nanoDoc: RNA modification detection using Nanopore raw reads with Deep One-Class Classification

Hiroki Ueda[1*]

[1]Division of Biological Data Science, Research Center for Advanced Science and Technology, University of Tokyo, Tokyo, Japan

[*]Corresponding author
4-6-1 Komaba, Meguro, Tokyo 153-8904, Japan
E-mail: ueda@biods.rcast.u-tokyo.ac.jp

Short title: Detection of RNA modifications using nanoDOC

Abstract

Advances in Nanopore single-molecule direct RNA sequencing (DRS) have presented the possibility of detecting comprehensive post-transcriptional modifications (PTMs) as an alternative to experimental approaches combined with high-throughput sequencing. It has been shown that the DRS method can detect the change in the raw electric current signal of a PTM; however, the accuracy and reliability still require improvement.

Here, we presented a new software, called nanoDoc, for detecting PTMs from DRS data using a deep neural network. Current signal deviations caused by PTMs are analyzed via Deep One-Class Classification with a convolutional neural network. Using a ribosomal RNA dataset, the software archive displayed an area under the curve (AUC) accuracy of 0.96 for the detection of 23 different kinds of modifications in *Escherichia coli* and *Saccharomyces cerevisiae*. We also demonstrated a tentative classification of PTMs using unsupervised clustering. Finally, we applied this software to severe acute respiratory syndrome coronavirus 2 data and identified commonly modified sites among three groups. nanoDoc is open source (GPLv3) and available at https://github.com/uedaLabR/nanoDoc

Introduction

To date, over 160 different types of post-transcriptional modifications (PTMs) have been identified. PTMs have long been known, in tRNA and rRNA, to play an important role in maintaining structure and adjusting codon usage. More recently, an increasing number of PTMs have been identified on other types of RNA (e.g. mRNA and non-coding RNA), and they regulate multiple aspects of RNA function, including alternative splicing, export, stability, and translation. PTMs play a role in the life cycle of various viruses and in the cellular response to a viral infection [1]. One such PTM is N^6-methyl adenine (m6A), introduced by the adenine methyltransferase family of enzymes (writers), which is specifically recognized by a set of RNA-binding proteins (readers). In addition, It has discovered that fat mass and obesity-associated protein (FTO) exhibits m6A demethylase (eraser) activity [2]; hence, PTMs are dynamic. The concept that the activity balance between the writers, readers, and eraser modulates cellular functions has been the basis for the rapidly developing field of “epitranscriptomics”. Therefore, transcriptome-wide PTM identification is the

key technology for epitranscriptomics.

For the first generation of epitranscriptomics tools, experimental methods, including detection of reverse transcription error pattern (RT-signature) with/without a specific chemical [3][4][5][6][7] and immunoprecipitation using antibodies specific to certain PTMs, [8][9] have been developed. Although these experimental methods have been adapted for the different kinds of PTMs, they also have limitations. For example, with respect to RT-signature detection, chemicalconditions are available for only a few types of modifications, and it is difficult to detect all modification sites if they are located in close proximity. Likewise, immunoprecipitation tends to have problems of resolution and antibody specificity. In addition, these experimental methods can only detect one type of modification; since different types of modifications are dynamically acting in cells, a comprehensive method to detect multiple PTMs, with a simple setup, is necessary for further progress in epitranscriptomics.

Nanopore single-molecule direct RNA sequencing (DRS) has been shown to be able to detect PTMs in addition to nucleotide base calling [10][11]. DRS has the advantage of detecting the RNA molecule directly, in contrast to short read sequencing, in which information of RNA modification is lost via the process of reverse transcription and polymerase chain reaction. For the detection of DNA modifications, recurrent neural network (RNN) models, such as DeepSignal [12] or DeepMod [13], have been developed. However, there are numerous types of RNA modifications (e.g. at least 19 types on mRNA), making the application of RNN models difficult, in terms of a deep neural network (DNN) model building and preparing the training data set. One way to acquire the training set for RNA modification is via the chemical synthesis of modified bases [14]. However, chemical synthesis is only available for selected PTMs, and it is difficult to generate all combinations of the modified and unmodified states of the nucleotide sequence. Therefore, the majority of current PTM detection algorithms rely on base-calling errors induced by modifications and external information, such as modification motifs or previously verified modification sites [15] [14]. However, these approaches rely on the external errors of unknown processing, and there is always a chance that any signal could be base-called as a same base to the reference. Furthermore, using a priori data for filtering is problematic when the data source contains errors, and any filtering method could create a bias that suppresses de novo discovery.

In another type of algorithm, the raw current signal is aligned against a theoretical signal by either the Hidden Markov Model (HMM) [16] [17] [18] or RNN [19], and the difference in raw signal levels is

compared. Since the current signal is disturbed by the existence of PTMs, statistical testing such as the Kolmogorov– Smirnov (KS) test [18][20], is used to detect PTMs. Identifying modification sites with these methods is useful and may lead to further classification of different PTMs at detection sites; however, the accuracy of these methods requires improvements.

In this study, we knew that detecting PTMs from non-modified *in vitro* signaling could be solved by One-Class Classification, which is widely used in machine learning. For example, Perera et al. [21] developed a state-of-art One-Class Classification algorithm, Deep One-Class (DOC), with a convolutional neural network (CNN). Recently, CNN has also been used in the analysis of nanopore data; for example, DeepBinner [22] can detect barcode sequences more accurately than conventional base caller and mapping reads approaches. Thus, using the DOC and CNN algorithms, we present a novel software to detect PTMs.

Design and implementation

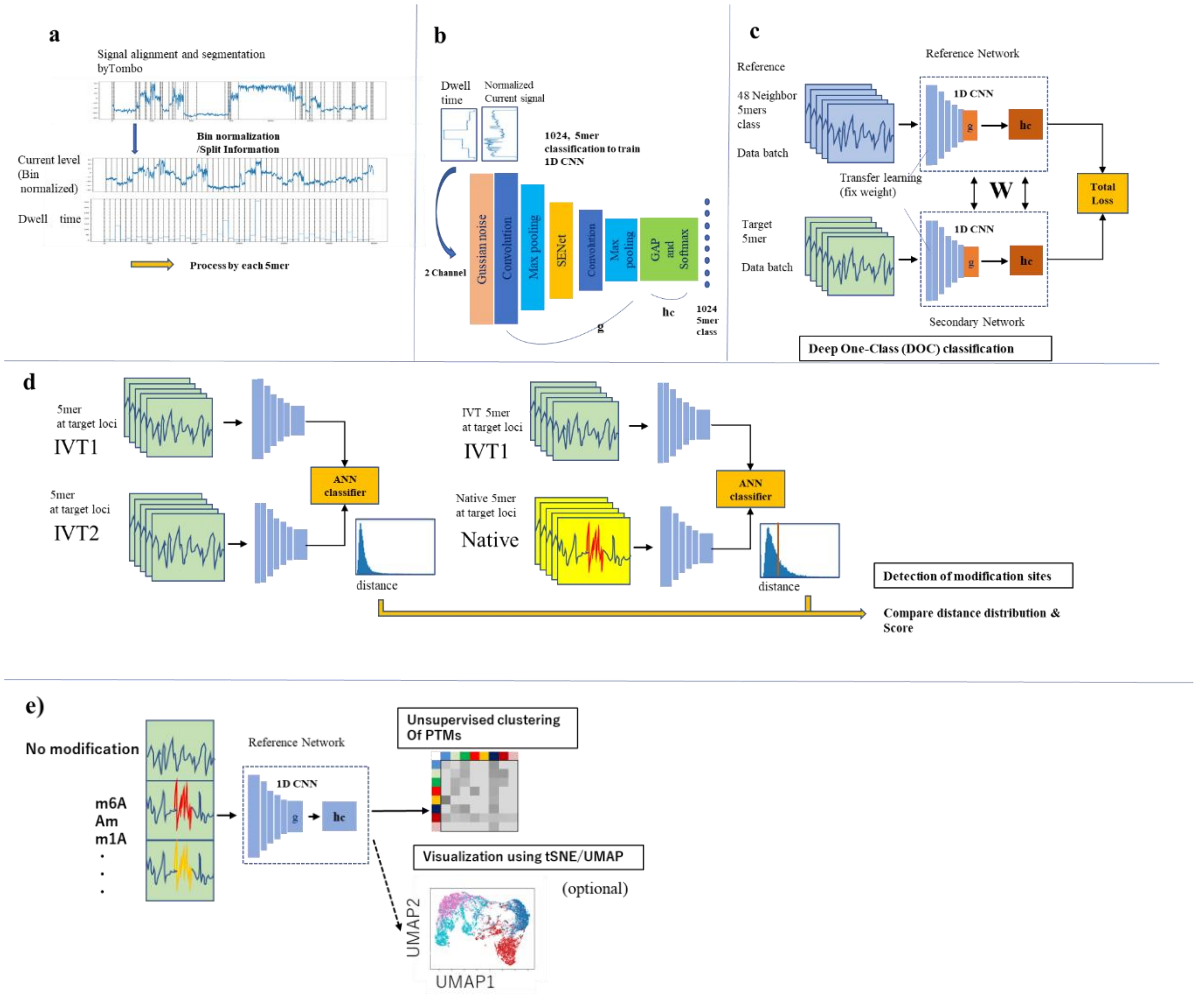


Figure 1 – The modification detection algorithm by nanoDoc

a) Decomposition of resquiggled signal to uniform length of 5-mer. The resquiggled signal was decomposed into two channels: one for current level and the other for dwell time interval. The current was either up- or down-sampled to fit in the uniform length interval.

b)) 1D-CNN was trained with 1024 labeled 5-mer combinations.

c) Deep One-Class (DOC) training for each 5-mer. (modified from *Learning Deep Features for One-Class Classification[21]*) Reference 5-mer and 5-mer with similar sequences were subjected to DOC classification.

d) Scoring with Approximate Nearest Neighbor Euclidian distance. Score is assigned based on Euclidian distance to reference dataset.

e) Unsupervised clustering of multiple PTMs. The feature 5mer data is subject to k-mean unsupervised clustering

nanoDoc

Preparation of data and Signal segmentation

Nanopore Multi-fast5/fast5 files containing the raw electric signal were base-called with Guppy (v3.5.2; Oxford Nanopore Technologies [ONT], Oxford, UK), with the configuration file "res_rna2_r941_min_flipflop_v001.cfg" downloaded from ONT rerio project [23]. Then, using ont_fast5_apt [24], multi-fast5 reads were converted to single fast5. The Tombo (v1.5) [18], resquiggle command was used to map reads to their respective references (Supplementary Table 2) with the inner mappy/minimap2 aligner [25], and raw signals were further assigned to the mapped region of the reference (Fig. 1a). The assigned raw signal for each genomic position was either up- or down-sampled to fit into equal bin size of 60 (Supplementary Fig 1). The original signal length of each interval was recorded in other fields for use in a latter process. The apache parquet [26] format was used to record the data.

Neural Network Architecture and initial training

From the *in vitro* (IVT) sequence of previous studies [27][14], the 5-mer signal was extracted for all 1024 possible 5-mer combinations for initial CNN training (Fig.1b). The CNN was constructed using Keras [28] with Tensorflow [29] backend. A one-dimensional CNN (1D-CNN) was constructed to capture the specific features of nanopore signals. The CNN architecture consisted of 64 layers including convolutional, batch normalization, and drop out layers reinforced by Squeezenet [30] architecture. (Supplementary Fig.2). nanoDoc took two channels of data as inputs: the first channel was the normalized current signal of 5-mer, and the second channel was dwell time information formatted to the same bin size. Class classification of the 1024 5-mer combinations was performed, using 1,000 signals for training and 200 signals for validation (Fig. 1b). Four NVIDIA Tesla v100 graphics processing units were used, and training was conducted up to 500 epochs for 454 mins. The epoch of 340 yielded the best validation accuracy of 0.32 (Supplementary Fig 3a).

Deep One-Class learning

DOC classification was performed using trained CNN. First, the weight with best accuracy (the epoch of

340) from the previous procedure was loaded into the CNN model. The top three layers were removed, and the convolutional layers (g) were connected in a parallel fashion with an additional sub classification network layer (hc), where the weight was shared between two CNNs (Fig. 1c). Each CNN was subject to transfer learning, where the weight is fixed, except for the top six layers (Fig. 1c).

Then, DOC Classification was performed for each of the 1024 5-mer combinations. The 5-mer raw data was utilized as input for the secondary CNN. Reference data was selected from 48 other 5-mers that had nucleotide sequences similar to the target (Supplementary Fig. 4; the loss function and how to choose neighbor sequences). The networks were trained for 2 epochs, as indicated in the DOC paper [21], to prevent overfitting.

Inference and distance search

At each genomic locus, IVT sequences, twice as many as the Native sequences, were selected for detection, to create an equal number of sequences in three raw data batches: IVT1, IVT2, and Native. A maximum of 20,000 signals were utilized for each data batch. For each 5-mer at the genomic locus, the DOC-trained weight was loaded to the CNN, as well as three data batches, as the program progressed along the reference. In the output layer, the 620 × 2 input dimension was reduced to 16 dimensions. Euclidian distances of the output vectors, from IVT1 to IVT2 and IVT1 to Native, were compared using approximate nearest neighbors (ANN)[31] with Faiss [32]. For each entry vector, five of the nearest vectors in the target dataset were selected, and the mean distance was summarized as the distribution (Fig. 1d).

Adjustment and scoring

The Euclidian distance from the Native signal to IVT signal should be close or equal to that from IVT to IVT, if there are no modifications in Native; however, the value increases with the existence of modifications. Here, we defined cdf_INV as the cumulative distribution function of the distance from IVT1 to IVT2 and cdf_Native as that from IVT1 to Native. Across the range of distances, cdf_Native should be smaller than cdf_INV, under the existence of modified bases. For scoring, we calculated the area between two accumulated functions over the percentile threshold.

TB = Total number of Data

x_thres = threshold x, where x is distance; 20 percentiles were calculated for each distribution of IVT 5-mers, and the threshold was set to that value.

Score = (Σ cdf_INV(x) - cdf_Native(x))/TB, where x > x_thres.

To avoid noise aggregation in the low depth area, we took statistical measures of the scores for different depths and used regression to adjust the variance in scoring.

The final score was normalized, from 0 to 1, as follows:

If score > 500, score = 1; otherwise, score = score/500.

(Supplementary Fig. 5 for scoring schema and Supplementary Method 1)

Clustering and extracting the feature.

The 16-dimensional feature output of the Native sequence was subjected to k-mean clustering and dimensional reduction uniform manifold approximation and projection (UMAP) [33].

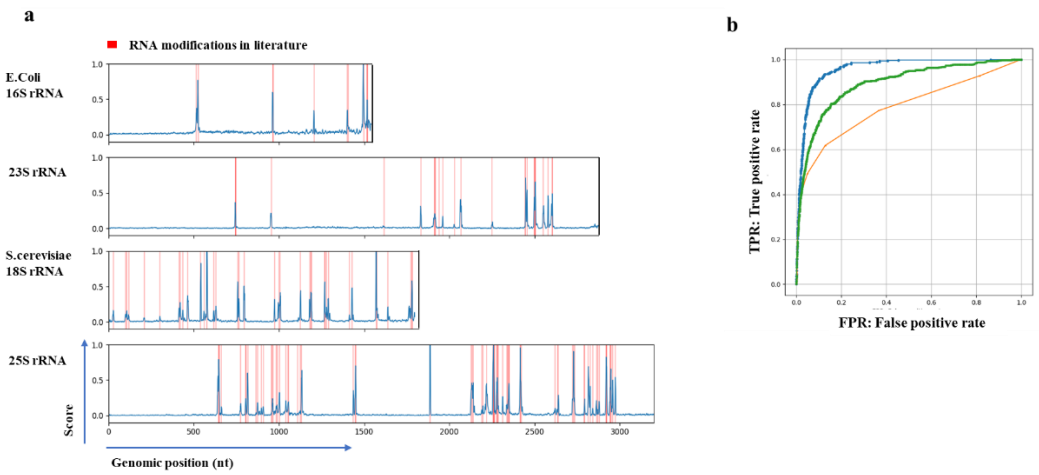
Results

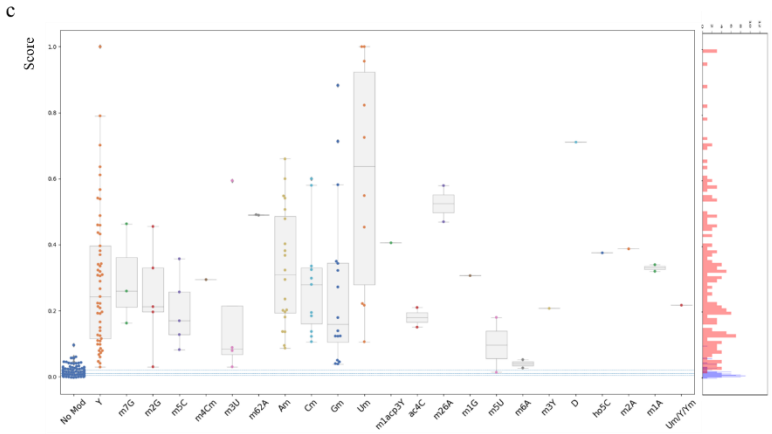
Detection of known PTMs in rRNA and evaluation

Previously, nanoDoc has been applied to the native rRNA sequencing data of *Escherichia coli* and *Saccharomyces cerevisiae* from a previous study[20]. The data was base-called and processed by Tombo, as described in the previous section, using the appropriate reference sequence (Supplementary Table 2). In addition, the list of known modification sites and KSStats values were downloaded from the GitHub repository of nanoShape [20]. The score distributions across the genomic position of each rRNA are depicted in Fig. 2a, where the red line represents the positions of known PTMs and the blue line shows the scores. To evaluate the accuracy, we conducted receiver operating characteristic (ROC) analysis with nanoDoc, KSStats, and mean current differences from Tombo. The results are shown in Fig. 2b, with the ROC curves for nanoDoc (blue), KSStats (Stephenson et al; green), and Tombo mean current difference (orange). The area under the curve (AUC) values were 0.96, 0.89, and 0.79, respectively (Supplementary Method 2). We examined the nanoDoc score for each of the 23 modification types (Fig. 2c). All scores at the 147

modification sites were higher than the 75th percentile baseline, except for one m5u modification site. The baseline was drawn from the collection of no modification sites. Note that the actual modification frequency at each known modification site was unknown, and a low score might have represented the low modification frequency at a given position. Wybutosine (yW) and Nm (2′-O-methylation) recorded the highest scores (0.28±0.23 out of 53 sites and 0.38±0.25 out of 57, respectively) among the PTMs, which implies that heavily modified bases and modifications on ribose are prone to detection. Although sample sizes for some other modification sites were not enough for general evaluation, the overall result suggested that majority of base modifications could be detected using this method.

For m6A modification, the score level was low (0.02 ± 0.0087) in the rRNA dataset. To investigate the m6A detection more precisely, we analyzed the synthesized RNA data from a previous study [14]. We examined 81 types of 5-mers with m6A at the center, "BB (m6A) BB" (where B = G, C, or U), against the reference "BB (A) BB" 5-mer. The AUC value for the detection of m6A modification, with this dataset, was 0.68 (Supplementary Fig. 6). Even though a small AUC value suggests that the method still needs to be improved for accurate detection of m6A modification, the m6A modification score was mostly within the detection range in the known m6A motif (DRACH motif). From 3-mer analysis, we also noticed that G(m6A)C, C(m6A)G or G(m6A)U, U(m6A)G are prone to detection compering to other sequence combinations. (Supplementary Fig. 6 b) We also examined how the modification frequency correlates with the score, using the frequency data from a previous study [34] (Supplementary Fig. 7). The results showed no correlation (r^2 = 0.15) between the score and the previously reported modification frequency. The possible reason for this low correlation is that the magnitude of current disturbance is dependent on the background 5-mer sequence composition, which makes it difficult to estimate the modification frequency from the nanoDoc score alone. Thus, further classification and data accumulation are necessary to predict the modification frequency from the score.





Score level of different kind of PTMs

Figure 2

a) Evaluation of nanoDoc using ribosomal RNA of *E.coli* and *C.servisea*
The red line show the position of known modification sites in literature. The bule line show the score level of nanoDOC at each position

b) The ROC curve nonoDoc (Blue), KS statistics Stephenson et.al. (Green) and Tombo mean current difference(Orange) AUC values are 0.96,0.89,0.79 respectively

c) The score levels are summarized by each modification type, and baseline is with no modification data (left most). Percentile lines for base lines were drawn 75%,50%25% form the top respectively.
The right panel show the score distribution of modified (red) and unmodified (blue) sites .

Clustering analysis of 5-mers with multiple modifications

Multiple PTMs were observed within the test data set(7 kinds of 5-mers, Supplementary Table 1). We extracted the feature information from raw signals (Fig. 1e) using CNN trained by DOC. The features information (16-dimensional output vector) was subjected to k-means clustering (Fig. 3). The figure in the top row depicts the UMAP representation of the feature signals (Supplementary Method 3).

The reference IVT sequences were classified into different clusters in 4 out of 7 cases, and in the remaining cases, the reference IVT 5-mer sequences were mainly attributed to the same clusters. Furthermore, 10 out of 15 modified 5-mer sequences were assigned to a single cluster with 50% or more specificity. Considering that the modification ratio is not 100% at each site, the result suggested that modified sequences were generally separated into unique clusters by modification types. (Fig. 3 and Supplementary Table 1). In an exceptional case, one of the modified sequences was assigned to reference IVT sequence cluster instead of forming a unique cluster in the analysis of Am/$m^6{}_2$A, and m4Cm/Cm, indicating the difficulty of separating either of these modifications. With the "ACUAU" 5-mer, we tried to separate 3 modifications and corresponding IVT sequences. (Fig. 3c). For the *S. cerevisiae* position 2345, three modifications of Um/Y/Ym have previously been reported[35]. However, the signal from that position hardly clustered with the Y modification cluster (position 2258), indicating that the Y modification at position 2345 in this data, if it exists, is very small in this sample.

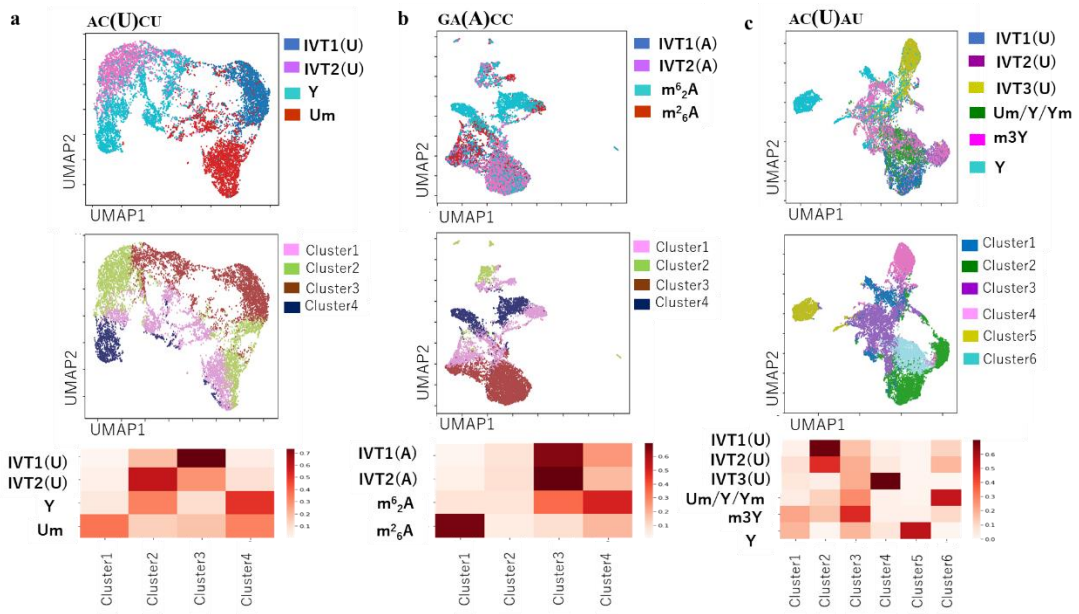


Figure 3
Separation of 5-mer sequence with multiple kinds of modified bases.(a~c)
UPPER row) UMAP representation of modified 5mer and unmodified IVT reference sequence.
Middle row) K-mean clustering result mapped onto UMAP representation
Bottom row) K-mean result for each labeled data set

Analysis of a severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) dataset

We analyzed SARS-CoV-2 DRS data of three different groups from Korea [27], Australia [36], and England [37] . We utilized the same procedure as described above. Kim et al. had prepared the IVT sequence, so we used the Korean IVT sequence as a reference to detect modifications. Ninety-one percent of reads were mapped to the reference in the IVT dataset, and the 29%–64% native reads were mapped to the reference (Supplementary Table 4). We observed a score draft with noise near the end of the IVT transcript. To reduce the error, we applied a window function for denoising. (Supplementary Method 4). The results of the overall scores are shown in Fig. 3a. All single nucleotide polymorphism sites in the English and Australian samples, against the Korean reference, were detected with scores from 0.03 to 1 (Supplementary Table 4). The other major peaks indicated the existence of PTMs. Many of the modifications have been previously reported in the 3' terminal region centering on the N protein region (Supplementary Fig. 10) [38][27]. We also conducted WEBLOGO [39] analysis with the nucleotide sequences of candidate sites and observed the weak trends of G/A/U enrichment at the target site and U/C enrichment in the +2 position of candidate sites. A score threshold of 0.05 was used, which corresponded to a false positive rate of 0.1 in the variation dataset, accounting for 117 candidates in total (Supplementary Table 5). More than half of the candidate sites from the three groups overlapped; however, 24, 13, and 21

candidates in the Korean, English, and Australian group, respectively, remained as unique candidates (Fig. 4b). In addition, 12 candidates overlapped with Kim et al. (Fig. 4c) with a match allowing for a ± 2 difference in genomic position; however, we observed that the majority of sites did not match because of using different software. We also compared the detected peaks to the short read RNA editing sites reported previously [40] (Supplementary Fig. 9) and observed almost no overlap between these two datasets. The RNA editing event, reported by a previous study that indicated very low allele frequency (median variant allele frequency = 0.009) on the SARS-CoV-2 transcriptome, might have been dismissed by this method. Further investigation, to identify each type of modification in the SARS-CoV-2 genome detected in this study, is ongoing.

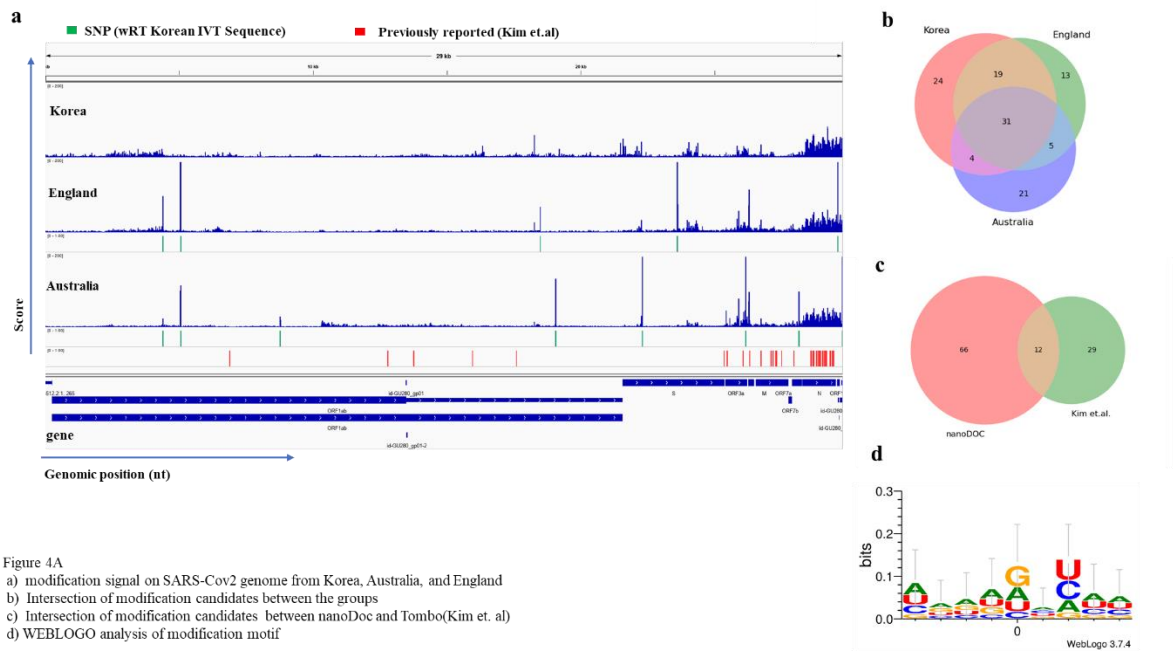


Figure 4A
a) modification signal on SARS-Cov2 genome from Korea, Australia, and England
b) Intersection of modification candidates between the groups
c) Intersection of modification candidates between nanoDoc and Tombo(Kim et. al)
d) WEBLOGO analysis of modification motif

Availability and Future Directions

The source code and training weight used in this study have been deposited at https://github.com/uedaLabR/nanoDoc.

The future direction of developing this software is to accumulate the data of different modifications for each 5-mer sequence and identify the different modification types. In addition, PTM detection without the IVT reference, as well as speed up and distributed commutation, are necessary to acquire an epitranscriptome map using this method on larger genomic size.

Data Availability

The ribosomal RNA data were taken from a previous study [20]. The INT/m6A modified sequence was also taken from past work [14], as were the SARS-CoV-2 DRS data for the three groups [27],[37],[38]. Davidson et al. kindly shared the raw data from the DRS sample. The reference sequence was acquired from the National Center for Biotechnology Information (Bethesda, MD, USA) (Supplementary Table 2). The data supporting the findings of this study are available within the article [and/or] its supplementary materials.

Acknowledgements

This work used the computational resources of AI Bridging Cloud Infrastructure (ABCI) provided by the National Institute of Advanced Industrial Science and Technology (AIST) through the High-Performance Computing Infrastructure (HPCI) research project (project ID: hp200145). A research and development grant from the Shimadzu Science Foundation (Kyoto, Japan) was applied to this study.

I would like to thank Dr. Hiroyuki Aburatani, Dr. Genta Nagae (The University of Tokyo, RCAST), Dr. Fumiko Kawasaki (RIKEN Center for Advanced Intelligence Project). and Dr. Shunpei Okada (Tokyo University of Science, Research Institute for Biomedical Science) for productive discussions and helpful advice.

Reference

[1] Williams GD, Gokhale NS, Horner SM. Regulation of Viral Infection by the RNA Modification N6-Methyladenosine. Annu Rev Virol 2019. https://doi.org/10.1146/annurev-virology-092818-015559.

[2] Jia G, Fu Y, Zhao X, Dai Q, Zheng G, Yang Y, et al. N6-Methyladenosine in nuclear RNA is a major substrate of the obesity-associated FTO. Nat Chem Biol 2011. https://doi.org/10.1038/nchembio.687.

[3] Sakurai M, Yano T, Kawabata H, Ueda H, Suzuki T. Inosine cyanoethylation identifies A-to-I RNA editing sites in the human transcriptome. Nat Chem Biol 2010. https://doi.org/10.1038/nchembio.434.

[4] Sakurai M, Ueda H, Yano T, Okada S, Terajima H, Mitsuyama T, et al. A biochemical landscape of A-to-I RNA editing in the human brain transcriptome. Genome Res 2014. https://doi.org/10.1101/gr.162537.113.

[5] Schwartz S, Bernstein DA, Mumbach MR, Jovanovic M, Herbst RH, León-Ricardo BX, et al. Transcriptome-wide mapping reveals widespread dynamic-regulated pseudouridylation of ncRNA and mRNA. Cell 2014. https://doi.org/10.1016/j.cell.2014.08.028.

[6] Bakin A, Ofengand J. Four Newly Located Pseudouridylate Residues in Escherichia coli 23S Ribosomal RNA Are All at the Peptidyltransferase Center: Analysis by the Application of a New Sequencing Technique. Biochemistry 1993. https://doi.org/10.1021/bi00088a030.

[7] Suzuki T, Ueda H, Okada S, Sakurai M. Transcriptome-wide identification of adenosine-to-inosine editing using the ICE-seq method. Nat Protoc 2015. https://doi.org/10.1038/nprot.2015.037.

[8] Meyer KD, Saletore Y, Zumbo P, Elemento O, Mason CE, Jaffrey SR. Comprehensive analysis of mRNA methylation reveals enrichment in 3′ UTRs and near stop codons. Cell 2012. https://doi.org/10.1016/j.cell.2012.05.003.

[9] Dominissini D, Moshitch-Moshkovitz S, Schwartz S, Salmon-Divon M, Ungar L, Osenberg S, et al. Topology of the human and mouse m6A RNA methylomes revealed by m6A-seq. Nature 2012. https://doi.org/10.1038/nature11112.

[10] Smith AM, Jain M, Mulroney L, Garalde DR, Akeson M. Reading canonical and modified nucleobases in 16S ribosomal RNA using nanopore native RNA sequencing. PLoS One 2019. https://doi.org/10.1371/journal.pone.0216709.

[11] Workman RE, Tang AD, Tang PS, Jain M, Tyson JR, Razaghi R, et al. Nanopore native RNA sequencing of a human poly(A) transcriptome. Nat Methods 2019. https://doi.org/10.1038/s41592-019-

0617-2.

[12] Ni P, Huang N, Zhang Z, Wang DP, Liang F, Miao Y, et al. DeepSignal: Detecting DNA methylation state from Nanopore sequencing reads using deep-learning. Bioinformatics 2019. https://doi.org/10.1093/bioinformatics/btz276.

[13] Liu Q, Fang L, Yu G, Wang D, Xiao C Le, Wang K. Detection of DNA base modifications by deep recurrent neural network on Oxford Nanopore sequencing data. Nat Commun 2019. https://doi.org/10.1038/s41467-019-10168-2.

[14] Liu H, Begik O, Lucas MC, Ramirez JM, Mason CE, Wiener D, et al. Accurate detection of m6A RNA modifications in native RNA sequences. Nat Commun 2019. https://doi.org/10.1038/s41467-019-11713-9.

[15] Jenjaroenpun P, Wongsurawat T, Wadley TD, Wassenaar TM, Liu J, Dai Q, et al. Decoding the epitranscriptional landscape from native RNA sequences. Nucleic Acids Res 2020. https://doi.org/10.1093/nar/gkaa620.

[16] Loman NJ, Quick J, Simpson JT. A complete bacterial genome assembled de novo using only nanopore sequencing data. Nat Methods 2015. https://doi.org/10.1038/nmeth.3444.

[17] Stoiber M, Quick J, Egan R, Eun Lee J, Celniker S, Neely R, et al. De novo Identification of DNA Modifications Enabled by Genome-Guided Nanopore Signal Processing 2016:094672. https://doi.org/10.1101/094672.

[18] GitHub - nanoporetech/tombo: Tombo is a suite of tools primarily for the identification of modified nucleotides from raw nanopore sequencing data. n.d. https://github.com/nanoporetech/tombo (accessed August 31, 2020).

[19] GitHub - nanoporetech/taiyaki: Training models for basecalling Oxford Nanopore reads n.d. https://github.com/nanoporetech/taiyaki (accessed August 31, 2020).

[20] Stephenson W, Razaghi R, Busan S, Weeks KM, Timp W, Smibert P. Direct detection of RNA modifications and structure using single molecule nanopore sequencing. BioRxiv 2020. https://doi.org/10.1101/2020.05.31.126763.

[21] Perera P, Patel VM. Learning Deep Features for One-Class Classification. IEEE Trans Image Process 2019. https://doi.org/10.1109/TIP.2019.2917862.

[22] Wick RR, Judd LM, Holt KE. Deepbinner: Demultiplexing barcoded Oxford Nanopore reads with deep

convolutional neural networks. PLoS Comput Biol 2018. https://doi.org/10.1371/journal.pcbi.1006583.

[23] GitHub - nanoporetech/rerio: Research release basecalling models and configurations n.d. https://github.com/nanoporetech/rerio (accessed August 31, 2020).

[24] GitHub - nanoporetech/ont_fast5_api: Oxford Nanopore Technologies fast5 API software n.d. https://github.com/nanoporetech/ont_fast5_api (accessed August 31, 2020).

[25] Li H. Minimap2: Pairwise alignment for nucleotide sequences. Bioinformatics 2018. https://doi.org/10.1093/bioinformatics/bty191.

[26] Apache Parquet n.d. https://parquet.apache.org/ (accessed September 2, 2020).

[27] Kim D, Lee JY, Yang JS, Kim JW, Kim VN, Chang H. The Architecture of SARS-CoV-2 Transcriptome. Cell 2020. https://doi.org/10.1016/j.cell.2020.04.011.

[28] RStudio. Keras. R Cheat Sheet 2017.

[29] Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, et al. TensorFlow: A system for large-scale machine learning. Proc. 12th USENIX Symp. Oper. Syst. Des. Implementation, OSDI 2016, 2016.

[30] Iandola FN, Moskewicz MW, Ashraf K, Han S, Dally WJ, Keutzer K. SqueezeNet. ArXiv 2016.

[31] Vaidya PM. An O(n log n) algorithm for the all-nearest-neighbors Problem. Discrete Comput Geom 1989. https://doi.org/10.1007/BF02187718.

[32] Johnson J, Douze M, Jegou H. Billion-scale similarity search with GPUs. IEEE Trans Big Data 2019. https://doi.org/10.1109/tbdata.2019.2921572.

[33] McInnes L, Healy J, Saul N, Großberger L. UMAP: Uniform Manifold Approximation and Projection. J Open Source Softw 2018. https://doi.org/10.21105/joss.00861.

[34] Yang J, Sharma S, Watzinger P, Hartmann JD, Kötter P, Entian KD. Mapping of complete set of ribose and base modifications of yeast rRNA by RP-HPLC and mung bean nuclease assay. PLoS One 2016. https://doi.org/10.1371/journal.pone.0168873.

[35] Sloan KE, Warda AS, Sharma S, Entian KD, Lafontaine DLJ, Bohnsack MT. Tuning the ribosome: The influence of rRNA modification on eukaryotic ribosome biogenesis and function. RNA Biol 2017. https://doi.org/10.1080/15476286.2016.1259781.

[36] Taiaroa G, Rawlinson D, Featherstone L, Pitt M, Caly L, Druce J, et al. Direct RNA sequencing and early evolution of SARS-CoV-2 2020. https://doi.org/10.1101/2020.03.05.976167.

[37] Davidson AD, Williamson MK, Lewis S, Shoemark D, Carroll MW, Heesom KJ, et al. Characterisation

of the transcriptome and proteome of SARS-CoV-2 reveals a cell passage induced in-frame deletion of the furin-like cleavage site from the spike glycoprotein. Genome Med 2020. https://doi.org/10.1186/s13073-020-00763-0.

[38] Viehweger A, Krautwurst S, Lamkiewicz K, Madhugiri R, Ziebuhr J, Hölzer M, et al. Direct RNA nanopore sequencing of full-length coronavirus genomes provides novel insights into structural variants and enables modification analysis. Genome Res 2019. https://doi.org/10.1101/gr.247064.118.

[39] Crooks GE, Hon G, Chandonia JM, Brenner SE. WebLogo: A sequence logo generator. Genome Res 2004. https://doi.org/10.1101/gr.849004.

[40] Giorgio S Di, Martignano F, Torcia MG, Mattiuz G, Conticello SG. Evidence for host-dependent RNA editing in the transcriptome of SARS-CoV-2. Sci Adv 2020. https://doi.org/10.1126/sciadv.abb5813.

Supporting Information

Supplemental methods

1. Scoring Schema

The extracted feature of IVT1 was compared to those of the IVT2 and Native signals, respectively, and the closest Euclidian distance in the target data set was calculated for each entry using ANN. For the distance calculation, the 16-dimensional vector output from the CNN was used (Supplemental Fig. 5a). Each distance was summarized as a distribution, and with the existence of the modified base, the distance was larger in the IVT1 vs. Native comparison (Supplementary Fig. 5b). The distance distribution was plotted as a cumulative distribution function. The blue, orange, and green line shows the distance function from the IVT1 vs. IVT2, IVT1 vs. Native, and Native vs. IVT1 (reverse direction) comparison, respectively. The surrounding area represents the disturbed signal feature from the IVT reference sequence. The score is calculated from this area and normalized by sample size (Supplementary Fig. 5c).

2. ROC analysis

For all genomic positions, each position was labeled as either True or False. The known modification

position in the rRNA dataset and the ±1 positions were assigned as True and False elsewhere. To avoid an unequal degree of bias for ROC analysis, the regions with a 50nt margin, from the first modification site to the last modification site, were used.

3. 5-mer modification analysis

The 5-mer signals were extracted from the Native and corresponding reference IVT sequences. Each signal was subjected to a CNN, and the output signals were further analyzed by k-means clustering. Faiss [32] was used for k-means clustering, with a cluster size of 4 to 6 and 500 iterations. UMAP was used for visualization purposes, which reduced the dimensions from 16 to 2.

4. Window function

The window function was applied for denoising.

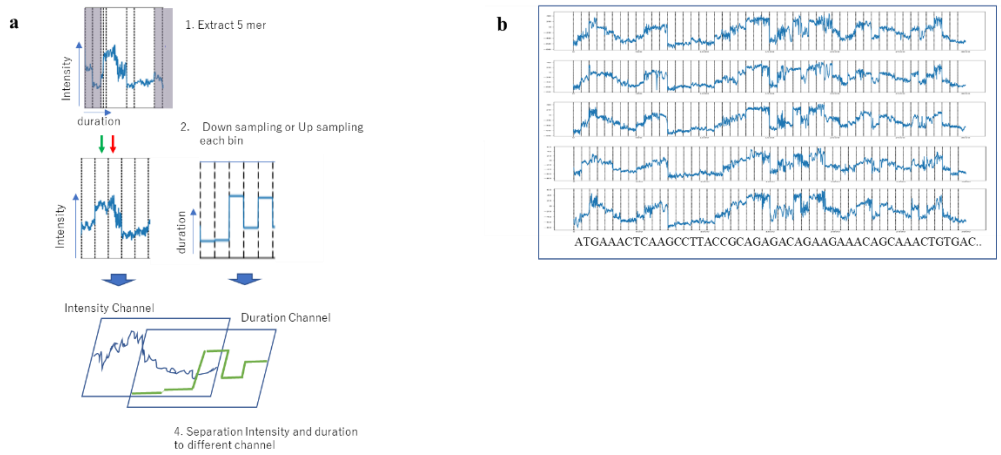
For each point, the following functions were applied:

Let n be the arbitral genomic position and Vn be a score at position n,

Vbaseline = Max ((Vn-4, Vn-5)/2, (Vn+4, Vn+5)/2) is to be used as the base line

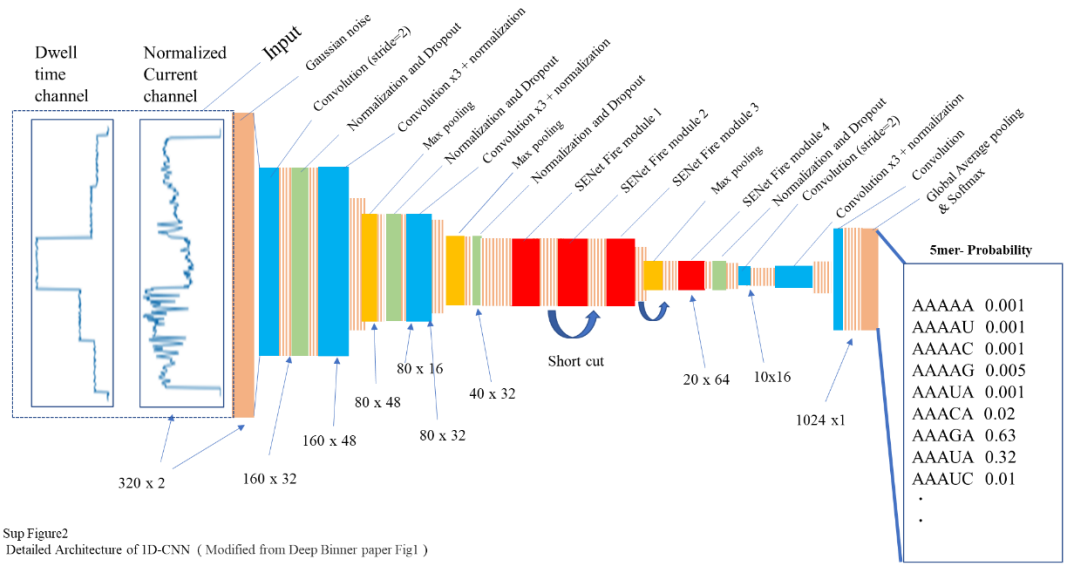
Vfilter = Vn – Vbaseline (where, n-3<= x <=n+3)

Supplemental figure 1



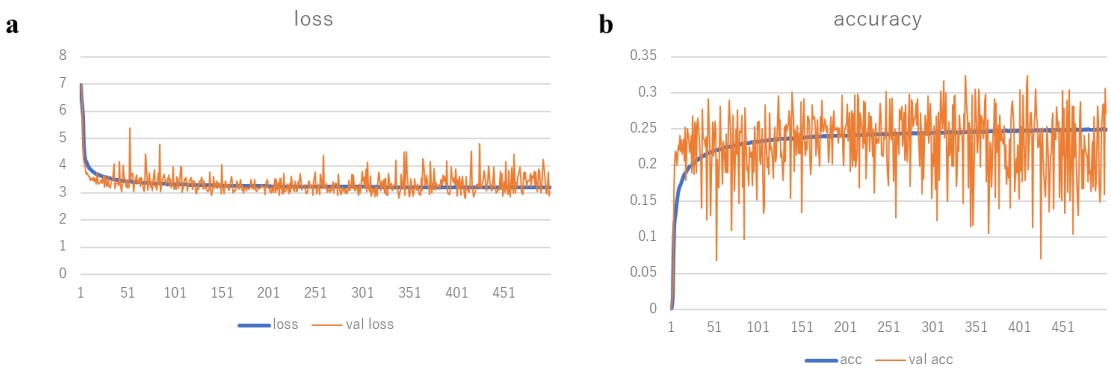
Sup Figure 1
a) schematic picture of signal uniformization and intensity-duration separation
b) Representation of equaled bin sized signal

Supplemental figure 2



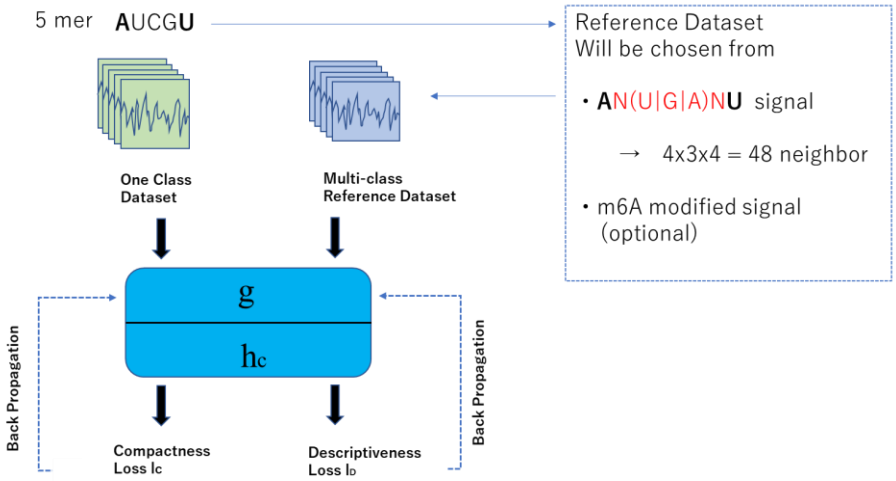
Sup Figure2
Detailed Architecture of 1D-CNN (Modified from Deep Binner paper Fig1)

Supplemental figure 3



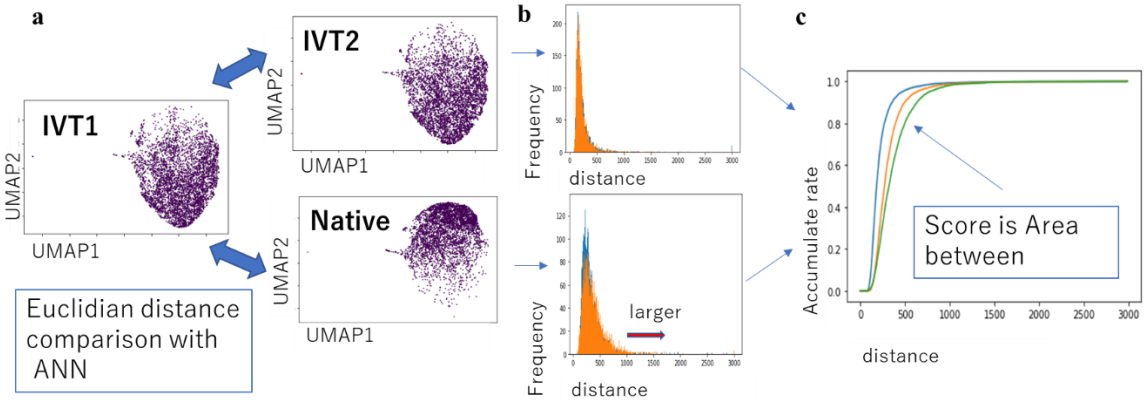
Sup Figure 3
a) Loss of Initial 1024, 5-mer training of 1D-CNN. Bule line shows the loss values and orange line shows validation loss
b) Accuracy of Initial 1024, 5-mer training of 1D-CNN. Bule line shows the accuracy values and orange line shows validation accuracy

Supplemental figure 4



Sup Figure 4
Target 5mer Dataset and reference Dataset are input to Deep One Class model for training. Compactness loss is calculated and used in back propagation. Likewise, Descriptive loss is used for reference network. The reference dataset is chosen with neighboring 5mer where nucleotide at 1st and 5th position is same as target and other nucleotide base are different from target 5mer. (Figure is modified from DOC paper fig 3.)

Supplemental figure 5

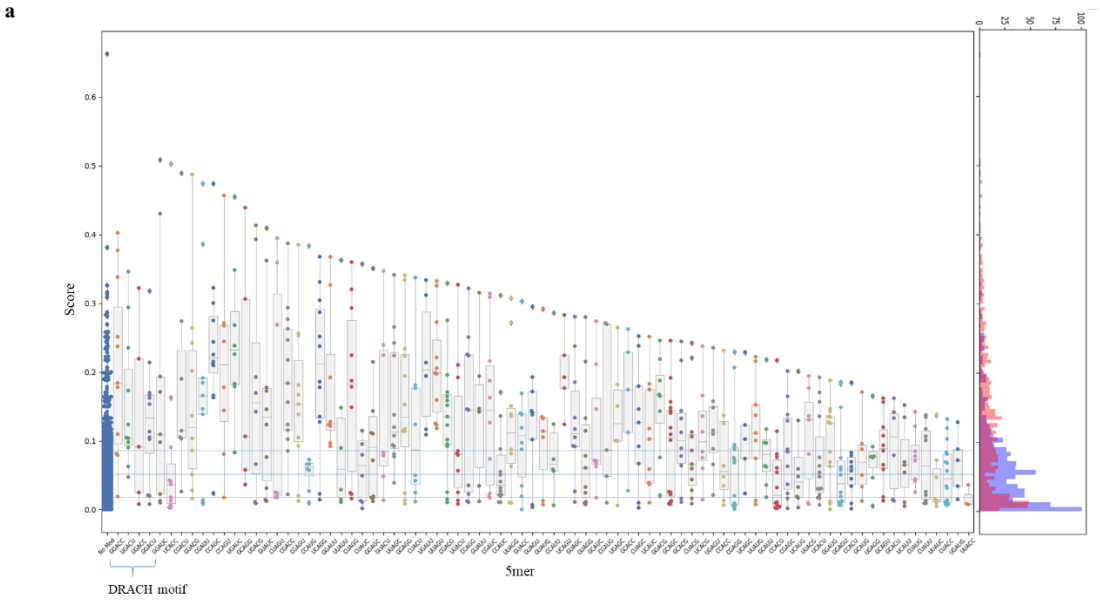


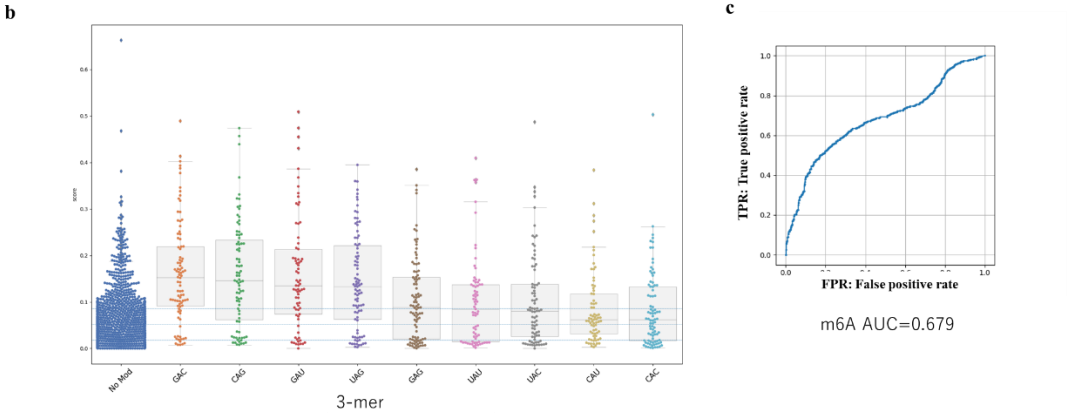
Scoring schema after feature extraction (Figs are taken from UCGAC vs UCG(m6A)C dataset)

a) The extracted feature of IVT1 is compared to IVT2 and Native signal respectively, and the closest Euclidian distance in the target data set is calculated for each entry using ANN. For the distance calculation 16 dimensional vector output from CNN is used.
b) Each distance is summarized as a distribution. With existence of a modification, the distance is larger in the IVT1 vs NATIVE comparison.

c) The distance distribution is plotted as a cumulative distribution function. The blue line shows the distance function from IVT1 vs IVT2 comparison, The orange line shows IVT1 vs NATIVE comparison, The green lines show NATIVE vs INT1 (reverse direction)
Surrounded area reflect the disturbed signal feature from IVT reference sequence. The score is calculated by normalizing this area to the sample size.

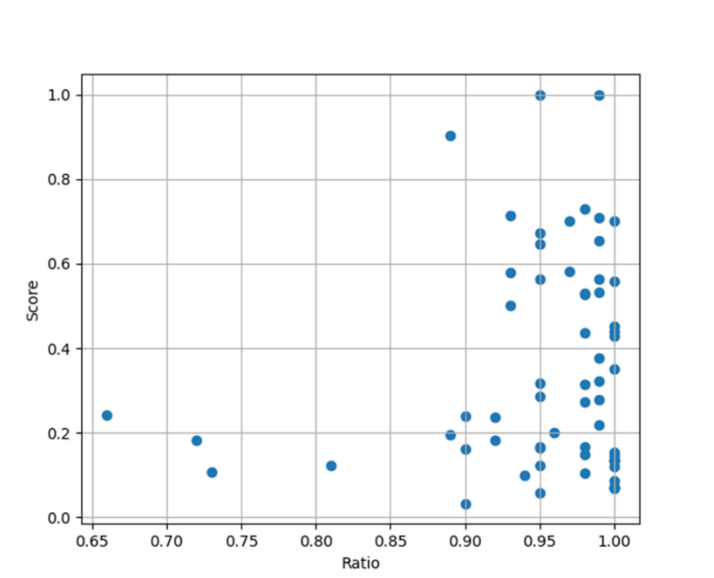
Supplemental figure 6





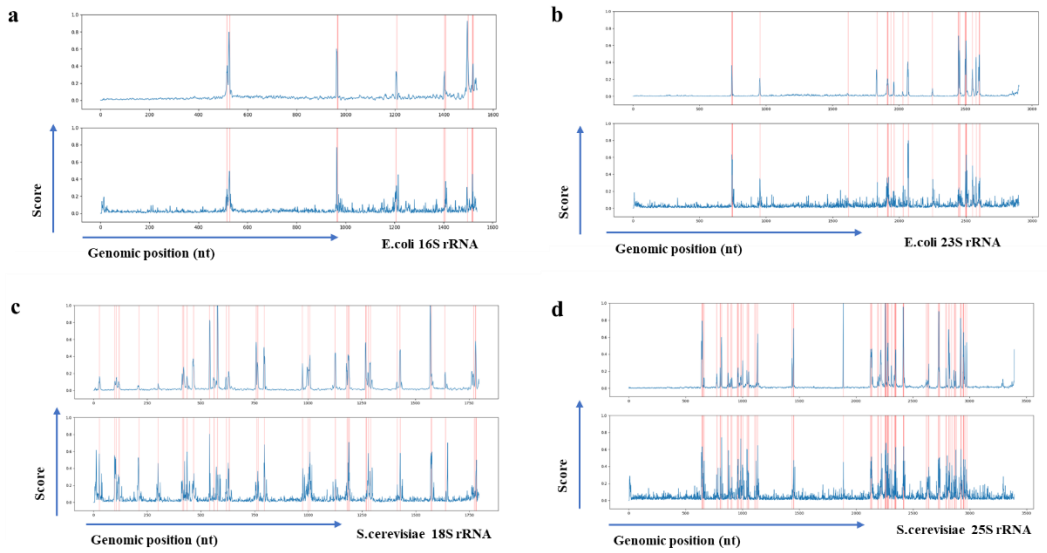
a)m6A score summarized by each 5-mer type and baseline with no modification (left most). Percentile line for baseline data was drawn for 75%, 50% and 25% from the top. 5-mer. 5-mer having DRACH motif is shown on the left
b)m6A score summarized by each 3-mer. G(m6A)C, C(m6A)G or G(m6A)U, U(m6A)G are prone to detect.
c)ROC curve of m6A detection using artificially modified dataset. AUC values for nanoDoc, KSStats, and Tombo mean current difference are 0.96, 0.89, and 0.79 respectively.

Supplemental figure 7



Score vs modification ratio reported by previous study. (33)

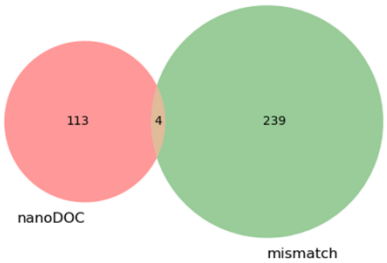
Supplemental figure 8



Sup Figure 5 Score comparison of nanoDoc (upper panel) to KS statistics Stephenson et.al. (lower panel)
a) E.coli 16S rRNA b) E.coli 23S rRNA
c) S.cerevisiae 18S rRNA d) S.cerevisiae 25S rRNA respectively

415

Supplemental figure 9

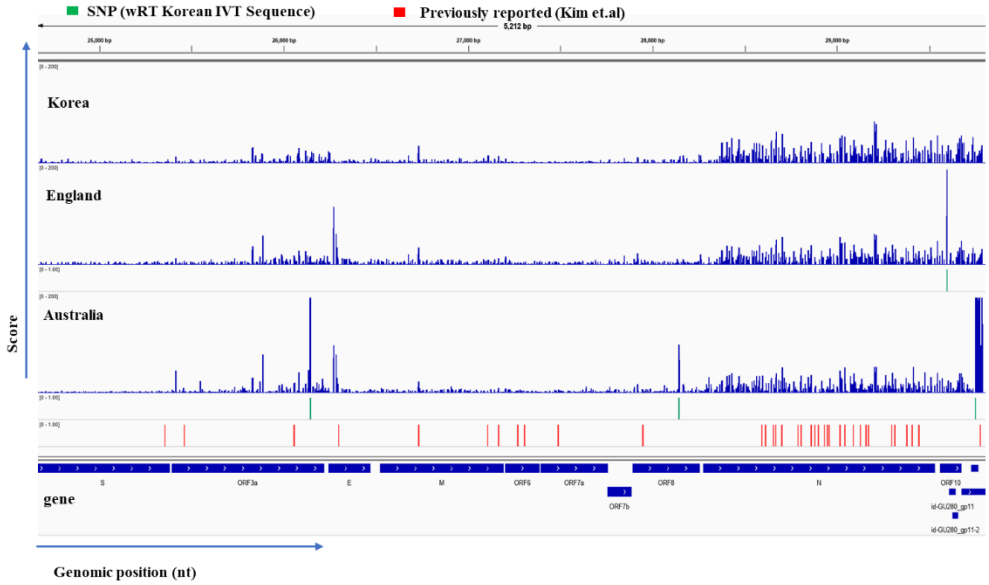


Position	ref alt	allele variant frequency
24369	AG	0.0083
25884	TC	0.0357
28549	AG	0.0089
28627	AG	0.0384
29029	TC	0.0144
29280	CT	0.0166

Overlap with this study and single nucleotide mismatch recurrently observed in short reads analysis reported by (29)

416

Supplemental figure 10



Sup Figure 8 RNA modification cluster on N protein region of SERS-CoV2 genome

417

418

Supplemental Table 1

5-mer with multiple modification and clustering result

5mer	modifications		
CGCCC	m4Cm	Cm	
	(ec16s 1402)	(sc18s 1639)	
CGUAA	m3U	Y	
	(ec16s 1498)	(ec23s 1911)	
GGAAC	m62A	Am	
	(ec16s 1518)	(sc18s 541)	
GAACC	m62A	m26A	
	(ec16s 1519)	(sc18s 1782)	
UUAAA	Am	m1A	
	(sc18s 619)	(sc25s 2142)	
ACUAU	Um/Y/Ym	m3Y	Y
	(sc25s 2347)	(ec23s 1915)	(sc25s 2258)
ACUCU	Y	Um	
	(sc25s 2264)	(sc25s 2421)	

ec16s: *E.coli* 16S rRNA
ec23s: *E.coli* 23S rRNA
sc18s: *S.cerevisiae* 18S rRNA
sc25s: *S.cerevisiae* 25S rRNA

CGCCC	cluster1	cluster2	cluster3	cluster4
IVT1	0.019	0.079	0.273333333	0.628667
IVT2	0.135	0.049	0.404333333	0.411667
m4Cm	0.324	0.122333333	0.218333333	0.335333
Cm	0.532	0.149	0.242333333	0.076667

CGUAA	cluster1	cluster2	cluster3	cluster4
IVT1	0.023666667	0.189333333	0.328	0.459
IVT2	0.160666667	0.554666667	0.015	0.269667
m3U	0.092333333	0.077333333	0.514333333	0.316
Y	0.265333333	0.175	0.012333333	0.547333

GGAAC	cluster1	cluster2	cluster3	cluster4
IVT1	0.087	0.600333333	0.043666667	0.269
IVT2	0.089666667	0.332333333	0.008333333	0.569667
m62A	0.097	0.324666667	0.048666667	0.529667
Am	0.134	0.002666667	0.836666667	0.026667

GAACC	cluster1	cluster2	cluster3	cluster4
IVT1	0.008	0.082666667	0.654666667	0.254667
IVT2	0.032	0.091666667	0.695	0.181333
m62A	0.076	0.076	0.350666667	0.497333
m26A	0.671666667	0.045666667	0.095666667	0.187

UUAAA	cluster1	cluster2	cluster3	cluster4
IVT1	0.011333333	0.035333333	0.530666667	0.422667
IVT2	0.018333333	0.698333333	0.083	0.200333
Am	0.011	0.02	0.250333333	0.718667
m1A	0.392666667	0.223	0.080333333	0.304

ACUCU	cluster1	cluster2	cluster3	cluster4
IVT1	0.002333333	0.187666667	0.768	0.042
IVT2	0.022666667	0.597	0.287333333	0.093
Y	0.057333333	0.331	0.105333333	0.506333
Um	0.359333333	0.140666667	0.173666667	0.326333

ACUAU	cluster1	cluster2	cluster3	cluster4	cluster5	cluster6
IVT1	0.017666667	0.688666667	0.158666667	0.02	0	0.115
IVT2	0.083666667	0.476666667	0.204333333	0.050667	0.002	0.182667
IVT3	0.061	0.029333333	0.206	0.697667	0.005	0.001
Um/Y/Ym	0.036333333	0.121	0.277333333	0.018333	0.008	0.539
m3Y	0.221666667	0.157333333	0.479666667	0.02	0.02	0.101333
Y	0.187666667	0.002666667	0.187666667	0.064667	0.555333	0.002

431

Supplemental Table 2

432 Reference file used in this study

E. coli (NC_000913.3.fa)
S. cerevisiae (R1-1-1_19960731.fsa)
Covid19 sequences are downloaded from GISAID(https://www.gisaid.org/) database
Korea: SERS-COV2 isolate SNU01/
England: SERS-COV2 BetaCoV/England/02/2020
Australia: SERS-COV2 Australia/VIC01/2020
Curlile transcript GEO database under the accession code GSE124309

Supplemental Table 3

SERS-COV2 DRS sequencing from 3 groups

	Sequence ID	Virus	kit	size	Mapped reads /number of read	Tombo mapped %
Kim et al	MT039890.1	SERS-COV2 isolate SNU01/	SQK-RNA002 R9.4.1	730GB	1,456,249 / 1,593,624 (in vitro) 570,916/879,679 (Infected)	91.30% 64.90%
Taiaroa et al	MT007544.1	SERS-COV2 Australia/VIC01/2020	SQK-RNA002 R9.4.1	130GB	198,765 / 680,347 (Infected)	29.20%
Davidson et al	EPI_ISL_407073 (GISAID)	SERS-COV2 BetaCoV/England/02/ 2020	SQK-RNA002 MIN106D R9 R9.4	160GB	496,027 / 1,520,319 (Infected)	32.60%

433

Supplemental Table 4

Score distribution at the position of SNP

position	England	Austraria	ref	alt
4402	0.15	0.03	C	T
5062	1	0.23	T	G
8782	-	0.057	T	C
18488	0.13	-	T	C
19065	-	0.27	T	C
22303		0.85	T	G
23605	0.73	-	T	G
26114	-	1	G	T
28144	-	0.19	T	C
29596	1	-	A	G
29750	-	0.55	A	G

434

Supplemental Table 5

435

436 The candidate modification sites on SERS-COV2 genome detected by nanoDOC

Genomic position	reference base	Korea(Score)	England(Score)	Australia(Score)
4404	A	0	0.173	0.045
5064	U	0	0.372	0.08
16325	U	0.067	0.008	0.003
18254	G	0.127	0.055	0.029
19067	A	0.003	0	0.108
21479	U	0.081	0.002	0.001
21573	U	0.106	0.001	0.003
21616	A	0.063	0	0
21658	A	0.066	0.001	0
22088	U	0.066	0.009	0.001
22257	U	0.091	0.071	0.064
22305	U	0	0	0.577
23598	U	0	0.101	0.006
23607	G	0	0.693	0
23627	G	0	0.158	0
23969	C	0.034	0.07	0.028
24029	U	0.086	0.058	0.025
24309	C	0.063	0.068	0.014
24370	U	0.083	0.014	0.019
25416	A	0.024	0.027	0.091
25830	C	0.066	0.062	0.065
25831	U	0.068	0.077	0.065
25887	U	0	0.12	0.162
26083	U	0.049	0.035	0.088
26084	C	0.064	0.059	0.056
26119	A	0.053	0.054	0.034
26136	A	0.005	0	0.097
26146	C	0	0.015	1
26211	C	0.011	0.015	0.058
26241	C	0.051	0.01	0.022
26271	G	0.007	0.245	0.202

26285	U	0	0.131	0.163
26730	U	0.073	0.074	0.047
28307	C	0.007	0.009	0.066
28375	G	0.087	0.06	0.049
28376	C	0.089	0.059	0.054
28404	U	0.089	0.078	0.071
28415	A	0.068	0.06	0.076
28429	U	0.105	0.077	0.081
28469	U	0.101	0.073	0.07
28494	U	0.056	0.039	0.043
28541	C	0.058	0.062	0.049
28549	C	0.085	0.054	0.045
28550	G	0.076	0.053	0.052
28562	G	0.084	0.058	0.069
28575	G	0.047	0.019	0.052
28581	U	0.081	0.082	0.07
28596	G	0.076	0.051	0.072
28614	A	0.063	0.054	0.04
28644	U	0.09	0.073	0.064
28645	G	0.095	0.074	0.061
28659	C	0.06	0.036	0.03
28671	U	0.134	0.104	0.082
28685	C	0.068	0.05	0.057
28704	U	0.126	0.116	0.106
28713	C	0.068	0.054	0.063
28764	A	0.064	0.051	0.03
28765	A	0.059	0.057	0.046
28803	A	0.072	0.069	0.073
28827	U	0.053	0.041	0.04
28844	G	0.065	0.06	0.05
28845	C	0.057	0.059	0.066
28859	A	0.09	0.082	0.075
28916	G	0.084	0.059	0.048
28919	A	0.049	0.032	0.052
28960	C	0.078	0.066	0.062
28961	U	0.074	0.067	0.054

29017	A	0.104	0.116	0.094
29027	C	0.117	0.089	0.058
29044	C	0.113	0.092	0.106
29072	C	0.075	0.061	0.069
29090	C	0.1	0.083	0.08
29099	G	0.06	0.052	0.048
29100	A	0.069	0.047	0.04
29135	A	0.078	0.067	0.058
29169	A	0.065	0.044	0.03
29191	U	0.075	0.034	0.035
29202	C	0.175	0.129	0.106
29203	G	0.169	0.123	0.111
29211	G	0.164	0.123	0.111
29222	U	0.061	0.043	0.033
29256	A	0.055	0.046	0.033
29265	G	0.099	0.072	0.077
29279	C	0.064	0.056	0.048
29300	A	0.087	0.084	0.084
29301	U	0.086	0.084	0.075
29315	A	0.062	0.064	0.058
29374	C	0.094	0.069	0.064
29385	C	0.053	0.057	0.045
29414	C	0.106	0.076	0.098
29415	C	0.101	0.078	0.109
29427	A	0.062	0.047	0.034
29442	A	0.05	0.056	0.044
29462	C	0.057	0.031	0.03
29477	A	0.069	0.06	0.087
29493	G	0.057	0.042	0.031
29510	G	0.095	0.074	0.067
29543	A	0.117	0.099	0.085
29557	A	0.039	0.037	0.052
29572	G	0.076	0.058	0.062
29594	U	0.04	0.204	0.029
29598	U	0	0.138	0
29602	C	0.076	0	0.052

29616	A	0.056	0.049	0.044
29654	U	0.085	0.055	0.052
29664	U	0.105	0.072	0.081
29676	A	0.116	0.099	0.083
29687	G	0.086	0.063	0.046
29736	G	0.103	0.071	0.013
29737	C	0.11	0.07	0.02
29762	A	0	0	0.39
29772	G	0.006	0	0.38
29773	C	0.054	0.031	0.12
29779	A	0	0	0.2
29785	G	0.076	0.065	0
29786	C	0.075	0.036	0.26
29790	U	0	0	0.29