Coral: a dual context-aware basecaller for nanopore direct RNA sequencing

Table of contents

Supplementary Note 1 Computation of $P(y|x)$ using the forward algorithm

Specifically, we introduce the forward variable $\alpha_{i,j} = \sum_{a \epsilon \mathcal{A}} P(y_0^i, a_0^i, a_i = j|x)$, representing the marginal probability for $P(y_0^i|x)$ when the alignment position of k-mer y_i is j. It follows:

$$P(y|x) = \alpha_{\boldsymbol{l},\boldsymbol{t}}$$

The calculation of $\alpha_{i,j}$ ($i \in [1, \boldsymbol{l}]$, $j \in [1, \boldsymbol{t}]$) is detailed as follows:

When $i > j$, $\alpha_{i,j} = 0$ (strict monotonic alignment)

When $i = 1$, $\alpha_{i,j}$ is given by:

$$\alpha_{i,j} = P\left(a_i = j \middle| y_0^{i-1}, x\right) * P\left(y_i \middle| y_0^{i-1}, a_i = j, x\right)$$

This can be further expressed as:

$$\alpha_{i,j} = P\left(y_i|s_i, h_j\right) * e_{i,j} \prod_{m=1}^{j-1} (1 - e_{i,m})$$

When $i > 1$ and $j \geq i$, $\alpha_{i,j}$ can be calculated based on the α_{i-1} obtained in the previous iteration:

$$\alpha_{i,j} = \sum_{k=1}^{j-1} \alpha_{i-1,k} * P\left(a_i = j \middle| a_{i-1} = k, y_0^{i-1}, x\right) * P\left(y_i \middle| y_0^{i-1}, a_i = j, x\right)$$

This can be simplified to:

$$\alpha_{i,j} = P\left(y_i|s_i, h_j\right) * e_{i,j} \sum_{k=1}^{j-1} \alpha_{i-1,k} \prod_{m=k}^{j-1} (1 - e_{i,m})$$

In logarithmic space, to accelerate the computation of the loss function, we employed a set of parallel operators for the efficient calculation of $\log \alpha_{i,j}$. The details are provided in the Supplementary Note 2.

Supplementary Note 2 Accelerated computation of $\log \alpha_{i,j}$

When $i > j$, $\log \alpha_{i,j} = -\infty$

When $i = 1$, $\log \alpha_{i,j} = \log P(y_i|s_i, h_j) + \log e_{i,j} + \sum_{m=1}^{j-1} \log(1 - e_{i,m})$

When $i > 1$,

$$\log \alpha_{i,j\geq i} = \log P(y_i|s_i, h_j) + \log e_{i,j} + \log \sum_{k=1}^{j-1} \alpha_{i-1,k} \prod_{m=k}^{j-1} (1 - e_{i,m})$$

$$= \log P(y_i|s_i, h_j) + \log e_{i,j} + \log \sum_{k=1}^{j-1} \alpha_{i-1,k} \frac{\prod_{m=1}^{j-1}(1-e_{i,m})}{\prod_{m=1}^{k-1}(1-e_{i,m})}$$

$$= \log P(y_i|s_i, h_j) + \log e_{i,j} + \log \prod_{m=1}^{j-1}(1 - e_{i,m}) \sum_{k=1}^{j-1} \frac{\alpha_{i-1,k}}{\prod_{m=1}^{k-1}(1-e_{i,m})}$$

$$= \log P(y_i|s_i, h_j) + \log e_{i,j} + \sum_{m=1}^{j-1} \log(1 - e_{i,m}) + \log \sum_{k=1}^{j-1} \frac{\alpha_{i-1,k}}{\prod_{m=1}^{k-1}(1-e_{i,m})}$$

$$= \log P(y_i|s_i, h_j) + \log e_{i,j} + \sum_{m=1}^{j-1} \log(1 - e_{i,m}) + \log \sum_{k=1}^{j-1} \exp\{\log \frac{\alpha_{i-1,k}}{\prod_{m=1}^{k-1}(1-e_{i,m})}\}$$

$$= \log P(y_i|s_i, h_j) + \log e_{i,j} + \sum_{m=1}^{j-1} \log(1 - e_{i,m}) + \log \sum_{k=1}^{j-1} \exp\{\log \alpha_{i-1,k} - \sum_{m=1}^{k-1} \log(1 - e_{i,m})\}$$

We define several auxiliary vectors as follows:

$\rho_i = \{\log P(y_i|s_i, h_1),\ \log P(y_i|s_i, h_2), \dots,\ \log P(y_i|s_i, h_T)\}\ (i \geq 1)$

$\eta_i = \{\log e_{i,1},\ \log e_{i,2}, \dots,\ \log e_{i,T}\}\ (i \geq 1)$

$\mu_i = \{0,\ \log(1 - e_{i,1}),\ \log(1 - e_{i,2}), \dots,\ \log(1 - e_{i,T-1})\}\ (i \geq 1)$,

$\nu_i = \{\sum_{m=1}^{1} \mu_{i,m},\ \sum_{m=1}^{2} \mu_{i,m},\ \dots,\ \sum_{m=1}^{T} \mu_{i,m}\}\ (i \geq 1)$

$o_i = \{-\infty,\ \log \alpha_{i-1,1} - \nu_{i,1},\ \log \alpha_{i-1,2} - \nu_{i,2}, \dots, \log \alpha_{i-1,T-1} - \nu_{i,T-1}\}\ (i > 1)$

$\pi_i = \{\log \sum_{m=1}^{1} \exp\{o_{i,m}\},\ \log \sum_{m=1}^{2} \exp\{o_{i,m}\}, \dots,\ \log \sum_{m=1}^{T} \exp\{o_{i,m}\}\}\ (i > 1)$

where ν_i can be obtained by a parallel $cumsum$ operator on the vector μ_i, and π_i can be obtained by a parallel $logcumsumexp$ operator on the vector o_i.

Consequently, the computation of the vector $\log \alpha_i$ is as follows (includes the case of $\log \alpha_{i,j<i} = -\infty$) :

When $i = 1$, $\log \alpha_i = \rho_i + \eta_i + \nu_i$

When $i > 1$, $\log \alpha_i = \rho_i + \eta_i + \nu_i + \pi_i$

Supplementary Note 3 Pseudo-code of decoding algorithm

Input: Raw signal x, Max target length $\hat{\boldsymbol{l}}$, Encoder hidden state dimension d_h, Decoder hidden state dimension d_s, Beam size b, K-mer alphabet $\mathbb{Y}$, $y_0 = \langle sos \rangle$, Latent variable sequence a
Output: Decoding k-mer sequence y^*

BEGIN
Initialize score matrix $\mathrm{Z} \in \mathbb{R}^{\hat{\boldsymbol{l}} \times \boldsymbol{t}} = \mathbf{0}_{\hat{\boldsymbol{l}} \times \boldsymbol{t}}$
Initialize prediction $C \in \mathbb{Y}^{\hat{\boldsymbol{l}} \times \boldsymbol{t}}$
Initialize beams alignment $A \in \mathbb{N}^{\hat{\boldsymbol{l}} \times \mathrm{b}} = \mathbf{0}_{\hat{\boldsymbol{l}} \times \mathrm{b}}$
Initialize backtrack point $\mathrm{Q} \in \mathbb{N}^{\hat{\boldsymbol{l}} \times \boldsymbol{t}} = \mathbf{0}_{\hat{\boldsymbol{l}} \times \boldsymbol{t}}$
Initialize decoder hidden states $s \in \mathbb{R}^{\hat{\boldsymbol{l}} \times \boldsymbol{t} \times d_s}$

$h = h(x) \in \mathbb{R}^{\boldsymbol{t} \times d_h}$ // encoder hidden states with length $\boldsymbol{t}$

$s_{1,\mathrm{j} \in [1,\boldsymbol{t}]} = \mathrm{Decoder}(\mathbf{0},\ y_0)$ // decoder use the previous hidden state and the current input

for j = 1 **to** t **do**
$Z_{1,j} = \max_{y_1 \in \mathbb{Y}} P(a_1 = j | s_{1,j}, h) * P(y_1 | s_{1,j}, h_j)$
$C_{1,j} = \operatorname{argmax}_{y_1 \in \mathbb{Y}} P(a_1 = j | s_{1,j}, h) * P(y_1 | s_{1,j}, h_j)$
$s_{2,j} = \mathrm{Decoder}(s_{1,\mathrm{j}}, C_{1,j})$ // updated decoder hidden states
end for

$A_1 = \operatorname{argtopk}_{j \in [1,\boldsymbol{t}]}(Z_{1,j}, \mathrm{top}k = \mathrm{b})$ // first K-mer alignment with top-k scores

for $i = 2$ **to** $\hat{\boldsymbol{l}}$ **do**
for $j = i$ **to** $\boldsymbol{t}$ **do**
for m **in** A_{i-1} **and** $m < j$ **do**
$val = \max_{y_i \in \mathbb{Y}} Z_{i-1,m} * P(a_i = j | a_{i-1} = m, s_{i,m}, h) * P(y_i | s_{i,m}, h_j)$
if $val > Z_{i,j}$ **then**
$Z_{i,j} = val$
$C_{i,j} = \operatorname{argmax}_{y_1 \in \mathbb{Y}} Z_{i-1,m} * P(a_i = j | a_{i-1} = m, s_{i,m}, h) * P(y_i | s_{i,m}, h_j)$
$\mathrm{Q}_{i,j} = m$
end if
end for

if $i < \hat{\boldsymbol{l}}$ **do**
$s_{i+1,j} = \mathrm{Decoder}\left(s_{i,\mathrm{Q}_{i,j}}, C_{i,j}\right)$
end if
end for
$A_i = \operatorname{argtopk}_{j \in [1,\boldsymbol{t}]}(Z_{i,j}, \mathrm{top}k = \mathrm{b})$
end for

$\mathrm{o}^* = \operatorname{argmax}_{\mathrm{o} \in [1,\hat{\boldsymbol{l}}]} \frac{\log Z_{\mathrm{o},\boldsymbol{t}}}{\mathrm{o}}$ // use length normalization to penalize short sequence

Obtain y^* by backtracking from $Z_{\mathrm{o}^*,\boldsymbol{t}}$

END

Table S1. Details of training dataset

Species	Flow cell version	Library kit	Data link
Arabidopsis	R9.4.1	SQK-RNA002	RNA dataset generated by Taiyaki: https://zenodo.org/records/4556885/files/taiyaki-rna.hdf5?download=1 RNA training and validation data: https://zenodo.org/records/4556951/files/rna-train.hdf5?download=1 https://zenodo.org/records/4556951/files/rna-valid.hdf5?download=1
H. Sapiens[1]	R9.4	SQK-RNA001	
Epinano synthetic	R9.4.1	SQK-RNA001	
C. elegans	R9.4	SQK-RNA001	
E. coli	R9.4	SQK-RNA001	

1. from the NA12878 project (BHAM Run1)

Table S2. Details of test datasets

Species	Flow cell version	Library kit	Transcriptome reference url	Fast5 dataset url
Arabidopsis	R9.4.1	SQK-RNA001	https://zenodo.org/records/4557005/files/transcriptomes.tgz?download=1	https://zenodo.org/records/4557005/files/arabidopsis-dataset.tgz?download=1
H. Sapiens[1]	R9.4	SQK-RNA001	https://ftp.ebi.ac.uk/pub/databases/gencode/Gencode_human/release_45/gencode.v45.transcripts.fa.gz	https://zenodo.org/records/4557005/files/human-dataset.tgz?download=1
Mus musculus	R9.4.1	SQK-RNA001	https://zenodo.org/records/4557005/files/transcriptomes.tgz?download=1	https://zenodo.org/records/4557005/files/mouse-dataset.tgz?download=1
S. cerevisiae S288C	R9.4.1	SQK-RNA002	https://zenodo.org/records/4557005/files/transcriptomes.tgz?download=1	https://zenodo.org/records/4557005/files/yeast-dataset.tgz?download=1
Populus trichocarpa	R9.4.1	SQK-RNA001	https://zenodo.org/records/4557005/files/transcriptomes.tgz?download=1	https://zenodo.org/records/4557005/files/poplar-dataset.tgz?download=1
Danio rerio	R9.4.1	SQK-RNA002	https://www.ncbi.nlm.nih.gov/datasets/genome/GCF_000002035.6	https://zenodo.org/records/11632496/files/zebrafish-dataset.zip?download=1

1. from the NA12878 project (HOPKINS RUN1)

Table S3. Basecalling performance of Coral and Coral-CTC on test datasets.

Dataset	Reads Count	Tool	Read Accuracy (%)		Mismatch (%)		Insertion (%)		Deletion (%)		Read Length (bp)		Unaligned Reads
Human	100000	Coral	**94.76**	**95.08**	**1.26**	**1.07**	**1.5**	**1.23**	**2.48**	**2.19**	**980**	**724**	**412**
		Coral-CTC	91.66	92.84	1.75	1.57	1.7	1.55	4.89	3.73	939	691	1232
Mouse	100000	Coral	**88.87**	**89.39**	3.5	3.3	2.63	2.37	**5**	**4.76**	**652**	**527**	**1290**
		Coral-CTC	87.61	88.47	**3.2**	**3.08**	**1.75**	**1.62**	7.44	6.46	576	464	4590
Yeast	100000	Coral	**91.6**	**92.57**	2.61	**2.21**	2.35	2.06	**3.43**	**3.09**	**852**	**605**	**1301**
		Coral-CTC	90.11	91.37	**2.56**	2.33	**1.78**	**1.65**	5.55	4.31	785	558	4677
Arabidopsis	100000	Coral	**93.78**	**94.24**	**1.61**	**1.36**	1.8	1.49	**2.8**	**2.59**	**972**	**861**	**470**
		Coral-CTC	91.65	92.69	1.86	1.65	**1.66**	**1.45**	4.83	3.95	918	802	1812
Poplar	100000	Coral	**91.41**	**92.45**	**2.36**	**2.03**	2.57	2.1	**3.65**	**3.23**	**794**	**665**	**778**
		Coral-CTC	89.98	91.24	2.45	2.21	**2.07**	**1.75**	5.49	4.42	747	625	3357
Zebrafish	100000	Coral	**88.6**	**89.57**	3.07	2.77	1.87	1.61	**6.46**	**5.7**	**964**	**831**	**1658**
		Coral-CTC	86.36	88.08	**2.85**	**2.64**	**1.41**	**1.23**	9.39	7.52	878	742	6580

1. The best results are marked as bold text.
2. The left and right side of the columns “Read Accuracy”, “Mismatch”, “Insertion”, “Deletion”, and “Read Length” represent the mean and median values respectively.

Table S4. Details of human transcriptome analysis datasets

Cell line	Dataset	Library kit	Data link
HEK293T	SGNex_Hek293T_directRNA_replicate2_run1	SQK-RNA001	http://sg-nex-data.s3.amazonaws.com/data/sequencing_data_ont/fast5/SGNex_Hek293T_directRNA_replicate2_run1/SGNex_Hek293T_directRNA_replicate2_run1.tar.gz
NA12878/ GM12878	JHU Runs 1,2,3,4,5; Notts Runs 1,2,3,4,5; OICR Runs 1,2,3,4,5; UBC Runs 1,2,3,4,5	SQK-RNA001	https://github.com/nanopore-wgs-consortium/NA12878/blob/master/nanopore-human-transcriptome/fastq_fast5_bulk.md
A549	SGNex_A549_directRNA_replicate6_run1	SQK-RNA002	http://sg-nex-data.s3.amazonaws.com/data/sequencing_data_ont/fast5/SGNex_A549_directRNA_replicate6_run1/SGNex_A549_directRNA_replicate6_run1.tar.gz
MCF-7	SGNex_MCF7_directRNA_replicate2_run3	SQK-RNA002	http://sg-nex-data.s3.amazonaws.com/data/sequencing_data_ont/fast5/SGNex_MCF7_directRNA_replicate2_run3/SGNex_MCF7_directRNA_replicate2_run3.tar.gz

Table S5. Basecalling performance of Guppy v6.5.7, RODAN v1.0, and Coral v1.0 on Human datasets used in downstream analysis.

Dataset	Reads Count	Tool	Read Accuracy (%)		Mismatch (%)		Insertion (%)		Deletion (%)		Read Length (bp)		Unaligned Reads
HEK293T	1116713	Guppy	91.03	91.78	2.23	1.98	2.47	2.22	4.27	3.85	919	**705**	80060
		Coral	**94.63**	**95.09**	**1.37**	**1.12**	**1.53**	**1.23**	**2.47**	**2.16**	911	699	**69240**
		RODAN	92.12	92.91	1.92	1.7	2.95	2.21	3.01	2.66	**920**	704	78519
NA12878	8540683	Guppy	89.53	90.48	2.63	2.31	2.23	2	5.61	5.01	**990**	**734**	2523486
		Coral	**94.24**	**95.06**	**1.42**	**1.07**	**1.68**	**1.26**	**2.67**	**2.2**	967	709	2176024
		RODAN	92.29	93.23	1.82	1.54	2.24	1.66	3.65	3.09	918	671	**2108927**
A549	1254612	Guppy	89.19	89.87	2.92	2.68	2.61	2.37	5.28	4.86	1106	853	22539
		Coral	**92.68**	**93.22**	**1.97**	**1.7**	**1.87**	**1.6**	**3.49**	**3.18**	**1107**	**856**	**17934**
		RODAN	90.28	91.26	2.58	2.31	2.97	2.13	4.17	3.71	1046	825	31110
MCF-7	7230921	Guppy	85.05	85.53	4.56	4.36	2.63	2.45	7.76	7.4	**728**	**595**	2007657
		Coral	**89.03**	**89.18**	**3.36**	**3.16**	2.12	1.9	**5.5**	**5.31**	693	562	**1507463**
		RODAN	86.71	87.1	3.89	3.81	**1.94**	**1.71**	7.45	6.93	563	478	1672852

1. The best results are marked as bold text.
2. The left and right side of the columns “Read Accuracy”, “Mismatch”, “Insertion”, “Deletion”, and “Read Length” represent the mean and median values respectively.

Table S6. Details of the top-10 abundant annotated transcripts uniquely discovered by Coral from the HEK293T dataset.

Chromosome	Transcript ID	Transcript Name	Transcript Length (bp)	Abundance	Coral Accuracy (%)	Guppy Accuracy (%)	RODAN Accuracy (%)
chr3	ENST00000338970.10	RPL14-201	7072	1387.99	**97.6**	92	94.65
chr11	ENST00000322028.5	POLR2L-201	876	344	**94.1**	91.95	92.24
chr19	ENST00000246802.10	NOP53-201	1504	341.97	**95.38**	92.82	93.26
chr19	ENST00000330133.5	MRPL54-201	608	201	**93.92**	91.43	92
chr16	ENST00000254108.12	FUS-201	1824	163.85	**94.57**	92.57	92.23
chr19	ENST00000587708.7	PSENEN-202	1124	137.15	**92.15**	90.17	90.05
chr19	ENST00000242783.11	PKN1-201	3120	132	**93.52**	91.62	91.56
chr2	ENST00000409240.5	DCTN1-203	4365	99.76	**94.04**	92.51	92.41
chr10	ENST00000370355.3	SCD-201	5245	98	**92.54**	91.07	91.55
chr11	ENST00000279281.8	VPS51-201	2661	95	**94**	92.28	92.17

1. The best results are marked as bold text.
2. Abundance is reported as the read count estimate of a transcript isoform. Each read contributes at most 1 count, either to a single transcript isoform or distributed as fractional counts to multiple transcript isoforms.
3. “Coral Accuracy”, “Guppy Accuracy” and “RODAN Accuracy” refer to the median read accuracy (by transcriptome alignment) for the reads assigned to an annotated transcript discovered only in Coral.

Table S7. Details of the top-10 abundant annotated transcripts uniquely discovered by Coral from the NA12878 dataset.

Chromosome	Transcript ID	Transcript Name	Transcript Length (bp)	Abundance	Coral Accuracy (%)	Guppy Accuracy (%)	RODAN Accuracy (%)
chr6	ENST00000376809.10	HLA-A-203	1535	4117.04	**96.62**	89.83	93.42
chr6	ENST00000376806.9	HLA-A-202	1854	2154.44	**96.81**	90.09	93.68
chr14	ENST00000216802.10	PSME2-201	793	1808.15	**94.76**	91.79	93.44
chr6	ENST00000706900.1	HLA-A-218	1472	580.05	**96.85**	89.97	93.48
chr17	ENST00000304992.11	PRPF8-201	7280	569.01	**93.47**	90.64	92.31
chr1	ENST00000251507.8	RABGAP1L-201	2899	373.4	**93.81**	91.36	92.95
chr3	ENST00000346169.7	EIF4G1-202	5405	366.7	**93.71**	90.21	92.01
chr8	ENST00000259089.9	BLK-201	2215	332.89	**94.04**	90.67	92.39
chr3	ENST00000265028.8	DNAJB11-201	1640	247.59	**94.24**	92.06	93.8
chr11	ENST00000529230.6	CKAP5-210	7172	225	**93.49**	90.99	92.27

1. The best results are marked as bold text.
2. Abundance is reported as the read count estimate of a transcript isoform. Each read contributes at most 1 count, either to a single transcript isoform or distributed as fractional counts to multiple transcript isoforms.
3. “Coral Accuracy”, “Guppy Accuracy” and “RODAN Accuracy” refer to the median read accuracy (by transcriptome alignment) for the reads assigned to an annotated transcript discovered only in Coral.

Table S8. Details of the top-10 abundant annotated transcripts uniquely discovered by Coral from the A549 dataset.

Chromosome	Transcript ID	Transcript Name	Transcript Length (bp)	Abundance	Coral Accuracy (%)	Guppy Accuracy (%)	RODAN Accuracy (%)
chr3	ENST00000305366.8	TM4SF1-201	1555	829.99	**91.78**	89.37	90.6
chr15	ENST00000267970.9	TSPAN3-201	6348	221.68	**92.55**	90.49	91.45
chr17	ENST00000306749.4	FASN-201	8464	191	**91.87**	90.18	90.68
chr22	ENST00000216218.8	ST13-201	3212	155	**93.44**	91.27	92.56
chr22	ENST00000252934.10	ATXN10-201	3294	151	**93.51**	91.35	92.64
chr2	ENST00000234420.11	MSH6-201	4265	138.85	**93.5**	91.37	92.42
chr1	ENST00000426263.10	SLC2A1-203	3384	122.43	**91.6**	89.29	90.5
chr2	ENST00000409240.5	DCTN1-203	4365	120.31	**93.33**	90.71	91.75
chr17	ENST00000225726.10	CCDC47-201	3283	119.43	**93.06**	91.04	91.63
chr17	ENST00000578552.6	GPS1-208	1841	118.46	**92.63**	90.07	91.25

1. The best results are marked as bold text.
2. Abundance is reported as the read count estimate of a transcript isoform. Each read contributes at most 1 count, either to a single transcript isoform or distributed as fractional counts to multiple transcript isoforms.
3. “Coral Accuracy”, “Guppy Accuracy” and “RODAN Accuracy” refer to the median read accuracy (by transcriptome alignment) for the reads assigned to an annotated transcript discovered only in Coral.

Table S9. Training configuration and basecaller configuration

Coral training configuration	
Batch size	30
Epoch number	18
Signal chunk length	4096
Training batches number	33333
Validation batches number	1000
Optimizer	Ranger
Optimizer weight decay	0.01
Initial learning rate	0.002
Learning rate scheduler	ReduceLROnPlateau
ReduceLROnPlateau patience	1
ReduceLROnPlateau factor	0.5
ReduceLROnPlateau threshold	0.1
ReduceLROnPlateau min lr	1e-5
Guppy (v6.5.7) basecalling configuration	
Download url	https://hub.docker.com/r/genomicpariscentre/guppy-gpu/tags?name=6.5.7
Basecalling command	guppy_basecaller -i [fast5_dir] --recursive -s [output_dir] --device cuda:0 -c rna_r9.4.1_70bps_hac.cfg -q 0 --disable_qscore_filtering
RODAN (v1.0) basecalling configuration	
Download url	https://github.com/biodlab/RODAN
Basecalling command	python RODAN/basecall.py -m RODAN/rna.torch -a RODAN/rnaarch [fast5_dir] > [output_dir]/rodan.fasta
Coral (v1.0) basecalling configuration	
Download url	https://github.com/BioinfoSZU/Coral
Basecalling command	python Coral/coral.py --prefix coral --gpu 0 [fast5_dir] [output_dir]

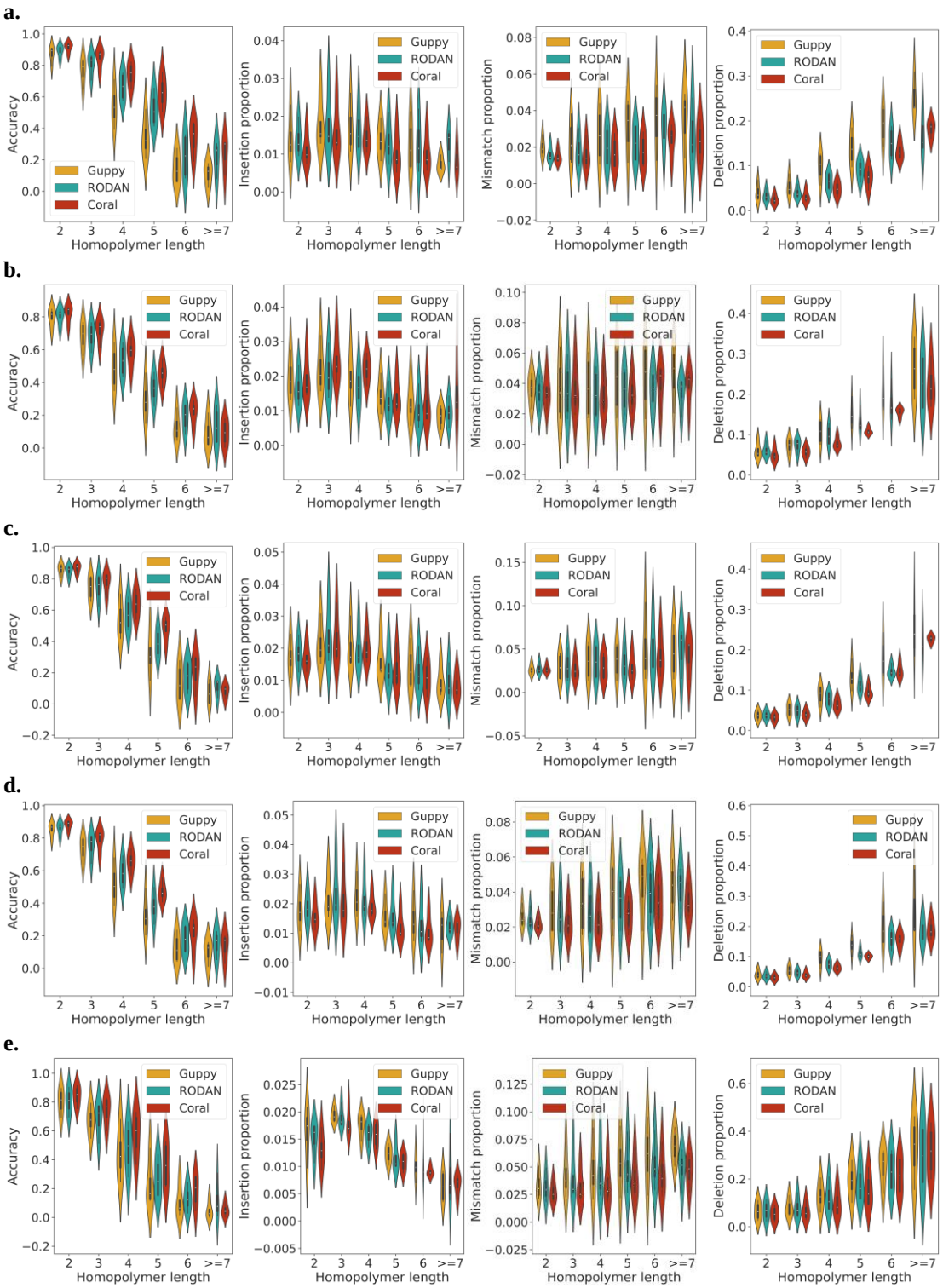

Fig. S1: Comparison of accuracy, insertion proportion, mismatch proportion, and deletion proportion for homopolymers of varying lengths across Guppy, RODAN, and Coral. a, Results for the Arabidopsis dataset. **b,** Results for the Mouse dataset. **c,** Results for the Yeast dataset. **d,** Results for the Poplar dataset. **e,** Results for the Zebrafish dataset.

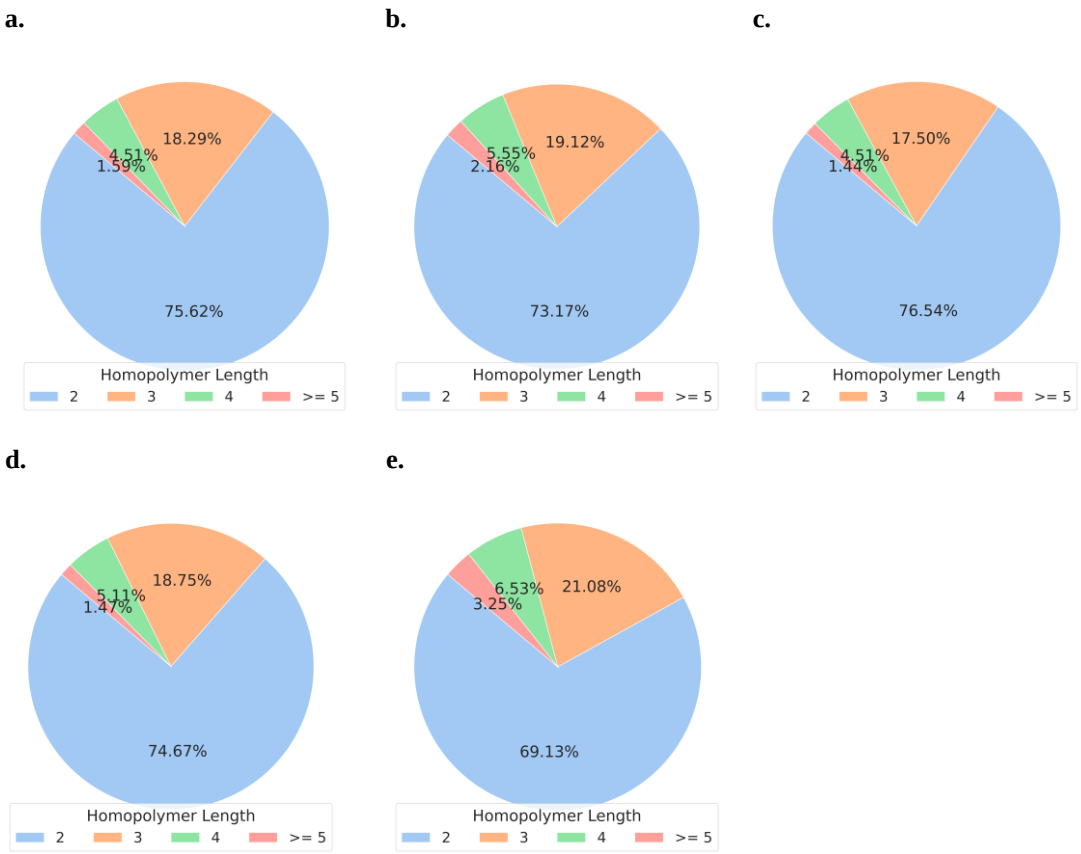

Fig. S2: Proportion of different homopolymer lengths. a, Results for the Arabidopsis dataset. **b,** Results for the Mouse dataset. **c,** Results for the Yeast dataset. **d,** Results for the Poplar dataset. **e,** Results for the Zebrafish dataset.

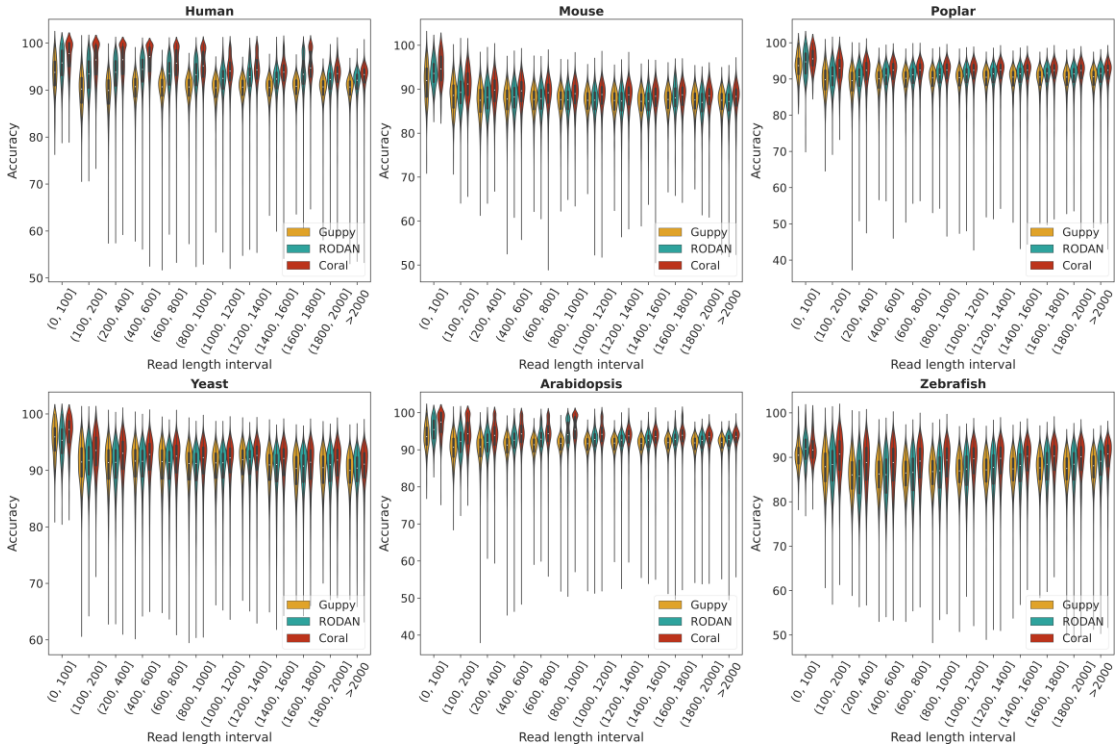

Fig. S3: The violin plots showing the basecalling accuracy as a function of read length for the six testing datasets.

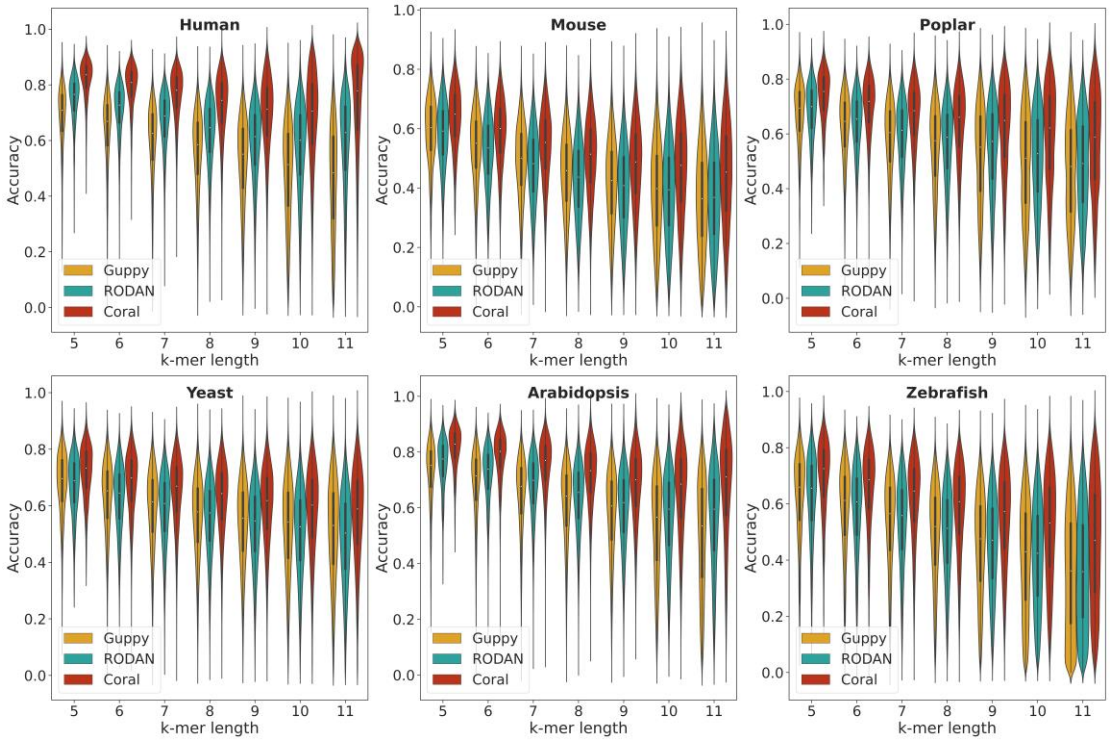

Fig. S4: The violin plots showing the accuracy of k-mer predictions (from lengths 5 to 11) for Coral, Guppy, and RODAN in different testing samples.

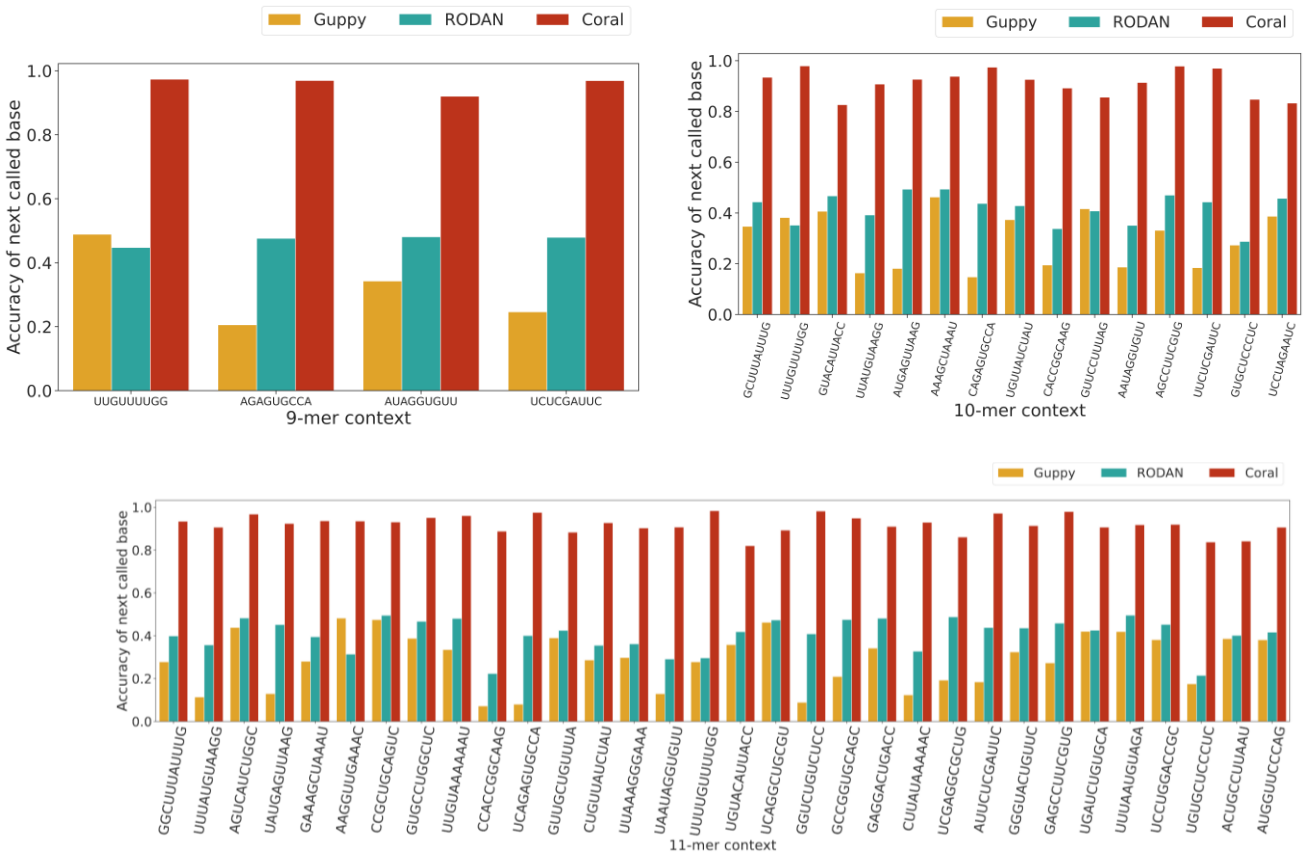

Fig. S5: Bar plots depicting the accuracy of the next called base given several specific 9-mer, 10-mer and 11-mer contexts in the human test dataset.

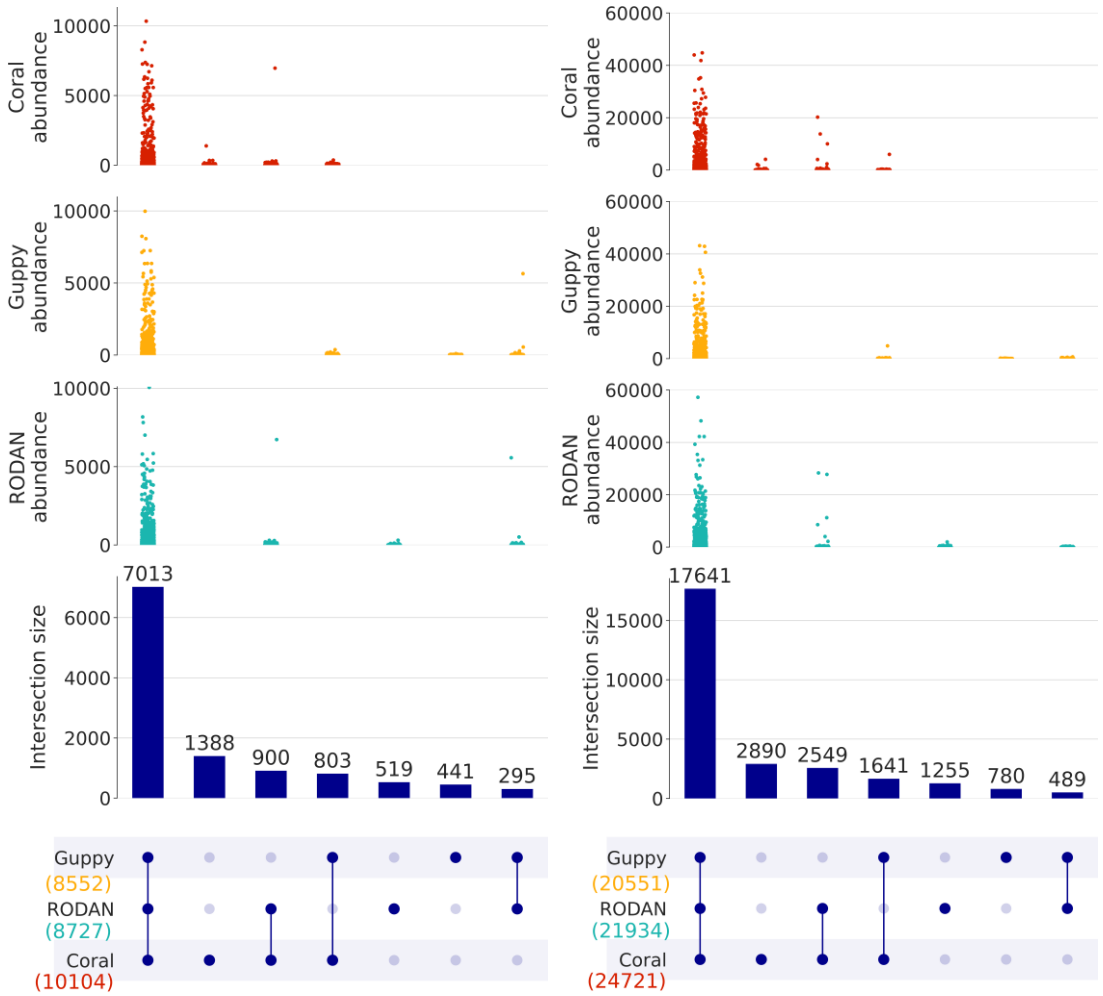

Fig. S6: Upset plots showing the overlap and unique annotated transcripts discovered by Guppy, RODAN, and Coral in the HEK293T (left) and NA12878 (right) datasets. The total number of annotated transcripts is shown in parentheses, with each method distinguished by a unique color. The estimated abundance across different transcript sets is shown in strip plots.

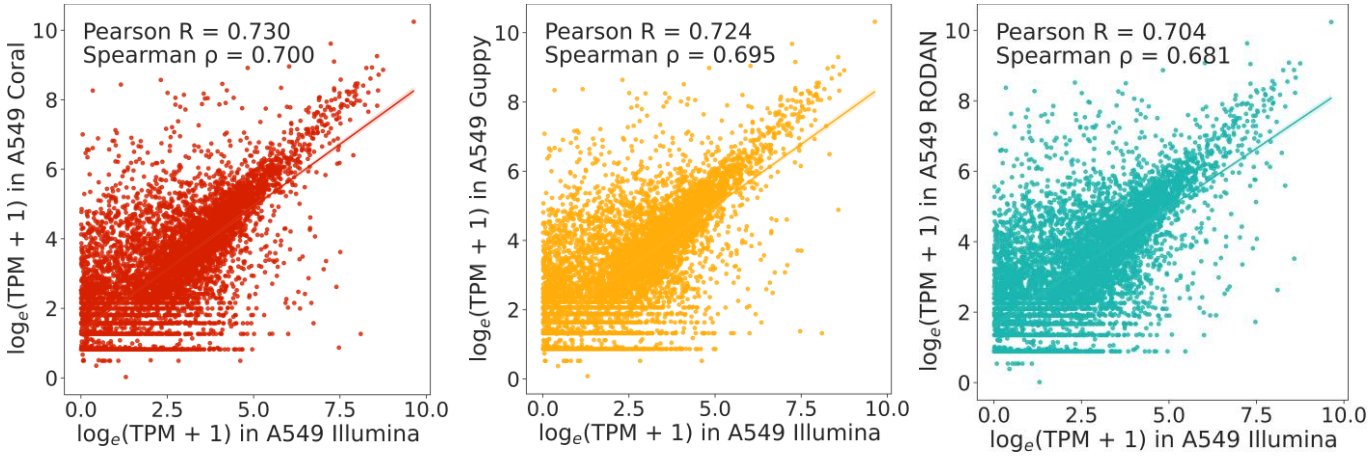

Fig. S7: Correlation between transcript expression estimates obtained from Coral/Guppy/RODAN reads (using ESPRESSO) and that obtained from Illumina RNA-seq reads (using Salmon) for all annotated transcripts detected by both long read and short read in the A549 dataset.

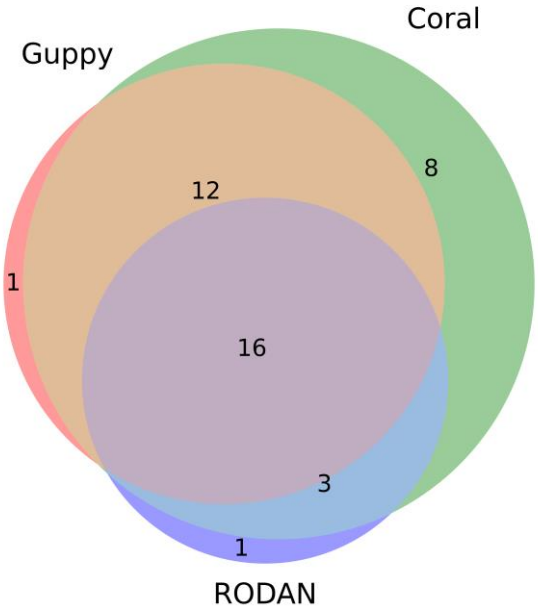

Fig. S8: Venn diagram of previously validated fusion genes rediscovered by Guppy, Coral, and RODAN using JAFFAL in the MCF-7 cancer line.

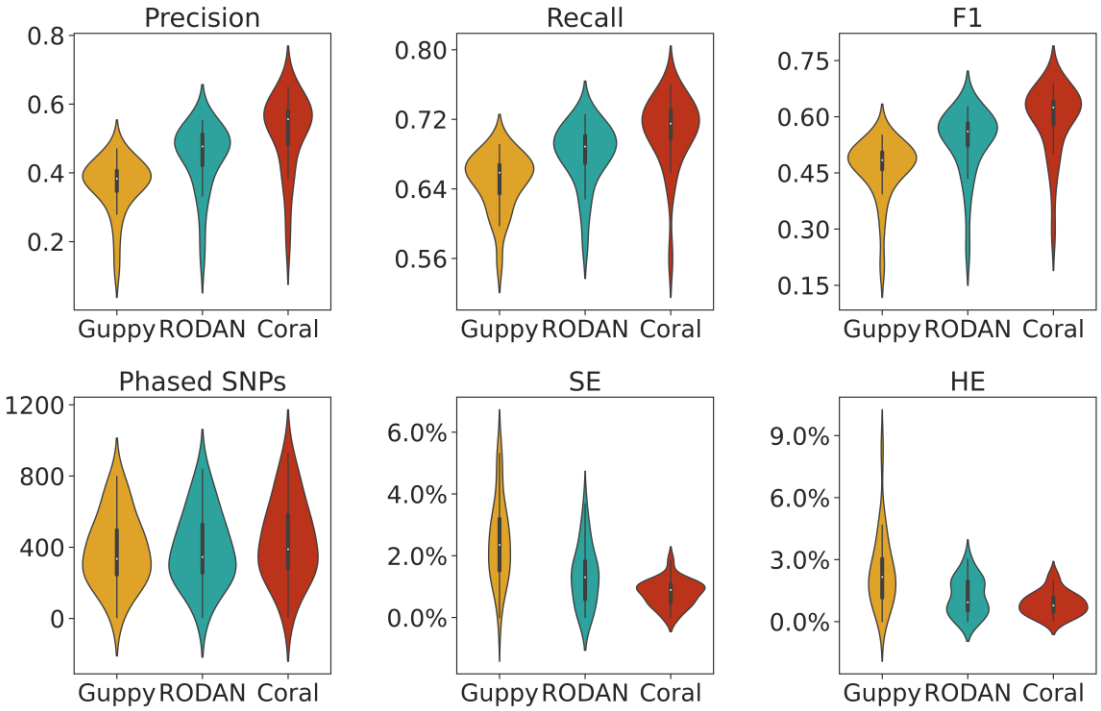

Fig. S9: Violin plots of SNP identification and haplotype performance compared to GIAB gold-standard across different chromosomes using ONT DRS data from GM12878. Precision, Recall, and F1 score were calculated by comparing the chromosome position, allele, and genotype between unphased SNPs against the GIAB gold standard. SE and HE, i.e., switch error rate and Hamming error rate, were calculated by comparing the haplotype differences between the phased blocks and the standard at the identical heterozygous variants.

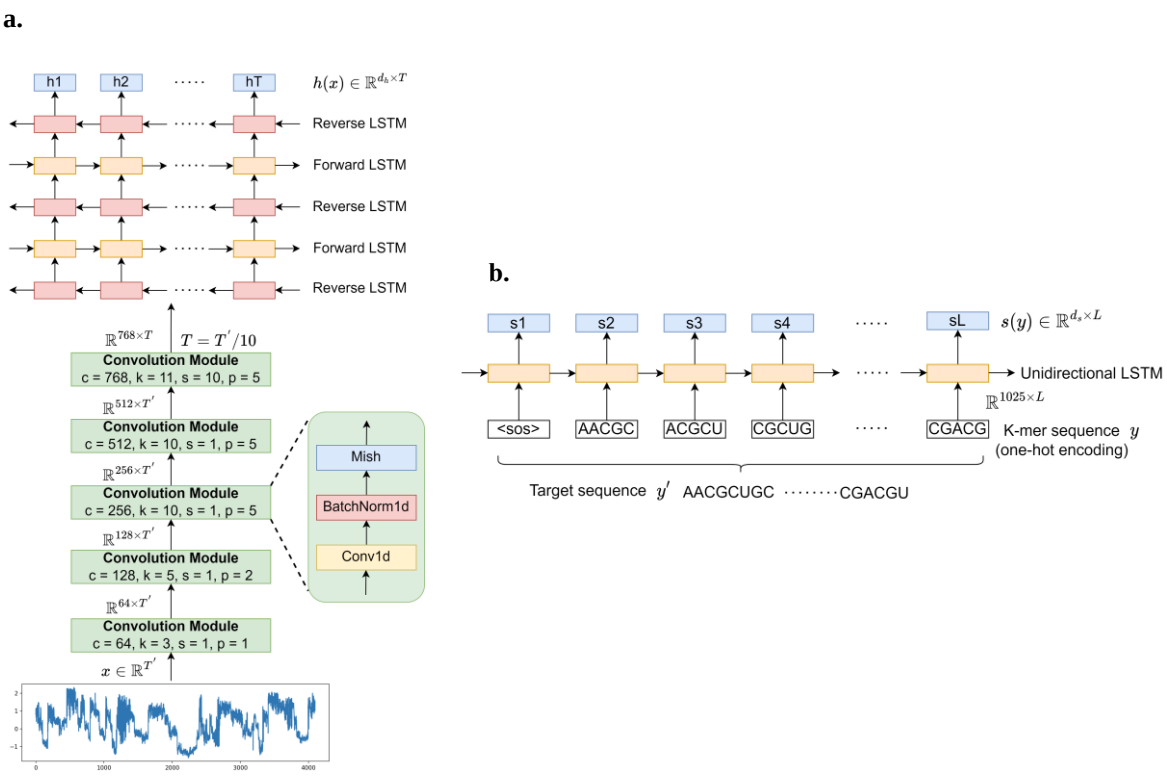

Fig. S10: Detailed architecture of Encoder and Decoder RNN in Coral model. a, The architecture of the encoder. The labels ('c', 'k', 's', and 'p') in the convolution module are represented as channel number, kernel size, stride size, and padding size respectively. **b,** The architecture of the decoder RNN. Target ribonucleotide sequence y' is converted to a k-mer (k=5) sequence y, and $y_0 = \langle \text{sos} \rangle$ is the placeholder representing the start of the seq.

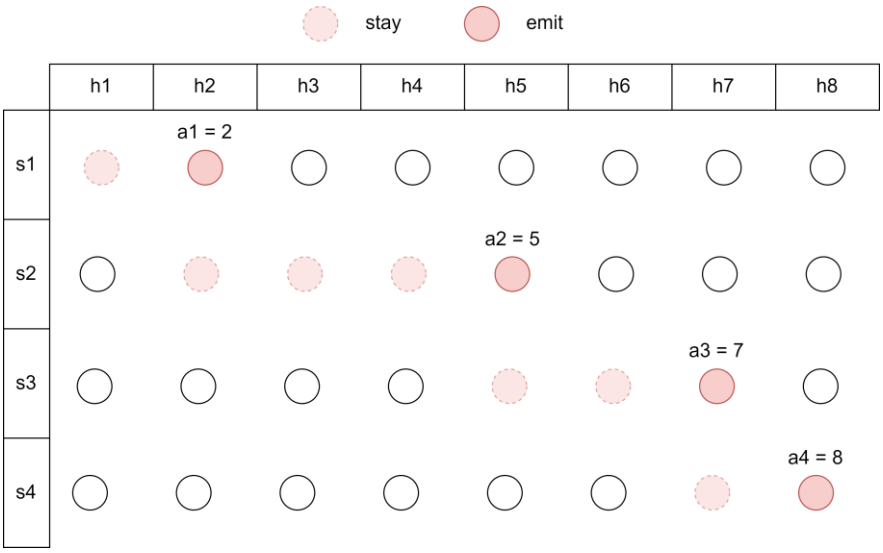

Fig. S11: Visualization of latent alignment variable transition using stay/emit states . The deep red nodes represent 'emit' states and light red dashed nodes represent 'stay' states. The sequence of latent alignment variables is $(a_0 = 0, a_1 = 2, a_2 = 5, a_3 = 7, a_4 = 8)$, and the corresponding state sequence is $(z_{1,1} = stay, z_{1,2} = emit, z_{2,2} = stay, z_{2,3} = stay, z_{2,4} = stay, z_{2,5} = emit, z_{3,5} = stay, z_{3,6} = stay, z_{3,7} = emit, z_{4,7} = stay, z_{4,8} = emit)$. In this case, $P(a_1 = 2|\cdots) = (1 - e_{1,1}) * e_{1,2}$, $P(a_2 = 5|a_1 = 2, \cdots) = (1 - e_{2,2}) * (1 - e_{2,3}) * (1 - e_{2,4}) * e_{2,5}$, and so on for other transition probabilities.