Supplementary Tables

Data set	Sequencing run	RNA template	Replicate	RTA barcode	RNA barcode
1	1	*V. chol.* glycine riboswitch	1	DPC_1	bc13
				DPC_2	bc14
				DPC_3	bc15
				DPC_4	bc16
2	2	Tetrahymena ribozyme	2	DPC_1	bc04
				DPC_2	bc05
				DPC_3	bc17
				DPC_4	bc18
3	3	*V. chol.* glycine riboswitch	1	WDX_bc01	bc12
				WDX_bc02	bc04
				WDX_bc03	bc05
				WDX_bc04	bc03
				WDX_bc05	bc06
				WDX_bc06	bc07
				WDX_bc07	bc01
				WDX_bc08	bc09
				WDX_bc09	bc02
				WDX_bc10	bc10
				WDX_bc11	bc11
				WDX_bc12	bc08
4	3	hc16	2	WDX_bc01	bc10
				WDX_bc02	bc09
				WDX_bc03	bc01
				WDX_bc04	bc08
				WDX_bc05	bc05
				WDX_bc06	bc07
				WDX_bc07	bc11
				WDX_bc08	bc03
				WDX_bc09	bc06
				WDX_bc10	bc04
				WDX_bc11	bc02
				WDX_bc12	bc12
5	4	HCV IRES	3	WDX_bc01	bc05
				WDX_bc02	bc06
				WDX_bc03	bc07
				WDX_bc04	bc01
				WDX_bc05	bc11
				WDX_bc06	bc03
				WDX_bc07	bc04
				WDX_bc08	bc10
				WDX_bc09	bc02
				WDX_bc10	bc09
				WDX_bc11	bc08
				WDX_bc12	bc12
6	4	bact. RNase P (not poly-adenylated)	4	WDX_bc01	bc09
				WDX_bc02	bc11
				WDX_bc03	bc10
				WDX_bc04	bc08
				WDX_bc05	bc12
				WDX_bc06	bc04
				WDX_bc07	bc01
				WDX_bc08	bc07
				WDX_bc09	bc02
				WDX_bc10	bc05
				WDX_bc11	bc03
				WDX_bc12	bc06

Table S1: Run-RNA-barcode to RTA assignments for RTA training/evaluation data

DPC-RTA	**Replicate1** *V. chol* gly Polyadenylated	**Replicate2** tetrahymena Polyadenylated	Total
Barcode 1	601	1856	2457
Barcode 2	600	2482	3082
Barcode 3	191	3141	3332
Barcode 4	558	975	1533
Total	1950	8454	10404

Table S2: Generated experimental DPC-RTA data, coverage per barcode and replicate after mapping and adapter detection.

WDX-RTA	**Replicate1** *V. chol* gly Polyadenylated	**Replicate2** hc16 Polyadenylated	**Replicate3** HCV IRES Polyadenylated	**Replicate4** bact RNase P Not polyadenylated	Total
Barcode 01	548	2604	411	872	4435
Barcode 02	633	3493	712	1476	6314
Barcode 03	433	3125	868	894	5320
Barcode 04	652	3078	188	1182	5100
Barcode 05	111	2132	237	285	2765
Barcode 06	162	963	626	1480	3231
Barcode 07	888	4806	701	1207	7602
Barcode 08	3120	2962	703	1894	8679
Barcode 09	324	5839	427	1305	7895
Barcode 10	1006	2802	857	2073	6738
Barcode 11	588	2459	888	1649	5584
Barcode 12	570	5575	360	749	7254
Total	9035	39838	6978	15066	70917

Table S3: Generated experimental WDX-RTA data, coverarge per barcode and replicate after mapping and adapter detection.

Barcode ID	Barcode sequence
Barcode 1	GGCTTCTTCTTGCTCTTAGG
Barcode 2	GTGATTCTCGTCTTTCTGCG
Barcode 3	GTACTTTTCTCTTTGCGCGG
Barcode 4	GGTCTTCGCTCGGTCTTATT

Table S4: DeePlexiCon RTA barcodes, sequences in 5' to 3' direction [7].

Barcode ID	Barcode sequence
Barcode 1	TTTTTACTGCCAGTGACT
Barcode 2	AGGGGAGAGAGCCCCCCC
Barcode 3	CACGTCATTTTCCACGTC
Barcode 4	GGAGGCCAGGCGGACCGA
Barcode 5	ACGGACCTTTTGACTTAA
Barcode 6	TATTGCATACTGCGCCGC
Barcode 7	CCACGGAGGGAGGATTGG
Barcode 8	TTACCGGCAGTGACGGAC
Barcode 9	CGAGATTGCATCCCCCCC
Barcode 10	TACCACCTGCCGGCGGCC
Barcode 11	GCCCGCCGGGGGAGAAGC
Barcode 12	TTTTTTTTACCGGCAGTT

Table S5: Optimized WarpDemuX barcodes, sequences in 5‘ to 3‘ direction.

Supplementary Methods

SM.1 Designing WarpDemuX Barcodes

Experimental design

The RTA adapter consists of two partially complementary oligos. To ensure compatibility with the sequencing chemistry we designed the adapter oligos similarly to the tested design described in Smith et al. [7], with the exception of allowing variation of GC content. In detail, the two last nucleotides of the 5' end of the forward RTA adapter (GG) were kept constant to reduce potential ligation biases. In addition, all adapters retained the 3' adapter sequence that is required for the RMX motor protein to ligate onto. The reverse adapter was designed as reverse complement of the forward adapter, with the exception of a 10 nucleotide dT 3' overhang to hybridize to the RNA poly(A) tail, as well as a Y anchor sequence 5' of the reverse complement RTA barcode. All oligos were ordered from IDT with standard desalting purification, and with forward adapters containing a 5' phosphorylation. The following scheme exemplifies the design:

forward adapter	/5Phos/GG–[*WDX barcode*]–GGTAGTAGGTTC
reverse adapter	GAGGCGAGCGGTCAATTTT–[*revcomp–WDX barcode*]–CCTTTTTTTTTT

Table SM.1.1: Custom RTA design scheme (5' to 3' orientation for both oligos)

In silico design

Barcode optimization addresses the need for a set of distinguishable signals that can be easily detected and separated. For barcodes of length L, an exhaustive search across all possible sets of k barcodes has complexity $\binom{4^L}{k}$, making such a search computationally infeasible: $\binom{6.9\times10^{10}}{12} \approx 2.3 \times 10^{121}$ for $L = 18$, $k = 12$.

Instead, we constructed a set of wave-like target signal patterns since we expect such patterns to improve segmentation, as well as distinction between barcodes and thus make for good barcodes in the context of our classification method.

Target Patterns Derivation

Across target patterns, we combined wave-like elements with diverse frequencies and amplitudes. Additionally, we maintained consistent minimal and maximal values across all target patterns to mitigate potential normalization biases during signal processing.

The customizable section of the RTA that serves as barcode spans 18 nucleotides. Given the sequencing k-mer size of 6, however, the preceding five constant nucleotides and the following constant two, as well as the first bases of the subsequent polyA tail also influence its signal. As a result, we expect 20 DNA and 3 DNA-polyA signal events. As there is no ONT k-mer model for mixed DNA-polyA k-mers, we assumed the signal of these k-mers to be an interpolation between the value of the last DNA k-mer and the expected signal level of a pure polA segment ($\sim$ 110 pA). When constructing the target patterns, we substituted the interpolation k-mers with a singular polyA segment observation, thus yielding patterns that span 21 observations.

Candidate Set Construction

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. These models are designed for DNA sequenced in the 5' to 3' direction. In dRNA-seq, however, the RTA is sequenced in 3' to 5' direction. To increase the likelihood that our modeled barcode signals reflect the sequencing data, we limited our barcode design to the subset of k-mers for which the 5'-3' to 3'-5' directional reversal likely has little effect. Specifically, we selected k-mers with minimal variation (less than 10 pA) in expected signal mean when compared to their mirrored counterparts. For example, if the expected signal level difference between the k-mer AAACCC and the k-mer CCCAAA is < 10 pA, then the k-mer AAACCC is selected.

Subsequently, we discretized the expected signal values for the selected k-mers (1524 in total) and their mirrored counterparts respectively into five equally-sized, rank-based buckets. Only those k-mers whose original and mirrored versions both fell within the lowest, middle, or highest rank buckets were further considered, leaving us with 531 k-mers. This discretization and selection strategy enhances the probability of identifying wave-like patterns–marked by alternating low and high values–when assembling the k-mers into barcodes. Additionally, this approach strengthens the stringency of our heuristic for ensuring robustness against directional reversal.

From the filtered set of k-mers, we generated all conceivable combinations of five k-mers, adhering to a stride of three. This means that the final three bases of one k-mer overlap with the initial three bases of the following k-mer. This approach yielded roughly 3.4×10^6 candidate barcode sequences, each 18 bases long.

For each, we constructed the full adapter sequence by pre- and appending the flanking regions of adapter. Using the ONT k-mer models we constructed the putative adapter signal and appended it with a singular observation of the expected polyA tail signal. The full adapter signal was normalized to zero mean and unit variance. We then selected the last 21 observations of the idealized adapter signal, corresponding to the variable barcode part and calculated the distance to each target pattern.

Candidate Selection

To select the set of $k = 12$ barcodes we selected the top 50 best-scoring (lowest distance) candidate barcodes per target pattern. For the $k = 12$ target patterns highlighted in Fig. 2A, this yielded a total of 600 unique barcodes, for which we computed all 600^2 pairwise DTWDs.

Lastly, within the matrix of 600^2 pairwise DTWDs we greedily approximate the Maximal Diversity Problem (MDP) to propose 12 candidate barcodes that we expect to be accurately identifiable and distinguishable. The greedy heuristic used to approximate the MDP was implemented as in Algorithm SM.1.1.

Algorithm SM.1.1: Greedy heuristic for solving the Maximum Diversity Problem.

input : Distance matrix $D \in \mathbb{R}_{\geq 0}^{n \times n}$, Number of elements to select m
output: Indices of the selected elements
$n \leftarrow$ number of rows in D;
first $\leftarrow \max_i(\sum_j D_{ij})$; `// Select the element with the maximum sum of distances`
selected $\leftarrow$ [first];
while *length of selected* $< m$ **do**
  maxMinDist $\leftarrow -1$;
  nextIndex $\leftarrow -1$;
  for $i \leftarrow 0$ **to** $n - 1$ **do**
    if $i \notin$ *selected* **then**
      minDist2Selected $\leftarrow \min\{D_{ij} : j \in \text{selected}\}$;
      if *minDist2Selected* $>$ *maxMinDist* **then**
        maxMinDist $\leftarrow$ minDist2Selected;
        nextIndex $\leftarrow i$;
      end
    end
  end
  Append *nextIndex* to selected;
end
return *selected*;

SM.2 WarpDemuX Signal Processing

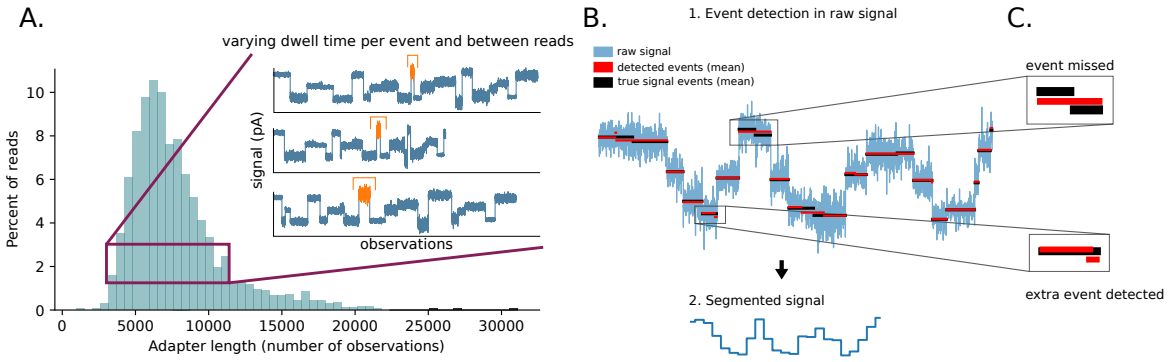

Figure SM.2.1: Intrinsic variation in dwell time poses a source of noise. (A) Varying dwell time per event and between reads results in varying length adapter signals. (B) Segmentation of the raw signal into events, with each segment represented by its average signal, reduces signal noise caused by differences in dwell time and within-event measurement noise (C) Signal noise causes imperfect segmentation in which some events are missed while other spurious events are detected.

To account for the inherent variation in dwell time (Fig. SM.2.1A), the raw signal is segmented (Fig. SM.2.1B) by identifying the most significant shifts in current level based on the running difference between neighboring windows of raw signal. The pseudo code is described in SM.2.1. Its parameters were tuned to reflect the translocation speed of the RTA, with a minimal segment length of 15 observations a sliding window width of 30 observations. Considering the known length of the adapter sequence, the signal is expected to contain 98 events. However, because segmentation is likely to be imperfect (Fig. SM.2.1C), with missed events posing a greater challenge than spurious events, we intentionally oversegment the signal by approximately 10% to detect 110 events. Each segmented event is then represented by its average signal. Post-segmentation, segment means are normalized to zero mean and unit variance. Only the variable barcode component of the segmented signal, expected to be represented within the last 25 segments, based on the known barcode sequence length and location in the adapter, is used for classification.

Algorithm SM.2.1: Greedy algorithm for raw signal segmentation based on the most significant shifts in signal intensity.

input : Raw signal X of length N
$l \leftarrow 6$; `// Minimal segment length`
$k \leftarrow 12$; `// Width of sliding window`
$n \leftarrow 110$; `// Number of segmentation points to identify`
$\mathcal{W} \leftarrow \emptyset$; `// Empty list`
$\mathcal{T} \leftarrow \emptyset$;
$\mathcal{C} \leftarrow [0, N]$;
$\mathcal{S} \leftarrow \emptyset$;
for $i \leftarrow 1$ **to** $N - k$ **do**
  $w_i \leftarrow X[i : i + k]$;
  Append w_i to $\mathcal{W}$;
end
for $i \leftarrow 1$ **to** $N - 2k$ **do**
  $\bar{w}_i \leftarrow$ mean of w_i;
  $\sigma_i^2 \leftarrow$ variance of w_i;
  $\bar{w}_{i+1} \leftarrow$ mean of w_{i+1};
  $\sigma_{i+1}^2 \leftarrow$ variance of w_{i+1};
  $t_{i,i+1} \leftarrow \frac{|\bar{w}_{i+1} - \bar{w}_i|}{\sqrt{\sigma_i^2 + \sigma_{i+1}^2}}$;
  Append $t_{i,i+1}$ to $\mathcal{T}$;
end
while *size of* $\mathcal{C} < n$ **do**
  Find index i of max absolute value in T;
  Append $i + k$ to $\mathcal{C}$;
  Exclude values within range l of i from T;
end
Sort values in $\mathcal{C}$;
for *each consecutive pair* (c_1, c_2) *in* $\mathcal{C}$ **do**
  $s \leftarrow$ mean of $X[c_1 : c_2]$;
  Append s to $\mathcal{S}$;
end
return $\mathcal{S}$;

Segmentation ensures that all inputs to the classification procedure are of uniform length. Due to the signal normalization, pairwise DTWDs are of similar magnitude across barcodes, negating the need for further scaling or normalization of the pairwise distance matrix.

SM.3 WarpDemuX Parameterization

SVM

We use an C-Support Vector Classification SVM implementation based on libsvm [34], with a one-vs-one scheme for multi-class support, as implemented in Scikit-Learn [35] (scikit-learn 1.3.1).

With `C`, the regularization parameter, set to 10.0, `kernel = 'precomputed'` and default settings otherwise.

DTWD

To compute the Dynamic Time Warping Distance (DTWD) between two barcode fingerprints, we use the Python Package `dtaidistance 2.3.10` [32], with a window size of 15 (upper bound to allowed shift from the diagonal) and a additive warping penalty of 0.1 per compression or expansion.

SM.4 Generating in vitro transcribed barcoded RNA

RNA barcoding strategy

In order to train and benchmark RTA-based barcoding strategies we generated in vitro transcribed RNAs of different templates containing varying RNA-encoded barcodes. This allows us to generate high accuracy ground truth labels for different RTA signals. To ensure that no signal spillover from RNA barcode into RTA signal was occuring we decided to enzymatically poly(A) tail these RNAs post in vitro transcription (although they already contain a 10 nt poly(A) tail that can be used for direct ligation of RTA adapters). Fig. SM.4.1 provides a visual summary of the protocol. Further, we also decided to randomize RNA-barcode to RTA assignments to eliminate the possibility of inadvertently training on RNA signal in the rare possibility of RTA adapter signal failure S1 .

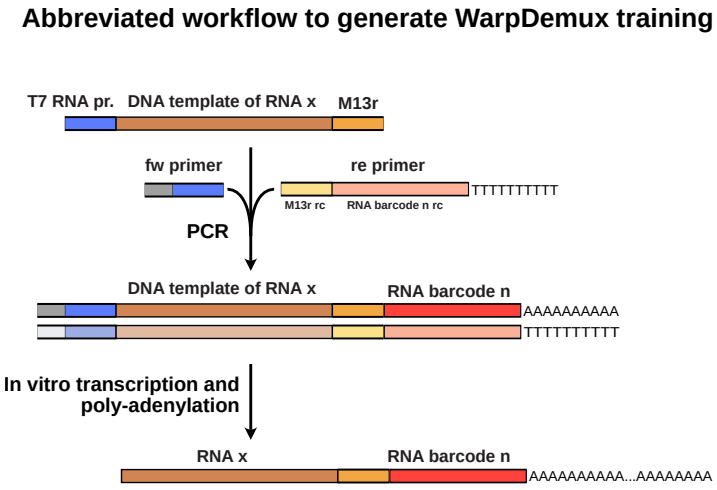

Figure SM.4.1: Abbreviated workflow of experimental protocol to generate WarpDemuX training data.

RNA barcode design

The RNA barcode sequences were designed with high RNA sequencing error rates in mind. Specifically, a large pool of 8 nt long barcode candidates was generated by random sampling, with some additional candidates supplemented by BARCOSEL [39], and a set of 18 barcodes that maximized inter-barcode Levenshtein edit distances was chosen. To balance purine and pyrimidine content in each barcode and double the number of variable nucleotides in the final barcodes a self-complementary stem-loop (i.e. [barcode]-GAGUA-[revcomp_barcode]) was generated for each barcode. To introduce these RNA barcodes into DNA templates reverse primers containing a 10 nt long T stretch at their 5' end, followed by the reverse complement of the barcode stem-loop, and finally the reverse complement of the M13-reverse sequence were ordered with standard desalting purity from IDT (Table SM.4.1).

RNA barcode ID	Barcode unique sequence	Reverse primer for PCR
RNA_bc01	GATAGTGCgagtaGCACTATC	TTTTTTTTTTGATAGTGCTACTCGCACTATCcaggaaacagctatgaccatg
RNA_bc02	GCAATACTgagtaAGTATTGC	TTTTTTTTTTGCAATACTTACTCAGTATTGCcaggaaacagctatgaccatg
RNA_bc03	GATTATCAgagtaTGATAATC	TTTTTTTTTTGATTATCATACTCTGATAATCcaggaaacagctatgaccatg
RNA_bc04	AGGTCAGGgagtaCCTGACCT	TTTTTTTTTTAGGTCAGGTACTCCCTGACCTcaggaaacagctatgaccatg
RNA_bc05	GAGGCTCCgagtaGGAGCCTC	TTTTTTTTTTGAGGCTCCTACTCGGAGCCTCcaggaaacagctatgaccatg
RNA_bc06	GCAACCAGgagtaCTGGTTGC	TTTTTTTTTTGCAACCAGTACTCCTGGTTGCcaggaaacagctatgaccatg
RNA_bc07	GACTCCATgagtaATGGAGTC	TTTTTTTTTTGACTCCATTACTCATGGAGTCcaggaaacagctatgaccatg
RNA_bc08	GTGTAATCgagtaGATTACAC	TTTTTTTTTTGTGTAATCTACTCGATTACACcaggaaacagctatgaccatg
RNA_bc09	ATGACGAAgagtaTTCGTCAT	TTTTTTTTTTATGACGAATACTCTTCGTCATcaggaaacagctatgaccatg
RNA_bc10	CATGTTGAgagtaTCAACATG	TTTTTTTTTTCATGTTGATACTCTCAACATGcaggaaacagctatgaccatg
RNA_bc11	CGAACTTCgagtaGAAGTTCG	TTTTTTTTTTCGAACTTCTACTCGAAGTTCGcaggaaacagctatgaccatg
RNA_bc12	TATCTGTGgagtaCACAGATA	TTTTTTTTTTTTGAGTACTACTCGTACTCAAcaggaaacagctatgaccatg
RNA_bc13	GACGGACCgagtaGGTCCGTC	TTTTTTTTTTGACGGACCTACTCGGTCCGTCcaggaaacagctatgaccatg
RNA_bc14	GAGGTCTGgagtaCAGACCTC	TTTTTTTTTTGAGGTCTGTACTCCAGACCTCcaggaaacagctatgaccatg
RNA_bc15	GCTCAATCgagtaGATTGAGC	TTTTTTTTTTGCTCAATCTACTCGATTGAGCcaggaaacagctatgaccatg
RNA_bc16	GACGATACgagtaGTATCGTC	TTTTTTTTTTGACGATACTACTCGTATCGTCcaggaaacagctatgaccatg
RNA_bc17	GACCGCCTgagtaAGGCGGTC	TTTTTTTTTTGACCGCCTTACTCAGGCGGTCcaggaaacagctatgaccatg
RNA_bc18	GAGTCCTTgagtaAAGGACTC	TTTTTTTTTTGAGTCCTTTACTCAAGGACTCcaggaaacagctatgaccatg

Table SM.4.1: RNA barcode design and reverse PCR primers to generate in vitro transcription templates. The constant M13r sequence is in lowercase.

PCR to generate barcoded DNA templates

PCR was performed on templates of the *Vibrio cholerae* glycine riboswitch, hc16 ligase, bacterial RNase P type A, tetrahymena riboswitch and Hepatitis C Virus IRES cloned into pJet (Thermo Fisher) flanked by a T7 RNA promoter sequence and an M13r sequence at 5' and 3' ends respectively as described in Bohn et al. [2] using Primestar GXL (Takara) according to the manufacturer's instructions. Reaction volume was 20 ul, annealing temp. 50 °C, extension time 30 seconds at 35 cycles with forward primers described in Table SM.4.2 and reverse primers as described in Table SM.4.1.

Template	FW Primer Sequence
HCV IRES	AAAGAAGACTTGGGGTAATACGACTCACTATAGGCCAGCCCCCGATTG
V. chol. Glycine riboswitch	TTCTAATACGACTCACTATAGGTCCGTTGAAGACTGCAGGAGAGTGG
bact. Rnase P	TTCTAATACGACTCACTATAGGAGAGGAGCAGGC
hc16	TTCTAATACGACTCACTATAGGTAGACTCGCAGGAAGTCTACCGAGTAAGAGAAAGAGGA
Tetrahymena ribozyme	TTCTAATACGACTCACTATAGGAGGGAAAAGTTATCAGGCATGCACCTGGT

Table SM.4.2: Forward PCR primers to generate in vitro transcription templates.

In vitro transcription

PCR products were quality controlled via agarose gel and purified by addition of 18 ul H2O and 38 ul SPRI beads (Omega Biotek NGS). After hybridization for 5 min under light agitation (300 rpm on hula mixer) beads were pelleted using a DynaMag-96 Side Magnet (Invitrogen). Supernatant was removed, beads were washed twice with 100 ul 80% EtOH and eluted in 20 ul H2O. DNA concentration was measured with Nanodrop and 250 fmol of DNA was then used in a 10 ul in vitro transcription reaction: 5 mM NTPs, 5 U/ul T7 RNA polymerase (homemade), 1 ul YIPP (NEB), 0.1 ul RNasin (Molox) in 40 mM Tris pH 7.5, 18 mM $MgCl_2$, 10 mM DTT, 1 mM spermidine, 5 mM NTPs, 40 U RNasin (Molox) and homemade T7 RNA polymerase. After in vitro transcription for 2 h at 37 °C the DNA template was digested by addition of 10 ul of DNase mix containing 1 ul 10x DNase I buffer and 0.2 ul of DNase I (NEB) followed by a 15 min incubation at 37 °C. The RNA was then purified by addition of 1.4 volumes of SPRI beads and two washes with 80% EtOH. Purified RNA was then quality controlled with a 1.5% agarose gel in 1x TAE and concentration was quantified via Nanodrop.

Poly(A) tailing

For poly(A) tailing 1 ug of IVT RNA was incubated with 0.5 ul E. coli poly(A) Polymerase (NEB), 1 ul 10 mM ATP in 1x poly(A) polymerase reaction buffer in a 10 ul reaction for 30 min at 37 °C, followed by

purification with 1.4 volumes of SPRI beads and two washes in 190 ul 80% EtOH before elution in 20 ul RNase-free H2O.

SM.5 Direct RNA sequencing library preparation

Annealing of RTA adapters

The forward and reverse strands of the RTA oligos were annealed following a procedure similar to that described in Smith et al. [7]. Specifically, oligos were diluted to 10 uM each in 30 mM HEPES-KOH (pH 7.5), 100 mM K-Acetate, heated to 95 °C for 1 min and cooled to 25 °C at a rate of 0.5 °C/min. The annealed RTA adapters were then diluted to 1.4 uM with RNase-free H2O and stored at -20 °C until use.

RTA adapter ligation

Volumes were scaled for direct RNA sequencing on a Flongle flow cell according to the protocols.io protocol of Krause [40] 2020. In detail, for each sample, 100 ng of poly-adenylated in vitro transcribed RNA in 4.4 ul, was transferred into a PCR tube, followed by addition of 1 ul previously annealed custom RTA adapter, 1.6 ul 5x NEBNext Quick Ligation Buffer (NEB) and 1 ul T4 DNA Ligase high concentration (NEB). The reaction was mixed well by pipetting and incubated at room temperature for 20 min. To stop the ligation 2 ul of 0.5 M EDTA was added to each reaction, samples were pooled, purified with 0.8 volumes of SPRI beads washed twice with 80 % EtOH, and eluted in 20 ul RNase-free H2O.

Reverse Transcription

To perform the reverse transcription a master mix containing 1.9 ul H2O, 15.7 ul 3x MarathonRT buffer (150 mM Tris-HCl pH 8.3, 600 mM KCl, 60 % (v/v) glycerol), 2.35 ul 100 mM DTT, 0.2 ul RNasin (Molox), 1.9 ul 50 mM MgCl2, 3 ul of 10 mM dNTPs and 2 ul MarathonRT (produced in house as described in [2]) was prepared and added to the 20 ul RNA-RTA ligation. After incubation at 42 °C for 60 min the reaction was stopped by addition of 2 ul Proteinase K (NEB), followed by another incubation at 42 °C for 15 min. The reaction was purified with 1 volume SPRI beads, washed thrice with 190 ul 80 % EtOH, and eluted in 30 ul RNase-free water (volume was increased to adjust for higher total input amount due to multiplexing).

Motor protein ligation

To proceed with Nanopore sequencing we next ligated on the motor protein. For this we transferred 10 ul of purified reverse-transcribed RNA-RTA library into a 1.5 ml DNA LoBind tube (Eppendorf), followed by addition of 2.5 ul H2O, 4 ul 5x NEBNext Quick Ligation Buffer (NEB), 2 ul RMX motor protein adapter (ONT SQK-RNA002), and 1.5 ul T4 DNA Ligase high conc. (NEB). The reaction was mixed by pipetting and incubated for 20 min at RT, followed by purification with 20 ul SPRI beads, washed twice with RNA wash buffer (WSB, ONT SQK-RNA002). The library was then eluted in 9 ul elution buffer (EB, ONT SQK-RNA002) and 8 ul was transferred into a new 1.5 ml DNA LoBind tube. The library was then loaded onto a Flongle FLO-FLG001 flow cell for sequencing.

SM.6 Data Acquisition and Basecalling software configuration

- MinKNOW `22.12.7`
- dorado `0.3.4`
- protocol `MIN106_RNA:FLO-FLG001:SQK-RNA002`
- dorado model: `rna002_70bps_hac@v3`