GigaScience

De novo transcriptome assembly: A comprehensive cross-species comparison of short-read RNA-Seq assemblers

--Manuscript Draft--

Manuscript Number:	GIGA-D-18-00307	
Full Title:	De novo transcriptome assembly: A comprehensive cross-species comparison of short-read RNA-Seq assemblers	
Article Type:	Research	
Funding Information:	Deutsche Forschungsgemeinschaft (Collaborative Research Center/Transregio 124 - “Pathogenic fungi and their human host: Networks of Interaction”, subproject B5)	Dr. Martin Hölzer
	Deutsche Forschungsgemeinschaft (SPP-1596 - “Ecology and species barriers in emerging viral diseases”)	Dr. Martin Hölzer
	Deutsche Forschungsgemeinschaft (CRC 1076 “AquaDiva”, subproject A06)	Dr. Martin Hölzer
Abstract:	Background: In recent years, massively parallel cDNA sequencing (RNA-Seq) has emerged as a fast, cost-effective and powerful technology to study entire transcriptomes in various manners. In particular, for non-model organisms and in the absence of an appropriate reference genome, RNA-Seq is used to reconstruct the transcriptome de novo. Although the de novo transcriptome assembly of non-model organism has been on the rise recently and new tools are developed frequently, there is still a knowledge gap about which assembly software should be used to build a comprehensive de novo assembly. Results: Here we present a large-scale comparative study in which ten de novo assembly tools are applied to nine RNA-Seq data sets spanning different kingdoms of life. Overall, we build more than 200 single assemblies and evaluated their performance on a combination of 20 biological-based and reference-free metrics. Our study is accompanied by a comprehensive and extensible Electronic Supplement that summarizes all data sets, assembly execution instructions, and evaluation results. We show that no tool provides the best results for all data sets. However, Trans-ABySS, Trinity, SOAPdenovo-Trans, and SPAdes generally outperformed the other compared tools. In addition, we observed species-specific differences in the performance of each assembler. Conclusions: We recommend a careful choice of evaluation metrics to select the best assembling results as a critical step in the reconstruction of a comprehensive de novo transcriptome assembly.	
Corresponding Author:	Martin Hölzer Friedrich Schiller University Jena Jena, Thüringen GERMANY	
Corresponding Author Secondary Information:		
Corresponding Author's Institution:	Friedrich Schiller University Jena	
Corresponding Author's Secondary Institution:		
First Author:	Martin Hölzer	
First Author Secondary Information:		
Order of Authors:	Martin Hölzer	
	Manja Marz	
Order of Authors Secondary Information:		
Additional Information:		

Question	Response
Are you submitting this manuscript to a special series or article collection?	No
Experimental design and statistics Full details of the experimental design and statistical methods used should be given in the Methods section, as detailed in our Minimum Standards Reporting Checklist. Information essential to interpreting the data presented should be made available in the figure legends. Have you included all the information requested in your manuscript?	Yes
Resources A description of all resources used, including antibodies, cell lines, animals and software tools, with enough information to allow them to be uniquely identified, should be included in the Methods section. Authors are strongly encouraged to cite Research Resource Identifiers (RRIDs) for antibodies, model organisms and tools, where possible. Have you included the information requested as detailed in our Minimum Standards Reporting Checklist?	Yes
Availability of data and materials All datasets and code on which the conclusions of the paper rely must be either included in your submission or deposited in publicly available repositories (where available and ethically appropriate), referencing such data using a unique identifier in the references and in the “Availability of Data and Materials” section of your manuscript.	Yes

Have you have met the above requirement as detailed in our Minimum Standards Reporting Checklist?	

GigaScience, 2018, 1–15

doi: xx.xxxx/xxxx
Manuscript in Preparation
Research

RESEARCH

De novo transcriptome assembly: A comprehensive cross-species comparison of short-read RNA-Seq assemblers

Martin Hölzer[1,2,*] and Manja Marz[1,2,3]

[1]RNA Bioinformatics and High-Throughput Analysis, Friedrich Schiller University Jena, Leutragraben 1, Jena, Germany, and [2]European Virus Bioinformatics Center, Friedrich Schiller University Jena, Leutragraben 1, Jena, Germany, and [3]FLI Leibniz Institute for Age Research, Beutenbergstraße 11, Jena, Germany

[*]To whom correspondence should be addressed. Email: martin.hoelzer@uni-jena.de

Abstract

Background: In recent years, massively parallel cDNA sequencing (RNA-Seq) has emerged as a fast, cost-effective and powerful technology to study entire transcriptomes in various manners. In particular, for non-model organisms and in the absence of an appropriate reference genome, RNA-Seq is used to reconstruct the transcriptome *de novo*. Although the *de novo* transcriptome assembly of non-model organism has been on the rise recently and new tools are developed frequently, there is still a knowledge gap about which assembly software should be used to build a comprehensive *de novo* assembly.
Results: Here we present a large-scale comparative study in which ten *de novo* assembly tools are applied to nine RNA-Seq data sets spanning different kingdoms of life. Overall, we build more than 200 single assemblies and evaluated their performance on a combination of 20 biological-based and reference-free metrics. Our study is accompanied by a comprehensive and extensible Electronic Supplement that summarizes all data sets, assembly execution instructions, and evaluation results. We show that no tool provides the best results for all data sets. However, `Trans-ABySS`, `Trinity`, `SOAPdenovo-Trans`, and `SPAdes` generally outperformed the other compared tools. In addition, we observed species-specific differences in the performance of each assembler.
Conclusions: We recommend a careful choice of evaluation metrics to select the best assembling results as a critical step in the reconstruction of a comprehensive *de novo* transcriptome assembly.

Key words: transcriptomics, RNA-Seq, assembly, *de novo*, comparison

Background

In the last decade, the sequencing of entire transcriptomes (RNA sequencing, RNA-Seq) has established as a powerful tool to understand versatile molecular mechanisms and to address various biological questions [1–6]. In particular for non-model organisms and in the absence of a suitable reference genome, RNA-Seq is used to reconstruct and quantify whole transcriptomes [1, 4, 5]. Thus, RNA-Seq allows the identification of differentially expressed genes, even if there is currently no reference genome available: The short reads, nowadays most commonly produced by Illumina systems, can be assembled into contigs [2, 4]. Ideally, each contig corresponds to a certain transcript isoform. A key challenge is the management of the resulting data set, especially if different tools and parameter settings are used for the construction of multiple *de novo* transcriptome assemblies. Even though a reference genome is available, it is still recommended to complement a gene expression study by a *de novo* transcriptome assembly to identify transcripts that have been missed by the genome assembly process

Compiled on: August 13, 2018.
Draft manuscript prepared by the author.

or are just not appropriately annotated [2].

At first glance, the transcriptome assembly process seems similar to genome assembly, but actually there are fundamental differences and various challenges. On the one hand, some transcripts might have a very low expression level, while others are highly expressed [2, 4, 6]. Especially in eukaryotes, potentially each locus produces several transcripts (isoforms) due to alternative splicing events [4]. Short reads derived from one exon can be part of multiple paths in the assembly graph. Therefore, the graph structure can be ambiguous and the represented isoforms can be difficult to resolve. Furthermore, some transcript variants with a low expression level might be considered as sequencing errors by various tools and removed from the assembly process [7]. As with genome assembly, repetitive regions are also a major problem for the construction of transcripts [8]. The assembly problem gets even more complicated as the transcriptome varies between different cell types, environmental conditions, and time points. A successful transcriptome assembler should address all of these issues and be able to recover full-length transcripts of different levels of expression.

The *de novo* transcriptome assembly of non-model organism has been on the rise recently and new tools are developed frequently. Now there is a knowledge gap: Which assembly software and parameter settings should be used to construct a *good* assembly? In addition, there is no consensus about which evaluation metrics should be used to evaluate the quality of multiple *de novo* transcriptome assemblies.

In the last decade, several tools have been developed specifically for *de novo* transcriptome assembly [9–16]. Some of them are build on top of already existing genome assembly tools [9, 11], others were specially designed for transcriptome assembly [10]. Some tools may fit the needs of eukaryotic transcripts, where alternative splicing has to be considered to construct different isoforms, whereas other tools can handle simpler prokaryotic transcripts. More complicating, different RNA-Seq library preparation protocols result in reads of different kinds: single-end vs. paired-end, strand-specific vs. not strand-specific, different insertion sizes as well as varying read lengths and can comprise protein- and/or non-coding transcripts.

Although the evaluation of *de novo* transcriptome assembly tools have been already performed in the past [6, 17–23], these studies often rely on limited data sets (e.g. a single species, a single sequencing protocol) or focus only on a subset of all currently available assembly tools.

Though, all of these studies agree on one point: currently, there is no optimal assembly tool for all RNA-Seq data sets. Different species, sequencing protocols and parameter settings need different approaches and adjustments of the underlying algorithms to obtain the best possible results. Merging the contigs of different assembly tools and parameter settings to overcome the different disadvantages of certain assemblers and to combine their advantages seems to be the best way to obtain a comprehensive *de novo* transcriptome assembly [20]. Nevertheless, knowing the advantages and disadvantages of each tool is an important step in the direction of an automated evaluation and merging algorithm for multiple *de novo* transcriptome assemblies.

Here, we present a comprehensive evaluation of ten *de novo* assembly tools (long-standing and novel ones) across various short-read RNA-Seq data sets of different species relying on different Illumina sequencing parameters and protocols (Fig. 1). In comparison to recent studies, we do not only focus on RNA-Seq data of one species or kingdom. Instead, we use data sets from bacteria, fungi, plants, and higher eukaryotes (Fig. 1). We also include data sets that underwent viral infections. Our study shows substantial differences between the assembly results of RNA-Seq data derived from various species. We tested promising biological-based and reference-free metrics of several evaluation tools (see Methods) to assess and compare the performance of each assembler. In a next step, such metrics could be used for an automized selection of *good* assemblies or contigs to build a more comprehensive and improved *cluster-assembly*. Our results give insights into the performance and usability of the different assemblers and how they perform on the different data sets. As far as our knowledge goes, this is the most complete comparison of short-read *de novo* transcriptome assembly tools currently available.

Data Description

Description of RNA-Seq data used for assembly

We included nine RNA-Seq data sets of five different species with available reference genomes and annotations (Tab. 1). The data sets cover different kingdoms of life, comprising representatives for bacteria (*Escherichia coli*), fungi (*Candida albicans*), plant (*Arabidopsis thaliana*), and higher eukaryotes (*Mus musculus*, *Homo sapiens*). The reference genomes, annotations, and coding sequences were obtained from Ensembl (release 87) [31]. For *E. coli* str. K-12 substr. MG1655 and *A. thaliana* reference data was obtained from the Ensembl bacteria [32] or plant [33] database (release 34), respectively. Genome and annotation data for *C. albicans* SC5314 were obtained from the *Candida* Genome Database (Ca22) [34].

From a previous study (PRJNA429171) we obtained three samples of an Ebola virus (EBOV) infected HuH7 cell line with total RNA extracted 3 h, 7 h, and 23 h post infection [30] (Tab. 1). For the evaluation, we concatenated the human genome data with the EBOV genome of strain Zaire, Mayinga (GenBank: NC_002549).

In addition, we quasi-simulated RNA-Seq data based on a selection of protein- and long non-coding transcripts of human chromosome 1 (chr1). We downloaded the human annotation GTF file and protein-coding sequences (excluding *ab initio* predictions) from Ensembl and selected all protein-coding genes of chr1 (2,044 genes), comprising 352 genes with one isoform, 196 with two isoforms and 1,496 with more than two isoforms. We extended this set of protein-coding genes by 1,075 non-coding genes from chr1. The combined set of protein- and non-coding genes was used to create a set of transcripts including all known isoforms with a length >200 nt and without ambiguous *N* bases from which paired-end reads were simulated. Our final set of transcripts comprised 12,793 protein-coding transcripts as well as 1,006 lincRNAs, 839 antisense RNAs, and 7 snoRNAs of human chr1. Overall 14,645 transcript sequences were used as an input for `flux simulator` [26] for RNA-Seq raw read simulation, yielding 60 million paired-end 100 nt reads (Tab. 1, Tab. S1). We used `flux simulator` as suggested for Illumina data, utilizing the default 76-bp error model. With this simulated sequences, we attempt to mimic a state-of-the-art RNA-Seq data set based on Illumina's Ribo-Zero protocol for library preparation and rRNA depletion, further multiplexed three times and sequenced on one HiSeq 2500 lane.

Details about all used RNA-Seq data sets can be found in Electronic Supplement Tab. S1 [35].

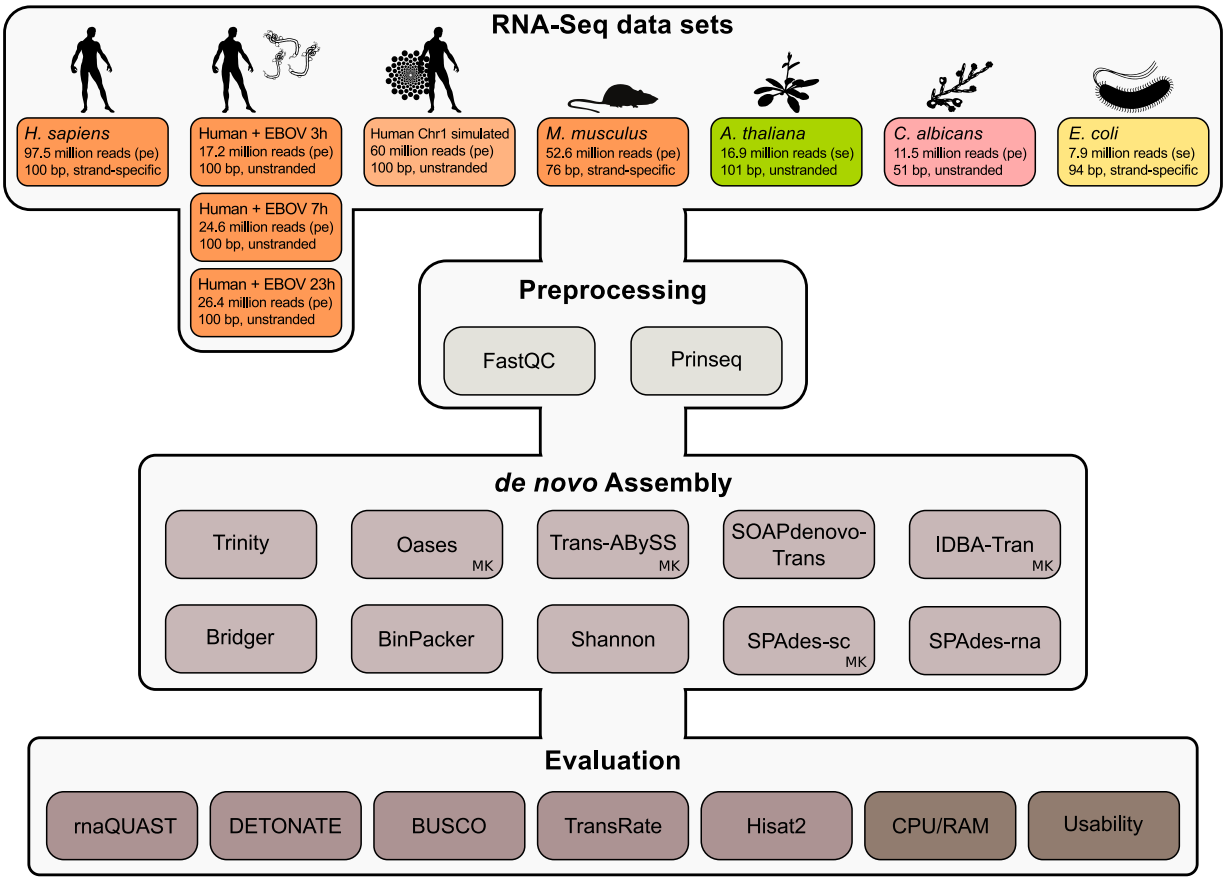

Figure 1. Overview of the used RNA-Seq data sets (orange – eukaryote, light orange – simulated human chromosome 1, green – plant, pink – fungi, yellow – bacterium) and evaluated assembly tools. Each data set was quality controlled with `FastQC` [24] and preprocessed with `Prinseq` [25] prior to assembly. Overall, more than 200 single *k*-mer assemblies were calculated. For details about the used data sets and assembly tools see Electronic Supplement Tab. S1 and Tab. S2, respectively. We selected a combination of 20 biological-based and reference-free metrics from the different evaluation tools to assess the quality of each assembly (Tab. 4; Methods). The CPU/RAM consumption and the usability of each assembler were not included in the calculated metric scores. Details can be found in the Methods. EBOV – Ebola virus; se/pe – single-end/paired-end; MK – the assemblers built-in multiple-*k*-mer approach was applied.

Quality control of all RNA-Seq data sets

We investigated the quality of each data set with `FastQC` [24] and used `Prinseq` [25] for an initial quality processing of all raw reads. Low-quality regions were trimmed with an average quality below 20 using a five base sliding window approach. Only reads that yielded a remaining read length of at least 25 nt were considered for further analysis. All reads including ambiguous `N` bases were removed. PolyA/T tails were trimmed. Details about the trimmed data, finally used for assembly, can be found in Electronic Supplement Tab. S1.

Data availability

The RNA-Seq data sets used in our study are publicly available and accessions can be found in the Methods and online Tab. S1.

Analyses

We used RNA-Seq data sets, assembly tools, and evaluation metrics as summarized in Fig. 1. Details can be found in the Methods and in the comprehensive online Electronic Supplement [35], providing deep insights into the performance of each assembler on each data set and individual metric. Overall, we evaluated ten *de novo* assembly tools (Fig. 1) and build more than 200 single *k*-mer assemblies.

With our selection of different data sets, we aim to represent not only various kingdoms of life, but also different experimental setups for RNA-Seq data: 1) single-end vs. paired-end data, 2) strand specificity vs. unstranded protocols, 3) polyA enriched vs. rRNA depleted library preparations, 4) different read lengths, and 5) different sequencing depths. The used RNA-Seq data sets are summarized in Fig. 1 and details can be found in Tab. 1 in the Methods and in Electronic Supplement Tab. S1.

The following sections present how each assembly tool performed for the different data sets and selected evaluation metrics (Methods Tab. 4). Tab. 2 shows the results for all 20 metrics and each assembly tool based on the *H. sapiens* (HSA) data set. Similar tables for all other data sets can be found in Electronic Supplement Tab. S9. The results that are shown for HSA in Tab. 2 correspond to the summarized *metric score* (*MS*) values shown for the HSA and all other data sets in Fig. 2. For each metric, an assemblers *MS* is increased by one if the resulting assembly arranges within the three best-ranked results for a given data set. For example, `Trinity` [10] achieved a *MS* of 8 for the *H. sapiens* data set across all 20 metrics evaluated and hereafter denoted as 8/20 (2, 2). We further sum up the *MS* for a single assembly tool over all data sets to calculate an *overall metric score* (*OMS*). Detailed definitions for calculating the *MS* and *OMS* values are given in the Methods.

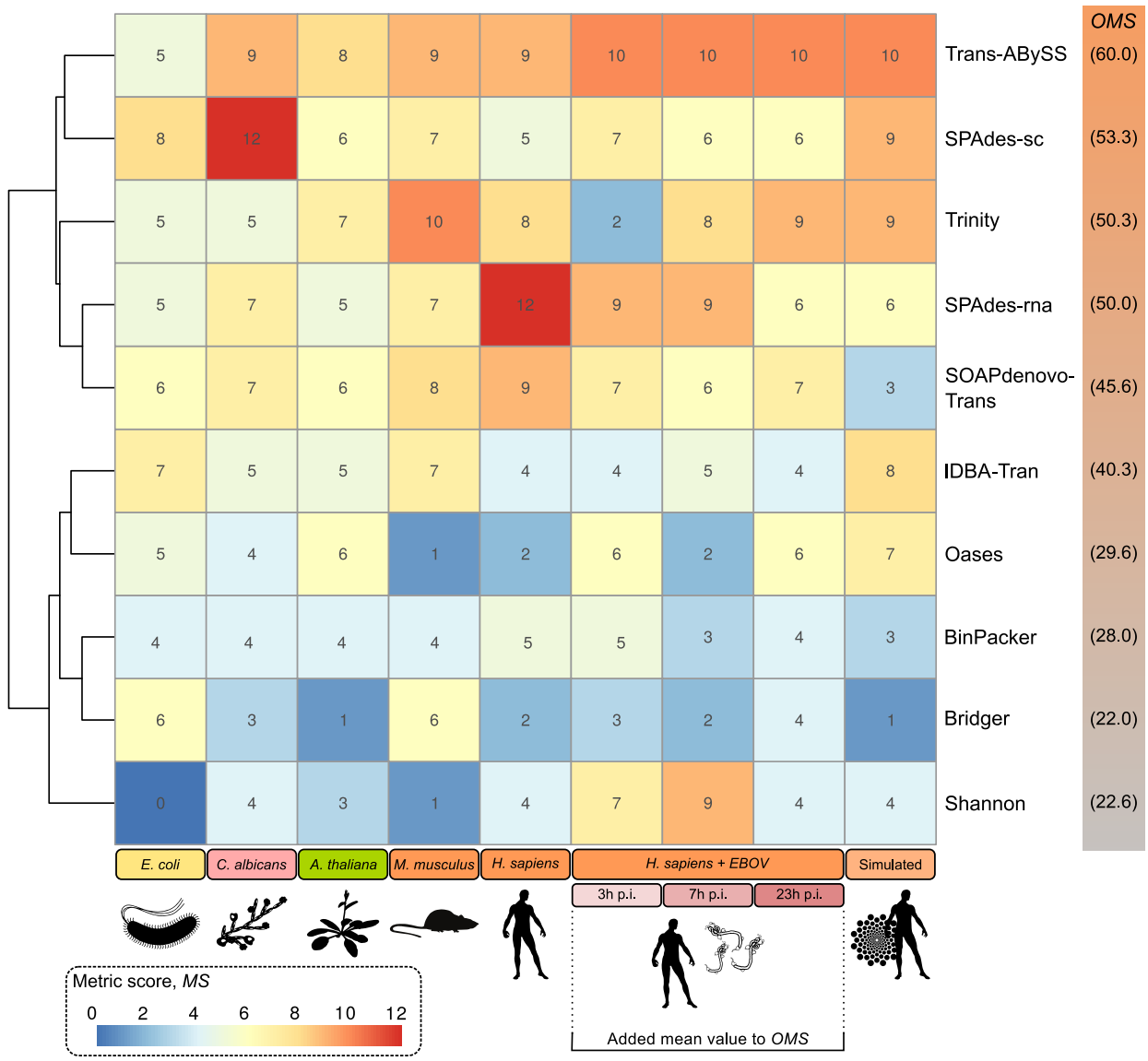

Figure 2. Heat map showing for each data set (column) and each assembler (row) the calculated *metric score MS* (detailed definition in the Methods). The *MS* for one assembly tool and a single data set is based on 20 pre-selected metrics (see Tab. 4 and Methods for details) and is shown in one cell in the heat map. For each metric, an assemblers *MS* is increased by one if the resulting assembly arranges within the three best-ranked results for this data set. The hierarchical clustering of the metric scores divides the assembly tools in two groups of generally high-ranked (upper half) and low-ranked (bottom half) tools. Numbers in brackets next to the assembler names present the summed up metric scores (*overall metric score*, *OMS*) for all nine data sets (see Methods). For the three similar human data sets infected with the Ebola virus (Fig. 1), we added the mean value to the *OMS*. Details about the metric results for the human data set (no infection) can be found in Tab. 2 and for all other data sets in Electronic Supplement Tab. S9.

Table 1. Nine RNA-Seq data sets were used for assembly. Study and run accession numbers are given for the NCBI short-read archive (SRA). For the HSA data set the ENCODE data center accession is provided. Read numbers are given in million. We simulated one artificial data set based on protein-coding and non-coding transcripts of human chromosome 1 (Chr 1) using `flux simulator` [26] (HSA-FLUX). Details can be found in Electronic Supplement Tab. S1. se/pe – single-end/paired-end; ss – strand-specfic. EBOV – Ebola virus; *x* h poi – total RNA extracted *x* hours post infection.

Nr.	Species	Id.	Kingdom	Study	Run	Protocol	Reads		Ref.
							num.	length	
1	*Escherichia coli*	ECO	Bacteria	PRJNA238884	SRR1173967	se, ss	7,9	94 nt	[27]
2	*Candida albicans*	CAL	Fungi	PRJNA213618	SRR1654847	pe	11,5	51 nt	[28]
3	*Arabidopsis thaliana*	ATH	Plant	PRJNA231064	SRR1049376	se	16,9	101 nt	[29]
4	*Mus musculus*	MMU	Mammal	PRJNA140057	SRR203276	pe, ss	52,6	76 nt	[10]
5	*Homo sapiens*	HSA	Mammal	ENCSR000AED	–	pe, ss	97,5	101 nt	–
Homo sapiens + EBOV		HSA-EBOV							[30]
6	*3 h poi*	-3h	Mammal+Virus	PRJNA429171	SRR6453200	pe	17,2	100 nt	
7	*7 h poi*	-7h	Mammal+Virus	PRJNA429171	SRR6453205	pe	24,7	100 nt	
8	*23 h poi*	-23h	Mammal+Virus	PRJNA429171	SRR6453206	pe	26,5	100 nt	
Simulated									
9	*Homo sapiens* Chr 1	HSA-FLUX	Mammal	–	–	pe	60,0	100 nt	–

Assembly tools performing diverse regarding different data sets and quality metrics

All evaluated assembly tools are summarized in Fig. 1 and Tab. 3. Finding the best parameter setting for each tool and each data set is obviously beyond the scope of this evaluation. Therefore, we used the default settings of each tool and adjusted only a few key parameters such as *k*-mer values and strand-specificity (see Methods for details). Full execution details and commands can be found in the Electronic Supplement, Files S3. For the tools with a built-in function to automatically merge the output of different *k*-mer values (`Oases`, `Trans-ABySS`, `IDBA-Tran`, `SPAdes-sc`; see Tab. 3), we applied a set of selected *k*-mers (for details see Files S3). If strand-specific data was used for the assembly, we applied the corresponding option of each tool. In application, one should try several different parameter settings and compare the resulting assemblies to optimize the whole assembly process. In particular, different *k*-mers should be tested and evaluated against each other [19]. Here, we carefully chose *k*-mer values to obtain a somewhat fair comparison between the assemblers, although some parameters may not be optimal.

Whenever a tool was difficult to install (e.g. due to missing dependencies) or could not be run on a specific data set, we attempted to debug the source code and in some cases also contacted the authors to solve the problem. Therefore, we also decided to share our experiences regarding the installation procedure and execution of each tool (Tab. 3).

Trans-ABySS.

Compared to the other tools, `Trans-ABySS` [9] achieved the highest re-mapping rates (98.56 % for *C. albicans*, 99.55 % for the simulated data; Fig. S4), however arranges only within the midfield or worse regarding the optimal score calculated by `TransRate` [39]. The *percentage of good mappings* and *uncovered bases* are generally bad (Tab. 2 and Tab. S6). On the other hand, the assemblies produced by `Trans-ABySS` achieved the best RSEM-EVAL scores (calculated with `DETONATE` [38]) over all nine data sets. No other assembler outperformed `Trans-ABySS` regarding this metric (Tab. S8). Therefore, the transcripts constructed by `Trans-ABySS` are well supported by the reads, used to build the assembly. `Trans-ABySS` performed good in all `BUSCO` [40] analyses and showed a high amount of complete (C) ortholog detections (Fig. 3, Fig. S7). However, many hits occur multiple times (complete and duplicated), for example in the *C. albicans* assembly (Fig. S7). This might be a result of the multiple *k*-mer approach (MK), when too many potential isoforms are assembled and not merged accurately at the end of the assembly process. We observed similar results for the MK runs of `Oases` [11]. Regarding the amount of fragmented (F) and missing (M) BUSCOs, `Trans-ABySS` arranges among the best performing tools (Fig. 3). `Trans-ABySS` achieved the highest *OMS* (see Methods for definition) of 60 of all assembly tools (Fig. 2) and performed best for the EBOV-infected human data sets and the simulated data of human chromosome 1. The lowest metric score was achieved for the bacterium data set (Fig. 2).

SPAdes-sc and -rna.

Although initially designed for single-cell and smaller bacterial-sized genome assemblies, we also included `SPAdes` [41] in our evaluation. It has previously been reported that when used in single-cell mode, the assembler achieves good results with RNA-Seq data [36]. This may be due to the uneven coverage optimization implemented for single-cell data, which also fits very well with the behavior of low and high-level expressed transcripts. Additionally, `SPAdes` has a special RNA-Seq mode. Therefore, we evaluated the performance of `SPAdes` in single-cell (`--sc`; `SPAdes-sc`) and transcriptome (`--rna`; `SPAdes-rna`) mode (Files S3) and present here the results of both parameter options together.

The re-mapping rates for both `SPAdes` parameter options are on a comparable level and arrange among the top mapping rates for all data sets (87.66–97,25 %, Fig. S4), although in `--rna` mode only one single *k*-mer can be used. Based on the `TransRate` metrics, `SPAdes` build the most accurate assemblies (Tab. S6). For almost all data sets, the `SPAdes-sc` and `-rna` assemblies achieved the highest *percentage of good mappings*, the lowest *percentage of uncovered bases*, and a low up to a moderate *amount of ambiguous bases* followed by `SOAPdenovo-Trans` [13]. The RSEM-EVAL scores of the `SPAdes` assemblies vary widely among the different RNA-Seq data sets. For some samples, `SPAdes-sc` achieves a better scoring than `SPAdes-rna`, and vice versa (Tab. S8). `SPAdes` assemblies arrange in the midfield of BUSCO detections, with the `--sc` mode performing generally better than the `--rna` mode (Fig. 3). When only comparing the amount of complete single-copy orthologs, the `SPAdes` assemblies generally outperform the other tools. Most likely due to the single *k*-mer used in `--rna` mode, `SPAdes-rna` assembled a lower amount of BUSCOs for some data sets (Fig. S7). `SPAdes-sc` is among the best performing tools for the detection of complete single-copy BUSCOs in the *C. albicans* transcrip-

Table 2. Here we show all 20 selected metrics (rows) based on the output of rnaQUAST [36], Hisat2 [37], DETONATE [38], TransRate [39] and BUSCO [40] for the transcripts assembled by all ten assembly tools (columns). Results are shown for the non-infected *Homo sapiens* RNA-Seq strand-specific paired-end library with read length 101 bp (accession number ENCSR000AED). In each row the three best-ranked values that contributed to the *metric score* (*MS*; Methods) are indicated with ***bold italic***. Details and much more statistics complementing this evaluation can be found in the Electronic Supplement, Fig. S4–Tab. S8. Summaries for all other data sets can be found in Tab. S9. The RSEM-EVAL score is multiplied by 10^9. The *Number of ambiguous bases* is given in thousand. N50 – the length of the shortest contig in the assembly, so that the accumulated bases of all contigs of this length or longer cover 50 % of all the bases in the assembly. F1 score – an F1 score of 1 would mean that all nucleotides/contigs in the estimated true assembly were recovered with at least 90 % identity. KC score – *k*-mer compression score reflecting the similarity of each assembly to DETONATEs estimated "true" assembly.

		Trinity	Oases	Trans-ABySS	SOAP-Trans	Bridger	BinPacker	IDBA-Tran	Shannon	SPAdes-sc	SPAdes-rna
	k-mer size	default	25,35,45,55,65	25,35,45,55,65	default	default	default	25,35,45,55,65	default	default	default
Evaluation metrics 1–20											
Hisat2											
1	Overall mapping rate	***91.66***	88.04	***98.36***	89.93	86.83	72.6	64.61	84.27	90.26	***90.76***
rnaQUAST											
2	Transcripts ≥1,000 nt	***72685***	***207474***	***68662***	27529	43201	22611	23516	31328	26245	15945
3	Database coverage	***0.23***	0.08	***0.29***	***0.1***	0.07	0.06	0.09	0.01	0.09	0.1
4	Misassemblies	2739	216128	2878	***279***	7329	5603	***302***	2837	1566	***570***
5	Mismatches per transcript	1.04	1.25	***0.61***	***0.27***	1.44	4.63	0.67	1.26	0.82	***0.41***
6	Average alignment length	***781.94***	343.48	258.13	218	654.41	***2335.73***	487.11	***711.83***	429.36	207.92
7	Mean isoform coverage	***0.52***	0.33	***0.49***	0.27	0.33	***0.7***	0.35	0.28	0.34	0.28
TransRate											
8	N50	1613	1230	1913	***3391***	1386	***3511***	566	1446	641	***16469***
9	Reference coverage	***0.23***	0.09	***0.27***	***0.09***	0.09	0.07	0.08	0	0.08	0.09
10	Mean ORF percentage	51.47	42.09	51.09	48.02	45.1	42.57	***52.46***	***55.7***	48.28	***55.04***
11	Optimal score[a]	0.08	0.02	0.08	***0.27***	0.14	0.07	0.25	0.07	***0.32***	***0.35***
12	Percentage good mappings[a]	0.22	0.06	0.17	***0.59***	0.32	0.26	0.49	0.22	***0.63***	***0.64***
13	Percentage bases uncovered[a]	0.66	0.94	0.66	0.33	0.42	0.84	***0.02***	0.5	***0.04***	***0.31***
14	Number of ambiguous bases	306314	843235	460747	241236	206635	***72918***	***138699***	***117068***	159111	186834
DETONATE											
15	Nucleotide F1	0.4	0.18	0.49	***0.57***	0.48	0.15	0.55	0.35	***0.58***	***0.56***
16	Contig F1	0.02	0.02	***0.2***	***0.21***	0.01	0	0.02	0.02	0.02	***0.09***
17	KC score	***0.49***	0.24	***0.56***	0.37	0.4	0.37	0.29	***0.42***	0.36	0.33
18	RSEM EVAL	-6.63	***-1.18***	-6.22	-9.03	-7.71	***-1***	-1.63	-8.95	-1.38	***-1.34***
BUSCO											
19	Complete single-copy BUSCOs	1401	1321	2079	2151	***2360***	1010	1677	2302	***2347***	***2551***
20	Missing BUSCOs	***1810***	1922	***1772***	2164	***1812***	4078	2615	2133	2457	2392

[a] Not available for the *E. coli* and *A. thaliana* data set because this metric is only calculated by TransRate in the case of paired-end data.

Table 3. Overview of the different *de novo* assembly tools evaluated in this study sorted by the year of publication. We further rated our experiences regarding the installation and usability of each tool (😄 – excellent, 🙂 – good, ☹ – unsatisfactory). These experiences might be subjective, nevertheless we want to share them to give non-experienced users an idea of how difficult it is to get each tool installed (*Setup*) and executed (*Usage*), see Methods for details. More details about runtime and memory consumption can be found in Electronic Supplement Fig. S10. MK – Whether or not the tool has a built-in multiple *k*-mer approach and is able to automatically merge the output of different *k*-mer runs.

Assembler	Version	MK	Setup	Usage	Runtime			Memory	Ref.	Year
					min	max	median	median		
Trans-ABySS	1.5.5	yes	☹	🙂	16 m	2 d 6 h 56 m	11 h 14 m	19.7 GB	[9]	2010
Trinity	2.3.2	no	🙂	😄	28 m	1 d 14 h 35 m	5 h 3 m	18.4 GB	[10]	2011
Oases[a]	0.2.08	yes	🙂	🙂	25 m	8 d 15 h 45 m	6 h 47 m	31.3 GB	[11]	2012
SPAdes-sc[b]	3.9.1	yes	😄	😄	10 m	7 h 45 m	2 h 33 m	25.3 GB	[41]	2012
SPAdes-rna[b,c]	3.9.1	no	😄	😄	9 m	9 h 10 m	2 h 20 m	19.5 GB	[41]	2012
IDBA-Tran	1.1.1	yes	🙂	🙂	7 m	8 h 49 m	2 h 44 m	9.6 GB	[12]	2013
SOAPdenovo-Trans	1.03	no	🙂	☹	1 m	1 h 48 m	24 m	26.4 GB	[13]	2014
Bridger[d]	2014-12-01	no	🙂	☹	11 m	21 h 11 m	5 h 9 m	30.4 GB	[14]	2015
BinPacker[d]	1.0	no	🙂	😄	5 m	15 h 57 m	3 h 3 m	27.9 GB	[15]	2016
Shannon	0.0.2	no	☹	🙂	9 m	10 h 45 m	3 h 18 m	83.6 GB	[16]	2016

[a] Oases was used on top of the *de novo* genome assembler Velvet (v1.2.10) [42].

[b] SPAdes, originally designed as a *de novo* genome assembler for single-cell data, was used in single-cell modus (-sc) and RNA-Seq modus (-rna).

[c] When running SPAdes in RNA-Seq modus, only a single *k*-mer value is allowed.

[d] Based on a splicing graph construction instead of *de Bruijn* graphs.

tome (Fig. 3).

SPAdes-sc arranges within the top three tools regarding our summed up metric score (*OMS*=53.3, Fig. 2), only outperformed by Trans-ABySS (*OMS*=60) and reached one of the highest metric scores for the *C. albicans* assembly (12 points). Comparable to SPAdes in single-cell mode, SPAdes-rna performed generally good on almost all data sets. However, the lowest scores of 5 were achieved for the two single-end data sets. It seems that SPAdes in general and especially in --rna mode performs better on paired-end data. SPAdes-rna reached also one of the highest metric scores (12) for the human data set without any infection (Tab. 2; last column). For this data set, the single-cell mode of SPAdes achieved a low *MS* of only 5 (Fig. 2). Based on these observations, we suggest that for larger eukaryotic RNA-Seq data sets the RNA mode of SPAdes should be applied.

It remains questionable, why SPAdes in --rna mode does only work on a single *k*-mer size (55 by default), although in genome assembly mode multiple *k*-mers can be used. In the online manual, the authors strongly recommend to not change this parameter [43]. However, the algorithm might be further improved by also allowing multiple *k*-mer values in the --rna mode of SPAdes.

Trinity.

The re-mapping rate of Trinity [10] was generally high and arranged between 85.56 % for *E. coli* and 97.29 % for *C. albicans* (Fig. S4). Trinity performed well regarding the TransRate metrics and DETONATE RSEM-EVAL scores on almost all data sets except HSA-EBOV-3h (Tab. S6 and S8). The assembler detected many complete BUSCOs for most of the data sets (Fig. 3). However, for the eukaryotic data sets, approximately the half amount of complete BUSCOs is included multiple times in the assembly. This might be a result of the sub-graphs Trinity relies on to detect different isoforms of one transcript [10]. Trinity achieved one of the top three *OMS* values (50.3) and the best *OMS* for the *M. musculus* data set (Fig. 2), that was among others used for the evaluation of the tool in the original Trinity paper [10]. Interestingly, Trinity performed generally good in constructing human transcripts out of the virus-infected data sets, except for the 3 h sample (2/20).

SOAPdenovo-Trans.

The re-mapping rate of SOAPdenovo-Trans [13] was generally high (>85 %), except for the *E. coli* data set (Fig. S4). SOAPdenovo-Trans performed quite well regarding most TransRate statistics and the calculated optimal score (Tab. S6). In most of the cases, only the SPAdes assemblies could outperform SOAPdenovo-Trans regarding the TransRate metrics. The RSEM-EVAL scores vary depending on the assembled RNA-Seq data set (Tab. S8). For the HSA-EBOV-23h and *M. musculus* sample SOAPdenovo-Trans achieved good RSEM-EVAL scores, whereas for the bacterial, the fungal, the plant and the simulated RNA-Seq data the tool places among the last three assemblers. The amount of complete and duplicated BUSCOs is very low (Fig. 3), which correlates with the tools ability to detect different isoforms (see *mean isoform coverage* calculated with rnaQUAST, Fig. S5). However, this could be also a result of the single *k*-mer approach. SOAPdenovo-Trans achieved a good *OMS* of 45.6 (Fig. 2). The assembler performed well on each evaluated data set (*MS* between 6–9) and only showed a lower metric score for the artificial data set of human chromosome 1 (3/20).

IDBA-Tran.

IDBA-Tran [12] had the lowest re-mapping rates for most data sets (Fig. S4) except for *A. thaliana* (89.04 %). However, the TransRate metrics of the IDBA-Tran assemblies are generally good (Tab. S6). Comparable to SOAPdenovo-Trans, some of the IDBATran results arrange within the top three assemblies regarding the optimal score calculated by TransRate. DETONATE RSEM-EVAL scores reveal a different picture, as IDBA-Tran is placed last regarding this metric and never reaches the top five (Tab. S8). Furthermore, IDBA-Tran is one of the tools with the lowest amount of complete BUSCOs and the highest amount of fragmented and missing BUSCOs (Fig. 3 and Fig. S7). IDBA-Tran is placed in the midfield of all metric scores (*OMS*=40.3, Fig. 2) and showed the best performance for the artificial data set (8/20), the *E. coli* data (7/17), and the *M. musculus* data (7/20).

Oases.

The re-mapping rates of Oases [11] were generally good (>85 %). However, they dropped for the simulated human data

(73.26 %), the HSA-EBOV-23h data (70.05 %) and the *E. coli* data (49.16 %) below acceptable thresholds (Fig. S4). Oases introduced the highest amount of ambiguous bases in the assemblies and arranges among the last places regarding the TransRate statistics (Tab. S6). Oases assemblies place in the last third regarding the RSEM-EVAL scores calculated by DETONATE. However, a comparable amount of complete BUSCOs could be detected, but many duplicate hits are still included, which could be again a result of the MK approach (Fig. 3). Oases performed best for the human simulated data (7/20), the EBOV-infected samples (6/20) and the plant data (6/20). The calculated metric score for the HSA-EBOV-7h data set might be comparatively low, because we were not able to calculate and include rnaQUAST [36] statistics for this assembly (see Methods). Oases achieved only an *OMS* of 29.6 (Fig. 2).

Bridger.

Bridger [14] assemblies resulted generally in high re-mapping rates between 87.35 % (*E. coli*) and 96.72 % (*C. albicans*, Fig. S4). For almost all TransRate metrics, the Bridger assemblies arrange in the midfield of scores and are already far away from the top scores produced by the SPAdes, SOAPdenovo-Trans and IDBA-Tran assemblies (Tab. S6). According to the RSEM-EVAL scores, Bridger is performing generally well among the top four tools (Tab. S8). Furthermore, Bridger performed well in the detection of complete BUSCOs with a moderate amount of duplicated hits. The amount of missing BUSCOs is comparably low (Fig. 3, Fig. S7). However, the general performance of the tool is comparatively humble, underlined by the lowest overall metric score of all assemblers (*OMS*=22, Fig. 2).

BinPacker.

The re-mapping rates of BinPacker [15] vary a lot (36.6–96.7 %, Fig. S4). The TransRate metrics of the BinPacker assemblies are comparable to the Bridger results, placing BinPacker among the low-performing tools according to the TransRate statistics (Tab. S6). On the other hand, BinPacker introduces only a low amount of ambiguous bases in the assemblies. The RSEM-EVAL score is comparatively low, except for the human simulated data where BinPacker achieves a scoring similar to Bridger and reaches the third place behind Trinity and Trans-ABySS (Tab. S8). Regarding the detection of orthologs, BinPacker had the lowest performance of all tools and was only able to assemble a reasonable amount of complete BUSCOs for *C. albicans*, HSA-EBOV-7h and the human simulated data set (Fig. 3 and Fig. S7). Overall, the performance of BinPacker (*OMS*=28, Fig. 2) is similar to Bridger, which is not surprising, because the assembler is build on the same principals. BinPacker showed a more consistent behavior than Bridger regarding the metric scores however only reached scores between 3 and 5 (Fig. 2).

Shannon.

The most variant re-mapping rates were observed for Shannon [16], ranging between 30.77 % for the human simulated data set and 96.51 % for *A. thaliana* (Fig. S4). The Shannon assemblies do not result in good TransRate optimal scores, however the percentage of uncovered bases is placed in the midfield of all scorings and Shannon does not introduce that many ambiguous bases in the assembled transcriptome (Tab. S6). The RSEM-EVAL scores of Shannon vary among the assembled data sets (Tab. S8). Regarding the amount of assembled complete BUSCOs, Shannon arranges in the midfield and showed a relatively high amount of duplicated hits (Fig. 3). Shannon achieved one of the lowest *OMS* (22.6, Fig. 2).

When designing this study, we also aimed to include an assembly tool that is not based on *k*-mers. Mira [44] (v4.0rc5) uses an overlap-consensus-graph for assembly and can be executed in EST mode for RNA-Seq data. However, for one human sample 62 h runtime were needed, >300 GB temporary files were produced and ~130 GB RAM consumed. Furthermore, we were not able to detect any BUSCO hits in the Mira assemblies. Due to this low performance and high runtime and memory consumption, we decided to remove the tool from our comparison.

Usability

We rated our experiences regarding the installation and usability of each tool (Tab. 3). These experiences may be subjective, but we want to share them to give inexperienced users an idea of how difficult it is to install and run each tool. Some of the tools rely on many dependencies and/or are difficult to compile (Shannon, SOAPdenovo-Trans, Trans-ABySS), at least on our test system without administrative permissions, while others could be installed easily (SPAdes). Furthermore, some assemblers need additional parameter files for execution (SOAPdenovo-Trans), are circuitous to run (Trans-AbySS, Oases, SOAPdenovo-Trans), need additional preprocessing steps of the reads (IDBA-Tran assumes paired-end reads to be in order *forward–reverse*), or are just not terminating for all data sets (Bridger), while with others we had no problems and could execute them straightforward (Trinity, SPAdes, BinPacker).

Bridger failed in the path search step for some of the generated temporary files. Therefore we performed the last step of Bridger by manually combining the transcript output. Furthermore, we had to start Bridger two times for each data set, because the tool crashed each time after the first start, but continued with the assembly when started a second time on the same output folder (Files S3).

In the past, Oases and Trans-ABySS were always circuitous to run, because the corresponding genome assemblers Velvet [42] and ABySS [45] needed to be executed first with multiple *k*-mers. These difficulties were somehow emasculated by new wrapper scripts provided by the developers to automatically execute the underlying genome assemblers.

Computational efficiency

Since *de novo* transcriptome assembly can involve the analysis of large sequencing data, computational efficiency is an important benchmark, especially for deep sequencing projects and large sample sizes. Furthermore, it is highly recommended to run multiple assemblies with different tools and parameter settings (for example different *k*-mers), so computation time is an important part of each tool. Tab. 3 summarizes the computational time and the median memory consumption of all data sets and assemblers. Details can be found in Electronic Supplement Fig. S10.

Runtime.. SOAPdenovo-Trans appeared to be the fastest algorithm with a median runtime of only 24 m, followed by SPAdes-rna (2 h 20 m), SPAdes-sc (2 h 33 m), IDBA-Tran (2 h 44 m), BinPacker (3 h 3 m), Shannon (3 h 18 m), and Bridger (5 h 9 m) (Tab. 3, Fig. S10). Older tools such as Oases (6 h 47 m) and Trans-ABySS (11 h 14 m), that are additionally based on a multiple *k*-mer strategy (MK), are comparatively slower. If those tools would be executed only on one *k*-mer, the runtime would be comparable with the other assemblers or even faster. SOAPdenovo-Trans can also run on different *k*-mers, but no automatic merge function for the different assemblies is implemented. The Trinity runtime (5 h 3 m) lays between the faster tools and the slower MK approaches, although the tool relies on one *k*-mer only. Although based on a MK strategy, IDBA-Tran and SPAdes are much faster than the older MK algorithms and

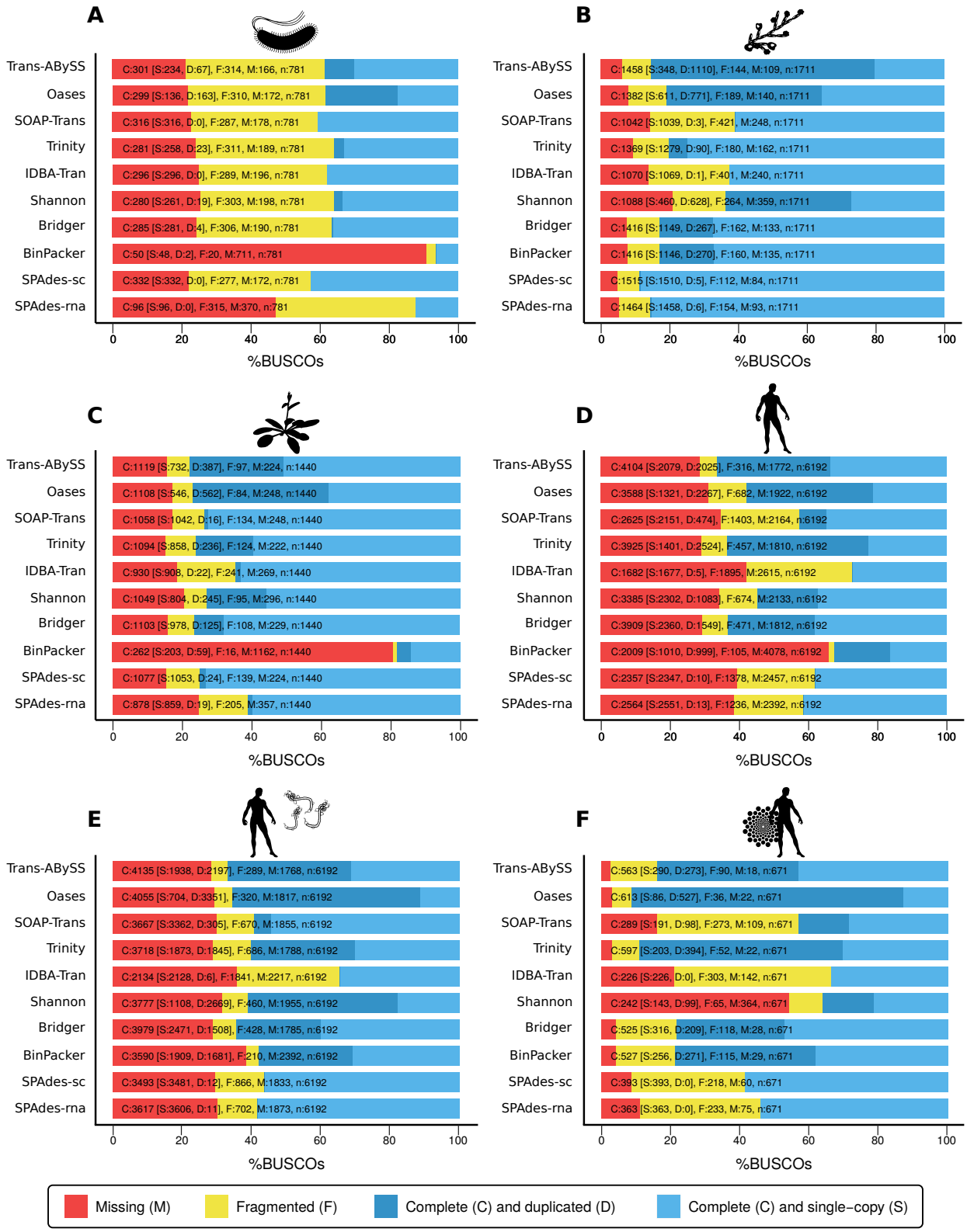

Figure 3. Selected BUSCO (*benchmarked universal single-copy orthologs*) [40] assessment results for *E. coli* **(A)**, *C. albicans* **(B)**, *A. thaliana* **(C)**, *H. sapiens* **(D)**, HuH7 cells infected with EBOV 7 h post infection **(E)** and `flux` simulated reads [26] of human chromosome 1 **(F)**. The numbers indicate the absolut amount of complete (C) and single-copy (S), complete and duplicated (D), fragmented (F), and missing (M) BUSCOs (see Methods for details). BUSCO results for all other data sets can be found in the Electronic Supplement, Fig. S7.

can compete against the other single-k-mer tools in the sense of speed.

Memory consumption.. IDBA-Tran appeared to be the tool with less memory consumption estimated over all data sets (median 9.6 GB, Tab. 3, Fig. S10). Shannon showed really high memory peaks (median 83.6 GB), especially for the larger data sets (more than 100 GB for the EBOV infected human samples, see Fig. S10), followed by Oases (31.3 GB), Bridger (30.4 GB) and BinPacker (27.9 GB).

When running Trinity we observed in the first phase of assembly (meaning in the first seconds up too few minutes, depending on the size of the input data set) very high memory peaks. For example, in the first five minutes of execution of all human data sets we noticed memory peaks of ~240 GB with Trinity. Immediately after this initial peak, the memory consumption dropped down to a comparatively normal level, see Fig. S10. We removed the high initial memory peaks for Trinity from our comparison to achieve a better overview of the memory usage of all assemblers. The high memory consumption in the first phase might be a result of the many individual *de Bruijn* graphs build by Trinity based on partitions of the sequence data [10].

Users should pay particular attention to planning enough processing power and time when using many tools for different parameter settings, especially when working on projects with high sequencing depth and large sample size.

Contamination of viruses drops performance of most assembly tools

Although not the main focus of this study, we were interested in how the assemblers work with RNA-Seq data as virus contamination increases, and whether they are still able to construct complete viral genomes. Therefore, we used Blast to search for contigs in the virus-infected assemblies (Fig. 1) that match the full genome of the Ebola virus (EBOV). The EBOV genome comprises a single-stranded RNA genome with negative orientation and a size of approximately 19 kb [46]. We assembled three human samples infected with EBOV at three different time points (Methods). Therefore, we were able to investigate how the different assemblers perform on increasing amounts of viral reads in the data (3 h: ~0.1 % viral reads, 7 h: ~2 %, 23 h: ~20 %; compare [30]).

Surprisingly, the performance of most assembly tools in constructing the viral RNA genome decreased with a higher amount of viral reads. For example, Trans-ABySS was able to construct the full EBOV genome out of the 3 h (18,926 nt alignment length, 99.984 % sequence similarity) and 7 h (18,903 nt, 99.974 %) data set, but failed on the 23 h data set (many small contigs, longest hit: 8,500 nt). In general, Trans-ABySS, SOAPdenovo-Trans, Shannon, Bridger, BinPacker, and SPAdes (--sc and --rna mode) performed well and constructed the full EBOV genome out of the 3 h data set. On the 7 h data set (~2 % viral reads), Trans-ABySS and SOAPdenovoTrans performed best. Bridger and BinPacker were only able to construct the same 10 kb partial EBOV genome. SPAdes-sc and SPAdes-rna assembled viral contigs up to a length of 12 kb and 14 kb, respectively. After 23 h post infection and a viral read contamination of ~20 %, only SOAPdenovo-Trans was able to construct the full EBOV genome with a high accuracy (18,901 nt, 99.53 %). Bridger and BinPacker constructed contigs of a length of 14.8 kb and 12 kb, respectively. Interestingly, Trinity was the only tool that was not able to construct any full-length EBOV genome out of the three human/virus-mixed data sets.

Discussion

Although the evaluation of *de novo* transcriptome assemblies was frequently performed in the past [6, 17–23], there is still a lack of knowledge which assembler should be used for which kind of RNA-Seq data. Furthermore, these studies rely on limited data sets (e.g. a single species, a single sequencing protocol) or focus only on a subset of all currently available assembly tools. Here, we present a comprehensive evaluation of ten *de novo* assembly tools across various RNA-Seq data sets of different kingdoms of life.

Using a combination of biological-based and reference-free metrics to evaluate an assembly

We evaluated biological/reference-based metrics and statistical/reference-free metrics only based on the input read data and the final assembly itself. Evaluation metrics are very important to assess the quality of a genome or transcriptome assembly. However, there is a lack of consensus which evaluation metrics work best for *de novo* transcriptome assembly.

For example, Rana *et al.* [47] compared different assemblers and k-mer strategies using killifish RNA-Seq data and based their comparisons on eleven selected metrics, such as contig number, N50 value, contigs >1 kb, re-mapping rate, number of full-length transcripts, number of open reading frames, DETONATEs RSEM-EVAL score and the percentage of alignments to closely related fish. Another study performed comparisons on peanut RNA-Seq data and evaluated the assemblies on metrics such as N50, average contig length, number of contigs and the number of full-length transcripts [48]. Moreton *et al.* also used the N50 length, the number of transcripts, the number of transcripts $\geq$1 kb and RMBT and CEGMA percentages when evaluating different assemblies of duck [49]. Surely, more information on which metrics best predict the quality of a *de novo* transcriptome assembly would help to establish "best practice" protocols that could be further utilized to develop automatic evaluations to improve assemblies.

There is still a general lack of which metrics should be used for an appropriate evaluation of *de novo* transcriptome assemblies. More complicating, we observed that some metrics are contradicting each other, such as the *optimal assembly score* calculated by TransRate [39] and the RSEM-EVAL score of DETONATE [38]. For example, assemblies of the *Homo sapiens* simulated data set achieved the best RSEM-EVAL scores for Shannon, Trans-ABySS, and Trinity, whereas IDBA-Tran performed worst (Tab. S8 and S9). However, IDBA-Tran achieved the third-best optimal TransRate score, only outperformed by the two SPAdes assemblies, and Shannon arranges on the next-to-last place regarding this metric (Tab. S6 and S9).

We conclude, that a careful selection of biological-based and reference-free evaluation metrics is necessary to select the best performing results out of multiple assembly runs. Based on our observations, we suggest to initially use reference-free metrics as provided by the TransRate [39] software. Generally, TransRates *optimal assembly score* seems to be a good measure for the quality of an assembly. Assemblies, that needed fewer contigs for a comprehensive description of the whole transcriptome achieved in most cases also good TransRate scores (Tab. S6). However, this score can be only calculated for paired-end RNA-Seq data at the moment.

If biological/reference-based metrics should be included, the *mean isoform coverage* calculated by rnaQUAST [36] as well as the scores calculated by BUSCO [40] are good metrics for the evaluation of the best assembly results.

Different species and RNA-Seq setups require specialized assembly tools

Although no tool performed dominantly best for all data sets, we found that Trans-ABySS [9], Trinity [10], SOAPdenovo-Trans [13] and SPAdes [41] produced consistently good assemblies among all data sets (Fig. 2).

SPAdes, although originally developed as *de novo* assembly tool for small genomes, produced also highly accurate transcriptome assemblies in both modes, for single-cell (SPAdes-sc) and RNA-Seq data (SPAdesrna). Interestingly, the single-cell mode outperformed the RNA mode for some of the data sets (Fig. 2). This might be a result of the missing multiple *k*-mer approach in the RNA mode. Therefore, a further improvement of the transcriptome assembly mode of SPAdes, by also taking advantage of multiple *k*-mers like already implemented in the genome and single-cell mode might further improve the performance for *de novo* transcriptome assembly construction. Taking a closer look at the BUSCO results, SPAdes produced in both modes the lowest amount of complete and duplicated transcripts (Fig. 3). This could indicate that SPAdes merge highly similar transcripts into single contigs, therefore losing similar isoforms. This behavior can be also observed when looking at the mean isoform coverage calculated with rnaQUAST (Fig. S5 and Tab. S9). Especially, SPAdes-rna does not perform well in constructing different isoforms, most likely a result of the missing multiple *k*-mer approach.

On closer examination of the BUSCO results (Fig. 3), Oases [11] performed well overall. However, the tool produced the highest amount of complete and duplicated hits, which might indicate that highly similar isoforms derived from the multiple *k*-mer approach are not efficiently merged. For all data sets, Oases constructed the highest amount of contigs, however did not achieve the best database coverage in all test cases. For example the Oases assembly of the *H. sapiens* data set comprises ~207,000 transcripts with a length >1000 bp, covering only 8 % of the reference transcripts (Tab. 2). In comparison, the Trans-ABySS assembly needed only ~68,000 contigs to achieve a database coverage of 29 % (Tab. S9). Therefore, Oases has the potential to create good assembly results, but also produces big assemblies with many contigs that might complicate and confuse downstream analyses.

With an average runtime of only 24 minutes over all data sets, SOAPdenovo-Trans [13] outperformed all other assemblers (Tab. 3, Fig. S10). Combined with the moderate memory consumption (median 26.4 GB), this makes SOAPdenovo-Trans the most resource-efficient tool evaluated in this study. However, it might be interesting to run multiple *k*-mer assemblies with SOAPdenovo-Trans and use another assembly merge strategy (e.g., conducted from Oases or TransABySS) to merge the final transcripts resulting from each run. In general, multiple *k*-mer approaches (Trans-ABySS, SPAdes-sc, IDBA-Tran, Oases) performed better than single *k*-mer approaches regarding isoform detection and assembly completeness.

As long as the amount of viral contamination in RNA-Seq data is low (~0.1 %), all assembly tools except Trinity, Oases, and IDBA-Tran generated accurate viral contigs with high similarity to the EBOV genome and a length >18 kb. Generally, SOAPdenovoTrans performed best on all three virus infected data sets by constructing accurate full-length contigs with high similarity to the EBOV genome. Therefore, it could be interesting to evaluate the performance of SOAPdenovo-Trans for the construction of RNA viral genomes out of meta-transcriptomic RNA-Seq data in the future.

Potential implications

Here, we present a large-scale comparative study by applying ten *de novo* assembly tools to nine RNA-Seq data sets comprising different kingdoms of life (Fig. 1). Overall, we calculated more than 200 single assemblies and evaluated their performance on different metrics (Tab. 4). All results are summarized in a comprehensive Electronic Supplement, that is easily extendible by more RNA-Seq data sets, new assembler versions, parameter settings and tools. We summarize some key findings from our comparative study:

(I) No tool performed dominantly best for all data sets. However, Trans-ABySS, Trinity, SOAPdenovo-Trans and SPAdes were among the best assembly tools (Fig. 2).
(II) SOAPdenovo-Trans performed best for the construction of the Ebola virus RNA genome at all three time points tested.
(III) SOAPdenovo-Trans had the lowest runtime, followed by IDBA-Tran, BinPacker, SPAdes and Bridger.
(IV) For assembly evaluation, we recommend a hybrid-approach by combining biological-based (e.g. BUSCO [40]) and reference-free metrics (e.g. TransRate [39], DETONATE [38]).

Limitations and future work

We still recommend applying different tools and parameter settings for *de novo* transcriptome assembly, followed by the evaluation of the output transcripts and selecting the best-performing results. This general idea needs to be investigated in more detail in future studies. Choosing the best assemblies based on appropriate metrics and the subsequent clustering procedure (without losing isoforms and introducing greater redundancy) are still challenging and open tasks.

Dynamic extension of this comparison.
A common problem of many comparative studies is that they can only provide limited suggestions based on the tools and data sets that have been available at the time point they were carried out. The Electronic Supplement presented here will be extended by other metrics, data sets, and assembly tools in future updates.

Cluster assembly.
Furthermore, the complementary performance of the top performing tools motivated the development of an ensemble method by combining the best performing methods to achieve an overall better assembly. Based on our findings, a pipeline should be developed, that automatically selects the top performing assemblies (or only the best transcripts from each assembly) using a hybrid approach of biological-based and reference-free metrics and clusters them based on sequence similarity to achieve a more comprehensive assembly.

For the large bioinformatics community working in the area of RNA-Seq, the development of a high-performing (accurate and fast) *de novo* transcriptome cluster workflow to automatically select and combine the output of top-performing assembly tools remains a challenging however crucial task.

Methods

Description of assembly tools and executed commands

We collected ten *de novo* assembly tools for the transcriptome reconstruction of the nine RNA-Seq data sets (Tab. 1), summarized in Tab. 3 and Electronic Supplement Tab. S2.

Out of those, six transcriptome assemblers are especially

designed to work on RNA-Seq data and are based on *de Bruijn* graphs: Trans-ABySS [9], Trinity [10], Oases [11], IDBA-Tran [12], SOAPdenovo-Trans [13], and Shannon [16].

Trans-ABySS and Oases are built on top of the *de novo* genome assemblers ABySS [45] (v1.5.1) and Velvet [42] (v1.2.10), respectively. Both support multiple *k*-mer values by running the underlying genome assembler multiple times and merging the assembled contigs. We executed Trans-ABySS (v1.5.5) and Oases (v0.2.08) with multiple *k*-mers (MK) and in strand-specific mode, if suitable (Files S3).

Trinity and SOAPdenovo-Trans (the later one build on the principles of SOAPdenovo2 [50]) are stand-alone *de novo* transcriptome assembly tools, also based on *de Bruijn* graphs but lacking an automated MK support. Whereas for SOAPdenovo-Trans different single *k*-mer values can be applied, Trinity relies on a fixed *k*-mer value of 25. Trinity (v2.3.2) was run with default parameters and, if suitable, in strand-specific mode (Files S3). For SOAPdenovo-Trans (v1.03), currently no strand-specific assembly is supported [13].

IDBA-Tran (v1.1.1), a novel assembly tool that claims to be more robust regarding uneven expression levels in RNA-Seq data [12], was run with multiple *k*-mers and has no option for strand-specific assembly (Files S3). IDBA-Tran assumes paired-end reads to be in order (->, <-; *forward–reverse*), therefore we manually converted reads if necessary.

Shannon (v0.0.2), a so-called information-optimal *de novo* RNA-Seq assembler [16], was used with a single default *k*-mer value and if suitable in strand-specific mode (--ss; Files S3).

We further included Bridger [14] (v2014-12-01) and BinPacker [15] (v1.0), two assembly tools that rely on *splicing graphs* [14] instead of *de Bruijn* graphs. Bridger provides a new framework for *de novo* transcriptome assembly, that "bridges" between techniques employed in the Cufflinks [51] pipeline and the Trinity tool, in order to overcome the limitations of Trinity. BinPacker was developed based on the principals of Bridger and utilizes similar to Shannon coverage information to efficiently dissolve corresponding isoforms. Bridger can only run with single *k*-mer values between 19 and 32 with a default of 25. We executed Bridger with the default *k*-mer and, if possible, with the strand-specific option (--SS_lib_type). However, for two strand-specific RNA-Seq data sets (*M. musculus*, *H. sapiens*) the tool failed and was executed in the default unstranded mode (Files S3). We observed problems with strand-specific paired-end data in this version of Bridger. The strand-specific assembly of the single-end *E. coli* data (--SS_lib_type F) was running without problems. BinPacker was executed on a single *k*-mer value and if suitable in strand-specific (-m F|RF) mode (Files S3).

We further included SPAdes [41] (v3.9.1), a widely used *de novo* genome assembler based on *de Bruijn* graphs and multiple *k*-mer values. We were interested, how good the tools optimization for single-cell assembly can be applied to RNA-Seq data and how the tool performs in contrast to the specialized transcriptome assemblers mentioned above. Since version 3.9.0, an RNA-Seq mode was implemented. We evaluated the performance of SPAdes in single-cell (--sc; SPAdes-sc) and RNA-Seq (--rna; SPAdes-rna) mode. Henceforth, we refer to SPAdes-sc and SPAdes-rna as two different assemblers, although both are based on the same tool.

In total, we calculated more than 200 single *k*-mer assemblies (Files S3). Each assembler was run on each data set (Fig. 1). If possible, multiple *k*-mers were used (Tab. 3). Trans-ABySS, Oases, and IDBA-Tran dispose a built-in functionality for multiple *k*-mers. SPAdes-sc can automatically choose multiple *k*-mers for the assembly process and was therefore executed with this default option. For the *E. coli*, *A. thaliana*, *H. sapiens*, and the artificial data sets *k*-mers 25, 35, 45, 55 and 65 were used with Trans-ABySS, Oases, and IDBA-Tran. *M. musculus* data was assembled with the *k*-mers: 25, 35, 45 and 55, because the read length is shorter in comparison to the bacterial and plant data sets. The short-read *C. albicans* data was run with *k*-mers 21, 27, 33 and 39. The EBOV infected HuH7 samples were run with *k*-mers 25, 29, 33, 37 and 41. All *k*-mer values were selected based on previous results for these data sets and in relation to the different read lengths and sequencing setups. All assemblers were run with default parameters, if not otherwise stated. Details about the execution of each tool on each data set can be found in the Electronic Supplement, Files S3.

Table 4. Selected evaluation metrics applied for each assembly and data set. Metrics highlighted in gray are biological/reference-based. All other metrics only rely on the reads used to build the assembly and/or the resulting contigs. Details can be found in the Methods.

Nr.	Tool	Selected metric	Ref.
1	Hisat2	Overall mapping rate	[37]
2	rnaQUAST[a]	Transcripts ≥1,000 nt	[36]
3		Database coverage	
4		Misassemblies	
5		Mismatches per transcript	
6		Average alignment length	
7		Mean isoform coverage	
8	TransRate	N50	[39]
9		Reference coverage	
10		Mean ORF percentage	
11		Optimal score[b]	
12		Percentage good mappings[b]	
13		Percentage bases uncovered[b]	
14		Number of ambiguous bases	
15	DETONATE	Nucleotide F1	[38]
16		Contig F1	
17		KC score	
18		RSEM-EVAL	
19	BUSCO	Complete single copy BUSCOs	[40]
20		Missing BUSCOs	

[a] Not calculated for the Oases assembly of the HSA-EBOV-7h data set.
[b] Not available for the *E. coli* and *A. thaliana* data set because only calculated by TransRate if paired-end data is available.

Evaluation metrics

We benchmarked all assembly results using five evaluation tools (Fig. 1) from which 20 metrics were selected (summarized in Tab. 4). Eight metrics are based on reference sequences and annotations, whereas the others are only based on the final assembly itself (the contigs) or the reads that were used to construct the assembly. We also evaluated the computational efficiency (runtime, memory) to assess the applicability of the tools for deeply sequenced data sets and/or large sample size.

Mapping rate.

We used Hisat2 [37] to map the quality controlled reads back to each assembly. The re-mapping rate can give first insights into the quality of a transcriptome assembly (Fig. S4), however further metrics are needed to assess a more complete picture of each assembler's performance.

rnaQUAST.

We used rnaQUAST [36] (v1.4.0) to calculate various statistics for each assembly and to demonstrate the completeness and cor-

rectness levels of the assembled transcripts. The tool was run with reference transcriptomes to calculate the *sensitivity* and *specificity* of an assembly. Furthermore, rnaQUAST calculates various bar plots and histograms to visualize basic statistics such as transcript lengths, mismatch rates and the number of transcript alignments per isoform. All plots and detailed statistics can be found in the Electronic Supplement, Fig. S5.

TransRate.
TransRate [39] examines an assembly and compares it to experimental evidence such as the reads the assembly was built on. One of our metrics relies on the optimal reference-free TransRate score that utilizes only the reads that were used to generate the assembly as an evidence (Tab. 4). Such a metric should be generally better to optimize the assembly process because the comparison to a reference will always penalize genuine biological novelty contained in the assembly. The score is produced for the whole assembly and for every single contig. Currently, the score can be only calculated for paired-end data. The score of an assembly is calculated as the geometric mean of all contig scores multiplied by the proportion of input reads that provide positive support for the assembly [39]. Thus, the score captures how confident one can be in what was assembled, as well as how complete the assembly is. The minimum possible score is 0.0, while 1.0 is the maximum score.

DETONATE.
We further used the DETONATE workflow: a pipeline for the "DE novo TranscriptOme rNa-seq Assembly with or without the Truth Evaluation" [38]. We mainly focused on DETONATEs RSEM-EVAL score. This statistically based evaluation score utilizes multiple factors, such as the compactness of the assembly and its support from the RNA-Seq reads. Therefore, the RSEM-EVAL score can be used to evaluate assemblies even when the ground truth is unknown. Assemblies with higher RSEM-EVAL scores are considered better. DETONATE was run for all assemblies as recommended in the online vignette [52]. The main metrics calculated by DETONATE can be found in Electronic Supplement Tab. S8.

BUSCO.
We benchmarked universal single-copy orthologs with BUSCO [40] (v2.0). The tool detects orthologous candidate genes in the assemblies and assesses the presence and abundance of single-copy orthologs as an evaluation criteria. The so-called BUSCOs are selected from OrthoDB orthologous groups at major species radiations requiring orthologs to be present as single-copy genes in the vast majority (> 90 %) of available species. BUSCO provides a quantitative assessment of the completeness of an assembly in terms of expected gene content. The results are further simplified into categories of (i) complete and single-copy, (ii) complete and duplicated, (iii) fragmented, or (iv) missing BUSCOs.

For the evaluation of the simulated human data set, the Euarchontoglires reference data set was reduced to BUSCO orthologs originating only from human chromosome 1 (# 671 BUSCOs). The full BUSCO output for each data set can be found in the Electronic Supplement, Fig. S7.

Calculation of evaluation scores for each assembler and data set

We investigate the performance of ten *de novo* assembly tools $a_k \in A$ ($k = \{1,\dots,10\}$) on nine RNA-Seq data sets $d_i \in D$ ($i = \{1,\dots,9\}$) using 20 pre-selected metrics $m_j \in M$ ($j = \{1,\dots,20\}$).

The *metric score MS* for an assembly tool a_k and a data set d_i can be calculated for all 20 pre-selected metrics $m_j \in M$ as

$$MS(d_i, a_k) = \sum_{j=1}^{20} S(d_i, m_j, a_k),$$

whereas S is 1 if the resulting assembly arranges within the top three results for any d_i,m_j,a_k triple:

$$S(d_i, m_j, a_k) = \begin{cases} 1, & \text{if } m_j \text{ is in the best three results} \\ 0, & \text{otherwise} \end{cases}$$

A $MS(d_i, a_k)$ of 12 would state that the assembler a_k performed for data set d_i within the top three scores of 12 out of 20 metrics (hereafter written 12/20).

To get a general overview of the performance of each assembler, we summed up the metric scores *MS* an assembler achieved for each data set $d_i \in D$ to calculate an *overall metric score* (*OMS*) for each assembler:

$$OMS(a_k) = \sum_{i=1}^{9} MS(d_i, a_k)$$

The three human RNA-Seq data sets treated with the Ebola virus 3, 7, and 23 h post infection [30] are based on the same sequencing parameters and comprise roughly the same amount of reads (Fig. 1 and Tab. S1). Due to this similarity, we decided to reduce the impact of systematic assembly errors when calculating the *OMS* for one assembly tool and used the mean of all three *MS* scores for these three data sets (Fig. 2). For example, Trans-ABySS [9] performed very well in constructing the human transcripts out of all three Ebola-infected data sets (three times 10/20), whereas IDBA-Tran [12] did not (4/20, 5/20, 4/20).

The maximum achievable metric score for the *E. coli* and *A. thaliana* data sets is 17 and not 20, because the *optimal score*, the *percentage of good mappings* and the *percentage of uncovered bases* are only calculated by TransRate [39] in the case of paired-end data. Please note, that for the HSA-EBOV-7h data set no rnaQUAST [36] statistics were calculated for the Oases [11] assembly. rnaQUAST was not able to finish the calculations when the Oases assembly for the HSA-EBOV-7h data set was included.

The calculated *metric scores* (*MS*) and *overall metric scores* (*OMS*) are summarized in Fig. 2.

Computational resources

All calculations were run on two symmetric multiprocessing servers with 14 TB storage (raid-5) and 48 CPU cores each, comprising four AMD Opteron 6238 CPUs and 512 GB RAM. Each assembly was executed on 48 threads.

Usability

We further aimed to install and run all tools without root rights on our test system (Debian GNU/Linux 8 (jessie) 64-bit). Of course, how easy a tool can be installed and executed heavily depends on the used machine, the server setup and how familiar the user is with the programing language the tool is based on. Nevertheless, it should be the goal of each public available piece of software to be as user-friendly as possible. Therefore, we collected our experiences during the installation and execution of each assembler to share our observations (Tab. 3).

Availability of supporting data and materials

This study is accompanied by a comprehensive Electronic Supplement publicly available at www.rna.uni-jena.de/supplements/assembly [35]. The electronic supplement will stay consistent with the results presented in this paper. Updates, including new assembly tools, versions, and data sets, will be marked and additionally linked on subpages online.

Declarations

Competing interests

None declared.

Funding

This work has been founded by the German Research Foundation (DFG) projects: Collaborative Research Center/Transregio 124 - "Pathogenic fungi and their human host: Networks of Interaction", subproject B5; DFG SPP-1596 - "Ecology and species barriers in emerging viral diseases"; and CRC 1076 "AquaDiva", subproject A06.

Authors' contributions

MM conceived the research idea. MH designed the project, performed calculations and analysis, interpreted the data and wrote the main manuscript. MM contributed in discussions and in proofreading the final manuscript. This work is part of the doctoral thesis of MH. All authors read and approved the final manuscript.

Acknowledgements

We delightfully thank the team of the HPC-Cluster ARA, located at the University of Jena, that was very helpful in speeding up preliminary calculations and tests.

References

1. Wang Z, Gerstein M, Snyder M. RNA-Seq: a revolutionary tool for transcriptomics. Nat Rev Genet. 2009 Jan;10:57–63.
2. Martin JA, Wang Z. Next-generation transcriptome assembly. Nat Rev Genet. 2011 Sep;12:671–682.
3. Corney DC. RNA-Seq using Next Generation Sequencing. Mater Methods. 2013;3:203.
4. Conesa A, Madrigal P, Tarazona S, Gomez-Cabrero D, Cervera A, McPherson A, et al. A survey of best practices for RNA-seq data analysis. Genome biology. 2016;17(1):13.
5. Hrdlickova R, Toloue M, Tian B. RNA-Seq methods for transcriptome analysis. Wiley Interdisciplinary Reviews: RNA. 2017;8(1):e1364.
6. Sahraeian SME, Mohiyuddin M, Sebra R, Tilgner H, Afshar PT, Au KF, et al. Gaining comprehensive biological insight into the transcriptome by performing a broad-spectrum RNA-Seq analysis. Nature communications. 2017;8(1):59.
7. Haas BJ, Zody MC. Advancing RNA-Seq analysis. Nat Biotechnol. 2010 May;28:421–423.
8. Lima L, Sinaimeri B, Sacomoto G, Lopez-Maestre H, Marchet C, Miele V, et al. Playing hide and seek with repeats in local and global *de novo* transcriptome assembly of short RNA-Seq reads. Algorithms for molecular biology : AMB. 2017;12:2.
9. Robertson G, Schein J, Chiu R, Corbett R, Field M, Jackman SD, et al. *De novo* assembly and analysis of RNA-seq data. Nat Methods. 2010;7(11):909–912. Available from: http://dx.doi.org/10.1038/nmeth.1517.
10. Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, et al. Full-length transcriptome assembly from RNA-Seq data without a reference genome. Nat Biotechnol. 2011;29(7):644–652. Available from: http://dx.doi.org/10.1038/nbt.1883.
11. Schulz MH, Zerbino DR, Vingron M, Birney E. Oases: robust *de novo* RNA-Seq assembly across the dynamic range of expression levels. Bioinformatics (Oxford, England). 2012 Apr;28:1086–1092.
12. Peng Y, Leung HCM, Yiu SM, Lv MJ, Zhu XG, Chin FYL. IDBA-tran: a more robust *de novo de Bruijn* graph assembler for transcriptomes with uneven expression levels. Bioinformatics (Oxford, England). 2013 Jul;29:i326–i334.
13. Xie Y, Wu G, Tang J, Luo R, Patterson J, Liu S, et al. SOAPdenovo-Trans: *de novo* transcriptome assembly with short RNA-Seq reads. Bioinformatics (Oxford, England). 2014 Jun;30:1660–1666.
14. Chang Z, Li G, Liu J, Zhang Y, Ashby C, Liu D, et al. Bridger: a new framework for *de novo* transcriptome assembly using RNA-seq data. Genome Biol. 2015;16:30.
15. Liu J, Li G, Chang Z, Yu T, Liu B, McMullen R, et al. BinPacker: Packing-Based *De Novo* Transcriptome Assembly from RNA-seq Data. PLoS Comput Biol. 2016;12:e1004772.
16. Kannan S, Hui J, Mazooji K, Pachter L, Tse D. Shannon: An Information-Optimal *de Novo* RNA-Seq Assembler. bioRxiv. 2016;p. 039230.
17. Kumar S, Blaxter ML. Comparing *de novo* assemblers for 454 transcriptome data. BMC genomics. 2010 Oct;11:571.
18. Chen G, Yin K, Wang C, Shi T. De novo transcriptome assembly of RNA-Seq reads with different strategies. Science China Life sciences. 2011 Dec;54:1129–1133.
19. Zhao QY, Wang Y, Kong YM, Luo D, Li X, Hao P. Optimizing *de novo* transcriptome assembly from short-read RNA-Seq data: a comparative study. BMC Bioinf. 2011 Dec;12 Suppl 14:S2.
20. Lu B, Zeng Z, Shi T. Comparative study of *de novo* assembly and genome-guided assembly strategies for transcriptome reconstruction based on RNA-Seq. Science China Life sciences. 2013 Feb;56:143–155.
21. Clarke K, Yang Y, Marsh R, Xie L, Zhang KK. Comparative analysis of *de novo* transcriptome assembly. Science China Life sciences. 2013 Feb;56:156–162.
22. Wang S, Gribskov M. Comprehensive evaluation of *de novo* transcriptome assembly programs and their effects on differential gene expression analysis. Bioinformatics. 2017;33(3):327–333.
23. Voshall A, Moriyama EN. Next-Generation Transcriptome Assembly: Strategies and Performance Analaysis. In: Abdurakhmonov I, editor. Bioinformatics in the Erea of Post Genomics and Big Data. London: IntechOpen; 2018. p. 15–36.
24. Andrews S. FastQC: a quality control tool for high throughput sequence data.; 2010. Available online at: http://www.bioinformatics.babraham.ac.uk/projects/fastqc.
25. Schmieder R, Edwards R. Quality control and preprocessing of metagenomic datasets. Bioinformatics. 2011;27:863–864.
26. Griebel T, Zacher B, Ribeca P, Raineri E, Lacroix V, Guigó R, et al. Modelling and simulating generic RNA-Seq experiments with the flux simulator. Nucleic Acids Res. 2012;40:10073–10083.
27. Thomason MK, Bischler T, Eisenbart SK, Förstner KU, Zhang A, Herbig A, et al. Global transcriptional start site mapping using differential RNA sequencing reveals

novel antisense RNAs in *Escherichia coli*. J Bacteriol. 2015 Jan;197:18–28.
28. Cottier F, Tan ASM, Chen J, Lum J, Zolezzi F, Poidinger M, et al. The transcriptional stress response of *Candida albicans* to weak organic acids. G3 (Bethesda, Md). 2015 Jan;5:497–505.
29. Lai Z, Schluttenhofer CM, Bhide K, Shreve J, Thimmapuram J, Lee SY, et al. MED18 interaction with distinct transcription factors regulates multiple plant functions. Nat Commun. 2014;5:3064.
30. Hölzer M, Krähling V, Amman F, Barth E, Bernhart SH, Carmelo VAO, et al. Differential transcriptional responses to Ebola and Marburg virus infection in bat and human cells. Sci Rep. 2016;6:34589. Available from: http://dx.doi.org/10.1038/srep34589.
31. Flicek P, Amode MR, Barrell D, Beal K, Brent S, Carvalho-Silva D, et al. Ensembl 2012. Nucleic Acids Res. 2012 Jan;40:D84–D90.
32. Ensembl. Bacteria Collection.; 2018. Accessed 02 July 2018. ftp://ftp.ensemblgenomes.org/pub/bacteria/release-34/gtf/bacteria_0_collection/escherichia_coli_str_k_12_substr_mg1655/.
33. Ensembl. Plant Collection.; 2018. Accessed 02 July 2018. ftp://ftp.ensemblgenomes.org/pub/release-34/plants/fasta/arabidopsis_thaliana/dna/.
34. CGD. *Candida* Genome Database.; 2018. Accessed 02 July 2018. www.candidagenome.org.
35. Hölzer M. Electronic Supplement. *De novo* transcriptome assembly: A comprehensive cross-species comparison of short-read RNA-Seq assemblers.; 2018. Accessed 02 July 2018. http://www.rna.uni-jena.de/supplements/assembly.
36. Bushmanova E, Antipov D, Lapidus A, Suvorov V, Prjibelski AD. rnaQUAST: a quality assessment tool for *de novo* transcriptome assemblies. Bioinformatics. 2016;32:2210–2212.
37. Kim D, Langmead B, Salzberg SL. HISAT: a fast spliced aligner with low memory requirements. Nat Methods. 2015;12(4):357–360. Available from: http://dx.doi.org/10.1038/nmeth.3317.
38. Li B, Fillmore N, Bai Y, Collins M, Thomson JA, Stewart R, et al. Evaluation of *de novo* transcriptome assemblies from RNA-Seq data. Genome Biol. 2014 Dec;15:553.
39. Smith-Unna R, Boursnell C, Patro R, Hibberd JM, Kelly S. TransRate: reference-free quality assessment of *de novo* transcriptome assemblies. Genome Res. 2016 Aug;26:1134–1144.
40. Simão FA, Waterhouse RM, Ioannidis P, Kriventseva EV, Zdobnov EM. BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. Bioinformatics (Oxford, England). 2015 Oct;31:3210–3212.
41. Bankevich A, Nurk S, Antipov D, Gurevich AA, Dvorkin M, Kulikov AS, et al. SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. J Comput Biol. 2012;19(5):455–477. Available from: http://dx.doi.org/10.1089/cmb.2012.0021.
42. Zerbino DR, Birney E. Velvet: algorithms for *de novo* short read assembly using *de Bruijn* graphs. Genome Res. 2008 May;18:821–829.
43. Bankevich A. rnaSPAdes manual.; 2018. Accessed 02 July 2018. http://spades.bioinf.spbau.ru/release3.10.1/rnaspades_manual.html.
44. Chevreux B, Pfisterer T, Drescher B, Driesel AJ, Müller WE, Wetter T, et al. Using the miraEST assembler for reliable and automated mRNA transcript assembly and SNP detection in sequenced ESTs. Genome Res. 2004;14(6):1147–1159.
45. Simpson JT, Wong K, Jackman SD, Schein JE, Jones SJM, Birol I. ABySS: a parallel assembler for short read sequence data. Genome Res. 2009 Jun;19:1117–1123.
46. Feldmann H, Klenk HD, Sanchez A. Molecular biology and evolution of filoviruses. Arch Virol Suppl. 1993;7:81–100.
47. Rana SB, Zadlock FJ, Zhang Z, Murphy WR, Bentivegna CS. Comparison of *De Novo* Transcriptome Assemblers and k-mer Strategies Using the Killifish, *Fundulus heteroclitus*. PLoS One. 2016;11:e0153104.
48. Chopra R, Burow G, Farmer A, Mudge J, Simpson CE, Burow MD. Comparisons of *de novo* transcriptome assemblers in diploid and polyploid species using peanut (Arachis spp.) RNA-Seq data. PLoS One. 2014;9:e115055.
49. Moreton J, Dunham SP, Emes RD. A consensus approach to vertebrate *de novo* transcriptome assembly from RNA-seq data: assembly of the duck (*Anas platyrhynchos*) transcriptome. Front Genet. 2014;5:190.
50. Luo R, Liu B, Xie Y, Li Z, Huang W, Yuan J, et al. SOAPdenovo2: an empirically improved memory-efficient short-read *de novo* assembler. GigaScience. 2012 Dec;1:18.
51. Trapnell C, Williams BA, Pertea G, Mortazavi A, Kwan G, van Baren MJ, et al. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. Nat Biotechnol. 2010 May;28:511–515.
52. Li B. Vignette for DETONATE.; 2018. Accessed 02 July 2018. http://deweylab.biostat.wisc.edu/detonate/vignette.html.