Demultiplexing and barcode-specific adaptive sampling for nanopore direct RNA sequencing

Wiep van der Toorn [1†], Patrick Bohn [2†], Wang Liu-Wei [1,3], Marco Olguin-Nava [2], Redmond P Smyth [2,4*], and Max von Kleist [1,5*]

*To whom correspondence should be addressed. redmond.smyth@helmholtz-hiri.de, max.kleist@fu-berlin.de
1 Systems Medicine of Infectious Disease (P5), Robert Koch Institute, Berlin, Germany
2 Helmholtz Institute for RNA-based Infection Research, Helmholtz Centre for Infection Research, Würzburg, Germany
3 International Max-Planck Research School for Biology and Computing (IMPRS-BAC), Berlin, Germany
4 Department of Medicine, University of Würzburg, Würzburg, Germany
5 Department of Mathematics and Computer Science, Freie Universität Berlin, Germany
† These authors contributed equally to this work.

Abstract

Nanopore direct RNA sequencing (dRNA-seq) enables unique insights into (epi-)transcriptomics. However, applications are currently limited by the lack of accurate and cost-effective sample multiplexing. We introduce WarpDemuX, an ultra-fast and highly accurate adapter-barcoding and demultiplexing approach. WarpDemuX enhances speed and accuracy by fast processing of the raw nanopore signal, use of a lightweight machine-learning algorithm and design of optimized barcode sets. We demonstrate its utility by performing a rapid phenotypic profiling of different SARS-CoV-2 viruses, crucial for pandemic prevention and response, through multiplexed sequencing of longitudinal samples on a single flowcell. This identifies systematic differences in transcript abundance and poly(A) tail lengths during infection. Additionally, integrating WarpDemuX into sequencing control software enables real-time enrichment of target molecules through barcode-specific adaptive sampling, which we demonstrate by enriching low abundance viral RNA. In summary, WarpDemuX is a broadly applicable, high-performance, and economical multiplexing solution for nanopore dRNA-seq, facilitating advanced (epi-)transcriptomic research.

Introduction

Nanopore direct RNA sequencing (dRNA-seq) is a powerful tool for transcriptomics [1]. Unlike other RNA sequencing methods, dRNA-seq does not require the conversion of RNA to cDNA. It therefore allows for the direct identification of RNA isoforms [2] and associated transcript features such as polyA tail length [3]. This not only provides a more comprehensive view of the transcriptome, it also preserves native RNA base modifications, offering a unique opportunity to study the dynamic epitranscriptome [4].

A common sequencing practice, known as multiplexing, allows different samples to be pooled and sequenced simultaneously. Multiplexing reduces costs while minimizing experimental variability, and is essential for modern sequencing-based assays, such as single cell sequencing [5]. Multiplexing involves the attachment of sample specific 'barcodes' to each sequencing library before pooling. This allows to re-identify sample-specific sequencing data for subsequent analysis. While multiplexing protocols for nanopore DNA-sequencing (DNA-seq) are available, no commercial multiplexing protocols have been released for dRNA-seq.

Multiplexing in nanopore sequencing is challenging due to its unique methodology, where individual molecules are threaded through a protein (nano-)pore that is embedded in an electrically-resistant polymer membrane. During pore translocation, features of the molecule cause systematic disruptions to an ionic current flow. Controlled movement is achieved using a motor protein associated with a DNA sequencing adapter that is ligated to the end of the molecule during library preparation. This leads to step-wise changes in electrical signal that are characteristic of a group of nucleotides (k-mer) passing through the pore at a given time [6]. Repeated measurements are made at each translocation 'event', and the number of repeated measurements per event is commonly termed the 'dwell time'. Intrinsic fluctuation of dwell time is a characteristic of nanopore sequencing, and the resulting temporal variation can pose challenges in accurately converting the signal into a nucleotide sequence ('basecalling').

For DNA-seq, commercial protocols incorporate the specific nucleotide barcode within the DNA sequencing adapter, which is later read out natively by the DNA basecaller (basecalling-based demultiplexing). dRNA-seq uses similar DNA sequencing adapters to attach the motor protein during library preparation, resulting in a hybrid (DNA-RNA) molecule. Because the DNA portion of molecule cannot be natively

basecalled by the RNA-specific basecaller, it is not possible to read out the specific nucleotide sequences of barcodes. Conceptually, this problem can be overcome by ligating RNA barcodes to the 3' end of the transcript. However, RNA barcodes require additional ligation and purification steps, which increases the complexity and duration of the library preparation process, and potentially introduces ligation biases. Moreover, the presence of the DNA sequencing adapter adjacent to the RNA barcode negatively affects the barcode basecall quality, complicating demultiplexing. Addressing this with linkers is undesirable due to high costs: RNA oligos are approximately 10 fold more expensive than DNA oligos (at €6 per nucleotide, ordering 12 16-mer barcodes amounts to roughly €1200)

Alternatively, barcode sequences can be incorporated into the DNA adapter and demultiplexed from their raw electrical signals (signal-based demultiplexing). The DNA adapter (RTA) contains an 18-nucleotide stretch that can be customized to include a barcode, and its ligation to the 3' end of the RNA is integral to dRNA-seq library preparation, thus barcoding would not complicate the experimental protocol. Signal-based demultiplexing was first realized in a method named DeePlexiCon (DPC) [7]. Its main algorithm transforms the raw adapter signal to images, which are subsequently classified using a deep convolutional neural network. Unfortunately, this approach requires extensive computational resources, such as GPUs, to achieve reasonable run times. Despite only using four barcodes, intrinsic variation in the dwell time produce inconsistent signal-transformed images leading to substantial misclassification.

Here, we developed WarpDemuX, an efficient and scalable method to design and detect RTA barcodes for nanopore dRNA-seq. WarpDemuX operates directly on the raw signal and does not require basecalling. It uses novel signal preprocessing and a fast machine learning algorithm for barcode classification. We obtain an optimized set of 12 RTA barcodes with maximized inter-barcode distances in the signal space for high accuracy demulitplexing. We also establish real-time dRNA-seq demultiplexing and barcode-specific adaptive sampling [8], previously not available for dRNA-seq.

Results

High demultiplexing accuracy, yield and speed using WarpDemuX

We aimed to establish a robust, low cost, and accurate barcoding methodology to enable dRNA-seq multiplexing and adaptive sampling experiments. We chose to place the barcode sequences within the customizable RTA adapter due to its compatibility with the dRNA-seq library preparation workflow. Because the DNA RTA adapter cannot be read by the RNA-specific basecaller, we implemented signal-based demultiplexing.

Dynamic Time Warping (DTW) is an algorithm used for comparing time-series with temporal distortions. This makes it ideally suited for analysing raw nanopore signals with their intrinsic variations in dwell-time [9, 10, 11, 12]. To apply this algorithm to dRNA-seq multiplexing, we integrated a Dynamic Time Warping Distance (DTWD) kernel into a Support Vector Machine (SVM) classifier. This DTWD-based kernel function captures the essential spatial and temporal signal information by quantifying how similar an unknown barcode is to known barcode patterns, termed 'fingerprints'.

For training the SVM-DTWD model, fingerprints were obtained by sequencing 'dual barcoded' molecules. These were generated by ligation of known RTA barcodes to distinct *in vitro* transcribed RNA molecules containing an additional in-line RNA barcode (see *Methods* for more details) (**Fig. 1a**). Unique combinations of RNA transcripts, in-line RNA barcodes and RTA barcodes were used to establish a ground truth label for the RTA barcode based on the basecalled sequence of the RNA transcript and RNA barcode (**Table S1-S3**). To extract the fingerprints, we first identified the adapter signal by locating the DNA/RNA boundary in the raw signal (**Fig. 1b**). We then segmented the adapter signal into discrete sequencing events (**Fig. 1c**). Finally, we extracted the last 25 segments, expected to contain the barcode signal (**Fig. 1a,c**).

In our barcode classification approach, a confidence score is computed to assess the reliability of each classification result (see *Methods*). By setting a confidence threshold and discarding predictions below it as 'unclassified', users can balance accuracy and yield according to their needs. It is important to note that the fraction of unclassified reads is influenced by the dataset's characteristics, not just the model's performance. Specifically, datasets with more noisy reads tend to have a reduced yield because these reads often fall below the confidence threshold, which is the desired behavior to ensure accuracy. To allow users to balance accuracy and yield independently of noise detection, we introduced a 'noise class' into the model. This integration enhances WarpDemuX's robustness and allows for precise tuning in diverse dataset conditions. The training data for this class includes fingerprints from outlier reads affected by signal irregularities, such as stalled adapters, blocked pores, or inaccurate DNA/RNA boundaries (see **SN.1**). Reads classified as noise are subsequently omitted from performance assessment.

For benchmarking, we initially obtained fingerprints for four RTA barcode sequences used in DPC (**Table S4**) [7]. At a confidence threshold of 0.5, WarpDemuX outperformed DPC in both demultiplexing accuracy (0.97% vs. 0.92%) and yield (unclassified 7.7% vs. 11.4%) (**Table 1**). This superior performance was achieved using just 1% of the training data per barcode (400 instances vs. 40,000), and WarpDemuX's enhanced classification performance over DPC held true for all confidence thresholds tested (see **SN.2**). Furthermore, when aiming for maximum yield by using a confidence threshold of 0, WarpDemuX still out-

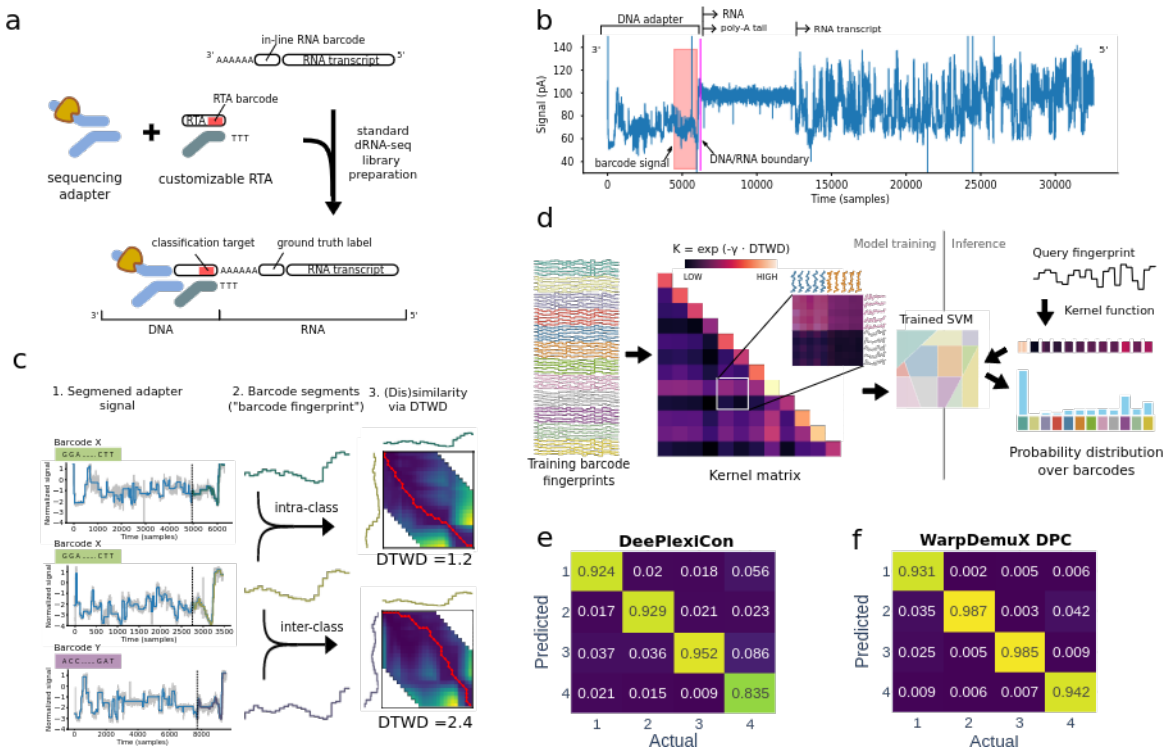

Figure 1: WarpDemuX model. (a) Generation of training data using 'dual barcoded' RNA molecules containing an inline RNA barcode positioned at the 3' end of the transcript, serving as a ground truth label, and an RTA barcode, introduced during standard dRNA-seq library preparation, which served as the target for classification. (b) The direct RNA sequencing raw signal denotes a time series of electric current measurement. The signal (blue) consists of three main components: the DNA RTA adapter signal, the RNA poly(A) tail and the RNA body. The last part of the adapter signal contains the RTA barcode to be classified (red). (c) The adapter signal is segmented into events, with each segment represented by its average signal. The final 25 segments of the adapter are designated as the RTA barcode specific "fingerprint". Dynamic Time Warping Distance (DTWD) is used to quantify the similarity between these fingerprints, with higher similarity (lower DTWD) observed within the same barcode class compared to different barcode classes. (d) Model training and classification of barcode fingerprints into barcode classes using the DTWD kernel function. Training of the Support Vector Machine (SVM) classifier with the DTWD kernel function on barcode fingerprints from sequencing reads captures the nuances of the barcode signals and promotes robust, noise-resilient classification. (e-f) Performance of models. Values reported reflect the recall (confusion matrix, normalized over the 'actual class'; true label), on both test sets combined, evaluated at a confidence cutoff of 0.5. (e) DeePlexiCon (DPC) CNN model. (f) WarpDemuX SVM-DTWD.

performed DPC (0.94 against 0.87) (see **SN.2**). Notably, in this setting, all data was assigned to barcodes rather than to the noise class. Altogether, these data argue that SVM-DTWD classification effectively captures the nuances of the barcode fingerprints and is robust to noise emanating from dwell-time variation. Importantly, this increase in accuracy was achieved with 10-times less compute time compared to DPC (14.4ms vs. 163ms per read on a standard laptop) (**Table 1**). In the described set-up, WarpDemuX-DPC classifies 4000 reads in less than 1 minute, with a peak memory usage (maximum resident set size) of less than 600 MB.

Optimized barcodes improve WarpDemuX performance

To expand beyond the pool of 4 barcodes, we next implemented a computational framework to obtain optimized sets of RTA barcodes with distinct signal profiles. This *de novo* design utilizes DTWD as a metric to maximize inter-barcode differences in signal space. In brief, we constructed a set of wave-like target signal patterns (**Fig. 2a**) that exhibited high inter-target DTWD. We then used the available DNA k-mer models (`github.com/nanoporetech/kmer_models`) to search for barcode sequences whose putative signals would closely match these target patterns (see **SM.1**). From this reduced number of candidate barcodes, we selected the set of k (here, $k = 12$) candidate barcodes that maximized all pairwise DTWDs (**Table S5**). Notably, this strategy avoided an exhaustive search across all possible sets of barcodes and thereby significantly reduced the computational costs of barcode set selection (**Fig. 2b**) (see *Methods* for technical details). As before, we generated ground-truth training data for the optimized WDX-RTA barcodes using distinct dual-barcoded RNA transcripts (**Table S1**).

We next assessed the performance of this design strategy by comparing non-optimized RTA barcodes (WDX-DPC) to four of the 12 optimized barcodes (WDX4). The use of optimized barcodes improved both

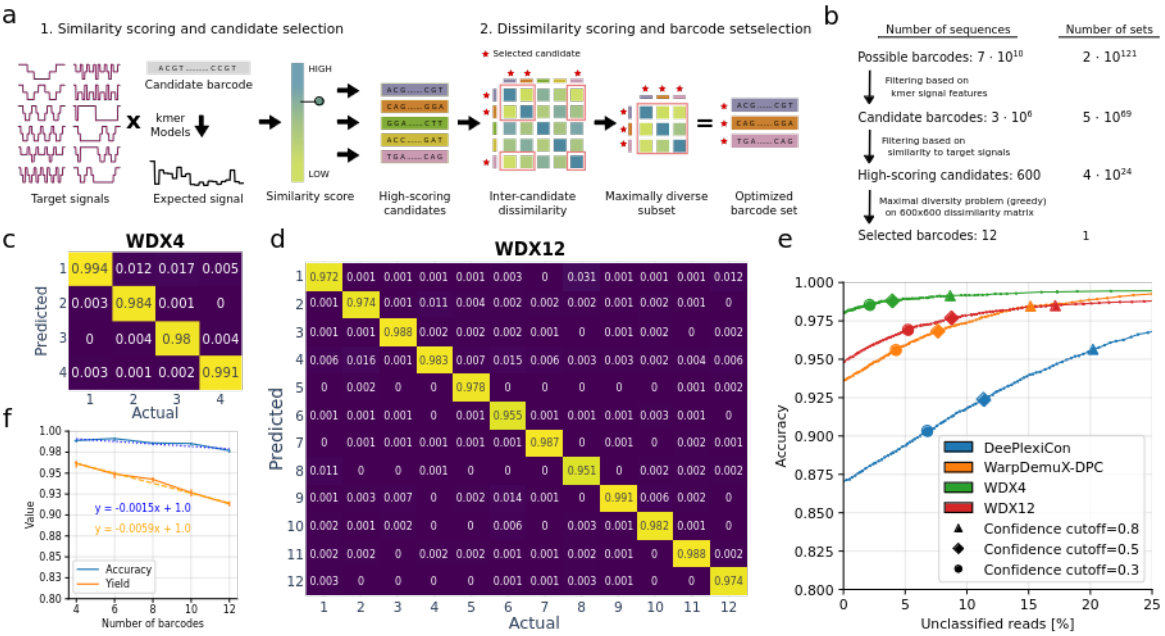

Figure 2: Optimized barcodes improve WarpDemuX performance. (a-b) *In silico* barcode design. (a) Distinct wave-like signal patterns enhance distinction between the individual target patterns. First, candidate barcodes are scored based on their similarity to a target pattern. Second, high scoring candidates are selected and an optimized set of barcodes is identified based on their mutual Dynamic Time Warping Distance (DTWD). (b) Identifying barcode sets using kmer models. Identification of 12 barcodes with maximal pairwise DTWDs. (c-d) The use of optimized barcodes improves accuracy and yield. Values reported reflect the recall (confusion matrix, normalized over the 'actual class'; true label), on both test sets combined, evaluated at a confidence cutoff of 0.5. (c) 4 optimized barcodes (WDX4) (d) 12 optimized barcodes (WDX12). (e) Relationship between accuracy and percentage of unclassified reads, modulated by the confidence cutoff. Values reflect the performance per model as the weighted averaged over their respective test sets. (f) Performance trends for accuracy and yield with increasing number of barcodes, evaluated at a confidence cutoff of 0.5. Linear regression analysis of the 10-fold cross-validated performance (weighted averaged over test sets) suggests scalability for a larger set of optimized barcodes. Measured performance is shown by solid lines, and linear fits are represented by dashed lines, with annotated formulas. R^2 values are 0.75 (accuracy; blue) and 0.97 (yield; orange).

accuracy (0.99 WDX4 vs. 0.97 WDX-DPC) and yield (96% WDX4 vs. 92.3% WarpDemuX-DPC) while run times remained unaltered (**Table 1 and Fig. 2c**). This confirmed that maximizing dynamic time-warping distance (DTWD) improves classification performance. When the model was extended to 12 distinct barcode classes (WDX12) accuracy only slightly decreased despite the complexity of distinguishing a larger number of barcode classes (0.99 WDX4 vs 0.98 WDX12; weighted average) (**Fig. 2d**). A decrease in prediction confidence resulted in a slightly higher rate of unclassified barcodes (4.0% to 8.8% at a confidence threshold of 0.5) (**Table 1**). Remarkably, the WDX12 model consistently outperformed DPC (4 barcodes) across all confidence thresholds in accuracy, yield and run time (**Fig. 2e** and **Table 1**), whilst expanding the number of barcode classes three-fold. Importantly, the performance of WarpDemuX only decreased slowly with increasing numbers of barcodes (**Fig. 2f**). With an optimized set of barcodes, we predict that 24 barcodes can be implemented at an accuracy of 96% with 14% of reads unclassified (confidence cutoff = 0.5) (**Fig. 2f**).

Model	Dataset	Reads per barcode (avg)	Reads classified as noise (%)	Confidence cutoff	Unclassified reads (%)	Accuracy	Precision	Recall	Run Time (ms per read)
DeePlexiCon	1	488	0.0	0.5	11.0	0.925	0.919	0.907	163.0
	2	2114	0.0	0.5	15.9	0.906	0.859	0.918	
WarpDemuX-DPC	1	488	0.0	0.5	9.5	0.965	0.957	0.967	14.4
	2	2114	0.0	0.5	7.3	0.969	0.964	0.958	
WDX4	3-5 (test set)	600	1.8	0.5	2.7	0.989	0.989	0.989	14.4
	6	1256	4.7	0.5	4.6	0.987	0.983	0.984	
WDX12	3-5 (test set)	600	2.2	0.5	8.8	0.977	0.976	0.977	15.6
	6	1256	4.8	0.5	10.1	0.975	0.973	0.975	

Table 1: Performance comparison between DeePlexiCon (DPC) and WarpDemuX (WDX) across datasets and barcoding strategies. Datasets 1 and 2 denote different RNA templates, respectively barcoded with DPC's four RTA barcodes, whereas datasets 3-6 refer to distinct RNA templates barcoded with optimized WDX RTA barcodes (WDX4: 4 barcodes used; WDX12: 12 barcodes used). Details on the data sets are provided in **Table S1**. Precision and recall reflect averages over barcode classes. Reported run times (CPU-only) refer to 10-fold cross-validated average computing time from 4000 reads on 8 CPU cores (11th Gen Intel(R) Core(TM) i7-1165G7 (2.80GHz) with 32GB memory).

Rapid phenotyping of SARS-CoV-2 viruses on a single flow cell

Rapid phenotypic profiling of viruses has critical importance in pandemic prevention and response, for which nanopore sequencing offers unique benefits [13]. Multiplexing samples on a single flow cell increases the speed and efficiency of profiling, while also reducing experimental variation. We demonstrate the benefits of WarpDemuX multiplexing by rapidly phenotyping two genetically distinct SARS-CoV-2 viruses, namely SARS-CoV-2 wild-type (Wuhan-Hu-1) and a variant expressing GFP in ORF8a. We extracted RNA from cells at 8, 24, 48 and 72 hours post-infection (hpi) as well as from viral supernatant at 72 hpi (**Fig. 3a** and **Fig. SN.3.1**). Cells of two uninfected mock samples served as negative controls. Samples were barcoded with WDX12 barcodes and sequenced on a single Minion R9.4.1 flow cell (**Fig. 3a**) (also see **SN.3**). Of the 363,283 reads passing a quality score (qscore) threshold of 10, we identified barcodes in 95.4% (346,691) of reads (**Fig. SN.3.2a-b**). Upon demultiplexing, the acquired reads exhibited a more than 50-fold difference between supernatant samples and some cellular samples. We then leveraged the heavily unbalanced nature of this multiplexed experiment to explore the effects of confidence threshold settings. Specifically, we determined the best confidence threshold by analysing the species alignments for each sample at different confidence score cutoffs (**Fig. SN.3.4**). This analysis revealed that a threshold of 0.99 minimized spillover into the two heavily underrepresented samples while preserving 69% of passing reads (236,952 reads). At this setting, over 90% of reads from supernatant samples were of SARS-CoV-2 origin (**Fig. SN.3.5c**. Moreover, EGFP sequences were limited to the SARS2-EGFP-infected samples, while those with WT ORF 8a were almost exclusively present in SARS2-WT-infected samples. Together, these data indicate excellent classification performance, and minimal crossover, even on unbalanced samples representing <1% of the total reads.

We next investigated the read length and read identity distributions of the demultiplexed samples **Fig. 3b-f**). We observed clear differences in read length distributions as the infection progressed (**Fig. 3b**). Discrete bands became visible at 24-72 hpi (**Fig. 3b** and **Fig. SN.3.6**), which could be attributed to distinct subgenomic RNAs by TRS-leader sequence annotation (**Fig. SN.3.7b-c**). This analysis showed striking differences in the relative abundances of individual subgenomic RNAs (sgRNAs), with the most abundant sgRNA N present at 22.3% (95% CI = 21.9, 22.7; 48 hpi), compared to the least present subgenomic RNA 10, which was present at 2.5×10^{-1}% (95% CI = 2.0×10^{-1}, 3.0×10^{-1}; 48 hpi) (**Fig. 3d**). The relative composition of sgRNAs, however, remained relatively stable throughout the infection cycle (**Fig. 3d**). Importantly, the ability to pool RNA from infected cells on a single flowcell allowed us to assess the replication kinetics of these viruses, without running multiple sequencing experiments. For both viruses, SARS-CoV-2 RNA was first detectable at 24 hpi, and reached a maximum at 48 hpi, comprising approximately 50% of all poly-adenylated RNA in the RNA extract, before declining to 30% at 72 hpi. No apparent difference in kinetics between the wildtype and EGFP-expressing virus was observed indicating insertion of EGFP does not effect viral replication over the time-frame tested (**Fig. 3c**).

Finally, we performed a poly(A) tail length analysis, as this is known to regulate RNA translation and stability, and is regulated during infection in other Coronaviruses (**Fig. 3g-i**). The polyA tail length of SARS-CoV-2 sgRNA was previously characterized for a single time point of 24 hpi [14], revealing two tail length populations with a low abundance of shorter tails. Using WarpDemuX, we were able to study the dynamics of poly(A) tail length across multiple time points and compare RNA from cells and virions. This revealed a broad tail length distribution for Vero mRNA, which was stable over time, with a median of 70 nt (2.5th-97.5th percentile: 14-227 nt; 48 hpi) , while SARS-CoV-2 RNA had significantly shorter poly(A) tail lengths (median 56 nt, 2.5th-97.5th percentile: 23-108 nt; 48 hpi; two-sided, independent T-test, $p < 0.001$), and exhibited regulation of tail length over the course of infection. Intriguingly, at 24 hpi, >99% of cellular SARS2 RNA was attributed to a population with a median tail length of 56 nt, with an additional population with a shorter poly(A) tail length around 29 nt becoming more abundant over time, increasing to 3% at 48 hpi and 15% at 72 hpi (**Fig. 3g** and **Fig. SN.3.10c**). These shorter tails in the cell appear to be limited to sgRNAs (**Fig. 3h**) and were not observed in virion RNA (**Fig. 3i**). The increased presence of shorter tail lengths at the later time points could reflect mRNA degradation upon cell death, compared to genomic RNA protected within membrane vesicles. In agreement, when compared to cellular RNA at 72 hpi, supernatant RNA within virions had a longer average tail length (52 nt cellular vs. 58 nt supernatant) (**Fig. 3i**).

Barcode balancing and enrichment of low abundance samples by adaptive sampling

Adaptive sampling, a unique feature of nanopore sequencing, allows real-time, programmable sequencing decisions (**Fig. 4a**) [12, 15]. This method operates by briefly inverting the voltage across the pore to reject unwanted molecules. One of its main applications is "barcode balancing", where rejection aims to unify read counts or total yields between barcodes during sequencing (**Fig. 4a**) [8]. Most commonly, adaptive sampling decisions are made on basecalled nucleotide sequences [16, 8, 15, 17, 18, 19]. However, basecalling is slow, allowing unwanted molecules to traverse a substantial distance through the pore before rejection, which negatively impacts pore lifetime and sequencing yield. Making adaptive sampling decisions on raw signal can be significantly faster [20], preserving pore integrity and resulting in better overall enrichment and/or yield of low abundance samples. Yet, this barcode balancing approach has not yet been implemented for

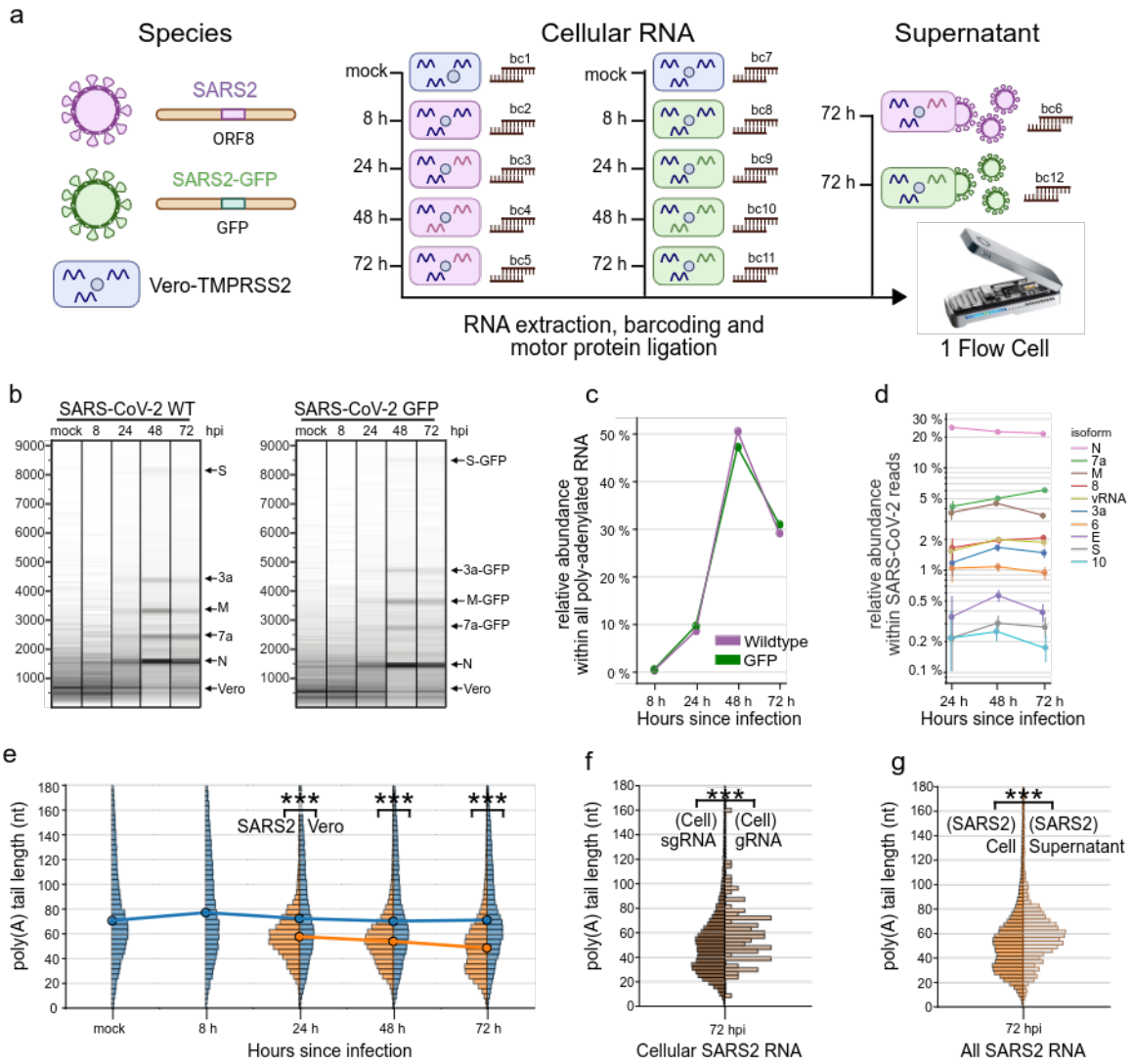

Figure 3: Analysis of SARS-CoV-2 infection kinetics with WarpDemuX. (a) Experimental workflow. Cellular RNA was extracted from uninfected Vero-TMPRSS2 cells, and Vero cells infected with rSARS-CoV-2 or rSARS-CoV-2-TurboGFP at different timepoints after infection, as well as from cleared supernatant of infected cells. The extracted RNA was barcoded with distinct WDX12 barcodes and sequenced. (b) Virtual gels (read length scaled on log scale, weighted by number of bases) of the WDX demultiplexed cellular samples. Distinct bands are labeled according to transcripts matching the size. (c) Relative abundance of SARS-CoV-2 RNA within all reads across timepoints for the WT and EGFP variant respectively. (d) Relative abundance of full-length SARS-CoV-2 sgRNA within all SARS-CoV-2 RNA (pooled analysis of wildtype and EGFP variants). (e) Poly(A) tail distribution of cellular RNA per species. SARS-CoV-2 represents pooled samples from WT and GFP variants (*** = $p < 0.001$; independent T-test with $n_{\text{Vero}} = 46456, 40751, 59569$ and $n_{\text{SARS2}} = 4638, 40276, 25359$ for 24, 48 and 72 hpi, respectively). (f) Poly(A) tail distribution of cellular SARS-CoV-2 sgRNA and genomic RNA (gRNA), gRNA defined as having >8.5 kb read length (*** = $p < 0.001$; independent T-test, with $n_{\text{sgRNA}} = 9614$ and $n_{\text{gRNA}} = 87$). (g) Poly(A) tail distribution of cellular vs. supernatant SARS-CoV-2 RNA (*** = $p < 0.001$; independent T-test with $n_{\text{Cell}} = 25359$ and $n_{\text{Supernatant}} = 909$).

dRNA-seq.

Leveraging WarpDemuX's classification speed, we implemented ultra-rapid adaptive sampling for dRNA-seq, aiming to make rejection decisions while the poly(A) tail of RNA molecules is still within the pore. We modified MinKNOW software to provide signal data earlier to the WarpDemuX pipeline, which required adaptations to internal read classification logic (**SN.6**). We also developed our own decision logic to begin rejecting reads once specific thresholds are reached. For this, we use dynamic thresholds that update based on the current sequencing distribution and guide the rejection of reads based on real-time metrics such as adapter counts, read counts, and number of bases (**SN.4**). We applied these decisions to 40 channels each on a MinION R9.4.1 flow cell (**Fig. 4b**). Notably, 81% of classifications finished within 60 ms after detection of a poly(A) tail, and 98% of classifications completed within 200 ms (**Fig. 4c**). Assuming a translocation

speed of 70 nt/s, classification occurs well within 14 nt of the start of the poly(A) tail. In practice, due to the latency of the different softwares, 50% rejections are made within 14 nt, and 95% within 34 nt, which would be within the poly(A) tail for the majority of reads (**Fig. 4d**).

To provide a rigorous test of the adaptive sampling pipeline, we turned to our SARS-CoV-2 samples, which were heavily unbalanced due to differing amounts of RNA present in cells and supernatant (**Fig. 4b**). All barcode balancing strategies led to a greater number of molecules being sampled, due to higher pore turnover (**Fig. 4e**). Importantly, the rejection of high abundance barcodes not only led to a more even barcode coverage, reducing the Gini coefficient from 0.44 to 0.15, but also to the enrichment of low abundance barcodes by 33-73% (**Fig. 4f**).

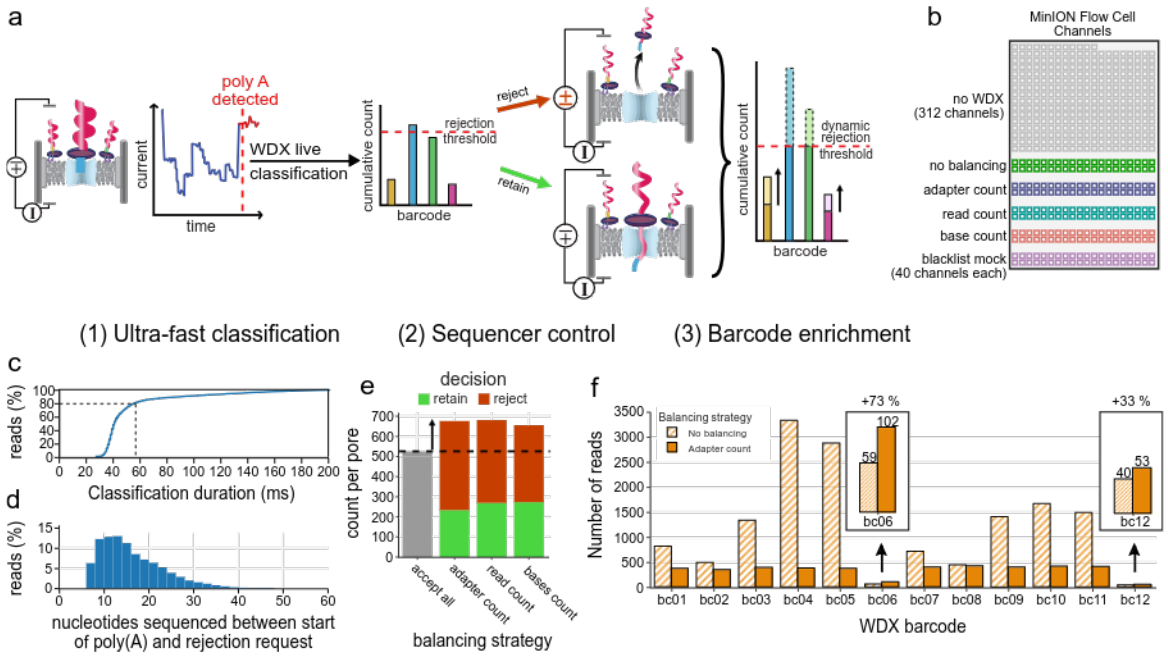

Figure 4: Live barcode balancing using WarpDemuX. (a) Principle of live barcode balancing. Raw signal data is accumulated and once the start of a poly(A) tail is detected, then submitted to the WarpDemuX live classifier. Based on its classification a decision is taken to either retain or reject the barcode, resulting in increased read turnover when rejecting. (b) Channel assignments for live barcode balancing during the experiment. (c) Cumulative distribution plot of duration of WarpDemuX barcode classification itself. (d) Histogram for number of nucleotides that passed through the pore from start of poly(A) until rejection decision. (e) Bar plot of number of adapter counts per pore per balancer, colored by decision taken after classification of adapter. (f) Bar plot showing total number of reads recovered with and without balancing on adapter counts. Inset shows enrichment for low abundance samples.

Discussion

Despite its transformative potential for RNA biology research, dRNA-seq has been limited in its applications and wider adoption by the lack of effective multiplexing techniques. To address this, we developed WarpDemuX, a novel barcode-design and demultiplexing method which exhibits high classification speed, accuracy, yield, and scalability. By leveraging these characteristics, we used WarpDemux to implement adaptive sampling for dRNA-seq, which enables for the first time barcode balancing and the enrichment of low abundance samples.

WarpDemuX introduces two key innovations: (i) a signal pre-processing step and (ii) a lightweight SVM-DTWD classifier. Signal pre-processing reduces data noise and mitigates spurious temporal correlations from varying dwell times (**SN.5**). The SVM-DTWD classifier requires minimal training data, and is computationally inexpensive. In comparison, HycDemux, a recent DTWD-aided demultiplexing method for nanopore-based DNA sequencing, also employs DTW of raw signals, but cannot be employed in real-time settings due to its cluster-based demultiplexing procedure [21]. DPC, the only tool for dRNA-multiplexing, converts raw signals into images for classification with a deep convolutional neural network [22] and was less accurate as well as 10 times slower than the CPU-based WarpDemuX. Ultra-fast classification allow us to implement barcode balancing for the uniform sequencing of multiplexed, uneven samples. Since we were able to make read decisions <200ms after the start of the poly(A) tail, virtually all poly-adenylated RNA were rejected before sequencing the main body of the RNA. Our fast decision time may have prevented pore blockage, because early rejection preempted the formation of RNA structures on the trans side of the pore,

which may otherwise hinder the ejection of the molecule back through the pore. In addition to the enrichment of low abundance barcodes through higher pore turnover, dynamic barcode balancing could also be used to extend flow cell lifespan and total yield by dynamically rejecting reads from barcodes that generate high numbers of blocked pores (SN.4 and SN.6). Implementations for demultiplexing and adaptive sampling are available at `github.com/KleistLab/WarpDemuX`.

WarpDemuX showed high classification accuracy, even on barcodes that were not explicitly optimized for DTWD-based separability. Further improvements in performance were achieved by designing 12 barcodes that were maximally separated in signal space, which can likely scale to at least 24 barcodes whilst maintaining a classification accuracy above 95%. Our design strategy is not inherently limited by the number of barcodes, however, increasing the total number of barcodes impacts classification confidence and yield. We therefore acknowledge the possibility of reaching a saturation point in creating distinct signal patterns, which would set an upper limit on the number of employable barcodes. Systematic efforts to test the limits of barcode design and classification could include the development of signal (k-mer) models for DNA sequenced in the 3' to 5' direction, as this would allow more accurate *in silico* design of distinct barcodes. Furthermore, as the number of DTWD calculations for classification scales with the number of barcode classes, the use of learned DTW kernels rather than full signal-to-signal comparisons could be explored to further speed up the method. Given that adapter extraction and segmentation currently account for 90% of the run time (see **SN.2**), computational advancements in these areas would accelerate barcode assignment, which would be useful for real-time analysis as sequencing yields continue to improve [23].

An important feature of our approach is that it operates independently of basecalling, and is thus not influenced by continued development of new basecallers. However, changes in sequencing chemistry will necessitate retraining of the SVM-DTWD kernel. On this note, ONT recently unveiled the new direct RNA sequencing kit SQK-RNA004, the successor to SQK-RNA002. This newer chemistry contains an RNA-sequencing-specific pore with faster motor protein. The library preparation approach remains consistent between the two versions, so that DNA barcodes proposed herein will seamlessly integrate with the SQK-RNA004 kit. However, the new RNA-specific pore displays different signal characteristics, suggesting potential benefits of developing RNA004-specific WarpDemuX barcodes, which is the subject of ongoing work.

We demonstrated the first application of dRNA-seq barcode-specific adaptive sampling and barcode balancing on SARS-CoV-2 infected cells. This enabled us to measure the replication kinetics of two different SARS-CoV-2 viruses by direct read out of viral RNA in a single rapid experiment. Standard replication assays using plaque assay take several additional days after sample collection [24, 25], meaning that WarpDemux can aid in rapidly assessing the replication capacity of novel SARS-CoV-2 variants as they emerge [26, 27]. We were also able to measure the relative abundance RNA isoforms and poly(A) tail length of SARS-CoV-2 samples taken at different time-points. Altered sub-genomic RNA expression patterns have been associated with SARS-CoV-2 lineages [26, 27], infection states [28], and can be regulated by host proteins [29]. WarpDemuX would allow these phenomenon to be investigated without sequencing batch effects. Similarly, changes in poly(A) tail length during infection have been previously observed for other coronaviruses ([30, 31], and could point to regulation of poly(A) tail length to affect viral RNA translations or genome packaging.

In summary, we envision that ultra-fast and high-accuracy dRNA-seq multiplexing will allow for more cost- and resource-effective profiling of RNA, especially when combined with adaptive sampling. As dRNA-seq continues to evolve, tools like WarpDemuX will play a pivotal role in maximizing the benefits of this transformative technology, catalyzing advancements in transcriptomics and epitranscriptomics research.

Methods

Adapter Signal Processing and Alignment

To account for the intrinsic variation in the dwell time, as well as the within-event measurement noise, we partition the raw signal of the adapter into n segments, based on the method introduced in ONT's signal analysis tool Tombo (`github.com/nanoporetech/tombo`; algorithmic details are included in **SM.2**). Each segment is subsequently represented by its average signal. To mitigate the risk of data loss due to missed events, we intentionally over-segment the signal by detecting approximately 10% more segments than expected based on the number of bases in the RTA. We normalize the segmented adapter signal and retain only the last 25 segments, which should contain the barcode signal given its known sequence length and relative location within the RTA.

To correct for the likely imperfect segmentation, we perform subsequent Dynamic Time Warping (DTW) to compute the distance between segmented barcode signals. DTW is a robust solution to align two sequences with temporal variation by compressing or stretching their temporal axis to minimize the cumulative Euclidean distance between corresponding elements (thus correcting for imperfect segmentation).

To compute the Dynamic Time Warping Distance (DTWD) between two segmented signals s_1 and s_2, $DDTW(s_1, s_2)$ we use the `dtaidistance` package [32]. The parameters used are detailed in **SM.3**.

Barcode Design via Dynamic Time Warping Distances

We developed a method to generate sets of k RTA barcodes with distinct signal profiles *de novo*, as it would be computational intractable to perform an exhaustive search over all possible barcode sets of size k (complexity $\binom{4^L}{k}$ for barcodes of length L, e.g. $\approx 2.3 \times 10^{121}$ for $L = 18$ and $k = 12$). First, we constructed a set of wave-like target signal patterns that exhibited high inter-target DTWD, since we expect such patterns to improve signal segmentation and barcode separability. Then, we searched for barcode sequences whose putative signal, constructed using the ONT k-mer models (`github.com/nanoporetech/kmer_models`), would closely match these target patterns. Because there are no DNA k-mer models for DNA sequenced in 3' to 5' direction, we filter for k-mers we expect have similar signal profiles in both sequencing directions, as detailed in **SM.1**. For the twelve target patterns highlighted in **Fig. 2a**, we selected the top 50 barcodes per target pattern, yielding a total of 600 barcodes, for which we computed all 600×600 pairwise DTWDs. Lastly, within the matrix of 600^2 pairwise DTWDs we solved the Maximal Diversity Problem (MDP, below) with k=12 to propose 12 candidate barcodes that we expect to be accurately identifiable and distinguishable. The MDP is defined as:

$$\begin{aligned} \max_{\delta(\cdot)} & \sum_{i=1}^{M-1} \sum_{j=i+1}^{M} DDTW(s_i, s_j)\delta(i)\delta(j) \\ \text{Subject to } & \sum_{i=1}^{M} \delta(i) = k \\ & \delta(i) = \{0, 1\}, \quad 1 \leq i \leq M \end{aligned} \tag{1}$$

where M denotes the entire set of barcode candidates (600 in our example), k is the number of distinct barcodes we aim to select, s_i is the segmented signal of barcode i, $DDTW(s_i, s_j)$ is the DTWD between s_i and s_j, and $\delta(i)$ is a dirac delta function indicating whether barcode i is selected. Detailed descriptions of the steps in the design process are included in **SM.1**.

Design and Synthesis of Dual-Barcoded RNA for Classifier Training

To evaluate our barcode design and train a classifier for demultiplexing, we generated labeled RTA signals by synthesizing and sequencing various custom RTA adapters attached to RNA molecules with integrated (in-line) RNA barcodes—serving as ground truth label. Notably, the in-line RNA barcodes used were integrated into the RNA transcripts using PCR, positioning them at the 5' end of the polyA tail to enhance the accuracy of basecalling and, consequently, the precision of barcode classification, compared to standard RNA barcodes that are ligated to the 3' end of the polyA tail. We randomized the RNA in-line barcode to RTA barcode assignment across replicates to circumvent any potential of bias introduced by the in-line RNA barcode signal. The generated data sets (1–6) are detailed in **Table S1**.

RNA synthesis

The barcoded RNA designated for training the WarpDemuX classifier was synthesized via *in vitro* transcription. To enrich the diversity of our dataset, we used a variety of RNA templates.

Briefly, starting from different DNA templates, 12 PCRs per template were performed with a forward primer containing a T7 RNA polymerase promoter and varying reverse primers to add different RNA barcodes to the 3' end, followed by a 10nt poly(A) tail. After purification of PCR products, they were used as templates for *in vitro* transcription, followed by DNA digest and RNA clean-up. The RNA was then further polyadenylated to ensure no signal spillover from the RNA barcode into the DNA signal. Further details and a visual summary of the protocol are included in **SM.4**.

Direct RNA sequencing and basecalling

The direct RNA sequencing library preparation was started by ligating 100 ng of poly-adenylated barcoded RNA onto a custom RTA adapter. Further library preparation was performed according to ONT's recommendations with the exception of use of MarathonRT as reverse transcription enzyme (see **SM.5**). After motor protein ligation, 7 ul of prepared library was loaded on a Flongle FLO-FLG001 flow cell, and reads were acquired until all pores became inactive. For a detailed description of all steps see **SM.5**. Reads were basecalled post-acquisition with the direct RNA model of ONTs research basecaller "Dorado". See **SM.6** for the full configuration.

WDX-RTA DNA barcode identification

We trained a support vector machine (SVM) to classify the sequenced dual-barcoded RNA reads into RTA barcode categories based on their RTA-barcode signal, using a combination of read mapping and RNA barcode classification to infer their ground truth labels.

To determine ground truth labels, reads were first uniquely mapped to RNA reference sequences with LAST [33]. Then, the RNA barcode region was identified and reads were associated with the RNA barcode that had the smallest Levenshtein distance to them. Reads for which this distance surpassed a defined threshold of 4 were discarded. Based on the specific combinations of RNA and RNA-RTA barcode pairs used during library preparation, the read assignments were used to establish the RTA-barcode ground truth labels required to train the classifier.

Data set selection

Datasets 1 and 2 served as the test sets for the WarpDemuX-DPC and DPC models. For the WarpDemuX models, we designed two distinct test sets: one comprising 600 reads per barcode from datasets 3–5, and another using the entirety of dataset 6, which contains RNA without enzymatic poly-adenylation.

We constructed validation sets for our models, comprising 200 reads per barcode, utilizing data from replicate 2–5 of the DePlexiCon training data, as detailed in Smith et al. [7], for the WarpDemuX-DPC model, and data from datasets 3–5 for the WarpDemuX models (WDX4, WDX12).

Training sets, consisting of 400 reads per barcode, were sampled from data from replicate 2–5 of the DePlexiCon training data [7] for the WarpDemuX-DPC model, and datasets 3–5 for the WarpDemuX models (WDX4, WDX12). First, 400 training instances for the noise class were selected based on the robust Z-score (defined below) of the median inter-class distance per read. Identification of noise instances is further detailed in **SN.1**. Then, to ensure the training set captured the full range of the original data distribution despite its limited size, we used greedy approximations of the Maximal- and Minimal Diversity Problem, stratified across barcode classes (implementation is included in **SN.1**), to select 2x80 instances per barcode that respectively maximized and minimized their mutual distance. The training set was completed with 240 randomly sampled instances per barcode. Supplementary materials provide further details on training set size determination (**SN.1**).

The robust Z-score is calculated as follows, where X_i represents the value under evaluation (median inter-class distance), Median(X) is the dataset's median, and MAD is the Median Absolute Deviation:

$$\text{Robust Z score} = \frac{X_i - \text{Median}(X)}{\text{MAD}}$$

$$\text{MAD} = \text{Median}\left(|X_i - \text{Median}(X)|\right)$$

All sampling processes for validation, training, and test sets were stratified across replicate groups to compensate for variations in sequencing yields, ensuring a balanced representation across all data.

Barcode Classification

SVM training

The RTA barcode classifiers were trained on preprocessed RTA barcode signals. To obtain these RTA barcode signals, we first identify the DNA adapter signal in the raw signal using ADAPTed (`github.com/KleistLab/ADAPTed`) and then segment, normalize and trim it, as described in 'Adapter Signal Processing

and Alignment'. For each RTA barcode signal, we generated a feature representation that quantifies how similar the signal is to the RTA barcode signals of reads with known labels. To do this, DTWDs between the signal and all labelled signals in the training data set are computed.

We then use Support Vector Machines (SVMs) for classification by introducing a Dynamic Time Warping Kernel function, K, which is defined as:

$$K(s_1, s_2) = \exp\left(-\gamma \cdot DDTW(s_1, s_2)\right) \tag{2}$$

where s_1 and s_2 are two preprocessed RTA barcode signal instances, and $\gamma = 1.2$ is a hyper-parameter of the kernel function.

During training, the SVM kernel matrix is based on the pairwise DTWDs between the barcode signals in the training data.

We use an SVM implementation based on libsvm [34], with a one-vs-one scheme for multi-class support, as implemented in Scikit-Learn [35]. Herein, calibrated class membership probability estimates are provided based on an internal 5-fold cross-validation combined with a multiclass extension to Platt scaling [36]. See **SM.3** for further details on the model parameterization.

WDX-RTA barcode classification

During inference, for each read to be classified, $K(s_1, s_2)$ is computed against all barcode signals in the training data, i.e. $s_2 \in B$, with $B = \{B_1, B_2, \dots B_k\}$, where B_i contains all segmented barcode signals that belong to RTA barcode i based on their ground truth label.

Each prediction in our classification system is paired with a confidence score, indicating the reliability of the predicted class label. Reads for which the prediction has a confidence score below a user-defined cutoff are discarded ('unclassified'). For comparability, we adopt the same method as used in DeePlexiCon and define the confidence score as the interval between the classifier's probability estimate of the predicted class and the second most probable class.

Sequencing of SARS-CoV-2 infected cells

To prepare the sequencing samples, 5 x 10^5 Vero E6 TMPRSS2 cells were infected in duplicate with two different recombinant SARS-CoV-2 viruses at MOI=0.01. Briefly, rSARS-CoV-2 WT and rSARS-CoV-2 GFP inoculum (generated as described in [37]) were prepared in DMEM supplemented with 1% FCS. Before the infection, Vero E6 TMPRSS2 cells were washed once with PBS and incubated with the respective inoculum for 1h at 37 °C with gently shake every 10 min. The inoculum was removed and fresh DMEM supplemented with 5% FCS, 100 mg/mL of penicillin and 100U/mL of streptomycin was added to the cells. Then, cells were harvested at 8 and 24 h post infection. A mock control of uninfected cells was also performed. Total RNA was extracted with Trizol (Invitrogen) using the manufacturer's recommendations, precipitated by addition of 0.1 volumes of 3 M NaOAc and 3 volumes of ice cold 100 % EtOH and incubation at -20 °C for >12 h, followed by centrifugation at 16,000 xg for 30 min at 4 °C, a wash with 70 % EtOH and another round of centrifugation before resuspension in 20 ul RNase-free H2O. RNA concentration was quantifed on Nanodrop and 1 ug of total RNA per sample was used for WarpDemuX-multiplexed direct RNA library preparation as described in **SM.5**. The generated reads were demultiplexed with the WDX 12 barcode model at a confidence threshold of 0.8 and reads were aligned with minimap (see **SN.3** for details).

Barcode-specific adaptive sampling

We used the Read Until API (see `github.com/nanoporetech/read_until_api`) to communicate with the MinKNOW server during sequencing. During sequencing, read data is streamed to Read Until channel caches in fixed-duration 'chunks,' and continually added to as sequencing of the read progresses. WarpDemuX analyzes these caches to detect RTA signals and classify barcodes.

Live demultiplexing begins by identifying the transition from adapter to poly(A) tail in the raw signal, utilizing ADAPTed's 'fast' method tailored for streaming data, after which the read is processed and classified.

To minimize the time spent sequencing unwanted reads, we developed a custom MinKNOW protocol with a reduced read chunk duration (100 ms; approx. 300 observations per chunk). Details are included in **SN.4**.

Due to the nature of dRNA-seq sample preparation, not all adapter molecules are ligated to a target molecule, leading to 'adapter-only' reads. These reads, while largely filtered by MinKNOW after sequencing, can skew the real-time WarpDemuX-classified barcode distribution, necessitating correction based on actual sequencer output (`pod5` files). WarpDemuX tracks these outputs to maintain an accurate sample distribution and estimate the sequenced base count per sample. The user can determine the aspect to be balanced and the degree of sensitivity applied.

Data Availability

Raw and basecalled sequencing data will be deposited at the European Nucleotide Archive (ENA).

Code Availability

WarpDemuX was implemented as a Python (3.8) command line tool with Cython-based signal processing to achieve fast CPU run time. Detailed usage and installation instructions as well as test data are provided in the code repository. Source code is available at `github.com/KleistLab/WarpDemuX` and will be deposited at Code Ocean.

Acknowledgements

The authors would like to thank Anke Sparmann for critical feedback, and the HPC Service of ZEDAT [38], Freie Universität Berlin, and Brouwer Art & Technology for computing time.

Conflict of Interest

An international priority patent application was filed jointly by RKI, HZI and FU Berlin on April 26, 2024, at the European Patent Office (EPO) under number PCT/EP2024/061629.

References

[1] Daniel R. Garalde et al. "Highly parallel direct RNA sequencing on an array of nanopores". In: *Nature Methods* 15.3 (Mar. 2018), pp. 201–206.

[2] Patrick Bohn et al. "Nano-DMS-MaP allows isoform-specific RNA structure determination". In: *Nat Methods* 20.6 (June 2023), pp. 849–859.

[3] Oguzhan Begik et al. "Nano3P-seq: transcriptome-wide analysis of gene expression and tail dynamics using end-capture nanopore cDNA sequencing". In: *Nature Methods* 20.1 (2023), pp. 75–85.

[4] Oguzhan Begik, John S. Mattick, and Eva Maria Novoa. "Exploring the epitranscriptome by native RNA sequencing". In: *RNA (New York, N.Y.)* 28.11 (Nov. 2022), pp. 1430–1439.

[5] Aleksandra A Kolodziejczyk et al. "The technology and biology of single-cell RNA sequencing". In: *Molecular cell* 58.4 (2015), pp. 610–620.

[6] Miten Jain et al. "Advances in nanopore direct RNA sequencing". In: *Nature Methods* 19.10 (Oct. 2022), pp. 1160–1164.

[7] Martin A. Smith et al. "Molecular barcoding of native RNAs using nanopore sequencing and deep learning". In: *Genome Research* 30.9 (Sept. 2020), pp. 1345–1353.

[8] Alexander Payne et al. *Barcode aware adaptive sampling for GridION and PromethION Oxford Nanopore sequencers*. 2022.

[9] Meinard Müller. "Dynamic time warping". In: *Information retrieval for music and motion* (2007), pp. 69–84.

[10] Lawrence R Rabiner and Biing-Hwang Juang. *Fundamentals of speech recognition*. Tsinghua University Press, 1999.

[11] Hoang Anh Dau et al. *The UCR Time Series Classification Archive*. Oct. 2018.

[12] Matthew Loose, Sunir Malla, and Michael Stout. "Real-time selective sequencing using nanopore technology". In: *Nature Methods* 13.9 (2016), pp. 751–754.

[13] Albert Perlas et al. "Latest RNA and DNA nanopore sequencing allows for rapid avian influenza profiling". In: *bioRxiv* (2024). DOI: `10.1101/2024.02.28.582540`.

[14] Dongwan Kim et al. "The architecture of SARS-CoV-2 transcriptome". In: *Cell* 181.4 (2020), pp. 914–921.

[15] Samuel Martin et al. "Nanopore adaptive sampling: a tool for enrichment of low abundance species in metagenomic samples". In: *Genome Biology* 23.1 (Jan. 2022), p. 11.

[16] Alexander Payne et al. "Readfish enables targeted nanopore sequencing of gigabase-sized genomes". In: *Nature Biotechnology* 39.4 (Apr. 2021), pp. 442–450.

[17] Isabel S. Naarmann-de Vries et al. "Adaptive sampling for nanopore direct RNA-sequencing". In: *RNA (New York, N.Y.)* 29.12 (Dec. 2023).

[18] Rory Munro et al. "A framework for real-time monitoring, analysis and adaptive sampling of viral amplicon nanopore sequencing". In: *Frontiers in Genetics* 14 (Mar. 2023).

[19] Jiaxu Wang et al. "Direct RNA sequencing coupled with adaptive sampling enriches RNAs of interest in the transcriptome". In: *Nature Communications* 15.1 (Jan. 2024), p. 481.

[20] Sam Kovaka et al. "Targeted nanopore sequencing by real-time mapping of raw electrical signal with UNCALLED". In: *Nature Biotechnology* 39.4 (Apr. 2021), pp. 431–441.

[21] Renmin Han et al. "HycDemux: a hybrid unsupervised approach for accurate barcoded sample demultiplexing in nanopore sequencing". In: *Genome Biology* 24.1 (Oct. 2023), p. 222.

[22] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet classification with deep convolutional neural networks". In: *Communications of the ACM* 60.6 (May 2017), pp. 84–90.

[23] Anna Wierczeiko et al. "NanopoReaTA: a user-friendly tool for nanopore-seq real-time transcriptional analysis". In: *Bioinformatics (Oxford, England)* 39.8 (Aug. 2023), btad492.

[24] Susan Payne. "Methods to Study Viruses". In: *Viruses* (2017), pp. 37–52.

[25] James Brett Case et al. "Growth, detection, quantification, and inactivation of SARS-CoV-2". In: *Virology* 548 (Sept. 2020), pp. 39–48.

[26] Matthew D. Parker et al. "Altered subgenomic RNA abundance provides unique insight into SARS-CoV-2 B.1.1.7/Alpha variant infections". In: *Communications Biology* 5.1 (July 2022), p. 666.

[27] Peter V. Markov et al. "The evolution of SARS-CoV-2". In: *Nature Reviews. Microbiology* 21.6 (June 2023), pp. 361–379.

[28] Chee Hong Wong et al. "Reduced subgenomic RNA expression is a molecular indicator of asymptomatic SARS-CoV-2 infection". In: *Communications Medicine* 1 (2021), p. 33.

[29] Nora Schmidt et al. "SND1 binds SARS-CoV-2 negative-sense RNA and promotes viral RNA synthesis through NSP9". In: *Cell* 186.22 (Oct. 2023), 4834–4850.e23.

[30] Jui-Hung Shien, Yi-Da Su, and Hung-Yi Wu. "Regulation of coronaviral poly(A) tail length during infection is not coronavirus species- or host cell-specific". In: *Virus Genes* 49.3 (Dec. 2014), pp. 383–392.

[31] Hung-Yi Wu et al. "Regulation of coronaviral poly(A) tail length during infection". In: *PloS One* 8.7 (2013), e70548.

[32] Wannes Meert et al. *DTAIDistance*. Oct. 2022. DOI: `10.5281/zenodo.7158824`.

[33] Michiaki Hamada et al. "Training alignment parameters for arbitrary sequencers with LAST-TRAIN". In: *Bioinformatics* 33.6 (Nov. 2016), pp. 926–928.

[34] Chih-Chung Chang and Chih-Jen Lin. "LIBSVM: A library for support vector machines". In: *ACM Transactions on Intelligent Systems and Technology* 2.3 (Apr. 2011), pp. 1–27.

[35] Fabian Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12.85 (2011), pp. 2825–2830.

[36] Ting-Fan Wu, Chih-Jen Lin, and Ruby C. Weng. "Probability Estimates for Multi-class Classification by Pairwise Coupling". In: *The Journal of Machine Learning Research* 5 (Dec. 2004), pp. 975–1005.

[37] Marco Olguin-Nava et al. "Building Blocks of Understanding: Constructing a Reverse Genetics Platform for studying determinants of SARS-CoV-2 replication". In: *bioRxiv* (2024).

[38] Loris Bennett, Bernd Melchers, and Boris Proppe. "Curta: A General-purpose High-Performance Computer at ZEDAT, Freie Universität Berlin". In: (2020). DOI: `10.17169/refubium-26754`.

[39] Panu Somervuo et al. "BARCOSEL: a tool for selecting an optimal barcode set for high-throughput sequencing". In: *BMC Bioinformatics* 19.1 (July 5, 2018), p. 257.

[40] Maximilian Krause. *Flongle DirectRNA Library preparation*. 2020. URL: `https://dx.doi.org/10.17504/protocols.io.bcwcixaw`.

[41] Aimee L. Jalkanen, Stephen J. Coleman, and Jeffrey Wilusz. "Determinants and implications of mRNA poly(A) tail size – Does this protein make my tail look big?" In: *Seminars in Cell & Developmental Biology* 34 (), pp. 24–32.

[42] Ivano Legnini et al. "FLAM-seq: full-length mRNA sequencing reveals principles of poly(A) tail length control". In: *Nature Methods* 16.9 (), pp. 879–886.

[43] Maximilian Krause et al. "tailfindr: alignment-free poly(A) length measurement for Oxford Nanopore RNA and DNA sequencing". In: *RNA* 25.10 (2019), pp. 1229–1241.

[44] Wang Liu-Wei et al. "Sequencing accuracy and systematic errors of nanopore direct RNA sequencing". In: *bioRxiv* (2023).