Extracting Part of Signal Representation from Direct RNA Squiggle for Modification Detection

Christopher Hendra [1 2 3 4] **Jonathan Göke** [4] **Alexandre Thiery** [3]

Abstract

Recent advances in the direct RNA sequencing technology has changed the landscape of RNA modifications study by facilitating computational detection of modifications using statistical or machine learning approaches. Most of these techniques require segmentation steps in order to align Nanopore squiggles to the candidate sites and they tend to produce large computational and storage overhead by indiscriminately segmenting the entire transcriptome. We introduce a segmentation-free approach to detecting RNA modification by directly extracting features from raw Nanopore signals that correspond to sites of interests. We further demonstrate the feasibility of our approach by achieving competitive performance in m6A detection against existing state-of-the-art methods.

1. Introduction

RNA modifications have been discovered since the 1950s(1; 2; 3) and have been found to play a prominent role in a wide range of biological processes(4; 5; 6). Several methods exist to detect these modifications, most prominently N^6-methyladenosine (m6A)(7; 8; 9; 10; 11; 12; 13; 14), pseudouridine (ψ) (15; 16; 17; 18), and N^5-methylcytosine (m5C) (19; 20; 21). These methods, while useful, require specific antibody or chemical reagents as well as experimental expertise that is beyond the reach of most computational labs. The recent development of direct RNA sequencing technology by Oxford Nanopore(22) allows the sequencing of native RNA molecules. This allows for computational detection of RNA modifications. Direct RNA sequencing technology infers nucleotide content of a given RNA molecule by making use of the electrical current generated as the molecule passes through the pores. The process of inferring the RNA sequence given the current information is called basecalling and this involves training methods based on Recurrent Neural Networks (RNN) or Convolutional Neural Networks (CNN)(23; 24; 25; 26) to predict a sequence of four canonical nucleotides (G, A, C, T). The presence of a modified nucleotide often results in a shift in the electrical current which can be exploited for RNA modification detection. Nevertheless, due to the noisy nature of Nanopore raw signal (squiggle), a segmentation step using either the softwares `Tombo`(27) or `Nanopolish`(28) is often required during the preprocessing step of many RNA modification detection algorithms(27; 29; 30; 31; 32; 33; 34; 35; 36; 37; 38) in order to match the raw signal features to candidate prediction sites. These algorithms typically segment the entire transcriptome, resulting in redundant storage space and wasted computational resources from unused segmented regions. Modifications such as m6A for example, mostly occur within 18 out of the 1024 possible 5-mer motifs(7; 8; 39) while other modifications such as m5C or pseudouridine only occur within segments containing the C or U nucleotides. To address some of these shortcomings, we propose a segmentation-free approach to localize features corresponding to modified positions. Instead of performing a segmentation pre-processing step over the entire transcriptome, we perform a targeted feature extraction by matching the similarity of each squiggle region to the target positions. We demonstrate that our approach performs comparably against other m6A classification methods. Furthermore, we achieve this result by making use of the signal features from the penultimate layer of a basecalling algorithm, which suggests a possible future integration between basecalling and modification detection.

2. Method

Our model takes in both sequence and raw signal to produce read-signal representation for modification detection (Figure 1). Firstly, we run a basecalling algorithm on each raw signal chunk to identify the squiggle segments containing the query positions. Next, we identify positions associated by each signal chunk by performing local alignment be-

[*] Equal contribution [1] NUS Graduate School's Integrative Sciences and Engineering Programme (ISEP) [2] Institute of Data Science (IDS), National University of Singapore [3] Department of Statistics and Applied Probability, National University of Singapore [4] Genomics Institute of Singapore, A*STAR. Correspondence to: Christopher Hendra <christopher.hendra@u.nus.edu>, Jonathan Göke <gokej@gis.a-star.edu.sg>, Alexandre Thiery <a.h.thiery@nus.edu.sg>.

The 2022 ICML Workshop on Computational Biology. Baltimore, Maryland, USA, 2022. Copyright 2022 by the author(s).

tween each basecalled sequence and its associated transcript using the Smith-Waterman algorithm(40) as implemented by the python library `parasail`(41). Inspired by recent work in database search(42), we apply scaled-dot product attention(43) to assign similarity score between the query positions and the high dimensional sequence output of the second last layer of our basecaller. The signal-sequence representation is then obtained by computing the weighted summation along the high dimensional sequence output of each signal chunk. Intuitively, the attention mechanism concentrates the representation on the squiggle elements that strongly match the query positions. The signal-sequence representation can then be used for modification detection via a Multiple Instance Learning (MIL) attention mechanism (44).

2.1. Basecalling

Recent version of basecallers are trained using sequence based neural networks such as CNN or RNN combined with a Connectionist Temporal Classification loss (CTC)(45). Under this approach, the network takes in raw current information and produces a sequence of probability vectors over each possible nucleotide. The probability of observing a particular nucleotide sequence is then computed by summing the probabilities of all possible paths through the sequence of probability simplex that will yield the sequence.

Let $\mathbf{x}_i \in \mathbb{R}^L$ be the i-th raw squiggle chunk of length L. We associate this short signal chunk with nucleotide sequence $s_i = \{G, A, C, T\}^{S_i}$ and transcript $z_i = (z_{i,1}, z_{i,2}, \dots, z_{i,n_{z_i}}) \in \mathbb{R}^{n_{z_i}}$. A perfect basecaller will predict nucleotide sequence s_i given the raw squiggle chunk $\mathbf{x}_i$. Furthermore, the nucleotide sequence s_i will then be locally aligned to the transcript z_i in the sense that $s_{i,j} = z_{i,k}$ for some $k \in \{1, \dots, n_{z_i}\}$ and for $j \in \{1, \dots, S_i\}$.

Let $f : \mathbb{R}^L \to \mathbb{R}^{D \times T}$ be function parameterized by a neural network that transforms the squiggle to a high dimensional representation vector of length T and dimension D. For this purpose, standard CNN and RNN neural architectures can be used. Here we associate a probability distribution for each of the T high dimensional representation over the four nucleotides G, A, C, T and a gap character ε. The gap character is removed during decoding step(46) and it allows the network to produce variable length output just as the raw signals can represent variable length nucleotide sequence. The probability distribution at each time step t can then be described as:

$$\{\pi_i^t\}_{1 \leq t \leq T} = (g \circ f)(\mathbf{x}_i) \tag{1}$$

Here $g : \mathbb{R}^{D \times T} \to (\Delta_5)^T$ is a neural network that outputs the conditional probability over the 5 symbols $\{G, A, C, T, \varepsilon\}$ given $f(\mathbf{x}_i)$. The notation Δ_5 refers to the probability simplex over 5 variables and $\pi_i^t = (\pi_{i,1}^t, \dots, \pi_{i,5}^t) \in \Delta_5$ for all $1 \leq t \leq T$.

The gap characters introduce many-to-one mapping between the the possible path π over the sequence of probability simplex $(\pi_{i,1}^t, \dots, \pi_{i,5}^t)$ and the true underlying sequence s_i. The path GεAεCεT and GACεT where the character ε is the gap character for example represent the same nucleotide sequence GACT. As such, the probability of observing the target sequence s_i is then computed as the sum of all possible sequences π with the gap character such that it is decoded as s_i.

$$p(s_i) = \sum_{\pi : B(\pi) = s_i} p(\pi \mathbf{x}_i) \tag{2}$$

where B is a decoding function that removes the gap characters from the predicted sequence. This sum can be computed using dynamic programming and during inference time, the predicted sequence $\hat{y}_i$ can be decoded using beam search or Viterbi algorithm. We train our own basecaller using the open source Bonito model (https://github.com/nanoporetech/bonito) with sequence ground truth obtained from Nanopolish eventalign(28).

2.2. Extraction of Positional Representation

After training a basecaller, we extract signal-sequence representation $h(\mathbf{x}_i, z_i, j)$ that capture the signal information from the read that matches the position j with respect to transcript $\mathbf{z}_i$. In order to do this, we extract features representation from the 10 flanking bases around the j-th position $[z_i]_{j,K}$ using a bidirectional LSTM R as $R\left([z_i]_{j,K}\right) \in \mathbb{R}^M$. Here we reason that the signal representation from f, the second last layer of our trained basecaller, can be informative since it is trained to maximize the probability of observing the underlying true sequence s_i. As such, we extract the signal-sequence representation with respect to f using the attention mechanism:

$$h(\mathbf{x}_i, z_i, j) = \left(f(\mathbf{x}_i) W_V\right)^T \text{Softmax}\left(\frac{f(\mathbf{x}_i) W_K R\left([z_i]_{j,K}\right)}{\sqrt{M}}\right) \tag{3}$$

for query $R\left([z_i]_{j,K}\right) \in \mathbb{R}^M$, key $f(\mathbf{x}_i) W_K \in \mathbb{R}^{T \times M}$ and value $f(\mathbf{x}_i) W_V \in \mathbb{R}^{T \times M}$. Here $W_K \in \mathbb{R}^{D \times M}$ and $W_V \in \mathbb{R}^{D \times M}$ are learnable parameters of the attention mechanism. Intuitively, the attention concentrates the representation of the signal value $f(\mathbf{x}_i) W_V$ on its sub-sequence that is the most similar to the query position $R\left([z_i]_{j,K}\right)$.

2.2.1. Detecting m6A modifications

Here we only update the parameters of the sequence network R while keeping the basecaller as a feature extractor. Previous work(36) has cast m6A detection problem as a Multiple Instance Learning (MIL) problem(47). Let

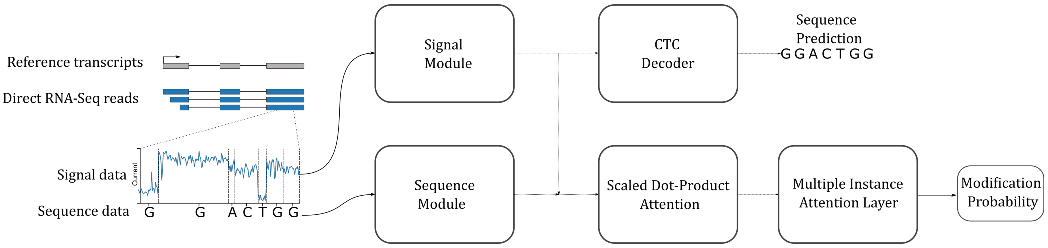

Figure 1. Overview of the model. The signal module comprises 3 CNN layers followed by 5 LSTM layers while the sequence module comprises 3 bidirectional LSTM layers followed by 1 linear layer. The scaled dot product attention combines the output the two modules to produce signal-sequence representation while the CTC decoder outputs sequence prediction from the signal representation. The MIL attention layer takes in the signal-sequence representation to output modification probability

$\{\mathbf{x}_{i,1}, \mathbf{x}_{i,2}, \ldots, \mathbf{x}_{i,N_i}\}$ be a collection of distinct signal chunks mapped to transcript z containing transcript position i. Each $\mathbf{x}_{i,j}$ can be associated with label $y_{i,j} \in \{0,1\}$ indicating whether each signal chunk contains m6A modification. However, we only have access to a single label $y_i \in \{0,1\}$ representing the modification status of the signal chunks collection instead of individual label $y_{i,j}$. Here we extract the signal-sequence representation $\{h(\mathbf{x}_{i,j}, z, i)\}_{1\leq j\leq N_i}$ for each signal chunk and combine their representation following the Attention-based Deep MIL framework(44)

$$h(\mathbf{x}_i, z, i) = \sum_{j=1}^{N_i} a_j h(\mathbf{x}_{i,j}, z, i) \tag{4}$$

where

$$a_j = \frac{\exp\left\{U^T \tanh\left(V^T h(\mathbf{x}_{i,j}, z, i)\right)\right\}}{\sum_{k=1}^{N_i} \exp\left\{U^T \tanh\left(V^T h(\mathbf{x}_{i,j}, z, i)\right)\right\}} \tag{5}$$

Here $U \in R^{H\times 1}$ and $V \in \mathbb{R}^{D\times H}$ are learnable parameters and a_j measures the relative contribution of raw signal chunk j in the collective representation $h(\mathbf{x}_i, z, i)$. The representation vector $h(\mathbf{x}_i, z, i)$ can then be used as a feature vector within a standard logistic regression classifier. The complete model is trained end-to-end by minimizing the cross-entropy loss with stochastic gradient descent.

3. Experiments

3.1. Dataset

HCT116 direct RNA Sequencing Data

The HCT116 cell line direct RNA sequencing data provided by the SG-NEX project(48) and split the dataset on the gene level into train, validation, and test sets. We use the training set for basecalling and m6A detection training and use the validation set for model selection. The m6A labels were generated using the m6ACe-seq protocol(13) and we follow the training procedure in (36) that restricts training sites to those harbouring DRACH motifs.

HEK293T direct RNA Sequencing Data

The HEK293T cell line direct RNA sequencing data is provided by (32). The m6A labels for this dataset is generated by m6ACE-seq(13) and miCLIP(12). We use this dataset to validate both basecalling performance as well as m6A classification performance. Similar to the HCT116 dataset, we restrict our prediction to the DRACH sites.

3.2. Basecalling and Mapping Accuracy

Dataset	Mean Acc	Median Acc	IoU
HCT116	91.2%	93.6%	88.1%
HEK293T	90.1%	93.5%	86.0

Table 1. Basecalling Accuracy and Intersection over Union on HCT116 and HEK293T datasets

We first evaluate the accuracy of our basecaller and whether it can identify positions within raw signal chunks correctly by aligning the predicted sequence to its underlying reference label using Smith-Waterman algorithm(40). We measure the mapped accuracy as:

$$\text{Accuracy} = \frac{\text{Number of Matched Bases}}{\text{Number of Reference Bases}} \tag{6}$$

In order to extract accurate feature representation, we need to identify whether the position we wish to model exist within a given read chunk. To do this, we align each the predicted sequence $\hat{y}_i$ to its reference transcript z_i and obtain a set of predicted positions $\hat{L}_i$ spanned by read i. The alignment step serves to correct small error in the prediction and so we do not necessarily need a very high basecalling accuracy to correctly identify the positions spanned by a given signal chunk. As such, we also measure the Intersection over Union (IoU) of the predicted positions $\hat{L}_i$ against the ground-truth transcript positions L_i represented in read i. This is given by:

5-mer Motifs	Model	ROC AUC (HCT116)	PR AUC (HCT116)	ROC AUC (HEK293T)	PR AUC (HEK293T)
18 motifs	m6ARaw (ours)	**0.930**	0.385	0.817	0.319
	m6Anet	0.926	**0.451**	**0.838**	**0.366**
	Tombo	0.707	0.121	0.507	0.0857
	EpiNano 1	0.776	0.206	0.710	0.240
	EpiNano 2	0.788	0.133	0.725	0.182
	EpiNano 3	0.781	0.176	0.722	0.213
	EpiNano 4	0.764	0.235	0.704	0.227
	EpiNano 5	0.736	0.167	0.670	0.170
12 motifs	m6ARaw (ours)	**0.927**	**0.609**	0.812	0.332
	m6Anet	0.916	0.565	**0.837**	**0.373**
	Tombo	0.759	0.296	0.504	0.099
	nanom6a	0.787	0.364	0.719	0.203
4 motifs	m6ARaw (ours)	**0.917**	0.497	0.797	0.390
	m6Anet	0.908	**0.543**	**0.825**	**0.440**
	Tombo	0.767	0.280	0.515	0.166
	MINES	0.792	0.340	0.708	0.326

Table 2. Performance comparison of m6ARaw against existing m6A detection methods

$$\text{IoU} = \frac{\hat{L}_i \cap L_i}{\hat{L}_i \cup L_i} \tag{7}$$

On the HCT116 cell line, we manage to achieve a mean accuracy of 91.2% and median accuracy of 93.6% while on the HEK293T cell line, it achieves a mean accuracy of 90.1% and median accuracy of 93.5%. Additionally, the model achieves an average IoU of 88.1% on the HCT116 cell line and 86.0% on the HEK293T cell line. This indicates that the model can recognize the underlying sequence of each signal chunk and its mapped alignment coincides strongly with the ground truth label. Another way to improve the mapping quality will be to consider the outputs from two adjacent overlapping signal chunks, a strategy implemented by the original Bonito basecaller(46), which we leave for future work.

3.3. m6A Modification Detection

Here we demonstrate that our approach can yield high quality signal-sequence representation for RNA modifications by by training the model to perform m6A detection based on the extracted signal-sequence representation. We call our approach here m6ARaw and compare the ROC AUC and PR AUC on several partition of the HCT116 against several existing methods(27; 30; 33; 49; 36) for m6A detection as detailed in (36). As we can see, on the HCT116 test set, our methods perform comparably (ROC AUC: 0.930, PR AUC: 0.385, ROC AUC: 0.927, PR AUC:0.609, ROC AUC:0.917, PR AUC:0.497) against m6Anet (ROC AUC: 0.926, PR AUC:0.451, ROC AUC: 0.916, PR AUC:0.565, ROC AUC:0.908, PR AUC:0.543) while outperforming all other methods. We observe similar results in the HEK293T cell line where our model outperform existing methods (ROC AUC: 0.817, PR AUC: 0.319, ROC AUC: 0.812, PR AUC:0.332, ROC AUC:0.797, PR AUC:0.390) and perform comparably against m6Anet (ROC AUC: 0.838, PR AUC: 0.366, ROC AUC: 0.837, PR AUC:0.373, ROC AUC:0.825, PR AUC:0.440). The results suggest that our approach can produce informative signal-sequence representation to detect m6A modifications and is competitive against existing segmentation-based approaches.

4. Discussion

In recent years, we have seen an increasing number of computational tools being developed to detect RNA modifications from direct RNA sequencing data. These tools have facilitated a growing number of studies into RNA modifications but at the same time require a lot of compute resources. Our study explores the possibility of streamlining such processes by avoiding segmentation steps and integrating modification detection to basecalling, thereby reducing the computational burden as well as storage space requirements from running such analysis. We demonstrate that this approach can produce informative representation to detect m6A modifications, achieving competitive performance against state-of-the-art method in m6A detection. In the future we are planning to extend this approach to other RNA modifications and we hope that this work can lay the foundation for further study into representation learning in the context of detecting RNA modifications directly from raw signals.

References

[1] W. E. Cohn and E. Volkin, "Nucleoside-5-Phosphates from ribonucleic acid," 1951.

[2] J. W. Kemp and F. W. Allen, "Ribonucleic acids from pancreas which contain new components," 1958.

[3] F. F. Davis and F. W. Allen, "Ribonucleic acids from yeast which contain a fifth nucleotide," *Journal of Biological Chemistry*, vol. 227, no. 2, p. 907–915, 1957.

[4] K. Xu, Y. Yang, G.-H. Feng, B.-F. Sun, J.-Q. Chen, Y.-F. Li, Y.-S. Chen, X.-X. Zhang, C.-X. Wang, L.-Y. Jiang, C. Liu, Z.-Y. Zhang, X.-J. Wang, Q. Zhou, Y.-G. Yang, and W. Li, "Mettl3-mediated m6a regulates spermatogonial differentiation and meiosis initiation," 2017.

[5] E. Yankova, W. Blackaby, M. Albertella, J. Rak, E. De Braekeleer, G. Tsagkogeorga, E. S. Pilka, D. Aspris, D. Leggate, A. G. Hendrick, N. A. Webster, B. Andrews, R. Fosbeary, P. Guest, N. Irigoyen, M. Eleftheriou, M. Gozdecka, J. M. L. Dias, A. J. Bannister, B. Vick, I. Jeremias, G. S. Vassiliou, O. Rausch, K. Tzelepis, and T. Kouzarides, "Small-molecule inhibition of METTL3 as a strategy against myeloid leukaemia," *Nature*, Apr. 2021.

[6] P. Nombela, B. Miguel-López, and S. Blanco, "The role of m6a, m5c and ψ rna modifications in cancer: Novel therapeutic opportunities," *Molecular Cancer*, vol. 20, no. 1, pp. 1–30, 2021.

[7] K. D. Meyer, Y. Saletore, P. Zumbo, O. Elemento, C. E. Mason, and S. R. Jaffrey, "Comprehensive analysis of mRNA methylation reveals enrichment in 3 UTRs and near stop codons," 2012.

[8] D. Dominissini, S. Moshitch-Moshkovitz, S. Schwartz, M. Salmon-Divon, L. Ungar, S. Osenberg, K. Cesarkas, J. Jacob-Hirsch, N. Amariglio, M. Kupiec, R. Sorek, and G. Rechavi, "Topology of the human and mouse m6a RNA methylomes revealed by m6a-seq," *Nature*, vol. 485, pp. 201–206, Apr. 2012.

[9] K. Chen, Z. Lu, X. Wang, Y. Fu, G.-Z. Luo, N. Liu, D. Han, D. Dominissini, Q. Dai, T. Pan, and C. He, "High-resolution n(6) -methyladenosine (m(6) a) map using photo-crosslinking-assisted m(6) a sequencing," *Angew. Chem. Int. Ed Engl.*, vol. 54, pp. 1587–1590, Jan. 2015.

[10] S. Ke, E. A. Alemu, C. Mertens, E. C. Gantman, J. J. Fak, A. Mele, B. Haripal, I. Zucker-Scharff, M. J. Moore, C. Y. Park, C. B. Vågbø, A. Kusśnierczyk, A. Klungland, J. E. Darnell, Jr, and R. B. Darnell, "A majority of m6a residues are in the last exons, allowing the potential for 3' UTR regulation," *Genes Dev.*, vol. 29, pp. 2037–2053, Oct. 2015.

[11] B. Molinie, J. Wang, K. S. Lim, R. Hillebrand, Z.-X. Lu, N. Van Wittenberghe, B. D. Howard, K. Daneshvar, A. C. Mullen, P. Dedon, Y. Xing, and C. C. Giallourakis, "m6A-LAIC-seq reveals the census and complexity of the m6a epitranscriptome," 2016.

[12] B. Linder, A. V. Grozhik, A. O. Olarerin-George, C. Meydan, C. E. Mason, and S. R. Jaffrey, "Single-nucleotide-resolution mapping of m6a and m6am throughout the transcriptome," *Nat. Methods*, vol. 12, pp. 767–772, Aug. 2015.

[13] C. W. Q. Koh, Y. T. Goh, and W. S. Sho Goh, "Atlas of quantitative single-base-resolution n6-methyl-adenine methylomes," 2019.

[14] D. Dierks, M. A. Garcia-Campos, A. Uzonyi, M. Safra, S. Edelheit, A. Rossi, T. Sideri, R. A. Varier, A. Brandis, Y. Stelzer, F. van Werven, R. Scherz-Shouval, and S. Schwartz, "Multiplexed profiling facilitates robust m6a quantification at site, gene and sample resolution," *Nat. Methods*, Sept. 2021.

[15] S. Schwartz, D. A. Bernstein, M. R. Mumbach, M. Jovanovic, R. H. Herbst, B. X. León-Ricardo, J. M. Engreitz, M. Guttman, R. Satija, E. S. Lander, *et al.*, "Transcriptome-wide mapping reveals widespread dynamic-regulated pseudouridylation of ncrna and mrna," *Cell*, vol. 159, no. 1, pp. 148–162, 2014.

[16] A. F. Lovejoy, D. P. Riordan, and P. O. Brown, "Transcriptome-wide mapping of pseudouridines: pseudouridine synthases modify specific mrnas in s. cerevisiae," *PloS one*, vol. 9, no. 10, p. e110799, 2014.

[17] T. M. Carlile, M. F. Rojas-Duran, B. Zinshteyn, H. Shin, K. M. Bartoli, and W. V. Gilbert, "Pseudouridine profiling reveals regulated mrna pseudouridylation in yeast and human cells," *Nature*, vol. 515, no. 7525, p. 143–146, 2014.

[18] N. Liu, Q. Dai, G. Zheng, C. He, M. Parisien, and T. Pan, "N6-methyladenosine-dependent RNA structural switches regulate RNA–protein interactions," 2015.

[19] J. E. Squires, H. R. Patel, M. Nousch, T. Sibbritt, D. T. Humphreys, B. J. Parker, C. M. Suter, and T. Preiss, "Widespread occurrence of 5-methylcytosine in human coding and non-coding rna," *Nucleic acids research*, vol. 40, no. 11, pp. 5023–5033, 2012.

[20] S. Hussain, J. Aleksic, S. Blanco, S. Dietmann, and M. Frye, “Characterizing 5-methylcytosine in the mammalian epitranscriptome,” *Genome biology*, vol. 14, no. 11, pp. 1–10, 2013.

[21] T. Huang, W. Chen, J. Liu, N. Gu, and R. Zhang, “Genome-wide identification of mrna 5-methylcytosine in mammals,” *Nature Structural & Molecular Biology*, vol. 26, no. 5, pp. 380–388, 2019.

[22] D. R. Garalde, E. A. Snell, D. Jachimowicz, B. Sipos, J. H. Lloyd, M. Bruce, N. Pantic, T. Admassu, P. James, A. Warland, M. Jordan, J. Ciccone, S. Serra, J. Keenan, S. Martin, L. McNeill, E. J. Wallace, L. Jayasinghe, C. Wright, J. Blasco, S. Young, D. Brocklebank, S. Juul, J. Clarke, A. J. Heron, and D. J. Turner, “Highly parallel direct RNA sequencing on an array of nanopores,” *Nat. Methods*, vol. 15, pp. 201–206, Mar. 2018.

[23] V. Boža, B. Brejová, and T. Vinař, “Deepnano: deep recurrent neural networks for base calling in minion nanopore reads,” *PloS one*, vol. 12, no. 6, p. e0178751, 2017.

[24] M. Stoiber and J. Brown, “Basecrawller: streaming nanopore basecalling directly from raw signal,” *BioRxiv*, p. 133058, 2017.

[25] H. Teng, M. D. Cao, M. B. Hall, T. Duarte, S. Wang, and L. J. Coin, “Chiron: translating nanopore raw signal directly into nucleotide sequence using deep learning,” *GigaScience*, vol. 7, no. 5, p. giy037, 2018.

[26] J. Zeng, H. Cai, H. Peng, H. Wang, Y. Zhang, and T. Akutsu, “Causalcall: nanopore basecalling using a temporal convolutional network,” *Frontiers in Genetics*, p. 1332, 2020.

[27] M. Stoiber, J. Quick, R. Egan, J. E. Lee, S. Celniker, R. K. Neely, N. Loman, L. A. Pennacchio, and J. Brown, “De novo identification of DNA modifications enabled by Genome-Guided nanopore signal processing.”

[28] N. J. Loman, J. Quick, and J. T. Simpson, “A complete bacterial genome assembled de novo using only nanopore sequencing data,” *Nat. Methods*, vol. 12, pp. 733–735, Aug. 2015.

[29] A. Leger, P. P. Amaral, L. Pandolfini, C. Capitanchik, F. Capraro, I. Barbieri, V. Migliori, N. M. Luscombe, A. J. Enright, K. Tzelepis, J. Ule, T. Fitzgerald, E. Birney, T. Leonardi, and T. Kouzarides, “RNA modifications detection by comparative nanopore direct RNA sequencing.” Nov. 2019.

[30] D. A. Lorenz, S. Sathe, J. M. Einstein, and G. W. Yeo, “Direct RNA sequencing enables ma detection in endogenous transcript isoforms at base-specific resolution,” *RNA*, vol. 26, pp. 19–28, Jan. 2020.

[31] H. Ueda, “nanodoc: RNA modification detection using nanopore raw reads with deep One-Class classification.”

[32] P. N. Pratanwanich, F. Yao, Y. Chen, C. W. Q. Koh, Y. K. Wan, C. Hendra, P. Poon, Y. T. Goh, P. M. L. Yap, J. Y. Chooi, W. J. Chng, S. B. Ng, A. Thiery, W. S. S. Goh, and J. Göke, “Identification of differential RNA modifications from nanopore direct RNA sequencing with xpore,” *Nat. Biotechnol.*, July 2021.

[33] Y. Gao, X. Liu, B. Wu, H. Wang, F. Xi, M. V. Kohnen, A. S. N. Reddy, and L. Gu, “Quantitative profiling of n-methyladenosine at single-base resolution in stem-differentiating xylem of populus trichocarpa using nanopore direct RNA sequencing,” *Genome Biol.*, vol. 22, p. 22, Jan. 2021.

[34] O. Begik, M. C. Lucas, L. P. Pryszcz, J. M. Ramirez, R. Medina, I. Milenkovic, S. Cruciani, H. Liu, H. G. S. Vieira, A. Sas-Chen, J. S. Mattick, S. Schwartz, and E. M. Novoa, “Quantitative profiling of pseudouridylation dynamics in native RNAs with nanopore sequencing,” *Nat. Biotechnol.*, vol. 39, pp. 1278–1291, Oct. 2021.

[35] M. T. Parker, G. J. Barton, and G. G. Simpson, “Yanocomp: robust prediction of m6a modifications in individual nanopore direct rna reads,” *bioRxiv*, 2021.

[36] C. Hendra, P. N. Pratanwanich, Y. K. Wan, W. S. Goh, A. Thiery, and J. Goeke, “Detection of m6a from direct rna sequencing using a multiple instance learning framework,” *bioRxiv*, 2021.

[37] W. Stephenson, R. Razaghi, S. Busan, K. M. Weeks, W. Timp, and P. Smibert, “Direct detection of rna modifications and structure using single-molecule nanopore sequencing,” *Cell genomics*, vol. 2, no. 2, p. 100097, 2022.

[38] A. Sethi, M. Guarnacci, A. Ravindran, A. Srivastava, J. Xu, K. Woodward, W. Hamilton, J. Gao, L. Starrs, R. Hayashi, *et al.*, “Identification of m6a and m5c rna modifications at single-molecule resolution from nanopore sequencing,” 2022.

[39] S. Schwartz, M. R. Mumbach, M. Jovanovic, T. Wang, K. Maciag, G. G. Bushkin, P. Mertins, D. Ter-Ovanesyan, N. Habib, D. Cacchiarelli, *et al.*, “Perturbation of m6a writers reveals two distinct classes of mrna methylation at internal and 5 sites,” *Cell reports*, vol. 8, no. 1, pp. 284–296, 2014.

[40] T. F. Smith, M. S. Waterman, *et al.*, "Identification of common molecular subsequences," *Journal of molecular biology*, vol. 147, no. 1, pp. 195–197, 1981.

[41] J. Daily, "Parasail: Simd c library for global, semi-global, and local pairwise sequence alignments," *BMC bioinformatics*, vol. 17, no. 1, pp. 1–11, 2016.

[42] Y. Tay, V. Q. Tran, M. Dehghani, J. Ni, D. Bahri, H. Mehta, Z. Qin, K. Hui, Z. Zhao, J. Gupta, *et al.*, "Transformer memory as a differentiable search index," *arXiv preprint arXiv:2202.06991*, 2022.

[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[44] M. Ilse, J. M. Tomczak, and M. Welling, "Attention-based deep multiple instance learning," Feb. 2018.

[45] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, pp. 369–376, 2006.

[46] J. Silvestre-Ryan and I. Holmes, "Pair consensus decoding improves accuracy of neural network basecallers for nanopore sequencing," *Genome biology*, vol. 22, no. 1, pp. 1–6, 2021.

[47] O. Maron and T. Lozano-Pérez, "A framework for Multiple-Instance learning," in *Advances in Neural Information Processing Systems 10* (M. I. Jordan, M. J. Kearns, and S. A. Solla, eds.), pp. 570–576, MIT Press, 1998.

[48] Y. Chen, N. M. Davidson, Y. K. Wan, H. Patel, F. Yao, H. M. Low, C. Hendra, L. Watten, A. Sim, C. Sawyer, V. Iakovleva, P. L. Lee, L. Xin, H. E. V. Ng, J. M. Loo, X. Ong, H. Q. A. Ng, J. Wang, W. Q. C. Koh, S. Y. P. Poon, D. Stanojevic, H.-D. Tran, K. H. E. Lim, S. Y. Toh, P. A. Ewels, H.-H. Ng, N. Gopalakrishna Iyer, A. Thiery, W. J. Chng, L. Chen, R. DasGupta, M. Sikic, Y.-S. Chan, B. O. P. Tan, Y. Wan, W. L. Tam, Q. Yu, C. C. Khor, T. Wüstefeld, P. N. Pratanwanich, M. I. Love, W. S. S. Goh, S. B. Ng, A. Oshlack, J. Göke, and SG-NEx consortium, "A systematic benchmark of nanopore long read RNA sequencing for transcript level analysis in human cell lines." Apr. 2021.

[49] H. Liu, O. Begik, and E. M. Novoa, "EpiNano: Detection of ma RNA modifications using oxford nanopore direct RNA sequencing," *Methods Mol. Biol.*, vol. 2298, pp. 31–52, 2021.