 and 46 regions identified in Tsht20. Results from *T. reesei, T. harzianum* and *T. virens* show more regions in which this property is true, with region counts of 203, 533 and 293 respectively. Under the assumption that gene finders should predict the same genes on the same strands, it is unexpected to find so many of these cases, although it is possible that these overlapping sense and anti-sense genes exist naturally[**?**].

In summary, gene finders agree partially or completely on the presence of a gene in the vast majority of cases. While gene finders generally agree on the presence of a gene, they tend to disagree on the underlying gene model more often than they agree, except in in the case of DC1 and Tsth20. This is likely due to only including two gene finders in their analysis rather than three, resulting in fewer opportunities for disagreement. This observation is true when applied to the start and stop positions of the gene, but further investigation and comparison of intronic and exonic sequences between genes may provide more insight. It is also possible that genome quality may affect agreement between gene finders as the underlying model may be trained on contigs that are larger or of different structure than the assembly it is applied to. This is particularly true in the cases of DC1 and Tsth20, which are composed of larger contigs in comparison to the assemblies from NCBI. Finally, regions identified in this analysis do not always fit the ideal scenario of one gene prediction from each tool per region. Regions in which there are more than three gene predictions were observed in all assemblies included in this work. In addition, we observe cases where gene finders predict genes on different strands.

2.8 InterProScan as Supporting Evidence for Predicted Genes

Pfam hits from InterProScan analysis are presented in Table 2.6, from which we can identify one major trend. In general, roughly 73-76% of proteins predicted by Braker2, GeneMark and RefSeq contain a match to a Pfam entry, except in the case of *T. harzianum*, which reports a considerably lower proportion of genes with Pfam matches in the RefSeq dataset. Why the proportion of RefSeq proteins with Pfam matches in *T. harzianum* is so low is unknown. This is promising performance for the gene finders as the RefSeq annotations demonstrate similar proportions of Pfam hits to predicted proteins. The total counts may be deceiving however, as predictions from Braker2 may result in more than one protein product per gene, whereas in the case of GeneMark, only one protein is produced per gene model. From visual inspection of Pfam hits mapped back to the references, it appears that in general, when gene finders agree that a gene is present, InterProScan reports the same Pfam match in all three predictions. An example of agreement between Braker2, GeneMark, RefSeq and InterProScan is shown in Figure 2.8. It is important to note that the position of the Pfam match in IGV does not indicate the true position of the Pfam match in the gene, but provides an indication of the presence of Pfam matches. Pfam matches are offset from the start of the gene based on the start and end position of the Pfam match in the protein sequence. For example, if a Pfam match has a start position 10 amino acids into the protein sequence, the corresponding start position in the resulting GFF is 10bp downstream from the start of the gene.

Assembly	Braker2	GeneMark	RefSeq
DC1	$\frac{10676}{14479}$	$\frac{8416}{11354}$	N/A
Tsth20	$\frac{11389}{15546}$	$\frac{9168}{12373}$	N/A
T. reesei	$\frac{8471}{11704}$	$\frac{6990}{9196}$	$\frac{6964}{9111}$
T. harzianum	$\frac{11370}{15408}$	$\frac{9061}{12164}$	$\frac{9293}{14065}$
T. virens	$\frac{11249}{15062}$	$\frac{8871}{11866}$	$\frac{9062}{12383}$

Table 2.6: Table showing the fractions of predicted genes with Pfam annotations from InterProScan as a percent

While cases of complete agreement are abundant, cases of disagreement also exist and in strange forms. In many cases, while the gene finders agree on the presence of a gene, only the RefSeq protein product contains a match to the Pfam database. Why this may be the case is unclear, and may warrant further investigation. There are also many cases in which InterProScan reports the same Pfam hits for individual proteins, but the gene models in the region do not agree. There are even cases such as the region shown in Figure 2.9, where Braker2 and GeneMark agree that two genes and their associated proteins and Pfam matches are separate, but RefSeq only reports one gene with multiple Pfam hits. There are also several cases where two tools are in agreement with proteins containing Pfam hits while another is not. Even more interesting are cases such as the one shown in Figure 2.10, in which Braker and GeneMark predictions contain Pfam matches while

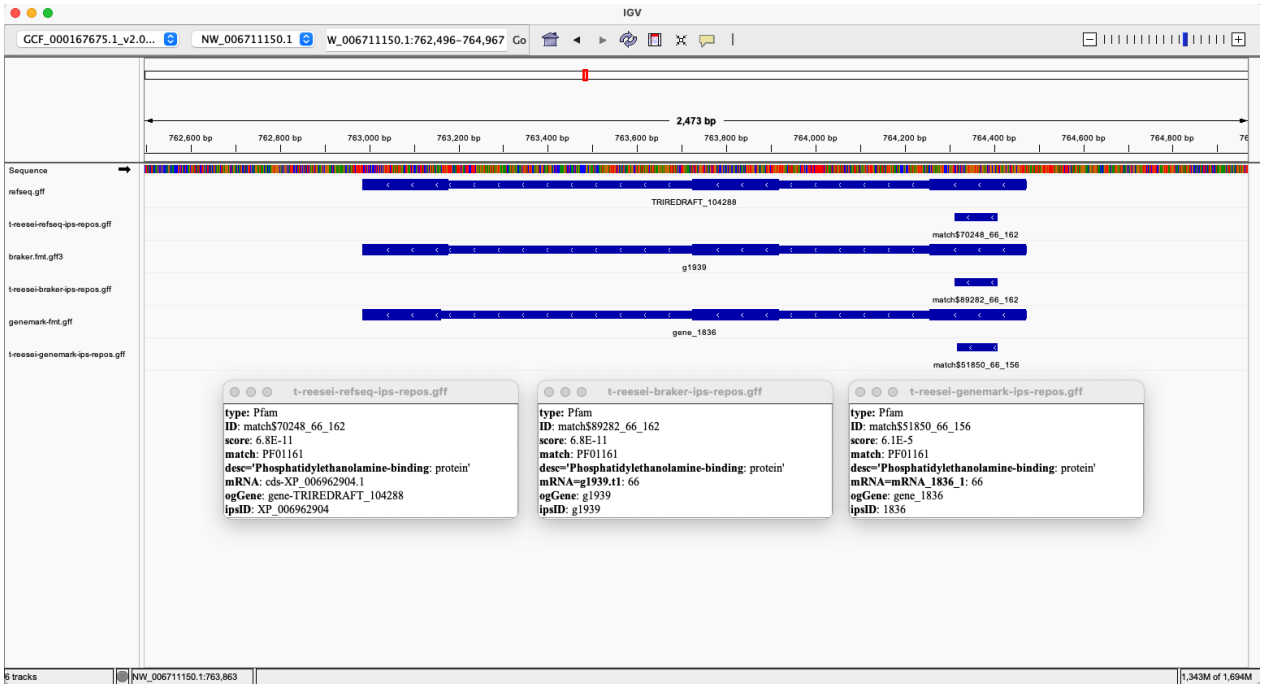


Figure 2.8: An IGV capture showing complete agreement between gene finders for both gene model and protein Pfam hits. The longer segments are predicted genes and the smaller segments are annotated Pfam matches, all with agreeing start and stop positions. The **desc** in each sub-window indicate agreeing Pfam annotations.

RefSeq does not report a gene at all. This may be the due to experimental data used in the RefSeq training process or a result of curation. Regardless of the tool, these cases demonstrate well that gene finders are not always in agreement even on well known proteins, to the point that predictions are not present even though protein products from other gene finders contain known Pfam matches.

In summary, the protein products from Braker2 and GeneMark predictions do contain matches to the Pfam database. The proportions of matches to total proteins are similar to that of the RefSeq annotation, sitting between 65 and 75 percent. While Pfam hits to Braker2, GeneMark and RefSeq proteins generally agree, we do observe regions in which there is disagreement in several forms.

2.9 BLAST Results

As a control, proteins from three related organisms were used as queries to tblastn[**?**] on each *Trichoderma* assembly. The organisms and their associated NCBI accession IDs are as follows: *T. atroviride* - GCF_000171015.1, *F. graminarium* - GCF_000240135.3, *S. cerevisiae* - GCF_000146045.2. Results from the tblastn runs are presented in Table 2.7. Initial tblastn results appear promising for both the *T. atroviride* and *Fusarium* datasets. All assemblies considered contain at minimum 89% of the reference protein sequences in the case of *T. atroviride* and a minimum of 75% in the case of *Fusarium*. In the case of *S. cerevisiae*, a minimum of 57% of reference proteins matched. It appears that the percentage of hits decreases as evolutionary

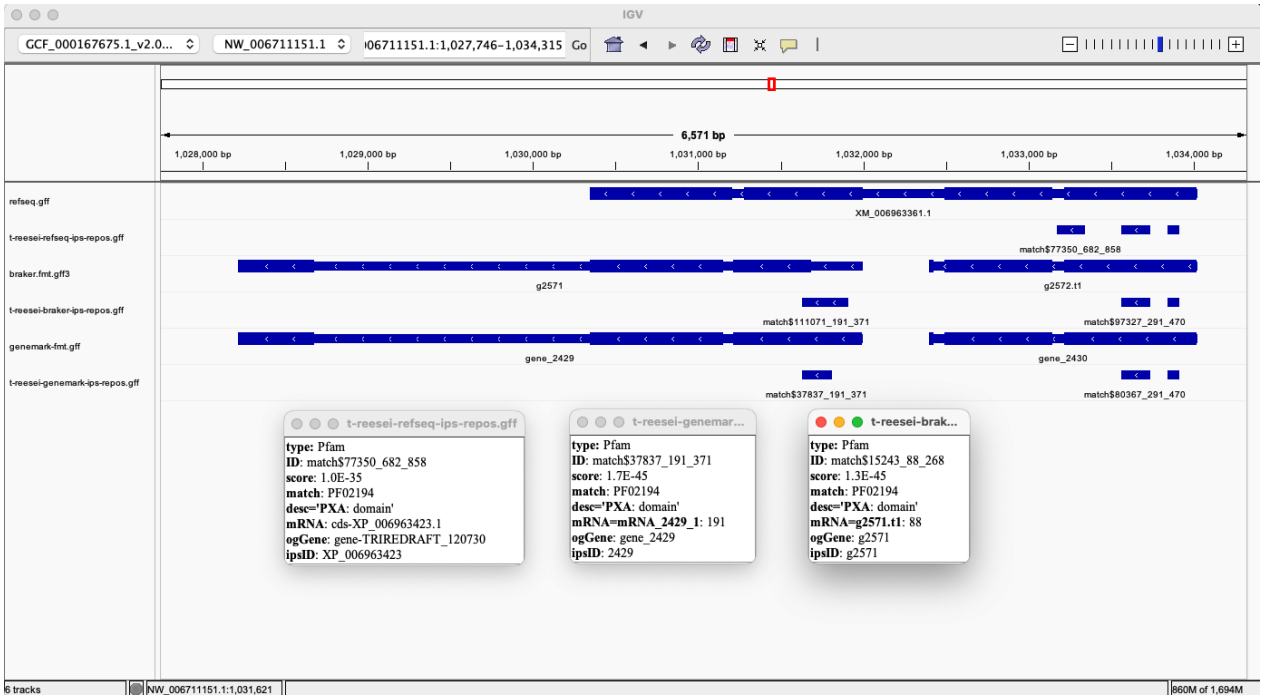


Figure 2.9: An IGV capture showing Braker2 and GeneMark reporting two genes and their resulting proteins and Pfam hits as separate, while RefSeq reports one gene, one protein and three Pfam matches.

Reference	Ref. Proteins	DC1	Tsth20	*T. reesei*	*T. harzianum*	*T. virens*
Trichoderma atroviride	11807	11552	11080	10601	11081	11078
Fusarium granminarium	13312	10327	10429	10064	10434	10490
Saccharomyces cerevisiae	6014	3537	3517	3445	3509	3500

Table 2.7: tblastn hits from reference protein sequences to selected *Trichoderma* assemblies. Hits are reported if the alignment length is greater than 30% of the reference protein length and if 30% of the alignment length has identical matches.

distance increases. These results provide rough validation that the assemblies contain potential for protein coding sequences.

While successful tblastn hits from input proteins are useful, it is worthwhile exploring those hits in the context of regions and gene predictions within them. Counts of genes from regions with tblastn hits are shown in Table 2.8. In a similar trend to Table 2.7, genes in regions with tblastn hits decrease as evolutionary distance increases. In DC1 and Tsth20, we observe that Braker2 predicts more genes in regions with tblastn hits than GeneMark. RefSeq is not included for DC1 and Tsth20, as there is no RefSeq annotation available. In *T. harzianum* and *T. virens*, Braker2 and RefSeq appear to have similar counts of genes in regions with tblastn hits. Interestingly, in the case of *T. reesei*, there are more genes from GeneMark in regions with tblastn hits, which is not the case in *T. harzianum* and *T. virens*.

Another interesting observation is comparing the coverage, or completeness, of the input and query se-

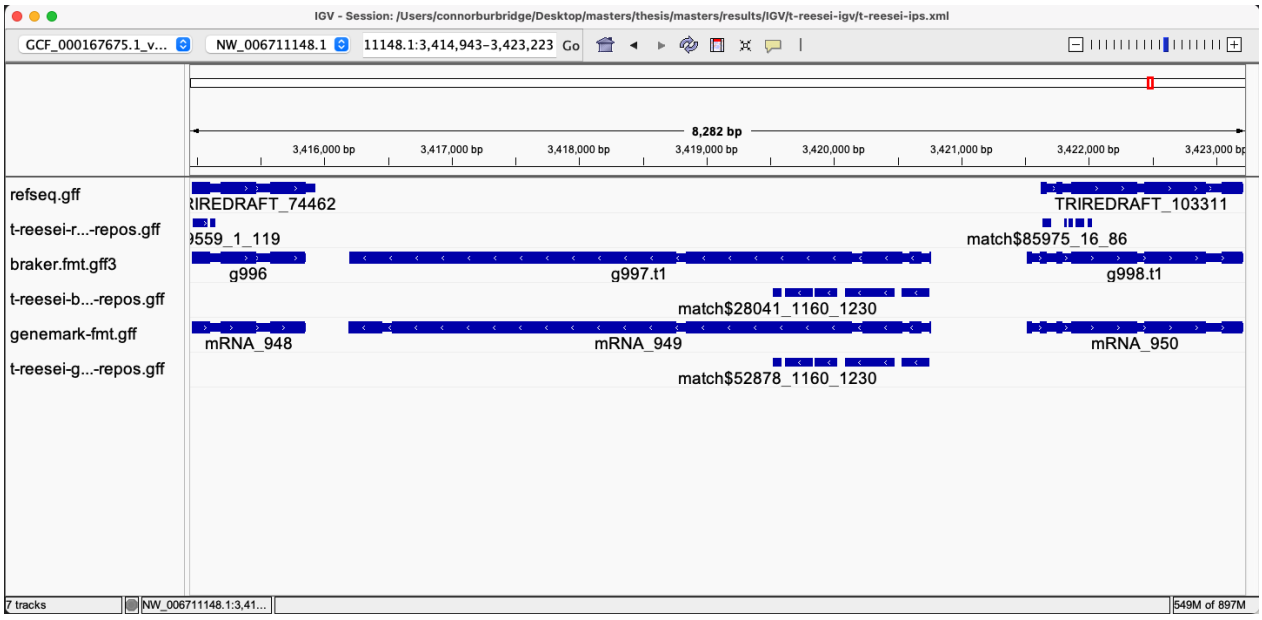


Figure 2.10: An IGV capture showing a scenario where GeneMark and Braker2 agree on a gene model with supporting Pfam evidence and RefSeq does not report any gene.

quence. As mentioned, it appears that high proportions of query sequences are being mapped to the subject, but the gene predictions are not as well represented in regions where those tblastn hits occur. A likely reason for this is inappropriate alignment parameters. More flexible gap parameters and the use of a different scoring matrix would likely produce better results, but optimization of parameters was not performed in this analysis. It is also possible that the post-alignment filtering criteria were inappropriate, resulting in short alignments that are true in nature, but may not be indicative of the presence of a truly similar protein.

Overall, it appears that using tblastn with proteins from related organisms may be a viable option as a control when no reference or model organism is available. At the very minimum, tblastn hits can be used as another piece of supporting evidence for the existence of a gene, with one caveat being imperfect coverage of predictions. The issue of imperfect coverage may be mitigated by parameter optimization. It is also possible that one could employ a similarity-based cutoff on the proportion of predicted genes in regions with tblastn hits as an indicator of gene finding performance, however this would require further analysis, preferably with more closely related organisms.

2.10 Genes in Regions of Anomalous GC Content

Table 2.9 shows the results of applying the same region finding process from earlier to segments of the genome identified as AT-rich. AT-rich is defined as a region of genomic sequence with percent GC composition less than 28% as determined in Section 2.2. In comparison to results from Section 2.7, we see that overall there are far fewer regions with gene predictions in AT-rich genomic segments. In DC1, Tsth20, there are very few regions with full support from both Braker2 and GeneMark, but more singletons. Again, as in Section

Subject	Query	Braker2	GeneMark	RefSeq
DC1	*T. atroviride*	5902	4679	N/A
DC1	*F. graminarium*	4955	4114	N/A
DC1	*S. cerevisiae*	2105	1850	N/A
Tsth20	*T. atroviride*	6065	4626	N/A
Tsth20	*F. graminarium*	5365	4191	N/A
Tsth20	*S. cerevisiae*	2211	1869	N/A
T. reesei	*T. atroviride*	5072	5174	4989
T. reesei	*F. graminarium*	4577	4685	4529
T. reesei	*S. cerevisiae*	2055	2114	2022
T. harzianum	*T. atroviride*	6363	4611	6835
T. harzianum	*F. graminarium*	5659	4198	5982
T. harzianum	*S. cerevisiae*	2424	1963	2560
T. virens	*T. atroviride*	6256	4437	6415
T. virens	*F. graminarium*	5568	4075	5664
T. virens	*S. cerevisiae*	2318	1861	2352

Table 2.8: Counts of transcripts predicted by Braker2, GeneMark and RefSeq in regions containing tblastn hits from *Ttrichoderma atroviride*, *Fusarium granminarium*, and *Saccharomyces cerevisiae*.

Assembly	Full Support	Partial Support	Singletons	No. Genes
DC1	11	N/A	20	42
Tsth20	2	N/A	9	13
T. reesei	25	18	54	194
T. harzianum	26	43	68	265
T. virens	8	11	0	49

Table 2.9: Total number of regions identified in each assembly followed by counts of regions (both agreement and singletons) in regions of AT-rich genomic sequence. It is important to note that in the cases of DC1 and Tsth20, full support indicates supporting gene predictions from both GeneMark and Braker2.

2.7, there are no regions with partial support as only two gene finding tools were applied to those assemblies. *T. reesei* and *T. harzianum* report more regions in AT-rich genomic sequence than the other assemblies, but with the majority of regions belonging to the singleton category. *T. virens* is an interesting case, reporting a similar numbers of regions and genes in AT-rich genomic sequence as DC1 and Tsth20. *T. virens* is also the only assembly to report zero singleton gene predictions. Why *T. virens* differs from the other RefSeq assemblies is unclear. In general, there are few regions with gene predictions in AT-rich genomic segments, and within these regions, the majority of cases are isolated singleton gene predictions. These observations differ greatly from those made in nucleotide composition agnostic approach in Section 2.7.

In addition to a breakdown of regions in AT-rich genomic sequence, understanding which gene finders predict more or fewer genes in these regions may be of interest. It is possible that an HMM, trained on genomic sequence with varying nucleotide content, may predict genes differently to an HMM trained only on sequences with uniformly distributed nucleotide composition as the variation in nucleotide composition may affect the various states and relationships between them in an HMM. Table 2.10 shows the number of genes predicted by each gene finding tool in regions of AT-rich genomic sequence. GeneMark appears to predict the fewest genes in AT-rich regions, while RefSeq appears to predict the most. Braker2 lies somewhere in the middle. Again, *T. virens* appears as an odd case, with very few predictions from all gene finders. Why *T. virens* differs from the other RefSeq assemblies is unclear.

Finally, to test the probability of any given gene prediction falling in an AT-rich genomic sequence, a two-sided binomial test was performed to determine if the number of genes predicted in AT-rich sequences is proportional to fraction of genomic sequence they comprise. The null hypothesis in this case is that the number of genes in AT-rich sequence is proportional to the total AT-rich sequence in the genome. The results of the test are shown in Table 2.11. In all cases, it appears that the gene finding tools selected for this analysis do not predict the same proportion of genes in AT-rich genomic sequence as in typical genomic sequence.

In summary, very few regions with gene predictions are present in AT-rich genomic sequence. Additionally, genes predicted in these regions tend to be isolated and not supported by other gene finders, although some

Assembly	Braker2	GeneMark	RefSeq
DC1	31	11	N/A
Tsth20	11	2	N/A
T. reesei	39	48	107
T. harzianum	81	30	154
T.virens	21	8	20

Table 2.10: Number of genes predicted by Braker2, GeneMark and RefSeq in AT-rich genomic sequence from each assembly. It is important to note that in the cases of DC1 and Tsth20, full support indicates supporting gene predictions from both GeneMark and Braker2.

Tool	DC1	Tsth20	*T. reesei*	*T. harzianum*	*T. virens*
Braker2	$9.56 * 10^{-181}$	$1.14 * 10^{-259}$	$2.68 * 10^{-96}$	$4.05 * 10^{-140}$	$1.35 * 10^{-35}$
GeneMark	$5.12 * 10^{-216}$	0.0	$5.66 * 10^{-49}$	$5.37 * 10^{-219}$	$5.31 * 10^{-35}$
RefSeq	N/A	N/A	$1.29 * 10^{-49}$	$2.44 * 10^{-205}$	$7.40 * 10^{-33}$

Table 2.11: p-values produced from a two-sided binomial test for each combination of tool and assembly.

agreement is observed. In terms of number of genes predicted, RefSeq tends to predict the most genes in these AT-rich regions while GeneMark predicts the fewest. Lastly, the selected gene finding tools do not predict genes in AT-rich sequences in proportion to the fraction of genomic sequence they comprise.

Conclusions

3.1 Selection of Gene Finding Tool

With all of these results, it makes sense to explore the question of which gene finding tool one should choose for optimal gene prediction performance. Comparisons drawn in this section are made in the context of *Trichoderma* assemblies and may not extend to other datasets. Observations from the results portion of this work have been converted to scores relative to the other gene finders. Scores range from zero to three, with values being: 0 - failing performance, 1 - passing performance, 2 - good performance, and 3 - excellent performance. Results from this work are summarized in Table 3.1. Ignoring availability and use of the gene finding tools, it would appear that RefSeq performs the best in the remaining categories, earning top marks in every category except in it's ability to predict very short genes. Braker2 earns second place; however, this does not capture Braker2's failure in predicting accurate numbers of genes in DC1, Tsth20, *T. harzianum* and *T. virens*. GeneMark comes in last, excelling only in number of genes predicted and Pfam support for the genes that it predicts.

Relating these observations to use-case scenarios, in the case that your organism of interest has a RefSeq annotation associated with it, the RefSeq gene prediction process appears to produce the best set of predictions. If users also have experimental evidence, such as RNAseq data under experimental conditions, it may be worthwhile training a Braker2 model and predicting genes with Braker2 to supplement the already well performing RefSeq gene predictions. In a similar case, if the organism is of RefSeq status, but the assembly in question is novel to the work being performed, training and prediction using a Braker2 model with experimental evidence either from the novel genome or from the RefSeq accession in addition to use of the RefSeq annotation would likely be the best option. If the organism of interest is not a RefSeq individual but training data is available for that organism, Braker2 is the next best option, although it is important to note that the application of a trained Braker2 prediction model to an organism from which the training data did not originate is not advised based on the results presented in this work (see 2.4). While it is true that the *T. reesei* genome differs from other *Trichoderma* genomes, it's status as a representative RefSeq organism makes it somewhat of a gold standard. In this case, applying a gene model trained using evidence from the gold standard produces biased numbers of genes predicted in other *Trichoderma* genomes. While Braker2 technically scores the second highest, users must be very careful when selecting training data, and ensure that the training data either comes from the organism of interest, or comes from a very closely related organism with a highly similar genome. In the case that no appropriate training data is available, GeneMark is still an option, and users can be confident that the tool predicts a reasonably accurate number of genes with supporting Pfam matches. It is also important to note that GeneMark does not perform as well in AT-rich regions as Braker2 and RefSeq, does not predict isoforms, and systematically fails to predict some BUSCO orthologs.

Category	Braker2	GeneMark	RefSeq
Availability	3	3	0
Ease of install	1	2	0
Ease of use	3	3	0
# of genes predicted	0	3	3
# of transcripts predicted	3	0	2
Predicts shortest genes	2	1	0
Predicts more shorter genes	1	0	3
BUSCO Performance	2	1	3
Performance in AT-rich sequence	2	1	3
Predictions with InterProScan support	3	3	3
Final Score (Publicly Available)	20	17	N/A
Final Score (Ignoring Availability)	13	9	17

Table 3.1: Table with scores attributed to performance of each gene finder in several categories. The score definitions for performance are as follows: 0 - fail, 1 - pass, 2 - good, 3 - excellent. Since RefSeq is not publicly available, it is marked as N/A in the publicly available final scores. The dashed line separates categories associated with availability and use from categories describing gene prediction performance.

When availability of a gene finding tool becomes a concern and RefSeq is not considered, the scores drop significantly for Braker2 and GeneMark as seen in the final row of Table 3.1. In this situation, Braker2 still outperforms GeneMark. If the organism of interest is not considered a representative RefSeq individual, but supporting evidence specific to that organism or a very closely related organism is available, a trained Braker2 prediction model will perform well. Again, in the case that the organism is not a RefSeq individual and no appropriate training data is available, GeneMark is still a reasonable option even with the previously identified caveats.

In summary, these results indicate that if your organism is a RefSeq organism, use the RefSeq annotation. If no RefSeq predictions are available but appropriate training data is, one should use Braker2. If the training data is of questionable similarity or not available at all, users can fall back on *ab initio* gene finders such as

GeneMark, which while not ideal, still predict genes with supporting evidence.

3.2 Future Work

While this work is extensive, there are still many areas that could be expanded on. The sheer number of gene finding tools available means that many more could be compared to the ones presented here. In addition, supplying more training data to Braker2 may improve gene finding performance, and training on RNAseq from different organisms could be used to improve the performance of Braker2. It is clear from the results that Braker2 performed worse than GeneMark and the RefSeq predictions, which was likely due to the training data being used from *T. reesei*. Different types of training data such as RNAseq, expressed sequence tags (ESTs), and protein sequences could also be used to train models that support external evidence. Another limitation of this work is that the gene finders were run with default parameters, and it is possible that tuning the parameters of the gene finders could improve performance.

Following the gene prediction process, there are a number of different analyses that could be performed on the predicted genes. For example, functional annotation of the predicted genes could be performed using tools such as BLAST2GO to assign Gene Ontology (GO) terms and KEGG pathways to the predicted genes. This could provide further insight into the functional roles of the predicted genes and their potential applications. More importantly in the context of this work, analysis of predicted genes with a tool like antiSMASH could be used to identify biosynthetic gene clusters (BGCs) in the predicted genes.

Expanding the number of genomes used in this work would also be beneficial, as the results presented here are based on a limited number of genomes. Including more genomes from different species and strains of *Trichoderma* could provide a more comprehensive understanding of gene finding performance across the genus. Additionally, comparing the performance of gene finders on other fungal genomes could provide insights into the generalizability of the findings.Examining the performance of gene finders on different assemblies of the same genome could also be useful, as different assemblies may have different levels of completeness and accuracy. This could help to identify tangible benefits of using one assembly over another, and could also provide insights into the limitations of gene finders when applied to different assemblies.

Finally, the results of this work could be used to inform future research in the field of fungal genomics, particularly in the case of the DC1 and Tsth20 genome assemblies. The results of these genome assemblies are high-quality, and contain near-chromosomal scale sequences. The predicted genes from these genomes are a rich resource for future research, and could be used to identify novel genes and gene families in DC1 and Tsth20. Further understanding of the mechanisms behind salt and drought tolerance as well as degradation of hydrocarbon in sub-optimal environments could be achieved by studying the predicted genes in these genomes.

References