nature protocols

Review article

https://doi.org/10.1038/s41596-025-01301-1

Tutorial: annotation of animal genomes

Received: 7 March 2025

Accepted: 23 October 2025

Published online: 28 January 2026

Check for updates

Zoe A. Clarke[1,2,9], Dustin J. Sokolowski[1,3,9], Ciaran K. Byles-Ho[4], Ruth Isserlin[2], Michael D. Wilson[1,4], Jared T. Simpson[1,3] & Gary D. Bader[1,2,5,6,7,8]

As DNA sequencing technologies improve, it is becoming easier to sequence and assemble new genomes from non-model organisms. However, before a newly assembled genome sequence can be used as a reference, it must be annotated with genes and other features. This can be conducted by individual laboratories using publicly available software. Modern genome annotations integrate gene predictions from the assembled DNA sequence with gene homology information from other high-quality reference genomes and take into account functional evidence (e.g., protein sequences and RNA sequencing information). Many genome annotation pipelines exist but have varying accuracies, resource requirements and ease of use. This genome annotation Tutorial describes a streamlined genome annotation pipeline that can create high-quality genome annotations for animals in the laboratory. Our workflow integrates existing state-of-the-art genome annotation tools capable of annotating protein-coding and non-coding RNA genes. This Tutorial also guides the user on assigning gene symbols and annotating repeat regions. Finally, we describe additional tools to assess annotation quality and combine and format the results.

Non-model organisms can provide a wealth of information, often revealing unique biological phenomena arising from their DNA sequence[1–3]. To study an organism's genome, its genes must first be identified and labeled with useful gene symbols. A high-quality genome sequence with accurate annotations improves downstream analysis, reducing false positives or negatives in diverse applications (e.g., RNA sequencing) and helping to identify novel traits in a species of interest.

Genomics for a non-model species starts with a de novo genome assembly, defined as the reconstruction of an organism's genomic DNA sequence from DNA sequencing data[4–6]. Recent advances in long-read genome sequencing now allow a single laboratory to generate a high-quality de novo genome assembly[4–7]. This research benefits from accessible methods to locate and label the genes and repeats in the assembly.

Genome annotation, defined as the identification of functional, structural and repetitive elements along a genome assembly, is continuously improving alongside sequencing and assembly methodologies[8–11]. Historically, genomes have been annotated by using comprehensive resources like RefSeq, Ensembl and 'Matched Annotation from NCBI and EBI' (MANE; https://useast.ensembl.org/info/genome/genebuild/mane.html)[12,13]. However, the increasing rate of production of new genome assemblies presents challenges for these traditional annotation hubs. Furthermore, they require that the genome assembly and associated functional data (e.g., RNA seq) be publicly available. This may present challenges if the genome assembly cannot be made publicly available or if the genome assemblers want to use annotations to evaluate and improve their assembly before making it public. As such, a community-driven approach to end-to-end genome annotation that can be performed in a single laboratory would be beneficial to many researchers.

Many bioinformatic tools exist to annotate assembled genomes, with each tool focusing on a different aspect of annotation (e.g., finding

[1]Department of Molecular Genetics, University of Toronto, Toronto, Ontario, Canada. [2]The Donnelly Centre, University of Toronto, Toronto, Ontario, Canada. [3]Ontario Institute for Cancer Research (OICR), Toronto, Ontario, Canada. [4]Genetics and Genome Biology, The Hospital for Sick Children, Toronto, Ontario, Canada. [5]Department of Computer Science, University of Toronto, Toronto, Ontario, Canada. [6]Lunenfeld-Tanenbaum Research Institute, Toronto, Ontario, Canada. [7]Princess Margaret Research Institute, University Health Network, Toronto, Ontario, Canada. [8]CIFAR Multiscale Human Program, CIFAR, Toronto, Ontario, Canada. [9]These authors contributed equally: Zoe A. Clarke, Dustin J. Sokolowski. e-mail: gary.bader@utoronto.ca

gene models, labeling gene symbols or finding repetitive elements)[14–17]. Although global genome assembly efforts such as the Vertebrate Genome Project and the Earth Biogenome Project have made high-level genome annotation recommendations, they tend to provide limited information on specific software pipelines[10,12,13,18–20]. The genomics community is currently missing clear documentation for bioinformaticians on how to integrate these tools and recommendations to generate complete genome annotations. Therefore, it is currently challenging to generate complete genome annotations without considerable bioinformatics and genomics expertise.

Our genome annotation Tutorial presents a systematic framework for annotating animal genomes on the basis of existing recommendations, benchmarks[11] and tools to support building and integrating genome annotations from multiple sources of biological evidence. The code associated with this Tutorial guides the user through the various command-line tools and scripts implemented in the pipeline (https://github.com/BaderLab/GenAnT).

Genome annotation workflow

Genome annotation involves installing, configuring and combining a diverse set of tools that integrate DNA sequences, public databases and often other sequencing data types (e.g., RNA sequencing results). The field of genome annotation contains considerable terminology (see Supplementary Table 1 for a glossary) and diverse file types (for further information, see Supplementary Methods: Genome file formats). Broadly, genome annotation consists of five steps: (i) identifying repetitive elements and masking repeats that can interfere with gene identification, (ii) identifying protein-coding/mRNA gene models, (iii) optimizing gene models by using multiple lines of evidence, (iv) adding non-coding RNA (ncRNA) gene models and (v) labeling gene models with the likely gene identity (i.e., gene symbol). Each step is described further in the subsequent sections of this Tutorial (Fig. 1). The tools used in this Tutorial were chosen because of their general accuracy, ease of use, performance in recent benchmarking studies[11] and ability to incorporate various data types to assimilate their results (Supplementary Table 2). Although many of the specific tools described in this Tutorial may be updated or replaced over time, these broad steps of genome annotation will remain the same. Alternative tools to the ones that we recommend are listed in Supplementary Table 3, and computational resources and bioinformatic skills required for the Tutorial are described in Box 1.

Step 1: repeat annotation and masking

The first step in genome annotation is to identify and mask repetitive regions and transposable elements (TEs; Fig. 1). The prevalence of these regions interferes with many sequence alignment–based tasks, like orthology mapping and syntenic alignments, because they create an intractable number of alignment matches. Furthermore, some TEs contain open reading frames (ORFs), which can be falsely identified as protein-coding genes. Thus, repeats should be masked (i.e., flagged or hidden) to reduce the computational time needed for the annotation process and the number of mistakes made when generating gene models.

Earl Grey[21] is a comprehensive and bioinformatically friendly tool that integrates and streamlines popular repeat annotation methods. Specifically, Earl Grey integrates multiple common repeat masking tools such as RepeatMasker[22], which maps repetitive elements from a database of known repeat sequences, and RepeatModeler[23], which identifies repeats de novo. It also uses multiple tools such as cd-hit-est[24], LTR_finder[25], rcMergeRepeats[23] and custom scripts to identify, annotate, filter and aggregate repeat regions genome wide. Earl Grey produces figure-quality summaries of a genome's TE landscape in conjunction with repeats annotated in general feature format (GFF), which are required for downstream analysis (GFF and other file formats are explained in Supplementary Methods: Genome file formats). Earl Grey relies on databases of repeat elements, such as Dfam[26], that are used to identify repeats in the target genome. The user can specify what clade of species they are working with, which indicates which repeat database Earl Grey should use (Box 2).

Earl Grey directly outputs a soft-masked genome (a widely used convention indicating repetitive DNA sequences by using lowercase characters in the FASTA file), as well as coordinates indicating repeat identity in a GFF or browser extensible data (BED) file. The GFF and BED files also indicate which repeat families the repetitive regions belong to, including TE class and family, repetitive non-coding RNA (e.g., small nucleolar RNA (snoRNA)) and simple repeats (e.g., low-complexity sequences consisting of one, or a few, bases consecutively repeated). Some genome annotation tools that are not used in this Tutorial require a hard-masked genome (repetitive element sequences converted to 'N' characters). Generally, genome annotation tools specify which type of masking is required in their documentation. Users who want to customize their repeat masking (e.g., refrain from masking snoRNA or hard mask) can customize the BED file output by Earl Grey and manually mask their genome assembly by using BEDTools[27]. Automated TE annotation relies on a curated database, like the DFam database[28]. Manual annotations may increase the resolution of species-specific TE families or TE families residing in complex, repetitive heterochromatic regions (e.g., specific centromeric alpha-satellites)[10], but performing such annotations is beyond the scope of this Tutorial.

Step 2: generating protein-coding gene models

Gene models are hypotheses about the locations of genes in the genome and their features (e.g., mRNA, exons and introns). These are supported by various evidence sources, some of which are easily accessible 'standard' sources (e.g., protein sequence databases and genome annotations from well-studied reference species like human and house mouse), and others are less accessible ('premium') sources that are specific to the target species being annotated (e.g., RNA-seq from the target species and genome annotations from a close relative; Fig. 1). Gene models can represent true positives (correctly located gene), false positives (e.g., a random ORF-like sequence) and false negatives (e.g., a real gene that was missed in the annotation process)[29] (Fig. 2). To reduce errors, it is important to use high-quality evidence for the existence of genes and annotation tools that perform well (Supplementary Table 2).

In this Tutorial, we describe and integrate two complementary approaches to generate high-quality gene models: homology-based annotation and transcriptome- and protein-guided annotation (Fig. 1). Broadly, homology-based genome annotation assumes that thousands of gene models will be shared between a reference species (e.g., house mouse) and a target species (e.g., woodchuck) at the level of DNA sequence similarity and gene structure (e.g., number of exons). The proportion of successfully mapped genes from the reference species to the target species depends on how closely related the species are and the quality of the reference genome's assembly and annotation. In contrast, transcriptome- and protein-guided genome annotation assumes that the location of uniquely mapped paired-end RNA-seq data represents an expressed region of the genome and is therefore a candidate for a gene model.

Resulting annotations vary depending on the quality of (i) the genome sequence being annotated, (ii) the evidence provided to inform the annotation (e.g., RNA-seq and homology) and (iii) the bioinformatic tool applied. Therefore, gene model selection, the process of identifying, merging and curating the best gene models from a set of candidate gene models, is crucial for generating a high-quality final genome annotation (Step 3; Box 3).

Homology-based genome annotation. Homology-based genome annotation involves the transfer of gene model information from a reference species (e.g., house mouse) to a target species of interest. Most gene structures and sequences are conserved across related species, making homologous alignments from a reference species

https://doi.org/10.1038/s41596-025-01301-1

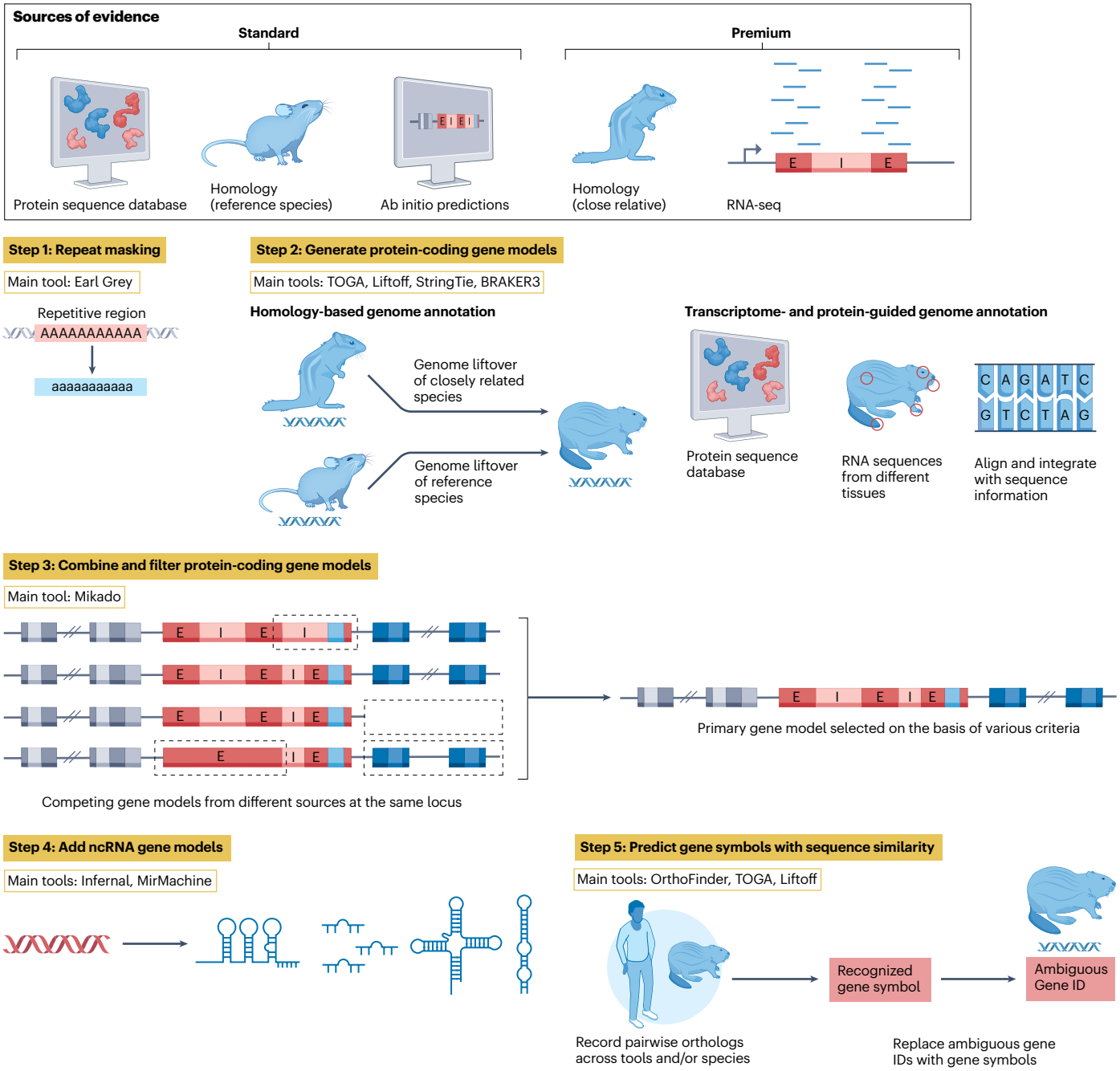

Fig. 1 | Genome annotation workflow. This genome annotation Tutorial identifies and classifies an animal genome assembly's repetitive and gene elements. We expect up to five sources of evidence when annotating a genome with the Tutorial. 'Standard' sources of evidence rely on the DNA sequence of the genome being annotated and publicly available resources. These include public databases, reference genome assemblies and annotations (e.g., house mouse and human) and the genome sequence being annotated. 'Premium' sources of evidence are not guaranteed to exist for every species but will improve genome annotations if used. These sources include transcriptome- and protein-sequence data and high-quality genome assemblies and annotations for closely related species (e.g., the same genus). The Tutorial is split into five steps. Step 1 uses the Earl Grey pipeline to identify and mask common repeats and transposable elements in the genome to reduce noise for subsequent steps. Step 2 uses homology between a reference and target genome, transcriptome- and protein-sequence data and analysis of the DNA sequence of the target assembly to predict protein-coding and long non-coding RNA gene models. We use four tools, Liftoff, TOGA, BRAKER3, and StringTie, to predict these gene models. Step 3 uses Mikado to evaluate the gene models from each source and integrate these gene models into complete mRNA and long non-coding RNA transcripts. Step 4 identifies candidate short ncRNA genes by aligning the target genome to the RNA families database and then evaluating these candidate sequences' predicted secondary structures against known ncRNA secondary structures. Finally, Step 5 uses OrthoFinder, Liftoff and TOGA to assign gene symbols to the gene models identified in Steps 2 and 3 by comparing transcript sequences, homology and gene order to one (or multiple) reference species. ID, identifier.

with high-quality gene structures an accurate and computationally efficient method to annotate the target species. One consideration of using homology-based annotations alone is that they cannot find species-specific gene models. In addition, any errors existing in the reference annotation will propagate to the new, target annotation. Errors are inevitable and exist in all annotations. Even the human

BOX 1

Computational requirements and technical challenges

Genome annotation is a computationally intensive process that requires significant computational time and resources. The requirements of the pipeline are dictated by the minimum requirements of the most intensive tools that we recommend. The most time-consuming and computationally intensive tools are Earl Grey[21], BRAKER3[9] and TOGA[17].

A high-performance computing cluster is recommended to run the Tutorial workflow in its entirety. Because different genomes require varying computational resources based on their characteristics (e.g., genome size and repeat content) and the amount of evidence used as input for the pipeline, the computational requirements are inherently unpredictable. Generally, each step of the Tutorial has required a maximum of 150 h of run time and 64 GB of random access memory (RAM). In our experience, StringTie, Liftoff and Mikado are desktop friendly, and it is technically possible to run all tools on a desktop computer (although TOGA is limited to small or partial genomes due to inherent workflow management), but we do not recommend it. The high-performance computing cluster system must be compatible with Singularity or Apptainer[75] container technologies, and we strongly recommend compatibility with the conda package and environment manager (https://conda.io). It is possible to use the Tutorial workflow to build protein-coding and lncRNA gene models without conda by using other methods of installation, but it takes a lot of manual work and is more challenging to control software tool dependencies.

Instructions for installing all tools and dependencies are available on GitHub (https://github.com/BaderLab/GenAnT), and here we offer three computational strategies for genome annotation. The first is a step-by-step Tutorial that walks a user through the genome annotation process line by line. This approach expects an intermediate level of bioinformatic experience, with some experience of bash scripting and R. Second, we provide a shell script that exports user-provided arguments into an environmental variable before running the Tutorial with no flow control (used for testing the pipeline). Third, we offer a Snakemake pipeline that allows the Tutorial to be run through a single configuration file[76]. This requires the user to be comfortable generating a YAML file and has a steeper learning curve than running the shell script, but it is the most computationally efficient and least prone to human error.

Common challenges that the user may experience are described in more detail in the documentation on GitHub, along with potential solutions. Tool-specific challenges are common, and much of the GitHub repository associated with the Tutorial is dedicated to facilitating the use of tools that are challenging to install and get working. For example, TOGA and BRAKER3 have specific formatting requirements for their input files that will otherwise cause the tools to crash. On GitHub, we provide scripts for editing input files to try to prevent such errors from happening. In addition, some tools produce intermediate files that need to be deleted before rerunning the tool if the tool crashes. We have specified in the Tutorial when this step is necessary and what files need to be deleted.

genome annotation is being iteratively improved upon[30]. Errors can be mitigated by selecting more-closely related reference species with high-quality genome assemblies and annotations[10].

Broadly, we recommend using reference genomes generated with long-read technologies with chromosomal-level resolution and a quality value score >40, indicating a low error rate of base calls[31]. These technologies and statistics should be reported in any assembly release. Currently, genomes assembled by consortia such as the Vertebrate Genome Project reliably surpass these standards. Genome assemblies using Pacbio HiFi reads (or equivalent) also reliably exceed these standards[32].

High-quality annotations can be assumed if such a genome is annotated by an annotation hub (i.e., RefSeq, Ensembl and MANE), although annotation completeness will still vary on the basis of available evidence for the species (e.g., RNA-seq from multiple tissues). We recommend that the user search these databases for a few of the most-closely related species, comparing these genome statistics and selecting the assembly and annotation (or assemblies and annotations from multiple references) with the most favorable statistics as a reference for gene liftover.

Two homology-based tools that often create high-quality annotations are Liftoff[16] and the Tool to infer Orthologs from Genome Alignments (TOGA)[17]. Liftoff is a gene liftover tool that aligns gene sequences from the reference genome to the target genome by using a single line of Unix code, making it quick and easy to use (Supplementary Table 2 and Fig. 3a). It uses minimap2[33] to align the genes from the reference genome to the target genome with high accuracy and relatively low computational resources. The alignment algorithms in minimap2[33] are optimized to work with sequences of the same or closely related species, making Liftoff ideal when the reference species is in the same genus as the target species. Liftoff takes a FASTA file and GFF or gene transfer format (GTF) file from a reference species and the FASTA file from the target species and creates a GFF/GTF output file for the target based on the reference annotations (Supplementary Table 2). It also provides the user with a list of unmapped genes, which may indicate alignment challenges. Because Liftoff is quick and easy to use, the user can generate annotations from multiple reference species and compare the resulting annotation quality to pick the best result (Box 3).

The second homology-based annotation tool that we recommend is TOGA, which can accurately annotate genes across vertebrates with higher rates of divergence (e.g., house mouse to naked mole-rat, ~70 million years diverged)[34,35]. TOGA can annotate more-divergent species because it relies on a chain file, which stores pairwise alignments connecting the reference and target species that allows for gaps in both sequences. It also relies on an exon-specific aligner, CESAR[36] to annotate exons, which aids in finding alignments between more divergent sequences compared to Liftoff (Supplementary Table 2). TOGA also uses syntenic information (i.e., gene order) to infer orthology. Generating the various files for TOGA is more bioinformatically involved and computationally expensive than using Liftoff; however, all processing can be done by using scripts provided by the Comparative Genomics Toolkit[37] (https://github.com/ComparativeGenomicsToolkit). In summary, both Liftoff and TOGA confer distinct advantages and can be used to identify distinct gene models that are combined in Step 3.

Transcriptome- and protein-guided genome annotation. Another way to annotate genomes is to use RNA- and/or protein-sequence alignment evidence to inform gene models. Alignment-based methods work by aligning RNA or protein sequences to the genome to determine the location of transcribed and/or protein-coding genes. The specific tools used to perform alignment-based annotation depend on the sequencing data available.

BOX 2

Parameter selection and adapting the Tutorial workflow to different clades

All software tools that we recommend require input parameters to be set, defined as required or optional arguments that influence the output of the tool. Because of the many parameters available, there are many possible parameter combinations. The Tutorial workflow was designed to minimize parameter selection in a number of ways. First, we prioritized bioinformatic tools that score highly across eukaryotic species in recent benchmarks[11], decreasing the likelihood that a method needs to be replaced until it is updated or outperformed by a new tool. Second, we prioritized tools that internally perform training or alignment steps (e.g., BRAKER3, TOGA and OrthoFinder) or contain pre-configured files and databases for a wide array of clades (e.g., Earl Grey, Mikado, and MirMachine). The parameters listed in the Github Tutorial 'config.yaml' file are sufficient to annotate vertebrate and invertebrate animal genomes. We have briefly categorized classes of parameters to consider below.

First, parameters designating which data are input are required for genome annotation. For example, TOGA and Liftoff require the genome assembly and annotations of the reference species to be specified, and StringTie and BRAKER3 require RNA-seq data to be specified. StringTie and BRAKER3 have additional parameters designating the 'type' of RNA-seq included (e.g., no RNA-seq versus unstranded RNA-seq versus stranded RNA-seq versus Iso-Seq). The scripts in the GitHub for our Tutorial will automatically pick the most appropriate version of each tool given the evidence provided. Therefore, if the user accidentally does not include their RNA-seq data in the 'config.yaml' file, the script will generate annotations in non-RNA-seq mode.

Second, parameters designating the clade of the target species need to be set for genome annotation. When running the Tutorial workflow, Earl Grey will annotate repeats on the basis of clade-specific repeat libraries, BRAKER3 will use clade-specific protein databases, MirMachine will use clade-specific pre-computed covariance matrices and Mikado will use clade-specific scoring files for transcript assembly. These parameters usually need to be input by the user, but some tools may default to a 'Eukaryote' clade when they are not specified. Specifying a clade will improve gene models.

Finally, parameters within each annotation tool can be subcategorized into: (i) alignment parameters, which adjust the stringency of RNA-seq, genome-to-genome and genome-to-database alignments; (ii) definitional parameters, which provide cutoffs for what should be considered a gene (e.g., a lncRNA is defined as having an ORF of <100 bp); and (iii) algorithm fine-tuning (e.g., to make StringTie-Iso-Seq gene models weigh more than StringTie-RNA-seq in Mikado). Manually tuning these parameters may improve genome annotations in specific cases, in which case we suggest generating an annotation by using default parameters in parallel as a positive control to see if the fine-tuning improves the results. The quality of each result can be compared to help determine how to optimize parameters for the best annotation (see Box 3). Details on how to adjust these parameters are outlined in the documentation on GitHub and the documentation for the individual tools.

If the user has access to RNA-seq data with the minimum requirements of 100-bp read length, a paired-end sequencing protocol and high sequencing depth (e.g., 50 million reads for most tissues and 100 million reads for tissues with high transcript diversity like brain and gonads)[38,39], then these data can be used to generate gene models through RNA-seq alignment (e.g., by using HISAT2 (ref. 40)) followed by a gene model caller (e.g., StringTie[41])[11]. Long-read RNA-seq (e.g., Iso-Seq) is becoming more commonplace and can capture entire transcripts (including intron-exon structure, transcript direction and poly(A) tail information at high resolution) within a single read[42]. Including RNA-seq/Iso-Seq data from a diverse range of tissues also helps minimize false negatives, because gene expression profiles vary across tissues. Tissue-specific transcripts can be captured, and certain tissues, such as the brain, lungs and gonads, are particularly valuable because they exhibit a broad range of gene expression, thereby improving the completeness of the annotation (for advice on how to combine RNA-seq data across various tissues, see Supplementary Methods: Combining RNA-seq derived transcripts).

One tool that integrates RNA-seq alignment information with protein sequence data and ab initio gene prediction (i.e., by using a trained algorithm to assign gene features, like the start and stop codon, from the genome sequence) is the most recent iteration of BRAKER3[9] (Supplementary Table 2 and Fig. 3a). The RNA sequences come from the species being annotated, whereas the protein sequences are typically from an online database of homologous sequences, like OrthoDb[43]. Internally, BRAKER3 uses HISAT2 (ref. 40) to align the short paired-end RNA-seq reads to the genome, StringTie[41] to create candidate gene models from these alignments and ProtHint[44] to predict coding sequence (CDS) regions by using these protein alignments (Supplementary Table 2). BRAKER3 is also compatible with stranded RNA-seq and long-read data as an alternative to traditional RNA-seq. These data are then used as 'hints' (i.e., estimations of CDS region and intron placements) when generating ab initio gene models with GeneMark-ETP[45] and Augustus[46]. BRAKER3 can also identify tRNAs, snoRNAs and untranslated regions (UTRs)[14], and if RNA-seq data is not available for the species (e.g., DNA derived from a sample with no RNA extraction possible, such as a wild-derived tail-clipping), then BRAKER3-protein (i.e., no RNA-seq) can generate useful gene models. In this Tutorial, we use both StringTie and BRAKER3 before combining results with the homology-based genome annotations and filtering gene models as described in Step 3 (Fig. 1).

Step 3: combining and filtering gene models

Completing steps one and two yields gene models from multiple homology-based and transcriptome- and protein-guided annotations. Many gene models will be identified across all annotations; however, some gene models will be method specific (Fig. 3c). Mikado is a tool that can combine, evaluate and filter gene models across multiple annotations in a way that mimics manual assembly curation. Mikado takes different GFF files as input and outputs a filtered GFF file that is often more accurate than any of the input annotation or evidence files on their own (Supplementary Table 2 and Fig. 3b).

Mikado functions by using information from external tools and internal filtering systems to identify the most likely gene models (Supplementary Table 2). For instance, Mikado filters out chimeric, fragmented or short transcripts with disrupted coding sequences[15]. It also scores gene models on the basis of their likelihood of being a real gene by using BLAST+[47] to compare predicted gene models to an existing

https://doi.org/10.1038/s41596-025-01301-1

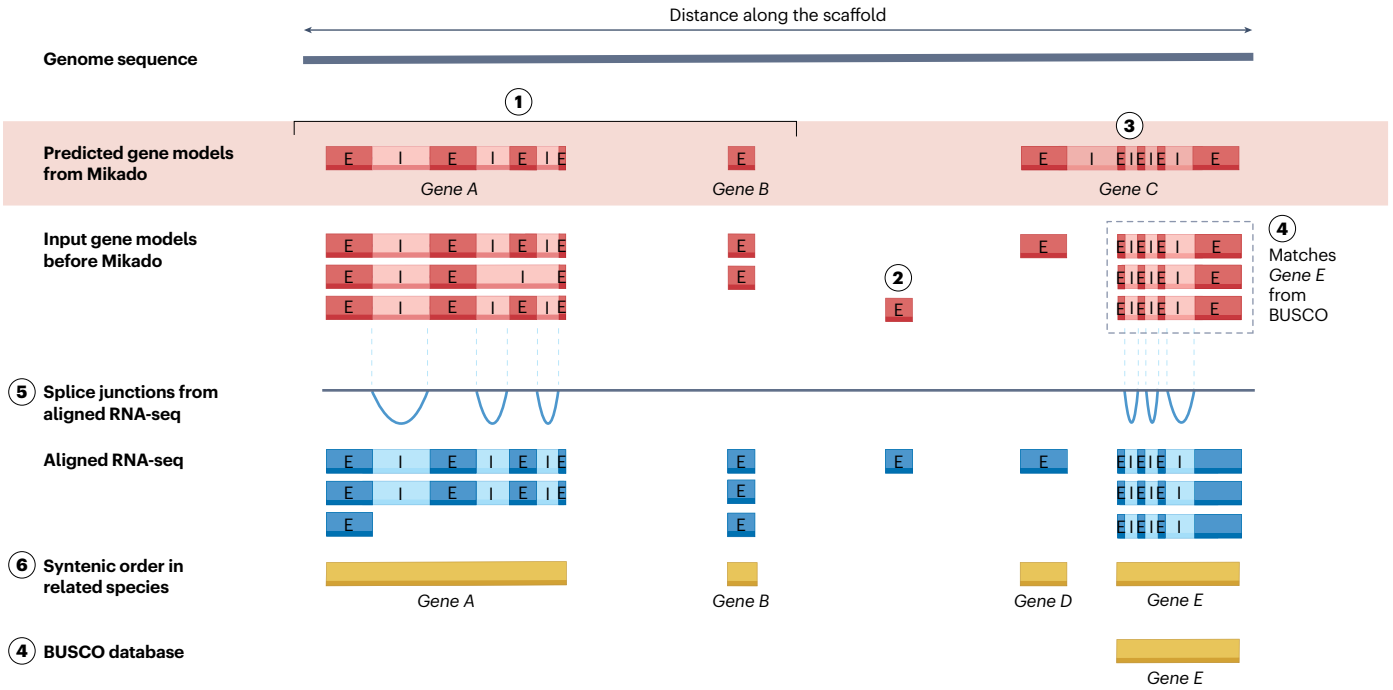

Fig. 2 | Visualizing evidence in a genome browser. Illustration of using various sources of evidence to evaluate misannotated '*Gene C*', which is a falsely annotated fusion of '*Gene D*' and '*Gene E*'. (1) Two genes that show evidence of high-quality annotations, '*Gene A*' and '*Gene B*'. The predicted gene model is a combination of each source of evidence before Mikado integration. Intron (translucent rectangle) and exon (rectangle) boundaries match splice junctions from RNA-seq data. The gene order and length also match the gene order and length found in a related species. (2) A gene model found in only one annotation method, with some evidence from RNA-seq data. Model (2) was filtered and not included by Mikado. (3) A falsely annotated intron that incorrectly connects '*Gene D*' and '*Gene E*' to form '*Gene C*'. The false intron in (3) is not found in any source of evidence before Mikado, there is no evidence of splice junctions in RNA-seq data and the two connected models represent two genes in a related species, namely '*Gene D*' and '*Gene E*'. (4) '*Gene E*' is a universal single-copy ortholog (BUSCO) gene. Overall, the evidence suggests that there is a low likelihood that '*Gene E*' has been fused to surrounding genes. (5) RNA-seq alignment data, which contain the most-robust evidence for exon junctions and intron-exon relationships in animal genome annotations. (6) The gene size and gene order of the same cluster of genes in a related species, which aids in annotating gene symbols.

protein database and therefore favors conserved genes; it also uses an internal scoring system to pick the best gene models from all sources of evidence and outputs a single GFF file (Supplementary Methods: Explanation of the Mikado scoring file (e.g. mammalian.yaml)). This scoring system can be adapted by user-provided parameters to preserve certain characteristics of the input gene models.

Integrated genome annotations are expected to be more comprehensive than those generated by any single method; however, automated methods evaluating hundreds of thousands of features will have some false-positives and false-negatives. For example, because Mikado prefers the longest gene model that passes its filters, a long gene model derived from a homologous sequence may incorrectly replace an alternative gene model that better matches a gene in the Benchmarking Universal Single Copy Orthologs (BUSCO)[48] database (for a description of BUSCO, see Box 3). The resulting annotation can still be improved by looking at additional evidence sources not used by Mikado, such as missing BUSCO genes that were present in any of the input assemblies but dropped by Mikado, manually evaluating gene order along a sequence (i.e., 'synteny') in a genome browser or visualizing functional genomics data that have been aligned to the annotated genome (Fig. 2).

Step 4: annotating non-coding RNA genes

NcRNA genes encode a diverse array of functional RNA molecules with various gene lengths and secondary structures. In this Tutorial, we describe methods to annotate long non-coding RNAs (lncRNAs), and various short non-coding RNA genes (e.g., small nuclear RNAs, snoRNAs, tRNAs, rRNAs, miRNAs).

LncRNAs are RNA molecules that are longer than 200 nt and do not contain an ORF longer than 100 aa[49]. LncRNAs sometimes have introns (i.e., sequences removed by splicing)[50] and may generate small peptides that play regulatory functions in the cell[49]. As such, it can be challenging to strictly delineate lncRNAs from mRNA[50]. This phenomenon translates to lncRNA being identified as gene models that are greater than 200 nt and that fail to be classified as mRNA. In our pipeline, lncRNA and mRNA are simultaneously classified by using Mikado (during Step 3).

Short non-coding RNA genes contain classes of non-coding molecules that serve a diverse array of regulatory functions within the cell. These ncRNA gene classes generate relatively short transcripts and, crucially, contain conserved sequences and secondary structures across species. These conserved features can be found in the RNA families database (RFam), an open-access database that stores alignments, secondary structures and covariance matrices of >4,000 ncRNA familes[28].

Short ncRNAs are annotated in two steps. The first step is called 'seeding', in which regions of the assembly are identified as ncRNA candidates by querying known gene models, repeat annotations and RNA families (by a BLAST search against the RFam database). The second step is the ncRNA evaluation step, in which potential ncRNA genes are classified by using INFERence of RNA ALignment (Infernal)[51] (Supplementary Table 2). This Tutorial uses MirMachine[52], which relies on Infernal but has clade-specific miRNA-specific secondary structures trained from MirGeneDB[53] (Supplementary Table 2 and Box 2). MiRNA annotations inferred by MirMachine provide additional gene models that cannot be detected with RFam alone.

BOX 3

Testing annotation quality

Genome annotation quality across species improves with genomic data quality, availability and tool development. However, it is important to recognize that no genome annotation will be perfect. Therefore, a user should aim for the highest quality genome annotation possible while considering the limitations of data quality and the human effort required for refinement. The quality of all annotated features should be rigorously assessed for each annotation generated in Step 2 of our Tutorial. This can tell the user if the data that they are using to generate the annotation is of sufficient quality and/or how well each tool is working with the data provided.

A commonly used way to assess the completeness and quality of the annotation is to compare the gene models found in the target genome to the BUSCO database[48], a tool that serves both as a database and statistical software. The BUSCO database consists of curated gene sequences from single-copy orthologs for all domains of life stored in the OrthoDB database[43]. When used in protein mode, the BUSCO software returns statistics indicating if the expected protein sequences are found, fragmented or missing (Fig. 3b). BUSCO scores are compared across annotations as a judge of quality, with higher BUSCO scores indicating higher-quality annotations. Generating a BUSCO score with the entire genome as the target indicates the maximum BUSCO score possible for that assembly.

It is also helpful to analyze feature statistics of a particular annotation (e.g., average exons per transcript, number of monoexonic transcripts and gene lengths), because outliers may indicate that there are inaccuracies. For instance, if an RNA-sequencing alignment–based annotation has a large number of monoexonic transcripts compared to a homology-based annotation, this suggests that the former annotation may be fragmented into artificially small transcripts. Mikado comes with a command that outputs a text file of summary statistics.

Different GFF files mapping to the same genome assembly can be compared with GffCompare[77]. Briefly, GffCompare inputs a 'query' GFF and a 'reference' GFF and outputs a parseable text file ('.stats') describing how well the base pairs, exons, introns and transcripts match each other. It can be valuable if the researcher has a set of experimentally validated or manually curated gene models for their species or when multiple GFF files exist for a species from one or multiple annotation efforts.

Finally, genomes contain collinear regions called syntenic blocks that are conserved across large evolutionary time spans[78]. In the context of a reference and target species for genome annotation (e.g., house mouse and woodchuck), these syntenic blocks typically contain a large number of genes in both species, and the orientation of these genes is often the same (Fig. 2). Synteny can also be used to manually or systematically identify missing annotations or misassembly by comparing genome browser snapshots between the reference and target species[17,79]. Although low throughput, the importance of manually inspecting genome annotations within the genome browser cannot be overstated. Genomes, annotations and functional data can be loaded into the Interactive Genome Viewer (IGV), a point-and-click program to support manual review (Fig. 2 and Supplementary Methods: Viewing annotations on IGV).

Step 5: sequence similarity–based transfer of gene symbols

Decades of research in model organisms have identified biological, molecular and cellular functions for many protein-coding and non-coding gene sequences in animal genomes. These gene functions are characterized by a gene symbol (e.g., estrogen receptor 1, *ESR1*). Identifying which sequences are predicted orthologs (i.e., derived from a single ancestral sequence) between species of interest and the most closely related model organism allows these gene functions and gene symbols to be applied to the species of interest. Assigning gene symbols is challenging because most genes in animal genomes originated from another gene (e.g., tandem duplication, gene fusion or translocation)[54], meaning that many genes have at least one paralogous gene with high sequence similarity in exons.

Both Liftoff and TOGA annotate the target species' gene structures and assign reference gene symbols to the target with a high rate of agreement with the gene symbols found in Ensembl annotations[17,55]. Therefore, these tools can be used to predict gene symbols for the final, integrated annotation (Supplementary Table 2). We transfer gene models by matching exons derived from TOGA and Liftoff to the final gene models, before transferring the gene symbol to the Mikado-filtered gene identifier data file column.

In addition, OrthoFinder[56], a tool that maps sequence-similarity relationships between proteins across two or more species on the basis of their sequences, can be used to identify predicted orthologs. OrthoFinder builds gene trees, considers gene duplication events, is considered to be one of the most accurate ortholog inference methods[57] and was used for gene naming in the DNA zoo annotation project[58]. OrthoFinder outputs lists of protein-protein sequence-similarity relationships that can be used to infer gene-gene relationships. An alternative to OrthoFinder is the Orthologous MAtrix (OMA) database and tool, which similarly maps orthologous relationships between species[59]. OMA uses a more-sensitive alignment algorithm, which may help discern some one-to-one orthologs missed by OrthoFinder; however, it is more challenging and computationally intensive to run. We have provided documentation on how to run OrthoFinder and add orthologous relationship output from any tool (including OMA) on the Tutorial's GitHub site.

Lastly, protein family and domain information can be added to coding sequences by using InterProScan[60]. These additional annotations provide evidence for gene function in genes that could not be identified with a unique gene symbol (e.g., unnamed gene-X has a zinc-finger domain)[8,60,61].

Each of the above methods will assign gene symbols independently, and most gene symbols should agree across methods and species. If gene symbols appear to disagree between methods, they may be aliases for the same gene (e.g., *ABC2* versus *ABCbeta*). This is especially noticeable if a species other than human or house mouse has been used for gene symbol prediction, because genes that have an easily recognizable gene symbol in humans may have a systematically assigned gene symbol in less-well-studied species (e.g., *IZUMO1* in humans is *LOC101976381* in the 13-lined ground squirrel). Other cases of gene symbol disagreement may occur if each method assigns a different member of the same gene family to a gene (e.g., *ABC1* versus *ABC2*). Occasionally, each method may call different one-to-one orthologs for the same gene (e.g., *ABC1* versus *DEF7*). These instances could be manually resolved by comparing syntenic gene orders between the target species and a reference species in that region (Fig. 2). Resolving these gene symbols is inherently annotation specific and would be performed downstream of this Tutorial.

https://doi.org/10.1038/s41596-025-01301-1

Fig. 3 | Tools used to generate protein-coding gene models. a, The four tools that we use to identify gene models in this Tutorial are: Liftoff, which transfers protein-coding and non-protein-coding models from between two closely related species (i.e., the same genus); TOGA, which transfers protein-coding models between two more distantly related species; StringTie, which builds gene models from RNA-seq alignment information; and BRAKER3, which uses RNA-seq and protein information to predict exon and intron location before using ab initio gene prediction to generate gene models. **b**, The distribution of universal single-copy ortholog (BUSCO) genes captured in each method. Gene models from each method are selected and integrated by using the Mikado gene selector, which should have more complete BUSCO scores than each individual method. **c**, The distribution of the number of gene models in the final genome annotation that came from each method after being filtered with Mikado. In this example, many gene models are derived from Liftoff and StringTie, which represent annotations from premium sources of evidence, namely homology of a close relative and RNA-seq data of multiple tissues, respectively.

Summary and future directions

In this Tutorial, we present a workflow consisting of various tools that perform the different components of the genome annotation process and integrate the resulting gene models. We provide descriptions of key file types and methods involved in the genome annotation process, as well as a detailed, practical guide on how to use and integrate these methods, assuming that the user has an intermediate level of bioinformatics experience. This pipeline was originally designed for mammalian genomes but can be effective across diverse animal species.

The aim of this Tutorial is to guide users through the genome annotation process on the basis of what is feasible and recommended with current technologies. Although we are confident that our workflow produces high-quality annotations based on current standards, limitations exist and should be recognized by the user; notably, we recommend specific genome annotation tools based on limited benchmarking literature[11] and established practices in the field. We found that TOGA, BRAKER3, StringTie[11] and Liftoff (Z.A.C. and D.J.S., unpublished data) were consistently top performers across various metrics, including BUSCO score, CDS length and false-positive rate when compared to existing annotations from Ensembl and RefSeq. There are also several new tools that take advantage of miniprot[62–65], which may result in improved annotations, and could therefore be incorporated into future workflows. Other recommended tools that do not directly contribute to the annotation of protein-coding genes (e.g., ncRNA gene annotation

and repeat masking) were chosen because they are one of the few that could perform a specific task.

Furthermore, although our Tutorial discusses annotating repetitive and genic features, it does not cover annotating *cis*-regulatory elements such as promoters, enhancers and repressed elements. Regulatory elements are species-, tissue-, developmental stage– and disease-specific and can be measured with experiments that profile the epigenome (e.g., ChIP-seq)[66]. Without performing such experiments, there is not yet an effective way to annotate these features. ChromHMM is the most popular tool to build these chromatin states[67]. This may change as epigenetic experiments become more scalable, but annotating these features is currently not typical of genome annotation projects.

As technology improves, genome annotation tools will perform more accurately and efficiently, and homology-based genome annotation will probably continue to be optimized in response to the influx of reference-grade or telomere-to-telomere de novo genome assemblies[32,68]. With these technological improvements will come the discovery of more clade-specific genes, isoform-level transcript resolution[69] and new feature types (e.g., promoter-enhancer pairs and methylation profiles[66,70]). These improvements will also help annotate activity in complex immune-system gene families (e.g., the T-cell receptor, immunoglobin genes and major histocompatibility complex genes[71]). There are also several less-commonly implemented, but promising innovations in genome annotation, such as using deep-learning models to annotate genes, TEs and splice sites[72] and using large language models in gene and regulatory feature annotation[73,74], that may become more prevalent in the future.

Although annotation tools and algorithms will continue to improve, the fundamental process of combining and filtering various annotations—along with rigorous quality assessment and manual refinement—will remain critical. For example, deep-learning models rely on high-confidence, well-curated annotations for training data, meaning that the iterative refinement of genome annotations using high-quality RNA-seq, protein and orthology-based evidence will remain essential for advancing automated annotation accuracy. As such, our Tutorial provides the genomics community with the infrastructure to generate high-quality genome annotations in individual laboratories now, while helping build the foundation for future high-throughput genome annotation efforts.

Data availability

Example data for the Tutorial are available at https://zenodo.org/records/14962941.

Code availability

Code containing Linux and R scripts to (i) guide the user step-by-step through the genome annotation process and (ii) provide a streamlined genome annotation workflow that includes a small example are available at https://github.com/BaderLab/GenAnT.

References

1. Eckalbar, W. L. et al. Transcriptomic and epigenomic characterization of the developing bat wing. *Nat. Genet.* **48**, 528–536 (2016).
2. Sulak, M. et al. *TP53* copy number expansion is associated with the evolution of increased body size and an enhanced DNA damage response in elephants. *eLife* **5**, e11994 (2016).
3. Moreno, J. A. et al. *Emx2* underlies the development and evolution of marsupial gliding membranes. *Nature* **629**, 127–135 (2024).
4. Sohn, J.-I. & Nam, J.-W. The present and future of de novo whole-genome assembly. *Brief. Bioinforma.* **19**, 23–40 (2018).
5. Cheng, H., Concepcion, G. T., Feng, X., Zhang, H. & Li, H. Haplotype-resolved de novo assembly using phased assembly graphs with hifiasm. *Nat. Methods* **18**, 170–175 (2021).
6. Larivière, D. et al. Scalable, accessible and reproducible reference genome assembly and evaluation in Galaxy. *Nat. Biotechnol.* **42**, 367–370 (2024).
7. Wang, B. et al. Long and accurate: how hifi sequencing is transforming genomics. *Genomics Proteom. Bioinforma.* **23**, qzaf003 (2025).
8. Ejigu, G. F. & Jung, J. Review on the computational genome annotation of sequences obtained by next-generation sequencing. *Biol. (Basel)* **9**, 295 (2020).
9. Gabriel, L. et al. BRAKER3: fully automated genome annotation using RNA-seq and protein evidence with GeneMark-ETP, AUGUSTUS, and TSEBRA. *Genome Res.* **34**, 769–777 (2024).
10. Yandell, M. & Ence, D. A beginner's guide to eukaryotic genome annotation. *Nat. Rev. Genet.* **13**, 329–342 (2012).
11. Freedman, A. H. & Sackton, T. B. Building better genome annotations across the tree of life. *Genome Res.* **35**, 1261–1276 (2025).
12. Gupta, P. K. Earth Biogenome Project: present status and future plans. *Trends Genet.* **38**, 811–820 (2022).
13. Rhie, A. et al. Towards complete and error-free genome assemblies of all vertebrate species. *Nature* **592**, 737–746 (2021).
14. Brůna, T., Hoff, K. J., Lomsadze, A., Stanke, M. & Borodovsky, M. BRAKER2: automatic eukaryotic genome annotation with GeneMark-EP+ and AUGUSTUS supported by a protein database. *NAR Genom. Bioinform.* **3**, lqaa108 (2021).
15. Venturini, L., Caim, S., Kaithakottil, G. G., Mapleson, D. L. & Swarbreck, D. Leveraging multiple transcriptome assembly methods for improved gene structure annotation. *Gigascience* **7**, giy093 (2018).
16. Shumate, A. & Salzberg, S. L. Liftoff: accurate mapping of gene annotations. *Bioinformatics* **37**, 1639–1643 (2021).
17. Kirilenko, B. M. et al. Integrating gene annotation with orthology inference at scale. *Science* **380**, eabn3107 (2023).
18. Ekblom, R. & Wolf, J. B. W. A field guide to whole-genome sequencing, assembly and annotation. *Evol. Appl.* **7**, 1026–1042 (2014).
19. Dominguez Del Angel, V. et al. Ten steps to get started in Genome Assembly and Annotation. *F1000Res.* **7**, ELIXIR-148 (2018).
20. Fiddes, I. T. et al. Comparative Annotation Toolkit (CAT)—simultaneous clade and personal genome annotation. *Genome Res.* **28**, 1029–1038 (2018).
21. Baril, T., Galbraith, J. & Hayward, A. Earl Grey: a fully automated user-friendly transposable element annotation and analysis pipeline. *Mol. Biol. Evol.* **41**, msae068 (2024).
22. Tarailo-Graovac, M. & Chen, N. Using RepeatMasker to identify repetitive elements in genomic sequences. *Curr. Protoc. Bioinforma.* **Chapter 4**, 4.10.1–4.10.14 (2009).
23. Flynn, J. M. et al. RepeatModeler2 for automated genomic discovery of transposable element families. *Proc. Natl Acad. Sci. USA* **117**, 9451–9457 (2020).
24. Li, W. & Godzik, A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics* **22**, 1658–1659 (2006).
25. Xu, Z. & Wang, H. LTR_FINDER: an efficient tool for the prediction of full-length LTR retrotransposons. *Nucleic Acids Res.* **35**, W265–W268 (2007).
26. Storer, J., Hubley, R., Rosen, J., Wheeler, T. J. & Smit, A. F. The Dfam community resource of transposable element families, sequence models, and genome annotations. *Mob. DNA* **12**, 2 (2021).
27. Quinlan, A. R. BEDTools: the Swiss-Army tool for genome feature analysis. *Curr. Protoc. Bioinforma.* **47**, 11.12.1-34 (2014).
28. Kalvari, I. et al. Rfam 14: expanded coverage of metagenomic, viral and microRNA families. *Nucleic Acids Res.* **49**, D192–D200 (2021).

29. Hoff, K. J., Lomsadze, A., Borodovsky, M. & Stanke, M. Whole-genome annotation with BRAKER. *Methods Mol. Biol.* **1962**, 65–95 (2019).
30. Nurk, S. et al. The complete sequence of a human genome. *Science* **376**, 44–53 (2022).
31. Rhie, A., Walenz, B. P., Koren, S. & Phillippy, A. M. Merqury: reference-free quality, completeness, and phasing assessment for genome assemblies. *Genome Biol.* **21**, 245 (2020).
32. Stanojevic, D., Lin, D., Florez De Sessions, P. & Sikic, M. Telomere-to-telomere phased genome assembly using error-corrected Simplex nanopore reads. Preprint at https://www.biorxiv.org/content/10.1101/2024.05.18.594796v1 (2024).
33. Li, H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* **34**, 3094–3100 (2018).
34. Zoonomia Consortium. A comparative genomics multitool for scientific discovery and conservation. *Nature* **587**, 240–245 (2020).
35. Lewis, K. N. et al. Unraveling the message: insights into comparative genomics of the naked mole-rat. *Mamm. Genome* **27**, 259–278 (2016).
36. Sharma, V., Elghafari, A. & Hiller, M. Coding exon-structure aware realigner (CESAR) utilizes genome alignments for accurate comparative gene annotation. *Nucleic Acids Res.* **44**, e103 (2016).
37. Armstrong, J. et al. Progressive Cactus is a multiple-genome aligner for the thousand-genome era. *Nature* **587**, 246–251 (2020).
38. Conesa, A. et al. A survey of best practices for RNA-seq data analysis. *Genome Biol.* **17**, 13 (2016).
39. Aken, B. L. et al. The Ensembl gene annotation system. *Database (Oxf.)* **2016**, baw093 (2016).
40. Kim, D., Paggi, J. M., Park, C., Bennett, C. & Salzberg, S. L. Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype. *Nat. Biotechnol.* **37**, 907–915 (2019).
41. Kovaka, S. et al. Transcriptome assembly from long-read RNA-seq alignments with StringTie2. *Genome Biol.* **20**, 278 (2019).
42. Gonzalez-Garay, M. L. Introduction to isoform sequencing using Pacific Biosciences technology (Iso-Seq). In *Transcriptomics and Gene Regulation* **Vol. 9** (ed. Wu, J.), 141–160 (Springer, Dordrecht, Netherlands, 2015).
43. Kuznetsov, D. et al. OrthoDB v11: annotation of orthologs in the widest sampling of organismal diversity. *Nucleic Acids Res.* **51**, D445–D451 (2023).
44. Brůna, T., Lomsadze, A. & Borodovsky, M. GeneMark-EP+: eukaryotic gene prediction with self-training in the space of genes and proteins. *NAR Genom. Bioinform.* **2**, lqaa026 (2020).
45. Brůna, T., Lomsadze, A. & Borodovsky, M. GeneMark-ETP significantly improves the accuracy of automatic annotation of large eukaryotic genomes. *Genome Res.* **34**, 757–768 (2024).
46. Stanke, M., Diekhans, M., Baertsch, R. & Haussler, D. Using native and syntenically mapped cDNA alignments to improve de novo gene finding. *Bioinformatics* **24**, 637–644 (2008).
47. Camacho, C. et al. BLAST+: architecture and applications. *BMC Bioinforma.* **10**, 421 (2009).
48. Manni, M., Berkeley, M. R., Seppey, M., Simão, F. A. & Zdobnov, E. M. BUSCO update: novel and streamlined workflows along with broader and deeper phylogenetic coverage for scoring of eukaryotic, prokaryotic, and viral genomes. *Mol. Biol. Evol.* **38**, 4647–4654 (2021).
49. Tian, H. et al. Current understanding of functional peptides encoded by lncRNA in cancer. *Cancer Cell Int.* **24**, 252 (2024).
50. Mattick, J. S. et al. Long non-coding RNAs: definitions, functions, challenges and recommendations. *Nat. Rev. Mol. Cell Biol.* **24**, 430–447 (2023).
51. Barquist, L., Burge, S. W. & Gardner, P. P. Studying RNA homology and conservation with Infernal: from single sequences to RNA families. *Curr. Protoc. Bioinforma.* **54**, 12.13.1–12.13.25 (2016).
52. Cagirici, H. B., Sen, T. Z. & Budak, H. mirMachine: a one-stop shop for plant miRNA annotation. *J. Vis. Exp.* https://doi.org/10.3791/62430 (2021).
53. Clarke, A. W. et al. MirGeneDB 3.0: improved taxonomic sampling, uniform nomenclature of novel conserved microRNA families and updated covariance models. *Nucleic Acids Res.* **53**, D116–D128 (2025).
54. Kaessmann, H. Origins, evolution, and phenotypic impact of new genes. *Genome Res.* **20**, 1313–1326 (2010).
55. Liao, W.-W. et al. A draft human pangenome reference. *Nature* **617**, 312–324 (2023).
56. Emms, D. M. & Kelly, S. OrthoFinder: phylogenetic orthology inference for comparative genomics. *Genome Biol.* **20**, 238 (2019).
57. Nevers, Y. et al. The Quest for Orthologs orthology benchmark service in 2022. *Nucleic Acids Res.* **50**, W623–W632 (2022).
58. Dudchenko, O. et al. De novo assembly of the *Aedes aegypti* genome using Hi-C yields chromosome-length scaffolds. *Science* **356**, 92–95 (2017).
59. Zahn-Zabal, M., Dessimoz, C. & Glover, N. M. Identifying orthologs with OMA: a primer. *F1000Res.* **9**, 27 (2020).
60. Jones, P. et al. InterProScan 5: genome-scale protein function classification. *Bioinformatics* **30**, 1236–1240 (2014).
61. Martin, F. J. et al. Ensembl 2023. *Nucleic Acids Res.* **51**, D933–D941 (2023).
62. Chao, K.-H. et al. Combining DNA and protein alignments to improve genome annotation with LiftOn. *Genome Res.* **35**, 311–325 (2025).
63. Zimin, A. V., Puiu, D., Pertea, M., Yorke, J. A. & Salzberg, S. L. Efficient evidence-based genome annotation with EviAnn. Preprint at https://www.biorxiv.org/content/10.1101/2025.05.07.652745v1 (2025).
64. Brůna, T. et al. Galba: genome annotation with miniprot and AUGUSTUS. *BMC Bioinforma.* **24**, 327 (2023).
65. Li, H. Protein-to-genome alignment with miniprot. *Bioinformatics* **39**, btad014 (2023).
66. Roadmap Epigenomics Consortium. et al. Integrative analysis of 111 reference human epigenomes. *Nature* **518**, 317–330 (2015).
67. Ernst, J. & Kellis, M. Chromatin-state discovery and genome annotation with ChromHMM. *Nat. Protoc.* **12**, 2478–2492 (2017).
68. Mc Cartney, A. M. et al. Chasing perfection: validation and polishing strategies for telomere-to-telomere genome assemblies. *Nat. Methods* **19**, 687–695 (2022).
69. Guizard, S. et al. nf-core/isoseq: simple gene and isoform annotation with PacBio Iso-Seq long-read sequencing. *Bioinformatics* **39**, btad150 (2023).
70. Fulco, C. P. et al. Activity-by-contact model of enhancer-promoter regulation from thousands of CRISPR perturbations. *Nat. Genet.* **51**, 1664–1669 (2019).
71. Parra, Z. E., Lillie, M. & Miller, R. D. A model for the evolution of the mammalian T-cell receptor α/δ and μ loci based on evidence from the duckbill platypus. *Mol. Biol. Evol.* **29**, 3205–3214 (2012).
72. Chen, Z., Ain, N. U., Zhao, Q. & Zhang, X. From tradition to innovation: conventional and deep learning frameworks in genome annotation. *Brief. Bioinforma.* **25**, bbae138 (2024).
73. Mendoza-Revilla, J. et al. A foundational large language model for edible plant genomes. *Commun. Biol.* **7**, 835 (2024).
74. Flamholz, Z. N., Biller, S. J. & Kelly, L. Large language models improve annotation of prokaryotic viral proteins. *Nat. Microbiol.* **9**, 537–549 (2024).
75. Kurtzer, G. M., Sochat, V. & Bauer, M. W. Singularity: scientific containers for mobility of compute. *PLoS ONE* **12**, e0177459 (2017).
76. Köster, J. & Rahmann, S. Snakemake—a scalable bioinformatics workflow engine. *Bioinformatics* **28**, 2520–2522 (2012).

77. Pertea, G. & Pertea, M. GFF utilities: GffRead and GffCompare. *F1000Res.* **9**, ISCB Comm J-304 (2020).
78. Liu, D., Hunt, M. & Tsai, I. J. Inferring synteny between genome assemblies: a systematic evaluation. *BMC Bioinforma.* **19**, 26 (2018).
79. Wu, F., Mai, Y., Chen, C. & Xia, R. SynGAP: a synteny-based toolkit for gene structure annotation polishing. *Genome Biol.* **25**, 218 (2024).

Acknowledgements

We thank Leanne Haggerty for her valuable insight on her own genome annotation experience and for providing feedback on our manuscript. All figures were generated with BioRender. This research was supported by the Canadian Institutes for Health Research (grant PJT 180542 to G.D.B.) and NSERC (RGPIN-2019-07014 to M.D.W.). D.J.S. is supported by The Ontario Genomics-CANSSI Postdoctoral Fellowship in Genome Data Science, the NSERC-PDF scholarship and the McLaughlin Centre Scholars Grant. J.T.S. is supported by the Ontario Institute for Cancer Research through funds provided by the Government of Ontario, the Government of Canada through Genome Canada and Ontario Genomics (OGI-136 and OGI-201) and the National Human Genome Research Institute (NHGRI560 project 5R01HG009190).

Author contributions

Z.A.C. and D.J.S. were responsible for conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing of the original draft, review and editing of the original draft, visualization, supervision and project administration. C.K.B.-H. was responsible for software, validation, investigation, resources and review and editing of the original draft. R.I. was responsible for software and review and editing of the original draft. M.D.W., J.T.S. and G.D.B. were responsible for review and editing of the original draft, supervision and funding acquisition.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41596-025-01301-1.

Correspondence and requests for materials should be addressed to Gary D. Bader.

Peer review information *Nature Protocols* thanks Marc Halfon, Jinagtao Lilue and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.