Published on Feb 23, 2024 by Arcadia Science **DOI**: 10.57844/arcadia-9602-3351

Predicted genes from the *Amblyomma americanum* draft genome assembly

We previously released a draft genome assembly for the lone star tick, *A. americanum.* We've now predicted genes from this assembly to use for downstream functional characterization and comparative genomics efforts.

Contributors (A-Z)

Feridun Mert Celebi, Seemay Chou, Rachel J. Dutton, Megan L. Hochstrasser, Elizabeth A. McDaniel, Austin H. Patton, Taylor Reiter, Emily C.P. Weiss

Version 2 · Mar 31, 2025

What's new?

We previously released a draft genome assembly and a long-read transcriptome assembly from the lone star tick, *Amblyomma americanum*. This assembly was approximately 90% complete and assembled into ~30,000 contigs. We decided to move forward with predicting genes from our draft genome assembly since annotation is limited for this tick species.

In this pub, we describe how we approached this through *de novo* transcriptome assembly, microbial decontamination, gene prediction, and validation analyses, which resulted in a set of predicted genes that is 81.5% complete and 8.7% redundant. We were encouraged that our set of gene models fell in the middle of the pack compared to those available for other tick species. Additionally, comparing the length distributions of our predicted proteins to protein hits in other tick references gave us confidence that our predicted gene models are around the expected lengths. Given the additional sequencing data and effort we'd need to improve the fragmented nature of the assembly and the encouraging sign that our predictions are of similar quality to other tick species, we decided to move forward with downstream functional analyses.

- This is a follow-up to work described in a **prior pub**, "*De novo* assembly of a long-read *Amblyomma americanum* tick genome." For complete background info and context, visit that pub and the **project narrative**, "Ticks as treasure troves: Molecular discovery in new organisms."
- You can find our **data** on NCBI at BioProject PRJNA932813, including our updated, decontaminated draft assembly and predicted genes and proteins. You can directly download the protein sequences here.
- You can find these same **data**, plus transcriptome assembly data and a file with classification for bacterial contigs, on Zenodo.
- You can find **code** for transcriptome assembly in this GitHub repo; microbial decontamination of the genome and gene model prediction, plus comparative analyses to tick references in this GitHub repo; and preprocessing tick reference proteomes/assembled transcriptomes for comparison in this GitHub repo.

The approach

> **SHOW ME THE DATA**: Access our data, including the decontaminated draft assembly, transcriptome assembly data, and predicted genes and proteins, on NCBI. You can find our transcriptome assembly data and classification for bacterial contigs on Zenodo.

To make our *Amblyomma americanum* genome useful, we next needed to predict which stretches of DNA correspond to genes. We first performed *de novo* transcriptome assembly using both publicly available *A. americanum* RNA-seq data and our long-read transcriptome assembly to determine which genes are expressed and the boundaries of the exons and introns in those genes. We then decontaminated the genome by removing microbial contigs. Using this decontaminated genome and transcriptome assembly, we predicted gene open reading frames and validated our predictions against proteins from other tick species.

De novo transcriptome assembly

To improve our annotation of the *A. americanum* genome, we first used publicly available RNA-seq data to build a *de novo* transcriptome (Table 1). RNA-seq data is commonly incorporated into gene prediction pipelines because it provides evidence for exons and splice sites **[1]**. One can either directly map RNA-seq reads to the genome or assemble them into transcripts and align them to the genome **[1]**. Since we used a combination of short- and long-read (PacBio IsoSeq) RNA-seq data, we chose the *de novo* assembly strategy for incorporating RNA-seq data into gene predictions.

SRA study accession	Number of samples	Sequencing type	Reference
SRP446981	24	Paired-end short-read (300 bp)	**[2]**
SRP032795	12	Paired-end short-read (200 bp)	**[3]**
SRP051699	4	Paired-end short-read (200 bp)	**[4]**
SRP052078; SRP052091; SRP052108; SRP052106; SRP052114; SRP052123; SRP052145; SRP052154	8	Paired-end short-read (152 bp)	N/A
SRP373454	1	PacBio IsoSeq	**[5]**

Table 1

Summary of publicly available RNA-seq data used to build the *A. americanum de novo* transcriptome.

View the **full metadata table** for the samples we analyzed in this workflow.

For short-read data, we followed pre-processing recommendations as outlined in the "Eel Pond Protocol" for *de novo* transcriptome assembly **[6][7]**. This approach is optimized for RNA-seq data from non-model organisms. It removes sequencing errors that could fragment the assembly while retaining low-coverage reads that could lead to a more complete assembly. We downloaded each sample from the NCBI Sequence Read Archive (SRA) using the `fasterq-dump` command in the SRA toolkit (version 3.0.6) **[8]**, quality- and adapter-trimmed the reads using fastp (version 0.23.4) **[9]**, and k-mer-trimmed and digitally normalized reads using the `trim-low-abund.py` script in khmer (version 3.0.0a3) **[10]**. Because the output of this command is interleaved reads, we split paired reads into separate files using the `repair.sh` command in the BBMap package (version 39.01) **[11]**.

Given that we combined RNA-seq data from multiple studies that had different variables (e.g., sex, tissue) or treatments, and that the complexity of RNA-seq data can impact the resultant quality of assembly **[12]**, we combined samples into 20 assembly groups that reflected similar underlying biological conditions. Each group was then assembled separately.

Organizing samples into assembly groups was a difficult decision to make, as increasing the number of assembly groups made merging transcriptomes difficult. The alternative would have been assembling all short reads from all samples simultaneously. However, we reasoned that since we already needed to merge our short-read assembly with our long-read assembly, and because we wanted to use multiple short-read assemblers to improve the completeness of our assembly **[13]**, this problem was unavoidable and would not be more difficult to solve with more assemblies.

We followed an assembly- and transcriptome-merging routine that is similar to the Oyster River Protocol for *de novo* transcriptome assembly **[6]**. We assembled each assembly group using Trinity (version 2.15.1) **[14]** and rnaSPAdes (version 3.15.5) **[15]**. Then, we combined and deduplicated these assemblies as well as the long-read assembly **[5]** together using a modified version of the orthofuser approach implemented in the Oyster River Protocol **[6]**.

> View the **workflow code** for our transcriptome assembly approach (DOI: 10.5281/zenodo.10601710).

We modified our approach to accommodate a long-read assembly and to work around issues we encountered with the scalability of the mapping step in the TransRate tool **[16]**. Our first step made each contig name unique by prepending the assembly group to the name using the `bbrename.sh` command in BBMap (version 39.01) **[11]**. Next, we used `mmseqs easy-cluster` in the MMSseqs2 package (version 14.7e284) **[17]** to remove perfect duplicates across all transcriptomes. We used the `subseq` command in the seqtk package (version 1.4) to remove duplicates **[18]** and removed transcripts shorter than 75 base pairs (bp) using the `seq` command in the seqkit package (version 2.5.1) **[19]**. Next, we used OrthoFinder (version 2.5.5) in DNA mode (`-d`) and with an MCL parameter of 12 to group transcripts into orthologous groups by assembly **[20]**. These groups represent transcripts that encode the same isoform or gene. To select a representative transcript from each group, we first scored the quality of each short-read transcript using the TransRate tool **[6][16]**. We then selected at least one transcript from each group by selecting either all long-read transcripts from the group if any long-read transcripts were present, or selecting the transcript with the highest overall score.

We used this selection approach because we reasoned that long-read transcripts are more likely to be high-quality than short-read transcripts and because we were unable to score the long-read transcripts using TransRate due to limitations in short-read mapping. This filtering approach produced our first merged transcriptome, but we then "rescued" potentially missing transcripts that were filtered out by these steps using DIAMOND BLASTx (version 2.1.8) **[21]** annotations against the SwissProt database as performed in orthofuser **[6]**. Lastly, we de-duplicated this final set of transcripts at 98% identity using `cd-blastx` in the CD-HIT package (version 4.8.1) **[22]**.

After assembly and merging, we next decontaminated the transcriptome. To do this, we first identified contaminant genomes in our transcriptome by running `sourmash gather` (`-k 51` , `--scaled 10000`) (version 4.8.3) against k-mer databases of bacterial, archaeal, protozoan, fungal, mammalian, other vertebrate, and plant genomes in GenBank **[23][24]**. We then downloaded the genomes for contaminants using `ncbi-genome-download` (version 0.3.3) **[25]**, and used the BLAST package (version 2.14.1) to make a BLAST database from these genomes (`makeblastdb`) and BLAST (`blastn`) each transcript against the database **[26]**. We removed transcripts that had a BLAST hit greater than length 100 nucleotides that matched at least 10% of the transcript with an identity greater than or equal to 80%. We removed them from the transcriptome using the `subseq` command in the seqtk package (version 1.4) **[27]**.

To evaluate the transcriptome, we performed four checks. First, we quantified the fraction of reads that mapped back to the assembly using the `quant` command in the Salmon package (version 1.10.2) **[28]**. Next, we used TransRate (without mapping mode) **[6][16]** to produce transcriptome quality statistics. Then, we used dammit **[29]** to orchestrate annotation including ORF detection with TransDecoder **[30]** — we used our fork to patch small bug fixes in the dammit code base. Last, we performed quality assessment via BUSCO (version 5.5.0) using transcriptome mode (`-m tran`) against the arachnida_odb10 lineage **[31]**. This is a BUSCO database containing 2,934 marker genes that have a single copy in most genomes in the *Arachnida* taxonomic class, of which *A. americanum* is a member.

We've documented our entire approach as a Snakemake workflow (version 7.31.0) **[32]** in this file.

> View the **workflow code** for our transcriptome assembly approach.

Microbial decontamination of the genome assembly

Before predicting genes, we identified and removed bacterial contigs that could either be from endosymbiotic taxa or contaminants. To assign taxonomy to each contig, we created a DIAMOND database of our existing clustered NCBI-nr database **[33]**, and ran `diamond blastx` using this database against the *A. americanum* contigs with DIAMOND (version 2.1.8) **[21]**. We then used the `blast2lca` program of MEGAN (version 6.25.3) **[34]** and the corresponding NCBI-nr MEGAN mapping file to parse the DIAMOND BLASTx results and produce a TSV with a taxonomic assignment per contig. We then calculated the length of each contig using Biopython (version 1.81) **[35]** and incorporated this with taxonomy information for contigs classified as bacteria or unknown. Using the R packages tidyverse (version 2.0.0) **[36]** and BioStrings (version 2.68.1) **[37]**, we selected the bacterial and unknown contigs longer than 1,000 bp to remove from the assembly.

> View our **code** for microbial decontamination of the genome and gene model prediction, plus comparative analyses to tick references (DOI: 10.5281/zenodo.10694669).

Gene prediction and validation

To predict gene models and proteins for the *A. americanum* draft genome, we used the nf-core genomeannotator workflow **[38]**, which is still under active development. We specifically accessed the latest `dev` branch from a specific commit and launched the workflow according to these commands. The pipeline first filters contigs by size with GAAS (version 1.20) **[39]** using a default minimum contig size of 5,000 bp to consider for gene model prediction. We then identified and masked the repeat sequences using RepeatModeler (version 2.0.2) **[40]** and RepeatMasker (version 4.1.2-p1) **[41]**. We first cleaned and reformatted the assembled transcripts with GAAS (version 1.2.0) **[39]** and exonerate (version 2.4.0) **[42]**, then mapped to the repeat-masked assembly with minimap2 (version 2.22) **[43]**. We used the mapped reads to create the GFF hints input to AUGUSTUS (version 3.4.0) **[44]**, which we used for initial gene model prediction. We used these gene models as input to EVidenceModeler (version 1.1.0) **[45]** to produce a set of non-redundant proteins. From the GFF output from the nf-core genomeannotator workflow, we created a GTF-formatted file of the annotations with AGAT **[46]**.

To validate the set of proteins output by EVidenceModeler, we first ran BUSCO (version 5.5.0) **[31]** in protein mode using the arachnida_odb10 lineage. To compare the predicted proteins to other tick species, we obtained available proteins or predicted proteins from transcriptomes of various tick species downloaded from accessions listed in Table 2 using our “protein-data-curation” Snakemake workflow. Briefly, proteomes or assembled transcripts are downloaded by the provided URL link. For assembled transcriptomes, TransDecoder (version 5.7.1) **[30]** predicts coding regions within transcripts. For species with multiple listed proteomes or transcriptomes, the workflow merges and clusters these at 90% sequence identity with CD-HIT (version 4.8.1) **[47]**. It also filters proteins to remove any protein smaller than 25 amino acids, and if isoform information is provided, it only keeps the longest isoform for a given protein. Additionally, the pipeline adds functional annotation information for each species’ proteome through KEGG annotations with KofamScan (version 1.3.0) **[48]**, EGGNOG annotations with eggNOG-mapper (version 2.1.10) **[49]**, and predicts signal peptides with DeepSig (version 1.2.5) **[50]**.

> View our **code** for preprocessing existing proteomes/assembled transcriptomes to obtain uniform proteome datasets to compare to our predicted *A. americanum* proteins (DOI: 10.5281/zenodo.10607898).

Species	Total protein count	Source	Accession
Dermacentor andersoni	22,843	Existing proteome	GCF_023375885.1
Dermacentor silvarum	22,390	Existing proteome	GCF_013339745.2
Haemaphysalis longicornis	23,852	Existing proteome	GCA_013339765.2
Hyalomma asiaticum	27,476	Existing proteome	GCA_013339685.2
Ixodes persulcatus	25,991	Existing proteome	GCA_013358835.2
Ixodes scapularis	20,184	Existing proteome	UP000001555
Rhipicephalus microplus	17,234	Existing proteome	GCF_013339725.1
Rhipicephalus sanguineus	20,838	Existing proteome	GCF_013339695.2
Amblyomma americanum	28,319	Genome with transcriptome assembly	This study
Amblyomma sculptum	11,655	Transcriptome	GEEX01
Dermacentor variabilis	18,937	Transcriptome	GGQS01
Ixodes ricinus	20,704	Transcriptome	GIDG01
Ornithodoros erraticus	18,386	Transcriptome	GFWV01,GIXX02
Ornithodoros moubata	24,072	Transcriptome	GIXP02, GFJQ02
Ornithodoros turicata	29,460	Transcriptome	GDIC01, GDIE01

Table 2

Tick species accession information.

For each tick species, we report the number of predicted proteins from our pipeline and whether we obtained existing proteomes directly or predicted proteins from assembled transcriptome accessions. For proteins obtained from existing accessions, we downloaded all proteins from the RefSeq protein accession for that species, except for *Ixodes scapularis*, where we downloaded the proteins from the UniProt proteome for that organism. For species where we predicted proteins from transcriptome assemblies, we accessed the raw assembled RNA-seq contigs from the NCBI transcriptome shotgun assembly database. For some species, we used multiple study accessions to predict proteins.

We then compared these tick proteomes against the filtered set of *A. americanum* proteins that we'd also clustered at 90% sequence identity and from which we removed proteins smaller than 25 amino acids. Therefore, statistics and figures of these comparisons are from proteomes that have all been filtered the same way. We created a workflow that makes pairwise `diamond blastp` comparisons with DIAMOND (version 2.1.8) **[21]** for every tick species proteome against the *A. americanum* proteome. Although each protein from a reference tick species was only used once in the `diamond blastp` search, some *A. americanum* proteins had multiple hits. We did not dereplicate these instances or pick the best hit since we wanted the `diamond blastp` comparisons for quick validation checks of total protein hits and length distributions of those hits. The workflow also calculates BUSCO quality statistics for each input proteome, where we ran BUSCO (version 5.5.0) **[31]** in protein mode using the arachnida_odb10 lineage. To parse and plot results from the `diamond blastp` results, we used the R packages tidyverse (version 2.0.0) **[36]**, ggridges (version 0.5.4) **[51]**, viridis (version 0.6.4) **[52]**, and ggpubr (version 0.6.0) **[53]**.

Additional methods

We used ChatGPT to help write and clean up code.

The data

> **SHOW ME THE DATA**: Access our data, including the decontaminated draft assembly, transcriptome assembly data, and predicted genes and proteins, on NCBI. You can find our transcriptome assembly data and classification for bacterial contigs on Zenodo.

De novo assembly produced a near-complete transcriptome

To improve the genome annotation, we used publicly available *Amblyomma americanum* RNA-seq data to build a transcriptome. RNA-seq improves eukaryotic genome annotation by providing additional evidence for gene models **[1]**. We assembled a transcriptome from 48 short-read RNA-seq samples and one long-read transcriptome (Table 1). We report quality statistics about the transcriptome below, in Table 3. The transcriptome contained 1.06 million transcripts that encoded 319,324 predicted coding domain sequences. The transcriptome was 97.5% complete (86.3% duplicated).

Metric	Value	Tool used
Number of transcripts	1,061,354 contigs	dammit
Number of base pairs	974,803,561 bp	dammit
Minimum transcript length	75 bp	dammit
Maximum transcript length	36,548 bp	dammit
Median transcript length	284 bp	dammit
Mean transcript length	918 bp	dammit
N50 length	3,176 bp	dammit
Number of 25-mers	947,208,357 k-mers	dammit
Number of unique 25-mers	428,744,143 k-mers	dammit
Number ambiguous bases	6,482 bases	dammit
Redundancy	55%	dammit
GC percentage	48%	dammit
Complete single-copy genes	2,859 (97.5%)	BUSCO
Complete and single-copy	328 (11.2%)	BUSCO
Complete and duplicated	2,531 (86.3%)	BUSCO
Fragmented single-copy genes	25 (0.9%)	BUSCO
Missing single-copy genes	50 (1.6%)	BUSCO

Table 3

Transcriptome quality metrics.

These metrics highlight that the transcriptome is highly redundant. This likely arises from multiple factors. The *A. americanum* RNA-seq samples are highly heterogeneous and variability may come from pooling samples before sequencing. There are also differences in the populations sampled — the RNA-seq samples we used to build this transcriptome come from ticks that originated from multiple independent populations around the United States, which studies have shown display high heterogeneity **[54]**. We don't think that this interferes with the usefulness of the transcriptome for gene model prediction in the genome, but we encourage others to exercise caution for other downstream use cases, such as differential gene expression transcript quantification.

Draft gene predictions and validations from a decontaminated assembly

We took the pseudohaploid, deduplicated draft genome assembly and identified and removed contigs classified as bacterial or unknown (Figure 1). This step removed 1,268 contigs for a new filtered assembly with 36,883 contigs. We then used this filtered assembly as the input for the nf-core/genomeannotator workflow from this specific commit to predict gene models and proteins.

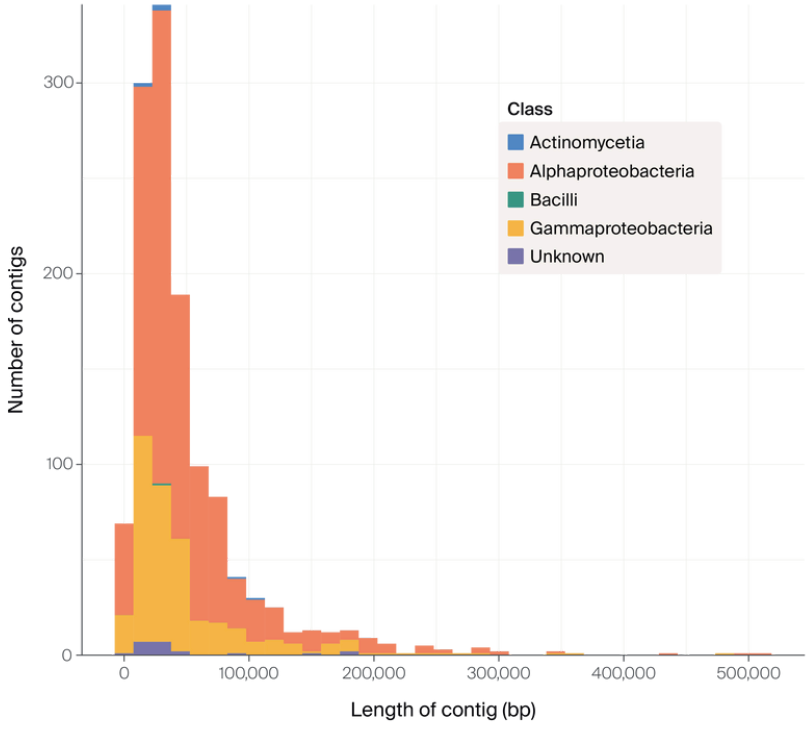

Figure 1

Contigs identified as bacterial or unknown and corresponding lengths.

Color corresponds to the class of bacteria that we taxonomically identified for that contig.

The set of 34,557 proteins we obtained had a BUSCO completeness of 81.5% and duplication of 8.4% against the arachnida_odb10 lineage. We then compared a reduced set of 28,319 predicted proteins that were filtered for a minimum length of 25 amino acids and clustered at 90% identity against filtered proteins we obtained or predicted from 14 other tick species. We checked: 1) BUSCO quality scores across tick references compared to the *A. americanum* proteome, 2) the number of identified homologs against other tick species, and 3) the distribution of alignment lengths of identified homologs to see if there is a high percentage of fragmented proteins in our dataset. From the BUSCO quality score comparisons, our *A. americanum* predicted proteins aren't as complete as those from other tick genome assembly efforts that were more curated and less fragmented than our draft genome (Figure 2). However, we're encouraged that the quality of predicted proteins for *A. americanum* falls somewhere in the middle of the pack when we compare to other tick assembly and annotation efforts.

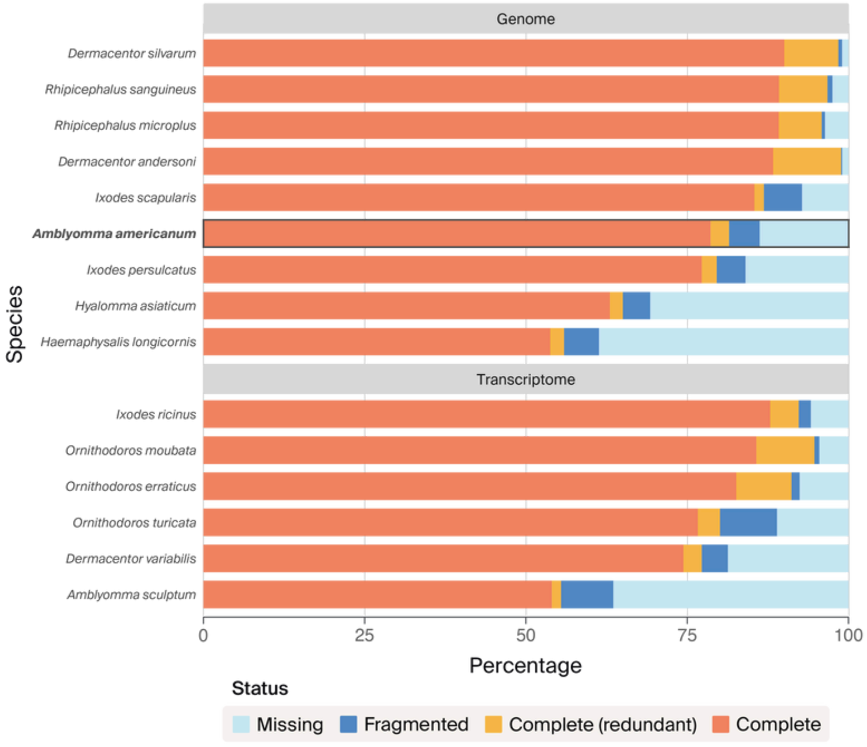

Figure 2

BUSCO scores for filtered tick proteomes compared to the filtered *Amblyomma americanum* proteome using the arachnida_odb10 lineage.

We curated proteins either directly from genome or assembled transcriptome references for each tick species listed in Table 2.

We then compared proteins from *A. americanum* to aligned proteins in the other tick species with pairwise `diamond blastp` comparisons. We calculated both the proportion of proteins from other tick species that had hits in the *A. americanum* proteome relative to the total number of predicted *A. americanum* proteins. The proportion of proteins from other tick species with hits relative to the total number of proteins from that tick species' proteome (Figure 3). For example, we predicted proteins from the tick *Amblyomma sculptum* based on a transcriptome assembly, and this was one of the least complete proteomes in our reference set (we've highlighted the *A. sculptum* points with red squares in Figure 3).

We identified hits for about 30% of the *A. americanum* proteome in *A. sculptum.* Conversely, 92% of *A. sculptum* proteins have a hit in the *A. americanum* proteome. We have represented this relationship between protein hits in both directions in Figure 3 to demonstrate that the relationship between the number of proteins is likely due to both the quality of the reference proteome and the evolutionary relatedness of that tick species to *A. americanum*.

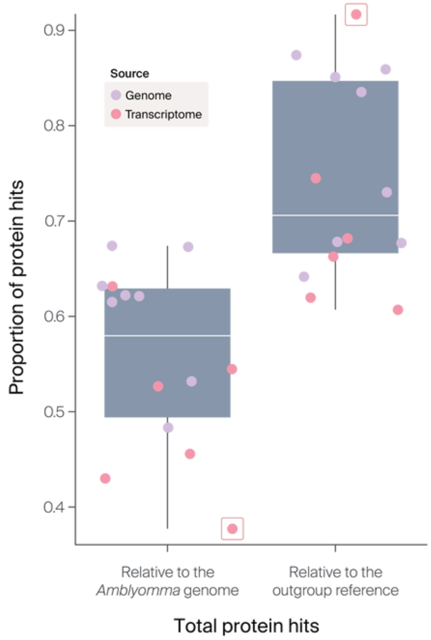

Figure 3

Distribution of protein hits from 14 other tick species relative to either the total number of proteins in the *A. americanum* proteome or to the total number in the corresponding tick reference proteome.

Points are colored by whether the proteome originated from a genome or if we predicted it from an assembled transcriptome. We've highlighted the points representing the *A. sculptum* proteome with red squares.

We then checked the length of our predicted proteins compared to the proteins from the other 14 tick species. For each hit, we divided the length of the *A. americanum* source protein by the length of the protein hit from one of the reference species. We filtered for proteins where this proportion was less than or equal to one, specifically looking for proteins that are highly fragmented in *A. americanum* or much shorter than the corresponding hit in the other species (Figure 4). Depending on the reference we compared to, 46–82% of protein hits from *A. americanum* were at least 90% the length of the reference protein. *A. sculptum* had the most proteins of similar length to matches in the *A. americanum* genome, which makes sense this is the most closely related species in our reference set. Encouragingly, this shows that compared to most tick references, the majority of *A. americanum* proteins are at least 90% the length of the reference hit protein and that there are not many fragmented proteins in our dataset compared to the references.

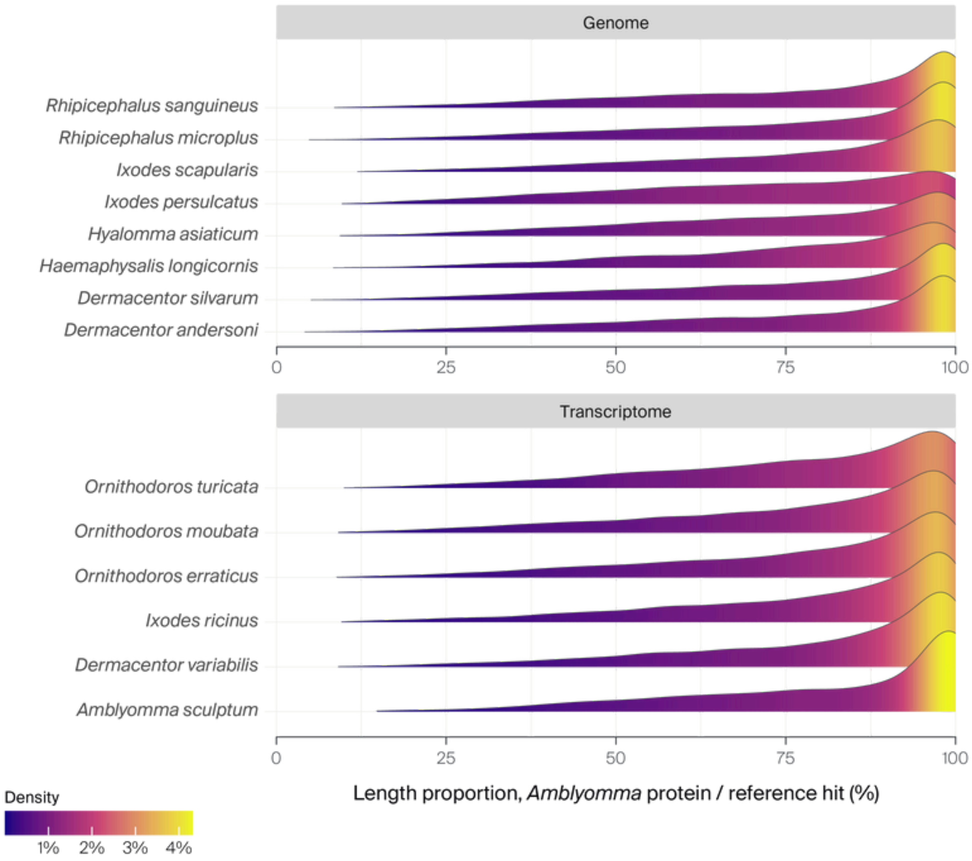

Figure 4

Distribution of protein lengths for *A. americanum* for hit proteins among 14 tick reference species.

Plots are separated by whether the proteins for the reference tick species were obtained directly from the genome or predicted from the assembled transcriptome of the species. Length proportion is calculated by the length of the source *A. americanum* protein divided by the length of the corresponding hit protein in that tick species. Color corresponds to the density of proteins with the calculated length proportion.

Key takeaways

We produced gene predictions from our *Amblyomma americanum* draft genome assembly with 81.5% completeness and 8.4% redundancy. Given that our draft assembly is quite fragmented (30,000 contigs with 90% completeness), we think we've obtained the best possible gene models we can using available tools without drastically increasing redundancy levels. The quality of our gene predictions is similar to those of other tick species. We've therefore decided to move forward with more intentional functional annotation and comparative analyses.

> **SHOW ME THE DATA**: Access our data, including the decontaminated draft assembly, transcriptome assembly data, and predicted genes and proteins, on NCBI. You can find our transcriptome assembly data and classification for bacterial contigs on Zenodo.

Next steps

In the future, if we undertake similar *de novo* transcriptome assembly efforts, we'd like to improve upon this approach. We think the deduplication procedure was unnecessarily complicated and sub-optimal — our BUSCO scores show a very duplicated transcriptome. However, TransRate was limited both by the number of contigs in the transcriptome that it could score in a given run (it did not work with one million transcripts) and by the number of short reads it could align (it failed with ~10 GB R1/R2 files), making it impossible to score all transcripts in a single TransRate run, as is implemented in the original orthofuser protocol. We're considering limiting ourselves to a single transcriptome assembler that outputs isoform information (Trinity or rnaSPAdes), but this will only work as a complete solution if we don't have a long-read transcriptome to combine with. We're also considering using a de Bruijn graph approach to identify transcripts with shared sequencing content, but if we take this approach, we'll need to validate it carefully.

References

1. Yandell M, Ence D. (2012). A beginner's guide to eukaryotic genome annotation. https://doi.org/10.1038/nrg3174

2 Lyu B, Li J, Niemeyer B, Anderson DM, Beerntsen B, Song Q. (2023). Integrative analysis highlights molecular and immune responses of tick Amblyomma americanum to Escherichia coli challenge. https://doi.org/10.3389/fcimb.2023.1236785

3 Radulović ŽM, Kim TK, Porter LM, Sze S-H, Lewis L, Mulenga A. (2014). A 24-48 h fed Amblyomma americanum tick saliva immuno-proteome. https://doi.org/10.1186/1471-2164-15-518

4 Karim S, Ribeiro JMC. (2015). An Insight into the Sialome of the Lone Star Tick, Amblyomma americanum, with a Glimpse on Its Time Dependent Gene Expression. https://doi.org/10.1371/journal.pone.0131292

5 Chou S, Poskanzer KE, Thuy-Boun PS. (2024). Robust long-read saliva transcriptome and proteome from the lone star tick, Amblyomma americanum. https://doi.org/10.57844/ARCADIA-3HYH-3H83

6 MacManes MD. (2018). The Oyster River Protocol: a multi-assembler and kmer approach for de novo transcriptome assembly. https://doi.org/10.7717/peerj.5428

7 Brown CT, Scott C, Crusoe MR, Sheneman L, Rosenthal J, Howe A. (2013). khmer-protocols 0.8.4 documentation. https://doi.org/10.6084/M9.FIGSHARE.878460

8 NCBI. (2023). The SRA Toolkit. https://github.com/ncbi/sra-tools

9 Chen S, Zhou Y, Chen Y, Gu J. (2018). fastp: an ultra-fast all-in-one FASTQ preprocessor. https://doi.org/10.1093/bioinformatics/bty560

10 Crusoe MR, Alameldin HF, Awad S, Boucher E, Caldwell A, Cartwright R, Charbonneau A, Constantinides B, Edvenson G, Fay S, Fenton J, Fenzl T, Fish J, Garcia-Gutierrez L, Garland P, Gluck J, González I, Guermond S, Guo J, Gupta A, Herr JR, Howe A, Hyer A, Härpfer A, Irber L, Kidd R, Lin D, Lippi J, Mansour T, McA'Nulty P, McDonald E, Mizzi J, Murray KD, Nahum JR, Nanlohy K, Nederbragt AJ, Ortiz-Zuazaga H, Ory J, Pell J, Pepe-Ranney C, Russ ZN, Schwarz E, Scott C, Seaman J, Sievert S, Simpson J, Skennerton CT, Spencer J, Srinivasan R, Standage D, Stapleton JA, Steinman SR, Stein J, Taylor B, Trimble W, Wiencko HL, Wright M, Wyss B, Zhang Q, zyme en, Brown CT. (2015). The khmer software package: enabling efficient nucleotide sequence analysis. https://doi.org/10.12688/f1000research.6924.1

11 Bushnell B. (2014). BBMap: A Fast, Accurate, Splice-Aware Aligner. https://www.osti.gov/biblio/1241166

12 Zhao Q-Y, Wang Y, Kong Y-M, Luo D, Li X, Hao P. (2011). Optimizing de novo transcriptome assembly from short-read RNA-Seq data: a comparative study. https://doi.org/10.1186/1471-2105-12-s14-s2

13 Hölzer M, Marz M. (2019). *De novo* transcriptome assembly: A comprehensive cross-species comparison of short-read RNA-Seq assemblers. https://doi.org/10.1093/gigascience/giz039

14 Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, Adiconis X, Fan L, Raychowdhury R, Zeng Q, Chen Z, Mauceli E, Hacohen N, Gnirke A, Rhind N, di Palma F, Birren BW, Nusbaum C, Lindblad-Toh K, Friedman N, Regev A. (2011). Full-length transcriptome assembly from RNA-Seq data without a reference genome. https://doi.org/10.1038/nbt.1883

15 Bushmanova E, Antipov D, Lapidus A, Prjibelski AD. (2019). rnaSPAdes: a *de novo* transcriptome assembler and its application to RNA-Seq data. https://doi.org/10.1093/gigascience/giz100

16 Smith-Unna R, Boursnell C, Patro R, Hibberd JM, Kelly S. (2016). TransRate: reference-free quality assessment of de novo transcriptome assemblies. https://doi.org/10.1101/gr.196469.115

17 Steinegger M, Söding J. (2017). MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. https://doi.org/10.1038/nbt.3988

18 Li H. (2023). seqtk. https://github.com/lh3/seqtk

19 Shen W, Le S, Li Y, Hu F. (2016). SeqKit: A Cross-Platform and Ultrafast Toolkit for FASTA/Q File Manipulation. https://doi.org/10.1371/journal.pone.0163962

20 Emms DM, Kelly S. (2019). OrthoFinder: phylogenetic orthology inference for comparative genomics. https://doi.org/10.1186/s13059-019-1832-y

21 Buchfink B, Xie C, Huson DH. (2014). Fast and sensitive protein alignment using DIAMOND. https://doi.org/10.1038/nmeth.3176

22 Li W, Godzik A. (2006). Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. https://doi.org/10.1093/bioinformatics/btl158

23 Reiter T. (2023). Sourmash cover databases. https://osf.io/6zk3d/

24 Irber L, Brooks PT, Reiter T, Pierce-Ward NT, Hera MR, Koslicki D, Brown CT. (2022). Lightweight compositional analysis of metagenomes with FracMinHash and minimum metagenome covers. https://doi.org/10.1101/2022.01.11.475838

25 Blin K. (2023). ncbi-genome-download. https://doi.org/10.5281/ZENODO.8192432

26 Ye J, McGinnis S, Madden TL. (2006). BLAST: improvements for better sequence analysis. https://doi.org/10.1093/nar/gkl164

27 Patro R, Duggal G, Love MI, Irizarry RA, Kingsford C. (2017). Salmon provides fast and bias-aware quantification of transcript expression. https://doi.org/10.1038/nmeth.4197

28 Scott C, Pierce T, Johnson LK, C. Titus Brown, Reid C, Badger TG, Luiz Irber. (2019). dib-lab/dammit: dammit v1.2. https://doi.org/10.5281/ZENODO.3569831

29 Haas BJ. (2023). TransDecoder. https://github.com/TransDecoder/TransDecoder

30 Simão FA, Waterhouse RM, Ioannidis P, Kriventseva EV, Zdobnov EM. (2015). BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. https://doi.org/10.1093/bioinformatics/btv351

31 Köster J, Rahmann S. (2012). Snakemake—a scalable bioinformatics workflow engine. https://doi.org/10.1093/bioinformatics/bts480

32 Reiter T. (2024). Clustering the NCBI nr database to reduce database size and enable faster BLAST searches. https://doi.org/10.57844/ARCADIA-W8XT-PC81

33 Huson DH, Beier S, Flade I, Górska A, El-Hadidi M, Mitra S, Ruscheweyh H-J, Tappu R. (2016). MEGAN Community Edition - Interactive Exploration and Analysis of Large-Scale Microbiome Sequencing Data. https://doi.org/10.1371/journal.pcbi.1004957

34 Cock PJA, Antao T, Chang JT, Chapman BA, Cox CJ, Dalke A, Friedberg I, Hamelryck T, Kauff F, Wilczynski B, de Hoon MJL. (2009). Biopython: freely available Python tools for computational molecular biology and bioinformatics. https://doi.org/10.1093/bioinformatics/btp163

35 Wickham H, Averick M, Bryan J, Chang W, McGowan L, François R, Grolemund G, Hayes A, Henry L, Hester J, Kuhn M, Pedersen T, Miller E, Bache S, Müller K, Ooms J, Robinson D, Seidel D, Spinu V, Takahashi K, Vaughan D, Wilke C, Woo K, Yutani H. (2019). Welcome to the Tidyverse. https://doi.org/10.21105/joss.01686

36 Pagès H, Aboyoun P, Gentleman R, DebRoy S (2023). Biostrings: Efficient manipulation of biological strings. https://doi.org/10.18129/B9.bioc.Biostrings

37 Ewels PA, Peltzer A, Fillinger S, Patel H, Alneberg J, Wilm A, Garcia MU, Di Tommaso P, Nahnsen S. (2020). The nf-core framework for community-curated bioinformatics pipelines. https://doi.org/10.1038/s41587-020-0439-x

38 Jacques Dainat, Mahesh Binzer-Panchal, Remi-Andre Olsen, Martin Dahlö, Darío Hereñú. (2020). NBISweden/GAAS: GAAS-v1.2.0. https://doi.org/10.5281/ZENODO.3835504

39 Smit AFA, Hubley R (2008–2015). RepeatModeler Open-1.0. http://www.repeatmasker.org

40 Smit AFA, Hubley R, Green P. (2013–2015). RepeatMasker Open-4.0. http://www.repeatmasker.org

41 Slater GSC, Birney E. (2005). Automated generation of heuristics for biological sequence comparison. https://doi.org/10.1186/1471-2105-6-31

42 Li H. (2018). Minimap2: pairwise alignment for nucleotide sequences. https://doi.org/10.1093/bioinformatics/bty191

43 Stanke M, Diekhans M, Baertsch R, Haussler D. (2008). Using native and syntenically mapped cDNA alignments to improve *de novo* gene finding. https://doi.org/10.1093/bioinformatics/btn013

44 Haas BJ, Salzberg SL, Zhu W, Pertea M, Allen JE, Orvis J, White O, Buell CR, Wortman JR. (2008). Automated eukaryotic gene structure annotation using EVidenceModeler and the Program to Assemble Spliced Alignments. https://doi.org/10.1186/gb-2008-9-1-r7

45 Dainat J. (2024). Nbisweden/agat: Agat-v1.4.0. https://doi.org/10.5281/zenodo.11106497.

46 Fu L, Niu B, Zhu Z, Wu S, Li W. (2012). CD-HIT: accelerated for clustering the next-generation sequencing data. https://doi.org/10.1093/bioinformatics/bts565

47 Aramaki T, Blanc-Mathieu R, Endo H, Ohkubo K, Kanehisa M, Goto S, Ogata H. (2019). KofamKOALA: KEGG Ortholog assignment based on profile HMM and adaptive score threshold. https://doi.org/10.1093/bioinformatics/btz859

48 Cantalapiedra CP, Hernández-Plaza A, Letunic I, Bork P, Huerta-Cepas J. (2021). eggNOG-mapper v2: Functional Annotation, Orthology Assignments, and Domain Prediction at the Metagenomic Scale. https://doi.org/10.1093/molbev/msab293

49 Savojardo C, Martelli PL, Fariselli P, Casadio R. (2017). DeepSig: deep learning improves signal peptide detection in proteins. https://doi.org/10.1093/bioinformatics/btx818

50 Wilke C. (2024). ggridges: Ridgeline Plots in 'ggplot2'. https://wilkelab.org/ggridges/

51 Garnier S, Ross N, BoB Rudis, Filipovic-Pierucci A, Galili T, Timelyportfolio, O’Callaghan A, Greenwell B, Sievert C, Harris DJ, Sciaini M, JJ Chen. (2023).

sjmgarnier/viridis: CRAN release v0.6.3. https://doi.org/10.5281/ZENODO.4679423

52 Kassambara A. (2023). ggpubr: 'ggplot2' Based Publication Ready Plots. https://rpkgs.datanovia.com/ggpubr/

53 Monzón JD, Atkinson EG, Henn BM, Benach JL. (2016). Population and Evolutionary Genomics of*Amblyomma americanum*, an Expanding Arthropod Disease Vector. https://doi.org/10.1093/gbe/evw080